CGI Advantage® 4

Accounts Payable Run Sheets Guide

Table of Contents

1 Purpose of the System Administration Guide

This manual is intended to help system administrators initiate, configure, monitor, and control all processing for CGI Advantage. The manual has five parts:

- The CGI Advantage System Administration Guide contains information about the CGI Advantage system architecture, and configuration (including the embedded third party components), post-installation setup, security configuration and considerations, workflow, job framework and its usage/maintenance, and other information pertinent to administering the application.
- The CGI Advantage HRM run sheet guides describe each process of CGI Advantage HRM in detail with its input, output, parameters, sort sequence, and selection criteria.
- The CGI Advantage Financial run sheet guides describe each process of CGI Advantage Financial in detail with its input, output, parameters, sort sequence, and selection criteria.
- The CGI Advantage HRM Payroll Engine System Administration Guide describes the system control tables and utilities for CGI Advantage HRM.
- The CGI Advantage VSS System Administration Guide describes each VSS process in detail with its input, output, parameters, sort sequence, and selection criteria.
- System administration tasks include setting up and maintaining application security, querying and viewing the application status through logs and reports, managing workflow, setting up and maintaining system tables, and other critical application maintenance tasks.

1.1 Common terms and glossary used

The terms "Job" and "Batch" have been used interchangeably throughout the document. Please note that the CGI Advantage technical architecture is flexible enough to support the execution of jobs/batch processes while the application is available for online usage. In other words, the jobs/batch processes are technically not required to be "offline" processes.

2 Description of Processes

This chapter describes the processes in CGI Advantage that are considered system administration processes. For each process, you see information on these topics:

- Description
- Steps to Run this Process (if applicable)
- When to Run
- Major Input
- Output
- Parameters – Batch and Custom
- Sort Sequence
- Selection Criteria
- Notes
- Problem Resolution

System Wide Batch Parameters:

System wide batch parameter fields are available with each batch program, which provide the path for the input/output directory. These parameters allow sites to easily and quickly update the path for individual batch processes.

System wide batch parameters can be defined at the System Level, Area Level, Chain Job level, Chain Level or Job level. There has to be a default value set for the system wide batch parameters at any of these levels mentioned above so that the process will generate, read or write the respective files from the given location.

System wide batch parameters are defined at the System Level on the System Level Process Parameters (BATSETUP) reference page, searching for the Catalog Label of *Batch Catalog* and then choosing the record-level action of *Edit*.

AMSROOT - Root directory of the batch files (for example, C:\AMSADV30\RTFiles)

AMSEXPORT - For files that are created by the program and need to remain after the job is completed (i.e. cannot be temporary files). This could include interface files that come from/go to third party sources (for example, $AMSROOT\ExportImport).

AMSIMPORT - For files that are used by the program and need to remain after the job is completed (that is, cannot be temporary files). This could include interface files that come from/go to third party sources (for example, $AMSROOT\ExportImport).

AMSLOGS - For batch framework log files. If the job requires its own log files, this is where it is put (for example, $AMSROOT\Logs).

AMSPARM - Batch job parameter files specific to a single job instance only (for example, $AMSROOT\Parms).

AMSTEMP - For temporary files, usually stamped with process ID (for example, C:\TEMP).

AMSSPOOL - Batch job report files, statistic files, exception reports, and so forth. These files may be sent to an OS print queue. File name is usually date and time stamped (for example, $AMSROOT\Spool).

Note:

Assumptions while implementing system wide batch parameters: It is assumed that wherever in the Job processes system wide batch parameter variables (that is, AMSEXPORT, AMSIMPORT, AMSROOT, AMSLOGS, AMSPARM, AMSTEMP, AMSSPOOL) are declared as input parameters, care should be taken to set the overrideable flag for that variable to *true*, otherwise the process may fail.

Pivot Date/Year Validation:

Note:

Assumption for date attributes: Set the Earliest Year (EARLIEST_YEAR) and Latest Year (LATEST_YEAR) on the Application Parameter reference page. When defining the year range, attention should be given to setting a range vast enough to accommodate all system impacts (such as imported transactions). The Job input date/year must lie between the above year range; otherwise, the process will fail.

2.1 Accounts Payable Batch and Chain Processes

Accounts Payable in Advantage Financial records and disburses payments for purchases of goods or services. Accounts Payable is the final step in the purchasing chain of events and includes the following processing:

Payments: Requests payment for goods received and services rendered. This process establishes a payable in the system and, if specified, liquidates the encumbrance.

Disbursements: Liquidates the payable and generates payment. The payment may be in the form of a check, a warrant, or an electronic funds transfer (EFT).

Disbursement Reconciliation: Stores information to reconcile Advantage Financial checks with bank information.

Check Cancellation: Cancels checks, warrants, and EFTs.

Special A/P processing:

- Procurement Card: Supports credit card purchases, typically for small dollar payments.
- 1099 Reporting: Collects 1099 data and generates 1099 forms for external vendors. This process also generates information to be sent to the IRS.
- Retainage: For specific contracts, A/P holds a percentage of each payment until the terms of the award have been completed. At that point, final payment is either released to the vendor or forfeited back to the client.

The processes that drive Accounts Payable in Advantage Financial are listed below in alphabetical order:

- Automated Disbursements Chain
- Automated Interest Calculation Process
- Automated Payment Hold
- Automatic EFT Reversal Confirmation
- Backup Withholding History
- Check Reconciliation
- Claim Schedule Numbering
- Claim Schedule Printing
- Consolidation By Payment Request Department
- Convert Bank File
- CTX EDI Formatting
- Disbursement Correction
- Disbursement Discard Chain
- Disbursement Parameters Updates
- Disbursement Printing
- Disbursement Request Initialization
- EF ACH Reversal
- EF ACH Transaction

- EFT/ACH Archive Process
- EFT/ACH File Reversal
- Electronic Payment Request Chain Job
- EWS Transaction
- Extended Payment Request Scheduling Batch Process
- Forms Table Load Chain
- Generate Positive Pay
- Generate Warrant Chain
- Initiate Backup Withholding
- IRS TIN/Name Match Extract
- IRS TIN/Name Match Update
- IRS Transmittal File Generation
- Journal Posting Initiator
- Load EDI Invoices
- Load Payment Hold by TIN
- Mass Cancellation
- Multi-Process Disbursement Printing
- Disbursement Printing for Multi-Process Disbursement Printing
- Multi Process Disb (Disbursement) Journal Posting
- Offline 1099 Process
- Payment Hold Archive Process
- PCard Chain
- Prenote Batch Process
- Print Backup Withholding Notices
- Print Payment Hold Notice
- Retainage Payout Process
- Retiree Billing
- Retiree Update
- Retiree VCUST Extract
- Returned ACH Transaction Process
- Scheduled Invoice Generation
- Select Payment Hold by TIN
- Stale Escheat Process
- Stale Process
- Escheat Pre-Selection Process

2.1.1 Automated Disbursements Chain

Job Name	AD Chain
Recommended Frequency	The AD Chain can be run daily as part of the nightly cycle or on demand. It can be run on demand for a small set of records by selecting the Online Disbursement indication for records on the Disbursement Parameter and Disbursement Request inquiry pages.
Single Instance Required	Yes
Can be restarted?	Yes, see the individual jobs for more details.
Reports generated	Yes, some of the jobs in the chain generate the exception report. Please refer to the individual jobs for more details.

Overview

The Automated Disbursement Chain (AD Chain) is a group of jobs that work together to create disbursement transactions from payment request transactions. The process selects authorized payments, edits payments for validity, processes payment adjustments, groups the payments, formats payments, and posts payment transactions. In short, it is the process that takes payment data and transforms this into a disbursement instrument (in the form of a check/warrant or EFT record). This process also generates the transactions to record the Intercept Transfers. The AD Chain has the following jobs (each of the jobs listed below, is described in subsequent sections):

1. AD XML Creation
2. System Hold Report
3. AD Upload
4. AD Submitter
5. AD Transaction Exception
6. AD Clean Up and IT XML Creation
7. Check Number Assignment
8. IT Upload
9. IT Submit

> *Note: Even though the above jobs in the chain can be run individually by disabling other jobs, it is recommended to always run the entire chain. The exception to this rule is the System Hold Report, which may be disabled with no negative impacts to the overall output of the chain.*

The acceptable job return code configuration depends on the business requirement. For example, if the requirement is that the subsequent jobs in the chain should continue only if the job ends with a return code of *Successful*, the Pre Condition Return Codes for all of the jobs should be set to *Successful*. If for some jobs in the chain, a *Non-Fatal Error* is an acceptable job return code, then that can also be configured. These configurations can be done in the Job setup page.

For baseline configuration, the System Hold Report will be kicked off when the AD XML Creation job ends with a return code of *Successful*, or *Non-Fatal Error*. The AD Upload job will then be kicked off only when the previous job ends with a return code of *Successful*.

If any of the jobs in the chain ends with a return code of Failed, Terminated or System Failure, all of the subsequent jobs will be set to Inactive.

Major Input

- Disbursement Parameter
- Disbursement Request
- Vendor/Customer
- System Wide Options
- Entity
- Bank
- Automated Transaction Numbering
- Intercept Request
- Intercept Options
- Multiple Intercept Request Configurations
- Transaction Component Requirements
- Application Parameter
- 1099 Processing Options and Controls
- Allowable Payments for Intercept by Debt Type

Major Output

- Disbursement Transactions
- Intercept Transfer Transactions
- AD Exception Report

Chain Job Return code

The following table shows the potential return codes for the AD chain job. Note that the Chain job will end with the highest return code across all of the jobs.

Return Code	Condition
Successful (1)	All of the jobs end successfully
Warning (4)	One of the jobs in the chain ends with a return code of *Warning*
Non-Fatal Error (8)	One of the jobs in the chain ends with a return code of *Non-Fatal Error*
Failed (12)	One of the jobs in the chain ends with a return code of *Failed*
Terminated (16)	One of the jobs in the chain ends with a return code of *Terminated*
System Failure (20)	One of the jobs in the chain ends with a return code of *System Failure*

Problem Resolution

If any of the jobs in the chain failed due to application errors it is advisable to restart the job after correcting the errors instead of rescheduling the job. Restarting the job will reduce the processing time since the job will resume from where it has last committed and select only the unprocessed records.

Please refer to the individual jobs for details regarding the specific job processes and problem resolution.

Automated Disbursement Chain: AD XML Creation Job

Job Name	AD XML Creation Job
Recommended Frequency	See AD Chain information
Single Instance Required	Yes
Can be restarted?	Yes. See the overview and the "Problem Resolution" section for more details.
Reports Generated	No. All the exceptions are only written to the log.

Overview

The AD XML Creation job in the AD chain selects the records from Disbursement Request (based on the criteria specified on the active records on the Disbursement Parameters reference page. The selected records will undergo various validations and calculations such as Vendor Editing, Discount, Penalty, Interest, Backup Withholding Calculations, Credit Memo Adjustments, and Payment Intercept Adjustments.

The job uses the following temporary tables to store the data for performing the validations and calculations. These temporary tables become input for the creation of the transaction xml files.

Disbursement Request Line (R_AP_DISB_RQST_LN)

Intercept Disbursement Request (INCT_DISB_RQST)

The AD XML Creation Job clears out data from the Disbursement Request Line table at the very beginning of the process so that no data exists on that table from a previous run. The Intercept Disbursement Request table is cleared by the corresponding Intercept Transfer transaction when processed to *Final*.

After performing all of the validations and calculations, this job creates the Disbursement transactions in the form of an XML file to be loaded into the transaction catalog. This job creates multiple Submit Parameter Files when the records are written into the XML file. These parameter files are created based on disbursement priorities and the number of processors available for running the AD chain job.

The grouping and consolidation logic for Credit Memo adjustment, Payment Intercept adjustment and the AD-XML depends upon the MAX_LINE_LIMIT property value of AD_DOC_VEND and AD_DOC_ACTG set on the Transaction Component Requirement reference page. If the Vendor Line Count or Accounting Line Count set is crossed then the remaining lines are created in the next transaction.

The following table shows the various steps that the AD XML Creation Job goes through and the messages issued at each step.

Process Steps	Messages
1. Parameter Validation	• Validating Batch Parameters • Parameters are valid or invalid depending on the Validation. If the parameter is invalid, the invalid value will be displayed in the log. • Batch Parameter validation completed.
2. Selection of Records	• Selecting eligible records from DISRQ based on the DISPA records. • If the selection returns 0 records, then the following message will be issued: “No eligible records found”. • Number of records (count) selected will be displayed. • Selection of records completed.
3. Vendor Editing & Amount Calculation	• Performing Vendor editing and Amount calculation • Number of records processed: ‘n’ • Number of records failed in the Vendor Active validation: ‘n’ • Number of records failed in the EFT validation: ‘n’ • Vendor Editing and Amount Calculation completed
4. Credit Memo Calculation	• Performing Credit Memo calculation • The following additional messages may be issued: • Number of credit memo lines selected for processing • Number of credit memo lines adjusted during that run • Number of payment records put on HOLD due to credit memo adjustment • Credit memo calculation completed
5. Splitting Credit Memo amount	• Splitting the credit memo amount • Splitting of credit memo amount completed
6. Disbursement Limit Check	• Performing Disbursement Limit check • The following additional messages will be issued: • Number of records passed Disbursement Limit check • Number of records put on HOLD due to Disbursement Limit check • Disbursement Limit check completed

Process Steps	Messages
7. Payment Intercept	• If the SOPT Intercept flag is *False*, the following message will be issued: Payment Intercept step bypassed since the SOPT option is false. • If the SOPT Intercept flag is *True*, the following messages will be issued: "Performing Payment Intercept". • The following additional messages will be issued: • Selecting eligible INTR records Records with the 'Eligible for Internal Intercept' flag set to *False* on the ENTY table for the combination of Entity & Debt Type on INTR table will not be eligible. Records with the 'Eligible for Internal Intercept' flag set to *True* on the ENTY table for the combination of Entity & Debt Type on INTR table will be eligible. • Selecting INTR records for miscellaneous vendor If the SOPT Intercept Miscellaneous Vendor Payments flag = True Records on DISRQ with 'MISC_ACCT_FL = True will be evaluated for intercept. • If the selection returns 0 records, then the following message will be issued: "No eligible records found – bypassing the Payment Intercept." • If the selected records are not equal to 0, then following message will be issued: "Eligible records found, processing the payment intercepts." • Count may be issued for the number of selected INTR records. • "Payment Intercept Completed" message will be issued when all the selected INTR records are intercepted.
8. AD XML creation	• Generating the AD XML file • AD XML file generated • Number of transactions included in that XML file may be provided. • Submit parameter file created

Restartability Information

If the job fails in any of the above steps the job can be restarted after resolving the error. If the job is restarted, it will start from the step where it failed earlier and won't start from the beginning. For example, the job failed in the step Disbursement Limit Check due to some fatal condition (table space error). If the job is restarted after resolving the table space issue, the job will start from the step Disbursement Limit check and not from the parameter validation step. Instead of restarting

the job, if the job is rescheduled with the same set of parameters after resolving the table space issue, the job will start from the beginning. There is no need to back out any updates when this job failed since this job updates only the temporary table and that gets cleared out when the job is rescheduled.

The restart will fail if another instance of this job has been scheduled and ran successfully before restarting the failed job. This is because the AD XML Creation job uses the temporary table for all processing and if another instance of the same job has been scheduled before restarting the failed job, the new job would have deleted the records from the temporary table related to the failed job. In that case, the new job should be scheduled.

Major Input

- Disbursement Parameters
- Disbursement Request
- Vendor/Customer
- System Wide Options
- Bank
- Entity
- Automated Transaction Numbering
- Intercept Request
- Transaction Component Requirements
- 1099 Processing Options and Controls
- Intercept Reference
    - Intercept Options
    - Intercept Fee
    - Intercept Disbursement Exception
    - Intercept Receivable Exception
    - Entity
    - Debt Type
    - Billing Profile
    - Contact
    - Allowable Payments for Intercept by Debt Type

Batch Parameters

Parameter	Description	Default Value
Action Code: Import (ACTN_CD)	Required. This field instructs the program to perform a certain action on the records. The only value that should be used is 171.	171

Parameter	Description	Default Value
Export Location at AD XML Creation Job (AMSEXPORT)	Required. The file location where output is exported.	No Default
Parameter Location at AD XML Creation Job (AMSPARM)	Required, The file location where the parameter file will be written.	No Default
Logs Location at AD XML Creation Job (AMSLOGS)	Required. The file location where output logs will be written.	No Default
Apply Overrides (APPLY_OVERRIDES)	Required. To apply overrides on transactions set to *True. False* will result in all override errors causing transactions to reject. The level of override applied will be that of the submitter user ID or the one specified in the Override Level parameter.	No Default
Offset Percentage (APPLY_OFFSET_PC)	Offset Percentages can be specified in two locations, read with the following logic to find a match with the highest rank. Should two matches have the same rank, the system will use the first it locates. • *True* - The system will verify INTDBEX first and then INTCLE for the matching record found. If a match is found on INTDBEX then INTCLE will not be read. • *False* - The system will verify INTDBEX only.	False
Bypass Auto Transaction Numbering (BYPS_ADNT_FL)	Required Field. Valid values are *True* or *False*.	
Chain Job ID (CHAIN_JOB_ID)	Required Field. This field would be used if we are running multiple instances of AD chain to differentiate all of the Job ID's for different chains run at the same time.	$$@CHAINJOBID@$$
Check Transaction Code (CHK_DOC_CD)	Required Field. This field refers to the Transaction that will be generated by the Automated Disbursement process.	AD

Parameter	Description	Default Value
Check Department Code (CHK_DOC_DEPT_CD)	Required Field. The transaction department to be used with the transaction code and prefix to find an Automatic Transaction Numbering entry. This will appear as the transaction department for all transactions created by the program.	No Default
Check Unit Code (CHK_DOC_UNIT_CD)	Required field if Require Transaction Unit Code flag is set to true for the transaction code on DCTRL Table, else it's an optional field. Provides security with a unit code enabling transaction access to be secured at a level below department.	No Default
Check Prefix (CHK_PFX)	Required Field. The transaction prefix to be used with the transaction code and department to find an Automatic Transaction Numbering entry. Will appear in the beginning of all transaction ID's for all transactions created by the program.	No Default
Commit Block Size (COMMIT_BLOCK)	Required field. Controls how many records are committed by the application at one time. The size should be compatible with technical capabilities and performance guidelines.	1
Transaction Status Code (1-Held, 2-Ready) (DOC_STA_CD)	Required field that will be used by the program when loading transactions. Valid values are 1 - Held and 2 - Ready. Transactions loaded as held will not be selected by another process searching for transactions to submit. Ready transactions will be selected by such processes. Held is often used when some user action is required.	
EFT Transaction Code (EFT_DOC_CD)	Required Field. This field refers to the Transaction that will be generated by the Automated Disbursement process.	EFT
EFT Department Code (EFT_DOC_DEPT_CD)	Required Field. The transaction department to be used with the transaction code and prefix to find an Automatic Transaction Numbering entry. This will appear as the transaction department for all	No Default

Parameter	Description	Default Value
	transactions created by the program.	
EFT Unit Code (EFT_DOC_UNIT_CD)	Required field if Require Transaction Unit Code flag is set to true for the transaction code on DCTRL Table, else it's an optional field. Provides security with a unit code enabling transaction access to be secured at a level below department.	No Default
EFT Prefix (EFT_PFX)	Required Field. The transaction prefix to be used with the transaction code and department to find an Automatic Transaction Numbering entry. This will appear in the beginning of all transaction ID's for all transactions created by the program.	No Default
Default Disbursement Department contact (DEF_DEPT_CNTAC)	Optional Field. This batch parameter will be used to retrieve the disbursement contact code that can be used to retrieve contact information.	
Error File Name (ERROR_FILE_NM)	Required (and protected) field that defines the name of the error file that will be created	/ADUpLoadError.txt
FILE_NM	File Name To Be Imported	/ADDocument.xml
Infer Transaction Dept from DISRQ`	Required field. If the value is set to *True*, then the system will generate the disbursement transaction using the Transaction Department that is inferred from the Consolidation Object 6 field of corresponding record on Disbursement Request. If the value is set to *False*, then the system will generate the disbursement transaction using the Transaction Department that is entered in the job parameter. Valid values are *True* and *False*.	True
Intercept Selection (INT_SEL)	Required field. The Intercept Selection batch parameter allows a user to specify the selection values that are valid for intercept during the disbursement process. Valid values are 1, 2, 3, 4, 5, 6 & 7. Multiple values can be entered when separated by commas.	

Parameter	Description	Default Value
Load Parameter File (LOAD_PARAM_FILE_NAME)	Required. Indicates the name of the SMU load parameter file that will be created by this job step and later used as input by subsequent jobs.	
Maximum Block Size (MAX_BLOCK_SIZE)	Required Field. This is to limit the number of transactions per parameter file. If the number of records written per parameter file exceeds the block size then there will be a new parameter file created for the same priority.	100
Number of Processors (NUM_OF_PROC)	Required Field. This value specifies the maximum number of parameter files that would be created for a given priority.	3
Override Level (OVERRIDE_LVL)	Optional. Override level used when that of the user ID submitting the chain is not desired. If entered, must be an integer between 1 and 10 to attempt to override errors.	
SMU job parameter file name (PARAMETER_FILE_PREFIX)	Required Field. The Parameter File Prefix name used in SMU job parameter file name. The parameter file name would be created with the given prefix.	ADSBMT
Select Block (SELECT_BLOCK)	Select Block Size. It is the number of Disbursement Request line records fetched at a time. If not entered, it is defaulted to 3000. Can be used for Performance tuning.	3000
Update Intercept Selection (UPD_INT_SEL)	Valid values are *True* or *False*.	
Progression Counter Size	Optional field. During processing, the job writes messages to the log to report on its progress based on the number of records already processed. When this parameter is specified, the value controls the interval at which these progression messages are written to the job log. A specified value should be a positive integer.	100
ZRODLR_LST_GRP_CHKS	The parameter which determines if the Check Numbers need to be	N

Parameter	Description	Default Value
	assigned for $0 Checks.	
User ID	Optional. When the User ID submitting the chain should not be used, this parameter supplies an alternative.	blank
SEQ_DISB_CAT_LIST	Multiple comma delimited Disbursements values.	blank

Major Output

- AD transaction XML File
- Updates to the temporary tables (R_AP_DISB_RQST_LN & INCT_DISB_RQST)
- AP Disbursement Workload table (R_AP_DISB_WORKLOAD)
- SMU job parameter file
- CheckNumberParam.txt

Job Return code

The following table shows the potential job return codes for the AD XML Creation job.

Return Code	Condition
Successful (1)	All of the selected payment records are processed successfully.
Warning (4)	No eligible records found. This could be because of the following reasons: • No active Disbursement parameter table record exists • No records found on the Disbursement Request table for the parameters entered on the Disbursement parameter table.
Non-Fatal Error (8)	When a Maximum Line Limit edit has been triggered.
Failed (12)	The job will fail under the following conditions: • Parameters are invalid • Unable to find the System Options record for the Fiscal Year associated with the Check Date • Run time exceptions for unexpected situations • Restart failed because another instance of the AD chain has already been run successfully • When this job ends with a return code of Failed, subsequent jobs in the chain will be set to inactive.

Return Code	Condition
	• Unable to find the 1099 Processing Options and Controls record for the Calendar Year associated with the Application Date.
Terminated (16)	This return code will be issued when the job is terminated by the user. When this job ends with the Return Code of *Terminated*, subsequent jobs in the chain will be set to inactive.
System Failure (20)	This return code will be issued when the job is terminated because of database server or network issues. When this job ends with the Return Code of *System Failure*, subsequent jobs in the chain will be set to inactive.

Sort Sequence

The Disbursement Request Lines are sorted by the following fields to create the AD XML file. This sorting logic decides the disbursement transaction structure AD XML file (that is, whether it is an AD transaction or EFT transaction, where to break the Transaction, where to break the Vendor Line and Accounting line)

- Disbursement Format
- Bank
- Cleared Date (Effective Date: for EF only; always spaces for AD)
- Priority
- Consolidation Object 1 (spaces is valid)
- Consolidation Object 2 (spaces is valid)
- Handling Code (always spaces for EFT)
- Consolidation Object 3 (spaces is valid)
- Consolidation Object 4 (spaces is valid)
- Disbursement Category (spaces is valid)
- Vendor/Payee Code
- Vendor/Payee Name (for miscellaneous vendors)
- Consolidation Object 5 (spaces is valid)
- Consolidation Object 6 (spaces is valid)
- Transaction Code
- Transaction Department
- Transaction ID

Selection Criteria

The AD XML Creation job selects the payment lines from Disbursement Request based on the following selection criteria:

- Active indication for a Disbursement Parameter record is *true*.

- The Scheduled Payment Date is between or equal to the beginning and end date of the Disbursement Parameter records.
- The Transaction Department, Bank, Disbursement Type, Online Disbursement indication, and Disbursement Priority of Disbursement Request equal the corresponding field on an active Disbursement Parameter record.
- Disbursement Request record is not on HOLD.

Problem Resolution

There is no need to back out any updates if this job fails in any of the steps since the job only updates the temporary table (R_AP_DISB_RQST_LN) which gets cleared when the job is rescheduled the next time. If the job, after writing the records to the temporary table, ends with a return code of *Failed*, *Terminated* or *System Failure* and another instance of the job has already been scheduled and run successfully, then this job should not be restarted – it should only be rescheduled. The job uses the check pointing logic and any step after writing the records to the temporary table will start using the temporary table records only. Since another instance of this chain ran successfully, the records on the temporary table no longer belong to the failed job; hence the failed job should not be restarted.

The following table shows the possible return codes and recommendations for each processing step.

Step 1: Parameter Validation:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	N/A	This step does not issue this return code.	This step does not issue this return code.
Non-Fatal Error (8)	N/A	This step does not issue this return code.	This step does not issue this return code.
Failed (12)	Required Parameters are not entered **Sample Message:** Check Transaction code cannot be blank.	Enter the Check Transaction code and restart the job.	Alternatively, the job can be rescheduled after correcting the parameters.
	Entered Parameters are not valid **Sample Message**: Department xxxx (xxxx being the value from the parameter) is not a valid Department.	Enter the correct Department and restart the job.	Alternatively the job can be rescheduled after correcting the parameters.

Possible Return Codes	Condition	Recommendation	Other Instructions
	System Option record not found **Sample Message**: System Option record not found for the Fiscal Year xxxx (xxxx being the year used in the lookup.	Make sure that the System Option record not found record exists for the FY associated with the Check date and restart the job.	Alternatively the job can be rescheduled after adding the System Option record not found record.
	Failed because of runtime exceptions for an unexpected situation.	The reason for the failure needs to be investigated before restarting the job.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can either be restarted or rescheduled.	
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can either be restarted or rescheduled.	

Step 2: Selection of records

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	Job ended with a Warning because there is no active Disbursement parameter table record. **Sample Message:** No Active Disbursement Parameter record found.	Make sure that the Disbursement Parameter has at least one active record and restart the job.	Alternatively, the job can be rescheduled with a different set of parameters.

Possible Return Codes	Condition	Recommendation	Other Instructions
	The job ended with a *Warning* because there are no eligible records found on Disbursement Request. **Sample Message:** No eligible records found on the Disbursement Request Table.	This could happen when there are no eligible records found on Disbursement Request that match the parameters entered on the Disbursement Parameter. Make sure to enter the correct parameters and restart the job.	Alternatively, the job can be rescheduled with a different set of parameters.
Non-Fatal Error (8)	N/A	This step does not issue this return code.	This step does not issue this return code.
Failed (12)	Failed because of runtime exceptions for an unexpected situation.	In this step the job can fail with fatal conditions only on encountering unknown exceptions. If that happens, investigate the exception reported by the process, resolve the error and restart the job.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can either be restarted or rescheduled.	
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can either be restarted or rescheduled.	

Step 3: Vendor editing and Amount calculation

This part of the process validates the Automated Payment Hold parameter on the Application Parameters (APPCTRL) table. If it is set to *False*, this part of the process will be skipped. If it is set to *True*, the process will only select records on DISRQ if the **Current Hold Type** field is *null* or **Allow Bypass Requests** flag is *checked* for a scheduled payment request.

Each disbursement request that is selected for processing must pass a series of Vendor edits. Any disbursement request that does not pass Vendor edits will not be processed further.

Disbursement requests for vendors or payees who are not approved to receive payments on the Vendor/Customer (VCUST) table, or if the Hold Payment flag is selected (on VCUST) will not be processed. The job verifies whether the vendor is Active or not on the VCUST table.

The batch process will also validate that the disbursement type of a DISRQ record is set appropriately. If a record contains a disbursement type of EFT but the vendor customer record for the payee or the vendor (if a payee is not present) has an EFT Status set to something other than 'Eligible for EFT', then the batch process will update the disbursement status to check or warrant. Also, if a record's associated Disbursement Parameter record has the EFT to Hardcopy check box set to *True*, then the record will have its disbursement type (i.e., check or warrant) set to the vendor customer code's default disbursement type as defined on the Vendor Customer (VCUST) table.

The batch process will determine if the records that remain on the DISRQ table, after the vendor editing steps complete, are eligible for any adjustments in the amount that will be disbursed. This includes calculations for discounts, tax, penalties, interest, backup withholding, contract withholding, 1042S withholding, and credit memos.

Discounts

A DISRQ record is considered eligible for discount if the Check Date defined on the DISPA record associated with the DISRQ record is less than or equal to the Discount Eligibility Date. The Discount Eligibility is calculated as follows:

- First, calculate the 'Effective Date' as the latter of the Service To date or Vendor Invoice Date (DISRQ Commodity Line tab). If both these dates are blank then the process picks up Transaction Date of Record from the DISRQ record as the Effective Date.
- Next, calculate the Discount Eligibility Date as the Effective Date + Discount Days from the discount terms.

If a DISRQ record has a discount term that is eligible for discount then the batch process will calculate the discount amount and update the DISRQ record Discount Amount field. The Discount Amount is calculated as follows:

- Discount Amount = (Line Amount - Retainage Amount - (Freight Amount + Tax Amt)) * (Disc Rate / 100)

Tax

If discounts are taken for a record on the DISRQ and the tax type specified for the DISRQ record is a percentage tax (i.e., not a flat tax as a flat tax will result in the same amount), the batch process will determine if the tax amount needs to be recalculated for the new payment amount (i.e., Line Amount – Discount Amount). If the SOPT record for the current fiscal year has the Recalculate Tax checkbox set to true, then the DISRQ records that have a discount amount will have the Tax Amount recalculated (if one exists).

The Tax Amount is recalculated as follows:

- Tax Amount = (Pretax Line Amount – Discount Amount) * (Total Tax Percentage /100)

The Tax Amount calculated above is subtracted from the Tax Amount on the DISRQ. The difference between these two values is subtracted from the Tax Amount on the DISRQ record (DISRQ record updated). Also, the differences in any Regular Tax (that is, non Use Tax) or Use Tax between payment time and disbursement time are accounted for in Backup Withholding and Contract Withholding calculations. The differences are calculated as follows:

- Regular Tax Adjusted Amount From Discount = Payment Regular Tax – ((Pretax Line Amount – Discount Amount) * Regular Tax Percentage / 100)
- Use Tax Adjusted Amount from Discount = Payment Use Tax – ((Pretax Line Amount – Discount Amount) * Use Tax Percentage / 100)

Penalties

If the SOPT record for the current fiscal year has the Calculate Penalties on Disbursement checkbox set to true, then the DISRQ records are eligible to calculate a penalty amount. A DISRQ record is also considered eligible for penalty if the Check Date defined on the DISPA record associated with the DISRQ record is on or after the Penalty Eligibility Date. The Penalty Eligibility Date is calculated as follows:

- First, calculate the 'Effective Date' as the latter of the Service To date or Vendor Invoice Date (DISRQ Commodity Line tab). If both these dates are blank then the process picks up Transaction Date of Record from the DISRQ record as the Effective Date.
- Next, calculate the Penalty Eligibility Date as the Effective Date + Penalty Lag Days from SOPT.

If a DISRQ record is eligible for penalty then the batch process will calculate the penalty amount and update the DISRQ record Penalty Amount field. The Penalty Amount is calculated as follows:

- Penalty Amount = (Rate Of Penalty/100) * (Line Amount - Retainage Amount - (Freight Amount + Tax Amount))

Interest

The interest calculation within the AD chain will only be performed if the SOPT option **Calculate Interest Outside of Disbursement** is *unchecked* and **Calculate Interest on Disbursement** is *checked*. DISRQ records that had discounts taken will not be eligible for interest payments.

If the SOPT record for the current fiscal year has the Calculate Interest on Disbursement checkbox set to true, then the DISRQ records are eligible to calculate an interest amount. A DISRQ record is considered eligible for interest if the Check Date defined on the DISPA record associated with the DISRQ record is on or after the Interest Eligibility Date. The Interest Eligibility Date is calculated as follows:

- First, calculate the 'Effective Date' as the latter of the Service To date or Vendor Invoice Date (DISRQ Commodity Line tab). If both these dates are blank then the process picks up Transaction Date of Record from the DISRQ record as the Effective Date.
- Next, calculate the Interest Eligibility Date as the Effective Date + Interest Lag Days from SOPT.

If a DISRQ record is eligible for interest then the batch process will calculate the interest amount and update the DISRQ record Interest Amount field. The Interest Amount is calculated as follows:

- Interest Amount = (Line Amount - Retainage Amount - (Freight Amount + Tax Amount)) * (Daily Disbursement Interest Rate [from SOPT] /100) * (Check Date – 'Effective Date')

Backup Withholding

DISRQ records are subject to backup withholding if a record meets the following criteria:

1. The Backup Withholding check box is set to true on the 1099P table AND
2. One of the conditions marked as A or B below is true:
    a. The 1099 Backup Withholding Status CVL on the 1099I table is set to *Eligible* and the 1099 Reportable flag is checked for the record's Vendor, TIN and TIN Type combination (as defined on VCUST table), AND

        For at least one of the following 4 COA elements (using this precedence order: Sub BSA, BSA, Sub Object, Object) the 1099 Income Code and Income Type fields are populated (as defined on SBSA, BSA, SOBJ, or SOBJ tables respectively), AND

Backup Withholding flag is *checked* for the corresponding Form Type and the 1099 Type of Income on TINC.

OR

b. The 1099 Backup Withholding Status CVL on the 1099I table is set to a value other than *Eligible* and the 1099-INT Backup Withholding and the 1099 Reportable flags are *checked* for the record's Vendor, TIN and TIN Type combination (as defined on VCUST table), AND

For at least one of the following 4 COA elements (using this precedence order: Sub BSA, BSA, Sub Object, Object) the 1099 Income Code and Income Type fields are populated (as defined on SBSA, BSA, SOBJ, or SOBJ tables respectively), AND

For the reportable COA element (per previous condition) and the 1099 Type of Income it is associated with, the Form Type field on the respective COA table = I, AND

The Backup Withholding flag is *checked* for the corresponding Form Type and the 1099 Type of Income on TINC.

3. **No** matching record exists on 1099E table using any of the following 3 lookups based the Fiscal Year specified on the record's Accounting Line:

- Actual Fund and Department from the Accounting Line;
- Actual Fund, but Department is *ALL* ;
- Fund is *ALL* and actual Department;

Records that meet all the selection criteria for backup withholding will have a backup withholding amount calculated as follows:

Backup Withholding Amount = (Backup Withholding Rate (from 1099P)/100) * (Line Amount – Both Tax Adjusted Amount from Discount – Discount – Retainage + Penalty + Interest – (Payment Use Tax Amount – Use Tax Adjusted Amount From Discount))

The calculated backup withholding amount updates the Backup Withholding Amount field on the DISRQ record.

1042S Withholding

DISRQ records are subject to 1042S withholding if a record meets ALL of the following criteria:

The 1042S Withholding check box is set to *true* on the 1099P table for the Calendar Year defined by the disbursement record's Check Date.

The 1042-S Ch.3 Recipient Code flag is *checked* for the Vendor Code on the VCUST table or the 1042-S Ch. 4 Status Code flag is *checked* for the Vendor Code on the VCUST table.

The 1042-S Backup Withholding Status on 1042I is either *blank* or *Pending*.

The 1042-S Recipient Account Number associated with the Vendor Code on the VCUST table has the 1042S Withholding check box set to *true* on the 1099 Reporting Information table.

The Object, Balance Sheet Account, Sub-Object or Sub Balance Sheet Account contains a 1042 Income Code and Income Type (that is, it is 1042 Reportable).

The combination of 1042-S Income Code, 1042-S Income Type associated with the applicable COA (Object, Sub Object, BSA or Sub BSA) and the 1042-S Ch.3 Recipient Code or 1042-S Ch. 4 Status Code associated with the Vendor Code is defined on the 1042-S Type of Income table with the Backup Withholding flag *checked*.

If the vendor is found to be eligible for 1042-S Withholding, the system will then retrieve the current applicable tax rate for this vendor as follows and calculate the actual 1042-S Withholding Amount of the Accounting Line:

If the vendor is determined to be a Ch. 3 vendor (that is, 1042-S Ch. 3 Recipient Code is selected on VCUST), then the system retrieves the 1042-S Tax Rate from the ICTX table for the combination of 1042-S Income Code (defined by eligible COA element), IRS Country of Residence and IRS Country Sub Code (defined on VCUST by the transaction's Vendor);

If the 1042-S Backup Withholding Status on 1042I is set to *blank* and a matching record is found on ICTX table, then the value in the 1042-S Ch.3 Tax Rate field is inferred from the matching record to the 1042-S Tax Rate field on the Accounting Line of the AD Transaction and use the Tax Rate during calculation;

If no matching record is found on ICTX, then the value in the 1042-S Backup Withholding Rate field on 1099P table (for the Calendar Year of the Record Date specified on the Header) is inferred to the 1042-S Tax Rate field on the Accounting Line of the AD Transaction and the Tax Rate is used during calculation.

If the 1042-S Withholding Status on 1042I is set to *Pending*, a look up is not performed on ICTX. The value in the 1042-S Backup Withholding Rate on 1099P is inferred. This rate is used to calculate 1042-S backup withholding on the transaction.

If the 1042-S Backup Withholding Status on 1042I is set to *Not Eligible*, then the 1042-S Backup Withholding Rate on 1099P is inferred, but the system does not calculate or withhold the backup withholding tax. The tax rate is used for reporting to the IRS.

If the vendor is determined to be a Ch. 4 vendor (that is, 1042-S Ch. 4 Status Code is selected on VCUST), then the system retrieves the 1042-S Tax Rate from the ICTX4 table for the combination of 1042-S Income Code (defined by eligible COA element), IRS Country of Residence and IRS Country Sub Code (defined on VCUST by the transaction's Vendor);

If the 1042-S Backup Withholding Status on 1042I is set to *blank* and a matching record is found on the ICTX4 table (based on 1042-S Income Code and IRS Country of Residence) then the tax rate from the matching record is inferred to the 1042-S Tax Rate field on the Accounting Line of the AD Transaction. This is the rate used if backup withholding calculation is required based on the 1042T BWH flag.

If no matching record is found on ICTX4, then the value in the 1042-S Ch. 4 Tax Rate field on 1099P table (for the Calendar Year of the Record Date specified on the Header) is inferred to the 1042-S Tax Rate field on the Accounting Line of the AD transaction and use the Tax Rate during calculation.

If the 1042-S Withholding Status on 1042I is set to *Pending* then the system does not perform a look up to ICTX4. The value in the 1042-S Chapter 4 Tax Rate on 1099P is inferred. This rate is used to calculate 1042-S backup withholding on the transaction.

If the 1042-S Backup Withholding Status on 1042I is set to *Not Eligible*, then the 1042-S Ch. 4 Tax Rate on 1099P is inferred, but the system does not calculate or withhold the backup withholding tax. The tax rate is used for reporting to the IRS.

Records that meet all the selection criteria for 1042S withholding will have a 1042S withholding amount calculated as follows:

> 1042S Withholding Amount = (Tax Rate / 100) * (Line Amount – Both Tax Adjusted Amount from Discount – Discount – Retainage + Penalty + Interest – (Payment Use Tax Amount – Use Tax Adjusted Amount from Discount))

The calculated 1042S-withholding amount updates the Backup Withholding Amount field on the DISRQ record.

Contract Withholding

1. A payment request line is exempt from Contract Withholding if the Contract Withholding Exempt check box is selected on DISRQ, Vendor, Object, Sub Object, Commodity, Program, Appropriation, Balance Sheet, or Sub Balance Sheet tables.

2. A payment request line is not eligible for Contract Withholding if any of the following conditions is satisfied:

   a. The payment request line is exempt from Contract Withholding (see above)

   b. The calculated Backup Withholding Amount or 1042S Withholding Amount is not zero.

   c. The Contract Withholding flag on 1099P is not selected for the year portion of the Application Date.

   d. The payment request line has Procurement Card Payment set to Yes on DISRQ and Apply Contract Withholding to PCard Payments is not selected on 1099P.

3. If a payment request line is eligible for Contract Withholding, then the Contract Withholding Amount is calculated as follows:

   Contract Withholding Amount = (Line Amount – Both Tax Adjusted Amount from Discount – Discount – Retainage + Penalty + Interest – (Payment Use Tax Amount – Use Tax Adjusted Amount from Discount)) * 1099P Contract Withholding Rate;

   Otherwise, set it to $0.

   Contract Withholding is calculated for credit memo lines like they are calculated for positive lines. For credit memo lines, the Contract Withholding amount is negative.

Retainage

If the referenced Award contains applicable retainage terms, a portion of each vendor payment is withheld for retainage. The amount withheld is determined by retainage terms that are stored on the Award. Retainage amount is calculated on the PR transaction and not during the AD process. The AD process only uses the Retainage amount stored on the DISRQ record to make the adjustment

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	N/A	This step does not issue this return code.	This step does not issue this return code.
Non-Fatal Error (8)	N/A	This step does not issue this return code.	This step does not issue this return code.
Failed (12)	Failed while restarting the job since another instance of the	The only option available for this scenario is to schedule a new job.	

Possible Return Codes	Condition	Recommendation	Other Instructions
	job has already been run successfully. **Sample Message**: Cannot restart the job since another instance of this job has already been run successfully.		
	1099 Processing Options and Controls record not found. **Sample Message**: 1099P record does not exist for Calendar Year xxxx (xxxx being the year used to look at the 1099P Table).	Make sure that the 1099P record exists for the Calendar Year associated with the Application Control Date and restart the job.	Alternatively the job can be rescheduled after adding the 1099P record.
	Job failed due to Fatal conditions.	At this step, the job can fail under the following two conditions: • Encounters any runtime exceptions • Failed during restart If the job fails because of the runtime exceptions, investigate the exception reported by the process, resolve the error and restart the job. If the job failed during restart, then schedule a new chain job.	
Terminated (16)	Job is terminated manually by the	The reason for the termination needs to be investigated. The job can	

Possible Return Codes	Condition	Recommendation	Other Instructions
	user.	either be restarted or rescheduled.	
System Failure (20)	When the job is terminated because of database server or network issues	The reason for the System Failure needs to be investigated. The job can either be restarted or rescheduled.	

Step 4: Credit Memo Calculation

In order to apply the appropriate criteria to consolidate credit memos, the system will examine the selected negative DISRQ records and determine whether it is a Single Check payment, a PCard payment, or all other payments, in that precedence order.

Payments to Miscellaneous Vendors – One check is generated for each vendor line and the miscellaneous vendors payments are not grouped with any other payments.

Single Check payments: If the Single Payment indicator is Yes, then credit memos are consolidated within the vendor line. In this event, the other credit memo consolidations are ignored: System Option of PCard Consolidation Option and the following Disbursement Parameter settings: Consolidate Credit Memo by Disbursement Grouping, Consolidate Credit Memo by AL Department, Consolidate Credit Memo by Payment Request Transaction Department, Include Address in Credit Memo Consolidation, and Match on Disbursement Priority.

Payments to Non-Miscellaneous Vendors with Non-Single Check – Payments for the same Vendor will be consolidated across the payment requests. It is possible that one Check / EFT will be generated for more than one payment request transaction.

PCard payments that have Procurement Card Payment set to Yes (checked) on the DISRQ table will be grouped and consolidated based on the SOPT option Pcard Consolidation Options.

Payments to Third-Party / Payees – Payments to Third parties / Payees will be grouped and consolidated based on the Payee code.

Since payment requests are disbursed at the accounting line level in Advantage, a payment request will be disbursed by more than one disbursement transaction if any of the consolidation fields are not the same for all associated accounting lines.

Within each classification, the credit memo consolidation logic will be carried out as follows:

The SOPT PCard Consolidation Option will control the credit memo consolidation logic as follows:

SOPT PCard Consolidation Option	Credit Memo consolidation will happen at this level
Payee	Bank Account, Payee Code, Payee Legal Name
Payee and Administrator ID	Bank Account, Payee Code, Payee Legal Name, and Administrator ID
Payee, Administrator ID and PCard ID	Bank Account, Payee Code, Payee Legal Name, Administrator ID, and PCard ID

The Disbursement Parameter option of Consolidate Credit Memo by Disbursement Grouping ensures that a credit memo is only applied to a positive payment with the same Disbursement Priority. This applies to all transaction types that can request a disbursement.

All other payments:

- The Disbursement Parameter option of Consolidate Credit Memo by Disbursement Grouping ensures a credit memo is only applied to a positive payment with the same Disbursement Priority. When Match on Disbursement Priority is enabled, without disbursement priority, all the credit memos will get adjusted against positive pay irrespective of matching priority. This applies to all transaction types that can request a disbursement.
- The DISPA Consolidate Credit Memo by Disbursement Grouping, Consolidate Credit Memo by AL Department, and Consolidate Credit Memo by Payment Request Transaction Department options and Include Address in Credit Memo Consolidation option will control the credit memo consolidation logic as follows:

DISPA Credit Memo Consolidation (with Include Address in Credit Memo Consolidation set to 'No')	**Credit Memo consolidation will happen at this level**
When Payee is present	
No Credit Memo Consolidation option is selected on DISPA	Bank Account, Payee Code, Payee Legal Name, Vendor Code, Vendor Legal Name
Consolidate Credit Memo by Disbursement Grouping	Bank Account, Payee Code, Payee Legal Name, Vendor Code, Vendor Legal Name, Disbursement Category
Consolidate Credit Memo by AL Department	Bank Account, Payee Code, Payee Legal Name, Vendor Code, Vendor Legal Name, Accounting Line Department
Consolidate Credit Memo by Payment Request Transaction Dept	Bank Account, Payee Code, Payee Legal Name, Vendor Code, Vendor Legal Name, Payment Request Transaction Dept (Consolidation Object 6)
Consolidate Credit Memo by Disbursement Grouping and Consolidate Credit Memo by AL Department are both selected	Bank Account, Payee Code, Payee Legal Name, Vendor Code, Vendor Legal Name, Disbursement Category, Accounting Line Department
Consolidate Credit Memo by Disbursement Grouping and Consolidate Credit Memo by Payment Request Transaction Department are both selected	Bank Account, Payee Code, Payee Legal Name, Vendor Code, Vendor Legal Name, Disbursement Category, Payment Request Transaction Dept (Consolidation Object 6)
Consolidate Credit Memo by AL Department and Consolidate Credit Memo by Payment Request Transaction Department are both selected	Bank Account, Payee Code, Payee Legal Name, Vendor Code, Vendor Legal Name, Accounting Line Department, Payment Request Transaction Dept (Consolidation Object 6)
Consolidate Credit Memo by Disbursement Grouping, Consolidate	Bank Account, Payee Code, Payee Legal Name, Vendor Code, Vendor Legal Name, Disbursement Category,

Credit Memo by AL Department and Consolidate Credit Memo by Payment Request Transaction Department are all selected	Accounting Line Department, Payment Request Transaction Dept (Consolidation Object 6)
When Payee is not present	
No Credit Memo Consolidation Option is selected on DISPA	Bank Account, Vendor Code, Vendor Legal Name
Consolidate Credit Memo by Disbursement Grouping	Bank Account, Vendor Code, Vendor Legal Name, Disbursement Category
Consolidate Credit Memo by AL Department	Bank Account, Vendor Code, Vendor Legal Name, Accounting Line Department
Consolidate Credit Memo by Payment Request Transaction Dept	Bank Account, Vendor Code, Vendor Legal Name, Payment Request Transaction Dept (Consolidation Object 6)
Consolidate Credit Memo by Disbursement Grouping and Consolidate Credit Memo by AL Department are both selected	Bank Account, Vendor Code, Vendor Legal Name, Disbursement Category, Accounting Line Department
Consolidate Credit Memo by Disbursement Grouping and Consolidate Credit Memo by Payment Request Transaction Department are both selected	Bank Account, Vendor Code, Vendor Legal Name, Disbursement Category, Payment Request Transaction Dept (Consolidation Object 6)
Consolidate Credit Memo by AL Department and Consolidate Credit Memo by Payment Request Transaction Department are both selected	Bank Account, Vendor Code, Vendor Legal Name, Accounting Line Department, Payment Request Transaction Dept (Consolidation Object 6)
Consolidate Credit Memo by Disbursement Grouping, Consolidate Credit Memo by AL Department and Consolidate Credit Memo by Payment Request Transaction Department are all selected	Bank Account, Vendor Code, Vendor Legal Name, Disbursement Category, Accounting Line Department, Payment Request Transaction Dept (Consolidation Object 6)

DISPA Credit Memo Consolidation (with Include Address in Credit Memo Consolidation set to 'Yes')	**Credit Memo consolidation will happen at this level**
Payee is present	
No Credit Memo Consolidation Option is selected on DISPA	Bank Account, Payee Code, Payee Legal Name, Payee Address Code, Vendor Code, Vendor Legal Name,

	Vendor Address Code
Consolidate Credit Memo by Disbursement Grouping	Bank Account, Payee Code, Payee Legal Name, Payee Address Code, Vendor Code, Vendor Legal Name, Vendor Address Code, Disbursement Category
Consolidate Credit Memo by AL Department	Bank Account, Payee Code, Payee Legal Name, Payee Address Code, Vendor Code, Vendor Legal Name, Vendor Address Code, Accounting Line Department
Consolidate Credit Memo by Payment Request Transaction Dept	Bank Account, Payee Code, Payee Legal Name, Payee Address Code, Vendor Code, Vendor Legal Name, Vendor Address Code, Payment Request Transaction Dept (Consolidation Object 6)
Consolidate Credit Memo by Disbursement Grouping and Consolidate Credit Memo by AL Department are both selected	Bank Account, Payee Code, Payee Legal Name, Payee Address Code, Vendor Code, Vendor Legal Name, Vendor Address Code, Disbursement Category, Accounting Line Department
Consolidate Credit Memo by Disbursement Grouping and Consolidate Credit Memo by Payment Request Transaction Department are both selected	Bank Account, Payee Code, Payee Legal Name, Payee Address Code, Vendor Code, Vendor Legal Name, Vendor Address Code, Disbursement Category, Payment Request Transaction Dept (Consolidation Object 6)
Consolidate Credit Memo by AL Department and Consolidate Credit Memo by Payment Request Transaction Department are both selected	Bank Account, Payee Code, Payee Legal Name, Payee Address Code, Vendor Code, Vendor Legal Name, Vendor Address Code, Accounting Line Department, Payment Request Transaction Dept (Consolidation Object 6)
Consolidate Credit Memo by Disbursement Grouping, Consolidate Credit Memo by AL Department and Consolidate Credit Memo by Payment Request Transaction Department are all selected	Bank Account, Payee Code, Payee Legal Name, Payee Address Code, Vendor Code, Vendor Legal Name, Vendor Address Code, Disbursement Category, Accounting Line Department, Payment Request Transaction Dept (Consolidation Object 6)
Payee is not present	
No Credit Memo Consolidation Option is selected on DISPA	Bank Account, Vendor Code, Vendor Legal Name, Vendor Address Code
Consolidate Credit Memo by Disbursement Grouping	Bank Account, Vendor Code, Vendor Legal Name, Vendor Address Code, Disbursement Category
Consolidate Credit Memo by AL Department	Bank Account, Vendor Code, Vendor Legal Name, Vendor Address Code, Accounting Line Department
Consolidate Credit Memo by Payment Request Transaction Dept	Bank Account, Vendor Code, Vendor Legal Name, Vendor Address Code, Payment Request Transaction Dept (Consolidation Object 6)

Consolidate Credit Memo by Disbursement Grouping and Consolidate Credit Memo by AL Department are both selected	Bank Account, Vendor Code, Vendor Legal Name, Vendor Address Code, Disbursement Category, Accounting Line Department
Consolidate Credit Memo by Disbursement Grouping and Consolidate Credit Memo by Payment Request Transaction Department are both selected	Bank Account, Vendor Code, Vendor Legal Name, Vendor Address Code, Disbursement Category, Payment Request Transaction Dept (Consolidation Object 6)
Consolidate Credit Memo by AL Department and Consolidate Credit Memo by Payment Request Transaction Department are both selected	Bank Account, Vendor Code, Vendor Legal Name, Vendor Address Code, Accounting Line Department, Payment Request Transaction Dept (Consolidation Object 6)
Consolidate Credit Memo by Disbursement Grouping, Consolidate Credit Memo by AL Department, and Consolidate Credit Memo by Payment Request Transaction Department are all selected	Bank Account, Vendor Code, Vendor Legal Name, Vendor Address Code, Disbursement Category, Accounting Line Department, Payment Request Transaction Dept (Consolidation Object 6)

Single check and non-single check PR accounting lines with negative amounts that cannot be adjusted will be placed on hold by the AD Chain process.

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	N/A	This step does not issue this return code.	This step does not issue this return code.
Non-Fatal Error (8)	This return code will be issued when the Max Line Limit validation fails during this step.	The job would have skipped those records that have failed in the Max Line Limit validation. In order to process payments for those records, before scheduling the new job the settings on the Transaction Component Requirements table for the Transaction Type AD needs to be changed appropriately. The other alternative is to modify the Credit Memo	

Possible Return Codes	Condition	Recommendation	Other Instructions
		lines into smaller amounts across several lines so that the lines can be fit into the Maximum line limit.	
Failed (12)	Failed while restarting the job since another instance of the job has already been run successfully. **Sample Message**: Cannot restart the job since another instance of this job has already been run successfully.	Schedule a new job.	
	Job failed due to Fatal conditions.	At this step, the job can fail under the following two conditions. • Encounters any runtime exceptions • Failed during restart If the job fails because of the runtime exceptions, investigate the exception reported by the process, resolve the error and restart the job. If the job failed during restart, then schedule a new job.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can either be restarted or rescheduled.	
System Failure (20)	When the job is terminated because of database	The reason for the System Failure needs to be investigated. The job can either be restarted or	

Possible Return Codes	Condition	Recommendation	Other Instructions
	server or network issues.	rescheduled.	

Step 5: Splitting Credit memo amount

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	N/A	This step does not issue this return code.	This step does not issue this return code.
Non-Fatal Error (8)	N/A	This step does not issue this return code.	This step does not issue this return code.
Failed (12)	Failed while restarting the job since another instance of the job has already been run successfully. **Sample Message**: Cannot restart the job since another instance of this job has already been run successfully.	Schedule a new job.	
	Job failed due to Fatal conditions.	At this step, the job can fail under the following two conditions. • Encounters any runtime exceptions • Failed during restart. If the job fails because of the runtime exceptions, investigate the exception reported by the process,	

Possible Return Codes	Condition	Recommendation	Other Instructions
		resolve the error and restart the job. If the job failed during restart, then schedule a new job.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can either be restarted or rescheduled.	
System Failure (20)	When the job is terminated because of database server or network issues	The reason for the System Failure needs to be investigated. The job can either be restarted or rescheduled.	If another instance of the job has already been scheduled and run successfully, then this job should not be restarted – it should only be rescheduled.

Step 6: Disbursement Limit Check

The disbursement limit check step determines if the remaining DISRQ records meet the Disbursement Limit amount criteria defined on all active DISPA records. This step is performed after all other amount calculations have occurred (for example, interest, penalties, and so forth). Records are sorted based on their priority. The Disbursement Limit Check is performed assuming there is no Contract Withholding to ensure the disbursements will not exceed the limit. The Net Line Amount (before Contract Withholding is subtracted) is used in determining the amount that counts against the disbursement limit. Note that it is possible that Contract Withholding will eventually be found to be $0 after payment grouping threshold evaluation. Therefore, by not subtracting the Contact Withholding, the logic ensures that the total disbursements will not exceed the DISPA disbursement limit. Once the disbursement limit is reached, the batch process will determine if any other grouped records (after the last set of grouped records that was deemed eligible under the disbursement limit) can be added to the cumulative amount and not exceed the disbursement limit. All records that are not used in meeting the disbursement limit will have the System Hold flag set to true and the System Hold Reason set to ‘Disbursement Cycle Limit Exceeded’.

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	N/A	This step does not issue this return code.	This step does not issue this return code.

Possible Return Codes	Condition	Recommendation	Other Instructions
Non-Fatal Error (8)	N/A	This step does not issue this return code.	This step does not issue this return code.
Failed (12)	Failed while restarting the job since another instance of the job has already been run successfully. **Sample Message**: Cannot restart the job since another instance of this job has already been run successfully.	Schedule a new job.	
	Job failed due to Fatal conditions	At this step, the job can fail under the following two conditions. • Encounters any runtime exceptions • Failed during restart If the job fails because of the runtime exceptions, investigate the exception reported by the process, resolve the error and restart the job. If the job failed during restart, then schedule a new job.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can either be restarted or rescheduled.	
System Failure (20)	When the job is terminated because of database server or network issues	The reason for the System Failure needs to be investigated. The job can either be restarted or rescheduled.	

Step 7: Payment Intercept

The payment intercept process goes through the following steps:

Select the eligible records from the Intercept Request Table (INTR)

The batch process will determine which INTR records are eligible for intercept. The INTR records eligible for intercept are selected as follows:

The Claim Status of the INTR record is set to active.

The INTR record must have an outstanding intercept amount.

- Internal Debt Outstanding Intercept Amount = (INTR) Outstanding Amount – (INTR) Disbursement Intercept Amount
- External Debt Outstanding Intercept Amount = (INTR) Outstanding Amount – (INTR) Disbursement Intercept Amount – (INTR) Transferred Amount

At least one of the Intercept Selection values from the AD Chain batch parameter (Intercept Selection field) must match an Intercept Selection value of the Entity associated with the debt record. The Intercept Selection value(s) for the Entity is recorded on the Entity (ENTY) table. If the Intercept Selection batch parameter is 'null' then no records will meet this condition resulting in all records on INTR being ineligible for intercept.

The INTR record does not meet the criteria of any active record on the Receivable Intercept Exception (INTREX) table.

If the Evaluate Intercept by Debt Type parameter on the Application Parameters (APPCTRL) table is *Yes*, then the job will verify if the INTR and DRL combination exists as an allowable payment/intercept combination on the Allowable Payments for Intercept by Debt Type (APIDT) table. If FY/Debt Type and/or Bank Account combination does exist on the APIDT and the INTR/DRL combination does not satisfy any APIDT record combination, the DRL record will be bypassed from intercept processing and will be processed for disbursement. Refer to the AD Chain run sheet in the *CGI Advantage Financial – Accounts Payable Run Sheets Guide* for more information. Also, refer to the "Allowable Payments for Intercept by Debt Type (APIDT)" topic of the *CGI Advantage Financial – Intercepts User Guide* for use of the parameter and the APIDT page.

INTR records that do not meet all of the above criteria will not be considered for the intercept logic steps.

Select the eligible records from the Disbursement Request Table

The batch process will first determine which DISRQ records are eligible for intercept. The DISRQ records that are ineligible for intercept are selected as follows:

If the DISRQ record has a Procurement Card Payment indication of *true*.

If the DISRQ record has a Vendor Code set as 'Miscellaneous' on its Vendor Customer (VCUST) record and the Intercept Misc Vendor Payments flag on SOPT is *false*.

If the DISRQ Net Line Amount is less than or equal to Zero then the DISRQ record is not eligible for intercept. Net Line Amount is calculated as follows:

> Net Line Amount = (Line Amount + Penalty – Discount + Interest –Withholding Amount – Retainage Amount – Credit Memo Adjustment Amount)

If a record on DISRQ with values of Transaction Code, Transaction Department, Transaction ID, Fund, Department, Unit, Object, Sub-Object, Appropriation, TIN, and TIN Type matches any combination of Transaction Code, Transaction Department, Transaction Unit, Transaction ID, AL Fund, AL Department, AL Unit, AL Object, AL Sub-Object, AL Appropriation, TIN, and TIN Type on Intercept Disbursement Exception (INTDBEX) and:

> The Offset Percentage is 0% or 100%, then the DISRQ record is not eligible for intercept.

The Offset Percentage is between 1 to 99 then the intercept amount will be calculated as per the Offset Percentage value set on INTDBEX. (Example: If Intercept Outstanding is $200.00 and the Offset Percentage is set to 40, the available Intercept Outstanding will be $200.00 x 40 = $80.00).

If a record on DISRQ with values of Transaction Code, Transaction Department, Transaction ID, Entity, Debt Type, Fund, Department, Unit, Object, Sub-Object, Appropriation, TIN, and TIN Type matches any combination of Transaction Code, Transaction Department, Transaction Unit, Transaction ID, AL Fund, AL Department, AL Unit, AL Object, AL Sub-Object, AL Appropriation, Entity (from INTR), Debt Type (from INTR), TIN, and TIN Type on Intercept Claiming Entity (INTCLE) and:

The Offset Percentage is 0% or 100%, then the DISRQ record is not eligible for intercept.

The Offset Percentage is between 1 to 99 then the intercept amount will be calculated as per the Offset Percentage value set on INTCLE. (Example: If Intercept Outstanding is $200.00 and the Offset Percentage is set to 40, the available Intercept Outstanding will be $200.00 x 40 = $80.00).

DISRQ records that meet one of the above exclusion criteria will not be considered for the subsequent intercept logic steps.

Records on the DISRQ and INTR tables that are selected based on the intercept selection criteria defined above are matched based on TIN and TIN Type. All records on the DISRQ that have a matching record on the INTR table are used in the subsequent payment intercept steps.

For Payment Intercept Adjustments, the Outstanding Amount available for intercept will be calculated as follows:

Internal Debts Calculations: Outstanding Amount - Intercept Amount - Federal Offset Amount

External Debts Calculations: Outstanding Amount - Intercept Amount - Transferred Amount - Federal Offset Amount - Federal Transfer Amount

Calculate the Default Intercept Fee and Supplementary Intercept Fee where applicable

Calculate Intercept Fees

Two sets of flat fees may be assessed to a DISRQ record that is matched with a debt record for an intercept. A Default Intercept Fee may be applied to charge an administrative fee for processing the intercept. A Supplementary Intercept Fee may be applied to charge an administrative fee required by an external entity for processing an intercept.

Intercept Fees will be calculated for those debt records that have the possibility of offsetting DISRQ records. This is determined by adding the debt record available amounts to intercept until they meet or exceed the grouped DISRQ amount that is available to be intercepted.

First, the batch process determines if an intercept fee may be calculated more than once for a debt. This is driven by the Intercept Frequency CVL on IOPT. This allows the user to decide how often an intercept fee may be applied for a receivable eligible for intercept.

If the Intercept Frequency CVL is set to ‘Once’, then a receivable transaction may have an intercept fee applied to it once. Subsequent attempts to intercept any line on a receivable transaction will not result in applying an intercept fee.

If the Intercept Frequency CVL is set to ‘Once Per Day’, then a debt record may have an intercept fee applied to it once per processing day (day defined as system date). Subsequent attempts during the same processing day to intercept any line on a receivable transaction will not result in applying an intercept fee. However, if the receivable transaction is intercepted on a different processing day, it may be eligible for the application of intercept fees.

Determine if Default Intercept Fee may be calculated

If a record is eligible for the intercept fee calculation steps, then the batch process determines if a default intercept fee may be calculated. The following criteria must be met in order for a debt record to be eligible for a default intercept fee.

The current fiscal year record on the Intercept Options table will determine whether a default intercept fee will be applied to an intercepted record. If the 'Apply Intercept Fee' checkbox is set to true then a default intercept fee may be taken, otherwise the process will skip to determine if a supplementary intercept fee may be taken.

The sum of the available amount to intercept {Outstanding Amount – Intercepted Amount} for all INTR records for the RE Transaction ID must be greater than or equal to the Minimum Debt Amount on the default Intercept Fee Setup table record. The Default Entity, Default Debt Type, and Default Intercept Fee Code defined on the current fiscal year record on the IOPT table is used to retrieve the record from the Intercept Fee Setup (INTF) that is used for this calculation.

If all the criteria for calculating a default intercept fee are met then the batch process calculates the default intercept fee and retains it for future processing. Please refer to the intercept fee calculation formula below for how the fee is calculated.

Determine if Supplementary Intercept Fee may be calculated

After the default intercept fee logic is complete for a record, the batch process determines if a supplementary intercept fee may be calculated. This is accomplished by taking the Entity and Debt Type of the INTRL record and performing a lookup to the INTF table. If a matching record on the INTF table is found and that record was not used for calculating the default intercept fee, then the record will have a supplementary fee calculated. If an INTF record is not found, then a supplementary fee will not be taken. Once calculated, the supplementary intercept fee amount will be retained for future processing. It will eventually increment the Supplementary Intercept Fee Amount for the DISRQ record and INTRL record. Please refer to the intercept fee calculation formula below for how the fee is calculated.

Intercept Fee Formula

Intercept fees are optional and come in two types: *standard* and *supplemental*. Both are setup on the Intercept Fee Setup (INTF) page. In the event that the maximum intercept amount (MIA) – an amount determined from any percentage defined on Intercept Request (INTR), Intercept Disbursement Exception (INTDBEX), or Intercept Claim Entity (INTCLE) record matched multiplied by the Disposable Income amount if present or the Payment Amount – is not sufficient to cover the applicable fee(s) and the debt being intercepted. In this event the Fee Order on Intercept Options (IOPT) drives processing logic.

That MIA is treated differently if Disposable Income is specified on the ABS payment request. When specified, the total fees applied and debt intercepted cannot exceed the MIA. When there is no Disposable Income, the MIA is the maximum amount of debt that can be intercepted. Fees applied are not limited by the MIA.

Fee Type = Split Fee (When Default Fee and Supplementary Fee both are available):

Case	Condition	Disposable Income (DI)	Payment Amount (PA)
A	**MIA calculated from PA/DI >= (fees + outstanding debt)**	Take fees, intercept all outstanding debt and pay any remainder	Take fees, intercept all outstanding debt and pay any remainder
	Example: PA/DI: $1000.00 MIA: $1000 to $775.00	Fees: $25.00 Debt: $750.00 Payment: $0.00 to $225.00	*Same as with DI*

	Default Fee=$20.00 Sup. Fee = $5.00 Outstanding Debt = $750.00		
B	**MIA calculated from PA/DI >= 2x Fees so no need to split**	Take fees, intercept remainder as debt and pay nothing	Take fees, intercept remainder as debt and pay nothing
	Example: PA/DI: $1000.00 MIA: $50.00 to $775.00 Default Fee=$20.00 Sup. Fee = $5.00 Outstanding Debt = $750.00	Fees: $25.00 Debt: $25.00 to $750.00 Payment: $250.00 to $950.00	*Different than DI because MIA only limits intercepted debt* Fees: $25.00 Debt: $50.00 to $775.00 Payment: $225.00 to $925.00
C	**MIA calculated from PA/DI < 2x Fees so split**	Half MIA. Use 1st half that has any rounded cent and pro-rate to fees and apply 2nd half to debt	Half MIA. Use 1st half that has any rounded cent and pro-rate to fees and apply 2nd half to debt
	Example: PA/DI: $1000.00 MIA: $49.99 Default Fee=$20.00 Sup. Fee = $5.00 Outstanding Debt = $750.00 **Example:** PA/DI: $1000.00 MIA: $25.00 Default Fee=$20.00 Sup. Fee = $5.00 Outstanding Debt = $750.00 **Example:** PA/DI: $1000.00 MIA: 21.33 Default Fee=$20.00 Sup. Fee = $5.00 Outstanding Debt = $750.00	MIA of 49.99 = $25.00.00 fees (*rounded up*) and 24.99 Debt Default Fee = $20.00 Supplemental Fee = $5.00 Debt: $24.99 Payment: $950.01 MIA of $25.00 = $12.50 fees and $12.50 Debt Default Fee = $10.00 Supplemental Fee = $2.50 Debt: $12.50 Payment: $975.00 MIA of 21.33 = 10.67 fees and 10.66 Debt Default Fee = 8.54 (*rounded up*) Supplemental Fee = 2.13 Debt: 10.66 Payment: $ 978.67	*Same as with DI*

Fee Type = Whole Fee (When Default Fee and Supplementary Fee both are available):

Case	Condition	Disposable Income (DI)	Payment Amount (PA)
A	**MIA calculated from PA/DI >= (fees + outstanding debt)**	Take fees, intercept all outstanding debt and pay any remainder	Take fees, intercept all outstanding debt and pay any remainder
	Example: PA/DI: $1000.00 MIA: $1000.00 to $775.00 Default Fee=$20.00 Sup. Fee = $5.00 Outstanding Debt = $750.00	Fees: $25.00 Debt: $750.00 Payment: $0.00 to 225.00	*Same as with DI*
B	**MIA calculated from PA/DI > Fees**	Take fees, intercept remainder as debt and pay nothing	Take fees, intercept remainder as debt and pay nothing
	Example: PA/DI: $1000.00 MIA: 50 to 774.99 Default Fee=$20.00 Sup. Fee = $5.00 Outstanding Debt = $750.00	Fees: $25.00 Debt: $25.00 to 749.99 Payment: $225.01 to $950.00	*Different than DI because MIA only limits intercepted debt* Fees: $25.00 Debt: $50.00 to $774.99 Payment: $ 200.01 to $925.00
C	**MIA calculated from PA/DI = Fees**	Do not intercept	Take fees, intercept nothing
	Example: PA/DI: $1000.00 MIA: $25.00 Default Fee=$20.00 Sup. Fee = $5.00 Outstanding Debt = $750.00	Fees: $0.00 Debt: $0.00 Payment: $1000.00	Fees: $25.00 Debt: $0.00 Payment: $0.00
D	**MIA calculated from PA/DI < Fees**	Do not intercept payment	Do not intercept payment
	Example: PA/DI: $1000.00	Fees: $0.00 Debt: $0.00	Fees: $0.00 Debt: $0.00

	MIA: 0.01 to 24.99 Default Fee=$20.00 Sup. Fee = $5.00 Outstanding Debt = $750.00	Payment: $1000.00	Payment: $1000.00

Calculate Intercept amount

Calculate Intercept Offset Amount

Disbursement Request records are offset by debt records in the order of the priority designated for the debt records. Note that prioritizing debt records for the offset is documented earlier in this design.

After intercept fees are calculated, the batch process will determine how much of the intercepted amount may offset the grouped DISRQ Net Line Amount (calculated earlier). The rule for determining whether the debt record plus the associated default and intercept fees may offset the DISRQ is as follows:

> Grouped DISRQ Net Line Amount > ((Default Intercept Fee + Supplementary Intercept Fee) * (1 + IOPT Intercept Margin Percentage / 100))

If the above formula is true then the Default and Supplementary Intercept fees may be taken against the grouped DISRQ records. This amount is reserved against the grouped DISRQ record's available amount to intercept.

If the Default and Supplementary fees cannot be taken, the batch process then determines if just the Default Intercept Fees can be taken. The rule for determining whether the DISRQ may be offset by the debt record plus the associated default intercept fee is as follows:

> Grouped DISRQ Net Line Amount > ((Default Intercept Fee Amount) * (1 + IOPT Intercept Margin Percentage/100))

If the above formula is true then the Default Intercept Fee may be taken against the grouped DISRQ records. This amount is reserved against the grouped DISRQ record's available amount to intercept.

If the Default and Supplementary Intercept fees or just the Default Intercept Fee cannot be taken the batch process then determines if just the Supplementary Intercept Fees can be taken. The rule for determining whether the debt record plus the associated default intercept fee may offset the DISRQ is as follows:

> Grouped DISRQ Net Line Amount > ((Supplementary Intercept Fee Amount) * (1 + IOPT Intercept Margin Percentage/100))

If the batch process determines that Default and Supplementary Fees cannot be taken against a grouped set of DISRQ records, it will calculate the intercept offset amount without the intercept fees (Fee Amounts set to $0). If a Default Intercept Fee and/or a Supplementary Intercept Fee are taken, then the fee amounts reduced the available amount to intercept on the DISRQ record. These amounts are reserved against the grouped DISRQ records before determining the intercept offset amount.

The intercept-offset amount is then calculated. The Intercept Amount on the grouped DISRQ records is set to the debt record's available amount to intercept. If the debt record's available amount to intercept is greater than the grouped DISRQ record's Net Line Amount (minus intercept fees as well) then the Intercepted Amount is set to the DISRQ grouped Net Line Amount. Note that it is possible that Contract Withholding will eventually be found to be $0 after

payment grouping threshold evaluation, in which case the actual amounts intercepted may be lower than the full amount possible.

Once the Intercept Offset amount is determined, the batch process writes the record to the Intercept Disbursement Request Line (INTRL) intermediary table with the amount actually intercepted and charged as Default and Supplementary Intercept Fees.

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	N/A	This step does not issue this return code.	This step does not issue this return code.
Non-Fatal Error (8)	This return code will be issued when the Max Line Limit validation fails during this step.	The job would have skipped those records that have failed in the Max Line Limit validation. In order to process payments for those records, before scheduling the new job the settings on Transaction Component Requirements for the Transaction Type AD needs to be changed appropriately.	
Failed (12)	Failed while restarting the job since another instance of the job has already been run successfully. **Sample Message**: Cannot restart the job since another instance of this job has already been run successfully.	Schedule a new job.	
	Job failed due to Fatal conditions.	At this step, the job can fail under the following two conditions. • Encounters any	.

Possible Return Codes	Condition	Recommendation	Other Instructions
		runtime exceptions • Failed during restart If the job fails because of the runtime exceptions, investigate the exception reported by the process, resolve the error and restart the job. If the job failed during restart, then schedule a new job.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can either be restarted or rescheduled.	
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can either be restarted or rescheduled.	

Step 8: AD XML Creation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	N/A	This step does not issue this return code.	This step does not issue this return code.
Non-Fatal Error (8)	N/A	This step does not issue this return code.	This step does not issue this return code.

Possible Return Codes	Condition	Recommendation	Other Instructions
Failed (12)	Failed while restarting the job since another instance of the job has already been run successfully.	**Sample Message**: Cannot restart the job since another instance of this job has already been run successfully. **Recommendation:** Schedule a new job.	
	Job failed due to Fatal conditions.	At this step, the job can fail under the following two conditions. • Encounters any runtime exceptions • Failed during restart If the job fails because of the runtime exceptions, investigate the exception reported by the process, resolve the error and restart the job. If the job failed during restart, then schedule a new job.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can either be restarted or rescheduled.	
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can either be restarted or rescheduled.	

Automated Disbursement Chain: System Hold Report Job

Job Name	System Hold Report Job
Recommended Frequency	See AD Chain information
Single Instance Required	Yes

Can be restarted?	No
Report Generated	Yes: System Hold Report

Overview

The System Hold Report job lists all of the accounting lines that were selected by the Automated Disbursements run that were not able to be liquidated during that run and were put on System Hold. This report also displays the reason (system hold reason) why the records could not be liquidated.

The System Hold Report job uses the Disbursement Request (DISRQ) and Disbursement Parameters (DISPA) tables as input to generate the System Hold Report by retrieving the records that are on System Hold on DISRQ that are associated with Active records on DISPA. Once the System Hold records are selected from DISRQ, the system performs a look up for the records on DISPA based on the Bank Account of the selected records, and retrieves the Credit Memo Consolidation Options selected on those records and generates the output.

Major Input

- Disbursement Parameters (R_AP_DISB_PARM)
- Disbursement Request (R_AP_DISB_RQST)

Batch Parameters

Parameter	Description	Default Value
AMSPARM	Parameter Location at AD Transaction System Hold Report Job (** **Refer to Note**: Assumptions for SWBP on page no. 7)	$$AMSROOT$$/Parms
Client Name for Report (CLIENT_NM)	Optional field. Entry of a value in this field specifies the name that will appear on the report.	No Default
Chain Job ID (CHAIN_JOB_ID)	Chain Job ID	$$@CHAINJOBID@$$

Major Output

The System Hold Report is produced by this job. This report lists the eligible payments that are put on System Hold during the prior run of the AD XML Generation process.

Batch Job Return Codes

The following table shows the potential job return codes for the System Hold Report.

Return Code	Condition
Successful (1)	Report generation is successful.
Warning (4)	Not applicable for this job.
Non-Fatal Error (8)	• Runtime exceptions encountered for any unexpected situations. • Technical failure
Failed (12)	Not applicable for this job.
Terminated (16)	This return code is issued when the job is terminated by the user.
System Failure (20)	This return code is issued when the job is terminated because of database server or network issues.

Sort Sequence

The report is sorted by Bank, then Vendor.

Within the Payments Eligible for System Hold section, the selected records are sorted alphabetically by the Referenced Payment transaction (Vendor Legal Name, Transaction Code, Transaction Department, Transaction ID, Vendor Line Number, Commodity Line Number, and Accounting Line Number).

Selection Criteria

The System Hold Report job selects the payment lines from Disbursement Request based on the following selection criteria to from the System Hold section of the report:

- Only active Disbursement Parameter records are selected
- The Scheduled Payment Date of the Disbursement Request record is between or equal to the From Date and To Date of a Disbursement Parameter record.
- The Transaction Department, Bank, Disbursement Type, Disbursement Category and Disbursement Priority of Disbursement Request are equal to the corresponding field on Disbursement Parameter.
- Disbursement Request records are marked as *System Hold*.

Problem Resolution

No database restore is required. If the job fails a new job should be scheduled after correcting the errors that caused the job to fail.

The following table shows the possible return codes and recommendations for each processing step.

Step 1: Batch Parameter Validation:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	N/A	This step does not issue this return code.	N/A
Warning (4)	N/A	This step does not issue this return code.	N/A
Non-Fatal Error (8)	Non-Fatal Error because of runtime exceptions caused by unexpected conditions.	The Non-Fatal Error reason needs to be investigated before rescheduling the job.	N/A
Failed (12)	N/A	This step does not issue this return code.	N/A
Terminated (16)	Job is terminated manually by the user.	A new job can be rescheduled.	N/A
System Failure (20)	Job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. A new job can be rescheduled.	N/A

Step 2: Report Generation:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	N/A	This step does not issue this return code.	N/A
Warning (4)	N/A	This step does not issue this return code.	N/A
Non-Fatal Error (8)	Non-Fatal Error because of runtime exceptions caused by unexpected conditions.	The Non-Fatal Error reason needs to be investigated before rescheduling the job.	N/A
Failed (12)	N/A	This step does not issue this return code.	N/A
Terminated (16)	Job is terminated manually by the user.	A new job can be rescheduled.	N/A
System Failure (20)	Job is terminated because of database server or Network issues.	The reason for the System Failure needs to be Investigated. A new job can be rescheduled.	N/A

Automated Disbursement Chain: AD Upload Job

Job Name	AD Upload Job
Recommended Frequency	See AD Chain information
Parallel processing enabled	No. Parallel processing is not supported for this job.
Can the job be restarted?	Optionally, based on the Save Restart Information parameter.
Exception report produced	No. All of the exceptions are only written to the log.

Overview

The AD Upload job loads the records from the XML File, generated by the AD XML Creation job, into the Transaction Catalog. This job uses the common utility to load the records into the Transaction Catalog. This job first validates the batch parameters. If the parameters are valid, then it loads the records into the Transaction Catalog. If the parameters are not valid, the job issues appropriate messages and ends with a status of Failed. Once the records are loaded into the Transaction Catalog, the summary information is written into the log as how many records were in the input file and how many records loaded successfully.

The job can be restarted if it fails, provided the Save Restart Information parameter is selected. If the failure occurred after the parameter validation, then the job should be restarted after resolving the errors. If the Save Restart Information parameter is not selected or if the restart is not the immediate option, then the new job can be rescheduled but before rescheduling the job, the transactions loaded by the failed job should either be processed or discarded so that they do not remain in the catalog.

Major Input

- AD Transaction XML file

Batch Parameters

Parameter	Description	Default Value
Parameter file (PARM_FILE)	Parameter file to Load Transactions	$$AMSPARM$$/LoadADParm.txt

Note: This PARM_FILE only contains the following subset of SMU parameters.

Parameter	Default Value
Action Code (ACTN_CD)	171
Commit Block Size (COMMIT_BLOCK)	1
Transaction Status Code (DOC_STA_CD)	2

Parameter	Default Value
Error File Name (ERROR_FILE_NM)	/ADUpLoadError.txt
File Name To Be Imported (FILE_NM)	/ADDocument.xml
Apply Overrides (APPLY_OVERRIDES)	True
By Pass Auto Transaction Number (BYPS_ADNT_FL)	True

Please refer to the "SMU Transaction Upload Job" run sheet in the *CGI Advantage Financial – Utilities Run Sheets* guide for the full list of SMU Transaction upload batch parameters.

Major Output

- AD Transactions in draft version
- EF Transactions in draft version

Batch Return Codes

The following table shows the potential job return codes for the AD Upload job.

Return Code	Condition
Successful (1)	All of the records are loaded into the Transaction Catalog successfully or the input file is empty.
Warning (4)	This Return Code will be issued when some of the records failed to load whereas all other records were loaded successfully.
Non-Fatal Error (8)	None of the records get loaded into the Transaction Catalog.
Failed (12)	• Parameters are invalid • When the input file is not found in the specified directory • Restart failed because another instance of the AD chain has already been run successfully • Runtime exceptions encountered for any unexpected situations When the job ends with a Return Code of *Failed*, subsequent jobs in the chain will be set to inactive.
Terminated (16)	This Return Code will be issued when the job is terminated by the user. When the job ends with a Return Code of *Terminated*, subsequent jobs in the chain will be set to inactive.
System Failure	This Return Code will be issued when the job is terminated because of database server or network issues. When this job

Return Code	Condition
(20)	ends with a Return Code of *System Failure*, subsequent jobs in the chain will be set to inactive.

Sort Sequence

N/A

Selection Criteria

N/A

Problem Resolution

If the job ends with a return code of *Failed* and above, the job can be restarted only when the Save Restart Information parameter is selected and another instance of the job has not been scheduled and run successfully. If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – a new job should only be scheduled.

If the job cannot be restarted immediately, then the transactions loaded by this job should be either processed or discarded before rescheduling the new job. To process the loaded transactions, submit the AD Submit job in that chain. This job would be set to Inactive since the AD Upload failed. The Submit job will submit the transactions that were generated during that AD chain. To discard the loaded transactions, submit the AD Clean Up and IT XML Creation job in that chain. The AD Clean Up and IT XML Creation job will discard all of the draft disbursement transactions (AD / EFT) from the Transaction Catalog.

The following table shows the possible return codes and recommendations for each processing step specific to the job in the chain. For general errors and recommendations, refer to the "SMU Transaction Upload Job" run sheet in the *CGI Advantage Financial – Utilities Run Sheets* guide.

Possible Return Codes	Condition	Recommendation	Other Instructions
Failed (12)	Failed while restarting the job since another instance of the job has already been run successfully. **Sample Message**: Cannot restart the job since another instance of this job has already been run successfully.	**Recommendation:** Schedule a new job.	

Automated Disbursement Chain: AD Submit Job

Job Name	AD Submitter
Recommended Frequency	See AD Chain information
Single Instance Required	No

Can be restarted?	Yes. See the Overview and the Problem Resolution sections for more details.
Report Generated	No. All of the exceptions are written to the error file.

Overview

This job submits the transactions generated by the AD XML Creation Job. This process is a part of the AD Chain that reads the R_AP_DISB_WORKLOAD table and SMU job parameter file to spawn multiple SysManUtil jobs. In order to run the AD Submit step in parallel, the Job Interaction Client (JIC) must be used (refer the *CGI Advantage System Administration Guide* for more information on JIC). Also, when running in parallel, the Process Assigned Jobs Only indication must be *false* in the Job Server Control for all job managers.

Major Input

- AP Disbursement Workload (R_AP_DISB_WORKLOAD)
- SMU job parameter file
- Draft Disbursement Transactions in the catalog

Batch Parameters

Parameter	Description	Default Value
Chain Job ID (CHAIN_JOB_ID)	A conditionally required field when running multiple instances of the AD Chain to differentiate each Job ID for different chains run at the same time.	$$@CHAINJOBID@$$
Iteration Count (ITERATION_COUNT)	The required number of iterations to process for a disbursement priority. Once the processing of an SMU is completed, AD Submitter looks for another unassigned job in the same priority to process the next set of transactions. If there are no jobs pending for the same priority, it waits for a few seconds (as per polling time) until the processing of other AD Submitter jobs is completed for that priority. Once the polling time elapses, the AD Submitter will looks for unassigned jobs. This process is repeated until the counter reaches the maximum Iteration count and times out automatically.	50
SMU Catalog ID (SMU_CATALOG_ID)	The required ID of the SysManUtil job to submit.	136

Parameter	Description	Default Value
Parameter File Prefix (PARAMETER_FILE_PREFIX)	The required prefix used to identify the SMU job parameter file name.	ADSBMT
Log Location (AMSLOGS)	The required file location for writing out and reading log files.	No Default
Parameter File Location (AMSPARM)	The required file location for writing out and reading parameter files.	No Default
Polling Time (Pooling Time)	A required number of seconds the AD Submitter to wait until the processing of other instances of the AD Submitter is completed for a given priority.	10000
Failed OOS Retry (SMU_FAILED_OOSRETRY_WARN)	When an SMU thread fails due to optimistic locking error and has an out of sync condition, with this parameter as Y, system will consider SMU job as successful and continue processing even though the spawned job returns 'Warning', the AD Submitter will show status as Successful and continue further process. If left blank, N defaults and the chain will stop processing at this step. When set to Y, any disbursements that did not submit will be discarded by the AD Cleanup job step, putting the Disbursement Request records back out there for payment in the following AD Chain run.	N

Note: This job uses only a subset of the SMU submit job parameters. For a full list of available parameters for the SMU submit job, refer to the “SMU Transaction Submit Job” run sheet in the *CGI Advantage Financial – Utilities Run Sheets* guide.

Major Output

The Transactions would have been processed to final or rejected.

Batch Return codes

The following table shows the potential job return codes for the individual AD submit job in the AD submitter.

If the AD Submitter uses the Job Interaction Client, then the return code of the AD Submitter job depends on the return codes of the individual jobs that were spawned during the processing. The following table shows the potential job return codes for the AD submitter job.

Return Code	Condition

Return Code	Condition
Successful (1)	All the transactions generated in that run submitted successfully.
Warning (4)	At least one spawned SMU instance contained an optimistic locking event and the Failed OOS Retry parameter was set to N. When the job ends with a Return Code of *Warning*, subsequent jobs in the chain is set to inactive.
Non-Fatal Error (8)	Not Applicable for this job.
Failed (12)	The job as failed to spawn any SMU jobs: • Input parameter file is not found. • Restart failed because another instance of the AD chain has already been run successfully. • Runtime exceptions encountered for any unexpected situations. When the job ends with a Return Code of *Failed*, subsequent jobs in the chain is set to inactive.
Terminated (16)	This Return Code is issued when the job is terminated by the user. When the job ends with a Return Code of Terminated, subsequent jobs in the chain are set to inactive.
System Failure (20)	This Return Code is issued when the job is terminated because of database server or network issues. When this job ends with a Return Code of System Failure, subsequent jobs in the chain are set to inactive.

Sort Sequence

N/A

Selection Criteria

N/A

Problem Resolution

If the job ends with a return code of *Failed* and above, the job can be restarted only when the Save Restart Information parameter is selected and another instance of the job has not been scheduled and run successfully. If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – a new job should only be scheduled.

If the restart is not an immediate option and the fatal error is because of a few transactions, the rest of the transactions can be submitted manually or discarded manually depending on the Issue. The Transaction IDs can be found on the input parameter file.

The following table shows the possible return codes and recommendations for each processing step specific to the job in the chain. For general errors and recommendations, refer to the “SMU Transaction Submit Job” run sheet in the *CGI Advantage Financial – Utilities Run Sheets* guide.

Possible Return Codes	Condition	Recommendation	Other Instructions
Failed (12)	Failed while restarting the job since another instance of the job has already been run successfully. **Sample Message**: Cannot restart the job since another instance of this job has already been run successfully.	Schedule a new job.	

Automated Disbursement Chain: AD Transaction Exception Job

Job Name	AD Transaction Exception
Recommended Frequency	See AD Chain information
Single Instance Required	Yes
Can be restarted?	Yes
Report Generated	Yes. The job generates the report in pdf and html formats.

Overview

This job in the AD chain generates an exception report that lists all of the errors encountered when the AD transaction was submitted in the earlier step. The report contains the following information:

- Payment Request transaction that was referenced on the rejected AD transaction
- Detailed error description along with the error code

The following table shows the progression messages issued in this job.

Process Steps	Messages
Creating exception report	• Validating batch parameters • Parameter validation completed • Selecting the records for reporting • Total Number of records selected to write Exception report: 'n' • Number of transactions processed for report: 'n' where n being the progression counter size. • Writing records to the report • Number of records inserted in report: 'n' where n

Process Steps	Messages
	being the progression counter size • Processing completed

Restartability Information

This job can be restarted if it is failed. The job has the following check points and when restarted, the job will start from the last saved check point.

- The first check point is at the starting point when the data is selected for inserting the record in R_AP_DISB_RQST_LN for the report.
- The second check point is before setting the selected records for report.

Major Input

- Draft Disbursement Transactions in the catalog with a Rejected status

Batch Parameters

Parameter	Description	Default Value
AMSLOGS	Logs Location at AD Transaction Exception Job (** **Refer to Note**: Assumptions for SWBP on page no. 7)	No Default
Client name for Report (CLIENT_NM)	Optional field. Entry of a value in this field specifies the name that will appear on the report.	No Default
Exception File Name (EXCEP_REP_FILE_NM)	Required field. Exception File Name	ExpRepFlat File.txt
REPORT_ID	Optional field. Report ID that will be displayed in the Exception Report.	No Default
Exception Severity Flag (EXP_SEV_FL)	If enabled, only errors with a severity of 'Error' or 'Severe' will be displayed on the exception report. When disabled, all errors will be displayed. Valid values are 0->False, 1->True	1
Progression Counter Size	Optional field. During processing, the job writes messages to the log to report on its progress based on the number of records already processed. When this parameter is specified, the value controls the interval at which these progression messages are	500

Parameter	Description	Default Value
	written to the job log. A specified value should be a positive integer.	

Output

- AD/EF Transaction Exception Report

Batch Return Codes

The following table shows the potential job return codes for the individual AD Transaction Exception job in the AD chain.

Return Code	Condition
Successful (1)	All of the transactions generated in that run submitted successfully.
Warning (4)	Not Applicable for this job.
Non-Fatal Error (8)	Not Applicable for this job.
Failed (12)	• Input file is blank. • Input parameter file is not found in the specified folder. • Restart failed because another instance of the AD chain has already been run successfully. • Runtime exceptions encountered for any unexpected situations. When the job ends with a Return Code of failed, subsequent jobs in the chain will be set to inactive.
Terminated (16)	This Return Code will be issued when the job is terminated by the user. When the job ends with a Return Code of *Terminated*, subsequent jobs in the chain will be set to inactive.
System Failure (20)	This Return Code will be issued when the job is terminated because of database server or network issues. When this job ends with a Return Code of *System Failure*, subsequent jobs in the chain will be set to inactive.

Sort Sequence

N/A

Selection Criteria

N/A

Problem Resolution

The following table shows the possible return codes and recommendations for each processing step specific to the job in the chain.

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	N/A	This step does not issue this return code.	N/A
Non-Fatal Error (8)	N/A	This step does not issue this return code.	N/A
Failed (12)	Input file is not entered. **Sample Message:** Exception file is required	Specify the Input file name and schedule a new job.	
	Input file is not found on the specified directory. **Sample Message:** The system cannot find the specified file.	Make sure that the specified file exists in the directory and restart the job. Alternatively, a new job can be scheduled. Sometimes, the directory path may not be correct. In that case, enter the correct directory path and restart the job. If the job cannot be restarted, schedule a new job.	
	Failed while restarting the job since another instance of the job has already been run successfully. **Sample Message**: Cannot restart the job since another instance of this job has already been run successfully.	Schedule a new job.	

Possible Return Codes	Condition	Recommendation	Other Instructions
	Failed because of runtime exceptions for an unexpected situation.	If the job fails with fatal conditions on encountering unknown exceptions, then investigate the exception reported by the process, resolve the error and restart the job. If the job cannot be restarted, schedule a new job.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can either be restarted or rescheduled.	
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can either be restarted or rescheduled.	

Automated Disbursement Chain: AD Clean Up and IT XML Creation Job

Job Name	AD Clean Up and IT XML Creation
Recommended Frequency	See AD Chain information
Single Instance Required	Yes
Can be restarted?	Yes. Refer to the “Overview” and the “Problem Resolution” sections for more details.
Report Generated	No. All of the exceptions are written to the error file.

Overview

The AD Clean Up and IT XML Creation job in the AD chain performs the following:

- Surveys rejected disbursement transactions, and depending on the type of error that caused transaction rejection, deletes rejected lines except when the line has retainage adjustments. The disbursement transaction is then re-processed a second time. The goal here is, if a disbursement transaction has 10 lines of which two were rejected, the system will try to reprocess just the eight payments lines to prevent bottlenecking of payments. There are some exceptions to this: If the transaction has any Intercept or the line has Retainage amount, then no accounting lines will deleted. The entire transaction will be skipped.

Similarly if the transaction has any credit memo, then also the transaction will be skipped. Finally, if the Payment Request is marked as single check and if any one of the AL has error, then the entire transaction is skipped and no lines will be deleted.

- Deletes any rejected disbursement transactions. Since these transactions cannot (with the exception of cancellation fields and specific dates) be edited for corrections processing, they are removed from the Transaction Catalog. After the accounting lines or transactions are discarded, Disbursement Request is updated with a System Hold Status of *true* and a System Hold Reason of transaction rejected.
- Once all rejected disbursement transactions are reprocessed and deleted from the transaction catalog, the AD Clean-up job creates the XML IT Upload file, which will be used to load Internal Transfer (IT) transactions to the transaction catalog. These transactions will record the intercept activity for all debt identified during the AD Chain processing, all intercept fees associated with the intercepted debt, the liquidation of receivables, and posting the intercepted monies to the appropriate accounts. Internal debt is identified if the debt record's transaction code is not a Receivable (RE) transaction type. One accounting line each will be created for the intercept amount, the default intercept fee amount (if one exists), and the supplementary intercept fee amount (if one exists).

The following table shows the processing steps and the progression messages issued in this job.

Process Steps	Messages
1. Parameter Validation	• Validating Batch Parameters • Parameters are valid or invalid depending on the Validation. If the parameter is invalid, the invalid value will be displayed in the log. • Batch Parameter validation completed
2. Perform AD cleanup	• Selecting eligible transactions for re-submit • Total number of transactions selected for Resubmit: 'n' • Re-submitting the AD transaction • Number of transactions resubmitted: 'n' • Re-submitting the AD transactions completed • Updating the hold status in DISRQ for failed transactions • Updating Hold status is completed • Selecting the draft transactions for Discard • Discarding rejected AD transactions • Number of transactions discarded = xxx.
3. Creation of IT Transaction xml file	• Selecting the eligible records to create IT transactions. • If no records found, the following message will be issued: No records found for the creation of the IT transaction. • Eligible records found for creation of the IT

Process Steps	Messages
	Transaction • Writing the records into the xml file • Creation of IT Transaction xml file completed

Restartability Information

This job can be restarted if it is failed. The job has the following check points:

- The first check point is before updating the record on Disbursement Request.
- The second check point is before resubmitting the failed Automatic Disbursement transactions.
- The third check point is before generating the ADCleanup.xml.
- The fourth check point is before discarding the failed transactions.

When restarted, the job will start from the last saved check point. For example, if the job encountered an error while re-submitting the failed transactions and ended with the return code of failed; when restarted, the job will start resubmitting the failed transactions and will not perform the Disbursement Request update step. If the job cannot be restarted due to any reason then a new job should be scheduled.

Major Input

- AD Submit Exception File
- Temporary Disbursement Intercept (INCT_DISB_RQST)
- Intercept Request (INTR)
- Intercept Activity (INTA)

Batch Parameters

Parameter	Description	Default Value
ACTN_CD	A required (and protected) action code that instructs the program to perform a certain action on its records.	171
Add Chain Job ID to IT Transaction XML File Name (ADD_JOBID_TO_ITXML_FILENM)	An optional output parameter which appends the Chain Job ID to the XML file of Intercept Transfer (IT) transactions. This ensures there is a unique file with each run and the previous is not over-written. When left blank, the value of *No* defaults so that the file name is ADCleanUp.xml. *Yes,* ensures the file name is changed as ADCleanup_<CHAINJOBID>.xml.	No Default

Parameter	Description	Default Value
Export Location at AD Clean Up and IT XML Creation Job (AMSEXPORT)	Required (** **Refer to Note**: Assumptions for SWBP on page no. 7)	No Default
Logs Location at AD Clean Up and IT XML Creation Job (AMSLOGS)	Required (** **Refer to Note**: Assumptions for SWBP on page no. 7)	No Default
Parameter Location at AD Clean Up and IT XML Creation Job (AMSPARM)	Required (** **Refer to Note**: Assumptions for SWBP on page no. 7)	No Default
APPLY_OVERRIDES	Apply Overrides	true
ASSIGN_DOC_ID	This field is used to assign Transaction IDs based on the value set: • If set to *true* - Auto generated Transaction ID will be included in XML. • If set to *false* - Transaction ID will be left blank in XML, and it will be generated when the transaction gets uploaded.	false
CHAIN_JOB_ID	Required field. This field is used if you are running multiple instances of the AD chain to differentiate all of the Job ID's for different chains run at the same time.	$$@CHAINJOBID@$$
Commit Block Size (COMMIT_BLK)	Required field. Controls how many records are committed by the application at one time. The size should be compatible with technical capabilities and performance guidelines.	10
IT Transaction Code) (DOC_CD)	Required field. This field refers to the Transaction that will be generated.	IT
IT Transaction Department Code (DOC_DEPT_CD)	Required field. The transaction department to be used with the transaction code and prefix to find an Automatic Transaction Numbering entry. It will appear as the transaction department for all	020

Parameter	Description	Default Value
	transactions created by the program.	
IT Transaction Prefix (DOC_PFX)	Required field. The transaction prefix to be used with the transaction code and department to find an Automatic Transaction Numbering entry. It will appear in the beginning of all transaction ID's for all transactions created by the program.	AUTO
IT Transaction Unit Code (DOC_UNIT_CD)	Required field. Provides security with a unit code enabling transaction access to be secured at a level below department.	
Error File Name (ERROR_FILE_NM)	Required (and protected) field that defines the name of the error file that will be created.	/IETExpt.txt
Exception File Name (EXCEP_REP_FILE_NM)	Required field that defines what file the program is to use to create an exception report.	ExpRepFlatFile.txt
XML File Name (FILE_NM)	Required field that defines what file will be uploaded.	/ADCleanUp.xml
LOAD_PARAM_FILE_NAME	Load parameter File	LoadITParm.txt
OVERRIDE_LVL	Override Level	No Default
Progression Counter Size	Optional field. During processing the job writes messages to the log to report on its progress based on the number of records already processed. When this parameter is specified, the value controls the interval at which these progression messages are written to the job log. A specified value should be a positive integer.	500
Select Block Size (SELECT_BLOCK)	This is the block size that is used to select the records to process at a time.	100
User ID	Optional field. The value entered must be valid on the User Information table.	blank

Output

- The failed AD / EF Transactions would have been processed to final or rejected
- AD Cleanup.xml – contains the Internal Exchange Transaction (IT) transactions
- Updates to the Disbursement Request for the rejected AD transactions
- The draft AD / EF transactions would have been discarded from the Transaction Catalog

Batch Return codes

The following table shows the potential job return codes for the AD Clean Up and IT XML Creation job in the AD Chain.

Return Code	Condition
Successful (1)	All of the transactions generated in that run submitted successfully.
Warning (4)	Not Applicable for this job.
Non-Fatal Error (8)	Not Applicable for this job.
Failed (12)	• Parameters are not valid. • Exception file name is not entered. • Restart failed because another instance of the AD chain has already been run successfully. • Runtime exceptions encountered for any unexpected situations. When the job ends with a Return Code of *Failed*, subsequent jobs in the chain will be set to inactive
Terminated (16)	This Return Code will be issued when the job is terminated by the user. When the job ends with a Return Code of *Terminated*, subsequent jobs in the chain will be set to inactive.
System Failure (20)	This Return Code will be issued when the job is terminated because of database server or network issues. When this job ends with a Return Code of *System Failure*, subsequent jobs in the chain will be set to inactive.

Sort Sequence

N/A

Selection Criteria

N/A

Problem Resolution

The following table shows the possible return codes and recommendations for each processing step in the AD Clean Up and IT XML Creation job.

1. Parameter Validation:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the parameters are valid.	N/A	N/A
Warning (4)	N/A	This step does not issue this return code.	N/A
Non-Fatal Error (8)	N/A	This step does not issue this return code.	N/A
Failed (12)	Invalid parameters. **Sample Message:** Invalid Prefix xxx (xxx being the value entered on the batch parameter page)	Correct the parameter and schedule a new job.	
	Required parameters are not entered. **Sample Message:** Exception file is required.	Specify the Input file name and schedule a new job.	
	Input file is not found on the specified directory. **Sample Message:** The system cannot find the specified file.	Make sure that the specified file exists in the directory and restart the job. Alternatively, a new job can be scheduled. Sometimes, the directory path may not be correct. In that case, enter the correct directory path and restart the job. If the job cannot be restarted, schedule a new job.	
	Failed while restarting the job since another instance of the job has already been run successfully. **Sample Message**: Cannot restart the job	Schedule a new job.	

Possible Return Codes	Condition	Recommendation	Other Instructions
	since another instance of this job has already been run successfully.		
	Failed because of runtime exceptions for an unexpected situation.	If the job fails with fatal conditions on encountering unknown exceptions, then investigate the exception reported by the process, resolve the error and restart the job. If the job cannot be restarted, schedule a new job.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can be either restarted or rescheduled.	
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can be either restarted or rescheduled.	

2. Performing AD Cleanup – This step will be performed only if the parameters are valid.

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	N/A	This step does not issue this return code.	N/A
Non-Fatal Error (8)	N/A	This step does not issue this return code.	N/A
Failed (12)	Failed because of runtime exceptions for an unexpected situation.	If the job fails with fatal conditions on encountering unknown exceptions then investigate the exceptions reported by the process, resolve the error and restart the job.	

Possible Return Codes	Condition	Recommendation	Other Instructions
		If the job cannot be restarted, schedule a new job.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can be either restarted or rescheduled.	
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can be either restarted or rescheduled.	

3. Creation of IT Transaction xml file

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	N/A	This step does not issue this return code.	N/A
Non-Fatal Error (8)	N/A	This step does not issue this return code.	N/A
Failed (12)	Failed because of runtime exceptions for an unexpected situation.	If the job fails with fatal conditions on encountering unknown exceptions then investigate the exceptions reported by the process, resolve the error and restart the job. If the job cannot be restarted, schedule a new job.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can be either restarted or rescheduled.	
System	When the job is	The reason for the System	

Possible Return Codes	Condition	Recommendation	Other Instructions
Failure (20)	terminated because of database server or network issues.	Failure needs to be investigated. The job can be either restarted or rescheduled.	

Automated Disbursement Chain: Check Number Assignment Job

Job Name	Check Number Assignment
Recommended Frequency	Daily as part of the nightly cycle or on demand.
Single Instance Required	Yes
Can be restarted?	Yes. Refer to the "Overview" and the "Problem Resolution" sections for more details.
Report Generated	Yes

Overview

This is an optional step in the AD chain that assigns the check numbers to the processed disbursed transactions. When the Check Assignment in AD Chain Processing (BATCH_CHK_ASSIGN) parameter on Application Parameters (APPCTRL) is set to *true*, check numbers will be assigned in this step to disbursement transactions processed successfully in that AD run. If the Check Assignment in AD Chain Processing parameter is set to *true*, then the Journal Posting control on the Transaction Control record for AD transaction codes should be set to *Asynchronous Posting*. This is to avoid the posting of disbursement transactions to the journals before assigning the check numbers.

This process uses the **Next Check Number** of respective **Bank Code** on the Bank table to assign unique Check Numbers for submitted AD transactions. If the **Check Stock Type** of respective Bank Code is set to *Plain* on the Bank table, then only one check number will be assigned for the AD transaction irrespective of the number of associated stub lines.

For EFT payments, if the EFT_NUM_USE_DATE (EFT Number Use Date) parameter on the Application Parameter table is set to *true* (or if a value is not provided), then the **Next Avail EFT No** on the Bank table will use the first 8 digits for the current date and the remainder of the digits a sequence number. If the EFT_NUM_USE_DATE parameter is set to *false*, then the **Next Avail EFT No** on the Bank table will not include the date in the EFT Number, but will use all allowed digits for the sequence number.

If the **Check Stock Type** of respective Bank Code is set to *Preprinted* on the Bank table, a Check Number will be assigned for every 35 stub lines (this number is hard coded) associated with the AD transaction. However, only the first Check Number will be considered the official Check Number and will be posted to the Check Reconciliation table as *Disbursed* or *Paid* (if the **Check Net Amount** is *$0*) whereas other Check Numbers will be posted to the Check Reconciliation table as *Voided*.

The Check Number Assignment job will sort disbursements using the fields specified on the Check/EFT Sort Precedence (CHKSORT) page for the **Disbursement Type**.

The Check Number Assignment job will group selected disbursements optionally based on being a $0 check and/or Disbursement Categories, so that checks may be printed as a sequentially numbered sub-group for holding or other processing. Within each sub-group, the process will sort the disbursements using the Check/EFT Sort Precedence (CHKSORT) table and assign Check and EFT numbers as noted below.

The Check Number Assignment will check the **Assign Overflow Check Numbers For Preprinted** (APPCTRL) parameter to determine if overflow pages will be printed for preprinted stock. If overflow pages should not be printed and the Bank Account uses preprinted stock, the process will not assign another check number to the overflow pages. If overflow pages should be printed and the checks are preprinted, the process will assign another check number for each overflow page.

Sorting logic for the selected disbursement records:

Group 1 – Checks for AD Payments Associated with Non-Held Disbursement Categories that meet the following criteria:

Disbursement Category is not in the list of values for the Assign Check Numbers for the Held Disbursement Categories as the Second Group of Checks parameter AND either of these is true:

- Dollar Amount not equal to $0 OR
- (Dollar Amount = $0) AND (Assign Check Numbers For $0 Checks as the Last Group of Checks parameter is 'N'.)

Group 2 – Checks for AD Payments Associated with Held Disbursement Categories that meet the following criteria:

Disbursement Category is in the list of values for the Assign Check Numbers for the Held Disbursement Categories as the Second Group of Checks parameter AND either of these is true:

- Dollar Amount is not equal to zero ($0) OR
- (Dollar Amount = $0) AND (Assign Check Numbers For $0 Checks as the Last Group of Checks parameter is 'N'.)

Group 3 - Checks for Zero Dollar AD Payments as follows:

- If the **Assign Check Numbers For $0 Checks as the Last Group of Checks** parameter is set to *Yes*, the batch job will select AD transactions with Total Check Amount equal to $0.00.
- If the Assign Check Numbers For $0 Checks as the Last Group of Checks parameter is set to *No*, then this group is empty.

Group 4 – EFT Payments that meet the following criteria:

- Disbursement Type is *EFT* and Check/EFT Number is *blank*.

Within each of the above groups, the disbursements will be sorted by the sorting fields and precedence defined on the Check/EFT Sort Precedence (CHKSORT) table for the Disbursement Type defined on the disbursement. Note: If Sort Field is set to *Reference Transaction Department* on the Check/EFT Sort Precedence (CHKSORT) page, the RFED_DOC_DEPT_CD attribute on AD_DOC_HDR must be populated using Configurable Formula (BOFRMLA). Refer to the *CGI Advantage Financial - Disbursement User Guide* for examples on how to use BOFRMLA to set the value for RFED_DOC_DEPT_CD.

Overflow Processing Logic for Preprinted Check Stock

Within a transaction, the processing logic determines if a check will be a single check number stub or require multiple check numbers (overflow). The process determines if a check requires

more than one check stub (an overflow condition) when **Stub Detail Line Count** (AD Transaction Header) is *greater than* **Maximum Number of Stub Lines** (APPCTRL).

The logic to assign check numbers to Preprinted checks will check the value of the **Assign Overflow Check Numbers For Preprinted** (APPCTRL) when printing overflow check stubs. The logic functions as follows:

- If the **Check Stock Type** for the **Bank Account Code** of the disbursement equals *Preprinted* and the **Assign Overflow Check Numbers For Pre–Printed Stock** (APPCTRL) equals *No*, the process will only assign one check number for the disbursement, including the overflow check stubs (these will not be printed).
- If the **Check Stock Type** for the **Bank Account Code** of the disbursement equals *Preprinted* and the **Assign Overflow Check Numbers For Preprinted Stock** (APPCTRL) equals *Yes*, the process will use the logic associated with the **Maximum Number of Stub Lines** (APPCTRL) to assign check numbers for preprinted checks. For overflow checks, assign a new check number for each additional overflow stub page.

The following table shows the processing steps and the progression messages issued in this job.

Process Steps	Messages
1. Parameter Validation	• Validating Batch Parameters. • Parameters are valid or invalid depending on the Validation. If the parameter is invalid, the invalid value will be displayed in the log. • Batch Parameter validation completed.
2. Assigning Check numbers	• Selecting eligible transactions for assigning check numbers • Total number of transactions selected ‘n’ • Check numbers assigned for ‘n’ transactions where ‘n’ being the progression counter size. • Check numbers assigned on all the selected transactions

Restartability Information

This job can be restarted if failed due to any reason. If the job is restarted, it will start from the beginning, that is, parameter validation. The restart job will only pick up the eligible records and will not pick up the transactions with check numbers even if the transactions fall under the same Disbursement Run ID.

Major Input

- AD / EFT transactions
- Bank Table

Batch Parameters

Parameter	Description	Default Value
AMSPARM	Parameter Location for the Check Assignment Job. (** **Refer to Note**: Assumptions for SWBP on page no. 7)	No Default
Chain Job ID (CHAIN_JOB_ID)	Required Field. This field would be populated with the Chain Job ID when the job runs as part of the AD chain. When the Job is scheduled as a stand-alone job, the Disbursement Run ID stored on the Disbursement transactions should be entered.	$$@CHAINJOBID @$$
Commit Block Size (COMMIT_SIZE)	Required field. Controls how many records are committed by the application at one time. The size should be compatible with technical capabilities and performance guidelines. The default value for this parameter is set to 100. This parameter is editable and the user can set it to a higher number.	100
Check Assignement mode (MODE_NAME)	Required Field. This field refers to whether the process is Check Assignment Mode or Disbursement Correction Mode. Valid values are 0 & 1. If MODE_NAME = 0 then the process runs in the Check Assignment Mode and if MODE_NAME = 1 then the process runs in the Disbursement Correction Mode.	0
Progression Counter Size	Optional field. During processing the job writes messages to the log to report on its progress based on the number of records already processed. When this parameter is specified, the value controls the interval at which these progression messages are written to the job log. A specified value should be a positive integer.	500

Major Output

- Updates to the AD transactions – Check numbers on the Header and Accounting Line
- Updates Bank – Next Available check number
- Update / Insert to the Check Reconciliation
- Updates to the Posting Line Catalog – Check Number gets updated on the Posting Lines

- Updates to the Intercept Activity – Check number gets updated
- Report showing the exceptions encountered during the processing

Batch Return Codes

The following table shows the potential job return codes for the Check Number Assignment job in the AD submitter.

Return Code	Condition
Successful (1)	All of the transactions generated in that run submitted successfully.
Warning (4)	This Return Code will be issued when no eligible transactions are selected for assigning check numbers.
Non-Fatal Error (8)	Not Applicable for this job.
Failed (12)	• Parameters are not valid • Chain ID / Parameter file is not found • Restart failed because another instance of the AD chain has already been run successfully. • Runtime exceptions encountered for any unexpected situations When the job ends with a Return Code of *Failed*, subsequent jobs in the chain will be set to inactive.
Terminated (16)	This Return Code will be issued when the job is terminated by the user. When the job ends with a Return Code of *Terminated*, subsequent jobs in the chain will be set to inactive.
System Failure (20)	This Return Code will be issued when the job is terminated because of database server or network issues. When this job ends with a Return Code of *System Failure*, subsequent jobs in the chain will be set to inactive.

Sort Sequence

Transaction Code, Transaction Department, Transaction ID, Transaction Version

Selection Criteria

Transaction Code, Transaction Department, Transaction ID, Transaction Version

Problem Resolution

The job can be restarted if it is failed. If for some reason, the job cannot be restarted, the new job can be scheduled by disabling other jobs in the AD chain. On the new job, enter the Chain job ID of the failed job as the 'Chain job ID parameter'.

The following table shows the possible return codes and recommendations for each processing step specific to the job.

Step 1: Parameter Validation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	N/A	This step does not issue this return code.	N/A
Non-Fatal Error (8)	N/A	This step does not issue this return code.	N/A
Failed (12)	Invalid parameters **Sample Message:** Commit block size should be a positive integer.	Enter the correct value and schedule a new job.	
	Required parameters are not entered. **Sample Message:** Chain Job ID cannot be blank.	Enter the Chain Job ID and schedule a new job.	
	Failed while restarting the job since another instance of the job has already been run successfully. **Sample Message**: Cannot restart the job since another instance of this job has already been run successfully	Schedule a new job.	
	Failed because of runtime exceptions for an unexpected situation.	If the job fails with fatal conditions on encountering unknown exceptions then investigate the exceptions reported by the process, resolve the error and restart the job. If the job cannot be restarted, schedule a new job.	
Terminated	Job is terminated	The reason for the termination needs to be	

Possible Return Codes	Condition	Recommendation	Other Instructions
(16)	manually by the user.	investigated. The job can either be restarted or rescheduled.	
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can either be restarted or rescheduled.	

Step 2: Assigning check numbers – This step will be performed only when the Parameter validation is successful.

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	N/A	This step does not issue this return code.	N/A
Non-Fatal Error (8)	N/A	This step does not issue this return code.	N/A
Failed (12)	Failed while restarting the job since another instance of the job has already been run successfully. **Sample Message**: Cannot restart the job since another instance of this job has already been run successfully.	Schedule a new job.	
	Failed because of runtime exceptions for an unexpected situation.	If the job fails with fatal conditions on encountering unknown exceptions then investigate the exceptions reported by the process, resolve the error and restart the job. If the job cannot be restarted, schedule a new job.	

Possible Return Codes	Condition	Recommendation	Other Instructions
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can either be restarted or rescheduled.	
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can either be restarted or rescheduled.	

Automated Disbursement Chain: IT Upload Job

Job Name	IT Upload
Recommended Frequency	Daily as part of the nightly cycle or on demand.
Single Instance Required	No
Can be restarted?	Yes. Refer to the "Overview" and the "Problem Resolution" sections for more details.
Report Generated	No. All of the exceptions are written to the error file.

Overview

The Intercept Transaction (IT) Upload job in the AD Chain loads the records from the AD Cleanup XML File, generated during the AD Clean Up and IT XML Creation job, into the Transaction Catalog. This job uses the common utility to load the records into the Transaction Catalog. This job first validates the batch parameters. If the parameters are valid, then it loads the records into the Transaction Catalog. If the parameters are not valid, the job issues appropriate messages and ends with a status of Failed. Once the records are loaded into the Transaction Catalog, the summary information is written into the log as how many records were in the input file and how many records loaded successfully.

The job can be restarted if it is failed provided the Save Restart Information parameter is selected. If the failure occurred after the parameter validation, then the job should be restarted after resolving the errors. If the Save Restart Information parameter is not selected or if the restart is not the immediate option, then the new job can be rescheduled but before rescheduling the job, the transactions loaded by the failed job should either be processed or discarded so that they do not remain on the Transaction Catalog.

Major Input

- AD Transaction XML file

Batch Parameters

Parameter	Description	Default Value
Parameter file (PARM_FILE)	Parameter file to Load Transactions	$$AMSPARM$$/LoadITParm.txt

Note: This PARM_FILE contains only the following subset of SMU parameters.

Parameter	Default Value
Action Code (ACTN_CD)	171
Commit Block Size (COMMIT_BLOCK)	100
Error File Name (ERROR_FILE_NM)	$$AMSLOGS$$/ADUpLoadError.txt
File Name To Be Imported (FILE_NM)	$$AMSEXPORT$$/ADDocument.xml
Apply Overrides (APPLY_OVERRIDES)	True

Refer to the “SMU Transaction Upload Job” run sheet in the *CGI Advantage Financial – Utilities Run Sheets* guide for the full list of SMU Transaction Upload batch parameters.

Major Output

- Intercept Transactions in draft version

Batch Return Codes

The following table shows the potential job return codes for the IT Upload job.

Return Code	Condition
Successful (1)	All of the records are loaded into the Transaction Catalog successfully or the input file is empty.
Warning (4)	This Return Code will be issued when some of the records failed to load whereas all other records were loaded successfully.
Non-Fatal Error (8)	None of the records get loaded into the Transaction Catalog.
Failed (12)	• Parameters are invalid • When the input file is not found in the specified directory • Restart failed because another instance of the AD chain has already been run successfully • Runtime exceptions encountered for any unexpected

Return Code	Condition
	situations When the job ends with a Return Code of *Failed*, subsequent jobs in the chain will be set to inactive.
Terminated (16)	This Return Code will be issued when the job is terminated by the user. When the job ends with a Return Code of *Terminated*, subsequent jobs in the chain will be set to inactive.
System Failure (20)	This Return Code will be issued when the job is terminated because of database server or network issues. When this job ends with a Return Code of *System Failure*, subsequent jobs in the chain will be set to inactive.

Sort Sequence

N/A

Selection Criteria

N/A

Problem Resolution

If the job ends with a return code of Failed and above, the job can be restarted only when the Save Restart Information parameter is selected and another instance of the job has not been scheduled and run successfully. If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – a new job should only be scheduled. In this case the IT Generated indication on the corresponding INCT_DISB_RQST table should be unchecked so that IT transactions can be generated for those records again. The INCT_DISB_RQST records can be identified by selecting the AD / EFT transactions generated during the failed AD chain (Chain ID can be used select records from the transaction catalog since Chain ID is populated as the Run ID on the transactions) from the Transaction Catalog and matching the selected transactions against the INCT_DISB_RQST table.

If the job cannot be restarted immediately, then the transactions loaded by this job should either be processed or discarded before rescheduling the new job. The loaded transaction can be processed either manually or by submitting the IT Submit job in the AD chain. Loaded transactions can only be discarded manually. The parameter file generated in the AD Clean Up and IT XML Creation job will have the transaction IDs and that can be used for the manual processing or discarding of the transaction.

The following table shows the possible return codes and recommendations for each processing step specific to the job in the chain. For general errors and recommendations, refer to the “SMU Transaction Upload Job” run sheet in the *CGI Advantage Financial – Utilities Run Sheets* guide.

Possible Return Codes	Condition	Recommendation	Other Instructions
Failed (12)	Failed while restarting the job since another instance of the job has already been run	**Recommendation:** Schedule a new job.	

Possible Return Codes	Condition	Recommendation	Other Instructions
	successfully. **Sample Message**: Cannot restart the job since another instance of this job has already been run successfully.		

Automated Disbursement Chain: IT Submit Job

Job Name	IT Submit
Recommended Frequency	Daily as part of the nightly cycle or on demand.
Single Instance Required	No
Can be restarted?	Yes. Refer to the "Overview" and the "Problem Resolution" sections for more details.
Report Generated	No. All of the exceptions are written to the error file.

Overview

The IT Submit job uses the SysManUtil utility to submit each IT transaction loaded by the IT Upload job for final processing to update the tables, through existing logic, and complete the AD chain.

Major Input

- SMU job parameter file
- Draft Disbursement Transactions

Batch Parameters

Parameter	Description	Default Value
Parameter File (PARM_FILE)	Required field. This file contains the Transaction IDs to be processed.	$$AMSPARM$$/SubmitITParm.txt
Exception File Name (EXCEP_REP_FILE_NM)	Required field that defines what file the program is to use to create an exception report.	$$AMSLOGS$$/ITExep.txt

Note: This PARM_FILE contains only the following subset of SMU parameters.

Parameter	Default Value
Action Code (ACTN_CD)	Transaction Submit
Save Restart Information? (RESTART_FL)	1
Transaction Code (DOC_CD)	Transaction Code parameter value entered on the AD Clean Up and IT XML Creation job
Transaction Department (DOC_DEPT)	Transaction Department parameter value entered on the AD Clean Up and IT XML Creation job
Transaction Identifier (DOC_ID)	Transaction Prefix parameter value entered on the AD Clean Up and IT XML Creation job followed by asterisk (*)
Transaction Status (DOC_STA)	1
Transaction Phase (DOC_PHASE_CD)	1

For the full list of available parameters for the SMU submit job, refer to the “SMU Transaction Submit Job” run sheet in the *CGI Advantage Financial – Utilities Run Sheets* guide.

Major Output

The Transactions would have been processed to final or rejected.

Batch Return Codes

The following table shows the potential job return codes for the IT submit job in the AD Chain.

Return Code	Condition
Successful (1)	All of the transactions generated in that run submitted successfully.
Warning (4)	Not Applicable for this job.
Non-Fatal Error (8)	Not Applicable for this job.
Failed (12)	• Input parameter file is not found. • Restart failed because another instance of the AD chain has already been run successfully. • Runtime exceptions encountered for any unexpected

Return Code	Condition
	situations. When the job ends with a Return Code of *Failed*, subsequent jobs in the chain will be set to inactive.
Terminated (16)	This Return Code will be issued when the job is terminated by the user. When the job ends with a Return Code of *Terminated*, subsequent jobs in the chain will be set to inactive.
System Failure (20)	This Return Code will be issued when the job is terminated because of database server or network issues. When this job ends with a Return Code of *System Failure*, subsequent jobs in the chain will be set to inactive.

Sort Sequence

N/A

Selection Criteria

N/A

Problem Resolution

If the job ends with a return code of *Failed* and above, the job can be restarted only when the Save Restart Information parameter is selected and another instance of the job has not been scheduled and run successfully. If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – a new job should only be scheduled.

If the restart is not an immediate option and the fatal error is because of a few transactions, the rest of the transactions can be submitted manually. The Transaction IDs can be found on the input parameter file.

The following table shows the possible return codes and recommendations for each processing step specific to the job in the chain. For general errors and recommendations, refer to the "SMU Transaction Submit Job" run sheet in the *CGI Advantage Financial – Utilities Run Sheets* guide.

Possible Return Codes	Condition	Recommendation	Other Instructions
Failed (12)	Failed while restarting the job since another instance of the job has already been run successfully. **Sample Message**: Cannot restart the job since another instance of this job has already been run successfully.	Schedule a new job.	

2.1.2 Automated Interest Calculation Process

Chain or Job Name	Automated Interest Calculation
Recommended Frequency	Nightly
Single Instance Required	Yes
Can be restarted?	Yes
Reports generated	N/A

Overview

The Automated Interest Calculation Chain job is used to calculate interest amounts and creates a XML after eligibility and table verifications from payment requests based on the Interest Ineligible indication and validations against the Interest Exceptions, Payment Scheduling, and Interest Control reference pages. The created XML is uploaded and submitted to the final state, if necessary. The chain job consists of the following batch jobs. They are as follows:

1. **IPR XML Creation:** This is the first job in the chain. This process calculates the interest for both Automatic and Manual Disbursement transactions on each accounting line provided they meet a certain criteria. The output of this process will be a XML file representing the new IPR transaction for each accounting line for which interest was calculated. It will also insert into the R_INTR_PYMT_INQ table with records for every Accounting line for which a XML was created.
2. **IPR Upload**: This is the second job in the chain job. The main purpose of this job is to upload the XML file created from the previous step.
3. **IPR Submit**: This is the third job in the chain job. The main purpose of this job is to submit the transactions uploaded using XML file created in first step.

Major Input

Data from the following tables/transactions:

- Automated and Manual Disbursement Transactions
- Disbursement Cash Management Options (R_DISB_CSH_MGMT)
- Interest Exception (R_INTR_EXP)
- Interest Accounting Line Crosswalk (R_INTR_ACTG_CRSWLK)
- Payment Hold Activity (R_PYMT_HLD_ACT)
- Vendor/Customer (R_VEND_CUST)
- Payment Scheduling and Interest Control (R_PSIC)
- Automatic Transaction Numbering (AUTO_DOC_NO)

Major Output

Following are the output of the chain job:

- XML file for Interest Payment Request Transactions
- Interest Inquiry (R_INTR_PYMT_INQ)
- Interest Payment Request Transaction

Chain Return Code

The following table indicates the potential return codes with which the chain will end:

Return Code	Condition
Successful (1)	All the jobs ends successfully
Warning (4)	One of the job in the chain ends with a return code of *Warning*
Failed (12)	One of the job in the chain ends with a return code of *Failed*
Terminated (16)	One of the job in the chain ends with a return code of *Terminated*
System Failure (20)	One of the job in the chain ends with a return code of *System Failure*

Problem Resolution

If any of the jobs in the chain failed due to application errors it is advisable to restart the job after correcting the errors instead of rescheduling the job. Restarting the job will reduce the processing time since the job will resume from where it has last committed and select only the unprocessed records.

Please refer to the individual jobs for details regarding the specific job processes and problem resolution.

Automated Interest Calculation Batch Process: IPR XML Creation

Chain or Job Name	IPR XML Creation
Recommended Frequency	See Automated Interest Calculated Chain
Single Instance Required	Yes
Can be restarted?	Yes
Reports generated	None

Overview:

The IPR XML Creation Process performs the interest calculation for every eligible accounting on AD and MD transactions. The eligibility is determined based on certain criteria.

Process Steps	Messages
1. Validate the Parameters	• Started Validating the Parameters • If parameters are invalid, then write the invalid

Process Steps	Messages
	parameter to job log
2. Selection of records	• Started selection of records • If the selection returns 0 records, then the following message will be issued: “No eligible record found”.
3. Calculate Interest	• Calculate Interest started. • Interest calculation completed.
4. Insert record	• Inserting the record into Inquiry Table started. • Inserting of record into Inquiry Table completed.
5. Create XML	• Writing error XML started. • Writing error XML completed.

Restartability

Job can be restarted. If the job fails then the job can be restarted at the first step. The Interest Calculated or Bypassed accounting line indication is used to determine if the disbursement transaction has been processed or not. The chain job should start processing only those records which do not have this indication set to *false*.

Major Input

- Automated and Manual Disbursement Transactions
- Disbursement Cash Management Options (R_DISB_CSH_MGMT)
- Interest Exception (R_INTR_EXP)
- Interest Accounting Line Crosswalk (R_INTR_ACTG_CRSWLK)
- Payment Hold Activity (R_PYMT_HLD_ACT)
- Vendor/Customer (R_VEND_CUST)
- Payment Scheduling and Interest Control (R_PSIC)
- Automatic Transaction Numbering (AUTO_DOC_NO)

Batch Parameters

Parameter	Description	Default Value
Parameter Folder Path AMSPARM	Parameter Folder Path	$$AMSPARMS$$
Commit Block Size COMMIT_SIZE	When the number of records that have been processed equals this parameter, a database commit is done.	100

Export/Import Directory AMSEXPORT	Export/Import Directory Location	$$AMSROOT$$/ExportImport
Bypass Budget Edits BYPASS_BUD_EDT	Bypass Budget Edits? Valid Values: 1/0	0
Bypass Cash Edits Flag BYPASS_CASH_EDT	By Pass Cash Edits? Valid Values: 1/0	0
Bypass Fund Edits BYPASS_FUND_EDT	By Pass Fund Edits? Valid Values: 1/0	0
End Date in MM/DD/CCYY Format END_DATE	End Date in MM/DD/CCYY Format	Today's Date if left blank
Comma Separated List of Crosswalk Fund Values FUND_CROSSWLK	Comma Separated List of Crosswalk Fund Values	
Include PR Info in Check Description? INCL_PR_INFO_CHK_DSCR	Include Payment Request (PR) Info in Check Description? Valid Values: 1/0	0
Include Internal Vendor Flag INC_INT_VEND	Include Internal Vendor? Valid Values: 1/0	0
Include Miscellaneous Vendor Flag INC_MISC_VEND	Include Miscellaneous Vendor? Valid Values: 1/0	0
IPR Transaction Code IPR_DOC_CD	Interest Payment Request (IPR) Transaction Code	
IPR Transaction Department IPR_DOC_DEPT	IPR Transaction Department	
Prefix of the Transaction Code Field on the IPR Transaction IPR_DOC_PREFIX	Prefix of the Transaction Code Field on the IPR Transaction	
IPR Transaction Unit Code IPR_DOC_UNIT_CD	IPR Transaction Unit Code	
Progression Counter	Progression Counter	100

PROG_CNTR		
SEL BLK SIZE SEL_BLK_SIZE	SEL BLK SIZE	100
Start Date in MM/DD/CCYY Format START_DATE	Start Date in MM/DD/CCYY Format	Today's Date if left blank
Use PR Transaction Department/Unit? USE_PR_DOC_DEPT	Use PR Transaction Department/Unit? Valid Values: 1/0	0

Major Output

- IPR XML transaction
- Records Inserted into the R_INTR_PYMT_INQ

Job Return Code

The following table shows the potential job return codes for the Inferences & Validations job:

Return Code	Condition
Successful (1)	All the records in the XML file are processed successfully
Warning (4)	This job does not issue this Return Code.
Non-Fatal Error (8)	This job does not issue this Return Code.
Failed (12)	The job will fail under the following conditions: • Parameters are invalid • Issues in SysManUtil while discarding the transaction • Technical/System failure
Terminated (16)	This Return Code will be issued when the job is terminated by the user. When this job ends with a return of code *Terminated* subsequent jobs in the chain will be set to inactive.
System Failure (20)	This Return Code will be issued when the job is terminated because of database server or network issues. When this job ends with a return of code *System Failure*, subsequent jobs in the chain will be set to inactive.

Sort Sequence

N/A

Selection Criteria

N/A

Sort Sequence

N/A

Problem Resolution

The following table shows the possible return codes and recommendations for each processing step.

Parameter Validation

- Verify the Calculate Interest Outside of Disbursement indication on System Options.
- Verify the Transaction Code
- Verify the Transaction Department
- Verify the Combination of entered Transaction prefix, Transaction Code and Transaction Department exists on the Automatic Transaction Numbering
- Verify the Transaction Unit
- If all the parameters are validated successfully then retrieve the records from the temporary holding table

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All the parameters are validated successfully.	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Job failed due to Fatal conditions	In this step, the job can fail under the following conditions. 1) Encounters any runtime exceptions and 2) Failed during restart. If the job fails because of the runtime exceptions, investigate the exception reported by the process, resolve the error and restart the job.	Schedule a new job

Possible Return Codes	Condition	Recommendation	Other Instructions
Terminated (16)	Job is terminated manually by the user.	Reason for the termination needs to be investigated. A new job can be scheduled	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.
System Failure (20)	When the job is terminated because of database server or network issues	Reason for the System Failure needs to be investigated. A new job can be scheduled.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.

Step 2: Selecting the records

- Select the both Automatic and Manual Disbursement records
- Verify Inclusion of Miscellaneous or Internal or PCard Vendors on the Disbursements using parameters
- Verify Interest Exceptions
- Verify the Payment Scheduling and Interest Control Exceptions.
- If the above verifications go through choose those records for next step

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	N/A	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	There are no records in the AD and MD tables for which Interest need to be calculated	N/A	Write a message to the job log “Log the message ‘No Record for Interest Calculation’”

	Failed because of runtime exceptions for an unexpected situation.	Failure reason needs to be investigated before scheduling a new job.	Failed because of runtime exceptions for an unexpected situation.
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated and a new job scheduled.	N/A
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated and a new job scheduled.	N/A

Step 3: Calculate Interest

- For the records selected from the above step calculate interest by determining the Interest Eligible Days, Interest Rate, Line Amount and the Number of Calendar Days.
- Consolidate the amount to determine if the threshold has been exceeded.
- If Threshold was exceeded then proceed to the next step for all Accounting Lines including the ones where threshold was not exceeded previously.

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	The Interest has been calculated successfully.	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	Errors that can be generated due to some failed validations while calculation of Interest	N/A	Write the error to the Job Log with all the necessary information like the Transaction ID and Accounting line Information
Failed (12)	Errors that can be generated due to some failed validations while calculation of Interest	Investigate the reason for the Interest not being calculated and re-run the chain.	Write the error to the Job Log with all the necessary information like the Transaction ID and Accounting line Information

	Failed because of runtime exceptions for an unexpected situation.	Failure reason needs to be investigated before scheduling a new job.	N/A
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated and a new job scheduled.	N/A.
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated and a new job scheduled.	N/A.

Step 4: Inserting the Record

- Insert into the R_INTR_PYMT_INQ table with all the Payment Request, Disbursement and Interest Information.

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	The record has been successfully inserted.	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	Errors that can be generated due to some failed validations while updating the corresponding record on the Interest payment inquiry table	N/A	Get the error thrown by the Interest Payment inquiry and then write it to the exception report.
Failed (12)	Failed due to issues in reading the data row	Investigate the reason for the data row not being read and re-run the chain.	N/A
	Failed because of runtime exceptions for an unexpected situation.	Failure reason needs to be investigated before scheduling a new job.	N/A
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated and a new job scheduled.	N/A.
System Failure (20)	When the job is terminated because of database server or	The reason for the System Failure needs to be investigated and a new job	N/A.

	network issues.	scheduled.	

Step 5: Create XML file

- For every accounting line for which the interest is applied a corresponding XML will be created. This XML will contain the Interest transaction information.

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	The XML has been successfully created.	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Failed due to issues in creating the report.	Investigate the reason for the report not being created and re-run the chain.	Write the error on the Job Log will all the necessary Information Like the Transaction ID, Vendor and Accounting Line Information
	Failed because of runtime exceptions for an unexpected situation.	Failure reason needs to be investigated before scheduling a new job.	N/A
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated and a new job scheduled.	N/A.
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated and a new job scheduled.	N/A.

Automated Interest Calculation Batch Process: IPR Upload

Job Name	IPR Upload Job
Recommended Frequency	See Automated Interest Calculated Chain
Parallel processing enabled	N/A

Can the job be restarted?	Optionally, based on the Save Restart Information parameter.
Exception report produced	No. All of the exceptions are only written to the log.

Overview

The Interest Payment Request (IPR) Upload job loads the records from the XML File (generated by the first job of the chain process) into the Transaction Catalog. This job uses the common utility to load the records into the Transaction Catalog.

Major Input

- IPR Transaction XML file

Batch Parameters

Parameter	Default Value
Action Code : Import (ACTN_CD)	171
Commit Block Size (COMMIT_BLOCK)	100
XML File Name (FILE_NM)	$$AMSEXPORT$$/ AutomatedInterestDoc_$$@CHAINJOBID@$$.xml
Transaction Status (DOC_STA_CD)	1
Export/Import Dir	AMSROOT$$/ExportImport
Bypass Auto Numbering (BYPS_ADNT_FL)	True

Please refer to the SMU Transaction Upload Job run sheet in the *CGI Advantage Financial – Utilities Run Sheet Guide* for the full list of SMU Transaction upload batch parameters.

Major Output

- IPR Transactions in draft version

Batch Return Codes

The following table shows the potential job return codes for the IPR Upload job.

Return Code	Condition
Successful (1)	All of the records are loaded into the Transaction Catalog successfully or the input file is empty.

Warning (4)	This return code will be issued when some of the records failed to load whereas all other records were loaded successfully.
Non-Fatal Error (8)	None of the records get loaded into the Transaction Catalog.
Failed (12)	• Parameters are invalid • When the input file is not found in the specified directory • Restart failed because another instance of the Automated Interest Calculation Batch Process chain has already been run successfully • Runtime exceptions encountered for any unexpected situations When the job ends with a Return Code of *Failed*, subsequent jobs in the chain will be set to inactive.
Terminated (16)	This Return Code will be issued when the job is terminated by the user. When the job ends with a Return Code of *Terminated*, subsequent jobs in the chain will be set to inactive.
System Failure (20)	This Return Code will be issued when the job is terminated because of database server or network issues. When this job ends with a Return Code of *System Failure*, subsequent jobs in the chain will be set to inactive.

Sort Sequence

N/A

Selection Criteria

N/A

Problem Resolution

If the job ends with a Return Code of *Failed* and above, the job can be restarted only when the Save Restart Information parameter is selected and another instance of the job has not been scheduled and run successfully. If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted a new job should only be scheduled.

If the job cannot be restarted immediately, then the transactions loaded by this job should be either processed or discarded before rescheduling the new job. To process the loaded transactions, submit the IPR Submit job in that chain. This job would be set to Inactive since the IPR Upload failed. The Submit job will submit the transactions that were generated during that Automated Interest Calculation Batch Process chain. To discard the loaded transactions, submit the IPR Clean Up. The IPR Clean Up will discard all records inserted in the Inquiry table for which the submission was not successful.

The following table shows the possible return codes and recommendations for each processing step specific to the job in the chain. For general errors and recommendations, refer to the SMU Transaction Upload run sheet in the *CGI Advantage Financial – Utilities Run Sheet Guide.*

Possible Return Codes	Condition	Recommendation	Other Instructions
Failed (12)	Failed while restarting the job since another instance of the job has already been run successfully. **Sample Message**: Cannot restart the job since another instance of this job has already been run successfully.	**Recommendation:** Schedule a new job.	

Automated Interest Calculation Chain: IPR Submit Job

Job Name	IPR Submit
Recommended Frequency	See Automated Interest Calculated Chain
Single Instance Required	Yes
Can be restarted?	Yes. See the Overview and the Problem Resolution sections for more details.
Report Generated	No. All of the exceptions are written to the error file.

Overview

This job submits the transactions listed in the input parameter file that were generated by the CalculateInterest XML Creation job. This process is a part of the Automated Interest Calculation Batch Process Chain batch job that reads SMU job parameter file perform Submit job. It will also update the Interest Payment Request Inquiry Table with the IPR information for transactions submitted successfully.

Major Input

- SMU job parameter file
- Draft IPR Transactions
- Interest Payment Request Inquiry (R_INTR_PYMT_INQ)

Batch Parameters

Parameter	Description	Default Value
BYPASS_WRKFLW_APPRV	By pass Approvals Flag	0
PARM_FILE_NM	Parameter File Name	$$AMSPARM$$/AutoIntr_ParamFile_$$@CHAINJOBID@$$.txt

Major Output

The Transactions would have been processed to final (or pending for approval if workflow set up is added) or rejected.

Batch Return codes

The following table shows the potential job return codes for the individual AD submit job in the AD submitter.

Return Code	Condition
Successful (1)	All of the transactions generated in that run submitted successfully.
Warning (4)	Not Applicable for this job.
Non-Fatal Error (8)	Not Applicable for this job.
Failed (12)	• Input parameter file is not found. • Restart failed because another instance of the Automated Interest Calculation Batch Process chain has already been run successfully. • Runtime exceptions encountered for any unexpected situations. When the job ends with a Return Code of *Failed*, subsequent jobs in the chain will be set to inactive.
Terminated (16)	This Return Code will be issued when the job is terminated by the user. When the job ends with a Return Code of *Terminated*, subsequent jobs in the chain will be set to inactive.
System Failure (20)	This Return Code will be issued when the job is terminated because of database server or network issues. When this job ends with a Return Code of *System Failure*, subsequent jobs in the chain will be set to inactive.

The following table shows the potential job return codes for the IPR submitter job.

Return Code	Condition
Successful (1)	All of the individual jobs ran successfully.
Warning (4)	One of the jobs failed with the Return Code of *Warning*.
Non-Fatal Error (8)	One of the jobs failed with the Return Code of *Non-Fatal Error*.
Failed (12)	One of the jobs failed with the Return Code of *Failed*.
Terminated (16)	One of the jobs failed with the Return Code of *Terminated*.

System Failure (20)	One of the jobs failed with the Return Code of *System Failure*.

Sort Sequence

N/A

Selection Criteria

N/A

Problem Resolution

If the job ends with a return code of *Failed* and above, the job can be restarted only when the Save Restart Information parameter is selected and another instance of the job has not been scheduled and run successfully. If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted a new job should only be scheduled.

If the restart is not an immediate option and the fatal error is because of a few transactions, the rest of the transactions can be submitted manually or discarded manually depending on the Issue. The Transaction IDs can be found on the input parameter file.

The following table shows the possible return codes and recommendations for each processing step specific to the job in the chain. For general errors and recommendations, refer to the SMU Transaction Submit Job run sheet in the *CGI Advantage Financial – Utilities Run Sheet Guide*.

Possible Return Codes	Condition	Recommendation	Other Instructions
Failed (12)	Failed while restarting the job since another instance of the job has already been run successfully. **Sample Message**: Cannot restart the job since another instance of this job has already been run successfully.	Schedule a new job.	

2.1.3 Automated Payment Hold

Chain or Job Name	Automated Payment Hold
Recommended Frequency	Nightly
Single Instance Required	Yes
Can be restarted?	Yes
Reports generated	Reports are generated in this batch job: • Excluded Payment Requests Exception Report • Payment Hold Maintenance Exclusion Report

Overview

This batch process selects payment request lines from Disbursement Request to determine the corresponding active request/removal payment hold records from the Payment Hold Maintenance reference page and applies a Hold Status at different Hold levels and identifies the highest priority Hold Type and sets this as the new Current Hold Type and Hold Level. It also updates the Payment Hold Activity inquiry page to close the hold and update the number of days held.

Steps for running the process:

1. Parameter Validation
2. Clean-Up
3. Closing or Applying
4. Update the Current Hold Level step:
    - Select all the records from Disbursement request (DISRQ) table where the Vendor or Award or Payment Request hold types are not Null and **Allow Bypass Requests** flag is set to *No* and Payments are not to a P-Card vendor and LN_AM is greater than zero and the Vendor is Non-Miscellaneous Vendor and USER_HLD_FL is *False* and DISB_MGMT_HLD_FL is *False*.
    - The process will find the Hold with the highest priority among the non-null values for Vendor, Award, and Payment Hold Type by comparing the Priority Hold Order for each Hold type.
    - The process will then determine if the record should be excluded.
    - Search the Payment Hold Exclusion table for matching records.

If a match is found for any of the below fields on the DISRQ record then an exact match is done for all the other non-null values on the record. If all the non-field values also match then the DISRQ record is considered for exclusion.

> The exclusion fields to compare to the Disbursement Request record are: Fiscal Year, Taxpayer ID, TIN Type, Vendor, Payment Request Transaction Code ,Payment Request Transaction Department, Payment Request Transaction ID, Award Transaction Code ,Award Transaction Department, Award Transaction ID, Department, Unit, Fund, Object, Appropriation, Sub Fund, Sub Object, Activity, Program, Object Class, Object Category, Object Type, Object Group.

If a match is found on the exclusion table then a record is added to the Excluded Payment Requests Exception Report with the message “Disbursement request excluded because it matched an exclusion record on the Payment Hold Exclusion table”

For each Exclusion table record found, that matches the current DISRQ record, write a record to the Payment Hold Maintenance Exclusion report identifying the matched Auto-Generated ID Number and the DISRQ record data.

- If any exclusion record was matched, the record should not be held and the following updates should be made to the Payment Hold Activity Table and Payment Hold Maintenance Table.

Update Payment Hold Activity table.

i. If Priority Hold is not equal to the Current Hold Type, update Payment Hold Activity table for the old Current Hold to show it was replaced by setting the Hold Removal date to current application date.

ii. Insert a record in Payment Hold Activity Table with the same values as the old Current Hold and set the Hold type and hold level to the highest priority Hold.

Update Payment Hold Maintenance table

i. Only if Highest Priority Hold Type not equal to Current Hold Type (hold was not applied before) update PHLDM for the old Current Hold Update to show it was replaced by setting the Last Current Hold Removal Date to current application control date

ii. Set the Current Hold Type and Current Hold Level to null on the DISRQ record

- If the record was not excluded, compare the highest Priority Hold and the Current Hold Type and if the highest Priority Hold is not equal to Current Hold Type then perform these updates:

Update Payment Hold Activity table.

i. If Priority Hold is not equal to the Current Hold Type, update Payment Hold Activity table for the old Current Hold to show it was replaced by setting the Hold Removal sate to current application date.

ii. Insert a record in Payment Hold Activity Table with the same values as the old Current Hold and set the Hold type and hold level to the highest priority Hold.

Update Payment Hold Maintenance table

i. Update PHLDM for the old Current Hold to show it was replaced by setting the Last Current Hold Removal Date to current application control date

ii. Update PHLDM for the Highest Priority Hold to show it was replaced by setting the Hold Request date and Last Current Hold Request Date to current application control date

Set the Current Hold Type and Current Hold Level to the highest priority hold type and hold level on the DISRQ record

The table below lists the progression messages that indicate the status of the job for each processing step.

Process Steps	Messages
1. Parameter Validation	• Validating Batch Parameters • Parameters are valid or invalid depending on the Validation. If the parameter is invalid, the invalid value will be displayed in the log. • Parameter validation completed
2. Clean Up Step	• Selecting eligible records for clean up. • If the selection returns 0 records, then the following message is issued: “No eligible record found for clean up”. • Number of records (count) cleaned is displayed. • At the end, the following message are issued: “Clean up of the records completed.”
3. Close or Applying	• Selecting eligible records for updating individual hold types and closing the low priority records. • If the selection returns 0 records, then the following message are issued: “No eligible records found to update”. • Number of records (count) processed is displayed. • At the end, the following messages are issued: • Updating of the records is completed. • Number of Holds Applied = XX (xx refers to the actual count) • Number of Holds Removed = XX (xx refers to the actual count)
4. Update the Current Hold Level step	• Selecting eligible records for the Current Hold Level update. • If the selection returns 0 records, then the following message are issued: “No eligible record found for the Current Hold Level update”. • Number of records (count) for which the Current Hold Level is updated is displayed. • At the end, the following message is issued: Update of the Current Hold Level completed.

Restartability

- This job has three steps and hence while implementing the restartability, the job first checks at which step did the job previously abort. Before actually setting the hold types, the job performs clean up steps and hence if the job is rescheduled after if it got aborted previously,

there won't be any discrepancies. The default commit block size is hundred and can be altered through the parameters.

- The job should be run in a single thread mode and will run after the Select Payment Hold by TIN batch process and before the baseline AD Chain batch process.

Major Input

- Disbursement Request (DISRQ) Table
- Vendor Customer (VCUST) Table
- Contact Table
- Department Table
- Disbursement Hold Exclusion Table.
- Payment Hold Options Table.
- Payment Hold Maintenance Table.
- Payment Hold Activity Table.
- Payment Hold Type Table.
- Discuss the batch parameters (see table below – the second row is provided for illustrative purposes only)
- The inputs mentioned in this section should be specific to that job only.

Parameter	Description	Default Value
COMMIT_BLK_SZ	Commit Block Size	100
SELECT_BLK_SZ	Select Block Size	100
PROG_CNTR	Progression Counter	100

Major Output

- Disbursement Request (DISRQ) Table.
- Payment Hold Maintenance Table.
- Payment Hold Activity Table.
- Payment Hold Maintenance Exclusion Report.
- Excluded Payment Requests Exception Report.

A record is inserted in the exception report when a DISRQ record satisfies more than one exclusion criteria, which means that one DISRQ record is matched with more than one record in the Payment Hold Exclusion table.

Job Return code

The following table shows the potential job return codes for the job:

Return Code	Condition
Successful (1)	All the selected payment records are processed successfully.
Warning (4)	N/A
Non-Fatal Error (8)	N/A
Failed (12)	The job fails under the following conditions: • Parameters are invalid • Run time exceptions for unexpected situations. • Restart failed because another instance of the Automated Payment Hold Batch Process has already been run successfully.
Terminated (16)	This return code is issued when the job is terminated by the user. When this job ends with a return of code Terminated subsequent jobs in the chain will be set to inactive.
System Failure (20)	This return code is issued when the job is terminated because of database server or network issues. When this job ends with a return of code System Failure, subsequent jobs in the chain will be set to inactive.

Sort Criteria

N/A.

Selection Criteria

At the clean up step:

Select a DISRQ record where Vendor, Award and Payment Request hold types are null but Current Hold Level and Current Hold type values are not null and

- **Allow Bypass Requests** flag is set to *No*.
- Payments to a non -P-Card.
- Line amount is greater than zero.
- A Miscellaneous Vendor is not used.
- User Hold is *False*.
- Disbursement Management Hold is *False.*

At the Closing or Applying Step:

Select a DISRQ record where:

- **Allow Bypass Requests** flag is set to *No*.
- Payments to a non -P-Card.

- Line amount is greater than zero.
- A Miscellaneous Vendor is not used.
- User Hold is *False*.
- Disbursement Management Hold is *False*.

At the Update the Current Hold Level step**:**

Select a DISRQ record where Vendor, Award or Payment Request hold types are not null and

- **Allow Bypass Requests** flag is set to *No*.
- Payments to a non -P-Card.
- Line amount is greater than zero.
- A Miscellaneous Vendor is not used.
- User Hold is *False*.
- Disbursement Management Hold is *False*.

Problem Resolution

The following table shows the possible return codes and recommendations for each processing step.

Step 1: Parameter Validation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All the parameters are validated successfully	N/A	N/A
Warning (4)	This process step does not end with this Return Code.	N/A	N/A
Non-Fatal Error (8)	This process step does not end with this Return Code.	N/A	N/A
Failed (12)	Job failed due to Fatal conditions	In this step, the job can fail under the following two conditions. 1) Encounters any runtime exceptions and 2) Failed during restart. If the job fails because of the runtime exceptions, investigate the exception reported	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.

Possible Return Codes	Condition	Recommendation	Other Instructions
		by the process, resolve the error and restart the job.	
	Failed while restarting the job since another instance of the job has already been run successfully. **Sample Message**: Cannot restart the job since another instance of this job has already been run successfully.	Schedule a new job.	
Terminated (16)	Job is terminated manually by the user.	Reason for the termination needs to be investigated. The job can either be restarted or schedule a new job.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.
System Failure (20)	When the job is terminated because of database server or network issues	Reason for the System Failure needs to be investigated. The job can either be restarted or schedule a new job.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.

Step 2: CleanuCleanup Step:

Selects records from Disbursement Request where:

- Vendor, Award and Payment Request hold types are blank
- Current Hold Level and Current Hold values are not blank
- Allow Bypass Requests indication is false
- Payments are not to a P-Card vendor
- Line Amount is greater than $0.00 (not a credit memo)
- Vendor is Non-Miscellaneous
- User Hold indication is false
- Disbursement Management Hold is false

Records selected to this point are checked matching records exist on Payment Hold Maintenance based on the current hold level of each record. If no matching record exists the job will clear the Current Hold Level and Current Hold Type on Disbursement Request and writes this record to the exclusion report.

If a matching record is found with a Hold action of *Remove Hold*, then the job will:

1. Payment Hold Maintenance – set the Hold Removal date and Last Hold Removal date based on the Application Date.
2. Payment Hold Activity – update the Number of Days Held
3. Disbursement Request – clear the Current Hold Level and Current Hold Type

If a matching record is found with a Hold action of *Request Hold*, then the job will:

1. Populate the Current Hold Level and Current Hold type to the appropriate Hold level fields on the Disbursement Request from Payment Hold Maintenance.

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All processing was successful	N/A	N/A
Warning (4)	This process step does not end with this Return Code.	N/A	N/A
Non-Fatal Error (8)	This process step does not end with this Return Code.	N/A	N/A
Failed (12)	Failed because of runtime exceptions for an unexpected situation.	Failure reason needs to be investigated before scheduling a new job.	Failed because of runtime exceptions for an unexpected situation.
Terminated (16)	Job is terminated manually by the user.	Restart ability is available so you can re-start the job and it selects the records from the last processed record for cleanup	N/A
System Failure (20)	When the job is terminated because of database server or network issues.	Restart ability is available so you can re-start the job and it selects the records from the last processed record for cleanup	N/A

Step 3: Close or Applying

Disbursement Request records will be considered all over again in this step to ensure the proper values are in all hold fields. Record selection is as follows:

- Allow Bypass Requests is *false*
- Payment is not to a P-Card vendor
- Line Amount is greater than $0.00
- Vendor is Non-Miscellaneous
- User Hold is *false*
- Disbursement Management Hold is *false.*

Each selected record is matched to a3

.3.3.3 3…. Payment Hold Maintenance record based on the hold level and vendor and/or Award/Payment Request transaction information.

- Search PHLDM for matching “Request Hold” records that are open and “Remove Hold” records for the same hold level and vendor and/or Award/Payment Request transaction information.

e. If a “Remove Hold” exists,

- Close the “Request Hold” and “Remove Hold” Record to “close it”.
- Update Payment Hold Activity Table (PHLDA) to show that the hold was removed and the number of days it was held.

f. If only “Request Hold” record exists and no “Remove Hold” exists, compare the “Request Hold” record to the DISRQ Hold fields of the same Hold Type.

- Only if the selected PHLDM request record has a higher priority than the DISRQ Hold, then replace the DISRQ Hold for this Hold Type with the current Request Hold record.
- If a matching DISRQ and PHLDM record with the highest priority is found and the current hold type and level are null then insert a record in the Payment Hold Activity Table (PHLDA).

g. For each Hold Type, if no PHLDM “Request Hold” or “Remove Hold” records are found, set the Hold fields to null on DISRQ for that Hold Type.

h. For hold type where only open “Remove Hold” records are present, set the Hold fields to null on DISRQ for that Hold Type, Close the PHLDM “Remove Hold” record. Update PHLDA record for Hold Removal date and Number of Days Held.

Possible Return Codes	**Condition**	**Recommendation**	**Other Instructions**
Successful (1)	The record has been successfully updated.	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Failed because of runtime exceptions for an	Failure reason needs to be investigated before	N/A

	unexpected situation.	scheduling a new job.	
Terminated (16)	Job is terminated manually by the user.	Restart ability is available so you can re-start the job and it selects the records from the last processed record.	N/A.
System Failure (20)	When the job is terminated because of database server or network issues.	Restart ability is available so you can re-start the job and it selects the records from the last processed record.	N/A.

Step 4: Update the Current Hold Level

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	The record has been successfully inserted.	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A.
Failed (12)	Failed because of runtime exceptions for an unexpected situation.	Failure reason needs to be investigated before scheduling a new job.	N/A
Terminated (16)	Job is terminated manually by the user.	Restart ability is available so you can re-start the job and it selects the records from the last processed record.	N/A.
System Failure (20)	When the job is terminated because of database server or network issues.	Restart ability is available so you can re-start the job and it selects the records from the last processed record.	N/A.

2.1.4 Automatic EFT Reversal Confirmation

Job Name	Automatic EFT Reversal Confirmation
Recommended Frequency	On demand. Alternatively, the job can be run before the EF ACH Reversal Chain, or the job can be configured to be part of the EF ACH Reversal Chain between the Build Flat File job and the Initiate Disbursement Cancellation job.
Single Instance Required	Yes
Can be restarted?	No
Reports generated	Yes

Overview

The Automatic EFT Reversal Confirmation process automatically confirms the Accounts Payable EFT reversals sent to the bank that are eligible to be confirmed automatically based on the lag days.

The steps involved in this process are:

1. **Parameter Validation**: First the process validates the batch parameters.
2. **Selection of Records:** Once the parameter validation is successful, the job calculates the cutoff date and selects records from the EFT Reversal (EFTREV) table. The job calculates the cutoff date by subtracting the number of business days specified by the Cancel EFT Reversal Lag Days (CAN_EFT_REV_LAG_DAYS) parameter on the Application Parameters (APPCTRL) table from the Application Date. The EFTREV records whose Reversal Status is equal to *Reversal sent to the Bank* and Date ACH Reversal File Created is before the calculated cutoff date are selected.
3. Process records:
   - If the EFTREV record Cancellation Type is equal to *Reschedule* and the Reschedule Payment Date is a past date, or the EFTREV record Cancellation Type is equal to *Hold* and the Hold Type, Payment Hold Type Department, and Payment Hold Type Unit on the EFTREV record is not a valid combination on the PHLDTD table or does not have a Hold Level of Automated Payment Request or User Hold, the record is written to the Automatic EFT Reversal Confirmation Exception Report without further processing.
   - Update the EFTREV record Reversal Status to *Reversal Confirmed*. Additional fields are populated using the job parameter values as needed if the Cancellation Type, Cancellation Reason, Reschedule Payment Date, or hold fields (Hold Type, Payment Hold Type Department, Payment Hold Type Unit, Hold Request Description) are not populated on the EFTREV record. The updated records are also written to the Automatic EFT Reversal Confirmation Report.

The following table shows the various steps that the Automatic EFT Reversal Confirmation job goes through and the messages issued at each step:

Process Steps	Messages
1. Parameter Validation	• Validating Batch Parameters

Process Steps	Messages
	• Parameters are valid or invalid depending on the validation. If the parameter is invalid, the invalid value is displayed in the log. • Parameter validation completed

2. Selection of Records	• After calculating the cutoff date, the following message is issued:
	  • EFT Reversal records with Reversal sent to the Bank status and Date ACH Reversal File Created before <<date>> will be processed.
	• If the selection returns 0 records, then the following message is issued:
	  • No records meet selection criteria.

3. Process Records	The following Progression messages are issued: • Number of records processed: 'n' • Number of records confirmed: 'n'

Restartability

The job cannot be restarted. If the job fails in any of the above steps, a new job should be scheduled after correcting the errors that caused the job to fail.

Major Input

- EFT Reversal (EFTREV)
- Application Parameters (APPCTRL)

Job Parameters

Name	Description	Value and Edits
CLIENT_NM	Optional. Client Name	No default
CAN_TYP	Required. Cancellation Type: 1 (Reschedule), 2 (Hold), 3 (Close), 8 (PR Cancellation) Must be an integer matching one of the valid values shown above.	No default
CAN_REAS_CD	Required. Cancellation Reason: 1 (Lost), 2 (Stolen), 3 (Never Received), 4 (Destroyed), 5	No default

Name	Description	Value and Edits
	(Incorrect Information) Must be an integer matching one of the valid values shown above.	
RESCHED_PYMT_DT	Conditionally required. Reschedule Payment Date (MM/DD/YYYY) Reschedule Payment Date is required if the Cancellation Type parameter is equal to *1 (Reschedule)*. Must be a date on or after the application date.	No default
HLD_TYP	Conditionally required. Hold Type Required if the Cancellation Type parameter is equal to *2 (Hold)*. The specified Hold Type, Payment Hold Type Department, and Payment Hold Type Unit must be a valid combination on the Payment Hold Type by Department (PHLDTD) table using wildcard logic and must have a Hold Level of Automated Payment Request or User Hold.	No default
PY_HLD_TYP_DEPT	Conditionally required. Payment Hold Type Department Required if the Cancellation Type parameter is equal to *2 (Hold)*.	No default
PY_HLD_TYP_UNIT	Conditionally required. Payment Hold Type Unit Required if the Cancellation Type parameter is equal to *2 (Hold)*.	No default
HLD_RQST_DSCR	Conditionally required. Hold Request Description Required if the Cancellation Type parameter is equal to *2 (Hold)*.	No default

Major Output

- EFT Reversal table
- Automatic EFT Reversal Confirmation Report
- Automatic EFT Reversal Confirmation Exception Report

Job Return Code

The following table shows the potential job Return Codes for the Automatic EFT Reversal Confirmation job:

Return Code	Condition
Successful (1)	All of the selected records are processed successfully
Warning (4)	No eligible records found. This could be because of the following reason: • No records meet selection criteria.
Non-Fatal Error (8)	Out of the selected records at least one record is written to the Exception report. This could be because of one of the following reasons: • Reschedule Payment Date must be on or after the Application Date • Hold Type-Dept-Unit must have a Hold Level of *Automated Payment Request* or *User Hold* on Payment Hold Type By Department (PHLDTD).
Failed (12)	The job may fail under the following conditions: • Parameters are invalid • Run time exceptions for unexpected situations.
Terminated (16)	This return code is issued when the job is terminated by the user.
System Failure (20)	This return code is issued when the job is terminated because of database server or network issues.

Sort Criteria

The selected EFTREV records are sorted by Bank Account Code (BANK_ACCT_CD), Reclamation flag, EFT Number, EFT Transaction Code, EFT Transaction Dept, EFT Transaction ID, and EFT Transaction Version Number.

Selection Criteria

The process selects all EFT Reversal table records that meet the following selection criteria:

- Reversal Status is equal to *Reversal sent to the Bank*, and
- Date ACH Reversal File Created is before the calculated cutoff date

Problem Resolution

The process cannot be restarted on failure; however, a new job can be scheduled after resolving the issue.

The following table shows the possible Return Codes and recommendations for each processing step:

Step 1: Parameter Validation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful **Sample Message:** "Parameter validation is completed"	N/A	N/A
Warning (4)	N/A	This step does not issue this return code.	This step does not issue this return code.
Non-Fatal Error (8)	N/A	This step does not issue this return code.	This step does not issue this return code.
Failed (12)	The required parameters are not entered. **Sample Message:** "Cancellation Type is required"	Enter valid information for all required parameters.	Reschedule the job with appropriate parameter values.
	The job failed because of runtime exceptions for an unexpected situation.	The reason for the failure needs to be investigated before re-scheduling the job.	Reschedule the job with appropriate parameter values.
Terminated (16)	The job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can be rescheduled.	Schedule a new job.
System Failure (20)	The job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can be rescheduled.	Schedule a new job.

Step 2: Selection of Records

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	If the selection returns 0 records, then the following message will be issued: **Sample Message:** "No records meet selection criteria."	Verify EFTREV records against selection criteria and setup records accordingly before re-scheduling the job.	Reschedule the job with appropriate records on EFTREV.
Non-Fatal Error (8)	N/A	This step does not issue this return code.	This step does not issue this return code.
Failed(12)	The job failed because of runtime exceptions for an unexpected situation.	The reason for the failure needs to be investigated before re-scheduling the job.	Reschedule the job with appropriate parameter values.
Terminated (16)	The job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can be rescheduled.	Schedule a new job.
System Failure (20)	The job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can be rescheduled.	Schedule a new job.

Step 3: Processing Of Records

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All the records were processed and successfully.	N/A	N/A
Warning (4)	N/A	This step does not issue this return code.	This step does not issue this return code.
Non-Fatal Error (8)	At least one record was written to Exception report.	The reason for the failure needs to be investigated before re-	Reschedule the job with appropriate

Possible Return Codes	Condition	Recommendation	Other Instructions
	Sample Message: "Resched Pymt Dt must be on or after the Application Dt "	scheduling the job.	parameter values.
Failed (12)	The job failed because of runtime exceptions for an unexpected situation.	The reason for the failure needs to be investigated before re-scheduling the job.	Reschedule job with appropriate parameter values.
Terminated (16)	The job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can be rescheduled.	Schedule a new job.
System Failure (20)	The job is terminated because of database server or network issues	The reason for the System Failure needs to be investigated. The job can either be restarted or a new job can be scheduled.	Schedule a new job.

2.1.5 Backup Withholding History

Chain or Job Name	Backup Withholding History
Recommended Frequency	Twice a year
Single Instance Required	Single
Can be restarted?	The job can be restarted
Reports generated	None

Overview

The Backup Withholding History batch process updates the Backup Withholding History Detail table with transaction information associated with backup withholding. The process selects records from the Accounting Journal (JACTG) table that are associated with one the following groups:

- Withheld amount associated with issued vendor payment, including cancelled vendor payments using the “Discard/Cancellation” functionality;
- Withheld amount associated with stale dated, escheated or cancelled vendor payment using the “Disbursement Reclassification (DC) transaction”;
- Remitted amount to IRS with withheld money;
- Refunded amount from IRS due to error in withholding;

This process will not select the Journal records that have already been selected during previous runs. Records that have a match on the data table (Backup Withholding History Daily Balance) will have their corresponding amounts updated; records without a match on the data table will be inserted.

This process will select records from the Accounting Journal and update the BWNPH table with accounting transactions information. The process will involve the following steps.

- Parameter Validation
- Selection of records from JRNL_ACTG table.
- Compare the records with SPEC and SPECFUND tables
- Insert/ Update the records in BWNPH table.

Process Steps	Messages
1. Parameter Validation	• Validating Batch Parameters • Parameters are valid or invalid depending on the Validation. If the parameter is invalid, the invalid value will be displayed in the log.

Process Steps	Messages
	• Batch Parameter validation completed
2. Process records	• Select Records from JACTG, SPEC, SPECFUND tables • Insert/ Update the records in BWNPH table

The Job will run as a Single Instance with a frequency of twice a year.

Major Input

The inputs for this batch are the following:

- JACTG table
- SPEC table
- SPECFUND table

Major Output

- Records will be inserted or updated in BWNPH table

The job will then insert records selected in steps 1 and 2 (in the Selection Criteria below) into the Backup Withholding History Daily Balance table. Amount(s) for each inserted record will be summarized by the Record Date, Fund, Sub Fund, Balance Sheet Account (BSA), Sub BSA and Fiscal Year.

For each record that satisfied the logic in steps 1a) of the Selection Criteria below, where the BSA specified in the Backup Withholding Payable field on SPEC (or SPECFUND) matches the BSA on the JACTG record, the job will populate 'Withheld/Refunded Amt' field with a 'Posting Amount' of the "Credit" record on the JACTG. The amount displayed will correspond to the actual amount of JACTG record, multiplied by (-1).

For each record that satisfied the logic in steps 1b) of the Selection Criteria below, where the BSA specified in the Backup Withholding Payable, Stale Dates, Escheat Dates or Cancelled Disbursements Payable fields on SPEC (or SPECFUND) matches the BSA on the JACTG record, the job will populate the 'Cancelled/Remitted Amt' field with a 'Posting Amount' of the "Debit" record on the JACTG.

Sort Criteria

N/A

Selection Criteria

1. The batch job will read the JACTG table and select records that meet one of the conditions:
   a) 'Posting Pair' = "E", AND

'BSA Type' = "Liability", AND
'PER_DC' <> "0", AND
'PER_DC' <> "99";

OR

'Transaction Type' = "CR", AND
'BSA Type' = "Liability", AND
'Line Function' <> "Liquidation", AND
'PER_DC' <> "0", AND
'PER_DC' <> "99";

b) 'Transaction Type' = "MD" or "AD" or "DC" or "JV" or "CR", AND
'BSA Type' = "Liability", AND
'Posting Line Function' <> "Liquidation", AND
'PER_DC' <> "0", AND
'PER_DC' <> "99";

c) 'Transaction Type' = "DC", AND
'BSA Type' = "Liability"; AND
'Line Function' (LN_FUNC_CD) <> "Liquidation", AND
'PER_DC' <> "0", AND 'PER_DC' <> "99";

If no eligible records could be selected on JACTG table, an error message will be written to the log and the job will end with a status code of *Warning*.

Note that the job excludes all Accounting Journal records that are posted in 'Accounting Period' of "0" and "99". This logic ensures that the Annual Close activities will not have any impact on the amounts inserted into BWRDI table.

2. The job will use the Fiscal Year (FY) of each record identified in step 1 above to read the SPEC table and retrieve the Balance Sheet Account codes as defined in the following fields:

    a. Backup Withholding Payable

    b. Stale Dated Payable

    c. Escheat Payable

    d. Cancelled Disbursements Payable

    If for any of the aforementioned four BSA fields their corresponding 'Override' flag is checked on SPEC table, then the system will retrieve the appropriate BSA code from the SPECFUND table.

    The job will filter out records selected in Step 1 (a & b) to fetch only those records with BSA values that match those on SPEC (or SPECFUND, if applicable) for the record's Fiscal Year. If a record could not be found on SPEC table for the corresponding Fiscal Year, an error message will be written to the log and the job will end with a status code of *Failed.*

Problem Resolution

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All the parameters are validated successfully	N/A	N/A

Possible Return Codes	Condition	Recommendation	Other Instructions
Warning (4)	No records were found eligible	Check the parameter values.	N/A
Non-Fatal Error (8)	This return code will be issued when the validation fails in this step	N/A	N/A
Failed (12)	Job failed due to Fatal conditions	If the job fails because of the runtime exceptions, investigate the exception reported by the process, resolve the error and restart the job.	Schedule a new job
	Failed while restarting the job since another instance of the job has already been run successfully. Cannot restart the job since another instance of this job has already been run successfully.	**Recommendation:** Schedule a new job.	
Terminated (16)	Job is terminated manually by the user.	Reason for the termination needs to be investigated. The job can either be restarted or schedule a new job.	Schedule a new job
System Failure (20)	When the job is terminated because of database server or network issues	Reason for the System Failure needs to be investigated. The job can either be restarted or schedule a new job.	Schedule a new job

Possible Return Code

Return Code	Condition
Successful (1)	All the selected records are processed successfully
Warning (4)	No eligible records found. This could be because of the following reason: • No matching records found between JACTG, SPEC and SPECFUND
Non-Fatal Error (8)	
Failed (12)	The job will fail under the following conditions: • Parameters are invalid • Unable to Insert Records in BWNPH table • Run time exceptions for unexpected situations. When this job ends with a return of code *Failed*, subsequent jobs in the chain will be set to *Inactive*.
Terminated (16)	This return code will be issued when the job is terminated by the user. When this job ends with a return of code *Terminated* subsequent jobs in the chain will be set to *Inactive*.
System Failure (20)	This return code will be issued when the job is terminated because of database server or network issues. When this job ends with a return of code *System Failure*, subsequent jobs in the chain will be set to *Inactive*.

Recommendation: If the job failed due to some invalid parameters, the recommendation should be to correct parameter and restart the job or Schedule a new job.

The job may take longer to process journal records if there are many gaps present in the accounting journal. The more gaps, the longer the process takes to complete. It is recommended to increase the select and commit block size (to around 20,000) to make the job run faster.

2.1.6 Check Reconciliation

The Check Reconciliation process maintains consistency between the bank's records and a client's disbursement records within CGI Advantage Financial. The process updates a check's status (for example, from *Outstanding* to *Paid* or *Cancelled*), and checks for discrepancies between CGI Advantage and the bank.

The Check Reconciliation process is comprised of five batch jobs that are chained together. Each step in the process is described below.

Description

Flat to XML Process

This process converts the flat file sent by the bank into XML format to be uploaded into the Check Exception table in the next step.

CHREC Load Process

This process reads the XML file (bank data) generated from the Flat File to XML process and uploads it to the Check Exception table. This process also sets the Recent Record flag to false for existing records on Check Exception table. (This flag is used by the Control Totals report discussed below).

Automated Check Reconciliation Process

This process reconciles the bank's records loaded to the Check Exception table with the system records on the Check Reconciliation table. Records that can be reconciled (check number and outstanding amount match) with a Status of *Disbursed* are marked as *Paid*, the Recently Reconciled flag set to True, and the record is moved to the Paid Checks table. If the previous Status is *Warranted*, then it is not updated to *Paid*, but the Recently Reconciled Flag is set to True. In addition, this process moves all payments to the Paid Checks table where the Cleared Date is not null and the Status is not *Disbursed* / *Warranted*. Records are moved to the Paid Checks table with their current Status (For example, *Paid* or *Cancelled*). Records from the bank file that do not have a match in CGI Advantage are marked as an Exception.

Check Exception Report

The Check Exception report is generated at the end of the Automated Check Reconciliation process, and lists all the records in the bank file that have no matching bank account number, or where a match was made on Check Number, but the check amount on the Check Reconciliation table did not match.

Control Totals Report

The Control Totals report is generated at the end of the Chain job process, and lists a summary of records reconciled during that run of the Automated Check Reconciliation process. The report is generated by selecting the records on the Check Exception table where the Recent Record flag is True.

When to Run

Monthly, weekly, daily, or on-demand.

Reconciled warrants will not be included in the Control Totals Report until the transactions are moved to the Paid Checks table. If you want to include the cleared warrants during the same run when they are reconciled with the bank file, then this job must be run after the Generate Warrant Reconciliation Transaction job, which creates Automated Warrant Reconciliation (AWR) transactions for cleared warrants.

Major Input

- Check Reconciliation table (R_AP_CHK_RECON)
- Check Exception table (R_AP_CHK_EXCP)
- Reconciliation file from bank with CLDT_DT field in format MM/DD/YY 00:00:00

Output

The Check Reconciliation process does the following:

- Inserts records into AP_PD_CHK table.
- Deletes records from R_AP_CHK_RECON table.
- Deletes records from R_AP_CHK_EXCP table.
- Generates the Exception Report.
- Generates the Control Totals Report

Parameters

Job	Parameter	Description	Default Value
Flat To XML	Attribute List (ATTRIBUTE_LIST)	Required Field. The process requires the order of the fields in the bank's data file as a parameter. If site is not expecting BAI Code and Bank Description to be part of the file received from the Bank, then they should use Attribute List. If site is expecting BAI Code and Bank Description to be part of the file received from the Bank, then they should use Attribute List. Before running, please ensure that the sequence of the attribute listed in this parameter matches the	**Attribute List 1:** BANK_NO,BANK_ACCT_NO,EXCP_CD,BANK_CHK_EFT_NO,BANK_CHK_EFT_AM,CLR_DT,CHK_EFT_NO,CHK_EFT_AM,CHK_EFT_ISS_DT,CHK_STA,CAN_REAS_CD,DEL_FL,DOC_CD,DOC_DEPT_CD,DOC_ID,AMS_ROW_VERS_NO **Attribute List 2:** BANK_NO,BANK_ACCT_NO,EXCP_CD,BANK_CHK_EFT_NO,BANK_CHK_EFT_AM,CLR_DT,CHK_EFT_NO,CHK_EFT_AM,CHK_EFT_ISS_DT,CHK_STA,CA

Job	Parameter	Description	Default Value
		sequence in the flat file provided by the bank for proper conversion. Please leave AMS_ROW_VERS_NO as the last item, although not in the bank file.	N_REAS_CD,DEL_ FL,DOC_CD,DOC_ DEPT_CD,DOC_ID, BAI_CD, BANK_DSCR,AMS_ ROW_VERS_ NO,
	Data Object Name (DATAOBJECT_N AME)	Required Field. This is used to create a Data Object Node in XML.	R_AP_CHK_EXCP
	Delimiter (DELIMITER)	Required Field. Defines what separates the fields in the bank's data file.	,
	Flat File Name (FLAT_FILE)	Required Field. The name of the text file that contains the banks data.	R_AP_CHK_EXCP_ 1.txt
	XML File Name (XML_FILE)	Required Field. Defines the name of the XML file that will be created by this process.	R_AP_CHK_EXCP_ 1.xml
	Export Location at Flat To XML Job (AMSEXPORT)	Required (** **Refer to Note**: Assumptions for SWBP on page no. 7)	No Default
	Parameter Location at Flat To XML Job (AMSPARM)	Required (** **Refer to Note**: Assumptions for SWBP on page no. 7)	No Default
	Import Location at Flat To XML Job (AMSIMPORT)	Required (** **Refer to Note**: Assumptions for SWBP on page no. 7)	No Default
CHREC Load	Client name for Report (CLIENT_NM)	Optional field. Entry of a value in this field specifies the name that will appear on report.	No Default

Job	Parameter	Description	Default Value
	Commit Block Size (COMMIT_BLOCK)	Required field. Controls how many records are committed by the application at one time. The size should be compatible with technical capabilities and performance guidelines	200
	XML File Name (XML_FILE)	Required Field. Defines the XML file name to be uploaded into the Check Exception table. The name corresponds to the file created in the previous job.	R_AP_CHK_EXCP_1.xml
	Import Location at Flat To XML Job AMSIMPORT	Required (** **Refer to Note**: Assumptions for SWBP on page no. 7)	No Default
Auto Check Recon	Process (Process)	Required (and protected) field that is used to control the Check Reconciliation process.	BATCH
	Exclude Stale Dated or Escheat records (EXCLD_STALE_DTD_ESCHEAT)	A required parameter that controls whether or not CHREC records that are stale should be moved to Paid Checks (PDCHK). If SOPT is setup to only stale date, the parameter should be *N* to ensure that the records are moved. If SOPT is setup to stale and escheat, the parameter should be set to *Y* so that stale records are not moved.	N
Check Exception	Client name for Report (CLIENT_NM)	Optional field. Entry of a value in this field specifies the name that will appear on report.	No Default
	Process (Process)	Required (and protected) field that is used to control the creation of the Check Exception Report.	REPORT

Job	Parameter	Description	Default Value
Control Totals Report	CLIENT_NM	Client Name	No Default
	RPT_ID	Report ID	No Default
	RPT_NM	Report Name	Check Exception Control Totals Report

Sort Sequence

None

Selection Criteria

- Select all the records from the Check Reconciliation table where the Cleared Date is not null and the Status is not *Disbursed* / *Warranted*. Move those into the Paid Checks table.
- Select all records from the Check Reconciliation table where the Cleared Date is not null, the Cleared Date is less than or equal to the Application Date and the Status is *Warranted*. Update the selected records and set the Recently Reconciled flag to *True* but keep the Status as *Warranted*.
- Select records from the Check Exception table where the delete flag is not selected. Reconcile those with the corresponding records from the Check Reconciliation table.
- Select records from the Check Exception Table where the delete flag is selected. Delete those from the table.

Problem Resolution

Flat to XML process

If the process fails for any reason, check to see whether an XML file of the name of the bank's flat file or the name you specified already exists in the correct data directory with the current system date and time. If the process did not create an XML file, or the created file is not in the proper format, then correct the problem and rerun the process.

2.1.7 Claim Schedule Numbering

Job Name	Claim Schedule Numbering
Recommended Frequency	After running AD Chain
Single Instance Required	Yes
Can be restarted?	No
Reports generated	None

Overview

The Claim Schedule Numbering job will assign Claim Schedule Numbers to all eligible Disbursement Transactions (AD, MD, and EFT) using Disbursement Format (DISF) detail and Claim Schedule Numbering (CSN) as input. Claim schedule numbering is used when the System Option of External Disbursement Option is set to Disbursements and Claims to provide the external disbursement system with the necessary information.

This job will also update the Claim Schedule Numbering Processed indication to *true* and the Claim Schedule Print Indicator to Ready to Print. These updates prevent the disbursement transaction from being selected again on a future numbering job and prepare the Claim Schedule Print Indicator for the Claim Schedule Printing job.

The job performs the following steps:

Process Steps	Messages
1. Parameter Validation	• Validating Batch Parameters • Parameters are valid or invalid, depending on the validation. If the parameter is invalid, the invalid value is displayed in the log. • Batch Parameter validation completed.
2. Selection and Update of Records	• Selecting eligible records from AD/MD/EFT transactions based on the DISF and CSN tables. • Updating disbursement transaction header section. • Updating Accounting and Posting lines on disbursement transactions is complete. • Processing of selected record is completed.

Major Input

- Claim Schedule Numbering (R_CLM_SCHED_NO)
- Disbursement Format (R_AP_DISB_FRMT)
- Automatic Disbursements (AD_DOC_HDR)
- Manual Disbursements (MD_DOC_HDR)

Batch Parameters

The following are the delivered parameter values, which may have been updated through Batch Setup to meet local needs.

Parameter	Description	Default Value
Disbursement Format (DISB_FRMT)	A required selection parameter of one or more disbursement formals. A single Disbursement Format or list of eligible multiple Disbursement Formats are allowed with commas separated. ALL is a valid value, where all disbursement formats are considered.	ALL
Total Dollar Amount Threshold (THLD_AM)	A required parameter to specify the maximum dollar amount allowed for all disbursements assigned to a given Claim Schedule Number. Once the Threshold Amount is reached, the job will break to a new Claim Schedule Number.	999999999.99
Commit Size (COMMIT_SIZE)	An optional performance parameter to control the number of records committed to output in a single instance. If left blank, 10 will default.	10

Major Output

- Claim Schedule Numbering (R_CLM_SCHED_NO)
- Automatic Disbursements (AD_DOC_HDR)
- Manual Disbursements (MD_DOC_HDR)

Job Return Codes

If this job does not finish successfully, there is no restarting. (Verify that statement)

Return Code	Condition
Successful (1)	The job ends as successful when all parameters are valid and at least one disbursement transaction was selected and updated.
Warning (4)	No records were found to match selection criteria.
Non-Fatal Error (8)	This job does not use this return code.

Return Code	Condition
Failed (12)	The job fails under the following conditions: • Parameters are invalid. • Run time exceptions for unexpected situations.
Terminated (16)	The job is terminated by the user.
System Failure (20)	A system failure is issued when the job is terminated because of database server or network issues.

Sort Criteria

None

Selection Criteria

The Claim Schedule Numbering job selects the eligible Disbursement Transactions based on the following selection criteria:

- Transaction Phase is *Final*.
- Transaction Function is *New* or *Modified*.
- Claim Schedule Number is *blank* and Claim Schedule Numbering Processed is *No* on the disbursement header.
- Disbursement Format has the Suppress Claim Schedule Numbering of *false*.
- Face Sheet Amount on the disbursement header is greater than $0. (If $0, the job assigns a Claim Schedule Number of all zeros to those Disbursement transactions).

Problem Resolution

If the job ends with a return code of Failed and above, the job does not have a restart logic that allows the job to pick up numbering where it left off. The disbursement transactions that were numbered will be rolled back to remove the assigned Claim Schedule Numbers. The job issues an error describing the error and the problem transaction and/or parameter. The error needs to be resolved before another job is schedule. The unused numbers that were removed when numbering failed will not be used again, and the next number will be assigned.

2.1.8 Claim Schedule Printing

Job Name	Claim Schedule Printing
Recommended Frequency	After Claim Schedule Numbering job.
Single Instance Required	Yes
Can be restarted?	No
Reports generated	None

Overview

The Claim Schedule Printing job prints the eligible disbursement transactions with Face Sheets and Remittance Advices. This job generates an XML file and PDF to print warrants, EFT's, and remittance advices. The Claim Schedule printing job can be run in two modes, that is normal print mode and re-print mode. The claim schedule print job can only be run when the External Disbursement Option on System Options is set to *Disbursements & Claims*.

The Claims Schedule Printing job will have the option to be run in two selection modes:

- Normal Mode (Default): When run in Normal Mode, the process selects disbursement transactions where the Claim Schedule Print Indicator on the Header of the disbursement transaction is equal to *Ready to Print* and the transaction's Disbursement Format on the Vendor Line is equal to one of the Disbursement Format parameter values specified.
- Reprint Mode: This mode is used to print claim schedules that have been printed previously, and in addition to those that have never been printed before. In this mode, the process selects disbursement transactions where the Claim Schedule Print Indicator on the Header is equal to *Ready to Print* or *Printed*, the transaction's Disbursement Format on the Vendor Line is equal to one of the Disbursement Format parameter values specified, and the Claim Schedule Number on the transaction is equal to or between the Starting Claim Number and Ending Claim Number ranges entered in the job parameters.

The job performs the following steps:

Process Steps	Messages
1. Parameter Validation	• Validating Batch Parameters • Parameters are valid or invalid, depending on the validation. If the parameter is invalid, the invalid value is displayed in the log. • Batch Parameter validation completed.
2. Selection and Update of Records	The following set of messages appears when the Forms to Generate parameter is Face Sheet. • Selecting records to generate Face Sheets. • If the selection returns zero records, then this step will be completed, and the following message will be issued: • "No record to generate".

Process Steps	Messages
	• Number of records selected to generate Face Sheet. • End Face Sheet Generation. The following set of messages appears when the Forms to Generate parameter is Remittance Advice. • Selecting records to generate Remittance Advice. • If the selection returns zero records, then this step will be completed, and the following message will be issued: • "No record to generate". • Number of records selected to generate Remittance Advice. • End Remittance Advice Generation.

Major Input

- Automatic Disbursements (AD_DOC_HDR)
- Manual Disbursements (MD_DOC_HDR)

Batch Parameters

The following are the delivered parameter values, which may have been updated through Batch Setup to meet local needs.

Parameter	Description	Default Value
Export Location (AMSEXPORT)	The required parameter to specify the location of the output files.	No Default
Log Location (AMSLOGS)	The required file location for writing out and reading output log files.	No Default
Parameter File Location (AMSPARM)	The required file location to specify the location of the output parameter files.	No Default
Disbursement Format (DISB_FRMT)	The required parameter when run in Normal Mode and prohibited in Reprint Mode. A single Disbursement Format or list of eligible multiple Disbursement Formats are allowed with a comma separated list. ALL is a valid value, where all disbursement formats are considered.	ALL
Claim Schedule Printing End Number (END_NUM)	A required parameter when run in re-print mode and is prohibited when run in Normal mode. The Claim Schedule End Number is used when selecting a specific range of claim schedule numbers to print during Reprint mode.	No Default
Claim Schedule Printing Start Number	A required parameter when run in re-print mode and is prohibited when run in normal mode. The Claim Schedule Start Number is used when	No Default

Parameter	Description	Default Value
(START_NUM)	selecting a specific range of claim schedule numbers to print during Reprint mode	
Forms to Generate (FORMS)	A required parameter to select record to generate forms based on Face Sheer or RA or both. Valid values: 1 - Face Sheet, 2 - RA, and 3 – Both.	3
Face Sheet Application Resource (FS_APP_RSRC)	An optional parameter, the Application Resource specified on Disbursement Format (DISF) for the Face Sheets being printed.	No Default
Face Sheet Print Job Code (FS_PRNT_JBCD)	An optional parameter, the Print Code specified on Disbursement Format (DISF) for the Face Sheets being printed.	No Default
Face Sheet Print Resource (FS_PRNT_RSRC)	An optional parameter, the Print Resource specified on Disbursement Format (DISF) for the Face Sheets being printed.	No Default
Remittance Advice Application Resource (RM_APP_RSRC)	A conditionally required field when the “Forms to Generate” parameter is equal to 2 (Remittance Advice) or 3 (Both). This parameter identifies the Application Resource to be used when printing Remittance Advice.	DISB_PRN
Remittance Advice Print Job Code (RM_PRNT_JBCD)	A conditionally required field when Forms To Generate parameter is equal to 2 (Remittance Advice) or 3 (Both). This parameter identifies the Print job code to be used when printing Remittance Advice.	RA_STD404C
Remittance Advice Print Resource (RM_PRNT_RSRC)	A conditionally required field when Forms To Generate parameter is equal to 2 (Remittance Advice) or 3 (Both). This parameter identifies the Print Resource to be used when printing Remittance Advice.	FMP2-256RA-4-HPP3005
Run Mode (RUN_MODE)	A required parameter to select disbursement transaction to print or reprint that satisfy the selection parameters and are marked as Ready to Print or Printed. Valid values: 1 - Normal 2 - Re-print.	1
View Forms (VIEW_FORMS)	A required output parameter that when set to Y sends output to PDF to print separately. Note a PDF print resource must be used. N sends forms output directly to print resource destination.	Y

Major Output

- Disbursement Transaction field 'Claim Schedule Print Indicator' updated from Ready to Print to Printed.
- The following XML files are created under Export Import folder:
  - DirectPayWarrant_Temp.xml
  - ORFCheck_Temp.xml
  - RevertApprop_Temp.xml
  - MRTA_Temp.xml
  - GL3220_Temp.xml
  - EFT_Temp.xml
  - ECLAIMS_Temp.xml
  - EscrowPayout_Temp.xml
  - RemAdvice_Temp.xml

Job Return Codes

If this job does not finish successfully, there is no restarting. (Verify that statement)

Return Code	Condition
Successful (1)	The job ends as successful when all parameters are valid and at least one disbursement transaction was selected and updated.
Warning (4)	No records were found to match selection criteria.
Non-Fatal Error (8)	This job does not use this return code.
Failed (12)	The job fails under the following conditions: • Parameters are invalid. • Run time exceptions for unexpected situations.
Terminated (16)	The job is terminated by the user.
System Failure (20)	A system failure is issued when the job is terminated because of database server or network issues.

Sort Criteria

None

Selection Criteria

The Claim Schedule Printing job selects eligible Disbursement Transactions based on the following section criteria:

- Transaction Phase is *Final*
- Transaction Function is *New* or *Modified.*

- Claim Schedule Number Processed is *true.*
- Claim Schedule Print Indicator is *Ready to Print* for normal mode and *Printed* for re-print mode.

Problem Resolution

If the job ends with a return code of Failed and above, the job will not have a restart logic that allows the job to pick up printing where it left off. The job will issue an error describing the error and the problem transaction and/or parameter. The error needs to be addressed before scheduling a new job. The new instance of the job can be executed without having to back out any transaction updates.

2.1.9 Consolidation By Payment Request Department

Chain or Job Name	Consolidation By Payment Request Department
Recommended Frequency	Nightly cycle or on demand
Single Instance Required	Yes
Can be restarted?	No
Reports generated	No

Overview

This batch job sets the Disbursement Request (DISRQ) records Consolidated Object 6 field with the Transaction Department Code from the Payment Request mapped to the respective DISRQ records. It is used as an automated mechanism within Advantage 3 disbursement processing to consolidate payment requests based on the Payment Request Transaction Department Code.

Process Steps

The high level flow of the Consolidation On Payment Request Department Batch Job is as follows:

Process Steps	Messages
1. Validate the Parameters	• Started Validating the Parameters • If parameters are invalid, then write the invalid param to job log
2. Build the SQL Update Statement	• Started Disbursement Request records Consolidated Object 6 field. • If there is an error committing the transaction, call to rollback and log an error that a problem was experienced updating DISRQ • Register the number of records updated inside of the Job Log

Restartability

Job cannot be restarted.

Major Input

Data from the following tables:

- R_AP_DISB_RQST – Disbursement Request Table

Batch Parameters

Parameter	Description	Default Value
DOC_CD	A single Transaction Code or Multiple Transaction Codes in a comma separated list	No Default

Major Output

- R_AP_DISB_RQST – Disbursement Request Table

Job Return Code

The following table shows the potential job return codes for the Inferences & Validations job:

Return Code	Condition
Successful (1)	The DISRQ records with the Transaction Department Code inside of the empty Consolidated Object 6 field are updated successfully.
Warning (4)	N/A
Non-Fatal Error (8)	N/A
Failed (12)	The job fails under the following conditions: • Parameters are invalid • Run time exceptions for unexpected situations. • If there were less than zero records updated
Terminated (16)	This return code will be issued when the job is terminated by the user. When this job ends with a return of code Terminated subsequent jobs in the chain will be set to inactive.
System Failure (20)	This return code will be issued when the job is terminated because of database server or network issues. When this job ends with a return of code System Failure.

Step 1: Parameter Validation

Validate the parameters, if all the parameters are validated successfully then make a call to update the Disbursement Request records

Step 2: Updating the records

Get a handle to the Update SQL to be executed as follows:

Create an update statement that sets CNSD_6_OBJ = DOC_DEPT_CD on all DISRQ records where CNSD_6_OBJ is NULL AND

If the member variable Transaction Code SQL is not empty Then

DOC_CD is in the set of transaction codes inside of the member variable Transaction Code SQL

End If the member variable Transaction Code SQL is not empty

Execute the Update SQL capturing the number of records updated as the return

If committing of this transaction fails Then

Call to rollback and log an error that a problem was experienced updating DISRQ

Return -1

End committing transaction failed

Return the number of records updated

Problem Resolution

The following table shows the possible return codes and recommendations for each processing step.

Step 1: Parameter Validation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All the parameters are validated successfully.	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Job failed due to Fatal conditions	In this step, the job can fail under the following conditions. 1) Encounters any runtime exceptions and 2) Failed during restart. 3) If the value supplied for Transaction Code is empty 4) If the Transaction Codes supplied are not validated successfully If the job fails because of the runtime	Schedule a new job

Possible Return Codes	Condition	Recommendation	Other Instructions
		exceptions, investigate the exception reported by the process, resolve the error and restart the job.	
Terminated (16)	Job is terminated manually by the user.	Reason for the termination needs to be investigated. A new job can be scheduled	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.
System Failure (20)	When the job is terminated because of database server or network issues	Reason for the System Failure needs to be investigated. A new job can be scheduled.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.

Step 2: Updating the records

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	N/A	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Failed because of runtime exceptions for an unexpected situation.	The job can fail under these conditions: • If committing of this transaction fails Failure reason needs to be investigated before scheduling a new job.	Failed because of runtime exceptions for an unexpected situation.
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated and	N/A

		since the job can be restarted	
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated and since the job can be restarted	N/A

2.1.10 Convert Bank File

Chain or Job Name	Convert Bank File
Recommended Frequency	On Demand
Single Instance Required	Yes
Can be restarted?	N/A
Reports generated	No

Overview

Some banks will send a flat file that is not compatible with the first job step in the Check Reconciliation process to convert to the necessary XML format. This conversion process will take input parameters that map data in the flat file to data that is necessary in the output file.

Major Input

Bank flat file

Batch Parameters

Parameter	Description	Default Value
Export Location for the Job (AMSEXPORT)	A required location to write output.	No Default
ABA Number (BANK_NO)	The required ABA Number that corresponds to the bank which sent that file.	No Default
The Name of the Incoming Bank file (INPUT_FILE)	The required name of the bank flat file.	No Default
Name of the File to be created (OUTPUT_FILE)	The required name of the output file.	No Default
Date Format (DATE_FRMT)	A required valid date format needed to convert the flat file data.	No Default
Attribute List	The required ordering of the fields in the bank's data file with start and end index, For example (ACCT_NO(13-24), BANK_NO(26-40)....).	No Default

Major Output

- Converted Flat File

Job Return Code

The following table shows the potential job return codes for the Convert Bank File job.

Return Code	Condition
Successful (1)	All the validations are performed successfully, and no errors are reported. The provided bank flat file converted successfully.
Warning (4)	No records found matching the selection criteria.
Non-Fatal Error (8)	Not applicable for this job.
Failed (12)	Parameter validation failed.
Terminated (16)	This Return Code is issued when the job is terminated by the user. When the job ends with a Return Code of Terminated, subsequent jobs in the chain are set to inactive.
System Failure (20)	This Return Code is issued when the job is terminated because of database server or network issues. When this job ends with a Return Code of System Failure, subsequent jobs in the chain are set to inactive.

Sort Sequence

N/A

Selection Criteria

None

2.1.11 CTX EDI Formatting

Chain or Job Name	CTX EDI Formatting
Recommended Frequency	Daily as part of the nightly cycle or on demand. This process runs after the baseline AD Chain batch process and before the baseline EF ACH Transaction batch process.
Single Instance Required	Single Instance
Can be restarted?	Yes
Reports generated	No

Overview

The CTX EDI Formatting batch process generates the CTX Addenda Records based on the EDI ANSI ASC X12 820 Remittance Advice/Payment Order. This batch process is only applicable to EFT transactions with the Disbursement Format of CTX.

Processing Steps

This process consists of following 5 basic steps:

- Parameter Validation:

  In this step, the process verifies the parameters. If the parameter validation is successful, the job goes to the next step. Otherwise the job ends with a return code of 'Failed'.

- Clearing the CTX EDI Segment table:

  In this step the job deletes all the records from the CTX EDI Segment table. These records were loaded into this table during the earlier run.

- Selection of Records:

  The selection of the eligible EFT records is dependent on the parameters. If the Run ID parameters are not entered then the job will select the all the EFT transactions from the AD Transaction Header where:

  - Transaction Phase Code = "Final" and
  - Transaction Function Code = "New" and
  - Disbursement Type = "EFT" and
  - Disbursement Format = "CTX" and
  - ACH File Generation Status = "Ready for Generation" and
  - No records found on the CTX EDI Addenda Table that matches the Bank Account and the Check EFT number of the selected EFT transaction and
  - Matching record exists on the AD Stub Details table

  If both of the parameters From Run ID and To Run ID are specified, then the process will select all EFT transactions on the AD Transaction Header where;

Run ID within the specified From Run ID / To Run ID range and

Transaction Phase Code = “Final” and

Transaction Function Code = “New” and

Disbursement Type = “EFT” and

Disbursement Format = “CTX” and

ACH File Generation Status = “Ready for Generation” and

No records are found on the CTX EDI Addenda Table that matches the Bank Account and the Check EFT number of the selected EFT transaction and

Matching record exists on the AD Stub Details table

The job will stop at this point if there are no eligible records found. The job return code will be set to ‘Warning’ in this case. If the job finds eligible records on the table, it will give a count of the selected records and go to the next step.

- Creation of EDI Segments and inserting records into the CTX EDI Segment table:

  For the selected EFT transactions, this process will create EDI segment records using the formatting rules.

  After creating the EDI segment record, the process will insert records into the CTX EDI Segment table. The records from this table will be used to generate the formatted CTX EDI Addenda Table.

  Progression messages will be issued after processing the number of records specified in the progression counter parameter.

- Inserting records into the CTX EDI Addenda table

  After creating the EDI Segment table records, the process will insert records into the CTX EDI Addenda table for every CTX EDI Segment table record. The records from this table will be used to generate the Addenda records on the EF ACH file.

  In the case of Miscellaneous Vendors, the details required by the CTX EDI Addenda process are not available on the VCUST table. Instead, this information must be retrieved from the EFT transaction that created the electronic payment to the vendor. In order to fulfill the requirement to generate CTX EDI Addenda details for miscellaneous vendors, the system will modify the source field values for the beginning segment for Payment Order / Remittance Advice as follows:

  - If the vendor on the EFT transaction is miscellaneous and no payee detail is populated on the Header of the transaction:

The source field for Segment ID BPR13 (DFI ID Number) will be retrieved from the ABA Number field on the Vendor Line’s Misc. Vendor ACH Details section of the EFT transaction.

The source field for Segment ID BPR15 (Account Number) will be retrieved from the Bank Account Number field on the Vendor Line’s Misc. Vendor ACH Details section of the EFT Transaction.

- If the payee detail is populated on the Header of the transaction, the source field for Segment ID BPR13 (DFI ID Number) will be retrieved from the ABA Number field while the Segment ID BPR15 (Account Number) will be retrieved from the Bank Account Number field on the VCUST table.
- In case of Non-Miscellaneous Vendors, the required details will be pulled from the VCUST table. The source field for Segment ID BPR13 (DFI ID Number) will be retrieved from the ABA Number field while the Segment ID BPR15 (Account Number) will be retrieved from the Bank Account Number field on VCUST table.

Progression messages will be issued after processing the number of records specified in the progression counter parameter.

Process Steps	Messages
1. Parameter Validation	• Validating Batch Parameters • Parameters are valid or invalid depending on the Validation. If the parameter is invalid, the invalid value will be displayed in the log. • If the required parameters are not provided then <<PARM_NM>> is required message is added on Job Log • Parameter validation completed
2. Clearing the CTX EDI Segment table	• CTX EDI Segment table is emptied
3. Selection of Records	• Selecting eligible records • Selecting records from AD_DOC_HDR that are matching with AD Stub Details table and not matches with CTX EDI Addenda Table • If the selection returns 0 records, then the following message will be issued: “No eligible records found for formatting”. • Otherwise “Total number of eligible records found:” i.e. number of records (count) selected will be displayed • Selection of eligible records completed
4. Creation of EDI Segments and inserting records into the CTX EDI Segment table	• Processing the selected records • Depending on the ‘Progression Counter’ following log messages will be displayed: • Generating EDI Segment record • Generating Segment Header section • Generating Segment Details section • Generating Segment Trailer section • Inserting record into the CTX EDI Segment table

Process Steps	Messages
	• <<count>> segment records are generated and inserted into the CTX EDI Segment table
5. Inserting records into the CTX EDI Addenda table	• Selecting all records from the CTX EDI Segment table • If the selection returns 0 records, then the following message will be issued: “No records found on CTX EDI Segment table”. • Otherwise “Total number of records found on CTX EDI Segment table:” i.e. number of records (count) selected will be displayed • Selection of records completed • Depending on the ‘Progression Counter’ following log messages will be displayed: • <<count>> records on CTX EDI Segment table are processed • <<count>> records are inserted on the CTX EDI Addenda table • Processing completed

Restartability Information:

This process is implemented with checkpoints. If the process fails for any reason (such as the network is down or the server is down), then the process restarts from the last point where it stopped.

If the job fails in any of the above steps the job can be restarted after resolving the error. If the job is restarted, it will start from the last point where it stopped.

For example, the job failed after processing 2 records from the total of 25 eligible records due to some fatal condition (table space error). If the job is restarted after resolving the table space issue, the job will start by selecting the records from 3rd transaction.

Instead of restarting the job, if the job is rescheduled with the same parameter value after resolving the table space issue, the job will start from the beginning.

The restart will fail if another instance of this job has been scheduled and ran successfully before restarting the failed job. This is because the CTX EDI Segment and CTX EDI Addenda tables are populated for earlier selection of records.

Major Input

- AD transaction header (AD_DOC_HDR) table
- AD Stub table
- CTX EDI Addenda (CTX_EDI_ADDEND) table

- Disbursement Format (R_AP_DISB_FRMT) table
- Vendor/ Customer (R_VEND_CUST) table

Parameter	Description	Default Value
FROM_RUN_ID	From Run ID Conditionally required. If value is populated in on the “To Run ID” parameter, then a value is required in this parameter.	No Default
TO_RUN_ID	To Run ID Conditionally required. If value is populated in on the “From Run ID” parameter, then a value is required in this parameter.	No Default
PROCESS_IND	Processing Indicator This is a required parameter. The ‘Processing Indicator’ can be set to either ‘T’ (i.e. ‘Test’) or ‘P’ (i.e. ‘Production’)	No Default
EFFECT_DT_SRC	Effective Date Source This is a required parameter. The ‘Effective Date Source’ can be set to either ‘1’ (i.e. ‘Cleared Date’) or ‘2’ (i.e. ‘Issue Date’)	No Default
COMMIT_SIZE	Commit Block Size Optional.	100
SEL_BLK_SIZE	Select Block Size Optional.	100
PROG_CNTR	Progression Counter Optional.	100

Major Output

- CTX EDI Addenda (CTX_EDI_ADDEND) table
- CTX EDI Segment (CTX_EDI_SEG) table

Job Return code

Return Code	Condition
Successful (1)	All the selected EFT transactions are processed successfully
Warning (4)	No eligible records found. This could be because of the following reasons: • No eligible records found on AD transaction header • No records found on the CTX EDI Segment table
Non-Fatal Error (8)	We are not setting this return code.
Failed (12)	The job fails under the following conditions: • Parameters are invalid • Run time exceptions for unexpected situations. • Restart failed because another instance of the process has already been run successfully
Terminated (16)	This return code will be issued when the job is terminated by the user.
System Failure (20)	This return code will be issued when the job is terminated because of database server or network issues.

Sort Criteria

The records are selected by ordering them –

RUN_ID, DOC_CD, DOC_DEPT_CD, DOC_ID, DOC_VERS_NO, BANK_ACCT_CD, CHK_NO

Selection Criteria

The selection of the eligible EFT records is dependent on the parameters. If the Run ID parameters are not entered then the job selects all the EFT transactions from the AD Transaction Header where:

- Transaction Phase Code = *Final* and
- Transaction Function Code = *New* and
- Disbursement Type = *EFT* and
- Disbursement Format = *CTX* and
- ACH File Generation Status = *Ready for Generation* and
- No records found on the CTX EDI Addenda Table that matches the Bank Account and the Check EFT number of the selected EFT transaction and
- Matching record exists on the AD Stub Details table

If both of the parameters From Run ID and To Run ID are specified, then the process selects all EFT transactions on the AD Transaction Header where;

- Run ID within the specified From Run ID / To Run ID range and
- Transaction Phase Code = *Final* and
- Transaction Function Code = *New* and
- Disbursement Type = *EFT* and
- Disbursement Format = *CTX* and
- ACH File Generation Status = *Ready for Generation* and
- No records are found on the CTX EDI Addenda Table that matches the Bank Account and the Check EFT number of the selected EFT transaction and
- Matching record exists on the AD Stub Details table

Problem Resolution

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All the parameters are validated successfully	N/A	N/A
Warning (4)	Job ended with a Warning because there are no eligible records on AD transaction header table as the Addenda records already exist on CTX EDI Addenda table or no records on the CTX EDI segment table while inserting records on the CTX EDI Addenda table. **Sample Message:** No eligible records found for formatting. No records found on CTX EDI Segment table		Alternatively the job can be rescheduled with different set of parameters.

Possible Return Codes	Condition	Recommendation	Other Instructions
Non-Fatal Error (8)	We are not setting this return code.		
Failed (12)	Job failed due to Fatal conditions	In this step, the job can fail under the following two conditions. 1) Encounters any runtime exceptions and 2) Failed during restart. If the job fails because of the runtime exceptions, investigate the Logger file, resolve the error and restart the job.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.
	Failed while restarting the job since another instance of the job has already been run successfully.	**Recommendation:** Schedule a new job.	
Terminated (16)	Job is terminated manually by the user.	Reason for the termination needs to be investigated. The job can either be restarted or schedule a new job.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.
System Failure (20)	When the job is terminated because of database server or network issues	Reason for the System Failure needs to be investigated. The job can either be restarted or schedule a new job.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.

2.1.12 Disbursement Correction

Description

Disbursement Correction is an optional batch process used to perform corrective actions such as void, renumber, and reprint checks that were produced in the disbursement cycle. The Disbursement Correction process is generally used after the printing of checks, and must generally occur before the Disbursements are "journalized" (i.e. posted to the Journal). Only the 'Reprint' action can be performed after journals are posted. The business process for Disbursement Printing is as follows:

1. The Automated Disbursement chain selects payment requests, generates disbursement transactions (AD and EFT transactions), and creates printing output.
2. The Disbursement Printing job selects checks to be printed and creates printing output.
3. The printing output (an ".xml" file for BIRT Server) is sent to the printing server.
4. The printing of checks is controlled on the printing server, and is subject to operator intervention, as operators must load check stock (Plain or Preprinted) and monitor printing.
5. If printing errors occur, Disbursements/Operations staff with appropriate security enters the correction information on the Disbursement Correction (DISBCP) parameters page and schedules the Disbursement Correction process.
    - The Disbursement Correction process uses the entries on the Disbursement Correction Parameters (DISBCP) page to select the appropriate records either from the Check Reconciliation (CHREC) table or from the disbursement transaction header, depending upon the Correction Type, and makes the appropriate updates to the Bank page, disbursement transaction, and Check Reconciliation page.
    - More details on the possible Correction Types appear below.
6. Assuming that "Asynchronous" posting is used for disbursement transactions, once printing issues are resolved, and all Disbursements are successfully printed, the Disbursements can be "Journalized"
    - Unlike other transactions, the Journals and Ledgers are often not updated at the time of AD Transaction Submit. Instead because the AD and EFT transactions are setup for asynchronous posting, the Journal Posting Indicator in the Disbursement Posting Lines is set to "Not Ready for Posting" when the disbursement transaction submits.
    - Once the Disbursement Corrections process is completed, the Journal Posting Initiator process must be executed to set the Journal Posting Indicator to "Ready to Post." Then the Journal Posting Engine must be executed to post these records to the Journal.
    - The reason for this delayed update to Journals is that once the Journal is updated with the check numbers, then none of 'Renumber' or 'Reassign' Correction Types can be invoked because the Journal and Ledger information cannot be modified by any of the Offline processes.

The following Correction Types can be performed on an individual check or range of checks using the Disbursement Correction batch process:

- **Void** –The 'Void' Correction Type can only be used to void checks that are not on Check Reconciliation or Paid Check tables). Once a record exists on the Check Reconciliation table, the check must be manually voided on the Check Reconciliation table using the Check Status Update link, rather than on the Disbursement Correction. (Note that if a check's status is changed to 'Void' on the Check Reconciliation table, no accounting entries are generated for the disbursement transaction. A JV must be processed in order to make any accounting

related changes.). The 'Void' Correction Type can be performed on either plain or preprinted check stocks. This action inserts records onto CHREC for the requested checks and set the Check Status to 'Void'.

- **Undo Void** – Incorrectly voided checks can be un-voided by using this Correction Type. This action can be used on either plain or preprinted check stocks. This action removes the voided checks from CHREC table.
- **Renumber** – There may be a need to renumber checks in case the Next Check number on the Bank table was assigned incorrectly. For example, check 420 might need to be renumbered to 402 because it was incorrectly entered as the Next Check Number on the Bank Table. This action can work in conjunction with one another to correct applicable checks before printing the checks.

  'Renumber' Correction Type should only be used if the Disbursement Printing job has not been executed. The Disbursement Printing job creates the .xml file that is used as input in the print server. If the Disbursement Printing job has been executed, the 'Renumber' Correction Type should not be used because this type does not recreate a new .xml file which is needed to reflect the new check number when the check is physically printed. If Disbursement Printing has already been executed and you choose to use this Correction Type, you must also insert a record for the Correction Type of 'Reprint' on this table so that the new checks numbers are update to the .xml file when the Disbursement Correction process is executed.

  The 'Renumber' Correction Type is not allowed once the disbursement transaction has been posted to journals. This action can be used on either plain or preprinted check stocks. This action performs the following updates:

Changes the Check Status of the old check to "Renumbered" from Disbursed/Warranted. It updates the Cleared Date of the 'Renumbered' checks with the current Application Date.

Copies the voided check and its information.

Inserts a record onto CHREC table for the new check using the New Start Number from DISBCP. The Cleared Date is blank for the inserted check.

Updates the disbursement transaction header and the Posting Line Catalog with the new check number.

Increments the Next Check Number field on the Bank table to the last check assigned plus one.

- **Void/Renumber** - Voiding and renumbering actions is generally used to renumber checks when the Next Check Number on the Bank table was incorrectly updated. This action can work in conjunction with one another to correct applicable checks before printing the checks.

  'Void/Renumber' Correction Type should only be used if the Disbursement Printing job has not been executed. The Disbursement Printing job creates the .xml file that is used as input in the print server. If the Disbursement Printing job has been executed, the 'Void/Renumber' Correction Type should not be used because this type does not recreate a new .xml file which is needed to reflect the new check number when the check is physically printed. If Disbursement Printing has already been executed and you choose to use this Correction Type, you must also insert a record for the Correction Type of 'Reprint' on this table so that the new checks numbers are update to the .xml file when the Disbursement Correction process is executed.

  This action is allowed only on preprinted check stock. This action performs the following updates:

Changes the Check Status of the requested check numbers on CHREC table to 'Void'. It updates the Cleared Date of the 'Void' checks with the current Application Date.

Copies the voided check and its information.

Inserts a new check number on CHREC (with the copied data) using the New Start Number from DISBCP. The Cleared Date is blank for the inserted check.

Updates the disbursement transaction header and the Posting Line Catalog with the new check number.

Increments the Next Check Number field on the Bank table to the last check assigned plus one.

- **Reassign/Reprint** – When a check/warrant is damaged or destroyed during the printing process and requires replacement, it may need to be reassigned and then reprinted. This Correction Type can be used on either preprinted or plain check stocks.

  This Correction Type is not allowed once the disbursement transaction has been posted to journals. This action performs the following updates:

  - Changes the Check Status of the requested check numbers on CHREC table to 'Void'. It updates the Cleared Date of the 'Void' checks with the current Application Date.
  - Inserts a new record for the requested check number on CHREC. The Cleared Date is blank for the inserted check. The Transaction ID, Check Amount, and Issue Date will remain the same as that of the check number that it was reassigned from. The Check Status will be set to either Disbursed or Warranted depending on the Disbursement Option setting on SOPT.
  - Updates the disbursement transaction header and the Posting Line Catalog with the new check number.
  - Updates the Print Status Indicator on the disbursement transaction to 'Ready for Reprint'.
  - Increments the Next Check Number field on the Bank table to the last check assigned plus one.

- **Void/Renumber/Reprint** – Void/Renumber/Reprint Correction Type is used to correct checks when a check is damaged, destroyed, or incorrectly numbered during the printing process and requires replacement. This Correction Type is only allowed on preprinted check stock. This action performs the following updates:

  - Changes the Check Status of the requested check numbers on CHREC table to 'Void'. It updates the Cleared Date of the 'Void' checks with the current Application Date.
  - Copies the voided check and its information.
  - Inserts a new check number on CHREC (with the copied data) using the New Start Number from DISBCP. The Cleared Date is blank for the inserted check.
  - Updates the disbursement transaction header and the Posting Line Catalog with the new check number.
  - Changes the Print Status Indicator on the AD Transaction Header to "Ready for Reprinting."
  - Increments the Next Check Number field on the Bank table to the last check assigned plus one.

- **Reprint** - The need to reprint occurs when a check is damaged or destroyed during the printing process and requires replacement. This Correction Type is allowed on preprinted or plain check stocks. 'Reprint' Correction Type is the only type that can be done before or after journals have been posted. This Correction Type changes the Print Status Indicator on the disbursement transaction from 'Printed' to "Ready for Reprinting" for the requested records. This Correction Type does not increment the Next Check Number on the Bank table.

When to Run

On demand

Major Input

- Disbursement Correction (R_AP_DISBCORR_PARM)
- Check Reconciliation (R_AP_CHK_RECON)
- Bank (R_AP_BANK_ACCT)
- Posting Line Catalog

Output

- Check Reconciliation (R_AP_CHK_RECON)
- AD Transaction (AD_DOC_HDR)
- Bank (R_AP_BANK_ACCT)
- Posting Line Catalog
- Disbursement Correction Report - An Exception report is generated at the end of the Disbursement Correction process that lists all checks that were corrected.
- Disbursement Correction Exception Report - An Exception report is generated at the end of the Disbursement Correction process that lists all DISBCP parameter records that failed to process. The report displays a section for each parameter record from the DISBCP page. For each check, it lists the reasons why it failed to process. There is a page break by Bank Account.

Parameters

The following parameters are required for the Disbursement Correction process.

Custom Parameters

Disbursement Correction (*R_AP_DISBCORR_PARM)*

Description (Caption)	Parameter (Attribute) Name	Data Type/Length
Check Stock Type	STK_TYPE	CVL_STK_TYP
Bank Account Code	BANK_ACCT_CD	Text/4
Correction Type	CORR_ACTN	Number/Integer
Correction Reason	CORR_REAS	Text/50
Start Number	OLD_STRT_CHK_NO	Text/15
End Number	OLD_END_CHK_NO	Text/15

Description (Caption)	Parameter (Attribute) Name	Data Type/Length
New Start Number	NEW_STRT_CHK_NO	Text/15
Active	ACTIVE_STA	Yes/No
Process Date	PROCESS_DT	Date

Batch Parameters

Job	Parameter	Description	Default Value
Disbursement Correction	Client Name for Report (CLIENT_NM)	Optional field. Entry of a value in this field specifies the name that will appear on the report.	No Default
	MODE_NAME	Required Field. This field refers to the Disbursement Correction Mode and should always be set to '1'.	1

Sort Sequence

Parameter Code

Selection Criteria

1. Selection of records from the Disbursement Correction Parameter table
   - Active Status is 'true' and Process Date is null
2. Selection of records for VOID action:
   - Start Check Number and End Check Number of Parameter Record are not available in the Check Reconciliation Table.
   - Bank Account Code is the same as Bank Account Code of the parameter record.
3. Selection of records from the Check Reconciliation table for all the disbursement correction actions except VOID action:
   - Check EFT number in the Check Reconciliation Table is between Start Check Number and End Check Number of the parameter record.
   - Bank Account Code is the same as Bank Account Code of parameter record.
4. Select records from AD Transaction Header where
   - Transaction Phase code = FINAL

- Transaction function code is not = CANCELLED
- Journal Posting Indicatory = 'Not Ready to Post'

5. Select records from the Bank Account Table when Bank Account Code is the same as Bank Account Code from Parameter Record.

Problem Resolution

The process has a rerun procedure as follows:

The ranges of records are picked up from the parameter table and records are inserted/updated in the Check Reconciliation table, the AD Transaction Header, and the Bank Record table. If the process does not find any corresponding record in the transaction header for a specific check within the specified check range, the process logs an error message and continues normal processing for the remaining checks.

If the process fails, the parameter record is not made inactive unless it loops through all the records. That is, the next time the process is run, the same parameter is picked up, the correction action continues normally, and the corrected checks are not picked up from the Check Reconciliation table.

2.1.13 Disbursement Discard Chain

Chain or Job Name	Online Disbursement Discard Chain Job
Recommended Frequency	Nightly cycle or on demand
Single Instance Required	Yes
Can be restarted?	No
Reports generated	Yes

Overview:

The Online Disbursement Discard Chain job is an automated process to discard unprocessed Manual Disbursements where each accounting line of the Manual Disbursement references a payment request and all of the referenced payment requests have already been disbursed, that is, none of the referenced payment requests appear on the DISRQ table.

There are four batch jobs in this chain job:

- Disbursement Selection
- SMU Reject
- SMU Discard
- Discard Report

The acceptable job return code configuration depends on the business requirement. For example, if the requirement is that the subsequent jobs in the chain should continue only if the job ends with a return code of *Successful*, the acceptable job return codes for all of the jobs should be set to *Successful*. If for some jobs in the chain, a Non-Fatal error is an acceptable job return code, then that can also be configured. These configurations can be done in the Job Setup page.

If any of the jobs in the chain ends with a return code of *Failed*, *Terminated* or *System Failure*, all of the subsequent jobs are set to *Inactive*.

Major Input

MD_DOC_HDR - This table will be queried to see which MDs can be discarded.

MD_DOC_ACTG - This table is analyzed to see if any references are Memo or made to Payment Request style transactions.

R_AP_DISB_RQST - This table is queried to ensure that the MD to be discarded does not contain any DISRQ records for the transaction the MD is referencing.

R_GEN_DOC_CTRL - If specific Transaction Codes are supplied, validate to ensure these transaction codes are MD Transaction Types.

Major Output

- Disbursement Selection
  - AllMDDiscardDocs.txt
  - PendingMDDiscardDocs.txt
- SMU Reject
  - Rejected transactions
- SMU Discard
  - Discarded transactions
- Discard Report
  - Discard report file

Chain Job Return code

The following table shows the potential return codes for the Disb Discard chain job. Note that the Chain job will end with the highest return code across all of the jobs.

Return Code	Condition
Successful (1)	All the Accounting Journal records are processed successfully
Warning (4)	No journal records found to process.
Non-Fatal Error (8)	N/A
Failed (12)	The job fails under the following conditions: • Parameters are invalid • Run time exceptions for unexpected situations.
Terminated (16)	This return code is issued when the job is terminated by the user. When this job ends with a return of code *Terminated* subsequent jobs in the chain will be set to *Inactive*.
System Failure (20)	This return code is issued when the job is terminated because of database server or network issues. When this job ends with a return of code *System Failure*.

Problem Resolution

If any of the jobs in the chain failed due to application errors it is advisable to restart the job after correcting the errors instead of rescheduling the job. Restarting the job will reduce the processing time since the job will resume from where it has last committed and select only the unprocessed records.

Please refer to the individual jobs for details regarding the specific job processes and problem resolution.

Disbursement Discard: Disbursement Selection

Chain or Job Name	Disbursement Selection
Recommended Frequency	Daily as part of the nightly cycle or on demand
Single Instance Required	Yes
Can be restarted?	No
Reports generated	No

Overview:

This batch job identifies the set of MDs to be discarded based upon certain criteria and write the transactions information out to an SMU parameter file. If any of those transactions are in a pending status, it writes those transactions out to a second parameter file for SMU Reject, to get the pending transactions in draft.

Process Steps

The following table shows the various steps that the Disbursement Selection Job goes through and the messages issued at each step:

Process Steps	Messages
1. Validate the Parameters	• Started Validating the Parameters • If parameters are invalid, then write the invalid param to job log
2. Build the query to identify the MDs to be discarded	• Build Update SQL to be executed • Execute the Update SQL
3. Write the transactions to be discarded to a Param file.	• Write out the transactions from Step 2 into a param file for the SMU process.

Restartability

Job cannot be restarted.

Major Input

Data from the following tables:

- MD_DOC_HDR
- MD_DOC_ACTG
- R_AP_DISB_RQST

- R_GEN_DOC_CTRL

Batch Parameters

Parameter	Description	Default Value
DOC_CD	(Required) This is a single Transaction Code or Multiple Transaction Codes in a comma separated list. If ALL supplied then all Transaction Codes are processed. Transaction Codes must be of MD Transaction Type.	
SELECT_BLOCK	(Optional) This is the number of records supplied when executing a query. It exists for different sites to tune performance.	3000
PROG_CTR_SZ	(Optional) This is the number of records after which a progression message will be pushed out to the job log. It exists for sites to ensure that when this job is long running and processing many records, we log messages so that the site knows the job is still running.	5000
COMMIT_BLOCK	(Optional) This is the number of records after which a commit should be performed. The commit is not only for writing to the DB but also appears to be a point for memory to be cleared out. It exists for different sites to tune performance.	1000
AMSEXPORT	(Required) But should not be overridden by the user; it exists so that we can access this common parameter inside of this job.	null

Major Output

- AllMDDiscardDocs.txt
- PendingMDDiscardDocs.txt

Job Return Code

The following table shows the potential job return codes for the Validations job:

Return Code	Condition
Successful (1)	All the Accounting Journal records are processed successfully
Warning (4)	No journal records found to process.
Non-Fatal Error (8)	N/A
Failed (12)	The job fails under the following conditions: • Parameters are invalid • Run time exceptions for unexpected situations.
Terminated (16)	This return code is issued when the job is terminated by the user. When this job ends with a return of code Terminated subsequent jobs in the chain will be set to inactive.
System Failure (20)	This return code is issued when the job is terminated because of database server or network issues. When this job ends with a return of code System Failure.

1. Step 1: Parameter Validation
    a. Validate the parameters, if all the parameters are validated successfully then make a call to update the Disbursement Request records
2. Step 2 & 3: Updating the records & Writing the Param file.
    a. Build the SQL to identify all of the MDs to be Rejected and Discarded
    b. Store inside of a return variable the SELECT aspect which will be the fields:

Transaction Type

Transaction Code

Transaction Department Code

Transaction ID

Transaction Version Number

Transaction Phase Code

    c. Add the FROM clause as MD_DOC_HDR
    d. Add the WHERE:

The Transaction Code is in the list or parameter Transaction Codes supplied or All Transaction Codes, AND

The transaction is New AND

The transaction is in either a Draft or Pending status AND

The every reference at the AL Level of the MD is to a PR or ABS Transaction Type AND

There are not any Memo references at the AL Level of the MD AND

No R_AP_DISB_RQST records exist for any of the transactions referenced at the AL Level of the MD

e. Add the ORDER BY Fields:

Transaction Phase Code

Transaction Code

Transaction Department Code

Transaction ID

Transaction Version Number

f. Execute the query using getResultSetBySQL

g. If the result set is null Then

   i. Set the job return code to be Warning

   ii. Save message to the job log that no records found to be discarded

   iii. Return success to caller

h. Start walking through the set of MDs

   i. If local progression counter >= member progression counter size Then

      Write out message to job log with the transaction that was previously processed

   ii. Reset local progression counter

   iii. If Call to store MD inside of Parm Files fails Then

      Return failure to caller

Problem Resolution

The following table shows the possible return codes and recommendations for each processing step.

Step 1: Parameter Validation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All the parameters are validated successfully.	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A

Possible Return Codes	Condition	Recommendation	Other Instructions
Failed (12)	Job failed due to Fatal conditions	In this step, the job can fail under the following conditions. 1) Encounters any runtime exceptions and 2) Failed during restart. 3) If the Transaction Code is invalid 4) If the file path is invalid. If the job fails because of the runtime exceptions, investigate the exception reported by the process, resolve the error and restart the job.	Schedule a new job
Terminated (16)	Job is terminated manually by the user.	Reason for the termination needs to be investigated. A new job can be scheduled	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.
System Failure (20)	When the job is terminated because of database server or network issues	Reason for the System Failure needs to be investigated. A new job can be scheduled.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.

Step 2 & 3: Updating the records & Writing the Param file.

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	N/A	N/A	N/A
Warning (4)	If the result set of the MDs is null	N/A	Save message to the job log that no records

			found to be discarded
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Failed because of runtime exceptions for an unexpected situation.	The job can fail under these conditions: • If the number of records updated is less than Zero • If committing of this transaction fails Failure reason needs to be investigated before scheduling a new job.	Failed because of runtime exceptions for an unexpected situation.
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated and since the job can be restarted	N/A
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated and since the job can be restarted	N/A

Disbursement Discard: SMU Reject

Chain or Job Name	SMU Reject
Recommended Frequency	Nightly
Single Instance Required	Yes
Can be restarted?	Yes
Reports generated	None

Overview

The SMU Reject job reads the parameter file with all the MDs that should be rejected out of *Pending* into *Draft* status, for eventual discard.

Restartability Information

Job cannot be restarted. If the job fails then the entire Param file is read and the MDs are rejected out of the *Pending* status to the *Draft* status.

Major Input

- PendingMDDiscardDocs.txt

Batch Parameters

Parameter	Description	Default Value
Parameter Directory AMSPARM	Parameter Directory location	$$AMSPARM$$
Chain Job ID CHAIN_JOB_ID	Chain Job ID	$$@CHAINJOBID@$$
Parameter File Name PARM_FILE_NM	The Parameter File Name	PendingMDDiscardDocs.txt
Table Action TABLE_ACTN	The table action that needs to be done.	Reject

Job Return code

The following table shows the potential job return codes for the SMU Reject job.

Return Code	Condition
Successful (1)	All the records in the txt file are processed successfully
Warning (4)	This job does not issue this Return Code.
Non-Fatal Error (8)	This job does not issue this Return Code.
Failed (12)	The job fails under the following conditions: • Issues in SysManUtil while discarding the transaction • Technical/System failure
Terminated (16)	This return code is issued when the job is terminated by the user. When this job ends with a return of code Terminated subsequent jobs in the chain will be set to inactive.
System Failure (20)	This return code is issued when the job is terminated because of database server or network issues. When this job ends with a return of code System Failure, subsequent jobs in the chain will be set to inactive.

Sort Sequence

N/A

Selection Criteria

N/A

Problem Resolution

The following table shows the possible return codes and recommendations for each processing step.

Step 1: SysManUtil – Reject

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All the records are successfully rejected. This is done by the SysManUtil "Reject" action functionality.	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Job failed due to Fatal conditions	In this step, the job can fail under the following two conditions. 1) Encounters any runtime exceptions and 2) Failed during restart. If the job fails because of the runtime exceptions, investigate the exception reported by the process, resolve the error and restart the job.	Schedule a new job
Terminated (16)	Job is terminated manually by the user.	Reason for the termination needs to be investigated. A new job can be scheduled	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.
System Failure (20)	When the job is terminated because of database server or	Reason for the System Failure needs to be investigated. A new job	If another instance of the job has already been scheduled and ran successfully, then

Possible Return Codes	Condition	Recommendation	Other Instructions
	network issues	can be scheduled.	this job should not be restarted – only new job should be scheduled.

Disbursement Discard: SMU Discard

Chain or Job Name	SMU Discard
Recommended Frequency	Nightly
Single Instance Required	Yes
Can be restarted?	No
Reports generated	None

Overview

The SMU discard job reads the parameter file that contains all the MDs that should be discarded.

Restartability Information

Job cannot be restarted. If the job fails then the entire Param file will be read and the MDs will be discarded.

Major Input

- AllMDDiscardDocs.txt

Batch Parameters

Parameter	Description	Default Value
Parameter Directory AMSPARM	Parameter Directory location	$$AMSPARM$$
Chain Job ID CHAIN_JOB_ID	Chain Job ID	$$@CHAINJOBID@$$
Parameter File Name PARM_FILE_NM	The Parameter File Name	AllMDDiscardDocs.txt

Table Action TABLE_ACTN	The table action that needs to be done.	Reject

Job Return code

The following table shows the potential job return codes for the SMU Discard job.

Return Code	Condition
Successful (1)	All the records in the txt file are processed successfully
Warning (4)	This job does not issue this Return Code.
Non-Fatal Error (8)	This job does not issue this Return Code.
Failed (12)	The job fails under the following conditions: • Issues in SysManUtil while discarding the transaction • Technical/System failure
Terminated (16)	This return code is issued when the job is terminated by the user. When this job ends with a return of code Terminated subsequent jobs in the chain will be set to inactive.
System Failure (20)	This return code is issued when the job is terminated because of database server or network issues. When this job ends with a return of code System Failure, subsequent jobs in the chain will be set to inactive.

Sort Sequence

N/A

Selection Criteria

N/A

Problem Resolution

The following table shows the possible return codes and recommendations for each processing step.

Step 1: SysManUtil – Discard

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All the records are successfully discarded. This is done by the SysManUtil "Discard" action functionality.	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Job failed due to Fatal conditions	In this step, the job can fail under the following two conditions. 1) Encounters any runtime exceptions and 2) Failed during restart. If the job fails because of the runtime exceptions, investigate the exception reported by the process, resolve the error and restart the job.	Schedule a new job
Terminated (16)	Job is terminated manually by the user.	Reason for the termination needs to be investigated. A new job can be scheduled	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.
System Failure (20)	When the job is terminated because of database server or network issues	Reason for the System Failure needs to be investigated. A new job can be scheduled.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.

Disbursement Discard: Discard Report

Chain or Job Name	Discard Report
Recommended Frequency	Daily as part of the nightly cycle or on demand
Single Instance Required	Yes
Can be restarted?	No
Reports generated	DisbDiscard Report

Overview:

This batch job is part of a chain job which is responsible for identifying a set of MD Transactions to be Discarded and Discarding the transactions. This step of the chain reports upon the success or failure of the two SMU steps for rejecting pending MDs and discarding MDs.

Process Steps

The following table shows the various steps that the Disbursement Selection Job goes through and the messages issued at each step:

Process Steps	Messages
1. Validate the Parameters	• Started Validating the Parameters • If parameters are invalid, then write the invalid param to job log
2. Acquire the MDs to be discarded	• Start acquiring the MDs that were supposed to be Discarded.
3. Render Report.	• Started Disb Discard Report generation.

Restartability

Job cannot be restarted.

Major Input

- AllMDDiscardDocs.txt
- MD_DOC_HDR

Batch Parameters

Parameter	Description	Default Value
SELECT_BLOCK	(Optional) This is the number of records supplied when executing a query. It exists for different	3000

	sites to tune performance.	
PROG_CTR_SZ	(Optional) This is the number of records after which a progression message is pushed out to the job log. It exists for sites to ensure that when this job is long running and processing many records, we log messages so that the site knows the job is still running.	5000
COMMIT_BLOCK	(Optional) This is the number of records after which a commit should be performed. Please understand that the commit is not only for writing to the DB but also appears to be a point for memory to be cleared out. It exists for different sites to tune performance.	1000
AMSEXPORT	(Required) But should not be overridden by the user; it exists so that we can access this common parameter inside of this job.	null

Major Output

- Disb Discard Report.

Job Return Code

The following table shows the potential job return codes for the Validations job:

Return Code	Condition
Successful (1)	All the MDs are successfully discarded.
Warning (4)	No journal records found to process.
Non-Fatal Error (8)	N/A
Failed (12)	The job fails under the following conditions: • Parameters are invalid • Run time exceptions for unexpected situations.
Terminated	This return code is issued when the job is terminated by the

Return Code	Condition
(16)	user. When this job ends with a return of code Terminated subsequent jobs in the chain will be set to inactive.
System Failure (20)	This return code is issued when the job is terminated because of database server or network issues. When this job ends with a return of code System Failure.

Step 1: Parameter Validation

Validate the parameters, if all the parameters are validated successfully then make a call to acquire the MDs that were supposed to be discarded.

Step 2: Report on the Discarded MDs

- Get a handle to the Discard Parm File
- Start Walking through the File

Get the Next Parm Line's Transaction Code, Transaction Dept Code, Transaction ID, Transaction Version Number, set into member variables

Increment the commit and progression counters

Build the Search Request

Call to see if the Transaction exists inside of MD_DOC_HDR; getObjectCount

If Transaction header does not exist then

- Increment successful discard counter
- Write out the transaction info to the report and success message

Else

- Increment failed discard counter
- Write out the transaction info to the report and failed message
- Write out to the report the number of successful and failed transactions discarded

- Return the number of records processed

Step 3: Generate Report

- Build the Report Header
- If no records were written to the Discard Report Then

  Write into the Comment Section the text “Nothing written to report!”
- Save the report [saveReportXML]

Problem Resolution

The following table shows the possible return codes and recommendations for each processing step.

Step 1: Parameter Validation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All the parameters are validated successfully.	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Job failed due to Fatal conditions	In this step, the job can fail under the following conditions. 1) Encounters any runtime exceptions and 2) Failed during restart. 3) If the file path is invalid. If the job fails because of the runtime exceptions, investigate the exception reported by the process, resolve the error and restart the job.	Schedule a new job
Terminated (16)	Job is terminated manually by the user.	Reason for the termination needs to be investigated. A new job can be scheduled	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.
System Failure (20)	When the job is terminated because of database server or network issues	Reason for the System Failure needs to be investigated. A new job can be scheduled.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted –

Possible Return Codes	Condition	Recommendation	Other Instructions
			only new job should be scheduled.

Step 2: Reporting on the Discarded MDs.

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	N/A	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Failed because of runtime exceptions for an unexpected situation.	The job can fail under these conditions: • If the Transaction Header exists on MD_DOC_HDR Failure reason needs to be investigated before scheduling a new job.	Failed because of runtime exceptions for an unexpected situation.
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated and since the job can be restarted	N/A
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated and since the job can be restarted	N/A

Step 3: Generating Report.

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	N/A	N/A	N/A
Warning (4)	N/A	N/A	N/A

Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Failed because of runtime exceptions for an unexpected situation.	The job can fail under these conditions: Failure reason needs to be investigated before scheduling a new job.	Failed because of runtime exceptions for an unexpected situation.
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated and since the job can be restarted	N/A
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated and since the job can be restarted	N/A

2.1.14 Disbursement Parameters Updates

Description

The Disbursement Parameters table holds the records that define the selection criteria and is used during the AD Chain. The Disbursement Parameters Update batch process is an automated mechanism for updating the Disbursement Parameters (DISPA) table. This allows user maintenance of this table to be reduced.

The process will automatically update all active records on the DISPA table with the 'To Date' selection criteria for the next disbursement process. It will also update the Check Date to be used by the next disbursement process.

When to Run

This batch process should be executed before the AD Chain (that is, disbursement process jobs) or a user may manually update disbursement parameters before the AD Chain is executed.

Major Input

- Disbursement Parameters (R_AP_DISB_PARM)

Output

Table Updates

Update the selected records by adding the Increment To Date Days value to the To Date and adding the Increment Check Date Days batch parameter value to Check Date on the Disbursement Parameters table.

If the calculated To Date and Check Date are defined as a weekend or a holiday (Holiday Flag is *True* and/or Federal / Bank Holiday flag is *True*) on the CLDT table, then one or more additional day(s) will be added to the calculated date until the Calendar Date entry for that day is not a weekend or a holiday.

Parameters

Job	Parameter	Description	Default Value
Disbursement Parameters Updates	Increment To Date Days (INCREMENT_TO_DATE_DAYS)	Required Field.	No Default
	Increment Check Date Days (INCREMENT_CHECK_DATE_DAYS)	Required Field.	No Default

Sort Criteria

None

Selection Criteria for Batch process

Select all records from the Disbursement Parameters (R_AP_DISB_PARM) table, which have a **Status** of *Active*.

Problem Resolution

No database restore is required. Correct the problem and rerun the job executing the program. No restoration of datasets or files from backups is required for this program.

2.1.15 Disbursement Printing

Job Name	Disbursement Printing
Recommended Frequency	This job can be scheduled in offline mode after the completion of the Automated Disbursement process. This job can also be run on demand. Online printing can be invoked using the Payment Printing action from the Manual Disbursement (MD) transaction.
Single Instance Required	Yes
Can be restarted?	No
Reports generated	No

Overview

The Disbursement Printing job in CGI Advantage prints checks, warrants and prints/emails remittance advices with check stub details consolidated and non-consolidated on Reference Transaction ID, Vendor Invoice No and Vendor Invoice Line Number when the Internal Disbursement model is used. This job generates only an xml file that can be used on an external application to print checks/warrants/remittance advice.

The Disbursement Printing job can be invoked from two different places as discussed below:

1. Invoking the printing action from the Manual Check transaction (MD) – Online Mode

   When the job is invoked from the MD transaction, the job will only print that particular transaction. This job, when invoked from the transaction, will print the Check / Warrant irrespective of the Disbursement model. Only a Check or Warrant can be printed using this option. Remittance Advices cannot be printed using this option. When the External Disbursement flag on SOPT is *True,* the transaction will not print anything. The transaction will only generate the output in the form of an XML file.

2. Invoking the job from the Job Manager for Automatic Disbursement transactions (AD) – Offline Mode

   When the job is invoked through the Job Manager, the job generates the output based on the setting of the External Disbursement flag on the SOPT table. When the External Disbursement flag on SOPT is *True,* the job will not print or email anything. Instead, it generates the output in the form of an xml file that can be used by the external entities to print the check or warrant. When the External Disbursement flag is *False* on SOPT, this job produces the following output based on the parameters entered.

   - Checks – for the AD transactions with Disbursement type set to Checks
   - Warrants – for the AD transactions with Disbursement type set to Warrants
   - Remittance Advices – for the EFT transactions. The remittance advice will be printed for those vendors who requested a remittance advice (Remittance Advice Required flag is *True* on the VCUST table) and Remittance Transmission Mode is “Postal” on the Disbursement Format (DISF) table.
   - Remittance advice will be sent as an email when Remittance Transmission Mode on Disbursement Format (DISF) table is set as either “Email - As an Attachment” or “Email - Embedded HTML”.

If no Payee detail is available on the AD_DOC_HDR table, and the Vendor on AD_DOC_VEND is Miscellaneous on VCUST, and the email address is not available on the Vendor Line of the EFT transaction (AD_DOC_VEND), then the system will not process Remittance Advice as an Email notification.

If the Payee detail is available on the AD_DOC_HDR table, the system will not process Remittance Advice as an Email notification.

In both the AD/MD Printing, the concatenation of address on a PDF printed by the Disbursement process depends on the APPCTRL parameter:

- If CONCAT_CHK_PRNT_ADDR is Y (Yes) - The address on the PDF printed by the Disbursement Printing process concatenates all fields in a single line and each field is separated by the tilde (~) character.
- If N (No), no concatenation occurs and separate lines are created.

In both the AD/MD Printing, the check stub consolidation for Line Amounts depends on the APPCTRL parameter:

- If the External Disbursement Flag on SOPT is False, then the CONSOLIDATE_CHECK_STUB parameter on the Application Parameters (APPCTRL) page is looked up to determine check stub consolidation.

If CONSOLIDATE_CHECK_STUB is True - The accounting lines are consolidated based on Reference Transaction ID, Vendor Invoice Number and Vendor Invoice Line Number.

If CONSOLIDATE_CHECK_STUB is False – The accounting lines are NOT consolidated based on Reference Transaction ID, Vendor Invoice Number and Vendor Invoice Line Number.

On consolidation of lines, the sum of the Line Amounts is taken for the same group and the Check Description is taken from the first line of the group that has Check Description entered.

- In both the AD/MD Printing, presence of optional sections in data file produced by the Disbursement Print Process depends on APPCTRL parameters:

If OPTL_SECTS_IN_DISB_PRNT_XML on APPCTRL has Parameter Value as FOOTER, then the FOOTER section is included in the data file.

If OPTL_SECTS_IN_DISB_PRNT_XML on APPCTRL has Parameter Value as BATCH_PAGES, then the BATCH_PAGES section is included in the data file.

If OPTL_SECTS_IN_DISB_PRNT_XML on APPCTRL has Parameter Value as FOOTER, BATCH_PAGES, then FOOTER and BATCH_PAGES sections are included in the data file.

- If *Blank*, optional sections are removed from data file.

The Forms generated by the process can be directed to a printer or can be generated as a PDF.

The View Form parameter is used as an indication to generate printer forms or PDF forms. If the parameter is *True* (assuming the Print Resource corresponds to the PDF type) then the form is generated as a PDF; otherwise, the form is directed to the printer. If the View Form parameter is *True,* then the 'Print Job Code' and 'Print Resource ID' parameter value should correspond to the *PDF* type. However, if the View Form parameter is *False* then the 'Print Job Code' and 'Print Resource ID' parameter value should correspond to the *Print* type.

The following table shows the various steps and messages issued at each step.

Process Steps	Messages
1. Parameter	The following messages will be displayed in both online and offline

Process Steps	Messages
Validation	mode. • Validating batch parameters. • Parameters are valid or invalid depending on the validation. If the parameter is invalid, then the message for invalid parameter will be displayed in the log. It will be followed by message “Batch parameter validation failed”. • Batch parameter validation completed.
2. Selection and Evaluation of Records	All of the following messages are displayed only in offline mode. The following set of messages appears when the Disbursement Type parameter is Check. • Start processing Check records • If the selection returns zero records, then this step will be completed and the following message will be issued: “No eligible Check records found”. • Number of records processed: <Count> • Total Check records processed: <Total Count> • Performing Print Status Indicator updates on Source table • If no record updated for print status then the message issued will be “No eligible records updated for Print Status Indicator” • If one or more records updated then the message displayed will be: “Total number of records updated for Print Status Indicator: <Total Count>” • Step 3 Messages The following set of messages appears when the Disbursement Type parameter is Warrant. • Start processing Warrant records • If the selection returns zero records, then this step will be completed and the following message will be issued: “No eligible Warrant records found”. • Number of records processed: <Count> • Total Warrant records processed: <Total Count> • Performing Print Status Indicator updates on Source table • If no record is updated for print status then message issued will be “No eligible records updated for Print Status Indicator” • If one or more records are updated then the message displayed will be: “Total number of records updated for Print Status Indicator: <Total Count>”

Process Steps	Messages
	• Step 3 Messages The following set of messages appears when the Disbursement Type parameter is RA. • Start processing RA records • If the selection returns zero records, then this step will be completed and the following message will be issued: "No eligible RA records found". • Number of RA (Email as an Attachment) records processed: <Count> • Number of RA (Email as Embedded HTML) records processed: <Count> • Total RA (Email as an Attachment) records processed: <Total Count> • Total RA (Email as Embedded HTML) records processed: <Total Count> • Performing Print Status Indicator updates on Source table • If no record is updated for print status then the message issued will be "No eligible records updated for Print Status Indicator" • If one or more records are updated then the message displayed will be: "Total number of records updated for Print Status Indicator : <Total Count>" • Step 3 Messages The following set of messages appears when the Disbursement Type parameter is EFT or Remittance Advice. • Start processing EFT records • If the selection returns zero records, then this step will be completed and the following message will be issued: "No eligible EFT (Remittance Advice) records found". • Number of EFT records processed: <Count> • Number of RA (Email as an Attachment) records processed: <Count> • Number of RA (Email as Embedded HTML) records processed: <Count> • Total EFT records processed: <Total Count> • Total RA (Email as an Attachment) records processed: <Total Count> • Total RA (Email as Embedded HTML) records processed: <Total Count>

Process Steps	Messages
	• Performing Print Status Indicator updates on Source table • If no record is updated for print status then the message issued will be “No eligible records updated for Print Status Indicator” • If one or more records are updated then the message displayed will be: “Total number of records updated for Print Status Indicator: <Total Count>” • Step 3 Messages The following set of messages appears when the Disbursement Type parameter is blank and the External Disbursement Flag is *True*. • Start processing Check/Warrant/RA records • If the selection returns zero records, then this step will be completed and the following message will be issued: “No eligible Check/Warrant/RA records found”. • Number of records processed: <Count> • Total Check/Warrant/RA records processed: <Total Count> • Performing Print Status Indicator updates on Source table • If no record is updated for print status then the message issued will be “No eligible records updated for Print Status Indicator” • If one or more records are updated then the message displayed will be: “Total number of records updated for Print Status Indicator: <Total Count>” • Start processing EFT records • If the selection returns zero records, then this step will be completed and the following message will be issued: “No eligible EFT records found”. • Number of records processed: <Count> • Total EFT records processed: <Total Count> • Performing Print Status Indicator updates on Source table • If no record is updated for print status then the message issued will be “No eligible records updated for Print Status Indicator” • If one or more records are updated then the message displayed will be: “Total number of records updated for Print Status Indicator: <Total Count>” • Step 3 Messages. The following set of messages appears when the Disbursement Type parameter is blank and the External Disbursement Flag is

Process Steps	Messages
	False. • Start processing Check/Warrant/RA records • If the selection returns zero records, then this step will be completed and the following message will be issued: "No eligible Check/Warrant/RA records found". • Number of records processed: <Count> • Total Check/Warrant/RA records processed: <Total Count> • Performing Print Status Indicator updates on Source table • If no record is updated for print status then the message issued will be "No eligible records updated for Print Status Indicator" • If one or more records are updated then the message displayed will be: "Total number of records updated for Print Status Indicator : <Total Count>" • Step 3 Messages for disbursement type "Check/Warrant/RA" • Start processing EFT records • If the selection returns zero records, then this step will be completed and the following message will be issued: "No eligible EFT records found". • Number of records processed: <Count> • Total EFT records processed: <Total Count> • Performing Print Status Indicator updates on Source table • If no record is updated for print status then the message issued will be "No eligible records updated for Print Status Indicator" • If one or more records are updated then the message displayed will be: "Total number of records updated for Print Status Indicator: <Total Count>" • Step 3 Messages for disbursement type "EFT" **Note:** 1. <Count> will be in multiple of value provided for parameter PROG_SZ_CTR. 2. <Total Count> will be the total number of records processed.
3. XML Creation and Printing	The following messages will be displayed when the External Disbursement Flag is *True* on SOPT. These messages are only for offline mode and appear only once. • Creating XML file for processed records • Creation of XML file completed

Process Steps	Messages
	The following messages will be displayed when the External Disbursement Flag is *False* on SOPT. These messages appear only in offline mode. If the Disbursement Type parameter is Check. i) Creating XML file and Printing Check records ii) Creation of XML file and Printing of Check records completed If the Disbursement Type parameter is Warrant. i) Creating XML file and Printing Warrant records ii) Creation of XML file and Printing of Warrant records completed If the Disbursement Type parameter is RA. i) Creating XML file and Printing RA records ii) Creation of XML file and Printing of RA records completed iii) Creating XML file and sending Remittance Advice as 'Email as an Attachment' iv) Creation of XML and sending Remittance Advice as 'Email as an Attachment' completed. v) Creating XML file and sending Remittance Advice as 'Email as Embedded HTML'. vi) Creation of XML file and sending Remittance Advice as 'Email as Embedded HTML' completed If the Disbursement Type parameter is EFT. i) Creating XML file and Printing EFT records ii) Creation of XML file and Printing of EFT records completed iii) Creating XML file and sending Remittance Advice as 'Email as an Attachment'. iv) Creation of XML and sending Remittance Advice as 'Email as an Attachment' completed. v) Creating XML file and sending Remittance Advice as 'Email as Embedded HTML'. vi) Creation of XML file and sending Remittance Advice as 'Email as Embedded HTML' completed. If the Disbursement Type parameter is blank then the following messages appear after the processing of Check/Warrant/RA (See Step 2 for same conditions). i) Creating XML file and Printing Check/Warrant/RA

Process Steps	Messages
	records ii) Creation of XML file and Printing of Check/Warrant/RA records completed After displaying the above two messages, it will process EFT records. On completion of EFT record processing (See Step 2 for same conditions) it prints the following messages: i) Creating XML file and Printing EFT records ii) Creation of XML file and Printing of EFT records completed The following message will be displayed upon the successful completion of the process. This message appears in both online and offline mode. i) Printing process completed successfully **Note:** See description for <Disbursement Type> in step 2.

Restartability Information

This job cannot be restarted. If the job failed due to any reason, a new job should be scheduled after correcting the errors that caused the job to fail.

Major Input

Tables

The following is a list of tables that are used as input for Disbursement Printing:

- AD Transaction Header (AD_DOC_HDR)
- AD Transaction Vendor (AD_DOC_VEND)
- AD Transaction Accounting (AD_DOC_ACTG)
- MD Transaction Header (MD_DOC_HDR)
- MD Transaction Vendor (MD_DOC_VEND)
- MD Transaction Accounting (MD_DOC_ACTG)
- Disbursement Category (R_AP_DISB_CAT)
- Disbursement Format (R_AP_DISB_FRMT)
- Vendor Customer (R_VEND_CUST)
- Bank (R_AP_BANK_ACCT)
- Contact (R_CNTAC)
- Department (R_DEPT)
- Unit (R_UNIT)
- Country (R_CTRY)

- Calendar Date (R_CLDT)
- Special Instructions (R_SPIS)
- System Options (R_EXP_SOPT)
- Application Control (IN_APP_CTRL) – There are a fair number of these documented in a section called “Disbursement Controls” under Application Parameter in the *Disbursement User Guide*.
- Hard Copy Disbursement Options (R_HARD_CPY_DISB_OPT)

Batch Parameters

Offline printing process (AD Transactions)

Parameter	Description	Default Value
Export Location (AMSEXPORT)	A required location to write output.	No Default
Timestamp in Output (APPEND_TIMESTAMP_TO_OUTPUT)	A required output parameter that when Y (yes) appends a timestamp to output file name. The stamp is yyyymmddhhmmss.	N
Append Disbursement Category and Bank Code to output file (APPEND_CTG_AND_BANK)	An optional output parameter that when Y (Yes) appends Disbursement Category and Bank Code from the parameter value to the output file name (that is, File name=Job ID + the current date in YYYYMMDD format + Disbursement category + bank account code). This should be done to further differentiate multiple files created.	N
Exclude Attributes File Name (EXCLUDE_ATTRS_FILE_NM)	An optional output parameter that can be used to provide the name of the config file (.txt) with a list of attributes to be excluded from the XML file from STUB_HEADER and STUB_LINE sections. This feature is used when data in the file needs to be reduced to only what is needed for external printing. Valid values are: STUB_HEADER:PAGE_NO,PAYER_USER_NM_1,PAYER_USER_NM_2;STUB_LINE:DOC_VEND_LN_NO,DOC_ACTG_LN_NO,PYMT_RQST_VEND_LN_NO,PYMT_RQST_COMM_LN_NO,PYMT_RQST_ACTG_LN_NO,DOC_CD,DOC_DEPT_CD,DOC_ID,DOC_COMM_LN_NO,TRKG_DT,SVC_FRM_DT,SVC_TO,ACTG_LN_DSCR,CNTAC_	No Default

Parameter	Description	Default Value
	NM,CNTAC_LOC,CNTAC_PH_EXT,HDLG_CD,SORT_1_OBJ,SORT_2_OBJ,SPC_INST,REPL_FL,ADV_TYP,DYNA_BNK,VEND_CD,TIN,TIN_TYP,VEND_LGL_NM,PAGE_NO,RFED_DOC_CD,RFED_DOC_DEPT_CD,RFED_DOC_ID,VEND_INV_LN_NO,VEND_INV_DT; Format : Only valid values are accepted when uploaded in the above format in a text file (.txt)	
Application Printing Resource (APPL_RSRC_ID)	A required output printing source from Application Resources (SCRSRC).	DISB_PRN
Bank (BANK_ACCT_CD)	A conditionally required selection parameter required when the disbursement category, handling code, or disbursement format is also entered and the System Option of External Disbursement is false. More than one can be entered if comma-separated.	No Default
CC Email (CC_EMAIL_ADDRESS)	An optional output parameter of a CC email address or addresses copied in the email notification. More than one can be entered if comma-separated.	No Default
Client Name (CLIENT_NM)	Optional name to appear on output reports and written to the Header field of the HTML embedded in the email of the Remittance Advice.	No Default
Commit Block Size (COMMIT_SIZE)	A required performance parameter to control the number of records committed to output in a single instance. If left blank, 100 will default.	100
Payer Department (DEPT_CD)	A required output parameter of a department code used as the payer department on generated output.	No Default
Payer Unit (UNIT_CD)	A required output parameter of a unit code used as the payer unit on generated output.	No Default
Payer Contact (CNTAC_CD)	A required output parameter for a payer contact code for the payer return address printed on the check.	No Default
Disbursement Category	An optional selection parameter of one	No Default

Parameter	Description	Default Value
(DISB_CAT)	or more disbursement categories. More than one can be entered if comma-separated.	
Disbursement Format (DISB_FRMT)	An optional selection parameter of one or more disbursement formats. More than one can be entered if comma-separated.	No Default
Handling Code (DISB_HAND_CD)	An optional selection parameter of one or more handling codes. More than one can be entered if comma-separated.	No Default
Disbursement Type (DISB_TYP_PARM)	An optional output parameter to control what output is created. If left blank, then all eligible records will be processed. Valid values include: 1 - Check, 2 - Warrant, 3 - Remittance Advice, and 4 - EFTs. When SOPT is *Disbursements & Claims*: • If the parameter value is other then *check* (*1*), then the job fails with error messages in the logs. • If the parameter value is ALL, then the job only processes check type records.	No Default
Federal Debt Type (FED_DEBT_TYP)	An optional output parameter of a debt type assigned to Federal debt. If specified, Federal Entity is required and the two must be defined on the Intercept External Allocation page.	No Default
Federal Debts Entity (FED_ENTY)	An optional output parameter of an entity assigned to Federal debt. If specified, Federal Debt Type is required and the two must be defined on the Intercept External Allocation page.	No Default
Form Needs Empty Sub Lines (FORM_DSGN_NEED_EMPTY_STUBLN)	A required output parameter that when Y ensures a form design that needs empty unused stub lines for alignment has that XML data. When set to N, blank stub lines are not written and the size of the file is smaller.	Y
From Run ID (FRM_RUN_ID)	A conditionally required selection parameter used in conjunction with the To Run ID parameter to identify a	No Default

Parameter	Description	Default Value
	specific group of checks/warrants that were created during the same batch run (AD Chain).	
To Run ID (TO_RUN_ID)	A conditionally required selection parameter used in conjunction with the From Run ID parameter to identify a specific group of checks/warrants that were created during the same batch run (AD Chain).	No Default
Include EFT Number in Subject (INCL_EFT_NO_IN_EMAIL_SUBJ)	An output parameter that indicates whether or not to include the EFT Number in the email subject when set to: Y.	N
Jet Form Server Location JETFORM_SERVER_LOCATION	The required location of output folder for the print server.	C:\CGIAPPS\AdvFormsModule\ProcessingCenter\Main\output
Print Job (PRINT_JOB_CD)	The required processing parameter of the print job.	DISB_PRN_FORM
Print Resource (PRINT_RSRC)	When using External Disbursements, this required parameter specifies the printer.	No Default
Progress Counter (PROG_SZ_CTR)	A required performance parameter that specifies the number of records processed before writing a progression message to the job log. If left blank, 100 defaults.	100
Select Block Size (SELECT_BLOCK)	A required performance parameter that controls the number of records selected for processing in a single instance. If left blank, 5000 defaults.	5000
User Name 1 (USER_NM_1)	An optional output parameter of a name printed as the Payee Name when using plain check stock.	No Default
User Name 2 (USER_NM_2)	An optional output parameter used when User Name 1 is not enough to get the full Payee Name.	No Default
Use AL Number (USE_DOC_ACTG_NO)	A required output parameter that when Y results in the DOC_ACTG_NO field being written to xml for printing. The value will only be included in the file if	N

Parameter	Description	Default Value
	all fields in the Stub Detail Summarization of DISF are set to True. If any values in the Stub Detail Summarization section of DISF are False, the DOC_ACTG_NO value will be blank. N to exclude DOC_ACTG_NO from the xml files.	
Use Line Description in Email Body (USE_LN_DSCR_IN_EMAIL_BODY)	A required output parameter that when Y will result in the Accounting Line Description instead of the Check Description being written in the email body.	N
Use Stub Detail (USE_STUB_DET)	A required indication to control what appears on check stubs. False produces the check stub details in check number order. False will default if this parameter is left blank. When set to *True*, the data on the Disbursement Stub Detail (STUBDET) page will be used, which is built from setup on the Disbursement Format (DISF) page. Note: The parameter value of USE_STUB_DETAIL on APPCTRL should always match the parameter value of USE_STUB_DETAIL on this job. The job will issue an exception if there is a mismatch.	False
View Forms (VIEW_FORMS)	A required output parameter that when set to Y sends output to PDF to print separately. Note a PDF print resource must be used. N sends forms output directly to print resource destination.	N
Email Application Resource (EMAIL_APPL_RSRC_ID)	A required resource for sending a remittance advice email from Application Resources (SCRSRC) when a Remittance Advice Transmission Mode is 'Email - As an Attachment'.	DISB_PRN
Email Print Job EMAIL_PRINT_JOB_CD	The required processing parameter of the print job that will email Remittance Advice when a Remittance Advice Transmission Mode is 'Email - As an Attachment'.	DISB_PRN_FORM_EMAIL
Email Print Resource	Print Resource Identifier used to email Remittance Advice when a Remittance	Email

Parameter	Description	Default Value
(EMAIL_PRINT_RSRC)	Advice Transmission Mode is 'Email - As an Attachment'.	
VW_EMAILED_RA_APPL_RSRC_ID	Application Resource Identifier used to produce a consolidated PDF of all Remittance Advices emailed when Remittance Advice Transmission Mode was 'Email - As an Attachment' or 'Email - Embedded HTML'.	
VW_EMAILED_RA_PRINT_JOB_CD	Print Job used to produce a consolidated PDF of all Remittance Advices emailed when Remittance Advice Transmission Mode was 'Email - As an Attachment' or 'Email - Embedded HTML'.	
VW_EMAILED_RA_PRINT_RSRC	Print Resource Identifier used to produce a consolidated PDF of all Remittance Advices emailed when Remittance Advice Transmission Mode was 'Email - As an Attachment' or 'Email - Embedded HTML'.	
Send Remittance Advice to Miscellaneous Vendors (SEND_REMT_ADV_MISC_VEND)	A required output parameter that when Y sends a remittance advice notification to miscellaneous vendors when an electronic payment is made.	N
Remittance Advice Format for Miscellaneous Vendors (REMT_ADV_FRMT_MISC_VEND)	A conditionally required format used to determine the default remittance advice format when a remittance advice is sent to a miscellaneous vendor with an electronic payment.	
Check Number Length On Check (CHECK_NUM_ON_CHECK_DATA)	A required parameter used to determine check number length to be printed on check data. Valid values are between 10 to 15.	15
Accounting Strip Fields (ACCOUNTING_STRIP_FLDS)	An optional output parameter that can be used to bring in Accounting Line fields. Valid values include: UNIT_CD, DEPT_CD, FUND_CD, RSRC_CD, FUNC_CD, ACTV_CD, SFUND_CD, SRSRC_CD, SUNIT_CD, SACTV_CD, SFUNC_CD, RPT_CD, SRPT_CD, MJR_PROG_CD, PPC_CD, PROG_CD, PHASE_CD, FPRFL_CD, FPRTY_CD, and so forth.	No Default

Parameter	Description	Default Value
Overflow Text Message (OFLW_TXT_MSG)	An optional output parameter to write overflow text message when MAX_STUB_LINES on APPCTRL has been exceeded. When this message is written, no stub line details are written to the check stub. Although the batch parameter field allows for 250 characters, only 79 characters will be printed to the check stub.	No Default
Convert Legal Name to uppercase (DISB_LGL_NM)	An optional output parameter that when set to True, converts the Legal Name to uppercase.	False
Write additional Alias and Legal Names and DOC_ID to the XML file (WRITE_ADDL_NAMESTO_XML)	An optional output parameter that when set to True, writes Alias Name Legal Name and DOC_ID to the XML file. The additional names allow you to customize your check stub forms.	False

Online printing process (Manual Checks – MD Transaction)

Custom Parameters

Parameter	Description	Default Value
PRINT_JOB_CD	Print Job Name	No Default
PRINT_RSRC	Print Resource The printer that the specified files should print to.	No Default
VIEW_FORMS	View Forms Optional field. View Forms: Y to send forms output to job status inquiries (must use a PDF print resource). N to send forms output directly to print resource destination.	No Default
VW_FORMS_DSCR	View Forms Description	No Default
DEPT_CD	Payer Department Code	No Default
UNIT_CD	Payer Unit Code	No Default

Parameter	Description	Default Value
CNTAC_CD	Payer Contact Code Must be valid on the Contact (R_CNTAC) table. This is the payer return address that will be printed on the check.	No Default
Output Directory	Export Location at Disbursement Printing Job Optional Field. If not entered then the value will be picked up from the upper level of the batch process hierarchy.	No Default
USER_NM_1	User Name 1 Will print the Payee Name on the check itself. Value should be entered when using plain check stock.	No Default
USER_NM_2	User Name 2 A continuation of the field above – should be used if name covers more than one line.	No Default
MSG_TXT	Message Text. This message gets displayed on checks or warrants.	No Default
Skip Mailing Address	Required. Valid values are: Yes or No. When set to Yes, the Mailing address will not be printed on the back of the check. When set to No, the mailing address will be printed on the back of the check.	No Value
Reprint	Required. Valid values are: Yes or No. Value Yes can be selected only when the check has already been printed.	No Value
Check Number Length On Check (CHECK_NUM_ON_CHECK_DATA)	A required parameter used to determine the check number length to be printed on check data. Valid values are between 10 to 15.	15

Parameter	Description	Default Value
Accounting Strip Fields (ACCOUNTING_STRIP_FLDS)	An optional output parameter that can be used to bring in Accounting Line fields. Valid values include: UNIT_CD, DEPT_CD, FUND_CD, RSRC_CD, FUNC_CD, ACTV_CD, SFUND_CD, SRSRC_CD, SUNIT_CD, SACTV_CD, SFUNC_CD, RPT_CD, SRPT_CD, MJR_PROG_CD, PPC_CD, PROG_CD, PHASE_CD, FPRFL_CD, FPRTY_CD, and so forth.	No Default

Major Output

- The following XML files are created under the directory specified by the AMSEXPORT parameter:

OfflineDisbursementPrinting.xml - when the Disbursement printing job is executed.

OnlineDisbursementPrinting.xml – when the Online printing is executed for MD transactions.

RAEmailAsAnAttachment.xml – One consolidated file for all RA records whose Remittance Transmission Mode is equal to Email as an Attachment.

RAEmailAsEmbeddedHTML.xml - One consolidated file for all RA records whose Remittance Transmission Mode is equal to Email - Embedded HTML.

RAEmailAsAnAttachment_<Count>.xml – One file for every RA record where Remittance Transmission Mode is equal to Email as an Attachment.

RAEmailAsEmbeddedHTML_<Count>.xml - One file for every RA record where Remittance Transmission Mode is equal to Email - Embedded HTML.

- The following instruments will be printed based on DISB_TYP_PARM:

Check – for the records with the Disbursement Type of Check.

Warrant - for the records with the Disbursement Type of Warrants.

Remittance Advice - for the records with a Check Amount of $0.00 and when the Check Number is empty.

EFT - for the records with the Disbursement Type of EFT and the Remittance Advice flag set to *True*.

- The following instruments will be emailed when DISB_TYP_PARM is either EFT or RA or blank:
    - Remittance Advice will be sent as a pdf attachment – for the records with Remittance Advice Transmission Mode as “Email as an Attachment”.
    - Remittance Advice will be sent as Email with embedded HTML - for the records with Remittance Advice Transmission Mode as ”Email - Embedded HTML”.
    - The following information will be obtained from Disbursement Format table:
        - Sender’s Email
        - Email Subject
        - Email Message: Text from this field will be displayed in the email as it is typed with two exact replaceable parameters %CHK_NO% and %CHK_EFT_ISS_DT%:

- Disbursement Printing replaces the occurrence of %CHK_NO% with Check/EFT Number (from EFT Header) and
- %CHK_EFT_ISS_DT% with Check/EFT Date (from EFT Header)
- Example: 'Please find attached the Remittance Advice for EFT Number %CHK_NO% dated %CHK_EFT_ISS_DT%' will be displayed on email message as 'Please find attached the Remittance Advice for EFT Number 201101010000023 dated 01-01-2011'.

- Table Updates

Update the Print Status Indicator on the AD_DOC_HDR.

Update the Print Status Indicator on the MD_DOC_HDR.

Updates Hard Copy Disbursement Count (R_HARD_CPY_DISB_CNT) when that feature is used with the number of hard copy payments made for a Fiscal Year, Department and Taxpayer ID.

Job Return Code

The following table shows the potential job return codes for the Disbursement Printing job.

Return Code	Condition
Successful (1)	All of the selected AD or MD records are processed successfully.
Warning (4)	No eligible records found. This could be because of the following reasons: • No eligible record found for specified disbursement type parameter. • No eligible record found for all of the disbursement types when no disbursement type is specified as a batch parameter.
Non-Fatal Error (8)	This job does not return a Non-Fatal Error in any condition.
Failed (12)	The job will fail under the following conditions: • Parameters are invalid. • Run time exceptions for unexpected situations.
Terminated (16)	This return code will be issued when the job is terminated by the user.
System Failure (20)	This return code will be issued when the job is terminated because of database server or network issues.

Sort Criteria

Offline mode retrieves data by a join on the AD_DOC_HDR, AD_DOC_VEND and AD_DOC_ACTG tables. It retrieves data in ascending order based on the following fields:

- Transaction Code (AD_DOC_HDR.DOC_CD)

- Transaction Department Code (AD_DOC_HDR.DOC_DEPT_CD)
- Transaction ID (AD_DOC_HDR.DOC_ID)
- Transaction Version Number (AD_DOC_HDR.DOC_VERS_NO)
- Transaction Vendor Number (AD_DOC_VEND.DOC_VEND_NO)
- Transaction Accounting Number (AD_DOC_ACTG.DOC_ACTG_NO)

Online mode retrieves data for a single MD Transaction. It selects related data from MD_DOC_HDR, MD_DOC_VEND and MD_DOC_ACTG by firing a separate query on the database for each table. Only the following field is used for the sorting of data from MD_DOC_ACTG.

- Transaction Accounting Number (MD_DOC_ACTG.DOC_ACTG_NO)

Selection Criteria

Offline Mode

The following conditions will be used by the offline mode as selection criteria. These criteria can be different based on input batch parameters. It retrieves data for the AD transaction with the use of a join on AD_DOC_HDR, AD_DOC_VEND and AD_DOC_ACTG.

If the Vendor Invoice Consolidation flag is *True* on APPCTRL the MD_DOC_ACTG table will have records grouped:

For a given AD transaction the Accounting lines will be grouped on three attributes

Reference transaction ID

Vendor invoice no

Vendor invoice line no

The line amounts will be the sum of those grouped lines and the check description and other details will be taken from the first line of the group which has the non blank check description.

Common conditions to retrieve any eligible record for Disbursement Printing:

- Transaction Function Code must not be “Cancellation”.
  - AD_DOC_HDR.DOC_FUNC_CD <> 3
- Transaction Phase should be “Final”
  - AD_DOC_HDR.DOC_PHASE_CD = 3
- Print Status Indicator should be “Ready for Original Print” or “Ready for Re Print”
  - AD_DOC_HDR.PRN_STA_IND = 2 OR AD_DOC_HDR.PRN_STA_IND = 3

It adds the following conditions based on the value of DISB_TYP_PARM parameter:

Batch parameter DISB_TYP_PARM = 1 (Check)

- Disbursement Type of AD transaction should be “Check”
  - AD_DOC_HDR.DISB_TYP = 1

Batch parameter DISB_TYP_PARM = 2 (Warrant)

- Disbursement Type of AD transaction should be “Warrant”
  - AD_DOC_HDR.DISB_TYP = 2

Batch parameter DISB_TYP_PARM = 3 (Remittance Advice)

- Check Amount should be zero and Check Number should be empty
  - AD_DOC_HDR.CHK_AM = 0 AND AD_DOC_HDR.CHK_NO IS NULL

Batch parameter DISB_TYP_PARM = 4 (EFT)

- Disbursement Type of AD transaction should be “EFT”
  - AD_DOC_HDR.DISB_TYP = 4
- If the External Disbursement Flag on SOPT is *False* then:
  - If Non-Miscellaneous Vendor then
  - the Remittance Advice Required Flag should to be *True*
  - AD_DOC_ACTG.REM_ADV_RQD_FL = 1
  - If Miscellaneous Vendor then:

If Remittance Advice Required for Miscellaneous Vendor parameter is ‘Y’ (Yes) on the job parameter, and

Remittance Advice Format on the job parameter has the Remittance Advice Transmission Mode set to "Email - As an Attachment" or Email - As an Embedded HTML on DISF table, and

No Payee details on the AD Header (AD_DOC_HDR)

Email ID is not null on the Vendor Line of the EFT Transaction (AD_DOC_VEND).

<u>If the value for the batch parameter DISB_TYP_PARM is not entered:</u>

If DISB_TYP_PARM is not entered, it should process records for all of the Disbursement Types. It fires two queries to retrieve AD Transactions. For both queries, it uses common conditions specified above. These queries use the following conditions when the DISB_TYP_PARM is empty.

- First Retrieval
  - Disbursement Type of AD transaction should be “Check” or “Warrant”. As this is not filtering any record for Check Amount not to be zero, it will also pick up records for “Remittance Advice”

AD_DOC_HDR.DISB_TYP = 1 OR AD_DOC_HDR.DISB_TYP = 2

- Second Retrieval
  - Disbursement Type of AD transaction should be “EFT”

AD_DOC_HDR.DISB_TYP = 4

- If the External Disbursement Flag on SOPT is *False* then

If Non-Miscellaneous Vendor then

- the Remittance Advice Required Flag should to be *True*
    - AD_DOC_ACTG.REM_ADV_RQD_FL = 1

If Miscellaneous Vendor then

- If Remittance Advice Required for Miscellaneous Vendor parameter is 'Y' (Yes) on the job parameter, and
- Remittance Advice Format on the job parameter is having Remittance Advice Transmission Mode as "Email - As an Attachment"/ Email - As an Embedded HTML on DISF table, and
- No Payee details on the AD Header (AD_DOC_HDR)
- Email ID is not null on the Vendor Line of the EFT Transaction (AD_DOC_VEND).

If the value for the batch parameter FED_ENTY and FED_DEBT_TYP are entered:

If one parameter is specified, then the other parameter should also be specified. If both parameters are specified, then the combination of "Federal Entity" (FED_ENTY) and "Federal Debt Type" (FED_DEBT_TYP) should be defined on the INTEA (R_INT_EA) table.

If combination of "Federal Entity" (FED_ENTY) and "Federal Debt Type" (FED_DEBT_TYP) not define on the INTEA (R_INT_EA) table then following message will log

"The combination of the batch parameters FED_ENTY and FED_DEBT_TYP is not defined on the INTEA table."

If both of the "Federal Entity" (FED_ENTY) and "Federal Debt Type" (FED_DEBT_TYP) parameters are specified and the combination of these two parameters is valid on the INTEA table, then the Disbursement Printing batch process will compare the Entity and Debt Type of current INTA (AP_INCT_ACTV) record against the values specified on respective parameters before generating the STUB_LINE_INTERCEPT_INFORMATION record.

- If the combination of Entity (ENTY_CD) and Debt Type (DEBT_TYP_CD) of selected INTA (AP_INCT_ACTV) record matches the combination of "Federal Entity" (FED_ENTY) and "Federal Debt Type" (FED_DEBT_TYP) parameters, then the process will perform the following:

Compare the Receivable Transaction ID (RE_DOC_ID) of selected INTA (AP_INCT_ACTV) record against the Debt Sequence ID (FED_DEBT_SEQ_ID) of the FEDMTCH (R_FED_INCT_MTCH) table to locate respective debt record.

If the debt record is found on the FEDMTCH table, then the process will perform the following:

- Populate the Creditor Agency Name (CRDTR_AGCY_NM) and Creditor Agency Contact Number (CRDTR_AGCY_PH_NO) of selected FEDMTCH record to the Intercept Contact Name (REMT_CNTAC) and Intercept Contact Phone (PH_NO), respectively, on the STUB_LINE_INTERCEPT_INFORMATION record being generated. Do **not** populate the Creditor Agency Contact Number from the **CRDTR_AGCY_PH_NO_UFMT** of the FEDMTCH table.
- Set both of the Intercept Contact Phone Extension (PH_EXT) and Intercept Contact E-Mail Address (EMAIL_AD) to blank on the STUB_LINE_INTERCEPT_INFORMATION record being generated.

If the debt record is not found on the FEDMTCH table, then the process will perform the following:

- Populate the Remittance Contact Name (REMT_CNTAC), Remittance Contact Phone (PH_NO), Remittance Contact Phone Extension (PH_EXT) and Remittance Contact E-Mail Address (EMAIL_AD) of selected INTA (AP_INCT_ACTV) record into respective fields of the STUB_LINE_INTERCEPT_INFORMATION record being generated.

Note:

This condition could be met if a debt intercepted by an AD payment was archived from the FEDMTCH table.

- If the combination of Entity (ENTY_CD) and Debt Type (DEBT_TYP_CD) of selected INTA (AP_INCT_ACTV) record does not match the combination of “Federal Entity” (FED_ENTY) and “Federal Debt Type” (FED_DEBT_TYP) parameters, then the process will perform the following:

Populate the Remittance Contact Name (REMT_CNTAC), Remittance Contact Phone (PH_NO), Remittance Contact Phone Extension (PH_EXT) and Remittance Contact E-Mail Address (EMAIL_AD) of selected INTA (AP_INCT_ACTV) record into respective fields of the STUB_LINE_INTERCEPT_INFORMATION record being generated.

If both parameters “Federal Entity” (FED_ENTY) and “Federal Debt Type” (FED_DEBT_TYP) are not specified, then the process will perform the following:

Populate the Remittance Contact Name (REMT_CNTAC), Remittance Contact Phone (PH_NO), Remittance Contact Phone Extension (PH_EXT) and Remittance Contact E-Mail Address (EMAIL_AD) of selected INTA (AP_INCT_ACTV) record into respective fields of the STUB_LINE_INTERCEPT_INFORMATION record being generated.

Batch parameters FRM_RUN_ID and TO_RUN_ID are entered:

Run ID of AD transaction should be greater than or equal to FRM_RUN_ID. Here Run ID of AD transaction indicates the job Run ID for the AD Chain through which the particular AD transaction is generated.

AD_DOC_HDR.RUN_ID >= value of parameter FRM_RUN_ID

Run ID of AD transaction should be less than or equal to TO_RUN_ID.

AD_DOC_HDR.RUN_ID <= value of parameter TO_RUN_ID

Note: One or multiple values in the Disbursement Category, Handling Code or Bank Code fields can be entered de-limiting each by a comma and no spaces per job run. If values have been entered into one or more parameter fields, the fields will be evaluated as an "and" condition.

Online Mode

Online mode retrieves data for a specified single MD transaction.

It retrieves data from MD_DOC_HDR with the following conditions:

Transaction Code is equal to Transaction Code of selected MD transaction

MD_DOC_HDR.DOC_CD = Value of parameter DOC_CD

Transaction Department is equal to Transaction Department of selected MD transaction

MD_DOC_HDR.DOC_DEPT_CD = Value of parameter DOC_DEPT_CD

Transaction ID is equal to Transaction ID of selected MD transaction

MD_DOC_HDR.DOC_ID = Value of parameter DOC_ID

Transaction Version Number is equal to Transaction Version Number of selected MD transaction

MD_DOC_HDR.DOC_VERS_NO = Value of parameter DOC_VERS_NO

After retrieving data from MD_DOC_HDR, it retrieves a related record from MD_DOC_VEND based on the following conditions:

Transaction primary information of MD_DOC_VEND should be matched with MD_DOC_HDR record.

MD_DOC_VEND.DOC_CD = MD_DOC_HDR.DOC_CD

MD_DOC_VEND.DOC_DEPT_CD = MD_DOC_HDR.DOC_DEPT_CD

MD_DOC_VEND.DOC_ID = MD_DOC_HDR.DOC_ID

MD_DOC_VEND.DOC_VERS_NO = MD_DOC_HDR.DOC_VERS_NO

Data from MD_DOC_ACTG will be retrieved after getting data from MD_DOC_VEND:

Transaction primary information for MD_DOC_ACTG should be matched with MD_DOC_VEND

MD_DOC_ACTG.DOC_CD = MD_DOC_VEND.DOC_CD

MD_DOC_ACTG.DOC_DEPT_CD = MD_DOC_VEND.DOC_DEPT_CD

MD_DOC_ACTG.DOC_ID = MD_DOC_VEND.DOC_ID

MD_DOC_ACTG.DOC_VERS_NO = MD_DOC_VEND.DOC_VERS_NO

MD_DOC_ACTG.DOC_VEND_NO = MD_DOC_VEND.DOC_VEND_NO

If the Vendor Invoice Consolidation flag is *True* on APPCTRL, the AD_DOC_ACTG table will have records grouped:

For a given MD transaction, the Accounting lines will be grouped on three attributes

Reference Transaction ID

Vendor Invoice No

Vendor Invoice Line No

The Line Amounts will be sum of those grouped lines and the Check Description and other details will be taken from the first line of the group that has the non blank Check Description.

Problem Resolution

If the job fails for any reason, then a new job should only be scheduled. There is no need to back out any updates if this job fails in any of the steps since the Print Status Indicator on the corresponding transaction gets updated if all records are processed successfully.

Step 1: Parameter Validation:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the parameters are validated successfully.	N/A	N/A
Warning (4)	N/A	This step does not issue the warning code.	N/A
Non-Fatal Error (8)	N/A	This job does not return the Non-Fatal Error	N/A
Failed (12)	Required parameter not specified **Sample Message:** Print Resource not specified	Enter Print Resource and schedule a new job.	.
	Entered parameters are invalid **Sample Message:** Invalid Payer Department Code	Enter the correct Department Code which is valid on the Department table and schedule a new job.	
	Entered parameter is zero or negative number **Sample Message:** Commit block size should be greater than zero	Enter a positive integer for commit block size and schedule a new job.	
	Entered parameter is not a valid integer number **Sample Message:** Commit block size should be a positive integer	Enter a valid positive integer for commit block size and schedule a new job.	
	It can fail in case of run time exception	The reason needs to be investigated for run time exception. Correct the problem and schedule a new job.	System error log and VLS log should be investigated to find out possible reason of exception.
	If the External Disbursement Flag on SOPT is *False* and if the Bank Account Code field is blank and if any value has been entered in the Disbursement Category, Handling Code or Disbursement Format parameters	Enter Bank Account Code and schedule a new job.	N/A
	If on SOPT the 'External	Set Check Stock Type	N/A

Possible Return Codes	Condition	Recommendation	Other Instructions
	Disbursement Flag' is *False* and the 'Print $0 Amount Check' is *False* and if any parameter value exists in the Disbursement Category, Handling Code or Disbursement Format field and Check Stock Type on Bank table is 'Preprinted	on the Bank table to 'Plain' and schedule a new job.	
	If Use Stub Detail parameter is *True* and either: • Disbursement Type parameter is anything other than Check or Warrant. • SOPT's External Disbursement flag is *Yes*. • Application Parameter (APPCTRL) record for CONSOLIDATE_CHECK_STUB is *True*.	Adjust the values, and schedule a new job.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated, and a new job should be scheduled.	
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated, and a new job should be scheduled.	

Step 2: Selection and Evaluation of Records:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	No eligible record found for disbursement printing **Sample Message:** No eligible Check records found	Verify the Disbursement Run ID and the Disbursement Type parameters entered on the parameter section. Verify whether they are the correct one. If not, enter appropriate parameters so	

Possible Return Codes	Condition	Recommendation	Other Instructions
		that the job selects the eligible records.	
Non-Fatal Error (8)	N/A	This job does not return the Non-Fatal Error code.	N/A
Failed (12)	In this step, the job can fail because of runtime exceptions for an unexpected situation.	The reason for the failure needs to be investigated. Schedule a new job after resolving the issues.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated before scheduling a new job.	.
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated before scheduling a new job.	

Step 3: XML Creation and Printing:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	N/A	This step does not issue the Warning code.	N/A
Non-Fatal Error (8)	N/A	This job does not return the Non-Fatal Error code.	N/A
Failed (12)	In this step, the job can fail only because of runtime exceptions for an unexpected situation.	The reason for the failure needs to be investigated. Schedule a new job after resolving the issues.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated before scheduling a new job.	
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated before scheduling a new job.	

2.1.16 Disbursement Request Initialization

Chain or Job Name	Disbursement Request Initialization
Recommended Frequency	Nightly cycle or on demand It is recommended to run this job before the Automated Disbursement process during the nightly cycle.
Single Instance Required	Yes
Can be restarted?	No
Reports generated	No

Overview

This Disbursement Request Initialization job has been created to:

1. Enforce consistency in Bank Accounts for payments from the same fund by re-inferring the Bank Account on the DISRQ table records from the Fund table. A **Bank Account Code** can be specified on the payment request Header level as well as the Accounting Line level. Either (or both) of these Bank Accounts may be different from the **Check/EFT Bank Account Code** associated with a referenced Fund code on the FUND table. The **Bank** Account **Code** from the Payment Request Accounting Line is stored on the Disbursement Request (DISRQ) table and used by the AD Chain to generate Check/EFT payments. Using the Disbursement Initialization batch job, your site may enforce that the Bank Accounts for the payments be consistent with the Bank Account from the Fund table. For all funds entered in the parameter, the batch process will re-infer the **Bank Account Code** from the FUND page based on the disbursement type, whether it equals *Check* or *EFT*. The process will update DISRQ records that match the Fund(s) listed to set the **Bank Account Code** on DISRQ to the **Bank Account Code** from the FUND page.

   The initialization job can be used only if the **Always Use Fund Bank Account Code for Disbursements** parameter on APPCTRL is set to *Yes*. If it is set to *No*, then the process will fail and generate an error that this parameter is only allowed if the parameter is equal to *Yes*. NOTE – This option should be used with care. Use of this option may result in negative payment request lines (credit memo lines) not being selected for payment.

2. Enforce consistency for the Single Payment flag for payment transaction codes entered as a batch process by setting the Single Payment flag to True on the DISRQ table records. Specifying disbursements as single payments (that is, bypassing consolidation) may be specified at the time of the Payment Request or by changing the **Single Payment** flag on the disbursement requests using one of the Disbursement Management pages or transaction (DISRQ, DISBMD, or DRM transaction).

   For all valid Transaction Codes entered in the parameter, the batch process will update DISRQ records to set the **Single Payment** flag to *Yes* except in two situations:

   - Procurement Card payments – This process will not reset any **Single Payment** flags for Procurement Card payments.
   - Payments where the total payment amount for a vendor line is < 0. This process will not set the **Single** Payment indicator for this situation to allow credit memos (negative lines) to be applied to other payments. Total of the payment request transaction Line Amounts on the Disbursement Request table for vendor line is not less than 0, where the Line

Amounts are summed based on the Transaction Code (DISRQ), Transaction ID (DISRQ), Transaction Department (DISRQ), and Vendor Code (DISRQ).

3. Update blank Consolidation Objects on DISRQ records based on the setup of the Consolidation Objects Configuration (COCNFG) page. The Populate Consolidation Objects batch parameter determines whether the Consolidation Objects on the DISRQ page need to be populated or not. The Re-Infer Consolidation Objects parameter determines if the batch job populates the Consolidation Object fields on the DISRQ page powerfully (meaning irrespective of whether the Consolidation Object fields are blank or not). This parameter is only used if the Populate Consolidation Objects parameter is set to *Y*. Together, these two parameters are used to determine how to update the Consolidation Object fields on the DISRQ page as follows:

   - If the Populate Consolidation Objects parameter is set to *Y* (Yes) and the Re-Infer Consolidation Objects parameter is set to *Y* (Yes), then the batch job uses the field values from the Consolidation Object fields on the COCNFG page to determine how to update the corresponding Consolidation Object fields on the DISRQ page. For example, if the Consolidation Object 1 field on the COCFNG page is set to *Payment Request Transaction Code*, the system updates the corresponding Consolidation Object 1 field on the DISRQ page with the value of the Transaction Code of the associated DISRQ record. This is a powerful inference, meaning that all values on the DISRQ page are replaced with what is specified on the COCNFG page, even if the Consolidation Object field on the DISRQ page is populated and the corresponding field on the COCNFG page is blank.
   - If the Populate Consolidation Objects parameter is set to *Y* and the Re-Infer Consolidation Objects parameter is set to *N*, then the job only updates those fields on the associated DISRQ record where the associated Consolidation Object fields are blank. It updates the Consolidation Object values on the DISRQ page as indicated by the value specified on the COCFNG page.
   - If the Populate Consolidation Objects parameter is *N* or if the Consolidation Object fields on the CONCFG page are blank (no records exist on the page), then the job writes an informational message on the job log, “No Consolidation Object Fields are updated on the DISRQ records.”

Logic for each function (Bank Account, Single Payment flag and Populate Consolidation Objects) is completely independent of the other; each depends on their own parameters.

This job can be run any time, but it is recommended to run the job before the Automated Disbursement (AD) Chain job.

Process Steps

The following table shows the various steps that the Disbursement Request Initialization Job goes through and the messages issued at each step:

Process Steps	Messages
1. Validate the Parameters.	• Parameter Validation failed • Parameter Validation was successful
2. Update the DISRQ for the Single Payment Flag and/or the Bank Account Code and/or Populate Consolidation Objects and/or Re-Infer Consolidation Objects.	

Restartability Information

If the job fails it can be restarted after resolving the error.

Major Input

Data from the following table:

- R_AP_DISB_RQST – Disbursement Request (DISRQ) table
- R_CNSD_OBJ_CNFG - Consolidation Objects Configuration (COCNFG) table

Batch Parameters

Parameter	Description	Default Value
DOC_CD	A single Transaction Code or Multiple Transaction Codes in a comma separated list.	No Default
FUND_CD	This will be a single Fund Code or Multiple Fund Codes in a comma separated list.	No Default
POP_CNSD_OBJ	This parameter is used to determine if the job should populate the Consolidation Objects on the DISRQ records.	N
REINFR_CNSD_OBJ	This parameter is used to determine if the job should re-infer the Consolidation Object values from the COCNFG table on the DISRQ records, if those values are already populated.	N

Major Output

- R_AP_DISB_RQST – Disbursement Request Table

Job Return Code

The following table shows the potential job return codes for the Inferences & Validations job:

Return Code	Condition
Successful (1)	This return code will be issued when the job processes all of the records successfully.
Warning (4)	This return code will be issued when no eligible records are found on the DISRQ table.
Non-Fatal Error (8)	N/A

Return Code	Condition
Failed (12)	The job will fail under the following conditions: • Parameters are invalid. • Run time exceptions for unexpected situations. • If there were less than zero records updated. • If the Populate Consolidation parameter value is not equal to 'Y' or 'N' or null (no value). • If the Populate Consolidation parameter value is equal to 'Y' and a record does not exist on the COCNFG table. • If the Re-Infer Consolidation Object parameter value is not equal to 'Y' or 'N' or is null (no value). • If the Re-Infer Consolidation Object parameter value is equal to 'Y' and the Populate Consolidation Object Parameter is N or not populated (null).
Terminated (16)	This return code will be issued when the job is terminated by the user.
System Failure (20)	This return code will be issued when the job is terminated because of database server or network issues.

Selection Criteria

The Disbursement Request Initialization batch process will perform logic based on the parameters entered. Logic for each function (Bank Account and Single Payment flag) is completely independent of the other; each depends on their own parameters. The following parameters are available to the batch job:

1. Replace Bank Account Fund List
    - Selection logic:

For all funds entered, the batch process will re-infer the Bank Account Code from the FUND page based on the Disbursement Type, whether equal to Check or EFT.

The process will update DISRQ records that match the Fund(s) listed to set the Bank Account Code on DISRQ to the Bank Account Code from the FUND page.

> NOTE – This option should be used with care. Use of this option may result in negative payment request lines (credit memo lines) not being selected for payment.

2. Set Single Payment Flag for Transaction Code List
    - Selection logic:
        - For all valid transaction codes entered, the batch process will update DISRQ records to set the Single Payment flag to ‘Yes’, where the following are true:
            - Transaction Code (DISRQ) matches one of the Transaction Code(s) listed, and

- Procurement Card Payment (DISRQ) is equal to 'No', and
- Total of the payment request transaction Line Amounts on the Disbursement Request table for vendor line is not less than 0, where the Line Amounts are summed based on the Transaction Code (DISRQ), Transaction ID (DISRQ), Transaction Department (DISRQ), and Vendor Code (DISRQ).

3. Populate Consolidation Objects and Re-Infer Consolidation Objects

- Selection logic:

If the Populate Consolidation Objects parameter is set to *Y* and the Re-Infer Consolidation Objects parameter is set to *Y*, then the batch process reads the Consolidation Objects field from the Consolidation Object Configuration table and updates all of the corresponding Consolidation Object fields on DISRQ records

> This is a powerful inference so all values on the DISRQ record are replaced with what is specified on the COCNFG table, even if the Consolidation Objects field on the DISRQ record are populated and the corresponding field on the COCNFG table is null. When the option on the COCNFG table is null, the batch process overwrites the value on the DISRQ record and blanks out the specified Consolidation Object value on the DISRQ record.

If the Populate Consolidation Objects parameter is set to *Y* and the Re-Infer Consolidation Objects parameter is set to *N*, then the job reads the Consolidation Object fields from the COCNFG table and the process will select only those DISRQ records where the associated Consolidation Object fields are blank and updates the Consolidation Object values with the value specified on the Consolidation Objects Configuration table.

If the Populate Consolidation Objects parameter is *N*' or if the CONCFG table is blank (no records exist on the table), then this logic bypassed.

Problem Resolution

The following table shows the possible return codes and recommendations for each processing step.

Step 1: Parameter Validation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the parameters are validated successfully. If all parameters are blank.	Populate parameters with valid values.	N/A
Warning (4)	If no records are selected from DISRQ for processing. **Sample Message:** No records selected for Infer	Make sure to enter correct parameters and run the job again.	N/A

Possible Return Codes	Condition	Recommendation	Other Instructions
	Consolidation Object values processing.		
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Job failed due to Fatal conditions.	In this step, the job can fail under the following conditions. • Encounters any runtime exceptions. • Failed during restart. • If the value supplied for Transaction Code is empty. • If the Transaction Codes supplied are not validated successfully. • If the Fund Code values are not validated successfully. • If the Populate Consolidation parameter value is not equal to 'Y' or 'N' or null (no value). • If the Populate Consolidation parameter value is equal to 'Y' and a record does not exist on the COCNFG table. • If the Re-Infer Consolidation Object parameter value is not equal to 'Y' or 'N' or is null (no value)	Schedule a new job.

Possible Return Codes	Condition	Recommendation	Other Instructions
		• If the Re-Infer Consolidation Object parameter value is equal to 'Y' and the Populate Consolidation Object Parameter is equal to N or not populated (null). **Sample Message:** • Valid values for the Populate Consolidation Object parameter are Y or N or blank. • A record must exist on the COCNFG table if the Populate Consolidation Object parameter is set to N or blank. • Valid values for the Re-Infer Consolidation Object parameter are Y or N or blank. • The Re-Infer Consolidation Object parameter cannot be set to Y if the Populate Consolidation Object parameter is set to N or blank. • If the job fails because of the runtime exceptions, investigate the exception	

Possible Return Codes	Condition	Recommendation	Other Instructions
		reported by the process, resolve the error and restart the job. • Parameter value must Y or N or null. • Provide setup on COCNFG table. • Set Populate Consolidation Object parameter to Y.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. A new job can be scheduled.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only a new job should be scheduled.
System Failure (20)	When the job is terminated because of database server or network issues	The reason for the System Failure needs to be investigated. A new job can be scheduled.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only a new job should be scheduled.

Step 2: Updating the records

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	N/A	N/A	N/A
Warning (4)	No eligible records found.	Make sure to enter correct parameters and run the job again.	N/A
Non-Fatal Error (8)	N/A	N/A	N/A

Failed (12)	Failed because of runtime exceptions for an unexpected situation.	The job can fail under these conditions: • If the number of records updated is less than Zero. • If committing of this transaction fails. Failure reason needs to be investigated before scheduling a new job.	Failed because of runtime exceptions for an unexpected situation.
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. A new job can be scheduled.	N/A
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. A new job can be scheduled.	N/A

2.1.17 EF ACH Reversal

Job Name	EF ACH Reversal
Recommended Frequency	This job can be executed more than one time per night. It is recommended to run this job before Advantage 3 Mass Cancellation and Check Reconciliation batch processes during the nightly cycle.
Single Instance Required	Yes
Can be restarted?	No
Reports Generated	Yes

Overview

The EF ACH Reversal batch process is a chain job which comprises the following three batch jobs:

- Build XML File
- Build Flat File
- Initiate Disbursement Cancellation

These jobs can be run as a **chain or independently.**

EF ACH Reversal: Build XML File

Job Name	Build XML File
Recommended Frequency	Daily as part of the nightly cycle or On Demand.
Single Instance Required	Yes
Can be restarted?	No
Reports Generated	• Requested EFT Reversals report • Exception Report

Overview

The Build XML File process creates the Reversal ACH file for the requested EFT payments.

Steps for running this process:

1. Parameter Validation: First the process will validate the batch parameters.

2. Selection of Records: Once the parameter validation is successful, the job will select the eligible records from the EFT Reversal Table.

   1. Batch parameter Bank codes (if specified)

      Select * from R_EFT_RVRSL

      Where BANK_ACCT_CD in (“BANK_ACCT_CD”)

      And ACT_FL = “True”

      And REVERS_STA = “Request Reversal”.

   2. Batch parameter Bank code is null

      Select * from R_EFT_RVRSL

      Where ACT_FL = “True”

      And REVERS_STA = “Request Reversal”.

3. Processing the records and creating the Reversal ACH File:

For the selected EFT Reversal Requested records, the job will create the ACH file to be sent to the bank.

- If the XML Vendor is miscellaneous and no payee vendor is available, the source field for Transaction Code for reversal will be modified to use the Account Type from the AD Transaction Vendor (AD_DOC_VEND) database table to determine the Transaction Code on the debit payment record type ‘6’ as follows:

If Account type on the Vendor Line of the AD Transaction is *Checking*, then the system sets the Transaction Code as 27.

If Account type on the Vendor Line of the AD Transaction is *Savings*, then the system sets the Transaction Code as 37.

- If the vendor is miscellaneous and no payee vendor is available, the source field for Receiving DFI Identification for reversal will be retrieved from the first eight digits of the ABA Number from the AD Transaction Vendor (AD_DOC_VEND) database table to determine the Receiving DFI Identification.
- If the vendor is miscellaneous and no payee vendor is available, the source field for Check Digit for reversal will be retrieved from the last digit of the ABA Number from the AD Transaction Vendor (AD_DOC_VEND) database table to determine the Check Digit.
- If the vendor is miscellaneous and no payee vendor is available, the source field for DFI Account Number for reversal will be retrieved from the first 17 digits of the Bank Account Number from the AD Transaction Vendor (AD_DOC_VEND) database table to determine the DFI Account Number.
- If the Vendor is miscellaneous and no payee vendor is available, the source field for Receiving Name for reversal will be retrieved from the Legal Name from the AD Transaction Vendor (AD_DOC_VEND) database table to determine the Receiving Name.

4. **Table updates:**

   Update EFT Reversal table.

   UPDATE R_EFT_RVRSL

   SET ACT_FL=”False”;

   UPDATE R_EFT_RVRSL

   SET REVERS_STA = “Reversal sent to the Bank”

   WHERE REVERS_STA = “Request Reversal”;

5. **Generate Reports:** After completing all the above steps, the job will generate the Requested EFT Reversals and Exception Report.

Restartability Information

The job cannot be restarted in this run mode. If the job fails in any of the above steps, a new job should be scheduled after correcting the errors that caused the job to fail.

Major Input

- EFT Reversal table
- Bank table
- Disbursement Format table
- CTX EDI Addenda table
- AD Transaction Vendor table
- Vendor Customer table

Batch Parameters

Parameter	Description	Default Value
Bank Code (BANK_ACCT_CD)	Optional; one or more bank code separated by comma can be entered.	No Default
Generate CTX EDI Addenda	Required Parameter. Valid values are *True* or *False*. Indicates whether or not the Addenda records for EFT reversals with the CTX format should be created.	No Default
Generate Credit Record	Required Parameter. Valid values are *True* or *False*. If set to *True*, a single summarized credit record will be created at the end of each ACH batch regardless of the ACH format. If set to *False* no summarized credit records will be created for ACH Batches	No Default

Transaction Code for Credit Record	Conditionally Required. When the Generate Credit Record parameter is set to *True*, this parameter is required. If entered valid values are *22* (automated deposit for Checking Account), *32* (automated deposit for Savings Account) or *42* (Automated general ledger deposit).	No Default
Lag Days	Required parameter. Entered value must be a positive numeric value. Represents the number of calendar days to be added to the Application Control Date to calculate the "Effective Entry Date" when creating the ACH file.	No Default
Company Entry Description	Required parameter. The entered value will be populated on the ACH Batch Header record for reversals.	REVERSAL
Credit Detail Discretionary Data	Required parameter. Length of the parameter should not exceed 2 characters.	No Default
Debit Detail Discretionary Data	Required parameter. Length of the parameter should not exceed 2 characters.	No Default
Company Entry Description for Reclamation	Required parameter. The entered value will be populated on the ACH Batch Header record for reclamations.	RECLAIM
Settlement Date (SETTLEMENT_DT)	Required parameter. Valid values are: • 1 (sets value to 000), • 2 (sets value to 3 blank spaces), and • alphanumeric string of length 3.	1

Major Output

- The job generates the following ACH files:
  - ACH CTX File with EFT Reversals
  - ACH CCD File with EFT Reversals
  - ACH PPD File with EFT Reversal

- The job updates the following tables:
  - EFT Reversal table

Job Return code

The following table shows the potential job return codes for EF ACH Reversal Process:

Return Code	Condition
Successful (1)	All of the validations are performed successfully and no errors as reported.
Warning (4)	No records found under the selection criteria.
Non-Fatal Error (8)	N/A
Failed (12)	The job fails under the following conditions: • Parameters are invalid. • No records that meet existing criteria were found • Run time exceptions for unexpected situations. When this job ends with a Return Code of *Failed*, subsequent jobs in the chain are set to *Inactive*.
Terminated (16)	This return code is issued when the job is terminated by the user. When this job ends with a Return Code of *Terminated*, subsequent jobs in the chain are set to *Inactive*.
System Failure (20)	This return code is issued when the job is terminated because of database server or network issues.

Selection Criteria

N/A

Sort Sequence

Within each Disbursement Format to process, the selected records will be sorted by Bank Account Code, Reclamation flag, EFT Number, EFT Transaction Code, EFT Transaction Dept, EFT Transaction ID, and EFT Transaction Version Number.

Problem Resolution

If the job ends with a return code of *Failed* and above, the job can be restarted.

The following table shows the various steps that the EF ACH Reversal Batch Process job goes through and the messages issued at each step.

Step 1: Parameter Validation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the table validations are successful, that is, there are no errors raised during validations.	N/A	N/A

Warning (4)	N/A	This step does not issue this return code.	
Non-Fatal Error (8)	If more than one Bank Code is entered separated by a comma but one of the values is not valid on the Bank Table **Sample Message:** "Bank Code xxxx is not valid on the Bank Table" (xxxx being the Bank Code).	The job should skip that Bank code and continue with the remaining valid Bank Codes	
Failed (12)	All of the required parameters are not entered. **Sample Message:** Generate CTX EDI Addenda parameter is required **Recommandation:** Enter the Generate CTX EDI Addenda parameter.	If the required parameters are not entered, enter the required parameters and restart the job.	
	Entered Parameters are not valid. **Sample Message**: "Entered Generate CTX EDI Addenda parameter is not a valid value" **Recommendation:** Enter a valid Generate CTX EDI Addenda parameter.	If the parameters are invalid, enter the valid parameter and restart the job.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can either be restarted or a new job scheduled.	
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can either be restarted or a new job scheduled.	

Step 2: Selection of records:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	Under the following conditions: • If Bank Code parameter is not null, the Generate Reversal ACH transaction shall select the EFT Reversal Table records with the Bank Code that matches the Bank Code parameter, the Active flag set to *True* and the Reversal Status set *to Request Reversal*. • If Bank Code parameter is null, the Generate Reversal ACH transaction shall select the EFT Reversal Table records with the Active flag set to *True* and the Reversal Status set to *Request Reversal*. **Sample Message:** "No eligible records found on the EFT Reversal Table". "No eligible records found on the EFT Reversal Table".	Make sure that the parameter value entered is correct. If the parameter value is incorrect, then enter the correct parameter value and run the chain again.	N/A
Non-Fatal Error (8)	N/A	N/A.	N/A
Failed (12)	Failed because of runtime exceptions for an unexpected situation.	This step is performed only if the parameter validation is successful. It can fail with fatal conditions only on encountering unknown exceptions. If that happens, investigate the exception reported by the process, resolve the error and reschedule the entire chain.	N/A

Terminated(16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated before rescheduling the entire chain.	N/A
System Failure(20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated before rescheduling the entire chain.	N/A

Step 3: Create and Update EFT Reversal table:

For the selected EFT Reversal Requested records, the job will create the ACH file to be sent to the bank. After generating the ACH file for the selected records, the process will update EFT Reversal table.

Possible Return Codes	Condition	Recommendation	Other instructions
Successful (1)	The system successfully creates and updates the EFT Reversal table.	N/A	
Warning (4)	N/A	N/A	
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Failed because of runtime exceptions for an unexpected situation.	Failure reason needs to be investigated before restarting the job.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can either be restarted or a new job scheduled.	
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can either be restarted or a new job scheduled.	

EF ACH Reversal: Build Flat File

Job Name	Build Flat File
Recommended Frequency	Daily as part of the nightly cycle or On Demand.
Single Instance Required	Yes

Can be restarted?	No
Reports Generated	No

Overview

The Build Flat File process will generate the Reversal ACH Files for the supported ACH formats of PPD+, CCD+, and CTX.Seperate formatted ACH Files will be generated for each of the valid formats. The main parameters for this job are the EF ACH XML File Name, Export Location at Build Flat File Job, Import Location at Build Flat File Job and Action Code.

Restartability

This job does not support restartability. If the job fails it has to be rescheduled again.

Major Input

- The EF ACH XML File Name.
- The Import and Export Locations
- Action Code.

Batch Parameters

Parameter	Description	Default Value
Action Code: Table Import (ACTN_CD)	This is a parameter for the Action Code.	201
Export Location at Build Flat File Job (AMSEXPORT)	This is a parameter for the Export location. Directory where the Flat Files are created.	
Import Location at Build Flat File Job (AMSIMPORT)	This is a parameter for the import location. Directory where the XML file is located.	
EF ACH XML File Name (FILE_NAME)	This is a parameter for the EF ACH Reversal XML File name.	EFACHReversalFile.xml
FLAT_FILE_EXTENSION	Flat file extension. Valid values are *DAT* or *txt*.	DAT
FLAT_FILE_PREFIX	Flat file prefix	Reversal
FLAT_FILE_SUFFIX	Flat file suffix	File

Major Output

Three Flat Files will be generated for each ACH format depending upon the format this job encounters.

Job Return code

The following table shows the potential job return codes for Build Flat File Process:

Return Code	Condition
Successful (1)	All of the validations are performed successfully and no errors as reported.
Warning (4)	N/A
Non-Fatal Error (8)	N/A
Failed (12)	The job fails under the following conditions: • Parameters are invalid. • EFACHReversalFile.xml was not found • Run time exceptions for unexpected situations. When this job ends with a Return Code of *Failed*, subsequent jobs in the chain are set to *Inactive*.
Terminated (16)	This return code is issued when the job is terminated by the user. When this job ends with a Return Code of *Terminated*, subsequent jobs in the chain are set to *Inactive*.
System Failure (20)	This return code is issued when the job is terminated because of database server or network issues.

Selection Criteria

N/A

Sort Sequence

N/A

Problem Resolution

If this job fails, then this job has to be rescheduled. All cases of failure have to be investigated before rescheduling the job. In case of Parameter validation failures, provide valid parameters while rescheduling the job.

Step 1: Parameter Validation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the table validations are successful, that is, there are no errors raised during validations.	N/A	N/A
Warning (4)	N/A	This step does not issue this return code.	
Non-Fatal Error (8)	N/A	This step does not issue this return code.	
Failed (12)	All of the required parameters are not entered.	If the required parameters are not entered, enter the required parameters and restart the job.	
	Entered Parameters are not valid. **Sample Message:** EF ACH XML File Name is required **Recommendation:** Enter valid values and Proceed	If the parameters are invalid, enter the valid parameter and restart the job.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can either be restarted or a new job scheduled.	
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can either be restarted or a new job scheduled.	

Step 2: Flat File Creation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	N/A	N/A	N/A
Warning (4)	N/A	N/A	N/A

Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Failed because of runtime exceptions for an unexpected situation.	The reason for the failure needs to be investigated before rescheduling the job.	Please refer to the Problem Resolution section
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can either be restarted or a new job scheduled.	Please refer to the Problem Resolution section
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can either be restarted or a new job scheduled.	Please refer to the Problem Resolution section

EF ACH Reversal: Initiate Disbursement Cancellation

Job Name	Initiate Disbursement Cancellation
Recommended Frequency	Daily as part of the nightly cycle or On Demand.
Single Instance Required	Yes
Can be restarted?	No
Reports Generated	• Exception Report • Initiated Cancellations of Reversed EFT Payments report

Overview

The Initiate Disbursement Cancellation job selects the eligible records from the EFT Reversal table and inserts records into the Disbursement Cancellation parameter table so that the Mass Cancellation process can cancel those EFT payments that were reversed by the bank.

Steps for running this process:

1. Parameter Validation: First the process will validate the batch parameters.
2. Selection of Records: Once the parameter validation is successful, the job will select the eligible records from the EFT Reversal Table with Reversal Status set to Reversal Confirmed.

   Select * from R_EFT_RVRSL

   Where REVERS_STA = *Reversal Confirmed*.

3. Processing Initiate Disbursement Cancellation:

The Initiate Disbursement Cancellation job will updates the EFT Reversal table.

If a matching record is found on the Check Recon table with the Status of *Disbursed* or *Warranted* then insert a record on the Disbursement Cancellation Parameters table as shown below:

- Bank Account Code – from the EFT Reversal Table
- Cancellation Type – from the EFT Reversal Table. If it is blank on the EFT Reversal Table, batch parameter value will be populated.
- Cancellation Reason – from the EFT Reversal Table. If it is blank on the EFT Reversal Table, batch parameter value will be populated.
- Hold Type – from the EFT Reversal table.
- Hold Request Description – from the EFT Reversal table.
- Payment Hold Type Department – from the EFT Reversal table.
- Payment Hold Unit Type – from the EFT Reversal table.
- Start Check Number – from the EFT Reversal Table Check EFT Number.
- End Check Number – from the EFT Reversal Table Check EFT Number.
- Active flag will be set to *True*.
- Scheduled Payment Date – from the EFT Reversal Table. If it is blank on the EFT Reversal Table, batch parameter value will be populated.
- Comments – will be populated with the text “Updated by the Reversal ACH Transaction Process”.

4. **Table updates:** Update EFT Reversal table.

   UPDATE R_EFT_RVRSL

   SET REVERS_STA = “EFT Cancel Initiated”

   WHERE REVERS_STA = “Reversal Confirmed”;

   UPDATE R_EFT_RVRSL

   SET EFT_CACEL_INIT_DT = Date on IN_APPCTRL;

   UPDATE R_EFT_RVRSL

   SET SYS_MSG =” Updated by the Reversal ACH Transaction Process”;

5. **Generate Reports:** After performing the all the steps, the job will generate the Exception Report and the Initiated Cancellations of Reversed EFT Payments Report.

Restartability Information

The job cannot be restarted in this run mode. If the job fails in any of the above steps, a new job should be scheduled after correcting the errors that caused the job to fail.

Major Input

- EFT Reversal table
- Check Recon table
- Paid Check table
- Bank table

Batch Parameters

Parameter	Description	Default Value
Cancellation Type (CANCEL_TYP)	This is a required parameter. Valid values are *Hold*, *Reschedule*, *Close* and *PR Cancellation*.	
Cancellation Reason (CANCEL_REASN)	This is a required parameter. The entered value must be valid on the Cancellation Reason CVL.	
Rescheduled Payment Date (RESCHD_PAY_DT)	Conditionally Required. Required only when the Cancellation Type is equal to *Reschedule*. If entered, the date must be greater than or equal to the Application control date.	

Major Output

The job generates the following ACH files:

- ACH CTX File with EFT Reversals
- ACH CCD File with EFT Reversals
- ACH PPD File with EFT Reversal

The job updates the following tables:

- EFT Reversal table

Job Return code

The following table shows the potential job return codes for Initiate Disbursement Cancellation Process:

Return Code	Condition
Successful (1)	All of the validations are performed successfully and no errors as reported.
Warning (4)	No records found under the selection criteria.

Non-Fatal Error (8)	N/A
Failed (12)	The job fails under the following conditions: • Parameters are invalid. • No records that meet existing criteria were found • Run time exceptions for unexpected situations. When this job ends with a Return Code of *Failed*, subsequent jobs in the chain are set to *Inactive*.
Terminated (16)	This return code is issued when the job is terminated by the user. When this job ends with a Return Code of *Terminated*, subsequent jobs in the chain are set to *Inactive*.
System Failure (20)	This return code is issued when the job is terminated because of database server or network issues.

Selection Criteria

N/A

Sort Sequence

N/A

Problem Resolution

If the job ends with a return code of *Failed* and above, the job can be restarted.

The following table shows the various steps that Initiate Disbursement Cancellation Process goes through and the messages issued at each step.

Step 1: Parameter Validation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the table validations are successful, that is, there are no errors raised during validations.	N/A	N/A
Warning (4)	If the Rescheduled Payment Date (RESCHD_PAY_DT) is *Blank* and the Cancellation Type (CANCEL_TYP) is *Reschedule*. **Sample Message:** "Rescheduled Payment Date is required when the	The process will default the Rescheduled Payment Date to the Application Control Date.	

	Cancellation Type is Reschedule.”		
Non-Fatal Error (8)	N/A	This step does not issue this return code.	
Failed (12)	All of the required parameters are not entered. **Sample Message:** “Cancellation Type is required.” **Recommendation:** Enter the Cancellation Type parameter.	If the required parameters are not entered, enter the required parameters and restart the job.	
	Entered Parameters are not valid. **Sample Message**: “Entered Cancellation Type is not a valid cancellation type.” **Recommendation:** Enter a valid cancellation type parameter.	If the parameters are invalid, enter the valid parameter and restart the job.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can either be restarted or a new job scheduled.	
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can either be restarted or a new job scheduled.	

Step 2: Selection of records:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	Under the following conditions: If there are no records found on the EFT Reversal Table with the status *Reversal Confirmed*.	Make sure that the parameter value entered is correct. If the parameter value is incorrect, then enter	N/A

	Sample Message: • "No eligible records found on the EFT Reversal Table".	the correct parameter value and run the chain again.	
Non-Fatal Error (8)	N/A	N/A.	N/A
Failed (12)	Failed because of runtime exceptions for an unexpected situation.	This step is performed only if the parameter validation is successful. It can fail with fatal conditions only on encountering unknown exceptions. If that happens, investigate the exception reported by the process, resolve the error and reschedule the entire chain.	N/A
Terminated (16)	Job is terminated manually by the user	The reason for the termination needs to be investigated before rescheduling the entire chain.	N/A
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated before rescheduling the entire chain.	N/A

2.1.18 EF ACH Transaction

Job Name	EF ACH Transaction
Recommended Frequency	After each instance of or after all instances of the AD Chain. If creating CTX payments, the CTX EDI Formatting job should be run first.
Single Instance Required	Yes
Can be restarted?	N/A
Reports generated	N/A

Overview

Automated Clearing House (ACH) participants can electronically transfer payment data in a structured, machine retrievable data format that permits data to be transferred without re-keying from a business application in one financial institution to a business application in another financial institution. This transfer process is called Electronic Funds Transfer (EFT).

The EF ACH Transaction process handles the following ACH formats:

- PPD+ (Prearranged Payment or Deposit)
- CCD+ (Cash Concentration or Disbursement)
- CTX (Corporate Trade Exchange)
- IAT (International ACH Transactions)

The EF ACH Transaction job is a chain job that consists of two individual jobs.

- **Build XML File**: Takes information from one or more runs of the AD Chain to create an ACH XML File.
- **Build Flat File**: Takes the XML file created in the first job to create the ACH Flat File.

If the USE_EFACH_CONFIG parameter on the Application Parameter (APPCTRL) page is set to *True*, the process reads through the ACH Configuration pages for information to create the ACH output file(s). For any missing Record Types in the ACH Configuration pages, the process uses existing code to create the ACH output file(s). Refer to the “ACH Configuration Pages” topic in the *Disbursement User Guide* for more details on the 3 ACH Configuration pages.

Major Input

- AD Transaction Header (AD_DOC_HDR)
- ACH XML File Configuration (ACHXML / R_ACH_XML_CNFG)
- ACH Flat File Configuration (ACHFF / R_ACH_FILE_CNFG)

Major Output

- XML File - EFACHFormat.xml
- One or more flats files with file name variations controlled by batch parameters.

- EFACHCCD File.DAT
- EFACHCTX File.DAT
- EFACHPPD File.DAT
- EFACHIAT File.DAT

Chain Return Codes

When an active job step in the chain ends with a Return Code of anything other than *Successful*, any remaining job steps have a Run Status of *Inactive.* Please see documentation on individual job steps for possible Return Codes for those steps, as *Warning* and *Non-Fatal Error* are not always possible outcomes.

Problem Resolution

Failure at any point in the chain requires addressing the failure and submitting a new instance of the chain.

EF ACH Transaction Chain: Build XML File job

Job Name	Build XML File
Recommended Frequency	See chain
Single Instance Required	N/A
Can be restarted?	No
Reports generated	No

Overview

The Build XML File process selects the EFT Transactions and creates the XML file to be used as input for the next process in the chain. This process posts the EFT Number assigned by the AD Chain process for successfully submitted EFT transactions to the ID Number of the credit payment record (6) on the ACH file. This process only selects those EFT transactions that are greater than zero and includes them in the output file that will be sent to the banks.

This process also posts detailed information on the addenda records where it generates a single addenda record for each accounting line associated with an EFT transaction. If an EFT accounting line has any adjustments (for example, intercept, discount, interest), then each adjustment amount associated with the Accounting Line is also posted to a separate addenda record. The following information is posted to the ACH CTX Addenda Records:

- Payment Request ID
- Vendor Invoice Number
- Amount (Payment, Discount, Intercept, etc.)
- The first 20 characters of the Check Description or one of the following values based on the amount type in case of adjustments (Discount, Backup Withholding, 3402(t) Contract Withholding, Penalty, Interest, Retainage, Freight, Intercept, Tax or Use Tax).

However, the information posted to the addenda records is not formatted based on the Exchange Data Interchange (EDI) ANSI ASC X12 Transaction Set 820 Remittance Advice/Payment Order.

This process generates a summary debit payment (“6”) within each Batch in the ACH file. The amount posted to this record represents the sum amounts of all the credit payments within a batch.

Additionally, this process generates an IAT ACH file for foreign bank branches based on whether the IAT is international or domestic. If a foreign bank branch has no Gateway Operator involved in the transaction, instead, Vendor ABA number will appear in both the Record 6 (field 3) and 5th IAT addenda record (714).

- If the domestic branch of a foreign bank handles foreign payments, then the addenda Record Type (718) is inserted in the generated ACH file.
- If the domestic branch of a foreign bank does not handle foreign payments, then the addenda Record Type (718) is not inserted in the generated ACH file.

If the EFT Payments are disbursed from more than one Bank Account Code during an AD Chain run, then the EF ACH Transaction process will group the EFT payments having the same Bank Account Code together (that is, in a separate ACH batch) within the ACH file.

If the USE_EFACH_CONFIG parameter on Application Parameter (APPCTRL) is set to *True*, the Build XML File job reads through the ACH XML Configuration (ACHXML) page for information to create the ACH XML file(s). For any missing record types in the ACH XML Configuration page, the process uses existing code to create the ACH XML file(s).

The process performs the following steps:

Process Steps	Messages
1. Parameter Validation	• Validating Batch Parameters • Parameters are valid or invalid depending on the Validation. If the parameter is invalid, the invalid value is displayed in the log. • All batch parameters are valid
2. File Creation	• Report folders mapped • Warning of no records found by format. • EF ACH XML file successfully generated.

Major Input

- AD Transaction Header (AD_DOC_HDR)
- AD Transaction Vendor (AD_DOC_VEND)
- AD Transaction Accounting (AD_DOC_ACTG)
- Vendor Customer (VCUST / R_VEND_CUST)
- Disbursement Format (DISF / R_AP_DISB_FRMT)
- Vendor ABA (VABA / R_VEND_ABA_MAINT)
- BANK (BANK / R_AP_BANK_ACCT)
- ACH XML File Configuration (ACHXML / R_ACH_XML_CNFG)

Parameter	Description	Default Value
Payer Contact Code (CNTAC_CD)	A required contact (CNTAC) that is the payer name/address that will be included with the electronic file.	No Default
Payer Department Code (DEPT_CD)	A required output parameter that supplies the payer department name.	No Default
Payer Unit Code (UNIT_CD)	A required output parameter that supplies the payer department name.	No Default
From Run ID (FRM_RUN_ID)	An optional selection parameter that specifies the beginning of a group of payments. If not entered, all available payments are selected.	No Default
To Run ID (TO_RUN_ID)	An optional selection parameter that specifies the end of a group of payments. If not entered, all available payments are selected.	No Default
Regenerate ACH File (REGEN_ACH_FILE)	A required output indication if the job produces new payment or regenerates the ACH file from a prior run. Valid values are *Y-Yes* and *N-No*.	N
Export Location at Build XML File Job (AMSEXPORT)	The required location to write the XML file.	$$AMSROOT$$/ExportImport

Receiving Name
(RCV_NM)
	A required output parameter that controls the Receiving Name written:
	• *1* = Receiving Name set to Vendor Customer Code
	• *2* = Receiving Name based on Organization Type:
	  • *Individual* – Last Name
	  • *Company* - Company Name
	  • Neither - Legal Name
	• *3* = Receiving Name is Last Name when => 17 characters. Last Name, First Name when Last Name < 17. Company Name when name fields are blank.
	• *4* = Receiving Name is the Vendor's Legal Name (First Name, Last Name, or Company
		1

Parameter	Description	Default Value
	Name) irrespective of Organization type.	
Credit Detail Discretionary Data (CD_DISCRT)	An optional parameter for the Discretionary Date field on credit records. If entered, the length should not exceed 2 characters. If not supplied, the Disbursement Format Discretionary Data value is used.	No Default
Debit Detail Discretionary Data (CD_DISCRT)	An optional parameter for the Discretionary Date field on debit records. If entered, the length should not exceed 2 characters. If not supplied, the Disbursement Format Discretionary Data value is used.	No Default
Company Entry Description (CE_DESC)	An optional output parameter that controls the length of what is written to the Company Entry Description field on the Header record. The field length in the file is limited to 10 characters.	No Default
Effective Source Date (EFECTIVEDATE)	A required parameter that controls what is written into the CLR_DT field of the output ACH file based on the following valid values: • *1 - Cleared Date* or • *2 - Issue Date* or • *3 – Issue Date and Cleared Date* When set to *3*, the process writes CHK/EFT Issue Date to the CLR_DT field of the ACH file and applies selection logic to fetch a distinct combination of Cleared Date and Check/EFT Issue Date of the records that need to be processed and creates batch header node per each distinct record.	No Default
Bank Code (BANK_CD)	An optional selection parameter of one or more bank codes (separated by commas. If left blank, then all banks are selected.	No Default
Generate Addenda for Debit Record (DR ADDENDA_GEN)	A required output parameter that controls the generation of addenda records for debit. Valid values are *True* or *False*.	False

Parameter	Description	Default Value
Generate CTX EDI Addenda (CTX_ADDENDA_GEN)	A required output parameter that controls the generation of addenda records for debit. Valid values are *True* or *False*. For the CTX format, this is typically true based on the EDI ANSI ASC X12 820 Remittance Advice/Payment Order.	False
Generate Debit Record (DR_GEN)	A required output parameter that controls the generation of debit records. Valid values are *True* or *False*.	No Default
Transaction Code for Debit Record (DR_TRANS_CD)	A conditionally required output parameter. Valid values are: • *27* - Automated payment for Checking Account • *37* - Automated payment for Savings Account • *47* - Automated general ledger deposit.	No Default
Include PCARD Payments (GEN_EFT_PCARD_PYMT)	A required output parameter that controls if procurement card payments are included. Valid values are *Yes* or *No*.	Yes
Settlement Date (SETTLEMENT_DT)	A required output parameter that controls what is written to the date field. • *1* - 000 • *2* - 3 blank spaces • *3* - alphanumeric string of length 3	1
CTX First Addenda Only (CTX_FIRST_ADDENDA_ONLY)	An optional output parameter for CTX files that specifies if only the first addenda will be written to the XML file: • Y - Writes only the first CTX addenda record to the XML file. • N – Writes all available CTX addenda records to the XML file. • Note: CTX Non-EDI Addenda records (Record 7) that are generated when the accounting line has any adjustments	N

Parameter	Description	Default Value
	(discounts, backup withholding, penalty, and so forth) are not impacted by this parameter. Adjustment lines are written in a separate addenda record based on the existing baseline format, or ACH structure.	
Validate Healthcare AL Check Description for CCD (VALIDATE_HC_CHK_DSCR)	An optional parameter that validates if the check description field is entered in the first accounting line of each EFT transaction with the CCD format for healthcare payments. Valid values are *Y* and *N*.	N
Read Contact Code from Unit/Department (CNTAC_CD_FRM_UNIT_DEPT)	An optional output parameter that when set to Y (yes) writes the contact (CNTAC) details that is payer name/address/phone that will be included with the output XML file with the following steps until a contact is found: 1. Disbursement contact code associated with the referenced transaction unit. 2. Disbursement contact code associated with the referenced transaction department. 3. Contact code specified in batch parameter CNTAC_CD (Payer Contact code). 4. 3 spaces when batch parameter CNTAC_CD is blank. When set to N, writes the contact (CNTAC) details that will be included with the output XML file as per the contact code specified in the CNTAC_CD batch parameter.	N
Generate Debit Record Formats (GEN_DEBT_REC_FRMTS)	A required output parameter when the GEN_DEBT_REC (Generate Debit Record) parameter is set to *True* and generates debit records to the ACH output file only for the ACH formats specified.	PPD, CCD, and CTX
Skip reading Company ID from BANK (SKIP_BANK_ACH_CMP_ID)	An optional output parameter that when set to Y(Yes) skips the existing overriding logic to fetch ACH Company ID from the BANK page and the COMP_ID (Company	N

Parameter	Description	Default Value
	identification) attribute in the ACH XML file is updated with the Company Identification value from the Disbursement Format (DISF). When set to N(No), the existing overriding logic to fetch ACH Company ID value from BANK applies. If both BANK and DISF pages do not have Company ID and the parameter is set to N, the process will fail. Note: The EF ACH Transaction process runs successfully even when SKIP_BANK_ACH_CMP_ID is set to Y, with a blank value in the Company ID field of DISF and BANK table, and updates with equivalent blank values in the COMP_ID attribute of the ACH output file generated.	

Major Output

- XML File (EFACHFormat.xml)
- Update AD_DOC_HDR for attribute “ACH_FILGEN_STA”

Sort Criteria

None

Selection Criteria

- Select distinct (Disbursement Format) from AD_DOC_HDR
- Where “Disbursement Type” = EFT
- When Run IDs are not the batch parameter and Regenerate ACH File is true, then

  Select distinct (Bank Account Code) from AD_DOC_HDR

  Where “Disbursement Type” = EFT

  And “Transaction Phase Code” = “Final”

  And “Disbursement Format” = Format from the above SQL

  And “ACH File Generation Status” In (‘Generated’)

  Select distinct (Clear Date) from AD_DOC_HDR

  Where “Disbursement Type” = EFT

  And “Transaction Phase Code” = “Final”

  And “Disbursement Format” = Format from the above SQL

And “Bank Account Code” = Bank Account Code from the above SQL

And “ACH File Generation Status” In (‘Generated’)

Select all records from AD_DOC_HDR

Where “Disbursement Type” = EFT

And “Disbursement Format” = Format from the above SQL

And “Bank Account Code” = Bank Account Code from the above SQL

And “Clear Date” = Clear Date from the above SQL

And “ACH File Generation Status” In (‘Generated’)

- When Run IDs are the batch Parameter and Regenerate ACH File is False, then

Select distinct (Bank Account Code) from AD_DOC_HDR

Where “Disbursement Type” = EFT

And “Transaction Phase Code” = “Final”

And “Disbursement Format” = Format from the above SQL

And “ACH File Generation Status” In (‘Ready For Generation’)

And “Run ID” >= From Run ID

And “Run ID” <= To Run ID

Select distinct (Clear Date) from AD_DOC_HDR

Where “Disbursement Type” = EFT

And “Transaction Phase Code” = “Final”

And “Disbursement Format” = Format from the above SQL

And “Bank Account Code” = Bank Account Code from the above SQL

And“ACH File Generation Status” In (‘Ready For Generation’)

And “Run ID” >= From Run ID

And “Run ID” <= To Run ID

Select all records from AD_DOC_HDR

Where “Disbursement Type” = EFT

And “Transaction Phase Code” = “Final”

And “Disbursement Format” = Format from the above SQL

And “Bank Account Code” = Bank Account Code from the above SQL

And “Clear Date” = Clear Date from the above SQL

And “ACH File Generation Status” In (‘Ready For Generation’)

And “Run ID” >= From Run ID

And “Run ID” <= To Run ID

Problem Resolution

The following table shows the potential job return codes for this job.

Return Code	Condition
Successful (1)	The job ends as successful when all parameters are valid, at least one EFT was selected, and the XML file is created successfully.
Warning (4)	Warning results when no records were found that matched selection criteria.
Non-Fatal Error (8)	The job has encountered a record selection error and skipped an EFT.
Failed (12)	The job fails under the following conditions: • Parameters are invalid or missing • Run time exceptions for unexpected situations. When this job fails, subsequent jobs in the chain are set to *Inactive*.
Terminated (16)	The job is terminated by the user. When this job is terminated, subsequent jobs in the chain are set to *Inactive*.
System Failure (20)	A system failure is issued when the job is terminated because of database server or network issues. When this job encounters a system failure, subsequent jobs in the chain are set to *Inactive*.

Problem Resolution

The following table shows the possible return codes and recommendations for each processing step.

Step 1: Parameter Validation

Return Code	Condition	Recommendation	Other Instructions
Successful (1)	The step ends as successful when all parameters are valid.	N/A	N/A
Warning (4)	This step does not end with this return code.	N/A	N/A
Non-Fatal Error (8)	This step does not end with this return code.	N/A	N/A
Failed (12)	This step failed due to an invalid parameter.	Address the invalid parameter before running another instance.	N/A

Return Code	Condition	Recommendation	Other Instructions
Terminated (16)	The job was terminated manually by the user.	The reason for the termination needs to be addressed.	N/A
System Failure (20)	The job was terminated because of database server or network issues.	The reason for the System Failure needs to be investigated.	N/A

Step 2: File Creation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	At least one EFT was found and the XML file created successfully.	N/A	N/A
Warning (4)	This job step ends as warning if no EFTs are found to match the selection criteria.	Evaluate your selection criteria against recent EFT data before submitting another instance.	N/A
Non-Fatal Error (8)	This job step ends as non-fatal when there is no Check Description on the first accounting line when the parameter requires it. **Example message:** Check Description is required for the Healthcare ACH File. Transaction Code: EFT, Transaction Department Code: XXX, Transaction ID: <Transaction ID> is not processed.	The bypassed EFT needs to be cancelled to reopen the payment request where a Check Description needs to be entered, and the request paid again.	N/A
Failed (12)	Job failed due to Fatal conditions.	In this step, the job can fail if it encounters any runtime exceptions. Investigate the exception reported by the process. Resolve the error and submit a new instance.	N/A

Possible Return Codes	Condition	Recommendation	Other Instructions
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. Once resolved, submit a new instance.	N/A
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. Either restart the job or schedule a new job.	If another instance of the job has already been scheduled and run successfully, then this job should not be restarted – only a new job should be scheduled.

EF ACH Chain: Build Flat File job

Job Name	Build Flat File
Recommended Frequency	See Chain
Single Instance Required	N/A
Can be restarted?	Yes
Reports generated	No

Overview

The Build Flat File process creates ACH file(s) with data from the EFACHFormat.xml.

If the USE_EFACH_CONFIG parameter on the Application Parameter (APPCTRL) page is set to *True*, the Build Flat File job reads through the ACH Flat File Configuration (ACHFF) page for information to create the ACH Flat file(s). For any missing record types in the ACH Flat File Configuration page, the process uses existing code to create the ACH Flat file(s).

The process performs the following steps:

Process Steps	Messages
1. Parameter Validation	• Validating Batch Parameters. • Parameters are valid or invalid depending on the Validation. If the parameter is invalid, the invalid value is displayed in the log.
2. File Creation	• Flat file generated for format. • Flat files generated successfully.

Major Input

- XML File (EFACHFormat.xml)
- ACH Flat File Configuration (ACHFF / R_ACH_FILE_CNFG)

Parameters

The following custom parameters are required for the EF ACH Transaction job.

Parameter	Description	Default Value
EF ACH XML File Name (FILE_NAME)	The required input file name created by the previous job step.	EFACHXMLFormatFile.xml
File Extension (FLAT_FILE_EXTENSION)	The required file extension. Valid values are *DAT* or *TXT*.	DAT
File Prefix (FLAT_FILE_PREFIX)	An optional prefix for the file created.	EFACH
File Suffix (FLAT_FILE_SUFFIX)	An optional suffix for the file created.	File
Return Code when Flat files are not generated (NO_FLAT_FILE_GEN_RET_CODE)	An optional parameter to update return code when no flat files are generated post job completion with valid values: 1 - Success, 4 - Warning. • Value 1 returns the Return Code of Successful. • Value 4 returns the Return Code of Warning.	4

Output

Flats files will be generated, with variations controlled by batch parameters.

- EFACHCCD File.DAT
- EFACHCTX File.DAT
- EFACHPPD File.DAT
- EFACHIAT File.DAT

Sort Sequence

None

Selection Criteria

None

Problem Resolution

No database restore is required. Correct the problem and rerun the job executing the program. No restoration of datasets or files from backups is required for this program.

Return Code	Condition
Successful (1)	The job ends as successful when all parameters are valid and the flat file creates successfully.
Warning (4)	This return code does not apply.
Non-Fatal Error (8)	This return code does not apply.
Failed (12)	The job fails under the following conditions: • Parameters are invalid. • Run time exceptions for unexpected situations.
Terminated (16)	The job is terminated by the user. When this job is terminated, subsequent jobs in the chain are set to *Inactive*.
System Failure (20)	A system failure is issued when the job is terminated because of database server or network issues.

Problem Resolution

The following table shows the possible return codes and recommendations for each processing step.

Step 1: Parameter Validation

Return Code	Condition	Recommendation	Other Instructions
Successful (1)	The step ends as successful when all parameters are valid.	N/A	N/A
Warning (4)	This step does not end with this return code.	N/A	N/A
Non-Fatal Error (8)	This step does not end with this return code.	N/A	N/A
Failed (12)	This step failed due to an invalid parameter.	Address the invalid parameter before running	N/A

Return Code	Condition	Recommendation	Other Instructions
		another instance.	
Terminated (16)	The job was terminated manually by the user.	The reason for the termination needs to be addressed.	N/A
System Failure (20)	The job was terminated because of database server or network issues.	The reason for the System Failure needs to be investigated.	N/A

Step 2: File Creation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	The XML file was found and converted to a flat file.	N/A	N/A
Warning (4)	This job step does not use this return code.	N/A	N/A
Non-Fatal Error (8)	This job step does not use this return code.	N/A	N/A
Failed (12)	Job failed due to Fatal conditions.	In this step, the job can fail if it encounters any runtime exceptions. Investigate the exception reported by the process, resolve the error, and submit a new instance.	N/A
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. Once resolved, submit a new instance.	N/A
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. Restart the job or schedule a new job.	If another instance of the job has already been scheduled and run successfully, then this job should not be restarted – only a new job should be

Possible Return Codes	Condition	Recommendation	Other Instructions
			scheduled.

2.1.19 EFT/ACH Archive Process

Chain or Job Name	EFT/ACH Archive Process
Recommended Frequency	The job can be run on nightly basis or on demand.
Single Instance Required	Yes.
Can be restarted?	No.
Reports generated	SMU Report(s)

Overview

This chain job involves two batch jobs namely EFT/ACH Archive Record Selection and AMSFacilitator job. The first job in this chain performs the selection of eligible records. On successful selection of eligible records, the second job submits multiple instances of the System Maintenance Utility (SMU) for archiving records selected by first job.

The purpose of the first job is to select eligible records from the target tables namely EFT Reversal, EFT Return, CTX EDI Addenda, Check Writer EFT Reversal, Check Writer EFT Returns, and AD Stub Details tables. The selected records are written to the parameter file with record keys being Bank Account Number and Check Number. This parameter file is the output of the first job. The AMSFacilitator job will begin the process of archiving the records by picking up these parameter files and store them in XML files.

No dependency for this chain job.

This chain implements the Data Warehouse Archived Record Queue process, which is used to retain the table names and key values for records that are archived and deleted. The infoAdvantage reporting system uses this information to distinguish between records that have been archived (as valid historical data) before deletion and records that have simply been deleted from the system (as no longer needed). The process is enabled for most archiving processes when the value of Application Parameter Enable Data Warehouse Archived Record Queue (ENABLE_DW_ARCH_QUEUE) is *True*. Please see the *CGI Advantage System Administration Guide* for more information regarding this process and the Application Parameter.

Major Input

- EFT Reversal table
- EFT Return table
- AD Stub Details table
- CTX EDI Addenda table
- AD/EFT Transaction
- Check Reconciliation table (R_AP_CHK_RECON)
- Paid Check table (AP_PD_CHK)
- Check Writer EFT Reversal table
- Check Writer EFT Returns table

Major Output

- XML file containing archived records
- SMU Report(s)
- U pdated target tables (EFT Reversal, EFT Return, AD Stub Details, Check Writer EFT Reversal, Check Writer EFT Returns, and CTX EDI Addenda) with selected records deleted from these tables.

Chain Job Return code

The following table shows the potential return codes for the EFT/ACH Archive job. Note that the Chain job will end with the highest return code across all of the jobs.

Return Code	Condition
Successful (1)	All of the jobs end successfully
Warning (4)	One of the jobs in the chain ends with a return code of Warning
Non-Fatal Error (8)	One of the jobs in the chain ends with a return code of Non-Fatal Error
Failed (12)	One of the jobs in the chain ends with a return code of Failed
Terminated (16)	One of the jobs in the chain ends with a return code of Terminated
System Failure (20)	One of the jobs in the chain ends with a return code of System Failure

Problem Resolution

If any of the jobs in the chain failed due to application errors it is advisable to re-schedule the job after correcting the errors.

Please refer to the individual jobs for details regarding the specific job processes and problem resolution.

EFT/ACH Archive Chain Job: EFT/ACH Archive Record Selection

Job Name	EFT/ACH Record Selection
Recommended Frequency	The job can be run on nightly basis or on demand.
Single Instance Required	Yes
Can be restarted?	No.
Reports generated	None

Overview

The EFT/ACH Archive Record Selection Batch Process is the first job in the job stream. The main purpose of this job is to select records for archiving from the target tables: EFT Reversal, EFT Return, AD Stub Details, Check Writer EFT Reversal, Check Writer EFT Returns, and CTX EDI Addenda tables per Bank Account Number and Check Number. Using Bank Account Number and Check Number as key information, records are selected and the process will build System Maintenance Utility parameter files that will be used to archive the selected records.

The following table shows the various steps that the EFT/ACH Archive Record Selection Job goes through and the messages issued at each step.

Process Steps	Messages
1. Parameter Validation	• Validating Batch Parameters • Parameters are valid or invalid depending on the Validation. If the parameter is invalid, the invalid value will be displayed in the log. • Batch Parameter validation completed
2. Selection of Records	• Selecting eligible records • If the selection returns 0 records, then the following message will be issued: “No eligible record found”. • Number of records (count) selected will be displayed in the end. In addition, every time the record count reaches the progression counter, number of records till this point will be displayed. • At the end, if all the steps for selection are executed successfully, message: “Selection of records completed.” will be displayed. • If any step during the record selection fails, then error will be issued.

Major Input

- EFT Reversal table
- EFT Return table
- Check Writer EFT Reversal table
- Check Writer EFT Returns table
- AD Stub Details table
- CTX EDI Addenda table
- AD/EFT Transaction table (AD_DOC_HDR)
- Check Reconciliation table (R_CHK_RECON)
- Check Paid table (AP_PD_CHK)

Batch Parameters

Parameter Name	Description	Default Value
Report Only (REPORT_ONLY)	Required non- editable field. This parameter defines the "Report_Only" parameter to define if records are to be purged in addition to reporting on the selected records.	N
File Location (FILE_LOCATION)	Required Field. This parameter defines the location to which the system generated parameter and XML files will be written.	$$AMSRoot$$/Parms
Tolerance (TOLERANCE)	This parameter defines a benchmark for the approximate number of records to be written to each XML archive file.	1000
Suppress SMU Reports Y or N (SUPPRESS_RPTS)	Required non- editable Field. This parameter defines the "SUPPRESS_REPORTS" parameter in the parameter file. When this parameter is defined as “Y” a related SMU report will not be generated for each processed parameter file. If “N” one SMU report is generated for each parameter file.	N
Progression counter (PROG_CTR_SZ)	Optional field. During processing, the job writes messages to the log to report on its progress based on the number of records already processed. When this parameter is specified, the value controls the interval at which these progression messages are written to the job log. A specified value should be a positive integer.	1000

Major Output

- Parameter file(s) with Job Manager Tables record keys.
- Add records to the Facilitator table with the parameter file name(s).
- ARCH_DW_QUEUE_REC – The Data Warehouse Archive Record table keeps track of the values for all primary keys of each record being archived and deleted. The table name and

attribute names are stored in ARCH_DW_QUEUE_TBL and linked to this record by unique ID.

- ARCH_DW_QUEUE_TBL – The Data Warehouse Archive table keeps track of the table name and primary key attribute names of each table being archived. If a record already exists for the table being archived, it will reuse the record. If it does not yet exist, it will be created.

Job Return code

The following table shows the potential job return codes for the EFT/ACH Record Selection job.

Return Code	Condition
Successful (1)	All of the selected payment records are processed successfully.
Warning (4)	No eligible records were found on any of the target tables.
Non-Fatal Error (8)	N/A
Failed (12)	The job fails under the following conditions: • Parameters are invalid • Run time exceptions for unexpected situations When this job ends with a return of code Failed, subsequent jobs in the chain will be set to inactive.
Terminated (16)	This return code will be issued when the job is terminated by the user. When this job ends with a return of code Terminated, subsequent jobs in the chain will be set to inactive.
System Failure (20)	This return code will be issued when the job is terminated because of database server or network issues. When this job ends with a return of code System Failure, subsequent jobs in the chain will be set to inactive.

Sort Sequence

N/A

Selection Criteria

This process involves selection of eligible records from the target tables namely EFT Reversal, EFT Return, CTX EDI Addenda and AD Stubs Details tables. Each record is eligible for archiving if it satisfies the following condition:

The Bank Account Number and Check Number for a record in the target table has no corresponding record with the same combination of Bank Account Number and Check Number in AD Transaction Header or Check Recon or Check Paid tables.

Selection of eligible records from Check Writer EFT Reversal and Check Writer EFT Returns tables takes place based on the following conditions.

The Bank Account Code, EFT Number, Transaction Code value of CE, Transaction Dept value from CW Department, and Transaction ID value from Check Writer File ID for a record in the target table cannot be found on the Check Reconciliation or Paid Checks tables.

The CW Department, Check Writer File ID, and EFT Number for a record in the target table has no corresponding record with the same combination of Check Writer Department, Check Writer File ID, and Check/EFT Number on the Check Writer Payment table.

These records are selected from all the six target tables and the selected records are deleted from their respective target tables after archival.

Problem Resolution

There is no need to back out any updates since this job will only select the records from the target tables.

The following table shows the possible return codes and recommendations for each processing step.

Step 1: Parameter Validation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All the parameters have been successfully validated	N/A	N/A
Warning (4)	N/A	This step doesn't issue this return code.	This step doesn't issue this return code.
Non-Fatal Error (8)	N/A	This step doesn't issue this return code.	This step doesn't issue this return code.
Failed (12)	Required Parameters are not entered **Sample Message:** File Location is not entered	Enter the File Location and restart the job.	Alternatively, the job can be rescheduled after correcting the parameters.
	Entered Parameters are not valid **Sample Message**: File Location is not valid	Enter the File Location and restart the job.	Alternatively, the job can be rescheduled after correcting the parameters.
	Failed because of runtime exceptions for an unexpected situation.	The reason for the failure needs to be investigated before restarting the job.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can	

		either be restarted or rescheduled.	
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can either be restarted or rescheduled.	

Step 2: Selection of records

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	The job ended with a Warning because there are no eligible records found on the target tables. **Sample Message:** No eligible records found on the target tables.	This could happen when there are no eligible records found on the target tables that do not match the Bank Account Number and Check Number on the AD/EFT Transaction Header, Check Recon and Check Paid.	N/A
Non-Fatal Error (8)	N/A	This step does not issue this return code.	This step does not issue this return code.
Failed (12)	Failed because of runtime exceptions for an unexpected situation.	In this step the job can fail with fatal conditions only on encountering unknown exceptions. If that happens, investigate the exception reported by the process, resolve the error and restart the job.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can either be restarted or rescheduled.	
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can either be restarted or	

		rescheduled.	

EFT/ACH Archive Chain Job: AMSFacilitator

Chain or Job Name	AMSFacilitator
Recommended Frequency	The job can be run on nightly basis or on demand.
Single Instance Required	Yes
Can be restarted?	Yes
Reports generated	SMU Report(s)

Overview

The AMSFacilitator job is the second job is this chain job. The purpose of the batch job is to archive all the records that are listed in the parameter file. The job also generates the SMU report(s) depending on the value of SUPPRESS_RPTS. This value is written to the parameter file in the first job. Once all the records have been stored in the xml file, all the eligible records are deleted from the target tables.

Major inputs

- Parameter file generated from the first job.

Batch Parameters:

Parameter Name	Description	Default Value
Archive Restore ID(1- Table Archive) (ARCHIVE_RESTORE_ID)	Required non-editable field. Entry of a value in this field specifies the System Maintenance Utility command to be executed during the Archive. Must be “1” for Archive.	1
Commit Block Size (COMMIT_BLOCK_SIZE)	Required field. Controls how many records are committed by the application at one time. The size should be compatible with technical capabilities and performance guidelines.	1000
Number of Processors (PROCESSOR_NO)	Required field. Defines the number of SMU processes to be executed simultaneously.	2
Run Number RUN_NO	This field need not be entered by user during SMU action “Table Archive”	

Sleep Time (Frequency to pool for free processing slots) SLEEP_TIME	Defines the frequency to pool for free processing slots when initiating SMU processes.	5
Update Facilitator Status UPDATE_STATUS	Update Facilitator Status. If Report Only is *Y* then Update Status must be *N*.	Y

Major Output

- XML file containing archived records
- SMU Report(s)
- Updated target tables (EFT Reversal, EFT Return, CTX EDI Addenda, Check Writer EFT Reversal, Check Writer EFT Returns, and AD Stub Details) with eligible records deleted from their respective tables.

The SMU Report(s) will only be generated if the Suppress Reports parameter is set to Y. In this case, a standard SMU report will be generated for each SMU parameter file created during the archiving process. The value of SUPPRESS_RPTS parameter in the first job is written to the parameter file which will be picked up by Facilitator during execution. In order to access the SMU report, find the SMU Job ID spawned by the Facilitator step. Then, find the SMU Job ID in the View All Jobs summary list. Click the View Reports link to see the SMU report.

Job Return code

The following table shows the potential job return codes for the EFT/ACH Record Selection job.

Return Code	Condition
Successful (1)	All of the selected payment records are processed successfully.
Warning (4)	N/A
Non-Fatal Error (8)	N/A
Failed (12)	The job fails under the following conditions: • Parameters are invalid • Run time exceptions for unexpected situations • SQL File name is not valid/written to the Parameter file. When this job ends with a return of code Failed, subsequent jobs in the chain will be set to inactive.
Terminated (16)	This return code will be issued when the job is terminated by the user. When this job ends with a return of code Terminated, subsequent jobs in the chain will be set to inactive.
System Failure (20)	This return code will be issued when the job is terminated because of database server or network issues. When this job ends with a return of code System Failure, subsequent jobs in the chain will be set to inactive.

Sort Sequence

N/A

Selection Criteria

N/A

Problem Resolution

Look into the job log for errors. Correct the problem and restart the job.

The following table shows the possible return codes and recommendations for each processing step.

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	This step does not issue this return code.	This step does not issue this return code.
Failed (12)	Failed because of runtime exceptions for an unexpected situation.	In this step the job can fail with fatal conditions only on encountering unknown exceptions. If that happens, investigate the exception reported by the process, resolve the error and restart the job.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can be restarted	
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can be restarted.	

2.1.20 EFT ACH File Reversal

Job Name	EFT ACH File Reversal
Recommended Frequency	This batch process can be run on demand.
Single Instance Required	No
Can be restarted?	Yes
Reports generated	No reports are generated. Forms are generated.

Description

The EFT ACH File Reversal batch process is used to correct the situation in which the entire ACH file with EFT payments was processed by the bank more than one time but the EFT payments were processed only once in Advantage. The process should be run only if the EFT transactions were sent to the bank erroneously more than one time since this job is not going to make any updates to the any transaction or table in the application.

Overview

This batch process can be used to reverse an entire ACH file with EFT payments or an entire ACH file with EFT Reversals.

The EFT ACH File Reversal batch process goes through the following steps:

Parameter Validation and Create EF ACH Reversal File

Process Steps	Messages
1. Parameter Validation	• Validating Batch Parameters • Parameters are valid or invalid depending on the validation. If the parameter is invalid, the invalid value is displayed in the log. • Batch Parameter validation completed
2. Create EF ACH Reversal File	• The input ACH file with EFT payments to be reversed is read sequentially where each record on this file is written to the output ACH file. • If the input file is either empty or not in the text format, the job ends with a job return code of Warning. • After posting each record on the input ACH file to the output ACH file with the exception of some fields, the Reversal file gets generated.

Steps for running this process:

1. Parameter Validation: First the process validates the batch parameters.
2. Create EF ACH Reversal File:

Once the parameter validation is successfully finished, the input ACH file is read sequentially and each record is written to the output ACH File. For the most part, each record on the input ACH file is posted as it is to the output ACH File with the exception of some fields (record type) listed below:

Set the counter (to search for Record type) to 1.

When counter = 1, copy all the Record type “1” records except File creation date (position 24-29) and File creation time (position 30-33) from the input ACH file and post it as the Record type “1” records on the Output ACH file.

- Populate the system run date as the File creation date (YYMMDD) and system run time as the File creation time(HHMM) to the output file.

- Increment the counter.
- When counter = 5, Copy all the Record type “5” records except
  - Service class code - position 02-04
  - Company Descriptive date – position 64-69 and
  - Effective Entry Date - position 70-75, from the input ACH file and post it as the Record type “5” records on the Output ACH file. Do the following changes to the output file.
    a. If the Service Class code is “200” (i.e. mixed Credits & Debits), then no changes.
    b. If the Service Class code is “220” (i.e. Credits only), then change it to “225” (i.e. Debits only).
    c. If the Service Class code is “225”, then change it to “220”.
    d. populate the system run date as the Company Descriptive date (YYMMDD)
    e. For Effective date parameter, Change it to the parameter Effective Date if not blank. Otherwise, change it to the System Run Date (YYMMDD).
- Increment the counter.
- When counter = 6, Copy all the Record type “6” records except
  - Transaction code - position 02-03
  - Discretionary Data – position 77-78, from the input ACH file and post it as the Record type “6” records on the Output ACH file. Do the following changes to the output file.
    a. If the Transaction code is “22”, then change it to “27”.
    b. If the Transaction code is “27”, then change it to “22”.
    c. If the Transaction code is “32“, then change to “37“.
    d. If the Transaction code is “37“, then change to “32“.
    e. If the Transaction code is “42“, then change to “47“.
    f. If the Transaction code is “47“, then change to “42“.
    g. If the **new** Transaction Code is set to “22”, “32” or “42”, then populate it from the parameter “Credit Detail Discretionary Data”.
    h. If the **new** Transaction Code is set to “27”, “37” or “47”, then populate it from the parameter “Debit Detail Discretionary Data”.

- Increment the counter.
- When counter = 8, Copy all the Record type “8” records except
  - Service class code - position 02-04
  - Total Debit Entry Dollar Amount – position 21-32 and
  - Total Credit Entry Dollar Amount - position 33-44, from the input ACH file and post it as the Record type “8” records on the Output ACH file.
- If the Service Class code is “200”, then no changes.
- If the Service Class code is “220”, then change it to “225”.
- If the Service Class code is “225”, then change it to “220”.
- Change the Total Credit Entry Dollar Amount as the Total Debit Entry Dollar Amount ($$$$$$$$$cc)
- Change the Total Debit Entry Dollar Amount as the Total Credit Entry Dollar Amount ($$$$$$$$$cc)
- Increment the counter.
- When counter=9, Copy all the Record type “9” records except
  - Total Debit Entry Dollar Amount in file– position 21-32 and
  - Total Credit Entry Dollar Amount in file - position 33-44, from the input ACH file and post it as the Record type “9” records on the Output ACH file.
    a. Change the Total Credit Entry Dollar Amount as the Total Debit Entry Dollar Amount ($$$$$$$$$cc)
    b. Change the Total Debit Entry Dollar Amount as the Total Credit Entry Dollar Amount ($$$$$$$$$cc)
- Save the output file.

When to Run

This process can be run on Demand.

Major Input

- EF ACH file

Major Output

- This process creates the EF ACH Reversal (.txt). The EF ACH Reversal (.txt) created by this process is named as per prefix (input ACH file) + “Reversal”.

Parameters

Job	Parameter	Description	Default Value
EF ACH File Reversal	ACH File Name	Required parameter. name of the ACH file with EFT payments to be reversed	No Default

		(fixed-length text file, 94 characters in length)	
	Effective Date	Optional parameter. If entered, the data must be a valid date and should be greater than or equal to the Application control date.	No Default
.	Credit Detail Discretionary Data	Required parameter. Entered value will be populated on the generated ACH Credit Detail record with the transaction code 22 or 32 or 42.	No Default
	Debit Detail Discretionary data	Required parameter. Entered value will be populated on the generated ACH Debit Detail record with the transaction code 27 or 37 or 47.	No Default

Sort Criteria

N/A

Selection Criteria

N/A

Problem Resolution

If the process fails due to any reason then:

- Look into log of job for errors.

The following table shows the various steps that the EF ACH File Reversal Process goes through and the messages issued at each step.

Step 1: Parameter Validation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the table validations are successful, that is, there are no errors raised during	N/A	N/A

	validations.		
Warning (4)	If the input ACH File is empty. **Sample Message:** " No records found on the input ACH File"	No further processing is required	
Non-Fatal Error (8)	N/A	This step does not issue this return code.	
Failed (12)	All of the required parameters are not entered. **Sample Message:** "Input ACH File is required" **Recommendation:** Enter thé input file	If the required parameters are not entered, enter the required parameters and restart the job.	
	Entered Parameters are not valid. **Sample Message**: "Credit Detail Discretionary Data parameter's length must be 2 characters" **Recommendation:** Enter a Credit Detail Discretionary Data of length 2.	If the parameters are invalid, enter the valid parameter and restart the job.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can either be restarted or a new job scheduled.	
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can either be restarted or a new job scheduled.	

2.1.21 Electronic Payment Request Chain Job

Chain or Job Name	Electronic Payment Request Chain Job
Recommended Frequency	The Electronic Payment Request chain can be run daily, monthly, quarterly or on an annual basis.
Single Instance Required	Yes.
Can be restarted?	Yes, see the individual jobs for more details.
Reports generated	Yes. The last job in the chain generates the exception report. Please refer to the individual jobs for more details.

Overview

The primary objective of the Electronic Payment Request Chain Job is to record Electronic Invoice details and process payment requests for Electronic Invoices of various service providers.

The Electronic Payment Request Chain Job has the following jobs (each of the jobs listed below, is described in subsequent sections):

1. Electronic Payment Request xml Generation
2. Electronic Payment Request Upload
3. Electronic Payment Request Submit
4. Electronic Payment Request Transaction Exception

The acceptable Pre Condition Return Code configuration depends on the business requirement. For example, if the requirement is that the subsequent jobs in the chain should continue only if the job ends with a Return Code of *Successful*, then the Pre Condition Return Codes for all of the jobs should be set to *Successful*. If for some jobs in the chain, a *Non-Fatal Error* is an acceptable job Return Code, then that can also be configured. These configurations can be done on the Job Setup (BATSETUP) page.

Major Input

Electronic Billing Inquiry (EBIT) table (R_UTLY_BILL_INQ)

Electronic Account Profile (EAPRO) table (R_UTLY_ACTG_PRFL)

Major Output

Electronic Payment Request (EPRC) transaction

Electronic Billing Inquiry (EBIT) table (R_UTLY_BILL_INQ)

Chain Job Return Code

The following table indicates the potential job Return Codes for the Electronic Payment Request Chain Job:

Return Code	Condition
Successful (1)	All of the jobs end successfully.
Warning (4)	One of the jobs in the chain ends with a Return Code of *Warning*.
Non-Fatal Error (8)	One of the jobs in the chain ends with a Return Code of *Non-Fatal Error*.
Failed (12)	One of the jobs in the chain ends with a Return Code of *Failed*.
Terminated (16)	One of the jobs in the chain ends with a Return Code of *Terminated*.
System Failure (20)	One of the jobs in the chain ends with a Return Code of *System Failure.*

Problem Resolution

If any of the jobs in the chain failed due to application errors it is advisable to reschedule the job after correcting the errors. Restarting the job reduces the processing time since the job resumes from where it last committed and selects only the unprocessed records.

Please refer to the individual jobs for details regarding the specific job processes and problem resolution.

Electronic Payment Request Chain: Electronic Payment Request xml Generation

Chain or Job Name	Electronic Payment Request xml Generation
Recommended Frequency	The Electronic Payment Request xml Generation job can be run daily, monthly, quarterly or on an annual basis.
Single Instance Required	Yes.
Can be restarted?	Yes.
Reports generated	Yes. The last job in the chain generates an exception report. Please refer to the individual jobs for more details.

Overview

This is the first job in the Electronic Payment Request Chain Job. It selects the records from the Electronic Billing Inquiry (EBIT) table that have the Payment Request Generated Date as null and produces an XML file containing output Electronic Payment Request transactions and updates the EBIT table with the generated Electronic Payment Request transaction information.

Major Input

Tables

- Electronic Billing Inquiry (EBIT) table (R_UTLY_BILL_INQ)

- Electronic Account Profile (EAPRO) table (R_UTLY_ACTG_PRFL)

Batch Parameters

Parameter	Description	Default Value
XML File	Required. This is the name of the xml output file that will be generated by the system and used as an input for the upload step.	UPRC1.xml
Load Parameter File (.txt)	Required. Indicates the name of the SMU load parameter file that will be created by this job step and later used as input to the Electronic Payment Request Upload job.	UPRCLoadParm.txt
Submit Parameter File (.txt)	Required. Indicates the file name of the SMU submit parameter file, which will be used as input to the Electronic Payment Request Submit job.	UPRCSubmitParm.txt
Exception Report File	Required. Indicates the file name and location of the exception report that will be generated by the Electronic Payment Request Submit job if any of the Electronic Payment Request transactions fail to submit.	UPRCExcep.txt
Parameter Location	Required. Indicates the location of the parameter files to be used in subsequent job steps within the chain.	$$AMSPARM$$
AMSEXPORT	Required. Indicates the file location on the server where the Electronic Payment Request Transaction xml file created by the Electronic Payment Request xml Generation step is to be stored.	$$AMSEXPORT$$

AMSLOGS	Required. Indicates the file location on the server where the exception logs created by the Electronic Payment Request Submit step are to be stored.	$$AMSLOGS$$
Transaction Code	Required. The Transaction Code of the Electronic Payment Request transactions to be generated.	EPRC
Transaction Department Code	Required. The Transaction Department used to create the Electronic Payment Request transactions. This value will be used if the Infer Transaction Dept and Transaction Unit from referenced Electronic Invoice (True or False)? parameter is set to False.	No default
Transaction Unit Code	Optional. The Transaction Unit used to create the Electronic Payment Request transactions. This value will be used if the Infer Transaction Dept and Transaction Unit from referenced Electronic Invoice (True or False)? parameter is set to False.	No default
Transaction Prefix	Optional. This will be the Prefix for the Transaction ID.	No default
Infer Transaction Dept and Transaction Unit from referenced Electronic Invoice (True or False)?	Required. If this value is set to False, the system sets the Electronic Payment Request Transaction Dept and Transaction Unit equal to the batch parameter values. If this value is set to True, the system infers the Department and Unit from the Electronic Invoice record.	True
Override Level	Optional. Override level used in the subsequent Submit job step.	No default

Apply Overrides (True/False)	Optional. This controls the Apply Overrides function while Submitting the transaction.	True
Commit Block Size	Required. Commit Block Size	1000
Progression Counter Size	Required. Progression Counter Size	100
Bypass Auto Number Flag	Required. Bypass Auto Transaction Numbering	True

Major Output

EPRC transaction xml file – An EPRC transaction xml output file will be generated by the system and used as an input for the upload step.

Job Return code

The following table shows the potential job return codes for the Electronic Payment Request xml Generation job.

Return Code	Condition
Successful (1)	Parameter validated successfully.
Warning (4)	No eligible records found. This could be because of the following reason: • No eligible records selected from the EBIT table for processing.
Non-Fatal Error (8)	N/A
Failed (12)	The job will fail under the following conditions: • Parameters are invalid. • Run time exceptions for unexpected situations. When this job ends with a Return Code of *Failed*, subsequent jobs in the chain will be set to *inactive*.
Terminated (16)	This Return Code is issued when the job is terminated by the user. When this job ends with a return code of *Terminated*, subsequent jobs in the chain are set to inactive.
System Failure (20)	This Return Code is issued when the job is terminated because of database server or network issues. When this job ends with a Return Code of *System Failure*, subsequent jobs in the chain are set to inactive.

Sort Sequence

- The selected records are sorted by IE Transaction Code, IE Transaction Dept, IE Transaction ID, IE Transaction VL, and IE Transaction CL.

Selection Criteria

- It selects the records that have the Payment Request Generated Date as null on the Electronic Billing Inquiry (EBIT) table.

Problem Resolution

The following table shows the possible Return Codes and recommendations for each processing step.

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	No eligible records selected from the EBIT Table for processing.	Verify whether the Electronic Billing Inquiry (EBIT) table contains any record with the Payment Request Generated Date as null.	N/A
Non-Fatal Error (8)	N/A	This step does not issue this Return Code.	N/A
Failed (12)	Required Parameters are not entered. **Sample Message:** EPRC Transaction Code cannot be blank.	Enter the EPRC Transaction Code and schedule a new job.	Job should be rescheduled after correcting the parameters.
	Entered Parameters are not valid. **Sample Message**: Apply Override value is invalid. Must be either *True* or *False*.	Enter the correct Apply Override value and schedule a new job.	
	DEPT_CD not found. **Sample Message**: Entered Department Code is not found on R_DEPT.	Make sure that the Transaction Department is valid and available on the DEPT table and schedule a new job.	
	Runtime Exception occurred.	The reason for the Runtime Exception needs to be investigated. Correct the	The System error log and VLS log should be investigated to find

		problem and schedule a new job.	out the possible reason for the exception.
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. Schedule a new job.	Job should be rescheduled after correcting the problem.
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. Schedule a new job.	The job should be rescheduled after correcting the problem.

Electronic Payment Request Chain: Electronic Payment Request Upload

Job Name	Electronic Payment Request Upload
Recommended Frequency	This job can be run daily, monthly, quarterly or on an annual basis.
Parallel processing enabled	No. Parallel processing is not supported for this job.
Can the job be restarted?	No.
Exception report produced	No. All of the exceptions are only written to the log.

Overview

The Electronic Payment Request (EPRC) Upload job loads the records from the XML File, generated by the Electronic Payment Request xml Generation job, into the Transaction Catalog. This job uses the common utility to load the records into the Transaction Catalog. This job first validates the batch parameters. If the parameters are valid, then it loads the records into the Transaction Catalog. If the parameters are not valid, the job issues appropriate messages and ends with a status of *Failed*. Once the records are loaded into the Transaction Catalog, the summary information is written into the log as how many records were in the input file and how many records loaded successfully.

Major Input

- EPRC Transaction XML file

Batch Parameters

Parameter	Description	Default Value
Parameter file (PARM_FILE)	Parameter file to Load Transactions	$$AMSPARM$$/UPRCLoadParm.txt

Note: PARM_FILE only contains the following subset of SMU parameters.

Parameter	Default Value
Action Code (ACTN_CD)	DOCIMPORT
File Name To Be Imported (FILE_NM)	$$AMSEXPORT$$/UPRC1.xml
Transaction Status code (DOC_STA_CD)	2
Apply Overrides (APPLY_OVERRIDES)	True
Override Level (OVERRIDE_LVL)	5
Commit Block Size (COMMIT_BLOCK)	1000
By Pass Auto Transaction Number (BYPS_ADNT_FL)	True

Please refer to the "SMU Transaction Upload Job" run sheet in the *CGI Advantage Financial – Utilities Run Sheets* guide for the full list of SMU Transaction Upload batch parameters.

Major Output

- EPRC transactions with a Phase of *Draft*.

Batch Return Codes

The following table shows the potential job Return Codes for the EPRC Upload job.

Return Code	Condition
Successful (1)	All of the records are loaded into the Transaction Catalog successfully or the input file is empty.
Warning (4)	This Return Code is issued when some of the records failed to load where as all other records were loaded successfully.
Non-Fatal Error (8)	None of the records get loaded into the Transaction Catalog.
Failed (12)	• Parameters are invalid. • The input file is not found in the specified directory. • Restart failed because another instance of the Electronic Payment Request chain has already been run successfully. • Runtime exceptions encountered for any unexpected situations. When the job ends with a Return Code of *Failed*, subsequent jobs in the chain are set to *Inactive*.
Terminated (16)	This Return Code is issued when the job is terminated by the user. When the job ends with a Return Code of *Terminated*, subsequent jobs in the chain are set to inactive.

System Failure (20)	This Return Code is issued when the job is terminated because of database server or network issues. When this job ends with a Return Code of *System Failure*, subsequent jobs in the chain are set to *Inactive*.

Sort Sequence

N/A

Selection Criteria

N/A

Problem resolution

The following table shows the possible Return Codes and recommendations for each processing step specific to the job in the chain. For general errors and recommendations, refer to the "SMU Transaction Upload Job" run sheet in the *CGI Advantage Financial – Utilities Run Sheets* guide.

Possible Return Codes	Condition	Recommendation	Other Instructions
Failed (12)	Failed while restarting the job since another instance of the job has already been run successfully. **Sample Message**: Cannot restart the job since another instance of this job has already been run successfully.	**Recommendation:** Schedule a new job.	

Electronic Payment Request Chain: Electronic Payment Request Submit

Job Name	Electronic Payment Request Submit
Recommended Frequency	Daily, monthly, quarterly or annual basis.
Single Instance Required	Yes
Can be restarted?	Yes. See the "Problem Resolution" sections for more details.
Report Generated	No. All of the exceptions are written to the error file.

Overview

This job submits the transactions listed in the input parameter file that were generated by the Electronic Payment Request Upload job.

Major Input

- SMU job parameter file
- Draft EPRC transactions on the Transaction Catalog

Batch Parameters

Parameter	Description	Default Value
Submit Parameter File (SUBMIT_FILE)	Submit Parameter File	$$AMSPARM$$/UPRCSubmitParm.txt

Note: SUBMIT_FILE only contains the following subset of SMU parameters.

Parameter	Default Value
Action Code (ACTN_CD)	DOCSUBMIT
Exception Report Ind (EXCEP_REP_IND)	4
File Name To Be Imported (FILE_NM)	$$AMSLOG$$/UPRCExcep.txt
Commit Block Size (COMMIT_BLOCK)	1000
Submit Progression Counter Size (DOCSUB_PROG_CTR_SZ)	1000
Generate Status (GENERATE_STATS)	True
By Pass Auto Transaction Number (BYPS_ADNT_FL)	True

This job uses only a subset of the SMU Submit job parameters. For a full list of available parameters for the SMU submit job, refer to the “SMU Transaction Submit Job” run sheet in the *CGI Advantage Financial – Utilities Run Sheets* guide.

Major Output

- The transactions are processed to *Final* or *Rejected*.

Batch Return codes

The following table shows the potential job Return Codes for the Electronic Payment Request Submit job in the Electronic Payment Request Chain.

Return Code	Condition
Successful (1)	All of the transactions generated in that run submitted successfully.
Warning (4)	Not Applicable for this job.

Non-Fatal Error (8)	Not Applicable for this job.
Failed (12)	• Input parameter file is not found. • Restart failed because another instance of the Electronic Payment Request Chain has already been run successfully. • Runtime exceptions encountered for any unexpected situations. When the job ends with a Return Code of *Failed*, subsequent jobs in the chain will be set to *Inactive*.
Terminated (16)	This Return Code is issued when the job is terminated by the user. When the job ends with a Return Code of *Terminated*, subsequent jobs in the chain are set to *Inactive*.
System Failure (20)	This Return Code is issued when the job is terminated because of database server or network issues. When this job ends with a Return Code of *System Failure*, subsequent jobs in the chain are set to *Inactive*.

Sort Sequence

N/A

Selection Criteria

N/A

Problem Resolution

If the job ends with a Return Code of *Failed* and above, the job can be restarted only when the Save Restart Information parameter is selected on the System Maintenance Utility Parameters page and another instance of the job has not been scheduled and run successfully. If another instance of the job has already been scheduled and run successfully, then this job should not be restarted – only a new job should be scheduled.

If the restart is not an immediate option and the fatal error is because of a few transactions, the rest of the transactions can be submitted manually or discarded manually depending on the issue. The Transaction IDs can be found on the input parameter file.

The following table shows the possible Return Codes and recommendations for each processing step specific to the job in the chain. For general errors and recommendations, refer to the “SMU Transaction Submit Job” run sheet in the *CGI Advantage Financial – Utilities Run Sheets* guide.

Possible Return Codes	Condition	Recommendation	Other Instructions
Failed (12)	Failed while restarting the job since another instance of the job has already been run successfully. **Sample Message**: Cannot restart the job since another instance of	Schedule a new job.	

	this job has already been run successfully.		

Electronic Payment Request Chain: Electronic Payment Request Transaction Exception

Job Name	Electronic Payment Request Transaction Exception
Recommended Frequency	Daily, monthly, quarterly or annual basis
Single Instance Required	Yes
Can be restarted?	No
Report Generated	Yes. The job generates the reports in pdf and html formats.

Overview

This job in the Electronic Payment Request Chain generates an exception report that lists all of the errors encountered when the EPRC transactions were submitted in the earlier step. The report contains the following information:

Rejected EPRC transaction

Detailed error description along with the error code

Major Input

Draft EPRC transactions on the Transaction Catalog with a Rejected Status.

Batch Parameters

Parameter	Description	Default Value
AMSLOGS	Required. Logs Location at EPRC Transaction Exception Job	$$AMSLOGS$$
CLIENT_NM	Optional. Name of the Client should display in the Report	No default
EXP_FILE_NM	Required. Name of the Flat file for output of exception report	UPRCExcep.txt

Major Output

EPRC Transaction Exception Report

Batch Return Codes

The following table shows the potential job Return Codes for the Electronic Payment Request Transaction Exception job in the Electronic Payment Request chain.

Return Code	Condition
Successful (1)	All of the transactions generated in that run submitted successfully.
Warning (4)	Not Applicable for this job.
Non-Fatal Error (8)	Not Applicable for this job.
Failed (12)	• Input file is blank. • Input parameter file is not found in the specified folder. • Restart failed because another instance of the Electronic Payment Request chain has already been run successfully. • Runtime exceptions encountered for any unexpected situations. When the job ends with a Return Code of *Failed*, subsequent jobs in the chain are set to *Inactive*.
Terminated (16)	This Return Code is issued when the job is terminated by the user. When the job ends with a Return Code of *Terminated*, subsequent jobs in the chain are set to *Inactive*.
System Failure (20)	This Return Code is issued when the job is terminated because of database server or network issues. When this job ends with a Return Code of *System Failure*, subsequent jobs in the chain are set to *Inactive*.

Sort Sequence

N/A

Selection Criteria

N/A

Problem resolution

The following table shows the possible Return Codes and recommendations for each processing step specific to the job in the chain.

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	N/A	This step does not issue this return code.	N/A
Non-Fatal Error (8)	N/A	This step does not issue this return code.	N/A

Failed (12)	Runtime Exception occurred. **Sample Message**: Exception in genUPRCExcepRep: <Runtime Exception Message>	If the job fails with fatal conditions on encountering unknown exceptions, then investigate the exception reported by the process, resolve the error and schedule a new job.	System error log and VLS log should be investigated to determine the possible reason for the exception.
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can either be restarted or rescheduled.	Job should be rescheduled after correcting the problem.
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can either be restarted or rescheduled.	Job should be rescheduled after correcting the problem.

2.1.22 EWS Transaction

Chain or Job Name	EWS Transaction
Recommended Frequency	Daily as part of the nightly cycle or on demand
Single Instance Required	Yes
Can be restarted?	No
Reports generated	Yes

Overview

EWS Transaction is a chain job that supports the functionality in Advantage Financial to interface to the Early Warning System (EWS) used by certain banks, such as, J.P. Morgan Chase. This functionality works similar to the Prenote functionality insofar as EFT information must be verified through EWS prior to a vendor becoming eligible for EFT. This chain job generates an outbound file that contains the account information for any vendors (or their address records) with *EWS Requested* in the EFT Status field of the Vendor/Customer or address record. The outbound file is sent to the bank, where the information is analyzed before sending back a reply file with any updates. Inbound files are later processed through the EWS Transaction chain job to update the Vendor/Customer or address records by changing the EFT Status field value and Generate EFT Payment flag, as needed.

The chain job is comprised of the following three batch jobs:

1. EWS Reply File
2. EWS Reports
3. Create EWS File

Major Input

Vendor/Customer table (R_VEND_CUST)

Address table (R_AD)

Prenotes in Progress table (PRENOTE_PRG_VW)

EWS Status Mapping table (R_EWS_EFT_STA)

System Options table

CSV files sent by the bank

Major Output

Vendor/Customer table (R_VEND_CUST)

Address table (R_AD)

Outbound file sent to bank

Chain / Job Return Code

The following table shows the potential Job Return Codes for the EWS Transaction chain:

Return Code	Condition
Successful (1)	All of the jobs end successfully.
Warning (4)	One of the jobs in the chain ends with a Return Code of *Warning*.
Non-Fatal Error (8)	One of the jobs in the chain ends with a Return Code of *Non-Fatal Error*.
Failed (12)	One of the jobs in the chain ends with a Return Code of *Failed*.
Terminated (16)	One of the jobs in the chain ends with a Return Code of *Terminated*.
System Failure (20)	One of the jobs in the chain ends with a Return Code of *System Failure*.

EWS Transaction: EWS Reply File

Job Name	EWS Reply File
Recommended Frequency	Daily as part of the nightly cycle or on demand.
Single Instance Required	Yes
Can be restarted?	No
Reports Generated	No

Overview

There are two separate reply files sent by the bank. The first reply file is used to determine if the file sent to the bank is in the correct format. For successfully formatted records, the second reply file includes the Account Status (AATCHEK) as well as the Account Ownership Status (AATAOA) values provided by the bank and that will be used to determine how to update the vendor's EFT Status and Generate EFT Payment flag.

The steps involved in this process are:

1. **Parameter Validation**: This step validates the batch parameters.
2. **Outbound file validation**: This step validates the first reply file, which indicates if the file sent to the bank is in the correct format.
3. **Update EFT Status**: This step extracts the Account Status and Account Ownership Status data from the second reply file. Based on those values, the job evaluates EWS Status Mapping (EWSSTA) to determine how to update the Vendor/Customer record's or address record's EFT Status and Prenote Return Reason fields. If EWS Status is changed to 'Eligible for EFT', the Generate EFT Payment flag will also be set to true. Any

Vendor/Customer or address records that have been in an 'EWS Pending' status for more days than allowed by the Number of EWS Days value defined in System Options (SOPT) will have EFT Status changed to 'Manual Review Needed'.

The following table shows the various steps that the EWS Reply File job goes through and the messages issued at each step:

Process Steps	Messages
1. Parameter Validation	If a parameter is invalid, a message indicating the error will be displayed in the log.
2. Outbound file validation	The first file (.out) will be used to determine if the file sent to the bank is in the correct format. The second file (.csv) is used to determine which Vendor/Customer and/or address records to update. • Reading Bank Files • Found 2 files to process. • Processing File: <*name of .out file*> • All record formats were valid. • End of File. • Processing File: <*name of .csv file*> • Processing Record: VEND_CUST_CD: [VS00009489] AD_ID: [null] AD_TYP: [null] with EWS Code of [EWS01] mapped to EFT Status: [3] • Account Status: [OPEN / VALID], Account Ownership Status: [AOA MATCH] • End of File
3. Update EFT Status	Based on AATCHEK and AATAOA values in the .csv file, update the EFT Status of the Vendor/Customer and/or Address records. • Updated 1 Records. • Retrieving Records that are to be set to Manual Review Needed based on Number of EWS Days • 1 Vendor Records set to Manual Review Needed.

Restartability

The job cannot be restarted. If the job fails in any of the above steps, a new job should be scheduled after correcting the errors that caused the job to fail.

Major Input

Two files (sent by the bank)

Job Parameters

Parameter	Description	Default Value
AMSIMPORT	Required and overridable This is the location where the job will search for the input file	$$AMSIMPORT$$
COMMIT_SIZE	Commit Block Size. This is optional.	100
SELECT_BLOCK	This is the block size that is used to select the records to process at a time	1000

Major Output

Updates to Vendor/Customer (R_VEND_CUST) and Address (R_AD) tables.

Job Return Code

The following table shows the possible Job Return Codes for the EWS Reply File job:

Return Code	Condition
Successful (1)	The input file has been processed successfully and no error was reported.
Warning (4)	The job gives a warning under the following conditions: • The Import folder doesn't have bank files to process • The input files have invalid extensions other than '.out' and '.csv'
Non-Fatal Error (8)	The job gives a Non-Fatal Error under the following conditions: • If the job is unable to commit the record • If the Vendor/Customer code is empty
Failed (12)	The job fails under the following conditions: • Parameters are invalid. • Files having `Unexpected File format` • If there is an issue while reading the CSV file When this job ends with a Return Code of *Failed*, subsequent jobs in the chain are set to *Inactive*.
Terminated (16)	This return code is issued when the job is terminated by the user. When this job ends with a Return Code of *Terminated*, subsequent jobs in the chain are set to *Inactive*.

System Failure (20)	This return code is issued when the job is terminated because of the database server or network issues. When this job ends with a Return Code of *System Failure*, subsequent jobs in the chain are set to *Inactive*.

Sort Criteria

N/A

Selection Criteria

N/A

Problem Resolution

If the job ends with a return code of *Failed* and above, the job can be rescheduled.

The following table shows the various steps that the EWS Reply File job performs and the messages issued at each step:

Step 1: Parameter Validation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the parameter validations are successful.	N/A	N/A
Warning (4)	N/A	This step does not issue this return code.	N/A
Non-Fatal Error (8)	N/A	This step does not issue this return code.	N/A
Failed (12)	All of the required parameters are not entered. **Sample Message:** "Import Location is required" **Recommendation:** Enter the AMSIMPORT parameter.	If the required parameters are not entered, enter the required parameters and reschedule the job.	N/A
Terminated (16)	The job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can be rescheduled.	N/A
System Failure (20)	The job is terminated because of the database	The reason for the System Failure needs to be	N/A

	server or network issues.	investigated. The job can be rescheduled.	

Step 2: Outbound file validation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	The file (.out) has been processed successfully and no error was reported.	N/A	N/A
Warning (4)	There are no files in the input folder.	In the input folder, add the files to be processed.	N/A
Non-Fatal Error (8)	The job is unable to commit the record. The Vendor/Customer code is empty.	Verify the Vendor/Customer record does not have any errors. The Vendor/Customer code must not be empty in the file.	N/A
Failed (12)	The file has an unexpected file format	The file must have the extension '.out'	N/A
Terminated (16)	The job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can be rescheduled.	N/A
System Failure (20)	The job is terminated because of the database server or network issues.	The reason for the System Failure needs to be investigated. The job can be rescheduled.	N/A

Step 3: Update EFT Status

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	The input file (.csv) has been processed successfully and updated the EFT status of the Vendor/Customer record.	N/A	N/A

Warning (4)	There are no files in the input folder.	In the input, folder have to add files to process	N/A
Non-Fatal Error (8)	The job is unable to commit the record. The Vendor/Customer code is empty.	Verify the Vendor/Customer record does not have any errors. The Vendor/Customer code must not be empty.	N/A
Failed (12)	If the file has an unexpected file format.	The file must have the extension '.csv'	N/A
Terminated (16)	The job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can be rescheduled.	N/A
System Failure (20)	The job is terminated because of the database server or network issues.	The reason for the System Failure needs to be investigated. The job can be rescheduled.	N/A

EWS Transaction: EWS Reports

Job Name	EWS Reports
Recommended Frequency	Daily as part of the nightly cycle or on demand.
Single Instance Required	Yes
Can be restarted?	No
Reports Generated	Yes

Overview

This EWS Reports job in the EWS Transaction chain can create three EWS reports that list the records to be included in the file that will be created in the chain job's third step (Create EWS File) or that show any EWS exceptions. The reports are:

Selection Report shows Vendor/Customer and Address records where the EFT Status is 'EWS Requested' and the EFT Format is 'CCD', 'PPD', or 'CTX'.

Invalid EFT Format Report shows Vendor/Customer and Address records where the EFT Status is 'EWS Requested' and the EFT Format is **not** 'CCD', 'PPD', or 'CTX'.

Exception Report shows Vendor/Customer and Address records where the EFT Status is 'EWS Rejected' or 'Manual Review Needed'.

The steps involved in this process are:

1. **Parameter Validation**: First the process validates the batch parameters.
2. **Generation of EWS Reports:** This job creates three reports Selection Report, Invalid EFT Format Report, and Exception Report.

The following table shows the various steps that the EWS Reply File job goes through and the messages issued at each step:

Process Steps	Messages
1. Parameter Validation	If a parameter is invalid, a message indicating the error will be displayed in the log.
2. Generation of EWS Reports	For each of the three reports: • Rendering report started • Rendering report completed. • Reports output folder mapped • HTML report file path: <insert path> • PDF report file path: <insert path>

Restartability

The job cannot be restarted. If the job fails in any of the above steps, a new job should be scheduled after correcting the errors that caused the job to fail.

Major Input

Vendor/Customer table (R_VEND_CUST)

Address table (R_AD)

Job Parameters

Parameter	Description	Default Value
CLIENT_NM	Client name for Report	No default
MAX_FLUSH_ COUNTER	Required. Max. Number of records to be processed before Flush	100
PROG_CTR_SZ	Progression Counter Size	5000
REPORT_ID	Report ID	No default
SELECT_BLOCK	This is the block size that is used to select the records to process at a time	1000

Major Output

Selection Report, Invalid EFT Format Report, Exception Report

Job Return Code

The following table shows the possible Job Return Codes for the EWS Reports job:

Return Code	Condition
Successful (1)	The reports have been created successfully.
Warning (4)	This job does not issue this Return Code.
Non-Fatal Error (8)	This job does not issue this Return Code
Failed (12)	The job fails if the Parameters are invalid.
Terminated (16)	This return code is issued when the job is terminated by the user. When this job ends with a Return Code of *Terminated*, subsequent jobs in the chain are set to *Inactive*.
System Failure (20)	This return code is issued when the job is terminated because of the database server or network issues. When this job ends with a Return Code of *System Failure*, subsequent jobs in the chain are set to *Inactive*.

Sort Criteria

N/A

Selection Criteria

N/A

Problem Resolution

If the job ends with a return code of *Failed* and above, the job can be rescheduled.

The following table shows the various steps that the EWS Reports job goes through and the messages issued at each step:

Step 1: Parameter Validation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the parameter validations are successful.	N/A	N/A
Warning (4)	N/A	This step does not issue this return code.	N/A

Non-Fatal Error (8)	N/A	This step does not issue this return code.	N/A
Failed (12)	All of the required parameters are not entered. **Sample Message:** "Select block size should be greater than zero" **Recommendation:** Enter Select block size as a positive integer	If the required parameters are not entered, enter the required parameters and reschedule the job.	N/A
Terminated (16)	The job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can be rescheduled.	N/A
System Failure (20)	The job is terminated because of the database server or network issues.	The reason for the System Failure needs to be investigated. The job can be rescheduled.	N/A

EWS Transaction: Create EWS File

Job Name	Create EWS File
Recommended Frequency	Daily as part of the nightly cycle or on demand.
Single Instance Required	Yes
Can be restarted?	No
Reports Generated	Yes

Overview

This Create EWS File job in the EWS Transaction chain generates a flat file in EWS format to send to the bank to evaluate those vendors with EFT Status of "EWS Requested". Once the file is generated, the job also updates the Vendor/Customer and Address record's EFT Status.

The steps involved in this process are:

1. **Parameter Validation**: First the process validates the batch parameters.
2. **Create Outbound File**: This step reads the EFT Status from the Vendor/Customer and Address tables, creates the outbound file for vendors with EWS Requested status, and then updates the EFT Status to 'EWS Pending'.

The following table shows the various steps that the Create EWS File job goes through and the messages issued at each step:

Process Steps	Messages
1. Parameter Validation	If a parameter is invalid, a message indicating the error will be displayed in the log.
2. Create Outbound File	If records are found: • Retrieving EFT Requested Records. • Writing to file: <insert .in file name> • Including Record to outbound File: VEND_CUST_CD:[*insert code*] AD_ID:[-] AD_TYP:[-] • Including Record to outbound File: VEND_CUST_CD:[*insert code*] AD_ID:[*insert code*] AD_TYP:[*insert code*] • Added <*insert number*> records to outbound file. If no records found: • No records found to process.

Restartability

The job cannot be restarted. If the job fails in any of the above steps, a new job should be scheduled after correcting the errors that caused the job to fail.

Major Input

Vendor/Customer table (R_VEND_CUST)

Address table (R_AD)

Job Parameters

Parameter	Description	Default Value
AMSEXPORT	Required. Export Location for Create EWS File Job	$$AMSROOT$$
CLIENT_PROGRAM_ID	Required. EWS Client Program ID	No default
INCLUDE_ADDR_FL	Required. Include Address Records in a file.	false
PROG_CTR_SZ	Progression Counter Size	5000
SELECT_BLOCK	This is the block size that is used to select the records to process at a time	1000

Major Output

Outbound file sent to bank

Updates to Vendor/Customer (R_VEND_CUST) and Address (R_AD) tables.

Job Return Code

The following table shows the possible job Return Codes for the EWS Reply File job:

Return Code	Condition
Successful (1)	The outbound file has been created successfully Updates EFT Status to 'EWS Pending' successfully
Warning (4)	If there are no records with EFT Status of 'EWS Requested'.
Non-Fatal Error (8)	This job does not issue this return code.
Failed (12)	The job fails if the parameters are invalid. The job fails if not able to commit the record.
Terminated (16)	This return code is issued when the job is terminated by the user. When this job ends with a Return Code of *Terminated*, subsequent jobs in the chain are set to *Inactive*.
System Failure (20)	This return code is issued when the job is terminated because of the database server or network issues. When this job ends with a Return Code of *System Failure*, subsequent jobs in the chain are set to *Inactive*.

Sort Criteria

N/A

Selection Criteria

N/A

Problem Resolution

If the job ends with a return code of *Failed* and above, the job can be rescheduled.

The following table shows the various steps that the Create EWS File job goes through and the messages issued at each step:

Step 1: Parameter Validation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the parameter validations are successful.	N/A	N/A
Warning (4)	N/A	This step does not issue this return code.	N/A
Non-Fatal Error (8)	N/A	This step does not issue this return code.	N/A
Failed (12)	All of the required parameters are not entered. **Sample Message:** "Export Location is required" **Recommendation:** Enter AMSEXPORT parameter.	If the required parameters are not entered, enter the required parameters and reschedule the job.	N/A
Terminated (16)	The job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can be rescheduled.	N/A
System Failure (20)	The job is terminated because of the database server or network issues.	The reason for the System Failure needs to be investigated. The job can be rescheduled.	N/A

Step 2: Create Outbound File

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	The Outbound file has been created successfully. Updates EFT status to 'EWS Pending' successfully.	N/A	N/A
Warning (4)	If there are no records with EFT Status of 'EWS Requested'.	Verify if any Vendor/Customer or address records with EFT Status of 'EWS Requested' exists.	N/A
Non-Fatal Error (8)	N/A	This step does not issue this return code.	N/A

Failed (12)	The job fails if the parameters are invalid. The job fails if not able to commit the record.	If the required parameter values are invalid, enter valid values and reschedule the job. Verify the Vendor/Customer or address record does not have any errors.	N/A
Terminated (16)	The job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can be rescheduled.	N/A
System Failure (20)	The job is terminated because of the database server or network issues.	The reason for the System Failure needs to be investigated. The job can be rescheduled.	N/A

2.1.23 Extended Payment Request Scheduling Batch Process

Chain or Job Name	Extended Payment Request Scheduling Batch Process
Recommended Frequency	Daily as part of the nightly cycle or on demand. This batch job is run after Automated Disbursement Batch Job and before AD Chain.
Single Instance Required	Yes.
Can be restarted?	Yes.
Reports generated	N/A

Overview

The Extended Payment Request Scheduling Batch Process sets and calculates the Schedule Payment Date, Disbursement Priority and Scheduling Reason code on Disbursement Request (DISRQ) table (R_AP_DISB_RQST). The process selects the records from the DISRQ records on certain criteria. Depending upon the input Schedule Reason Codes parameters, the payment lags from the reference tables R_DEPT_OBJ_PYCTRL, R_DISB_CSH_MGMT, R_PSIC, SOPT table are added to certain dates on the Disbursement Request table to calculate Scheduled Payment Date. If Immediate Payment Department is mentioned in the input parameters, then the process sets the Scheduled Payment Dates on the Disbursement Request table to the current application date for departments that should be immediately scheduled. The process calculates the Scheduled Payment Date for the selected records from the R_AP_DISB_RQST table for the departments that are not mentioned in the Immediate Payment Department input parameter. The process assigns Scheduling Reason Code and Disbursement Priority based on the Assigned Scheduled Payment Date. This batch process is restartable.

The Job return status is set to *Successful* if the disbursement transactions are successfully processed and is set to *Failed* if an exception is encountered.

The following table shows the various steps that the Job goes through and the messages issued at each step.

Process Steps	Messages
1. Parameter Validation	• Validating Batch Parameters • Parameters are valid or invalid depending on the Validation. If the parameter is invalid, the invalid value will be displayed in the log. • Batch Parameter validation completed.
2. Selection of Records	• Selecting eligible records from DISRQ for Immediate Payment Departments. • Selecting eligible records from DISRQ for other departments not mentioned in Immediate Payment

Process Steps	Messages
	Departments. • If the selection returns 0 records, then the following message will be issued: No Records found to process Other Payment Department. • Number of records (count) processed will be displayed after every commit.

Restartability Information

If the job fails in any of the above steps the job can be restarted after resolving the error. If the job is restarted, it starts from the step where it failed earlier and does not start from the beginning. For example, the job failed in the step Immediate Payment Department due to some fatal condition (table space error) and if the job is restarted after resolving the table space issue, the job will start from the step Immediate Payment Department and not from the parameter validation step. Instead of restarting the job, if the job is rescheduled with the same set of parameters after resolving the table space issue, the job will start from the beginning.

Major Input Tables

- Disbursement Request (R_AP_DISB_RQST)
- Disbursement Cash Management Options (R_DISB_CSH_MGMT)
- Payment Scheduling and Interest Control (R_PSIC)
- Scheduling Reason Code (R_SCHDL_REAS)
  Department/Object Payment Control (R_DEPT_OBJ_PYCTRL)
  SOPT (R_EXP_SOPT)

Batch Parameters

Parameter	Description	Default Value
Immediate Payment Departments	This is an optional parameter. This parameter is used to specify Immediate Payment Departments for which Immediate Payment Schedule Reason code is applied.	No Default
Immediate Payment Scheduling Reason Code	This is an optional parameter. This parameter is used to specify Immediate Payment Schedule Reason code that is applied to the records for the departments from the Immediate Payment Schedule Departments parameter.	No Default

Global Payment Scheduling Reason Code	This is an optional parameter. This parameter is used to specify Global Payment Scheduling Reason Code.	No Default
Minority/Women Owned Business Payment Scheduling Reason Code	This is an optional parameter. This parameter is used to specify Minority/Women Owned Business Payment Scheduling Reason Code.	No Default
Government Payment Scheduling Reason Code	This is an optional parameter. This parameter is used to specify Government Payment Scheduling Reason Code.	No Default
Hold Removal Minimum Age Scheduling Reason Code	This is an optional parameter. This parameter is used to specify Hold Removal Minimum Age Scheduling Reason Code	No Default
Interest Eligibility Payment Scheduling Reason Code	This is an optional parameter. This parameter is used to specify Interest Eligibility Payment Scheduling Reason Code	No Default
Payment Type Scheduling Reason Code	This is an optional parameter. This parameter is used to specify Payment Type Scheduling Reason Code	No Default
Department/Object Payment Scheduling Reason Code	This is an optional parameter. This parameter is used to specify Department/Object Payment Scheduling Reason Code	No Default
Discount Payment Scheduling Reason Code	This is an optional parameter. This parameter is used to specify Discount Payment Scheduling Reason Code	No Default
Reassign All Schedule Dates	This is an optional parameter. This parameter is used to specify the value of *Y* or *N*. If value is *Y*, then the record from the Disbursement will be selected where schedule code is not manual, if value is *N* on the parameter, then the record from the Disbursement will be selected where schedule code is null	No Default
Select Block Size	This is the block size that is used to select the records to process	No Default

	at a time.	
Commit Block Size	This is used to commit the data as a time.	No Default
Progression counter size	Will give the counter of the progress of the process.	No Default
Pre-Processing Payment Scheduling Reason Code	Indicates the scheduling Reason Code to set for Pre-Processing Payment Request transactions.	No Default
Pre-Processing Payment Request Transaction	Indicates the Transaction Code(s) of the Pre-Processing Payment Request transactions.	No Default

Output

Scheduled Payment Date, Scheduling Reason Code and Disbursement Priority is updated on Disbursement Request (R_AP_DISB_RQST).

Job Return code

The following table shows the potential job return codes for the job.

Return Code	Condition
Successful (1)	All the selected payment records are processed successfully
Warning (4)	N/A
Non-Fatal Error (8)	N/A
Failed (12)	The job fails under the following conditions: • Parameters are invalid • Unable to find the SOPT or DCMOPT record for the Fiscal year associated with the DISRQ. • Run time exceptions for unexpected situations.
Terminated (16)	This return code will be issued when the job is terminated by the user.
System Failure (20)	This return code will be issued when the job is terminated because of database server or network issues. When this job ends with a return of code System Failure.

Sort Sequence

Transaction Code, Transaction Department Code, Transaction ID, Transaction Vendor Line number, Transaction Commodity Line number, Transaction Accounting Line number.

Selection Criteria

- If the Transaction code on the DISRQ matches any of the Pre-Processing Payment Request Transaction batch parameter.
- If Immediate Payment Department are mentioned in the input parameters, then the process will set the Scheduled Payment Dates on Disbursement Request to the current application date for Departments that should be immediately scheduled.
- When 'Reassign All Schedule Dates' parameter is set to Yes, the job selects the records from Disbursement Request (DISRQ) table as follows:
    - The Current Hold Type is not blank.
    - The Scheduling Reason Code is defined as a "Manually Scheduled" Type on the Scheduling Reason Code table or is blank.
    - The Transaction is an IPR transaction.

- When 'Reassign All Schedule Dates' is set to No, the job selects records that meet both of the following criteria:
    - The Current Hold Type is blank.
    - The Scheduling Reason Code is blank.
    - The Transaction is not an IPR transaction.

Problem Resolution

If the job fails due to any reason, the job can be restarted.

2.1.24 Forms Table Load Chain

Chain or Job Name	Forms Table Load Chain
Recommended Frequency	Nightly or on demand.
Single Instance Required	Yes
Can be restarted?	Yes
Reports generated	Yes • Tax Form Reporting Detail Table Load Exception Report • Forms Reporting Table Load Processing Report

Overview

The Forms Table Load Chain has the following jobs:

1. 1099 Flat Files to XML
2. Load XML Files to Detail Tables
3. Load Forms Reporting Tables

Note: Even though the above jobs in the chain can be run individually by disabling other jobs, it is recommended to always run the entire chain.

The Forms Table Load Chain in CGI Advantage Financial is a group of jobs that work together to load the payee, payer, and tax form data into the Forms Reporting Detail tables and Form Reporting tables from the flat files generated by the Offline 1099 batch process.

The first process in the chain converts the flat files generated by the Offline 1099 process into an XML format to be used by subsequent process in chain. After converting the flat files to XML, the second process reads the Reporting Payer files (both external and internal), recipient (Payee) files (external, internal, and 1042) and tax form output files (both external and internal) generated by the Offline 1099 batch process and load data into the respective Tax Reporting Payer table, Tax Recipient table, and Forms Reporting Detail table. Once the file is processed, the process renames the Offline 1099 reporting payer files; appending the job Run ID and application date to the end of the file name, so that the same files will not be selected for processing again. The process records statistics such as the number of records read from the input file, the number of records loaded successfully, and the number of rejected records.

Finally, the third process selects the Forms Reporting Detail table records, consolidates the records if required as per the Tax Forms Options and Parameter table settings, and inserts records into the respective Forms Reporting table. The process also reads the Tax Reporting Payer and Tax Recipient tables to retrieve payer and recipient information populated on the tax forms.

The Forms Table Load Chain process includes two reports. The first report is the Tax Form Reporting Detail Table Load Exception Report which lists all records captured during each of the previous table load processes for tax reporting payers, tax recipients, and tax forms reporting detail tables. For each of the load processes, the exception report will report on the following items captured:

File Name (data file loaded)

Table Load (target table of load)

Number of records read from the input file

Number of records loaded successfully

Number of rejected records.

The second report, the Forms Reporting Table Load Processing Report generates the Load Processing report with the exceptions encountered and overall and box totals while processing the Forms Reporting Detail table records being loaded into the Form Reporting tables.

Major input for the Forms Table Load Chain Job:

Output files from the Offline 1099 batch process

Tax Form Options and Parameter (TAXOPT) table

Major updates by the Forms Table Load Chain Job:

Forms Reporting Tables

Forms Reporting Detail Tables

Tax Reporting Payer Table

Tax Recipient Table

Tax Form Reporting Detail Table Load Exception Report

Tax Form Reporting Table Load Processing Report

Chain Job Return Codes

The following table shows the potential Return Codes for the Forms Table Load Chain job. Note that the chain job will end with the highest Return Code across all of the jobs.

Return Code	Condition
Successful (1)	All of the jobs end successfully.
Warning (4)	One of the jobs in the chain ends with a Return Code of *Warning*.
Non-Fatal Error (8)	N/A
Failed (12)	One of the jobs in the chain ends with a Return Code of *Failed*.
Terminated (16)	One of the jobs in the chain ends with a Return Code of *Terminated*.
System Failure (20)	One of the jobs in the chain ends with a Return Code of *System Failure*.

Problem Resolution

If any of the jobs in the chain failed due to application errors it is advisable to restart the job (when possible) after correcting the errors instead of rescheduling the job. Restarting the job reduces

the processing time since the job resumes from where it has last committed and selects only the unprocessed records. When restarting is not an option, the entire chain can be run again.

Please refer to the individual jobs for details regarding the specific job processes and problem resolution.

1099 Flat Files to XML

Chain or Job Name	1099 Flat Files to XML
Recommended Frequency	Refer to chain job
Single Instance Required	Yes
Can be restarted?	No
Reports generated	No

Overview

This first job in the chain generates XML files from flat files which were generated by the Offline 1099 batch job.

This job step consists of following basic steps:

1. **Parameter Validation:** In this step, the process verifies the parameters. If the parameter validation is successful, the job goes to the next step. Otherwise, the job ends with a Return Code of *Failed*.
2. **Processing 1099 Flat files:** This process reads the Reporting Payer files (both external and internal), recipient (Payee) files (external, internal and 1042) and tax form output files (both external and internal) generated by the Offline 1099 batch process and generates XML for the Reporting Payer table, Tax Recipient table and the Forms Reporting Detail table. Once the file is processed, the process renames the Offline 1099 reporting payer files; appending the job Run ID and application date to the end of the file name, so that the same files are not selected for processing again.

Process Steps

The following table shows the various steps that the job goes through and the messages issued at each step:

Process Steps	Messages
1. Parameter Validation	• Validating Batch Parameters. • If AMSEXPORT is not provided, then “The AMSEXPORT parameter file cannot be empty” is recorded in the job log. • If AMSEXPORT is not a valid directory, then “The value entered in the AMSEXPORT parameter is not a valid directory” is recorded in the job log. • If AMSIMPORT is not provided, then “The AMSIMPORT parameter file cannot be empty” is

Process Steps	Messages
	recorded in the job log. • If AMSIMPORT is not a valid directory, then “The value entered in the AMSIMPORT parameter is not a valid directory” is recorded in the job log. • If AMSPARM is not provided, then “The AMSPARM parameter file cannot be empty” is recorded in the job log. • If AMSPARM parameter is not a valid directory, then “The value entered in the AMSPARM parameter is not a valid directory” is recorded in the job log. • If Commit Size is not provided, then “Commit Size defaulted to 1000” is recorded in the job log. • If Commit Size is not a valid integer greater than zero, then “Commit Size must be a positive whole number greater than zero” is recorded in the job log. • If Progression Counter is not provided, then “Progression Counter defaulted to 1000” is recorded in the job log. • If Progression Counter is not a valid integer greater than zero, then “Progression Counter must be a positive whole number greater than zero” is recorded in the job log. • If Common Chain Parameters File is not provided, then “Common Chain Parameters file name is not specified” is recorded in the job log. • If Common Chain Parameters file does not start with “.txt”, then “Common Chain Parameters file should have file extension of .txt” is recorded in the job log. • Parameter validation completed.
2. Processing of Records	Reading Flat File <<Flat File Name>> If there are no records to process, then the following message is issued: “No records found”.

Major Input

Internal Payer text file name (Payer.txt)

External Payer text file name (ExtPayer.txt)

Internal Payee text file name (Payee.txt)

External Payee text file name (ExtPayee.txt)

Internal Form 1099-MISC text file name (FormMISC.txt)

Internal Form 1099-NEC text file name (FormNEC.txt)

Internal Form 1099-INT text file name (FormINT.txt)

Internal Form 1099-G text file name (FormG.txt)

Internal Form 1099-S text file name (Form1099S.txt)

Internal Form 1042-S text file name (Form1042S.txt)

External Form 1099-MISC text file name (ExtFormMisc.txt)

External Form 1099-NEC text file name (ExtFormNEC.txt)

External Form 1099-INT text file name (ExtFormINT.txt)

External Form 1099-G text file name (ExtFormG.txt)

External Form 1099-S text file name (ExtForm1099S.txt)

External Form 1099-A text file name (ExtFormA.txt)

External Form 1099-C text file name (ExtFormC.txt)

External Form 1099-R text file name (ExtFormR.txt)

Internal 1042-S Payee text file name (Payee1042.txt)

Batch Parameters

Parameter	Description	Default Value
AMSEXPORT	Export file location at Flat to XML Job	$$AMSROOT$$/ExportImport
AMSIMPORT	Export file location at Flat to XML Job	$$AMSROOT$$/ExportImport
AMSPARM	Parameter Location at Flat to XML Job	$$AMSROOT$$/Parms
CHAIN_PARM_FILE	Common Chain Parameters File (.txt)	TaxRptDtlTblExcp.txt
CLIENT_NM	Client Name	
COMMIT_SIZE	Commit Block	1000
LOAD_DATA_IN_UPPERCASE	Load Data in Uppercase	Yes
DELIMITER	Delimiter	
PROG_CTR_SIZE	Progression Message Counter	100
ATTR_LIST_PAYER	Payer (Internal) text file attribute list.	PYR_TIN,PYR_TIN_TYP,RPT_1099_PYR,PYR_NM,PYR_NM_2,PYR_AD,PYR_CITY,PYR_ST,PYR_ZIP,PYR_PH,PYR_CH3_STA_CD,PYR_CH4_STA_CD,PYR_GIIN,PYR_CT

Parameter	Description	Default Value
		RY,PYR_FRGN_ID,TAX_YR
ATTR_LIST_PAYER_B	Payer (External) text file attribute list.	PYR_TIN,PYR_TIN_TYP,RPT_1099_PYR,PYR_NM,PYR_NM_2,PYR_AD,PYR_CITY,PYR_ST,PYR_ZIP,PYR_PH,PYR_CH3_STA_CD,PYR_CH4_STA_CD,PYR_GIIN,PYR_CTRY,PYR_FRGN_ID,TAX_YR
ATTR_LIST_PAYEE_1	Payee (Internal) text file attribute list.	RECP_TIN,RECP_TIN_TYP,RECP_SN,RECP_NM_1,RECP_NM_2,RECP_AD,RECP_CITY,RECP_ST,RECP_ZIP,ACCT_NO,RPT_1099_PYR
ATTR_LIST_PAYEE_2	Payee (External) text file attribute list.	RECP_TIN,RECP_TIN_TYP,RECP_SN,RECP_NM_1,RECP_NM_2,RECP_AD,RECP_CITY,RECP_ST,RECP_ZIP,ACCT_NO,RPT_1099_PYR
ATTR_LIST_1099M_1	Form 1099-MISC text file attribute list.	PYR_TIN,PYR_TIN_TYP,RECP_TIN,TAX_YR,TRANS_TYP_IND,PERMIT_NO,RECP_OLD_TIN,TRANS_DT,RENT_AM,ROYL_AM,OTHR_INC_AM,FED_INC_WHLD_AM,FISH_BOAT_ROYL_AM,MED_HC_PYMT_AM,SPYMT_DIV_INT_AM,EXS_GLDN_PYMT_AM,CROP_INS_PYMT_AM
ATTR_LIST_1099M_2	Form 1099-MISC text file attribute list continuation.	GROSS_PD_ATY_AM,S409A_DFR_AM,S409A_INC_AM,ST_WHLD_1_AM,ST_ID_NO_1,ST_INC_1_AM,PYR_DIR_SALES_FL,NQLF_DFR_COMP_AM,FISH_PUR_RESALE_AM,FATCA_FIL_RQ,ACCT_NO,RPT_1099_PYR,RECP_TIN_TYP,RECP_OLD_TIN_TYP,CMNTS,BAT_NO,RPT_DEPT,RPT_UNIT,RPT_LOC,OLD_FORM_TYP_IND
ATTR_LIST_1099I_1	Form 1099-INT text file attribute list.	PYR_TIN,PYR_TIN_TYP,RECP_TIN,TAX_YR,TRANS_TYP_IND,PERMIT_NO,RECP_OLD_TIN,TRANS_DT,INTR_INC_AM,ERLY_WDRWL_PNTY_AM,INTR_BOND_OBLG_AM,FED_INC_WHLD_AM,INVS_AM,FGN_TAX_PD_AM,FGN_US_POSS
ATTR_LIST_1099I_2	Form 1099-INT text file attribute	TAX_EXMP_AM,PRV_ACTV_BOND_AM,MKT_DISC_AM,BOND_PRE

Parameter	Description	Default Value
	list continuation.	MN_AM,BOND_TREAS_OBLG_AM,BOND_TAX_EXMP_AM,CUSIP_NO,ST_1,ST_ID_NO_1,ST_WHLD_1_AM,ACCT_NO,RPT_1099_PYR,RECP_TIN_TYP,RECP_OLD_TIN_TYP,CMNTS,BAT_NO,RPT_DEPT,RPT_UNIT,RPT_LOC,OLD_FORM_TYP_IND
ATTR_LIST_1099G_1	Form 1099-G text file attribute list.	PYR_TIN,PYR_TIN_TYP,RECP_TIN,TAX_YR,TRANS_TYP_IND,PERMIT_NO,RECP_OLD_TIN,TRANS_DT,UEMPL_COMP_AM,ST_LCL_INC_RFND_AM,AM_FOR_TAX_YR,FED_INC_WHLD_AM,FED_INC_WHLD_AM,RTAA_PYMT_AM,TAX_GRNT_AM
ATTR_LIST_1099G_2	Form 1099-G text file attribute list continuation.	AGR_PYMT_AM,TRAD_BUS_FL,MKT_GAIN_AM,ST_1,ST_ID_NO_1,ST_WHLD_1_AM,ACCT_NO,RPT_1099_PYR,RECP_TIN_TYP,RECP_OLD_TIN_TYP,CMNTS,BAT_NO,RPT_DEPT,RPT_UNIT,RPT_LOC,OLD_FORM_TYP_IND
ATTR_LIST_1099S_1	Form 1099-S text file attribute list.	PYR_TIN,PYR_TIN_TYP,RECP_TIN,TAX_YR,TRANS_TYP_IND,PERMIT_NO,RECP_OLD_TIN,TRANS_DT,CLSNG_DT,GRS_PRCDS_AM,AD_LGL_DSCR_1,AD_LGL_DSCR_2,XFER_RECVD_FL,XFER_FRGN_PRSN_FL,BUYR_PART_RE_TAX,ACCT_NO,RPT_1099_PYR
ATTR_LIST_1099S_2	Form 1099-S text file attribute list continuation	RECP_TIN_TYP,RECP_OLD_TIN_TYP,CMNTS,BAT_NO,RPT_DEPT,RPT_UNIT,RPT_LOC,OLD_FORM_TYP_IND
ATTR_LIST_1042S_1	Form 1042-S text file attribute list.	PYR_TIN,PYR_TIN_TYP,RECP_TIN,TAX_YR,TRANS_TYP_IND,PERMIT_NO,RECP_OLD_TIN,TRANS_DT,ACCT_NO,INC_CD,CH3_TAX_RT,CH3_EXMP_CD,GRS_INC_AM,WHLD_ALW_AM,NET_INC_AM,WHLD_AM,TAX_WH_OTH_AGNT_AM,WA_TAX_PD_AM,TOT_WHLD_CR_AM,RPYMT_AM,PYR_ST_TAX_NO,WHLD_ST,RPT_1099_PYR
ATTR_LIST_1042S_2	Form 1042-S text file attribute list	PYR_TIN_NM_1,PYR_TIN_1042S,ST_WHLD_AM,CH4_TAX_RT,CH4_EXMP_CD,CH_IND,TAX_NOT_D

Parameter	Description	Default Value
	continuation.	EP,WA_EIN,WA_CH3_STA_CD,WA_CH4_STA_CD,WA_NM,WA_GIIN,WA_CTRY_CD,WA_FRGN_TIN,WA_AD,WA_CITY,WA_ST,WA_CTRY,WA_ZIP,PRIM_WA_NM,PRIM_WA_EIN,ENT_EIN,ENT_CH3_STA_CD
ATTR_LIST_1042S_3	Form 1042-S text file attribute list continuation	ENT_CH4_STA_CD,ENT_NM,ENT_GIIN,ENT_CTRY_CD,ENT_FRGN_TAX_ID,ENT_AD,ENT_CITY,ENT_ST,ENT_CTRY,ENT_ZIP,PYR_GIIN,PYR_CH3_STA_CD,PYR_CH4_STA_CD,RECP_TIN_TYP,RECP_OLD_TIN_TYP,CMNTS,BAT_NO,UNIQ_FORM_ID,AMND_NO,PRO_BAS_RPT_FL,WHLD_SUB_YR
ATTR_LIST_1099A_1	Form 1099-A text file attribute list.	PYR_TIN,PYR_TIN_TYP,RECP_TIN,TAX_YR,TRANS_TYP_IND,PERMIT_NO,RECP_OLD_TIN,TRANS_DT,ACQ_ABAN_DT,BAL_PRIN_OUT_AM,FAIR_MRK_VL_AM,PERSN_LIAB_FL,DSCR,ACCT_NO,RPT_1099_PYR,RECP_TIN_TYP,RECP_OLD_TIN_TYP,CMNTS,BAT_NO,RPT_DEPT,RPT_UNIT,RPT_LOC,OLD_FORM_TYP_IND
ATTR_LIST_1099C_1	Form 1099-C text file attribute list.	PYR_TIN,PYR_TIN_TYP,RECP_TIN,TAX_YR,TRANS_TYP_IND,PERMIT_NO,RECP_OLD_TIN,TRANS_DT,ID_EVNT_DT,DEBT_DCHRG_AM,INT_INCL_AM,DEBT_DSCR,PERSN_LIAB_FL,ID_EVNT_CD,FAIR_MRK_VL_AM,ACCT_NO,RPT_1099_PYR,RECP_TIN_TYP,RECP_OLD_TIN_TYP
ATTR_LIST_1099C_2	Form 1099-C text file attribute list continuation.	CMNTS,BAT_NO,RPT_DEPT,RPT_UNIT,RPT_LOC,OLD_FORM_TYP_IND
ATTR_LIST_1099R_1	Form 1099-R text file attribute list.	PYR_TIN,PYR_TIN_TYP,RECP_TIN,TAX_YR,TRANS_TYP_IND,PERMIT_NO,RECP_OLD_TIN,TRANS_DT,GRS_DIST_AM,TAXABLE_AM,CAP_GAIN_AM,FED_INC_WHLD_AM,EMPL_CNTRB_PREM_AM,NET_UNR_APPR_AM,DIST_CD,OTHR_AM,TOT_EMPL_CNTRB_AM,IRR_ALOC_AM

Parameter	Description	Default Value
ATTR_LIST_1099R_2	Form 1099-R text file attribute list continuation.	FRST_ROTH_CNTRB_AM,ST_WHLD_1_AM,ST_PYR_ST_NO_1,ST_DIST_1_AM,LCL_WHLD_1_AM,LCL_NM_1,LCL_DIST_1_AM,FATCA_FIL_RQ,DT,PYMT_1099R,ACCT_NO,RPT_1099_PYR,RECP_TIN_TYP,RECP_OLD_TIN_TYP,CMNTS,BAT_NO,RPT_DEPT,RPT_UNIT,RPT_LOC,OLD_FORM_TYP_IND
ATTR_LIST_1099N_1	Form 1099-NEC text file attribute list	PYR_TIN,PYR_TIN_TYP,RECP_TIN,TAX_YR,TRANS_TYP_IND,PERMIT_NO,RECP_OLD_TIN,TRANS_DT,NON_EMPL_COMP_AM,FED_INC_WHLD_AM,ST_WHLD_1_AM,ST_ID_NO_1,ST_INC_1_AM,PYR_DIR_SALES_FL,ACCT_NO,RPT_1099_PYR,RECP_TIN_TYP,RECP_OLD_TIN_TYP
ATTR_LIST_1099N_2	Form 1099-NEC text file attribute list continuation	CMNTS,BAT_NO,RPT_DEPT,RPT_UNIT,RPT_LOC,OLD_FORM_TYP_IND
ATTR_LIST_PAYEE_3	Payee (1042) text file attribute list.	RECP_TIN,RECP_TIN_TYP,RECP_SN,RECP_NM_1,RECP_NM_2,RECP_AD,RECP_AD_1,RECP_CITY,RECP_ST,RECP_ZIP,RECP_CD,RECP_CTRY,ACCT_NO,RECP_RES_CTRY_CD,RPT_1099_PYR,RECP_FGRN_TIN,RECP_DOB,RECP_CH4_STA_CD,RECP_GIIN,WHLD_ST,RECP_LOB
DATAOBJECT_NAME_01	The Tax Reporting Payer database table name.	R_TAX_RPT_PYR
DATAOBJECT_NAME_02	The Tax Recipient database table name.	R_TAX_RECPNT
DATAOBJECT_NAME_03	The Form 1099-MISC Detail database table name.	R_FORM_1099MIS_DET
DATAOBJECT_NAME_04	The Form 1099-INT Detail database table name.	R_FORM_1099INT_DET

Parameter	Description	Default Value
DATAOBJECT_NAME_05	The Form 1099-G Detail database table name.	R_FORM_1099G_DET
DATAOBJECT_NAME_06	The Form 1099-S Detail database table name.	R_FORM_1099S_DET
DATAOBJECT_NAME_07	The Form 1042-S Detail database table name.	R_FORM_1042S_DET
DATAOBJECT_NAME_08	The Form 1099-A Detail database table name.	R_FORM_1099A_DET
DATAOBJECT_NAME_09	The Form 1099-C Detail database table name.	R_FORM_1099C_DET
DATAOBJECT_NAME_10	The Form 1099-R Detail database table name.	R_FORM_1099R_DET
DATAOBJECT_NAME_11	The Form 1099-NEC Detail database table name.	R_FORM_1099NEC_DET
FLAT_FILE_NM_01	The Internal Payer text file name. This file is the input for this job.	Payer.txt
FLAT_FILE_NM_02	The External Payer text file name. This file is the input for this job.	ExtIncomePayer.txt
FLAT_FILE_NM_03	The Internal Payee text file name. This file is the input for this job.	Payee.txt
FLAT_FILE_NM_04	The External Payee text file name. This file is the input for this job.	ExtPayee.txt

Parameter	Description	Default Value
FLAT_FILE_NM_05	The Internal Form 1099-MISC text file name. This file is the input for this job.	FormMISC.txt
FLAT_FILE_NM_06	The Internal Form 1099-INT text file name. This file is the input for this job.	FormINT.txt
FLAT_FILE_NM_07	The Internal Form 1099-G text file name. This file is the input for this job.	FormG.txt
FLAT_FILE_NM_08	The Internal Form 1099-S text file name. This file is the input for this job.	Form1099S.txt
FLAT_FILE_NM_09	The Internal Form 1042-S text file name. This file is the input for this job.	Form1042S.txt
FLAT_FILE_NM_10	The External Form 1099-MISC text file name. This file is the input for this job.	ExtFormMisc.txt
FLAT_FILE_NM_11	The External Form 1099-INT text file name. This file is the input for this job.	ExtFormINT.txt
FLAT_FILE_NM_12	The External Form 1099-G text file name. This file is the input for this job.	ExtFormG.txt
FLAT_FILE_NM_13	The External Form 1099-S text file name. This file is the input for this	ExtForm1099S.txt

Parameter	Description	Default Value
	job.	
FLAT_FILE_NM_14	The External Form 1099-A text file name. This file is the input for this job.	ExtFormA.txt
FLAT_FILE_NM_15	The External Form 1099-C text file name. This file is the input for this job.	ExtFormC.txt
FLAT_FILE_NM_16	The External Form 1099-R text file name. This file is the input for this job.	ExtFormR.txt
FLAT_FILE_NM_17	The Internal 1042-S Payee text file name. This file is the input for this job.	Payee1042.txt
FLAT_FILE_NM_18	The Internal Form 1099-NEC text file name. This file is the input for this job.	FormNEC.txt
FLAT_FILE_NM_19	The External Form 1099-NEC text file name. This file is the input for this job.	ExtFormNEC.txt
XML_FILE_NM_01	The Tax Reporting Payer (Internal) output file name. XML file is created with this name and is stored in the Export file location.	TaxPayer.xml
XML_FILE_NM_02	The Tax Reporting Payer (Internal) output file name. XML file is created with this name and is stored in the Export file location.	ExtTaxPayer.xml

Parameter	Description	Default Value
XML_FILE_NM_03	The Tax Recipient (Internal) output file name. XML file is created with this name and is stored in the Export file location.	TaxRecipient.xml
XML_FILE_NM_04	The Tax Recipient (External) output file name. XML file is created with this name and is stored in the Export file location.	ExtTaxRecipient.xml
XML_FILE_NM_05	The Internal Form 1099-MISC output file name. XML file is created with this name and is stored in the Export file location.	Form1099M.xml
XML_FILE_NM_06	The Internal Form 1099-INT output file name. XML file is created with this name and is stored in the Export file location.	Form1099I.xml
XML_FILE_NM_07	The Internal Form 1099-G output file name. XML file is created with this name and is stored in the Export file location.	Form1099G.xml
XML_FILE_NM_08	The Internal Form 1099-S output file name. XML file is created with this name and is stored in the Export file location.	Form1099S.xml
XML_FILE_NM_09	The Internal Form 1042-S output file name. XML file is created with this name and is stored	Form1042S.xml

Parameter	Description	Default Value
	in the Export file location.	
XML_FILE_NM_10	The External Form 1099-MISC output file name. XML file is created with this name and is stored in the Export file location.	ExtForm1099M.xml
XML_FILE_NM_11	The External Form 1099-INT output file name. XML file is created with this name and is stored in the Export file location.	ExtForm1099I.xml
XML_FILE_NM_12	The External Form 1099-G output file name. XML file is created with this name and is stored in the Export file location.	ExtForm1099G.xml
XML_FILE_NM_13	The External Form 1099-S output file name. XML file is created with this name and is stored in the Export file location.	ExtForm1099S.xml
XML_FILE_NM_14	The External Form 1099-A output file name. XML file is created with this name and is stored in the Export file location.	ExtForm1099A.xml
XML_FILE_NM_15	The External Form 1099-C output file name. XML file is created with this name and is stored in the Export file location.	ExtForm1099C.xml
XML_FILE_NM_16	The External Form 1099-R output file name. XML file is	ExtForm1099R.xml

Parameter	Description	Default Value
	created with this name and is stored in the Export file location.	
XML_FILE_NM_17	The Tax Recipient (1042-S) output file name. XML file is created with this name and is stored in the Export file location.	1042STaxRecipient.xml
XML_FILE_NM_18	The Internal Form 1099-NEC output file name. XML file is created with this name and is stored in the Export file location.	Form1099E.xml
XM_FILE_NM_19	The External Form 1099-NEC output file name. XML file is created with this name and is stored in the Export file location.	ExtForm1099E.xml

Major Output

Tax Reporting Payer (Internal) output file name (TaxPayer.xml)

Tax Reporting Payer (External) output file name (ExtTaxPayer.xml)

Tax Recipient (Internal) output file name (TaxRecipient.xml)

Tax Recipient (External) output file name (ExtTaxRecipient.xml)

Internal Form 1099-MISC output file name (Form1099M.xml)

Internal Form 1099-NEC output file name (Form1099E.xml)

Internal Form 1099-INT output file name (Form1099I.xml)

Internal Form 1099-G output file name (Form1099G.xml)

Internal Form 1099-S output file name (Form1099S.xml)

Internal Form 1042-S output file name (Form1042S.xml)

External Form 1099-MISC output file name (ExtForm1099M.xml)

External Form 1099-NEC output file name (ExtForm1099E.xml)

External Form 1099-INT output file name (ExtForm1099I.xml)

External Form 1099-G output file name (ExtForm1099G.xml)

External Form 1099-S output file name (ExtForm1099S.xml)

External Form 1099-A output file name (ExtForm1099A.xml)

External Form 1099-C output file name (ExtForm1099C.xml)

External Form 1099-R output file name (ExtForm1099R.xml)

Tax Reporting Payer (1042-S) output file name (1042STaxRecipient.xml)

Restartability Information

This job cannot be re-started. If flat files have failed to create XML files, the errors can be found on the processing log. Reschedule the job after investigating the errors in the job logs**.**

Chain / Job Return Code

The following table shows the potential job Return Codes for the Forms Table Load Chain job.

Return Code	Condition
Successful (1)	All flat files are processed successfully.
Warning (4)	No records found. This could be because of the following reason: • No records found in the flat files
Non-Fatal Error (8)	N/A
Failed (12)	The job may fail under the following conditions: • Parameters are invalid. • Run time exceptions for unexpected situations. When this job ends with a Return Code of *Failed,* subsequent jobs in the chain will be set to *Inactive*.
Terminated (16)	This Return Code is issued when the job is terminated by the user. When this job ends with a Return Code of *Terminated,* subsequent jobs in the chain are set to *Inactive*.
System Failure (20)	This Return Code is issued when the job is terminated because of database server or network issues. When this job ends with a Return Code of *System Failure*, subsequent jobs in the chain are set to *Inactive*.

Sort Criteria

N/A

Selection Criteria

N/A

Problem Resolution

The following table shows the possible Return Codes and recommendations for each processing step.

Step 1: Parameter Validation:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	This step does not issue this Return Code.	N/A	N/A
Non-Fatal Error (8)	This step does not issue this Return Code.	N/A	N/A
Failed (12)	Required Parameters are not entered. **Sample Message:** The AMSEXPORT Parameter File cannot be empty	Schedule a new job by passing required and valid parameter values.	N/A
	The job failed because of runtime exceptions for an unexpected situation.	The reason for the failure needs to be investigated before re-scheduling a new job.	
Terminated (16)	The job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can either be restarted or a new job can be scheduled.	N/A
System Failure (20)	The job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can either be restarted or a new job can be scheduled.	N/A

Step 2: Processing 1099 Flat Files

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the parameters are validated successfully.	N/A	N/A

Possible Return Codes	Condition	Recommendation	Other Instructions
Warning (4)	The job ended with a warning because there is no flat file to process it. **Sample Message:** No records found:	Please run the Offline 1099 batch job in Testing and Magnetic Mode after setting the Use Tax Reporting parameter to *Yes* on the APPCTRL page.	Alternatively, the job can be rescheduled with a different set of parameters.
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	The job failed due to fatal conditions.	In this step, the job may fail under the following two conditions: • Encounters any runtime exceptions • Failed during restart If the job fails because of the runtime exceptions, investigate the exception reported by the process, resolve the error and restart the job.	If another instance of the job has already been scheduled and run successfully, then this job should not be restarted. Only a new job should be scheduled.
	The job failed while restarting the job since another instance of the job has already been run successfully. **Sample Message**: Cannot restart the job since another instance of this job has already been run successfully.	**Recommendation:** Schedule a new job.	
Terminated (16)	The job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can either be restarted or a new job can be scheduled.	If another instance of the job has already been scheduled and run successfully, then this job should not be restarted. Only a new job should be

Possible Return Codes	Condition	Recommendation	Other Instructions
			scheduled.
System Failure (20)	The job is terminated because of database server or network issues.	The reason for the system failure needs to be investigated. The job can either be restarted or a new job can be scheduled.	If another instance of the job has already been scheduled and run successfully, then this job should not be restarted. Only a new job should be scheduled.

Load XML Files to Detail Tables

Chain or Job Name	Load XML Files to Detail Tables
Recommended Frequency	Refer to the chain job
Single Instance Required	Yes
Can be restarted?	No
Reports generated	Yes. Tax Form Reporting Detail Table Load Exception Report.

Overview

This job starts by first validating the batch parameters. If the parameters are valid, then it loads the records from the XML file(s) generated by the 1099 Flat Files to XML job to the Tax Reporting Payer, the Tax Recipient, and the Tax form reporting detail records. If the parameters are not valid, the job issues appropriate messages and ends with a status of *Failed*.

Process Steps

The following table shows the various steps that the job goes through and the messages issued at each step:

Process Steps	Messages
1. Parameter Validation	• Validating Batch Parameters • If AMSEXPORT is not provided, then "The AMSEXPORT parameter file cannot be empty" is recorded in the job

Process Steps	Messages
	log. • If AMSEXPORT is not a valid directory, then “The value entered in the AMSEXPORT parameter is not a valid directory” is recorded in the job log. • If Parameter file name is not provided, then “Parameter file name is not specified” is recorded in the job log. • If Parameters file name do not start with “.txt”, then “Parameter file should have file extension of .txt” is recorded in the job log. • Parameter validation completed.
2. Processing XML file	• If XML file failed to load table, then “<<XML File Name>>The XML file has failed to load successfully” is recorded in the job log. • If XML file is loaded to the table, then “<<XML File Name>> Records loaded into Table for XML file Name” is recorded in the job log.

Major Input

Tax Reporting Payer (Internal) file name(TaxPayer.xml)

Tax Reporting Payer (External) file name (ExtTaxPayer.xml)

Tax Recipient (Internal) file name (TaxRecipient.xml)

Tax Recipient (External) file name (ExtTaxRecipient.xml)

Internal Form 1099-MISC file name (Form1099M.xml)

Internal Form 1099-NEC file name (Form1099N.xml)

Internal Form 1099-INT file name (Form1099I.xml)

Internal Form 1099-G file name (Form1099G.xml)

Internal Form 1099-S file name (Form1099S.xml)

Internal Form 1042-S file name (Form1042S.xml)

External Form 1099-MISC file name (ExtForm1099M.xml)

External Form 1099-NEC file name (EXTForm1099N.xml)

External Form 1099-INT file name (ExtForm1099I.xml)

External Form 1099-G file name (ExtForm1099G.xml)

External Form 1099-S file name (ExtForm1099S.xml)

External Form 1099-A file name (ExtForm1099A.xml)

External Form 1099-C file name (ExtForm1099C.xml)

External Form 1099-R file name (ExtForm1099R.xml)

Tax Reporting Payer (1042-S) file name (1042STaxRecipient.xml)

Batch Parameters

Parameter	Description	Default Value
AMSEXPORT	$$AMSROOT$$/ExportImport	Export file location at Flat to XML Job
CHAIN_PARM_FILE	TaxRptDtlTblExcp.txt	Parameters File (.txt)

Restartability Information

This job cannot be re-started. If records are failed to be loaded to the table, the error can be found on the processing log. Reschedule the job after investigating the errors in the job logs.

Major Output

Tax Reporting Payer table (R_TAX_RPT_PYR)

Tax Recipient table (R_TAX_RECPNT)

Form 1099-MISC table (R_FORM_1099MIS_DET)

Form 1099-NEC table (R_FORM_1099NEC_DET)

Form 1099-INT table (R_FORM_1099INT_DET)

Form 1099-G table (R_FORM_1099G_DET)

Form 1099-S table (R_FORM_1099S_DET)

Form 1042-S table (R_FORM_1042S_DET)

Form 1099-A table (R_FORM_1099A_DET)

Form 1099-C table (R_FORM_1099C_DET)

Form 1099-R table (R_FORM_1099R_DET)

Chain / Job Return Code

The following table shows the potential job Return Codes for the Load XML Files to Detail Tables job.

Return Code	Condition
Successful (1)	All XML files are processed successfully.
Warning (4)	No records found. This could be because of the following reason: • No records found in the XML files
Non-Fatal Error (8)	N/A
Failed (12)	The job may fail under the following conditions:

	• Parameters are invalid. • Run time exceptions for unexpected situations. When this job ends with a Return Code of *Failed*, subsequent jobs in the chain will be set to *Inactive*.
Terminated (16)	This Return Code is issued when the job is terminated by the user. When this job ends with a Return Code of *Terminated*, subsequent jobs in the chain are set to *Inactive*.
System Failure (20)	This Return Code is issued when the job is terminated because of database server or network issues. When this job ends with a Return Code of *System Failure*, subsequent jobs in the chain are set to *Inactive*.

Sort Criteria

N/A

Selection Criteria

N/A

Problem Resolution

The following table shows the possible Return Codes and recommendations for each processing step.

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the parameters are validated successfully.	N/A	N/A
Warning (4)	The job ended with a warning because there is no XML File to process it. **Sample Message:** No records found:	Please run the Offline 1099 batch job in Testing and Magnetic Mode after setting the Use Tax Reporting parameter as *Yes* on the APPCTRL page.	Alternatively, the job can be rescheduled with a different set of parameters.
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	The job failed due to fatal conditions.	In this step, the job may fail under the following two conditions: • Encounters any runtime exceptions	If another instance of the job has already been scheduled and run successfully, then this job should not be restarted. Only a new

Possible Return Codes	Condition	Recommendation	Other Instructions
		• Failed during restart If the job fails because of runtime exceptions, investigate the exception reported by the process, resolve the error and restart the job.	job should be scheduled.
	The job failed while restarting the job since another instance of the job has already been run successfully. **Sample Message**: Cannot restart the job since another instance of this job has already been run successfully.	**Recommendation:** Schedule a new job.	
Terminated (16)	The job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can either be restarted or a new job can be scheduled.	If another instance of the job has already been scheduled and run successfully, then this job should not be restarted. Only a new job should be scheduled.
System Failure (20)	The job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can either be restarted or a new job can be scheduled.	If another instance of the job has already been scheduled and run successfully, then this job should not be restarted. Only a new job should be scheduled.

Load Forms Reporting Tables

Chain or Job Name	Load Forms Reporting Tables
Recommended Frequency	Refer to the chain job
Single Instance Required	Yes
Can be restarted?	No

Reports generated	Yes. Forms Reporting Table Load Processing Report

Overview

This job step consists of the following two basic steps:

1. Parameter Validation: In this step, the process verifies the parameters. If the parameter validation is successful, the job goes to the next step. Otherwise, the job ends with a Return Code of Failed.
2. Processing Tax Form Reporting Detail records: This process selects records on each of the Forms Reporting Detail tables, consolidates the records based on the Consolidation options on the Tax Form Options and Parameter table (Consolidate Internal and External Income field) and inserts the records into the appropriate Forms Reporting table. The process also populates the Recipient and Reporting Payer information on the respective Forms Reporting table records from the Recipient and Tax Reporting Payer tables. Once the Forms Reporting Detail table record is processed, the batch process marks that record as being processed by setting the processed flag to Yes.

Process Steps

The following table shows the various steps that the job goes through and the messages issued at each step:

Process Steps	Messages
1. Parameter Validation	• Validating Batch Parameters • If Commit Block Size is not provided, then “Defaulting Commit Block Size to 1000” is recorded in the job log. • If Commit Block Size is not a valid integer greater than zero, then “Commit Block Size should be a positive integer” is recorded in the job log. • If Progression Counter is not provided, then “Defaulting Progression Message Counter to 100” is recorded in the job log. • If Progression Counter is not a valid integer greater than zero, then “Progression Message Counter should be a positive integer” is recorded in the job log. • If Tax Year is empty, then "Tax Year cannot be empty” is recorded in the job log. • If Tax Year is not valid, then "Tax Year <<Tax Year>> is not valid on the Fiscal Year (FY) table” is recorded in the job log. • If Default Reporting Payer is not valid, then "The Default Reporting Payer does not exist on 1099 Reporting Payer Information table” is recorded in the job log.

<table>
<tr><th>Process Steps</th><th>Messages</th></tr>
<tr><td></td><td>

If Form Type is not valid, it indicates that the entered Form Type is not a valid value.
Valid values are as per below chart and same are recorded in the job log.

<table>
<tr><th>Forms</th><th>Valid Form Type value</th></tr>
<tr><td>1042-S</td><td>N</td></tr>
<tr><td>1099-G</td><td>G</td></tr>
<tr><td>1099-INT</td><td>I</td></tr>
<tr><td>1099-MISC</td><td>M</td></tr>
<tr><td>1099-S</td><td>S</td></tr>
<tr><td>1099-A</td><td>A</td></tr>
<tr><td>1099-C</td><td>C</td></tr>
<tr><td>1099-R</td><td>R</td></tr>
<tr><td>1099-NEC</td><td>E</td></tr>
</table>

Parameter validation completed.

</td></tr>
<tr><td>2. Processing of Records</td><td>

Selection of eligible records from Form <<Form Type>> Reporting Detail Table completed".
Processing the selected Form <<Form Type>> Reporting Detail records.
Depending on the progression Counter size the “<<Progression Count>> record(s) processed” is displayed in the job log.
Total <<TOTAL COUNT>> Reporting Detail record(s) are processed.
Processing Form <<Form Type>> Reporting Detail Table completed.

</td></tr>
</table>

Major Input

Tax Reporting Payer table (R_TAX_RPT_PYR)

Tax Recipient table (R_TAX_RECPNT)

Form 1099-MISC table (R_FORM_1099MIS_DET)

Form 1099-NEC table (R_FORM_1099NEC_DET)

Form 1099-INT table (R_FORM_1099INT_DET)

Form 1099-G table (R_FORM_1099G_DET)

Form 1099-S table (R_FORM_1099S_DET)

Form 1042-S table (R_FORM_1042S_DET)

Form 1099-A table (R_FORM_1099A_DET)

Form 1099-C table (R_FORM_1099C_DET)

Form 1099-R table (R_FORM_1099R_DET)

Tax Form Options and Parameter Table (R_TAX_OPT_PARM)

Batch Parameters

Parameter	Description	Default Value
CLIENT_NM	Client Name	No default
COMMIT_BLOCK_SIZE	Commit Block	1000
DFLT_RPT_PYR	Default Reporting Payer	No default
FORM_TYP	Form Type	No default
PROG_CTR_SZ	Progression Message Counter	100
TAX_YR	Tax Year	No default

Restartability Information

This job step is implemented with checkpoints. If the process fails for any reason (such as the network is down or the server is down), then the process can be restarted from the last point where it stopped.

If the job fails in any of the above steps after Parameter Validation; it can be restarted by resolving the errors. Any job failed during parameter validations would not be restarted. If the job is restarted, it will start from the step where it failed earlier and won't start from the beginning.

Instead of restarting the job, if a new job is scheduled with the same set of parameters after resolving the table space issue, the job will start from the first step.

The restart will fail if another instance of this job has been scheduled and ran successfully before restarting the failed job.

Major Output

Tax Reporting Payer table (R_TAX_RPT_PYR)

Tax Recipient table (R_TAX_RECPNT)

Form 1099-MISC table (R_FORM_1099MIS_DET)

Form 1099-NEC table (R_FORM_1099NEC_DET)

Form 1099-INT table (R_FORM_1099INT_DET)

Form 1099-G table (R_FORM_1099G_DET)

Form 1099-S table (R_FORM_1099S_DET)

Form 1042-S table (R_FORM_1042S_DET)

Form 1099-A table (R_FORM_1099A_DET)

Form 1099-C table (R_FORM_1099C_DET)

Form 1099-R table (R_FORM_1099R_DET)

Form 1042-S Reporting Detail Table (R_FORM_1042S_RPT)

Form 1099-G Reporting Detail Table (R_FORM_1099G_RPT)

Form 1099-INT Reporting Detail Table (R_FORM_1099INT_RPT)

Form 1099-MISC Reporting Detail Table (R_FORM_1099MIS_RPT)

Form 1099-S Reporting Detail Table (R_FORM_1099S_RPT)

Form 1099-A Reporting Detail Table (R_FORM_1099A_RPT)

Form 1099-C Reporting Detail Table (R_FORM_1099C_RPT)

Form 1099-R Reporting Detail Table (R_FORM_1099R_RPT)

Chain / Job Return Code

The following table shows the potential job Return Codes for the Load Forms Reporting Tables job.

Return Code	Condition
Successful (1)	All of the selected records are processed successfully.
Warning (4)	No eligible records found.
Non-Fatal Error (8)	N/A
Failed (12)	The job may fail under the following conditions: • Parameters are invalid. • Run time exceptions for unexpected situations. When this job ends with a Return Code of *Failed*, subsequent jobs in the chain will be set to *Inactive*.
Terminated (16)	This Return Code is issued when the job is terminated by the user. When this job ends with a Return Code of *Terminated*, subsequent jobs in the chain are set to *Inactive*.
System Failure (20)	This Return Code is issued when the job is terminated because of database server or network issues. When this job ends with a Return Code of *System Failure*, subsequent jobs in the chain are set to *Inactive*.

Sort Criteria

Tax Form Reporting Detail table

Tax Year, Reporting Payer, TIN, TIN Type, 1099-S Number (1099-S tax reporting detail table only), Account Number.

Selection Criteria

Processing Tax Form Options and Parameter (TAXOPT) records

- Tax Year matches Tax Year (batch parameter)
- Form Type is not null or Form Type matches Form Type (batch parameter)
- Select Tax Form Reporting Detail records.

For each selected TAXOPT record above, select records from the respective Tax Form Reporting Detail table for the Form Type on the TAXOPT record where the following conditions are true: (RPTDTL denotes Reporting Detail Table)

- Tax Year (RPTDTL) matches with Tax Year (TAXOPT); AND
- Processed flag (RPTDTL) is set to *No*.

Problem Resolution

The following table shows the possible Return Codes and recommendations for each processing step.

Step 1: Parameter Validation:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	This step does not issue this Return Code.	N/A	N/A
Non-Fatal Error (8)	This step does not issue this Return Code.	N/A	N/A
Failed (12)	The required parameters are not entered. Sample Message: Tax Year cannot be empty.	Schedule a new job by passing the required and valid parameter values.	N/A
	The job failed because of runtime exceptions for an unexpected situation.	The reason for the failure needs to be investigated before scheduling a new job.	
Terminated (16)	The job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can either be restarted or a new job can be scheduled.	N/A

Possible Return Codes	Condition	Recommendation	Other Instructions
System Failure (20)	The job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can either be restarted or a new job can be scheduled.	N/A

Step 2: Processing Tax form reporting detail records

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the parameters are validated successfully.	N/A	N/A
Warning (4)	No eligible records were found on the tables for processing.	Review the selection parameters to verify that they were correct.	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	The job failed due to fatal conditions.	In this step, the job may fail under the following two conditions. • Encounters any runtime exceptions • Failed during restart If the job fails because of runtime exceptions, investigate the exception reported by the process, resolve the error and restart the job.	If another instance of the job has already been scheduled and run successfully, then this job should not be restarted. Only a new job should be scheduled.
	The job failed while restarting the job since another instance of the job has already been run successfully. **Sample Message**: Cannot restart the job since another instance of this job has already been run successfully.	**Recommendation:** Schedule a new job.	

Possible Return Codes	Condition	Recommendation	Other Instructions
Terminated (16)	The job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can either be restarted or schedule a new job.	If another instance of the job has already been scheduled and run successfully, then this job should not be restarted. Only a new job should be scheduled.
System Failure (20)	The job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can either be restarted or a new job can be scheduled.	If another instance of the job has already been scheduled and run successfully, then this job should not be restarted. Only a new job should be scheduled.

2.1.25 Generate Positive Pay

Chain or Job Name	Generate Positive Pay
Recommended Frequency	On Demand
Single Instance Required	Yes
Can be restarted?	N/A
Reports generated	Generate Positive Pay Exception Report

Overview

Bank Positive Pay is an anti-fraud service offered by commercial banks. It protects payers against altered checks and counterfeit check fraud. After checks are issued, the payer department transmits to their bank a list of checks that were issued in the disbursement run. This list is imported into the bank's system. Later, when the checks are presented to the bank for payment, the bank matches each check presented against the transmitted lists. If the information on the check, matches the transmitted information, the check is processed for payment. If the information does not match, then the check is not cleared and the transmitting department is notified.

The Generate Positive Pay job generates one Positive Pay text file per bank according to the information on the Positive Pay Configuration (PPCNFG) page, which stores details regarding each bank's file format specifications. Each file contains one detail line for each selected record on Check Reconciliation (CHREC), and at each bank's option one header line and/or one summary line.

When check records are selected from CHREC, a hidden Positive Pay indication is set to *true*. This ensures newly issued checks are not selected and sent to the bank more than once. If any issued checks are cancelled before the check is reconciled, this process will only select the cancelled check record if the check has a Positive Pay indication of *true* and the parameters are established to select cancelled checks. Check records that are issued and cancelled in the same day (or within the same cycle) will not be selected by Generate Positive Pay because the check was not issued or sent to the vendor.

Major Input

Check Reconciliation (CHREC / R_AP_CHK_RECON)

Bank (BANK / R_AP_BANK_ACCT)

Automated Disbursement Transaction Header (AD_DOC_HDR)

Automated Disbursement Transaction Vendor (AD_DOC_VEND)

Manual Disbursement Transaction Header (AD_DOC_HDR)

Manual Disbursement Transaction Vendor (AD_DOC_VEND)

Check Writer Payment (CWPYMT / R_AP_CW_PYMT)

Check Status Mapping (CHKMAP / R_CHK_STA_MAP)

Batch Parameters

Parameter	Description	Default Value
Export Location (AMSEXPORT)	The required directory where the positive pay TXT files are written.	$$AMSROOT$$/ ExportImport
Bank (BANK_ACCT_CD)	An optional selection parameter to list one or more banks for positive pay file creation. Separate multiple values with a comma.	No Default
From Cancelled Check Date (CANC_FROM_LAST_DT)	A conditionally required selection parameter necessary when the Check Status parameter is 4, 5, or 7.	No Default
To Cancelled Check Date (CANC_TO_LAST_DT)	A conditionally required selection parameter necessary when the Check Status parameter is 4, 5, or 7.	No Default
Check Status (CHECK_STATUS)	A required selection parameter: (1 - Disbursed, 2 - Warranted, 4 - Cancelled, 5 - Stale Dated, 7 - Escheated). Separate multiple values with a comma.	1, 2, 4, 5, 7
Commit Block Size (COMMIT_BLOCK)	A performance parameter that controls the number of records written to the positive pay file in an update instance.	500
Disbursement Transaction Codes (DISBURSEMENT_DOC_CD)	A required selection parameter of transaction codes. Separate multiple values with a comma.	MD, AD, CW
File Date (FILE_DT)	A conditionally required output parameter for the File Date attribute, when required by the bank.	No Default
File Name Prefix (FILE_NAME_PREFIX)	A required prefix used to name the positive pay files.	POSITIVEPAY
Form Issue Date (FROM_ISSUE_DT)	A conditionally required selection parameter when the check status parameter is 1 or 2.	No Default
To Issue Date (TO_ISSUE_DT)	A conditionally required selection parameter when the check status parameter is 1 or 2.	No Default
From Non-Cancelled Checks	A conditionally required selection	No Default

Date (NON_CANC_FROM_LAST_DT)	parameter for non-cancelled checks when the check status parameter is 1 or 2.	
TO Non-Cancelled Checks Date (NON_CANC_TO_LAST_DT)	A conditionally required selection parameter for non-cancelled checks when the check status parameter is 1 or 2.	No Default
Progression Counter (PROG_CTR_SZ)	A required performance parameter that is used for progression messaging in the job log.	No Default
Selection Block Size (SELECT_BLOCK)	A required performance parameter that controls how many CHREC records are selected in an instance.	No Default
Generate Output File with Run Date (GEN_FILE_WITH_RUN_DT)	A required parameter to identify if the output file will contain a date stamp or not. Valid values are *Y* (yes) and *N* (no). The stamp will appear at the end of the file name in the form of YYYYMMDD. With the stamp, the process will not overwrite the file for a given bank the following day.	Y

Sample Parameter entries:

Choose newly issued checks (non-cancelled)

Status 1,2

From Issue Date (required)

To Issue Date (required)

From Non-Cancelled Checks Date (optional)

To Non-Cancelled Checks Date (optional)

Choose cancelled checks only

Status 4,5,7

From Cancelled Checks (required)

To Cancelled Checks (required)

Choose newly issued checks and checks that were cancelled today

Status 1,2,4,5,7

From Issue Date (required)

To Issue Date (required)

From Non-Cancelled Checks (optional)

To Non-Cancelled Checks (optional)

From Last Action Date of Cancelled Checks Date (required)

To Last Action Date of Cancelled Checks Date (required)

Note: Separate totals cannot be provided for non-cancelled and cancelled disbursements in the same job execution. If totals are required for cancelled and a separate total is required for non-cancelled disbursements, two separate executions are required.

Major Output

Positive Pay text file(s)

Generate Positive Pay Exception Report: The following reason codes will be displayed on the report.

- The check has not been printed and the status is Disbursed.
- The check has not been printed and the status is Warranted.
- Detail field is not present on the Positive Pay Configuration (PPCNFG).

Job Return Code

The following table shows the potential job Return Codes for the Generate Positive Pay job:

Return Code	Condition
Successful (1)	All of the selected records are processed successfully. The job will have a report generated to identify the skipped records. This report assists in identifying the payments that the job first selected and then skipped as part of the selection and update process.
Warning (4)	This return code is issued under the following condition: • No eligible records are found on Check Reconciliation.
Failed (12)	The job may fail under the following conditions: • One or more parameter(s) are invalid. • One or more required parameter(s) are not entered. • Run time exceptions for unexpected situations.
Terminated (16)	This return code is issued when the job is terminated by the user.
System Failure (20)	This return code is issued when the job is terminated because of database server or network issues.

Sort Criteria

Check Reconciliation records selected are sorted by:

- Bank (BANK_ACCT_CD)

- Check/EFT Number (CHK_EFT_NO)
- Transaction Code (DOC_CD)
- Transaction Department Code (DOC_DEPT_CD)
- Transaction ID (DOC_ID)

Selection Criteria

The selection criteria for obtaining records from Check Reconciliation (CHREC) are:

Bank is in the list of values in the Bank Account Code parameter or all Banks are selected when the parameter is left blank.

Positive Pay Template ID on the Bank (BANK) record for each selected Bank Account Code is not blank.

Check/EFT Amount is greater than zero.

Check Status is in the list of values in the Check Status parameter.

Transaction Code is in the list of values in the Disbursement Transaction Code parameter.

Issue Date is between the From Issue Date and To Issue Date parameter. Used to select records where the Check Status is *Disbursed* or *Warranted* and the records have not been processed for Generate Positive Pay (Positive Pay Indication = *false*).

Last Action Date is between the From Non-Cancelled Checks Date and To Non-Cancelled Checks Date parameter, but only for records where the Check Status is *Disbursed* or *Warranted* and the records have not been processed for Generate Positive Pay (Positive Pay indication = *false*).

Last Action Date is between the From Cancelled Check Date and To Cancelled Check Date parameter where the Check Status is *Cancelled*, *Stale Dated*, or *Escheat* and the records have been previously processed by the Generate Positive Pay process (Positive Pay indication = *true*).

Print Status Indicator = Printed or Re-Printed on the latest 'Final' version of the corresponding transaction/record.

Problem Resolution

The job can be scheduled again, if it is interrupted for any reason. The job creates a Positive Pay file for each Bank selected and all the Check Reconciliation records for a given bank are updated to mark them as having been processed for Generate Positive Pay. After the file for each bank is completed and saved in the export directory successfully, all of the Check Reconciliation updates are committed to the database. The next run will ignore all records that are processed successfully and will process the unsuccessfully completed files. Those that were in progress (not completed successfully) are processed from scratch, and are written over with a new file.

The following tables show the possible Return Codes and recommendations for each processing step:

Step 1: Parameter Validation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the parameters are validated successfully.	N/A	N/A
Warning (4)	This step does not issue this return code.	N/A	N/A
Non-Fatal Error (8)	This step does not issue this return code.	N/A	N/A
Failed (12)	All of the required parameters are not entered or are invalid. **Sample Message:** Positive Pay File Name Prefix is required.	If a value is not provided for a required parameter, then enter a valid value for the required parameter and schedule a new job.	N/A
	Entered parameters are not valid. **Sample Message:** Transaction Code 'XX' does not exist on the Transaction Control (DCTRL) table.	If the parameters are invalid, enter a valid value for the parameter and schedule a new job.	N/A
	The job failed because of runtime exceptions for an unexpected situation.	The reason for the failure needs to be investigated before scheduling a new job.	N/A
Terminated (16)	The job is terminated manually by the user.	The reason for the termination needs to be investigated before scheduling a new job.	N/A
System Failure (20)	The job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated before scheduling a new job.	N/A

Step 2: Selection from Check Reconciliation (CHREC)

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the selected records processed successfully.	N/A	N/A
Warning (4)	No records are selected. This is a normal condition.	Confirm the selection criteria and schedule a new job, if necessary.	N/A
Non-Fatal Error (8)	This step does not issue this return code.	N/A	N/A
Failed (12)	The job fails during the attempt to write a TXT file.	Verify that the file system is not full and the job manager has read and write access before scheduling a new job.	N/A
	The job fails because of runtime exceptions for an unexpected situation.	The reason for the failure needs to be investigated before scheduling a new job.	N/A
Terminated (16)	The job is terminated manually by the user.	The reason for the termination needs to be investigated before scheduling a new job.	N/A
System Failure (20)	The job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated before scheduling a new job.	N/A

2.1.26 Generate Warrant Chain

Chain or Job Name	Generate Warrant Chain
Recommended Frequency	On Demand
Single Instance Required	Single
Can be restarted?	No
Reports generated	None

Overview

The Generate Warrant Chain creates Warrant Reconciliation transactions to reclassify warrant payment to cash. The process selects Recently Reconciled records with the Status as *Warranted* from the Check Reconciliation (CHREC) table.

The process uses the disbursement transaction information (Transaction Code, Transaction Dept, and Transaction ID) on the selected records to retrieve the disbursement transaction and generates the Automatic Warrant Reconciliation (AWR) transaction using Multi – threaded functionality. Once the AWR transactions are created, the process submits all transactions and writes all of the rejected transactions to the job log.

Process Steps	Messages
1. Parameter Validation	• Validating Batch Parameters. • Parameters are valid or invalid depending on the validation. If the parameter is invalid, the invalid value will be displayed in the log. • Parameter validation completed
2. Selection of Records	• If the selection returns 0 records, then the following message is issued: “Number of AWR transaction(s) created:0”. • If the CHREC record is found but the corresponding AD/MD is not available on the Transaction Catalog (for all selected records), then the following message is issued: • The Record for DOC_CD: <<DOC_CD>>, DOC_DEPT_CD: <<DOC_DEPT_CD>>, DOC_ID: <<DOC_ID>>, BANK_CD: <<BANK_CD>>, CHK_NO: <<CHK_NO>> is not available in Transaction Calatog. • If the Warrant Reconciliation transaction is already created for the Check no and bank account combination, then the following message is issued: • "Warrant Reconciliation Transaction already generated for the DOC_CD: <<DOC_CD>>,DOC_DEPT_CD: <<DOC_DEPT_CD>>, DOC_ID: <<DOC_ID>> ,BANK_CD: <<BANK_CD>>, CHK_NO: <<CHK_NO>> combinations" • Number of AWR transaction(s) created: record count

Process Steps	Messages
3. Process Records	• Parameters validated • Setting up job for parameter file <AWRSubmitParm<xxx>.txt> • Schedule job with Id= <Job Id> • Enabled job with Id= <Job Id> • Created job <Job Id> for Parm file <AWRSubmitParm<xxx>.txt> • Committed child jobs • Beginning concatenation of exception files • Record stats for job < Job ID> provides the statistics about records loaded and submitted • Total number of records processed for load • Total number of records processed for submitted • Deleting part exception file • Deleting split input xml file • Total number of records processed for Resubmit • Run ended
4. Report Generation	N/A

Major Input

- Check Reconciliation table (R_AP_CHK_RECON)

Generate Warrant Reconciliation Transaction Job:

This step includes selection of eligible records and generation of the AWR xml file.

Batch Parameters

Parameter	Description	Default Value
Transaction Code	The Transaction Code of the transaction that will be generated by the Generate Warrant Reconciliation Transaction process. Required. The default value can be modified. If the Transaction Code is not blank, the system shall validate that the Transaction Code has a Transaction Sub-	AWR

	Type of *WR* on the Transaction Control (DCTRL) table.	
Transaction Department Code	The Transaction Department used to create the Warrant Reconciliation transactions. Required. Should be valid on the Department table and should be valid for the current application date's Fiscal Year on the Department Fiscal Year Controls (DEPTFY) table.	No Default
Transaction Prefix	This will be the Prefix for the Warrant Reconciliation Transaction ID. Required. The value should be available on the ADNT table for the Transaction Code and Transaction Department Code for the Current Fiscal Year.	No Default
Transaction Unit Code	The Transaction Unit used to create the Warrant Reconciliation transactions. Should be valid. Required. Transaction Unit is a required field if "Require Transaction Unit Code" flag is set to true for the transaction code on DCTRL; otherwise, it is an optional field. If entered, it should be valid on the Unit table for the entered Transaction Department.	No Default
Override Level	Override level used in the Submit job step.	No Default

	Optional. If entered, must be an integer between 0 and 10.	
Commit Block	Commit Block size Optional field. The value should be a positive integer. If blank or invalid, it defaults to 1000.	No Default

Major Output

Automated Warrant Reconciliation (AWR) transaction.

AWR Load and Submit Job:

This step employs the Multi-Threaded Transaction Loader utility to import and submit the AWR transactions.

The AWR Load and Submit job splits large Advantage transaction files into smaller sized transaction files. The application only accepts an XML file as input. Splitting is based on the following input parameters:

1. Block size - The number of Advantage Transaction blocks to be written to a split file before the write moves to the next set of files.
2. Thread Count - The total number of split files to be created.

For example: If an input file has 31 Advantage table records and the Block size is 5 and the thread count is 3, then 3 split files are created and the first 5 records from the input file are written to the first split file, and the next 5 written to the next split file, and so forth. Therefore, when completed, the first file would contain 11 table records, and the second and third files would contain 10 each (the first five records are written to file 1, the next five to file 2, the next five to file 3, the next five to file 1 again, and so on. As a result, the final record is written to file 1 resulting in eleven records in file 1, and 10 records in files 2 and 3).

Parameter	Description	Default Value
FILE_LIST	Comma separated list of input XML files	AWRDocument_$$@CHAINJOBID@$$.xml
FILE_PREFIX	File Prefix for Parameter and Data Files created by jobs	AWRSubmit
I_SMU_APPLY_OVERRIDES	Apply Overrides	True
I_SMU_BYPS_ADNT_FL	SysmanUtil Import Parameter - ByPass ADNT	True
I_SMU_OVERRIDE_LVL	Override Level	10
MODE	Mode (1=Import only,2-Import & Submit, 3-	2

	Import, Other Action, Submit)	
SMU_CTLG_ID	Catalog ID of the System Maintenance Utility job that is spawned as the child process.	3
STAGGER_TIME	The lag time, in seconds, between the spawning of each child process.	0
S_SMU_EXCEP_REP_FILE_NM	AWR Submit Exception File Name	ExpRepAWRSubmit.txt
THREAD_COUNT	Number of Jobs to Start	
LOG_STATUS_INTERVAL	Logging frequency (in seconds) for controller thread reporting status of child threads to the system log.	300 Seconds (5 minutes). Cannot be less than 300 seconds. The system will use the default of 300 if a value less than 300 is specified.
SLEEP_INTERVAL	Polling frequency (in seconds) for internal controller thread for checking child processes.	5 seconds. Cannot be less than 5 seconds. The system will use the default value of 5 if a value less than 5 is specified.

Note: The default values listed are those delivered with the software. Actual values may vary based on your site's setup.

Chain Job Return Code

The following table shows the potential job Return Codes for the Generate Warrant Chain:

Return Code	Condition
Successful (1)	All of the selected records are processed successfully.
Warning (4)	No eligible records found. This could be because of the following reason: • No matching records found in the CHREC table.
Non-Fatal Error (8)	Out of selected records from the CHREC table, for at least one record, the corresponding AWR transaction could not be submitted to *Final*. For Example: • Corresponding AD/MD transactions for the CHREC records are not present on the Transaction Catalog. • DC transactions for the combination of DOC_CD, DOC_DEPT_CD, DOC_ID, CHECK_NO, and

	BANK_ACCT_CD are already created. • The newly created AWR transaction could not be submitted.
Failed (12)	The job may fail under the following conditions: • Parameters are invalid. • Commit transaction is not successful. • Run time exceptions for unexpected situations.
Terminated (16)	This Return Code is issued when the job is terminated by the user.
System Failure (20)	This Return Code is issued when the job is terminated because of database server or network issues.

Sort Criteria

The selected records are ordered by CHK_EFT_NO.

Selection Criteria

The process selects all Check Reconciliation (R_AP_CHK_RECON) table records that meet the following selection criteria:

- Status = *Warranted*,
- Recently Reconciled flag = *True*, and
- Transaction Code with Sub-Type = *AD/MD*

Problem Resolution

The job cannot be restarted on failure; however, a new job can be rescheduled after fixing the issue. The following table shows the possible Return Codes and recommendations for each processing step:

Step 1: Parameter Validation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A

Failed (12)	The required parameters are not entered. **Sample Message:** Transaction Unit entered is either invalid, inactive or not within the effective date range.	Enter a unit that is valid and active.	
	The job failed because of runtime exceptions for an unexpected situation.	The reason for the failure needs to be investigated before rescheduling the job.	
Terminated (16)	The job is terminated manually by the user.	The reason for the termination needs to be investigated. Then the job can be rescheduled.	
System Failure (20)	The job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. Then the job can be rescheduled.	

Step 2: Selection of Records

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	If the selection returns 0 records, then the following message is issued: **Sample Message:** "No eligible records found on the CHREC table."	Verify CHREC records against the selection criteria and set up records accordingly before rescheduling the job.	
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	If the CHREC record is found, but the corresponding AD/MD transaction is not available on the Transaction Catalog (for all selected records), then the following message	Ensure that the appropriate AD/MD transactions exist for further processing.	Reschedule the job with appropriate records on CHREC.

	is issued: **Sample Message:** "The Record for DOC_CD: <<DOC_CD>>, DOC_DEPT_CD: <<DOC_DEPT_CD>>, DOC_ID: <<DOC_ID>>, BANK_CD: <<BANK_CD>>, CHK_NO: <<CHK_NO>> is not available in Transaction Catalog."		
	The job failed because of runtime exceptions for an unexpected situation.	The reason for the failure needs to be investigated before rescheduling the job.	Reschedule the job with appropriate parameter values.
Terminated (16)	The job is terminated manually by the user.	The reason for the termination needs to be investigated. Then a new job can be scheduled.	Schedule a new job.
System Failure (20)	The job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. Then a new job can be scheduled.	Schedule a new job.

Step 3: Processing of Records

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All the transactions were processed and submitted successfully.	N/A	N/A
Warning (4)	No eligible records were found.	Check the parameter values.	N/A
Non-Fatal Error (8)	This Return Code will be issued when at least one AWR transaction fails to submit to *Final*.	N/A	N/A

Possible Return Codes	Condition	Recommendation	Other Instructions
Failed (12)	The job failed due to fatal conditions.	If the job fails because of runtime exceptions, investigate the exception reported by the process, resolve the error and reschedule the job.	Schedule a new job
Terminated (16)	The job is terminated manually by the user.	The reason for the termination needs to be investigated. Then a new job can be scheduled.	Schedule a new job
System Failure (20)	The job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. Then a new job can be scheduled.	Schedule a new job

2.1.27 Initiate Backup Withholding

Chain or Job Name	Initiate Backup Withholding
Recommended Frequency	Twice a year
Single Instance Required	Single
Can be restarted?	Yes
Reports generated	Initiate Backup Withholding report

Overview

The Initiate Backup Withholding batch process automatically initiates Backup Withholding for payees and vendors within appropriate timeframes according to IRS regulations. The process selects applicable records from the 1099I table, for which the number of calendar days since the last B Notice was sent is greater than or equal to the number of Compliance Days (the number of Compliance Days is defined on 1099P table for the current Calendar Year).

The process updates selected taxpayer records on the 1099I table as subject for Backup Withholding (i.e. set the '1099 Backup Withholding Status' CVL to "Eligible") to start withholding associated payments.

In addition to updating appropriate records on the 1099I table, the batch process generates a Backup Withholding Initialization report. This report lists all records that were selected from the 1099I and flagged as subjects to Backup Withholding (this means the report reflects the state of the records after they have been updated by the Initiate Backup Withholding job). The record information listed in the report is inferred from the 1099I table and includes the following columns:

- TIN
- TIN Type
- TIN Notice Date
- TIN Notice Counter
- Days since last TIN Notice
- Withholding Tracking Start Date

The process will update tax payer records on the 1099I table. The process will involve the following steps.

- Parameter Validation
- Selection of records from 1099I table
- Update the records in 1099I table
- Generate Initiate Backup Withholding Report

Process Steps	Messages
1. Parameter Validation	• Validating Batch Parameters • Parameters are valid or invalid depending on the Validation. If the parameter is invalid, the invalid value will be displayed in the log. • Batch Parameter validation completed
2. Process records	• Select Records from 1099I • Update the records in 1099I table
3. Generate report	• Generate Initiate Backup Withholding Report

The Job will run as a Single Instance with a frequency of twice a year.

Major Input

- 1099I table

Major Output

The following will be the output for this job:

- Records will be updated in 1099I table
- Initiate Backup Withholding Report. The report will be sorted by TIN and TIN Type. This report also includes the 'Total number of 1099I records selected to start Backup Withholding'

Sort Criteria

N/A

Selection Criteria

The process will select all **1099 Reporting Information table** records that meet the following selection criteria:

- The 1099 Backup Withholding Status = Pending, AND
- The TIN Notice Counter is not Empty AND
- The number of days since the notice was sent is greater than or equal to the number of compliance days.

Problem Resolution

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All the parameters are validated successfully	N/A	N/A
Warning (4)	No records were found eligible	Check the parameter values.	N/A
Non-Fatal Error (8)	This return code will be issued when the validation fails in this step	N/A	N/A
Failed (12)	Job failed due to Fatal conditions	If the job fails because of the runtime exceptions, investigate the exception reported by the process, resolve the error and restart the job.	Schedule a new job
	Failed while restarting the job since another instance of the job has already been run successfully. Cannot restart the job since another instance of this job has already been run successfully.	**Recommendation:** Schedule a new job.	
Terminated (16)	Job is terminated manually by the user.	Reason for the termination needs to be investigated. The job can either be restarted or schedule a new job.	Schedule a new job
System Failure (20)	When the job is terminated because of database server or network issues	Reason for the System Failure needs to be investigated. The job can either be restarted or schedule a new job.	Schedule a new job

Possible Return Code

Return Code	Condition
Successful (1)	All the selected records are processed successfully
Warning (4)	No eligible records found. This could be because of the following reasons: • No matching records found in 1099I
Non-Fatal Error (8)	
Failed (12)	The job will fail under the following conditions: • Parameters are invalid • Unable to Update Records in 1099I table • Run time exceptions for unexpected situations. When this job ends with a return of code *Failed,* subsequent jobs in the chain will be set to *Inactive*.
Terminated (16)	This return code will be issued when the job is terminated by the user. When this job ends with a return of code *Terminated* subsequent jobs in the chain will be set to *Inactive*.
System Failure (20)	This return code will be issued when the job is terminated because of database server or network issues. When this job ends with a return of code *System Failure*, subsequent jobs in the chain will be set to *Inactive*.

Recommendation: If the job failed due to some invalid parameters, the recommendation should be to correct parameter and restart the job or Schedule a new job.

2.1.28 IRS TIN/Name Match Extract

Job Name	IRS TIN/Name Match Extract
Recommended Frequency	The IRS TIN/Name Match Extract batch job can be run daily as part of the nightly cycle or on demand.
Single Instance Required	Yes.
Can be restarted?	Yes.
Reports generated	Yes. The IRS TIN/Name Match Extract Selection Report is generated.

Overview

The IRS TIN/Name Match Extract batch job in CGI Advantage Financial extracts Taxpayer ID/Name information from vendor registration and creates a bulk file to be submitted to the IRS for matching the IRS records using the IRS TIN/Name Match e-Service.

This extract process selects all records from the 1099 Reporting Information (1099I) table where Taxpayer ID (TIN) and TIN Type are populated and the Send to IRS and 1099 or 1042-S Reportable flags are checked.

Each TIN, TIN Type and Name combination that is extracted is written to a text file to be sent to the IRS. The Name selected from 1099I table will be modified to remove special characters that are not accepted by the IRS. The IRS TIN/Name Match Extract Selection report produced by the extract process will list all 1099I entries selected for a given run and include a note for any records where the Name was revised to match the IRS special character requirements. Once a record is successfully extracted, the extraction process updates 1099I to uncheck the 'Send to IRS' flag.

This job can be restarted if it is failed. If the job is restarted, it will start from the step where it failed.

The following table shows the various steps that the IRS TIN/Name Match Extract batch job goes through and the messages issued at each step.

Process Steps	Messages
1. Parameter Validation	• Validating Batch Parameters • If the 'IRS Special Characters' parameter is not provided then "Missing IRS Allowable Special Characters batch parameter" will be displayed in the log. • If the values in the 'IRS Special Characters' parameter are not in form of a 'comma separated' list of values then ''Invalid list of special characters provided in the IRS Allowable Special Characters batch parameter" will be displayed in the log. • If 'Maximum Records to Select' parameter is not provided then "'Missing Maximum Records to Select batch parameter" will be displayed in the log. • If 'Maximum Records to Select' parameter is not a valid

Process Steps	Messages
	integer greater than zero then ’‘Invalid value specified in the Maximum Records to Select batch parameter’ will be displayed in the log. • If ‘Commit Size’ parameter is not provided then “Commit Size is defaulted to 1000” will be displayed in the log. • If ‘Commit Size’ parameter is not a valid integer greater than zero then ’‘Commit Size must be a positive whole number greater than zero” will be displayed in the log • If ‘Progression Counter’ parameter is not provided then “Progression Counter’ is defaulted to 1000” will be displayed in the log. • If ‘Progression Counter’ parameter is not a valid integer greater than zero then “Progression Counter’ must be a positive whole number greater than zero” will be displayed in the log. • If ‘AMSEXPORT ‘parameter is not provided then “‘The AMSEXPORT Parameter File cannot be empty” will be displayed in the log. • If ‘AMSEXPORT‘ parameter is not a valid directory then ’‘The value entered in the AMSEXPORT parameter is not a valid directory” will be displayed in the log. • If ‘AMSPARM‘ parameter is not provided then “‘The AMSPARM Parameter File cannot be empty” will be displayed in the log. • If ‘AMSPARM‘ parameter is not a valid directory then ’‘The value entered in the AMSPARM parameter is not a valid directory” will be displayed in the log. • Parameter validation completed.
2. Selection of Records	• Selecting eligible records. • Selection of eligible records completed.
3. Processing Of Records	• Processing the selected records. • If <<Total Count>> is greater than Maximum Records to Select then “The maximum number of records to select has been reached.” will be displayed in the log. • Depending on the progression Counter size the ‘<<Progression Count>> record(s) processed’ may be displayed in the log. • If no records are selected to process then ‘No records found to processes’ will be displayed in the log. • Total <<Total Count>> record(s) are processed. • Processing completed.

When to Run

The IRS TIN/Name Match Extract batch process should be run to extract Taxpayer ID information from vendor registration information in Advantage.

Restartability Information:

This process is implemented with checkpoints. If the process fails for any reason (such as the network is down or the server is down), then the process restarts from the last point where it stopped.

If the job fails in any of the above steps the job can be restarted after resolving the error. If the job is restarted, it will start from the last point where it stopped.

For example, the job failed after processing 2 records from the total of 25 eligible records due to some fatal condition (table space error). If the job is restarted after resolving the table space issue, the job will start by selecting the records from the 3rd transaction.

Instead of restarting the job, if the job is rescheduled with the same parameter value after resolving the table space issue, the job will start from the beginning.

The restart will fail if another instance of this job has been scheduled and ran successfully before restarting the failed job.

Major Input

- 1099 Information Table (R_TIN_1099_INFO)

Batch Parameters

Parameter	Description	Default Value
AMSEXPORT	Export Location at IRS TIN/Name Match Extract Selection Job Defaulted to pre-defined value.	$$AMSROOT$$/ExportImport
AMSPARM	Parameter Location at IRS TIN/Name Match Extract Selection Job Defaulted to pre-defined value.	$$AMSROOT$$/Parms
IRS_ALLOW_SPEC_CHAR	IRS Allowable Special Characters Required Field. Special Characters separated with commas.	
MAX_REC_TO_SEL	Maximum Records to	

	Select Required Field. Any numeric value greater than zero (Note: The IRS currently supports a maximum of 100,000 entries on the output file that is created by this batch process and manually uploaded to the IRS TIN/Name Match e-Service. As such, the value entered for this parameter should not be greater than 100,000).	
COMMIT_SIZE	Commit block size Optional field. The value for this parameter should be a positive integer. If not entered, it is defaulted to 1000. This parameter is required to control progression messages detailing the progress of record processing.	1000
PROG_CTR_SIZE	Progression Counter Optional field. The value for this parameter should be a positive integer. If not entered, it is defaulted to 1000. This parameter is used to show the progress of record processing.	1000

Major Output

- IRS TIN Name Match <<YYYYMMDD>>.txt
- IRS TIN/Name Match Extract Selection Report

Job Return Code

The following table shows the potential return codes for the IRS TIN/Name Match Extract Selection job.

Return Code	Condition
Successful (1)	All of the selected records are processed successfully by adding records into Text file, adding records into Reports and updating records for Send To IRS is Unchecked.
Warning (4)	No records found to process. This could be because of the following reason: • No records found on 1099 Reporting Information (R_TIN_1099_INFO) table.
Non-Fatal Error (8)	The maximum number of records to select has been reached. This could be because of the following reason: • Total Record Count greater than Maximum Records to Select.
Failed (12)	The job will fail under the following conditions: • Parameters are invalid. • Unable to update Send to IRS on processed Record-1099 Reporting Information (1099I) record. • Run time exceptions for unexpected situations.
Terminated (16)	This return code will be issued when the job is terminated by the user.
System Failure (20)	This return code will be issued when the job is terminated because of database server or network issues.

Sort Sequence

Taxpayer ID (TIN), TIN Type (TIN_TYP) ascending order

Selection Criteria

Select records from the 1099 Reporting Information (1099I) table where Taxpayer ID (TIN) and TIN Type are populated and the Send to IRS and 1099 or 1042-S Reportable flags are checked.

Problem Resolution

The following table shows the possible return codes and recommendations for each processing step.

Parameter Validation:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful	Successful	N/A	N/A

(1)			
Warning (4)	N/A	This step does not issue this return code.	This step does not issue this return code.
Non-Fatal Error (8)	N/A	This step does not issue this return code.	This step does not issue this return code.
Failed (12)	Required Parameters are not entered **Sample Message:** Missing IRS Allowable Special Characters batch parameter.	Enter the IRS Allowable Special Characters.	Alternatively, the job can be rescheduled after correcting the parameters.
	Failed because of runtime exceptions for an unexpected situation.	The reason for the failure needs to be investigated before restarting the job.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can either be restarted or rescheduled.	
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can either be restarted or rescheduled.	

Processing Of Records

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the selected 1099 Information records are processed successfully.	N/A	N/A
Warning (4)	No records found to process.	N/A	N/A

Possible Return Codes	Condition	Recommendation	Other Instructions
Non-Fatal Error (8)	The maximum number of records to select has been reached.	N/A	N/A
Failed (12)	Job failed due to Fatal conditions.	In this step, the job can fail under the following two conditions. 1) Encounters any runtime exceptions and 2) Failed during restart. If the job fails because of the runtime exceptions, investigate the exception reported by the process, resolve the error and restart the job.	Schedule a new job.
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. A new job can be scheduled.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only a new job should be scheduled.
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. A new job can be scheduled.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only a new job should be scheduled.

2.1.29 IRS TIN/Name Match Update

Chain or Job Name	IRS TIN/Name Match Update
Recommended Frequency	The IRS TIN/Name Match Update batch job can be run daily as part of the nightly cycle or on demand.
Single Instance Required	Yes.
Can be restarted?	Yes.
Reports generated	Yes. IRS TIN/Name Match Update Exception Report is generated.

Overview

The IRS TIN/Name Match Update batch job in CGI Advantage Financial updates entries on the 1099 Reporting Information table based on the IRS TIN/Name Match Status file that is returned from the IRS TIN/Name Match e-service.

This job reads data from the Input file and the Update IRS Match Status, IRS Match Status Date, and IRS Tracking Number on the 1099 Reporting Information (R_TIN_1099_INFO) table based on Taxpayer ID and TIN Type.

The record will be added to the 'IRS TIN/Name Match Update Exception' report when:

TIN Name on the input file and the TIN Name on the 1099 Reporting Information (R_TIN_1099_INFO) table (after scrubbing any special characters) are found different.

Taxpayer ID, TIN Type combination not found on 1099 Reporting Information (R_TIN_1099_INFO) table.

IRS Match Status is any value other than 'Name/TIN combination matches IRS records'.

This job can be restarted if it is failed. If the job is restarted, it will start from the step where it failed.

The following table shows the various steps that the IRS TIN/Name Match Update batch job goes through and the messages issued at each step.

Process Steps	Messages
1. Parameter Validation	• Validating Batch Parameters • If 'IRS Special Characters' parameter is not provided then "Missing IRS Allowable Special Characters batch parameter" will be displayed in the log. • If values in the 'IRS Special Characters' parameter are not in the form of a 'comma separated' list of values then ''Invalid list of special characters provided in the IRS Allowable Special Characters batch parameter" will be displayed in the log. • If 'IRS Tracking Number' parameter is not provided then "IRS Tracking Number is defaulted to <<Job ID>>" will be displayed in the log.

Process Steps	Messages
	• If 'IRS Tracking Number' parameter is not a valid integer greater than zero then " Invalid IRS Tracking Number'' will be displayed in the log. • If 'Parm File' parameter is not provided then "Parameter file name is not specified" will be displayed in the log. • If 'Parm File' parameter does not end with ".txt ", then "Parameter file should have file extension of .txt '' will be displayed in the log. • If 'Commit Size' parameter is not provided then "Commit Size is defaulted to 1000" will be displayed in the log. • If 'Commit Size' parameter is not a valid integer greater than zero then ''Commit Size must be a positive whole number greater than zero" will be displayed in the log. • If 'Progression Counter' parameter is not provided then "Progression Counter is defaulted to 1000" will be displayed in the log. • If 'Progression Counter' parameter is not a valid integer greater than zero then "Progression Counter must be a positive whole number greater than zero" will be displayed in the log. • If 'AMSEXPORT' parameter is not provided then "'The AMSEXPORT Parameter File cannot be empty" will be displayed in the log. • If 'AMSEXPORT' parameter is not a valid directory then ''The value entered in the AMSEXPORT parameter is not a valid directory" will be displayed in the log. • If 'AMSPARM' parameter is not provided then "'The AMSPARM Parameter File cannot be empty" will be displayed in the log. • If 'AMSPARM' parameter is not a valid directory then ''The value entered in the AMSPARM parameter is not a valid directory" will be displayed in the log. • Parameter validation completed.
2. Processing Of Records	• Start reading records from the Input file. • If any record on the input file does not have all column details populated, then "Less than 5 fields were detected on the input file. Expecting data in the first, second, third, fourth and fifth fields for TIN Type, TIN, Name, Account Number and IRS Match Status respectively" will be displayed in the log. • Depending on the progression Counter size the '<<Progression Count>> record(s) processed' will be displayed in the log. • If no records are selected to process then 'No records found

Process Steps	Messages
	to processes' will be displayed in the log. • Total <<Total Count>> record(s) are processed. • Reading completed.

When to Run

N/A

Restartability Information:

This process is implemented with checkpoints. If the process fails for any reason (such as the network is down or the server is down), then the process restarts from the last point where it stopped.

If the job fails in any of the above steps the job can be restarted after resolving the error. If the job is restarted, it will start from the last point where it stopped.

For example, the job failed after processing 2 records from the total of 25 eligible records due to some fatal condition (table space error). If the job is restarted after resolving the table space issue, the job will start by selecting the records from the 3rd transaction.

Instead of restarting the job, if the job is rescheduled with the same parameter value after resolving the table space issue, the job will start from the beginning.

The restart will fail if another instance of this job has been scheduled and ran successfully before restarting the failed job.

Major Input

- IRS TIN Name Match <<YYYYMMDD>>.txt

Batch Parameters

Parameter	Description	Default Value
AMSEXPORT	Export Location of IRS TIN/Name Match Update Job Defaulted to pre-defined value.	$$AMSROOT$$/ExportImport
PARM_FILE	File Name Required. Any file name that ends with .txt is allowed.	
IRS_ALLOW_SPEC_CHAR	IRS Allowable Special Characters Required Field. Special Characters separated with commas.	

IRS_TRKG_NO	IRS Tracking Number Optional. Any numeric value greater than zero is allowed.	
COMMIT_SIZE	Commit block size Optional field. The value for this parameter should be a positive integer. If not entered, it is defaulted to 1000. This parameter is required to control progression messages detailing the progress of record processing.	1000
PROG_CTR_SIZE	Progression Counter Optional field. The value for this parameter should be a positive integer. If not entered, it is defaulted to 1000. This parameter is used to show the progress of record processing.	1000

Major Output

- IRS TIN/Name Match Update Exception Report
- 1099 Information Table (R_TIN_1099_INFO)

Sort Sequence

N/A

Selection Criteria

N/A

Job Return code

The following table shows the potential return codes for the IRS TIN/Name Match Update job.

Return Code	Condition
Successful (1)	N/A
Warning (4)	No records found to process. This could be because the <<Input file >> is empty.

Return Code	Condition
Non-Fatal Error (8)	Total <<Total Count>> record(s) are processed. This could be because of ”Total Record Count greater than equal to one “
Failed (12)	The job will fail under the following conditions: • Parameters are invalid • Unable to update IRS Match Status, IRS Match Status date, and IRS Tracking Number on Processing Record-1099 Information (1099I) record. • Run time exceptions for unexpected situations.
Terminated (16)	This return code will be issued when the job is terminated by the user.
System Failure (20)	This return code will be issued when the job is terminated because of database server or network issues.

Problem Resolution

The following table shows the possible return codes and recommendations for each processing step.

Parameter Validation:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	N/A	This step does not issue this return code.	This step does not issue this return code.
Non-Fatal Error (8)	N/A	This step does not issue this return code.	This step does not issue this return code.
Failed (12)	Required Parameters are not entered **Sample Message:** Missing IRS Allowable Special Characters batch parameter.	Enter the IRS Allowable Special Characters.	Alternatively, the job can be rescheduled after correcting the parameters.

	Failed because of runtime exceptions for an unexpected situation.	The reason for the failure needs to be investigated before restarting the job.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can either be restarted or rescheduled.	
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can either be restarted or rescheduled.	

Processing Of Records

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	N/A	N/A	N/A
Warning (4)	No records found to process	N/A	N/A
Non-Fatal Error (8)	Processing Records more than and equal to one	N/A	N/A
Failed (12)	Job failed due to Fatal conditions	In this step, the job can fail under the following two conditions. 1) Encounters any runtime exceptions and 2) Failed during restart. If the job fails because of the runtime exceptions, investigate the exception reported by the process, resolve the error and restart the job.	Schedule a new job.
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. A new job can be scheduled	If another instance of the job has already been scheduled and ran successfully,

Possible Return Codes	Condition	Recommendation	Other Instructions
			then this job should not be restarted – only a new job should be scheduled.
System Failure (20)	When the job is terminated because of database server or network issues.	Reason for the System Failure needs to be investigated. A new job can be scheduled.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only a new job should be scheduled.

2.1.30 IRS Transmittal File Generation

Job Name	IRS Transmittal File Generation
Recommended Frequency	On Demand
Single Instance Required	Yes
Can be restarted?	No
Reports generated	IRS Transmittal Report (1042-S) IRS Transmittal Report (1099)

Overview

This job will generate data files of IRS 1099 and/or 1042-S tax information that conform to the IRS specifications (see publications 1187 and 1220) for transmitting to the IRS through the IRS Filing Information Returns Electronically (FIRE) Production system. IRS Transmittal reports list summary information from the generated IRS Transmittal files.

The job involves the following steps:

1. Parameter Validation
2. Selection from the Form 1099/1042-S Reporting tables

The job will create the transmittal file based on the Transmittal File Submission Type batch parameter. The Transmission Type from the Form Reporting tables specifies whether original data or corrected data will be created for the record. If original records and corrected records need to be transmitted, the IRS Transmittal File Generation batch must be scheduled twice; once for original records and another for corrected records.

After all the records are written to the transmittal file, the associated Last Transmittal Date and Last Transmittal Run ID are updated for each record written to the Transmittal file.

Process Steps	Messages
1. Parameter Validation	• Validating Batch Parameters • Parameters are valid or invalid depending on the Validation. If the parameter is invalid, the invalid value will be displayed in the log. • Batch Parameter validation completed
2. Selection of Records	• Selecting eligible records • If the selection returns 0 records, then the following message will be issued: “No eligible record found”. • Number of records (count) selected will be displayed. • At the end, the following message

Process Steps	Messages
	will be issued: “Selection of records completed.”

Restartability Information

This job may not be restarted. No special action is needed when a job is terminated before scheduling a new job, because any existing files created during an interrupted job will be overwritten, and no important database updates are made until all files are created. Records are marked as *In Progress* during the run, and all *In Progress* records have Last Run ID and Last Transmitted Date set after the job has finished record selection and all files are created successfully. All records marked as *In Progress* are unmarked at the start of the job and when the job ends successfully.

This job can only be run as a single thread to ensure that no other job running will compete for selection and update of the same records.

Major Input

- Form 1042-S Reporting (R_FORM_1042S_RPT) table
- Form 1099-A Reporting (R_FORM_1099A_RPT) table
- Form 1099-C Reporting (R_FORM_1099C_RPT) table
- Form 1099-G Reporting (R_FORM_1099G_RPT) table
- Form 1099-INT Reporting (R_FORM_1099INT_RPT) table
- Form 1099-MISC Reporting (R_FORM_1099MIS_RPT) table
- Form 1099-NEC Reporting (R_FORM_1099NEC_RPT) table
- Form 1099-R Reporting (R_FORM_1099R_RPT) table
- Form 1099-S Reporting (R_FORM_1099S_RPT) table
- Tax Form Field Box Cross Reference (R_TAX_FLD_BOX_REF) TAXBXRF table
- 1099 Reporting Payer Information (R_1099_RPT_PYRINFO) table

Batch Parameters

Note: The default values listed are those delivered with the software. Actual values may vary based on your site’s setup.

Parameter	Description	Default Value
AMSEXPORT	Export Location at IRS Transmittal File Generation job. Required. This must be a valid directory and it is used to determine where the files will be written.	$$AMSROOT$$/ExportImport
APPLY_FED_THRESHOLD	Apply Federal Threshold	1

	Required. Valid values are '1' (Yes) or '2' (No). This is used to filter records during selection.	
FORM_TYPE	Form Type Optional. This is a list separated by commas, each must be one of the following values in upper case: • 'N' – 1042S • 'E' – 1099N • 'G' – 1099G • 'I' – 1099I • 'M' – 1099M • 'S' – 1099S • 'A' – 1099A • 'C' – 1099C • 'R' – 1099R If blank, all form types are eligible for selection. If entered, only form types entered are eligible for selection. This is used to filter records during selection.	No Default
IRS_TRANS_FILE_NM	IRS Transmittal File Name (Optional) Must be less than or equal to 17 characters. A three-digit sequence starting with 001 will be added. This is used to filter records during selection. If a value is not provided, the file name will be generated in the format 'CCYYAAAAAFTNNNNN_<run ID>.txt'. The values used are: CCYY – Tax Year AAAAA – Transmittal Control Code F – Tax Form (for example, 'I' for 1099-INT) T – Submission Type ('O' for Original, 'C' for Correction,	No Default

	'R' for Replacement, and 'T' for Test) NNNNN – File Sequence, starting with 00001 <run ID> – Run ID for the current job	
ONLY_SELECT_1099M_BOX7 (Valid only for Tax Year 2018 and 2019)	Only Select 1099-MISC, Box 7 Records should be set to *Y* if reporting non-employee compensation only. Optional. Valid values are *Y* or *N*. This is used to filter records during selection.	N
PRIOR_TAX_YEAR	Prior Tax Year if running the job for a prior tax year. Conditionally Required. Required if Tax Year is not entered. Must be valid on Fiscal Year (FY) table. This is used to filter records during selection.	No Default
RPT_1099_PYR	Reporting Payer Optional. This is a list separated by commas, each must be a valid value in the 1099 Reporting Payer Information (1099RP) table. When this parameter is provided, the Transmittal Control Code (TCC) parameter must be blank. This is used to filter records during selection.	No Default
TAX_YEAR	Tax Year (most current) Conditionally Required. Required if Prior Tax Year is not entered. Must be valid on Fiscal Year (FY) table. This is used to filter records during selection.	No Default
TRANS_FILE_TYPE	Transmittal File Submission Type Required. This is a list separated by commas, each must be one of the following values:	No Default

	'1' – Original '2' – Correction/Amended '3' – Replacement '4' – Test If *3* or *4* is chosen, no other value may be in the list. This is used to filter records during selection.	
TRANS_RUN_ID	Transmitted Run ID Required when the Transmittal File Submission Type parameter is '3' (Replacement). This is used to filter records during selection.	No Default
TRNS_CTRL_CD	Transmittal Control Code (TCC) Optional. This is a list separated by commas, each must be a valid value in the 1099 Transmitter Information (1099TI) table. When this parameter is provided, the Reporting Payer parameter must be blank. Please note that 1099TI contains records for 1042-S as well as all 1099 tables. This is used to filter records during selection.	No Default

Major Output

- IRS Transmittal Files (text)
- Form 1042-S Reporting (R_FORM_1042S_RPT) table
- Form 1099-A Reporting (R_FORM_1099A_RPT) table
- Form 1099-C Reporting (R_FORM_1099C_RPT) table
- Form 1099-G Reporting (R_FORM_1099G_RPT) table
- Form 1099-INT Reporting (R_FORM_1099INT_RPT) table
- Form 1099-MISC Reporting (R_FORM_1099MIS_RPT) table
- Form 1099-NEC Reporting (R_FORM_1099NEC_RPT) table
- Form 1099-R Reporting (R_FORM_1099R_RPT) table
- Form 1099-S Reporting (R_FORM_1099S_RPT) table

Job Return Code

The following table shows the potential job Return Codes for the IRS Transmittal File Generation job.

Return Code	Condition
Successful (1)	All of the selected Form tables/records are processed successfully.
Warning (4)	No eligible records found. This could be because of the following reasons: The records may have been selected and processed in an earlier run. Filter parameters were set so that no records met the criteria.
Non-Fatal Error (8)	This Return Code is not returned by this job.
Failed (12)	The job may fail under the following conditions: One (or more) of the job parameters is invalid. Insufficient disk space in the Export directory.
Terminated (16)	This Return Code is issued when the job is terminated by the user. When this job ends with a Return Code of *Terminated*, subsequent jobs in the chain are set to *Inactive*.
System Failure (20)	This Return Code is issued when the job is terminated because of database server or network issues. When this job ends with a Return Code of *System Failure*, subsequent jobs in the chain are set to *Inactive*.

Sort Criteria

Each Form table is selected individually.

For each table, records are retrieved in order by Transaction Control Code (TCC) and Correction Type, and each time one of these values changes, a new file is started.

Within each 1042-S file, records are grouped in order by Withholding Agent and Chapter Indicator.

Within each 1099 file, records are grouped in order by Payer.

Inside each group, records are ordered by the Payee's Account Number.

Selection Criteria

The 1042-S and/or 1099 tables are selected according to the Form Type parameter. If the Form Type parameter is blank, selection will be made from all form type tables.

From each selected table, records are filtered when:

The Tax Year parameter matches Tax Year in the Form record.

The Transmittal Control Code (TCC) parameter matches TCC in the 1099 Reporting Payer Information (1099RP) table, which is then joined to the Form Record by Reporting Payer. Please note that even though the 1099RP table indicates "1099", TCC values for the 1042-S Form table are also included. When this parameter is blank, no records are filtered.

The Reporting Payer parameter matches Reporting Payer in the Form record. When this parameter is blank, no records are filtered.

Apply Federal Threshold parameter is set to *1* (Yes) and Threshold Met in the Form record is checked. When this parameter is set to *2* (No), no records are filtered.

(For Tax Year 2018 and 2019 only) Only Select 1099-MISC, Box 7 Records is set to *Y* then records from F1099M will be selected where income reported is greater than zero. If this parameter is set to *N*, no records are filtered for these criteria.

Records are also filtered by Transmittal File Submission Type parameter value:

Original: Form records are filtered where Transmission Type is *Original*, Current Record flag is *checked* (Yes), and the Last Transmitted Date field is blank.

Correction/Amended: Form records are filtered where Transmission Type is *Corrected* or *Amended*, Current Record is *checked* (Yes), and the Last Transmitted Date field is blank.

Replacement: Form records are filtered where the Transmitted Run ID parameter matches Last Transmitted Run ID in the Form record and the Last Transmitted Date field is not blank.

Test: Form records are filtered where Transmission Type is *Original*, Current Record is *checked* (Yes), and the Last Transmitted Date is blank. A file with at least 11 "B"/"Q" records can be used to test that the file is formatted to IRS specifications and by sending it to the IRS through their Filing Information Returns Electronically (FIRE) system.

Valid Characters

IRS provides some guidelines on valid characters on recipient fields. For specific field guidance, refer to the IRS publications. Advantage allows these characters other than alphanumeric values:

- Name fields: blank, ampersand (&), and hyphen (-)
- Address fields: blank, ampersand (&), hyphen (-), comma (,), apostrophe (’), forward slash (/), pound (#), period (.), and percent (%)
- City fields: blank, ampersand (&), and hyphen (-)
- Zip fields: blank and hyphen (-)

Problem Resolution

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the parameters are validated successfully.	N/A	N/A
Warning (4)	The job ended with a Warning because there were no records selected.	Review the Form tables for records that should be included, adjust parameter values	N/A

Possible Return Codes	Condition	Recommendation	Other Instructions
		accordingly, and schedule a new job.	
Non-Fatal Error (8)	This Return Code is not returned by this job.	N/A	N/A
Failed (12)	The job failed due to fatal conditions.	In this step, the job may fail under the following two conditions: • Encounters any runtime exceptions • Failed during restart If the job fails because of the runtime exceptions, investigate the exception reported by the process, resolve the error, and schedule a new job.	No action is necessary to schedule another job. Any existing files from a prior run will be overwritten, and no important database updates are made until all files are created.
Terminated (16)	The job is terminated manually by the user.	The reason for the termination needs to be investigated. A new job should be scheduled.	No action is necessary to schedule another job. Any existing files from a prior run will be overwritten, and no important database updates are made until all files are created.
System Failure (20)	The job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. A new job should be scheduled.	No action is necessary to schedule another job. Any existing files from a prior run will be overwritten, and no important database updates are made until all files are created.

2.1.31 Journal Posting Initiator

When to Run

This process will be run on demand. When the automatic disbursement process is creating records using the asynchronous posting setting on the Transaction Control Table, the Journal Posting Initiator process will be run after a disbursement run has been approved.

Description

This process will read transactions with a specified Journal Posting Indicator of "Not Ready to Post", and update the Journal Posting Indicator to "Ready to Post". The process runs off of the user entered parameters specified below. If no parameters are specified, the process will survey the entire Transaction Catalog table and select all transactions whose Phase is 'Final' and whose posting lines are marked "Not Ready to Post" (not recommended since in absence of parameters this will take too long to run). Each parameter that is entered will further narrow the search and selection of transactions. For example, if Bank Account Code and a Disbursement Run ID are specified, the program will only select the transactions that match all these specified criteria (that is, both Bank Account AND Disbursement Run ID).

Output

Report (Title: Journal Posting Initiated)

Sort Criteria

Report is sorted by Transaction Code, Transaction Dept, Transaction ID, Transaction Version, Vendor Line, Commodity Line, Accounting Line, Posting Line, Bank Account Code.

Parameters

Batch Parameters

Job or Job # in a Chain	Parameter	Description	Default Values
JOURNAL POSTING INITIATOR	Client Name (CLIENT_NM)	Optional field for the name of client to be appear on report	No default
	Transaction Code (PARM_DOC_CD)	Optional selection criteria for a specific transaction code.	No default
	Starting Transaction Department Code (ST_PARM_DOC_DEPT_CD)	Optional selection criteria for a transaction department code to be used in a range.	No default

Job or Job # in a Chain	Parameter	Description	Default Values
	Starting Transaction ID (ST_PARM_DOC_ID)	Optional selection criteria for a transaction ID to be used in a range.	No default
	Ending Transaction Department Code (END_PARM_DOC_DEPT_CD)	Optional selection criteria for a transaction department code to be used in a range.	No default
	Ending Transaction ID (END_PARM_DOC_ID)	Optional selection criteria for a transaction ID to be used in a range.	No default
	Transaction Date (PARM_ACCPT_DT)	Optional selection criteria for a transactions with a specific transaction (transaction) date.	No default
	Disbursement Run (DISB_RUN)	A required Yes or No that signifies whether the process is being run for a particular disbursement run or on the other selection criteria.	Y
	Disbursement Run ID (DISB_RUN_ID)	Optional field if the Disbursement Run flag is Y. Otherwise the parameter is not used.	No default
	Bank Account (BANK_ACCT_CD)	Optional selection criteria for a specific posting lines that have a certain bank code value.	No default
	Disbursement Type (DISB_TYP)	Optional selection criteria for a specific posting lines that have a certain disbursement type value.	No default
	Disbursement Category (DISB_CAT)	Optional selection criteria for a specific posting lines that have a certain disbursement category value.	No default

Job or Job # in a Chain	Parameter	Description	Default Values
	Disbursement Format (DISB_FRMT)	Optional selection criteria for a specific posting lines that have a certain disbursement format value.	No default
	Disbursement Handling Code (DISB_HAND_CD)	Optional selection criteria for a specific posting lines that have a certain disbursement handling code.	No default
	COMMIT_BLK	Added a COMMIT_BLK parameter to Journal Posting Initiator.	
	SELECT_BLK	Added a SELECT_BLK parameter to Journal Posting Initiator.	

Selection Criteria

Select all transactions whose Phase is 'Final' and whose posting lines are marked "Not Ready to Post" plus the entered criteria on parameter list.

Troubleshooting

If no parameters are entered and/or Disbursement Run ID is ‘N’ the process might take a long time to complete since it searches through Transaction Catalog for all the transactions.

2.1.32 Load EDI Invoices

Chain or Job Name	Load EDI Invoices
Recommended Frequency	The Load EDI Invoices Chain can be run daily as part of the nightly cycle or on demand.
Single Instance Required	Yes.
Can be restarted?	Yes, see the individual jobs for more details.
Reports generated	Yes, some of the jobs in the chain generate the exception report. Please refer to the individual jobs for more details.

Overview

The Load EDI Invoices chain job in CGI Advantage Financial is a group of jobs that work together to create invoice transactions from invoice (810 transaction set) data transmitted by a vendor via Electronic Data Interchange (EDI), and generates response Functional Acknowledgement (997 transaction set) and Application Advice (824 transaction sets) that are returned to the vendor. Information on the 810, 997, and 824 transaction sets can be found by purchasing the X12 EDI Transaction Sets.

The Load EDI Invoices chain job reads 810 transactions, validates the data, crosswalks the Unit of Measure and Commodity Code when indicated as a batch parameter, and creates Invoice transactions if all the validations passed. After validating parameters, the system creates a 997 transaction for all invoices in the file and subsequently creates 824 transactions for invoices that passed 997 validations but did not pass 824 validations. This process also loads EDI data into tables to allow users to view data if necessary. If an invoice is created through the EDI process, the invoice will store the EDI Header Record ID on the invoice and an Action link will take the user to the EDI Header table to view the data. The Load EDI Invoices chain has the following jobs, which are described in subsequent sections:

1. Prepare EDI Invoices
2. Load EDI Invoices
3. Submit EDI Invoices
4. EDI Invoice Exception Report

Note: Even though the above jobs in the chain can be run individually by disabling other jobs, it is recommended to always run the entire chain.

The acceptable job return code configuration depends on the business requirement. For example, if the requirement is that the subsequent jobs in the chain should continue only if the job ends with a return code of *Successful*, the acceptable job return codes for all of the jobs should be set to *Successful*. If for some jobs in the chain, a Non-Fatal error is an acceptable job return code, then that can also be configured. These configurations can be done in the Job Setup page. For baseline configuration, the Load EDI Invoices job is kicked off only when the previous job ends with a return code of Successful. The Submit EDI Invoices and Exception report jobs are kicked off when previous step end with Non-Fatal error.

If any of the jobs in the chain ends with a return code of *Failed*, *Terminated* or *System Failure*, all of the subsequent jobs will be set to Inactive.

File Selection

A file cannot be processed until the following general conditions are met:

Input File Location parameter contains a valid value.

EDI File Name to be Processed parameter contains a valid value. The file should be a text file and the .txt extension should be provided when providing the file name.

Specification File Name parameter contains a valid value.

824 Application Advice File Name Prefix parameter value is provided.

997 Functional Acknowledgement Response File Name Prefix parameter value is provided.

Advantage Interchange ID Qualifier parameter value is provided.

Advantage Interchange ID parameter value is provided.

Contact Code for Response File parameter value is provided.

One or more types of invoice is selected:

- Electronic Invoices
- Award Referencing Invoices
- Stand Alone Invoices

Additionally, the following parameters must be set, but their values will generally be defaulted and not required changes for each run:

Output File Location

Log Files

Parameter File Location

Commit Block Size

Exception Report Indicator for Submit Mode

Exception Report Transaction Error Mode

XML Export File Name

Common Chain Parameter File

Progression Counter

Maximum Number of Errors in Invoices

Finally, the chain allows several more parameters to be populated. Some of the parameters are required in that a value of true or false must be entered. The parameters are discussed in more detail within the specific jobs of the Load EDI Invoices chain job.

Minor Input

Automated Transaction Number (ADNT) table

Electronic Billing Data Source page (EBDS) table

Department page (DEPT) table

Transaction Control page (DCTRL) table

Integration Control page (INTCTRL) table

Unit page (UNIT) table

Major Input

Commodity page (COMM) table

Contact page (CNTAC) table

EDI Data file

EDI Specification file

Electronic Account profile (EAPRO) table

Electronic Data Interchange Unit of Measure Cross Walk (UOMC) table

External Commodity Cross Walk (ECOMMX) table

Transformation File (optional)

Unit of Measure (UOM) table

Vendor Customer (VCUST) table

Major Output

Electronic Data Interchange File (EDIF) table

Electronic Data Interchange Header (EDIH) table

Electronic Data Interchange Detail (EDID) table

Invoice transactions:

- Electronic
- Award Referencing
- Standalone

Chain Job Return Code

The following table shows potential return codes for the Load EDI Invoices chain job. Note that the chain job will end with the highest return code across all of the jobs.

Return Code	Condition
Successful (1)	All of the jobs end successfully.
Warning (4)	One of the jobs in the chain ends with a return code of “Warning”.
Non-Fatal Error (8)	One of the jobs in the chain ends with a return code of “Non-Fatal Error”.
Failed (12)	One of the jobs in the chain ends with a return code of “Failed”.
Terminated (16)	One of the jobs in the chain ends with a return code of “Terminated”.
System Failure (20)	One of the jobs in the chain ends with a return code of “System

Return Code	Condition
	Failure".

Problem Resolution

If any of the jobs in the chain failed due to application errors it is advisable to restart the job after correcting the errors instead of rescheduling the job. Restarting the job will reduce the processing time since the job will resume from where it has last committed and select only the unprocessed records.

Please refer to the individual jobs for details regarding the specific job processes and problem resolution.

Load EDI Invoices Chain: Prepare EDI Invoices Job

Chain or Job Name	Prepare EDI Invoices
Recommended Frequency	The Prepare EDI Invoices job can be run daily as part of the nightly cycle or on demand.
Single Instance Required	Yes.
Can be restarted?	No.
Reports generated	No. All errors will be written to the log.

Overview

The Prepare EDI Invoices job parses the EDI data file, validates, and prepares the EDI Invoice data in Advantage Financial. 824 and 997 response transactions are generated if applicable. If the file passes all file-level validations, then the system creates appropriate Advantage Invoice transaction XML files for each validated invoice.

This job step consists of the following basic steps:

1. **Parameter Validation**: In this step, the process will verify the parameters. If the parameter validation is successful, the job will go to the next step. Otherwise, the job will end with a return code of Failed.
2. **Processing the EDI data file**: In this step, the system will first evaluate each invoice for 997 errors. 997 errors are based on the specification file. They indicate that data in the file may not be accurate. For example, there may be more data segments in the file than are expected based on the specification. There may be a transaction set header that does not have a transaction set trailer. If an invoice does not pass 997 validations, the system does not continue to evaluate the invoice for 824 validations. If the invoice does pass 824 validations, the system continues evaluating and processing the invoice. An invoice may fail 824 validations for problems such as commodity code could not be found, unit of measure could not be found, the system calculated invoice total varies from the file's indicated invoice total. The list of possible 824 errors and their potential causes can be found in the *CGI Advantage – EDI Invoice Standards Guide*.

The main processing logic involved in this step includes but not limited to:

Load EDI specification file using the file name provided as a batch parameter.

Load EDI data file using the file name provided as a batch parameter.

Validate data file beginning to confirm the file correctness and identify delimiters used in the file.

Validate control segments to confirm the data structure at a high level.

Parse Transaction Sets.

Perform XLST transformation to manipulate incoming data if necessary.

Perform EDI data validations against each Transaction Set.

Generate Functional Acknowledgement (997) response for each accepted/rejected Transaction Set.

Extract invoice data as per the mapping requirements.

Perform business validations.

Generate Remittance Advice (824) response for each Transaction Set rejected.

3. **Create Invoice XML**: This last step creates the invoice xml for each invoice that passed 997 and 824 validations.

The following table shows the various steps that the Prepare EDI Invoices job goes through and the messages issued at each step.

Processing Steps	Messages
1. Parameter Validation	• Validating the batch parameters. • Batch Parameter validation completed. • Parameters validation failed. • Parameter specific messages are described in the Batch Parameter section below.
2. Process EDI data file	• EDI File Processing started. • EDI File Processing ended. • No Invoices accepted from input file.
3. Create Invoice XML	• Created XML for ### invoices.

Restartability Information

If the job fails in any of the above steps, it is recommended to always run a new instance after resolving the error.

Minor Input

Automated Transaction Number (ADNT) table

Electronic Billing Data Source page (EBDS) table

Department page (DEPT) table

Transaction Control page (DCTRL) table

Integration Control page (INTCTRL) table

Unit page (UNIT) table

Major Input

Commodity page (COMM) table

Contact page (CNTAC) table

EDI Data file

EDI Specification file

Electronic Account Profile (EAPRO) table

Electronic Data Interchange Unit of Measure Cross Walk (UOMC) table

External Commodity Cross Walk (ECOMMX) table

Transformation File (optional)

Unit of Measure (UOM) table

External Commodity Cross Walk (ECOMMX) table

Vendor Customer (VCUST) table

Batch Parameters

Parameter	Description	Default Value	Related Error Messages	Error Resolution
824 Application Advice File Name Prefix (824_FILE_NM_PFX)	Parameter to indicate the 824 file name prefix. If an invoice does not pass all 824 business rule validations, the invoice is added to the 824 file. If all invoices passed 824 validations, an 824 file will not be created.	No Default	- 824 Application Advice File Name Prefix parameter is required.	- The parameter is required, and a value was not provided. Enter a name for the 824 Application Advice File Name. An extension is not required for the parameter. Files will be saved as .txt.

Parameter	Description	Default Value	Related Error Messages	Error Resolution
997 Functional Acknowledgement Response File Name Prefix (997_FILE_NM_PFX)	Parameter to indicate the 997 file name prefix. If a file passes all interchange control and functional group edits, then a 997 transaction is created for every single invoice in the file. The transactions note if the invoice passed 997 data validations or was rejected for errors. Only one 997 file is created for all of the invoices in the 810 file.	No Default	- 997 Functional Acknowledgement File Name Prefix parameter is required.	- The parameter is required, and a value was not provided. Enter a name for the 997 Functional Acknowledgment Response File Name. An extension is not required for the parameter. Files will be saved as .txt.

Parameter	Description	Default Value	Related Error Messages	Error Resolution
(Required) Address Code required in transaction set (TRUE OR FALSE)? (AD_CD_REQ_FL)	Required Field. Indicates if an invoice will be rejected if a valid Address Code cannot be found. If this is set to True and an Address Code is not found the invoice will reject. If this is set to False and an Address Code is not found the invoice will be created, but will not submit to final unless a user determines the Address Code.	false	- (A) Address Code required parameter is required. - (B) Invalid value entered for Address Code Required in Transaction Set parameter. Value must be either true or false.	- (A) The parameter is required, and a value was not provided. Enter a value of either true or false for the parameter. - (B) A value other than true or false was entered as the parameter value. To correct, enter a value of either true or false for the parameter.
Output File Location	This is the location where 997 and 824 files will be stored after the Prepare Invoices job finishes.	$$AMSEXPORT$$	- (A) AMSEXPORT parameter is required - (B) Invalid Parameter: AMSEXPORT is not a valid directory on the server	- (A) The parameter is required, and a value was not provided. Enter a valid value. - (B) The parameter location entered does not exist as a directory on the server. Enter a valid value.
Input File Location	This is the location where the job will look for the input file.	$$AMSIMPORT$$	- Invalid Parameter: AMSIMPORT is not a valid directory on the server.	- The parameter location entered does not exist as a directory on the server. Enter a valid value.

Parameter	Description	Default Value	Related Error Messages	Error Resolution
Log files	This is the location where logs for the Load EDI Invoices chain job can be found.	$$AMS ROOT$ $/Logs		
Parameter File Location	This is the location of the parameter file for the Load EDI Invoices chain. The parameters are set based on the values entered for the parameters online prior to chain execution.	$$AMS PARM$ $	- (A) AMSPARM parameter is required. - (B) Invalid Parameter: AMSPARM is not a valid directory on the server.	- (A) The parameter is required, and a value was not provided. Enter a valid value. - (B) The parameter location entered does not exist as a directory on the server. Enter a valid value.
Client Name (CLIENT_NM)	Client Name on the Exception report.	No Default		
(Required) Contact Code for Response Files (CNTAC_CD)	This is the Address Contact Code that contains the information that should be used for the EDI response (997 and/or 824) file. This is contact information for the site. This corresponds to data found on the Contact (CNTAC) table.	No Default	- (A) Contact Code for Response Files parameter is required. - (B) Contact Code does not exist on Contact page.	- (A) The parameter is required, and a value was not provided. Enter a valid Contact Code. - (B) The Contact Code entered does not exist on the Contact page. Either add a record to the contact page that will be used for EDI Processing, or correct the Contact Code.

Parameter	Description	Default Value	Related Error Messages	Error Resolution
Enable Commodity Code Crosswalk (COMM_XW_FL)	Required. If this is set to True, it allows the site to use the External Commodity Crosswalk table to translate the vendor's Product/Service identifiers to Advantage Commodity Codes.	false	- (A) Enable Commodity Code Crosswalk parameter is required. - (B) Invalid value entered for Enable Commodity Code Crosswalk Parameter. Value must be either true or false.	- (A) The parameter is required, and a value was not provided. Enter a value of either true or false for the parameter. - (B) A value other than true or false was entered as the parameter value. To correct, enter a value of either true or false for the parameter.

Parameter	Description	Default Value	Related Error Messages	Error Resolution
Commodity Code Crosswalk Qualifier/Scheme for Miscellaneous Charges (COMM_XW_QUAL_SCH)	Indicates the Qualifier/Scheme value to be used to translate EDI miscellaneous service charge codes (e.g. SAC segment “Service, Promotion, Allowance, or Charge Code” values, spec EDI data element 1300) to Advantage Commodity Codes. If this parameter is used, the Enable Commodity Code Crosswalk parameter must also be set to true.	No Default	- (A) Commodity Code Crosswalk Qualifier is required when Enable Commodity Code Crosswalk is set to True. - (B) Commodity Code Crosswalk Qualifier does not exist on the External Commodity Code Crosswalk page.	- (A) The Enable Commodity Code Crosswalk parameter for the Load EDI Invoices chain job was set to true and a Commodity Code Crosswalk Qualifier was not provided as an additional parameter. To correct, enter a value in the Commodity Code Crosswalk Qualifier/Scheme parameter. - (B) The Enable Commodity Code Crosswalk parameter for the Load EDI Invoices chain job was set to true and the Commodity Code Crosswalk Qualifier does not exist on the External Commodity Crosswalk page. To correct change the value entered as the parameter, or add the parameter value to the External Commodity Code Crosswalk table.

Parameter	Description	Default Value	Related Error Messages	Error Resolution
Commit Block Size (MAX_BLOCK_SIZE)	Required Field. This is to limit the number of transactions per parameter file. If the number of records written per parameter file exceeds the block size then there will be a new parameter file created for the same priority.	100	- Commit block size defaulted to ###	
Delimiter Character Set (DELIM_CHAR_SET)	Required Field. Parameter to indicate the delimiters that may be used in an EDI transaction set.	*\\~/|^+':><	- Delimiter Character Set parameter is required.	- The parameter is required, and a value was not provided. Enter values that may be allowable delimiters such as *\\~/|^+':><
Default Commodity Code (DFLT_COMM_CD)	Provides a default Advantage Commodity Code to be used when the vendor does not send a valid Advantage Commodity Code and either the External Commodity Crosswalk does not result in a valid Commodity Code or the External Commodity Crosswalk is disabled.	No Default	- Commodity Code is not found on Commodity Code table.	- Either the Commodity Code does not exist on the Commodity page, or the Commodity Code is not Active on the Commodity page. To correct, either enter a different Commodity Code, add the Commodity Code to the Commodity Code page, or check the Active box on the Commodity page.

Parameter	Description	Default Value	Related Error Messages	Error Resolution
Electronic Billing Data Source code (EB_DATA_SRC_ID)	Indicates the Data Source to be used when creating Electronic Invoices, must be a valid value on Electronic Billing Data Source (EBDS). Only required if Electronic Invoices Allowed parameter is set to true.	No Default	- Electronic Billing Data Source ID is not found on Electronic Billing Data Source table.	- The Electronic Billing Data Source ID entered does not exist on the Electronic Billing Data Source page. To correct either enter a different Electronic Billing Data Source or add the Electronic Billing Data Source to the Electronic Billing Data Source page.
(Required) EDI File Name to be Processed	This is the name and location of the EDI 810 transaction file that should be processed.	$$AMS EXPORT$$/	- (A) EDI File Name to be Processed parameter is required. - (B) The EDI File could not be found.	- (A) The parameter is required, and a value was not provided. Enter a valid value. - (B) Either the EDI File Name was blank, or the EDI File could not be found. To correct enter a valid location and file name.
(Required) Specification File Name	This is the name and location of the specification file that should be used to process the 810 transaction file.	$$AMS PARM$$/	- (A) Specification File parameter is required. - (B) The Specification File could not be found.	- (A) The parameter is required, and a value was not provided. Enter a valid value. - (B) Either the Specification File Name was blank, or the Specification File could not be found. To correct enter a valid location and file name.

Parameter	Description	Default Value	Related Error Messages	Error Resolution
(Required) Enable VCUST default Vendor Address ID inference when vendor provided Address Code is either not found or invalid (true/false)?	Allows the site to infer VCUST default payment vendor address ID when ID is missing/ incorrect in the input file.	False	An invalid value was provided for Enable VCUST default Vendor Address ID inference when the vendor provided Address Code is either not found or an invalid parameter. The value must be either *true* or *false*.	- (A) The parameter is required, and a value was not provided. Enter a value of either *true* or *false* for the parameter. - (B) A value other than *true* or *false* was entered as the parameter value. To correct, enter a value of either *true* or *false* for the parameter.
Exception Report Indicator for Submit Mode (EXCEP_REP_IND)	Required Field. This is the output format for the EDI Exception Report. (1=Detailed/ 2=Failed Transactions/ 3=Processed Transactions/ 4=Failed Transaction Lines/ 5 = Transaction Status)	4	- (A) Exception Report Indicator parameter was left blank and system assumed value of 4. - (B) Exception Report Indicator for Submit Mode must be either 1, 2, 3, 4, or 5.	- (A) The Exception Report Indicator parameter was blank and as a result, the system assumed a value of 4. Nothing needs to be corrected when this message is received. - (B) The value entered for the Exception Report Indicator was not a value of 1, 2, 3, 4, 5, or blank. Enter a valid value.

Parameter	Description	Default Value	Related Error Messages	Error Resolution
Exception Report Transaction Error Mode (EXP_RPT_TYP)	Required Exception Report Type (1=Detailed, 2=Summarized, if not entered then defaulted to Detailed). If Summarized is selected, only summary information is provided. If Detailed is selected, the system lists each individual invoice and the status of each invoice.	1	- (A) Exception Report Transaction Error Mode parameter was left blank and system assumed value of 1. - (B) Exception Report Transaction Error Mode must be either 1 (Detailed) or 2 (Summary).	- (A) The Exception Report Transaction Error Mode parameter was blank and as a result, the system assumed a value of 1. Nothing needs to be corrected when this message is received. - (B) The value entered for the Exception Report Transaction Error Mode was not a value of 1, 2, or blank. Enter a valid value.
XML Export File Name (FILE_NM)	Required Field. The name of the EDI 810 xml file that is created by the Prepare Invoices job and will be processed by the Load EDI invoice job of the chain.	$$AMSEXPORT$$/EDIInvoices.xml	- (A) XML Export File Name is not specified. - (B) XML Export File Name should have a file extension of .xml.	- (A) The parameter is required, and a value was not provided. Enter a valid value. - (B) An export file name was provided, but it did not contain an extension of .xml. Provide a file name with a .xml extension.

Parameter	Description	Default Value	Related Error Messages	Error Resolution
Advantage Interchange ID (IC_ID)	Required Field. Interchange ID indicating the identifier the Advantage entity intends to identify itself with. This is expected to be a value consistent with the specification for the 810 ISA08 (Interchange Receiver ID) and 997/824 ISA06 (Interchange Sender ID) data elements. If the number of characters is less than 15, the system pads the parameter with blanks so that the field size is 15 characters.	No Default	- (A) Advantage Interchange ID is required. - (B) Advantage Interchange ID should be 15 characters or less.	- (A) The parameter is required, and a value was not provided. Enter a valid Advantage Interchange ID. - (B) An Advantage Interchange ID was provided but it exceeded 15 characters long. Enter an Advantage Interchange ID that is 15 characters or less.

Parameter	Description	Default Value	Related Error Messages	Error Resolution
Advantage Interchange ID Qualifier (IC_ID_QUAL)	Required Field. Interchange ID Qualifier indicating the type of identifier the Advantage entity intends to identify itself with, provided for validation purposes (against the Receiver ID of incoming files) and for providing Sender identification (on outgoing response files). This is expected to be one of the values allowed for the ISA07 (Interchange ID Qualifier) data element	No Default	- (A) Advantage Interchange ID Qualifier is required. - (B) Advantage Interchange ID Qualifier should be 2 Characters long.	- (A) The parameter is required, and a value was not provided. Enter a valid Advantage Interchange ID Qualifier. - (B) An Advantage Interchange ID Qualifier was entered, but the Interchange ID Qualifier was not two characters long. Enter a valid value.

Parameter	Description	Default Value	Related Error Messages	Error Resolution
(Required) Electronic Invoices Allowed (true or false)? (IE_ALW_FL)	Required Field. If this is set to True the system will attempt to first create Electronic Invoices before creating Award Referencing and/or Stand Alone invoices. The system requires that one or more of either the Electronic Invoices Allowed, Award Referencing Invoices Allowed, or Stand Alone Invoices Allowed parameter is set to true. If an invoice contains an Electronic Account number and the Electronic Invoices Allowed parameter is set to false, the system will reject that invoice as the assumption is an electronic invoice should have been created for that invoice. If it is known that a file contains Electronic Account numbers, and Electronic Invoices are	false	- (A) At least one of Electronic Invoices Allowed, Award Referencing Invoices Allowed, or Stand Alone Invoices Allowed must be set to true. Value must be true or false. - (B) Electronic Invoices Allowed is required. - (C) Invalid value entered for Electronic Invoices Allowed Parameter. Value must be either true or false. - (D) Electronic Billing Data Source ID parameter is required.	- (A) The Load EDI Invoices chain job requires that at least one type of invoice be created. To correct, a value of true must be entered for at least one of Electronic, Award Referencing, or Stand Alone Invoices. - (B) The parameter is required, and a value was not provided. To correct, enter a value of either true or false for the parameter. - (C) A value other than true or false was entered as the parameter value. To correct, enter a value of either true or false for the parameter. - (D) The Electronic Invoices Allowed parameter was set to true, but an Electronic Billing Data Source ID parameter was not entered. To correct, provide a value for the Electronic Billing Data Source.

Parameter	Description	Default Value	Related Error Messages	Error Resolution
Electronic Invoice Transaction Code (IE_DOC_CD)	Transaction Code used for the creation of Electronic Invoices. This is only required if the Electronic Invoices Allowed parameter is set to true. If it is entered and the Electronic Invoices Allowed parameter is set to false the system will ignore the Electronic Invoice Transaction Code parameter and will not do any validations for this parameter.	No Default	- (A) Electronic Invoice Transaction Code parameter is required. - (B) Electronic Invoice Transaction Code is not found on Transaction code table.	- (A) The Electronic Invoices Allowed parameter was set to true, but an Electronic Invoice Transaction Code parameter value was not provided. To correct, enter a valid value for the parameter. - (B) The Electronic Invoices Allowed parameter was set to true, and either the Electronic Invoice Transaction Code does not exist on the Transaction Control page, or the Electronic Invoice Transaction Code is not Active on the Transaction Control page. To correct, enter a valid value.

Parameter	Description	Default Value	Related Error Messages	Error Resolution
Electronic Invoice Transaction Department (IE_DOC_DEPT_CD)	Department code used for the creation of Electronic Invoice transactions. This is only required if the Electronic Invoices Allowed parameter is set to true. If it is entered and the Electronic Invoices Allowed parameter is set to false the system will ignore the Electronic Invoice Transaction Department parameter and will not do any validations for this parameter.	No Default	- (A) Electronic Invoice Transaction Dept Code parameter is required. - (B) Electronic Invoice Transaction Dept Code is not found on Dept table. - (C) UTILITY_ENABLED parameter is not set to true on INTCTRL page for Electronic Invoice Department and Electronic Invoice Unit specified.	- (A) The Electronic Invoices Allowed parameter was set to true, but an Electronic Invoice Transaction Department parameter value was not provided. To correct, enter a valid value for the parameter. - (B) The Electronic Invoices Allowed parameter was set to true, and either the Electronic Invoice Transaction Dept Code does not exist on the Department page, or the Electronic Invoice Transaction Dept Code is not Active on the Department page. To correct, enter a valid value. - (C) The Electronic Invoices Allowed parameter was set to true, but a record does not exist or is not set to true on the Integration Control page for the parameter UTILITY_ENABLED in regards to the Electronic Invoice Department and Electronic Unit specified. Note that “ALL” is considered a valid Unit and/or Department.

Parameter	Description	Default Value	Related Error Messages	Error Resolution
Electronic Invoice Transaction Prefix (IE_DOC_PFX)	Optional automatic transaction numbering prefix used for the generation of Transaction IDs during the creation of Invoice transactions. This is only required if the Electronic Invoices Allowed parameter is set to true. If it is entered and the Electronic Invoices Allowed parameter is set to false the system will ignore the Electronic Invoice Transaction Prefix parameter and will not do any validations for this parameter.	No Default	- (A) Electronic Invoice Transaction Prefix parameter is required. - (B) Entry does not exist in the Automatic Transaction Numbering page for Fiscal Year, Electronic Invoice Transaction Code, Electronic Invoice Department Code and Electronic Invoice Prefix.	- (A) The Electronic Invoices Allowed parameter was set to true, but an Electronic Invoice Transaction Prefix parameter value was not provided. To correct, enter a valid value for the parameter. - (B) The Electronic Invoices Allowed parameter was set to true, but an entry does not exist in the Automatic Transaction Numbering (ADNT) page for Fiscal Year, Transaction Code, Department Code and Prefix. To correct, enter a valid value.

Parameter	Description	Default Value	Related Error Messages	Error Resolution
Electronic Invoice Transaction Unit (IE_DOC_UNIT_CD)	Optional. Unit Code used for the creation of Electronic Invoice transactions. This is only evaluated if the Electronic Invoices Allowed parameter is set to true. If it is entered and Electronic Invoices Allowed parameter is set to false the system will ignore the Electronic Invoice Transaction Unit parameter and will not do any validations for this parameter.	No Default	- (A) Electronic Invoice Transaction Unit is not found on Unit table. - (B) UTILITY_ENABLED parameter is not set to true on the Integration Parameters page for Electronic Invoice Department and Electronic Invoice Unit specified.	- (A) The Electronic Invoices Allowed parameter was set to true, but either the Electronic Invoice Transaction Unit Code does not exist on the Unit page, or the Electronic Invoice Transaction Unit Code is not Active on the Unit page. To correct, enter a valid value for the parameter. - (B) The Electronic Invoices Allowed parameter was set to true, but a record does not exist or is not set to true on the Integration Control page for the parameter UTILITY_ENABLED in regards to the Electronic Invoice Department and Electronic Unit specified. Note that “ALL” is considered a valid Unit and/or Department.

Parameter	Description	Default Value	Related Error Messages	Error Resolution
Electronic Invoices - Infer Dept and Unit from EAPRO (true or false)? IE_INFER_EAPRO	Optional. If this value is set to false, the system sets the Electronic Invoices Transaction Dept and Transaction Unit equal to the batch parameter values for Electronic Invoices. If this value is set to true, the system infers the Department and Unit from the EAPRO record, with the following exceptions: 1. If Department is set to ALL on EAPRO, then Department will be set to the Electronic Invoice Department defined in the batch parameters. 2. If Unit is set to ALL on EAPRO, then Unit will be set to the Electronic Invoice Unit defined in the batch parameters.	True	Invalid value provided for Electronic Invoices - Infer Dept and Unit from EAPRO parameter. When Electronic Invoices are desired, value for this parameter must be either true or false.	If the (Required) Electronic Invoices Allowed (true or false)? parameter is set to True, and the Electronic Invoices – Infer Dept and Unit from EAPRO is set to a value other than true or false, this error will be issued. To correct, enter a valid value for the parameter.

Parameter	Description	Default Value	Related Error Messages	Error Resolution
(Required) Award Referencing Invoices Allowed (true or false)? (IN_ALW_FL)	Required Field. If this is set to True the system will attempt to create Award Referencing Invoices. If the Electronic Invoices Allowed parameter is also set to true, the system first tries to create Electronic Invoices before attempting to create Award Referencing invoices. The system requires that one or more of either the Electronic Invoices Allowed, Award Referencing Invoices Allowed, or Stand Alone Invoices Allowed parameter is set to true.	false	- (A) At least one of Electronic Invoices Allowed, Award Referencing Invoices Allowed, or Stand Alone Invoices Allowed must be set to true. Value must be true or false. - (B) Award Referencing Invoices Allowed parameter is required. - (C) Invalid value entered for Award Referencing Invoices Allowed Parameter. Value must be either true or false.	- (A) The Load EDI Invoices chain job requires that at least one type of invoice be created. To correct, a value of true must be entered for at least one of Electronic, Award Referencing, or Stand Alone Invoices. - (B) The parameter is required, and a value was not provided. To correct, enter a value of either true or false for the parameter. - (C) A value other than true or false was entered as the parameter value. To correct, enter a value of either true or false for the parameter.

Parameter	Description	Default Value	Related Error Messages	Error Resolution
Award Referencing Invoice Transaction Code (IN_DOC_CD)	Transaction Code used for the creation of an award-referencing Invoice. This is only required if the Award Referencing Invoices Allowed parameter is set to true. If it is entered and the Award Referencing Invoices Allowed parameter is set to false the system will ignore the Award Referencing Invoice Transaction Code parameter and will not do any validations for this parameter.	No Default	- (A) Award Referencing Invoice Transaction Code parameter is required. - (B) Award Referencing Invoice Transaction Code is not found on Transaction code table.	- (A) The Award Referencing Invoices Allowed parameter was set to true, but an Award Referencing Invoice Transaction Code parameter value was not provided. To correct, enter a valid value for the parameter. - (B) The Award Referencing Invoices Allowed parameter was set to true, but either the Award Referencing Invoice Transaction Code does not exist on the Transaction Control page, or the Electronic Invoice Transaction Code is not Active on the Transaction Control page. To correct, enter a valid value.

Parameter	Description	Default Value	Related Error Messages	Error Resolution
Award Referencing Invoice Transaction Department (IN_DOC_DEPT_CD)	Department code used for the creation of award-referencing Invoice transactions. This is only required if the Award Referencing Invoices Allowed parameter is set to true. If it is entered and the Award Referencing Invoices Allowed parameter is set to false the system will ignore the Award Referencing Invoice Transaction Department parameter and will not do any validations for this parameter.	No Default	- (A) Award Referencing Invoice Transaction Dept Code parameter is required. - (B) Award Referencing Invoice Transaction Dept Code is not found on Dept table.	- (A) The Award Referencing Invoices Allowed parameter was set to true, but an Award Referencing Invoice Transaction Department parameter value was not provided. To correct, enter a valid value for the parameter. - (B) The Award Referencing Invoices Allowed parameter was set to true, but either the Award Referencing Invoice Transaction Dept Code does not exist on the Department page, or the Award Referencing Invoice Transaction Dept Code is not Active on the Department page. To correct, enter a valid value.

Parameter	Description	Default Value	Related Error Messages	Error Resolution
Award Referencing Invoice Transaction Prefix (IN_DOC_PFX)	Automatic transaction numbering prefix used for the generation of Transaction IDs during the creation of Award Referencing Invoice transactions. This is only required if the Award Referencing Invoices Allowed parameter is set to true. If it is entered and the Award Referencing Invoices Allowed parameter is set to false the system will ignore the Award Referencing Invoice Transaction Prefix parameter and will not do any validations for this parameter.	No Default	- (A) Award Referencing Invoice Transaction Prefix parameter is required. - (B) Entry does not exist in the Automatic Transaction Numbering page for Fiscal Year, Award Referencing Invoice Transaction Code, Award Referencing Invoice Department Code, and Award Referencing Invoice Prefix.	- (A) The Award Referencing Invoices Allowed parameter was set to true, but an Award Referencing Invoice Transaction Prefix parameter value was not provided. To correct, enter a valid value for the parameter. - (B) The Award Referencing Invoices Allowed parameter was set to true, but an entry does not exist in the Automatic Transaction Numbering (ADNT) page for Fiscal Year ,Transaction Code, Department Code and Prefix. To correct, enter a valid value.

Parameter	Description	Default Value	Related Error Messages	Error Resolution
Award Referencing Invoice Transaction Unit (IN_DOC_UNIT_CD)	Unit code used for the creation of award-referencing Standalone Invoice transactions. This is only evaluated if the Award Referencing Invoices Allowed parameter is set to true. If it is entered and the Award Referencing Invoices Allowed parameter is set to false the system will ignore the Award Referencing Invoice Transaction Unit parameter and will not do any validations for this parameter.	No Default	- Award Referencing Invoice Transaction Unit is not found on Unit table.	- The Award Referencing Invoices Allowed parameter was set to true, but either the Award Referencing Invoice Transaction Unit Code does not exist on the Unit page, or the Award Referencing Invoice Transaction Unit Code is not Active on the Unit page. To correct, enter a valid value.

Parameter	Description	Default Value	Related Error Messages	Error Resolution
(Required) Stand Alone Invoices Allowed (true or false)? (IS_ALW_FL)	Required Field. If this is set to True the system will attempt to create Stand Alone Invoices. If the Electronic Invoice Allowed parameter and the Award Referencing Invoices Allowed parameters are also set to true, the system will only attempt to create Stand Alone invoices if the other two invoices could not be created. The system requires that one or more of either the Electronic Invoices Allowed, Award Referencing Invoices Allowed, or Stand Alone Invoices Allowed parameter is set to true.	false	- (A) At least one of Electronic Invoices Allowed, Award Referencing Invoices Allowed, or Stand Alone Invoices Allowed must be set to true. Value must be true or false. - (B) Stand Alone Invoices Allowed parameter is required. - (C) Invalid value entered for Stand Alone Invoices Desired Parameter. Value must be either true or false.	- (A) The Load EDI Invoices chain job requires that at least one type of invoice be created. To correct, a value of true must be entered for at least one of Electronic, Award Referencing, or Stand Alone Invoices. - (B) The parameter is required, and a value was not provided. To correct, enter a value of either true or false for the parameter. - (C) A value other than true or false was entered as the parameter value. To correct, enter a value of either true or false for the parameter.

Parameter	Description	Default Value	Related Error Messages	Error Resolution
Stand Alone Invoice Transaction Code (IS_DOC_CD)	Transaction Code used for the creation of non-Electronic Standalone Invoices. This is only required if the Stand Alone Invoices Allowed parameter is set to true. If it is entered and the Stand Alone Invoices Allowed parameter is set to false the system will ignore the Stand Alone Invoice Transaction Code parameter and will not do any validations for this parameter.	No Default	- (A) Stand Alone Invoice Transaction Code parameter is required. - (B) Stand Alone Invoice Transaction Code is not found on Transaction code table.	- (A) The Stand Alone Invoices Allowed parameter was set to true, but a Stand Alone Invoice Transaction Code parameter value was not provided. To correct, enter a valid value for the parameter. - (B) The Stand Alone Invoices Allowed parameter was set to true, but either the Stand Alone Invoice Transaction Code does not exist on the Transaction Control page, or the Electronic Invoice Transaction Code is not Active on the Transaction Control page. To correct, enter a valid value.

Parameter	Description	Default Value	Related Error Messages	Error Resolution
Stand Alone Invoice Transaction Department (IS_DOC_DEPT_CD)	Department code used for the creation of non-Electronic Standalone Invoice transactions. This is only required if the Stand Alone Invoices Allowed parameter is set to true. If it is entered and the Stand Alone Invoices Allowed parameter is set to false the system will ignore the Stand Alone Invoice Transaction Department parameter and will not do any validations for this parameter.	No Default	- (A) Stand Alone Invoice Transaction Dept Code parameter is required. - (B) Stand Alone Invoice Transaction Dept Code is not found on Dept table.	- (A) The Stand Alone Invoices Allowed parameter was set to true, but a Stand Alone Invoice Transaction Department parameter value was not provided. To correct, enter a valid value for the parameter. - (B) The Stand Alone Invoices Allowed parameter was set to true, but either the Stand Alone Invoice Transaction Dept Code does not exist on the Department page, or the Stand Alone Invoice Transaction Dept Code is not Active on the Department page. To correct, enter a valid value.

Parameter	Description	Default Value	Related Error Messages	Error Resolution
Stand Alone Invoice Transaction Prefix (IS_DOC_PFX)	Automatic transaction numbering prefix used for the generation of Transaction IDs during the creation of non-Electronic Standalone Invoice transactions. This is only required if the Stand Alone Invoices Allowed parameter is set to true. If it is entered and the Stand Alone Invoices Allowed parameter is set to false the system will ignore the Stand Alone Invoice Transaction Prefix parameter and will not do any validations for this parameter.	No Default	- (A) Stand Alone Invoice Transaction Prefix parameter is required. - (B) Entry does not exist in the Automatic Transaction Numbering page for Fiscal Year, Stand Alone Invoice Transaction Code, Stand Alone Invoice Department Code, and Stand Alone Invoice Prefix.	- (A) The Stand Alone Invoices Allowed parameter was set to true, but a Stand Alone Invoice Transaction Prefix parameter value was not provided. To correct, enter a valid value for the parameter. - (B) The Stand Alone Invoices Allowed parameter was set to true, but an entry does not exist in the Automatic Transaction Numbering (ADNT) page for Fiscal Year ,Transaction Code, Department Code and Prefix. To correct, enter a valid value.

Parameter	Description	Default Value	Related Error Messages	Error Resolution
Stand Alone Invoice Transaction Unit (IS_DOC_UNIT_CD)	Unit code used for the creation of non-Electronic Standalone Invoice transactions. This is only evaluated if the Stand Alone Invoices Allowed parameter is set to true. If it is entered and the Stand Alone Invoices Allowed parameter is set to false the system will ignore the Stand Alone Invoice Transaction Unit parameter and will not do any validations for this parameter.	No Default	- Stand Alone Invoice Transaction Unit is not found on Unit table.	- The Stand Alone Invoices Allowed parameter was set to true, but either the Stand Alone Invoice Transaction Unit Code does not exist on the Unit page, or the Stand Alone Invoice Transaction Unit Code is not Active on the Unit page. To correct, enter a valid value.

Parameter	Description	Default Value	Related Error Messages	Error Resolution
(Required) Maximum Number of Errors in Invoices	This is the maximum number of errors that can be found within invoices before the job will fail. Errors are counted from both 997 validations and 824 validations. Once the maximum number of errors is reached, the system stops processing and stops creating the 997and 824 files. These files may not reference all of the invoices in the 810 file as processing stopped before all invoices could be evaluated and processed.	100	- (A) Maximum Number of Errors in Invoices parameter is required. - (B) Maximum Number of Errors in Invoices must be an Integer and greater than or equal to zero.	- (A) The parameter is required, and a value was not provided. To correct, enter a valid value. - (B) A value was provided for the parameter, but the value was either not an integer or was less than zero. To correct, enter a value equal to or greater than 0.

Parameter	Description	Default Value	Related Error Messages	Error Resolution
Common Chain Parameter File (PARM_FILE)	Required Field. Required name identifying the common chain parameter file (.txt).	$$AMSPARM$$/EDIParm.txt	- (A) AMSPARM parameter is required. - (B) Invalid Parameter: AMSPARM is not a valid directory on the server. - (C) Common Chain Parameter file for Load file should have file extension of .txt.	- (A) The parameter is required, and a value was not provided. To correct, provide a valid value. - (B) The value entered could not be found on the server. To correct, enter a valid value. - (C) The value entered does not have an extension of .txt. To correct, add a file extension of .txt.
Progression Counter (PROG_CTR_SZ)	Required Field. Parameter to list at what point the system should create a batch log message to indicate the number of invoices processed.	1000	- Progression counter size defaulted to ###.	- A value was not entered for the progression counter and as a result the system defaulted the progression counter size. There is no need to correct anything.

Parameter	Description	Default Value	Related Error Messages	Error Resolution
Award Parsing Information (REFD_DOC_FORM)	Must be a valid Java regular expression (RegEx) using named capture groups to identify the Department, Transaction Code, and Transaction ID respectively. The regular expression will inform the system of the acceptable format for the Award transaction reference. The name of the named capture groups must be DocDept, DocCode, DocID. This parameter is only used when the Award Referencing Invoices Allowed parameter is set to true and the system is creating an Award Referencing invoice. This parameter is discussed in more detail in the XYZ section.	No Default	- Award Reference Parsing Information parameter should be a valid Java regular expression and must contain DocCode, DocDept, and DocID in the expression.	- The Award Referencing Invoices Allowed parameter was set to true, and either a valid Java regular expression was not provided to indicate Award Parsing Information, or a valid Java regular expression was provided but the expression did not contain DocCode, DocDept, and DocID. The system uses those to determine how to parse award reference information. To correct ensure your valid java expression contains a DocCode, DocDept, and DocID field.

Parameter	Description	Default Value	Related Error Messages	Error Resolution
Reject All on Error (TRUE or FALSE)? (RJCT_ALL_ON_ERR_FL)	Required Field. True/False parameter indicating whether all invoices in the file should be considered rejected if there is at least one invoice that is rejected. "True" will require a file to pass all validations, both 997 and 824, before a single invoice will be processed. "False" will allow invoices to be individually processed if there are no invoice-level errors found.	false	- (A) Reject All on Error parameter is required. - (B) Invalid value entered for Reject All on Error Parameter. Value must be either true or false.	- (A) The parameter is required, and a value was not provided. Enter a value of either true or false for the parameter. - (B) A value other than true or false was entered as the parameter value. To correct, enter a value of either true or false for the parameter.
Transformation Specification File Name (TRANS_SPECS_FILE_NM)	Location of transformation file to change values in the EDI file. This parameter is discussed in more detail in the XYZ section.	No Default	- Transformation Specification file could not be found based on the parameter provided.	- The file specified could not be found. Either the folder location is wrong for the file, or the file does not exist. To correct, enter a valid location and file name.

Parameter	Description	Default Value	Related Error Messages	Error Resolution
(Required) Enable Unit of Measure Code Crosswalk (true or false)? (UOM_XW_FL)	Required Field. Enables use of the Unit of Measure Crosswalk table to translate vendor's Unit of Measure identifiers to Advantage Unit of Measure values.	false	- (A) Enable Unit of Measure Code Crosswalk parameter is required. - (B) Invalid value entered for Enable Unit of Measure Crosswalk Parameter. Value must be either true or false.	- (A) The parameter is required, and a value was not provided. Enter a value of either true or false for the parameter. - (B) A value other than true or false was entered as the parameter value. To correct, enter a value of either true or false for the parameter.
(Required) Create Invoice with invalid address (true or false)? (USE_INVALID_ADD_FL)	Required Field. This determines if the system creates an invoice if an Address Code cannot be found. If this is set to true then an invoice will be created but a user will need to determine the correct address for the invoice once the invoice is created in Advantage in Draft phase.	false	- (A) Create Invoice with invalid address parameter is required. - (B) Invalid value entered for Create Invoices with invalid Address Code Parameter. Value must be either true or false.	- (A) The parameter is required, and a value was not provided. Enter a value of either true or false for the parameter. - (B) A value other than true or false was entered as the parameter value. To correct, enter a value of either true or false for the parameter.

Parameter	Description	Default Value	Related Error Messages	Error Resolution
(Required) Vendor Code required in transaction set (true or false)? (VEND_CD_REQ_FL)	Required Field. Requires the vendor to send their vendor code in the 810 transaction set. If this is not sent or is not a valid value and the parameter is set to True the invoice will be rejected.	false	- (A) Vendor Code required parameter is required. - (B) Invalid value entered for Vendor Code Required in Transaction Set parameter. Value must be either true or false.	- (A) The parameter is required, and a value was not provided. Enter a value of either true or false for the parameter. - (B) A value other than true or false was entered as the parameter value. To correct, enter a value of either true or false for the parameter.
(Required) Enable Vendor Invoice Number Modification (TRUE OR FALSE)? (VEND_INV_NO_MOD_FL)	Required Field. Allows Advantage to modify a vendor's invoice number if multiple invoice numbers are used for different dates. If this option is invoked the system will append the invoice date to the end of the invoice number.	false	- (A) Enable Vendor Invoice Number Modification parameter is required. - (B) Invalid value entered for Enable Vendor Invoice Modification Parameter. Value must be either true or false.	- (A) The parameter is required, and a value was not provided. Enter a value of either true or false for the parameter. - (B) A value other than true or false was entered as the parameter value. To correct, enter a value of either true or false for the parameter.

Additional Information Regarding Certain Parameters

Prior to beginning execution of the job, there are a few parameters that require a detailed explanation.

Specification File Name to be processed parameter:

Before site exchanging files with trading partners, a set of rules defining their EDI files and how they should be processed have to be established and documented. This document is known as the *CGI Advantage – EDI Invoice Standards Guide*. Examples of rules in this document can include the requirement of a data segment/element and whether it is optional or mandatory or the

validity of a value in a data element or the number of times a loop or group of segments can be used.

The *CGI Advantage – EDI Invoice Standards Guide* is the most important document for a user creating a mapping of the corresponding EDI file received into the system.

The Advantage application uses an XML based mapping (also called EDI specifications document) to define the rules and requirements that were documented in the *CGI Advantage – EDI Invoice Standards Guide* to process the incoming files. The XML element naming mimics the EDI generic terms for easy understanding but also includes additional attributes that help the processing.

The system supports multiple forms of implementation guides to facilitate different kinds of invoice processing within the system. The types of invoices Advantage supported are Electronic Invoices, Award Referencing Invoices and Standalone Invoices. If a site is using multiple implementation guides to meet their needs, each type of implementation guide should have a corresponding EDI specification file defined. The Prepare EDI invoices step in the chain job provides several batch parameters that enables site to use the appropriate EDI specifications file based on the EDI data file being processed.

The following diagram depicts the structure of an EDI specification XML file used by the Advantage application.

The root element called specifications and contains child element segmentSpecification which will be repeated for each segment as defined in the CGI Advantage – EDI Invoice Standards Guide.

Each segmentSpecification XML element further has child elements to define EDI segment attributes, rules and collection of EDI elements that belong to the segment. These are explained in detail in later sections.

Each elementSpecification XML further has child elements to define EDI element attributes, rules for that element. These are explained in detail in later sections.

```
<specifications>
    <segmentSpecification>
        <id>ISA</id>
        <description>Interchange Control Header</descript
        <usage>M</usage>
        <maximumUse>1</maximumUse>
        <maximumElements>16</maximumElements>
        <loopSegment>false</loopSegment>
        <loopRepeat>0</loopRepeat>
        <elementSpecification>
        <elementSpecification>
        <elementSpecification>
        <elementSpecification>
        <elementSpecification>
        <elementSpecification>
        <elementSpecification>
        <elementSpecification>
        <elementSpecification>
        <elementSpecification>
        <elementSpecification>
        <elementSpecification>
        <elementSpecification>
        <elementSpecification>
        <elementSpecification>
        <elementSpecification>
        <elementSpecification>
    </segmentSpecification>
    <segmentSpecification>
    <segmentSpecification>
    <segmentSpecification>
```

Segment Definition parameter**:**

The diagram and table below explains the attributes of a segment specification.

```
<specifications>
    <segmentSpecification>
        <id>ISA</id>
        <description>Interchange Control Header</description>
        <usage>M</usage>
        <maximumUse>1</maximumUse>
        <maximumElements>16</maximumElements>
        <loopSegment>false</loopSegment>
        <loopRepeat>0</loopRepeat>
```

Attribute	Description	Comments
ID	This is the ID of the segment.	Case sensitive for system processing and should match exactly to the value defined in the *CGI Advantage – EDI Invoice Standards Guide*.

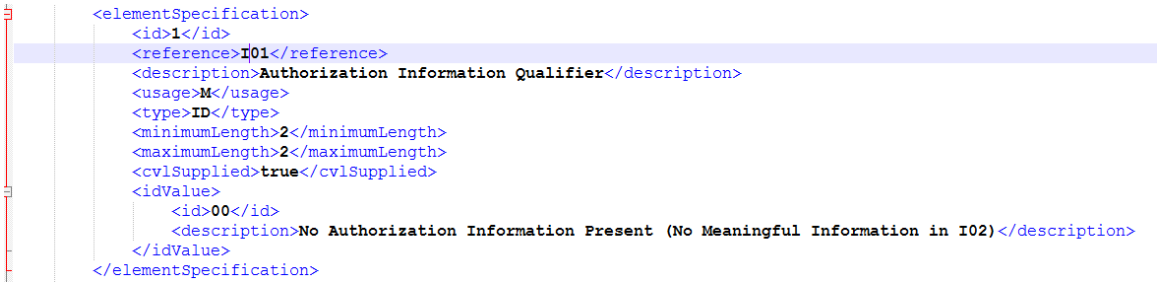
Attribute	Description	Comments
description	This is the text description of the data segment.	Informational. Not intended for processing.
usage	This contains the standard requirement (for example, **M** - Mandatory, **O** - Optional etc…) setting of the data segment.	Case sensitive for system processing and should match exactly to the value defined in the *CGI Advantage – EDI Invoice Standards Guide*.
maximumUse	This is the number of times segments can be used.	Must be a positive numeric value.
loopSegment	This indicates whether the segment is a beginning of a loop.	Must be a true or false value.
loopRepeat	Applies only when loopSegment attribute equals to *true*. For other cases, value must be 0.	Must be a positive numeric value.

Additionally, a loop segment definition includes list of child segments included within each loop group as shown below.

```
<segmentSpecification>
    <id>N1</id>
    <description>Name</description>
    <usage>M</usage>
    <maximumUse>2</maximumUse>
    <maximumElements>4</maximumElements>
    <loopSegment>true</loopSegment>
    <childSegment>
        <id>N1</id>
    </childSegment>
    <childSegment>
        <id>N2</id>
    </childSegment>
    <childSegment>
        <id>N3</id>
    </childSegment>
    <childSegment>
        <id>N4</id>
    </childSegment>
    <childSegment>
    <loopRepeat>3</loopRepeat>
```

Element Definition parameter**:**

The diagram and table below explains the attributes of a data element specification.



```
<elementSpecification>
    <id>1</id>
    <reference>I01</reference>
    <description>Authorization Information Qualifier</description>
    <usage>M</usage>
    <type>ID</type>
    <minimumLength>2</minimumLength>
    <maximumLength>2</maximumLength>
    <cvlSupplied>true</cvlSupplied>
    <idValue>
        <id>00</id>
        <description>No Authorization Information Present (No Meaningful Information in I02)</description>
    </idValue>
</elementSpecification>
```

Attribute	Description	Comments
ID	This is represents element position in within the segment.	Must be a numeric value.
reference	This is the element reference as defined in the *CGI Advantage – EDI Invoice Standards Guide*.	Should exactly match the *CGI Advantage – EDI Invoice Standards Guide* as this value is transmitted back in functional acknowledgement (997) response for element data validation failures.
description	This is the text description of the data segment.	Informational. Not intended for processing.
usage	This contains the standard requirement (for example, **M** - Mandatory, **O** - Optional etc…) setting of the data element as defined in the *CGI Advantage – EDI Invoice Standards Guide*.	Case sensitive for system processing and should match exactly to the value defined in the *CGI Advantage – EDI Invoice Standards Guide*.
type	This represents type (for example, ID, DT, AN etc…) of the data coming in the element.	Case sensitive for system processing and should match exactly to the value defined in the *CGI Advantage – EDI Invoice Standards Guide*.
minimumLength	This represents minimum length of data allowed for element.	Must be a positive numeric value.
maximumLength	This represents maximum length of data allowed for element.	Must be a positive numeric value.
cvlSupplied	This is applicable for ID type elements and represents whether system should validate the data value against a valid list of codes.	Must be a *true* or *false*. If set to true, set of valid values must be listed in the specification XML.

Attribute	Description	Comments
idValue	If a data element can only be a specific ID, this is where the list of possible ID's are defined. If the data should not be validated against a list of IDs, the cvlSupplied value should be false and no codes provided for the idValue. If the data should be validated against a list of IDs, the cvlSupplied value should be true and codes should be provided for the idValue.	idValues should in most situations correspond to EDI values.

Transformation Specification File Name parameter**:**

While EDI specifications dictate what is acceptable for EDI, sites may need to manipulate incoming data to update segments and data elements so that data passes EDI and business validations. For example, if a vendor sends in monthly service information, but does not use the EDI code of SV for service, then a transformation process will be needed for the system to automatically change that code to SV so that the system will process that data and map the data correctly to Advantage Invoices.

The Advantage process makes use of XSLT (Extensible Stylesheet Language Transformations) as the format for specifying transformation rules because it is a well-established industry-standard specification for XML-to-XML transformations. XSLT is already used in several areas of Advantage such as ABI (Advantage Business Integration) and for site-specific transformations of form printing data before transmission to the Advantage forms printing server (for example, Adobe, BIRT).

The transformation process provided by this batch process is simple in nature, capable of analyzing data in the incoming EDI 810 file, rearranging data, and introducing new data based on hardcoded values in the transformation. The transformation process is not capable of performing business logic such as inferring values from Advantage tables into the EDI data or performing table lookups to determine what logic to perform. The transformation process is also not capable of injecting whole new data segments into the transaction set if such a need arises for a site. Most importantly, it is expected that the data element values modified using transformation process comply with EDI standards and Advantage requirements. Any violations will result rejection of the invoice data.

Examples of such transformations collected from business cases include:

- If SAC04 (Data element 04 in the SAC segment) = 'PRB002' (previous account balance) and SAC01 = A (allowance), change SAC01 to 'C' (Charge).
- If SAC04 = 'LPC001' (delinquent payment change) and SAC01 = A (allowance) and SAC05 > 0 (Amount greater than zero), change SAC01 to 'C' (Charge).
- If BIG04 populated with 'UTILITY', blank the value.
- A '2' is added to the FEIN number sent in the REF segment (qualified with TJ) .

As suggested above, none of these examples rely on business logic, rather just basic analysis of the incoming EDI data and substitution with hardcoded data. Some of these transformation rules may be specific to a single vendor. Therefore, the batch process supports a "transformation specification file" to be provided for each execution of the batch process as parameter. Thus when the process is run for a file from Vendor A, transformation rules specific to Vendor A – possibly containing hardcoded values specific to Vendor A – can be used. When the process is run for a file from another vendor, a transformation rules specific to that vendor can be supplied,

and so on. Transformation rules are optional and the system will not invoke the process if transformation specification file is not provided as a parameter for the batch execution.

Leveraging XLST for data manipulation requires a temporary transformation of the text files to/from XML. To accommodate that, the batch process uses the following temporary transformations during the processing which is transparent to the end user invoking the job.

Most importantly, defining transformation rules in XSLT format requires an understanding of the XML schema on which the rules are executed.

The following picture shows the XML structure that is created and used in the transformation process.

- The root element called EdiTransaction and contains child element Segment repeated as appear in the EDI data file.
- Each Segment XML element contains attributes “id”, “Order” and child XML element called Element repeated as data values appear within the segment. Attribute “id” indicates the segment ID and “Order” specifies the position of the segment within the EDI data file transaction set.
- Each Element XML element contains attribute “Order” and text only value. Attribute “Order” specify the order in which element value appear in the segment and text only value represents the element data value.

```
<?xml version="1.0" encoding="UTF-8"?>
- <EdiTransaction>
  - <Segment id="ST" Order="1">
        <Element Order="1">810</Element>
        <Element Order="2">57850019</Element>
    </Segment>
  - <Segment id="BIG" Order="2">
        <Element Order="1">20150521</Element>
        <Element Order="2">49777110007820150521</Element>
        <Element Order="3"/>
        <Element Order="4">Utility</Element>
    </Segment>
  - <Segment id="REF" Order="3">
        <Element Order="1">VR</Element>
        <Element Order="2">976197619</Element>
        <Element Order="3">FEDERAL TAX ID</Element>
    </Segment>
  - <Segment id="N1" Order="4">
        <Element Order="1">BT</Element>
        <Element Order="2">MDOT</Element>
        <Element Order="3">91</Element>
        <Element Order="4">497771100078</Element>
    </Segment>
```

Below is an example XLST transformation rule that processes and replaces data element values with a blank if certain conditions are met. In simple words, the following rule dictates to the XSLT processor that:

If XML element is being processed and parent has an attribute “id” with value of “BIG”

and XML Element “Order” attribute has a value of ‘4’

and XML Element text value equal to ‘Utility’

THEN

Replace the value with a blank value.

```
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0">

  <xsl:output
     method="xml"
     indent="yes"
  />
  <xsl:template match="EdiTransaction">
      <EdiTransaction>
        <xsl:copy-of select="@*" />
        <xsl:apply-templates/>
      </EdiTransaction>
  </xsl:template>

  <xsl:template match="Segment">
    <Segment>
       <xsl:copy-of select="@*" />
        <xsl:apply-templates/>
    </Segment>
  </xsl:template>
  <xsl:template match="Element">
    <Element>
       <xsl:copy-of select="@*" />
       <xsl:choose>
         <xsl:when test="../@id='BIG' and @Order='4' and .='Utility'"></xsl:when>
            <xsl:otherwise>
              <xsl:apply-templates/>
           </xsl:otherwise>
       </xsl:choose>
    </Element>
```

```
<?xml version="1.0" encoding="UTF-8"?><EdiTransaction>
<Segment Order="1" id="ST">
<Element Order="1">810</Element>
<Element Order="2">57850019</Element>
</Segment>
<Segment Order="2" id="BIG">
<Element Order="1">20150521</Element>
<Element Order="2">4977711000782015052l</Element>
<Element Order="3"/>
<Element Order="4">Utility</Element>
</Segment>
```

```
<?xml version="1.0" encoding="UTF-8"?><EdiTransaction
<Segment Order="1" id="ST">
<Element Order="1">810</Element>
<Element Order="2">57850019</Element>
</Segment>
<Segment Order="2" id="BIG">
<Element Order="1">20150521</Element>
<Element Order="2">4977711000782015052l</Element>
<Element Order="3"/>
<Element Order="4"/>
</Segment>
```

If the Transformation Specification File Name parameter is provided with a valid Transformation file, for each transaction set the system identifies within the data file, data will be transformed into an XML file. XSLT transformation rules will run against the XML file which will result in a next XML file with the transformed data. Both the XML files (original and transformed) are temporarily saved in the export file directory for the processing. Then, the system compares transformed data to the original values and absorbs all the changes and continues with the further validations. At this point the validations for 997 and 824 errors will begin. After processing all transaction sets, temporary XML files will be deleted by the system. The business users can view the transformed in EDI header and Detail table for any further inquiries.

Award Parsing Information parameter**:**

If an Award Referencing transaction is an outcome and Purchase Order details are provided in the incoming file, the system uses a standard java regex expression that utilizes named caption groups with a specific naming standard for caption groups to match and parse the award information. A batch parameter is provided to supply the java regex expression according to the how the Award Reference is used by the vendor. A major advantage of grouping is that various regex operators can be applied to these groups, readable, flexible and can handle different data formats received in the file easily.

The three award transaction components the Advantage application expects are Department, Transaction Code and Transaction ID and the parameter that supplies regex expression must use the following named group captions within the expression.

- **DocDept** –Transaction Department

- **DocCode** – Transaction Code
- **DocID** – Transaction ID

For example, if a site is expecting Award Transaction information from a vendor in the format (Transaction Department – Transaction Code – Transaction ID), the sample below shows a regex expression matching the string.

RegEx: (?<docdept>\w*)-(?<doccode>\w*)-(?<docid>.*)
Sample Value : A100-PO-123456789
Match: YES
Transaction Department: A100
Transaction Code : PO
Transaction ID : 123456789

If a site is expecting Award Transaction information from a vendor in the format (Transaction ID – Transaction Code – Transaction Department) the sample below shows a regex expression to match the string.

RegEx: (?<docid>\w*)-(?<doccode>\w*)-(?<docdept>.*)
Sample Value : 123456789-PO-A100
Match: YES
Transaction Department: A100
Transaction Code : PO
Transaction ID : 123456789

For instances where award information is not delimited by a specific character, the site must know and use the exact length for each named group and apply the format accordingly. Below is an example regex expression if the first four characters of the Award Reference contain the transaction Department, the next two characters contain the transaction code, and the last two to twenty characters contain the Transaction ID.

RegEx: (?<docdept>\w{4})(?<doccode>\w{2})(?<docid>.{2,20})
Sample Value : A100PO123456789
Match: YES
Transaction Department: A100
Transaction Code : PO
Transaction ID : 123456789

The batch job parameter validations verify that the format provided for the parameter is in a valid format, but does not validate that each invoice has the correct format. After parameters are validated, as the system processes invoices if an Award Reference is provided but it cannot be parsed according to the parameter, the system will reject that invoice with an 824 error since it cannot determine the Award Reference with the information available on the invoice.

997 Functional Acknowledgement Response File Name Prefix parameter**:**
As part of EDI file processing, the system generates a 997 response file to report the status of interchange received. 997 validations are those that verify the file contains the correct information. It verifies items such as mandatory segments are in the file, valid codes are used, data is in the correct format, etc. The 997 response indicates the status of processing of each

invoice. An invoice can pass 997 validations but fail for 824 validations. An invoice that fails 997 validations will not have 824 validations performed against it.

Generally the file includes information on each transaction set within an invoice. However, if a Functional Group or Interchange Control Group fails validations then individual transaction sets will not be validated and will not be in the 997 file.

The response file includes status at functional group level and also by each individual transaction set received in the incoming file. For failed transactions, all the errors encountered will be reported and the invoice's status will be set to Rejected. If a transaction passes all the validations, the status is set to Accepted. If the Acknowledgement Requested value on ISA segment element 13 is equal to '1', the system also creates a TA1 segment in the 997 response to report Interchange Control status.

The system uses the naming convention below for 997 response file names. 997 files are saved to the export directory after the completion of the Prepare EDI Invoices job.

<997 Functional Acknowledgement Response File Name Prefix >_<Sender ID (ISA06 value)>_<Batch Run Date>.txt

824 Application Advice Response File Name Prefix parameter**:**

After invoices are validated for 997 errors, the invoice is validated for 824 errors. 824 errors are errors such as calculations are not correct in the invoice, an address could not be found, the Award Reference could not be found, etc. The detailed list of possible 824 error messages is included in the *CGI Advantage – EDI Invoice Standards Guide*.

If no errors were found for the invoice, the system will not create an 824 transaction for that invoice. All 824 errors will be written in one file. If there are not any 824 errors for any of the invoices in the file, the system will not create an 824 file for the incoming 810 file.

Major Output

- Electronic Data Interchange File (EDIF) table
- Electronic Data Interchange Header (EDIH) table
- Electronic Data Interchange Detail (EDID) table
- Invoice Transactions XML
- 997 and 824 EDI response files (if applicable)

Job Return code

The following table describes potential return codes for Prepare EDI Invoices job.

Return Code	Condition
Successful (1)	EDI data file processed and invoices accepted/rejected per business requirements.
Warning (4)	None of the EDI invoices accepted and XML not created.
Failed (12)	The job will fail under the following conditions: • Parameters Validation failed.

Return Code	Condition
	• Unable find input data and EDI specification files at the location specified in the batch parameter. • Run time exceptions for unexpected situations. When this job ends with a return of code Failed, subsequent jobs in the chain will be set to Inactive.
Terminated (16)	This return code will be issued when the job is terminated by the user. When this job ends with a return of code Terminated subsequent jobs in the chain will be set to Inactive.
System Failure (20)	This return code will be issued when the job is terminated because of database server or network issues. When this job ends with a return of code System Failure, subsequent jobs in the chain will be set to Inactive.

Sort Criteria

N/A

Selection Criteria

N/A

Problem Resolution

The following table shows the possible return codes and recommendations for each processing step.

Step 1: Parameter Validation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All the parameters are validated successfully.	N/A	N/A
Warning (4)	Job ended with a warning because there are no good invoices in the file to create transaction XML.	This is a valid scenario.	N/A
Non-Fatal Error (8)	N/A	This step does not issue this return code.	N/A

Possible Return Codes	Condition	Recommendation	Other Instructions
Failed (12)	Job failed due to Fatal conditions.	In this step, the job can fail under the following two conditions. 1) Encounters any runtime exceptions and 2) Failed during restart. If the job fails because of the runtime exceptions, investigate the exception reported by the process, resolve the error and restart the job.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.
	Failed while restarting the job since another instance of the job has already been run successfully. **Sample Message**: Cannot restart the job since another instance of this job has already been run successfully.	**Recommendation:** Schedule a new job.	
Terminated (16)	Job is terminated manually by the user.	Reason for the termination needs to be investigated. The job can either be restarted or schedule a new job.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled..
System Failure (20)	When the job is terminated because of database server or network issues	Reason for the System Failure needs to be investigated. The job can either be restarted or schedule a new job.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled..

Step 2: Process EDI data file

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All the parameters are validated successfully.	N/A	N/A
Warning (4)	Job ended with a warning because there are no good invoices in the file to create transaction XML.	This is a valid scenario. Examine the 997 file that was created and verify the errors in the file against the specification file used. The vendor may have sent data in the wrong format, or the EDI Specification file may need to be updated to match the vendor's format.	N/A
Non-Fatal Error (8)	N/A	This step does not issue this return code.	N/A
Failed (12)	Job failed due to Fatal conditions.	In this step, the job can fail under the following two conditions. 1) Encounters any runtime exceptions and 2) Failed during restart. If the job fails because of the runtime exceptions, investigate the exception reported by the process, resolve the error and restart the job.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.
	Failed while restarting the job since another instance of the job has already been run successfully. **Sample Message**: Cannot restart the job since another instance of this job	**Recommendation:** Schedule a new job.	

Possible Return Codes	Condition	Recommendation	Other Instructions
	has already been run successfully.		
Terminated (16)	Job is terminated manually by the user.	Reason for the termination needs to be investigated. The job can either be restarted or schedule a new job.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.
System Failure (20)	When the job is terminated because of database server or network issues	Reason for the System Failure needs to be investigated. The job can either be restarted or schedule a new job.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.

Step 3: Create Transaction XML

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All the parameters are validated successfully.	N/A	N/A
Warning (4)	Job ended with a warning because there are no good invoices in the file to create transaction XML.	This is a valid scenario.	N/A
Non-Fatal Error (8)	N/A	This step does not issue this return code.	N/A
Failed (12)	Job failed due to Fatal conditions.	In this step, the job can fail under the following two conditions. 1) Encounters any	If another instance of the job has already been scheduled and ran successfully, then this job should not be

Possible Return Codes	Condition	Recommendation	Other Instructions
		runtime exceptions and 2) Failed during restart. If the job fails because of the runtime exceptions, investigate the exception reported by the process, resolve the error and restart the job.	restarted – only new job should be scheduled.
	Failed while restarting the job since another instance of the job has already been run successfully. **Sample Message**: Cannot restart the job since another instance of this job has already been run successfully.	**Recommendation:** Schedule a new job.	
Terminated (16)	Job is terminated manually by the user.	Reason for the termination needs to be investigated. The job can either be restarted or schedule a new job.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.
System Failure (20)	When the job is terminated because of database server or network issues.	Reason for the System Failure needs to be investigated. The job can either be restarted or schedule a new job.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.

Load EDI Invoices Chain: Load EDI Invoices Job

Chain or Job Name	Load EDI Invoices

Recommended Frequency	The Load EDI Invoices job can be run daily as part of the nightly cycle or on demand.
Single Instance Required	Yes.
Can be restarted?	Optionally, based on the Save Restart Information parameter.
Reports generated	No. All errors will be written to the log.

Overview

The Load EDI Invoices job loads the records from the XML File, generated by the Create EDI Invoices job, into the Transaction Catalog. This job uses the common utility to load the records into the Transaction Catalog. This job first validates the batch parameters. If the parameters are valid, then it loads the records into the Transaction Catalog. If the parameters are not valid, the job issues appropriate messages and ends with a status of Failed. Once the records are loaded into the Transaction Catalog, the summary information is written into the log as how many records were in the input file and how many records loaded successfully.

The job can be restarted if it fails, provided the Save Restart Information parameter is selected. If the failure occurred after the parameter validation, then the job should be restarted after resolving the errors. If the Save Restart Information parameter is not selected or if the restart is not the immediate option, then the new job can be rescheduled but before rescheduling the job, the transactions loaded by the failed job should either be processed or discarded so that they do not remain in the catalog.

Restartability Information

The job can be restarted only when the Save Restart Information Flag parameter is set to *True*. If the parameter is set to False, then the job cannot be restarted.

Major Input

- Invoice XML file

Batch Parameters

Parameter	Description	Default Value
Maximum Block Size (MAX_BLOCK_SIZE)	Required Field. This is to limit the number of transactions per parameter file. If the number of records written per parameter file exceeds the block size then there will be a new parameter file created for the same priority.	100
Apply Override in Import Mode	Apply Override in Import Mode.	true

(APPLY_OVERRIDES)		
Bypass Approvals in Import Mode (BYPASS_APPROVAL)	System Maintenance Utility flag to indicate Bypass Approvals in Import Mode. If set to True the Bypass Approval Indicator of all imported transactions is set to True. The Bypass Approval Indicator allows the transaction to be submitted without going through approval processing (workflow). Note that if a transaction is subsequently edited by a user online, the Bypass Approval Indicator is removed.	false
Bypass ADNT in Import Mode (BYPS_ADNT_FL)	System Maintenance Utility flag to indicate Bypass ADNT in Import Mode.	true
Commit Block Size (COMMIT_SIZE)	Optional number of records to commit at a time. If left blank, the ADV30Parms.ini file will supply a value.	10
Transaction Status for Import Mode (DOC_STA_CD)	System Maintenance Utility Transaction Status for Import Mode. Required field that will be used by the program when loading transactions. Valid values are 1 - Held and 2 - Ready. Transactions loaded as Held will not be selected by another process searching for transactions to submit. Ready transactions will be selected by such a program. Held is often used when some user action will be required.	2
Detail Error Message (DTL_IMP_MSG_FL)	Indicates whether error message details for each failed, to be processed record should be recorded in the error file.	false

Error File Name (ERROR_FILE_NM)	Required Field. Name of the Error file.	$$AMSLOGS$$/LoadEDIExcept.txt
Name of input xml. (FILE_NM)	Required Field. Name of input xml.	$$AMSEXPORT$$/EDIInvoices.xml
Override Level for Import Mode (OVERRIDE_LVL)	System Maintenance Utility Override Level for Import Mode. The override level of the user is set to facilitate the transaction being submitted without someone having to open the transaction manually and apply overrides from the online system. Note that if a transaction is subsequently edited by a user online, the override level is removed.	N/A
Save Restart Information Flag (RESTART_FL)	System Maintenance Utility to indicate Save Restart info in Submit Mode. If value is true then job can be restarted if there are errors. If value is false then job cannot be restarted if there are errors.	true

Please refer to the “SMU Transaction Upload Job” run sheet in the *CGI Advantage Financial – Utilities Run Sheets* guide for the full list of SMU Transaction upload batch parameters.

Major Output

Electronic Invoice transactions in Draft state.

Award Referencing Invoice transactions in Draft state.

Stand Alone Invoice transactions in Draft state.

Job Return code

The following table describes potential return codes for the Load EDI Invoices job.

Return Code	Condition
Successful (1)	EDI data file processed and invoices accepted/rejected per business requirements.
Failed (12)	The job will fail under the following conditions:

Return Code	Condition
	• Parameters are invalid. • Unable find input data and EDI specification files at the location specified in the batch parameter. • Run time exceptions for unexpected situations. When this job ends with a return of code Failed, subsequent jobs in the chain will be set to Inactive.
Terminated (16)	This return code will be issued when the job is terminated by the user. When this job ends with a return of code Terminated subsequent jobs in the chain will be set to inactive.
System Failure (20)	This return code will be issued when the job is terminated because of database server or network issues. When this job ends with a return of code System Failure, subsequent jobs in the chain will be set to Inactive.

Sort Criteria

N/A

Selection Criteria

N/A

Problem Resolution

If the job cannot be restarted immediately, then the transactions loaded by this job should be either processed or discarded before rescheduling the new job. To process the loaded transactions, submit the Submit EDI Invoices job in that chain. This job would be set to Inactive since the Load EDI invoices job failed. The Submit job will submit the transactions that were generated during that Load EDI Invoices chain.

Possible Return Codes	Condition	Recommendation	Other Instructions
Failed (12)	Failed while restarting the job since another instance of the job has already been run successfully. **Sample Message**: Cannot restart the job since another instance of this job has already been run successfully.	**Recommendation:** Schedule a new job.	

Load EDI Invoices Chain: Submit EDI Invoices Job

Chain or Job Name	Submit EDI Invoices
Recommended Frequency	The Submit EDI Invoices job can be run daily as part of the nightly cycle or on demand.
Single Instance Required	No
Can be restarted?	Yes. See the “Overview” and the “Problem Resolution” sections for more details.
Reports generated	No. All of the exceptions are written to the error file.

Overview

This job submits the transactions listed in the input parameter file that was generated by the Create EDI Invoices job. This process is a part of the Load EDI Invoices chain batch job that reads the SMU job parameter file to spawn multiple SysManUtil Submit jobs. In order to run the EDI chain Submit step in parallel, the Job Interaction Client (JIC) must be used (refer the *CGI Advantage System Administration Guide* for more information on JIC). Also, when running in parallel, the **Process Assigned Jobs Only** flag must be set to FALSE in the Job Server Control for all job managers.

Restartability Information

The job can be restarted only when the Save Restart Information Flag parameter is set to True. If the parameter is set to False, then the job cannot be restarted**.**

Major Input

SMU job parameter file

Draft Invoice Transactions in the catalog

Batch Parameters

Parameter	Description	Default Value
Parameter file (PARM_FILE)	Parameter file to Submit Transactions.	$$AMSPARM$$/EDIParm.txt

This PARM_FILE contains only the following subset of SMU parameters.

Parameter	Default Value
Action Code (ACTN_CD)	DOCSUBMIT
COMMIT_SIZE (COMMIT_SIZE)	1
Progression Counter (PROG_CTR_SZ)	1000
Save Restart Information Flag (RESTART_FL)	True

Exception File for Submit Mode (EXCEP_REP_FILE_NM)	$$AMSLOGS$$/SubmitEDIExcept.txt

Note: This job uses only a subset of the SMU submit job parameters. For a full list of available parameters for the SMU submit job, refer to the "SMU Transaction Submit Job" run sheet in the *CGI Advantage Financial – Utilities Run Sheets* guide.

Major Output

The transactions would have been submitted to Final or Rejected.

Job Return code

The following table describes potential return codes for Prepare EDI Invoices job.

Return Code	Condition
Successful (1)	EDI data file processed and invoices accepted/rejected per business requirements.
Failed (12)	The job will fail under the following conditions: • Parameters are invalid • Unable find input data and EDI specification files at the location specified in the batch parameter. • Run time exceptions for unexpected situations. When this job ends with a return of code Failed, subsequent jobs in the chain will be set to Inactive.
Terminated (16)	This return code will be issued when the job is terminated by the user. When this job ends with a return of code Terminated subsequent jobs in the chain will be set to Inactive.
System Failure (20)	This return code will be issued when the job is terminated because of database server or network issues. When this job ends with a return of code System Failure, subsequent jobs in the chain will be set to Inactive.

Sort Criteria

N/A

Selection Criteria

N/A

Problem Resolution

If the job ends with a return code of Failed and above, the job can be restarted only when the Save Restart Information parameter is selected and another instance of the job has not been

scheduled and run successfully. If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – a new job should only be scheduled.

If the restart is not an immediate option and the fatal error is because of a few transactions, the rest of the transactions can be submitted manually or discarded manually depending on the Issue. The Transaction IDs can be found on the input parameter file.

The following table shows the possible return codes and recommendations for each processing step specific to the job in the chain. For general errors and recommendations, refer to the "SMU Transaction Submit Job" run sheet in the *CGI Advantage Financial – Utilities Run Sheets* guide.

Possible Return Codes	Condition	Recommendation	Other Instructions
Failed (12)	Failed while restarting the job since another instance of the job has already been run successfully. **Sample Message**: Cannot restart the job since another instance of this job has already been run successfully.	**Recommendation:** Schedule a new job.	

Load EDI Invoices Chain: EDI Invoice Exception Report Job

Chain or Job Name	EDI Invoice Exception Report
Recommended Frequency	The EDI Invoice Exception Report job can be run daily as part of the nightly cycle or on demand.
Single Instance Required	Yes
Can be restarted?	Yes
Reports generated	Yes. The job generates the report in pdf and html formats.

Overview

This job in the Load EDI Invoices chain generates an exception report that lists all of the errors encountered when the Invoice transaction was submitted in the earlier step. This report will also be generated when the first job, Prepare EDI invoices, is in a warning state. The report also provides information on the invoices from the incoming 810 file and the status of those invoices after the first three jobs of this chain are executed. This report will have 2 sections:

Invoices from the EDI file and their Status from the EDI Invoice Header table.

Invoice transactions that failed during executing the SMU job in the EDI Invoice Submit Job (data comes from SubmitEDIExcept.txt). The report will include the reason(s) why the transaction failed to submit if the user chooses to view the detailed report.

The report will have two formats – Summary and Detail. Summary provides the total only of each invoice status and does not provide the detailed information. Detailed provides a listing of each individual invoice and additionally provides the error message status for invoices not submitted to final in Advantage. User choices for the Load EDI Invoices batch and Submit EDI Invoices batch will also drive the data in the EDI transaction exception report.

The following table shows the progression messages issued in this job.

Processing Steps	Messages
Validation Parameters	• Parameter File Name cannot be blank. • Parameter Location is invalid. • Parameter File Not Found at Parameter Location. • Parameter File is empty. • Exception Report File Name cannot be blank. • Exception Report File Path is invalid. • Parameter File Not Found at Exception Report File Path.
Generation EDI Exception Report	• Reports output folder mapped. • Rendering report started. • Rendering report completed.

Restartability Information

This job cannot be restarted in this step. If the job fails a new job should be scheduled after correcting the errors that caused the job to fail**.**

Major Input

Invoices associated to the processed file from the first batch job and their invoice statuses on the EDI Invoice Header (EDIH) table.

SubmitEDIExcept.txt which contains the data for the invoices that failed to submit from the Submit EDI Invoices batch job.

Common chain parameter file. The common parameter file contains the parameter details for the type of report format and the client name.

Batch Parameters

Parameter	Description	Default Value
Exception File Name (EXCEP_REP_FILE_NM)	Required field. Indicates the file name of the exception report that will be generated by the EDI Submit Invoices batch.	EDIDocExcept.txt
Exception Report Output location for Generate Exception Report	Required field. Exception Report Output location.	$$AMSLOGS$$

(AMSLOGS)		
Common Chain Parameter File (AMSPARM)	Required field. Defines the parameter file that is to be used by the Exception Report to create the report. This is the parameter file created by the Prepare EDI Invoices xml Generation job.	EDIParm.txt
Parameter Location for Generate Exception Report Job (PARM_FILE)	Indicates the location that the Parameter file can be found.	$$AMSPARM$$

Major Output

EDI Invoice exception report

Job Return code

The following table describes potential return codes for EDI Invoice Exception report job.

Return Code	Condition
Successful (1)	All of the validations are performed successfully and there are no records eligible for exception.
Non-Fatal Error (8)	• For the invoice status section: not all invoices have a status of Processed. • For the SMU status section: not all invoices have a status of Submitted. • An invoice transaction fails to submit to Final.
Failed (12)	The job will fail under the following conditions: • Input parameter is not found • Runtime exceptions are encountered. • Run time exceptions for unexpected situations. When this job ends with a return of code Failed, subsequent jobs in the chain will be set to Inactive.
Terminated (16)	This return code will be issued when the job is terminated by the user. When this job ends with a return of code Terminated subsequent jobs in the chain will be set to Inactive.
System Failure (20)	This return code will be issued when the job is terminated because of database server or network issues. When this job ends with a return of code System Failure, subsequent jobs in the chain will be set to Inactive.

Sort Sequence

None

Selection Criteria

Select all the records from EDI Header where the EDI Invoice File Number is the EDI file number that was processed in the first batch job, Prepare EDI Invoices.

Select all the records from Invoice Transaction Header table where the EDI Invoice Header Number is the EDI Header number from EDI Header table, for EDI file number that was processed in Prepare EDI Invoices batch.

Problem Resolution

If the process fails for any reason, check if all parameters Exception File Name, Exception Report Output location, Common Chain Parameter File and Parameter Location for Generate Exception Report Job are correct. Correct the problem and rerun the process.

2.1.33 Load Payment Hold by TIN

Chain or Job Name	Load Payment Hold by TIN Chain
Recommended Frequency	Nightly
Single Instance Required	Yes
Can be restarted?	Yes
Reports generated	Payment Hold TIN Records Exclusion Report

Overview

The Load Payment Hold by TIN chain job is used to load the records from the interface xml file into a temporary holding table. It then inserts or updates each record into the Payment Hold by TIN Table. The chain job consists of 2 batch jobs. They are as follows:

- **SysManUtil**: This is the first job is the chain job. The main purpose of this job is to select the records from the interface xml file and load it into the temporary holding file. This job loads the xml file and inserts records into the temporary holding table R_PYMT_HLD_TIN_TMP.
- **Load Payment by TIN Process**: The purpose of this job is to read all the records from the temporary holding table and then either updates the corresponding record on the Payment Hold by TIN table or to insert a new record into the Payment Hold by TIN Table. This job uses R_PYMT_HLD_TIN_TMP created in the SysManUtil step as an input which contains the records that need to be processed.

Major Input

Data from the following tables:

- Interface XML file
- Temporary Holding Table (R_PYMT_HLD_TIN_TMP)

Major Output

Following are the output of the chain job:

- Payment Hold TIN Records Exclusion Report
- Payment Hold by TIN Table (R_PYMT_HLD_TIN)

Chain Return Code

The following table indicates the potential return codes with which the chain will end:

Return Code	Condition
Successful (1)	All the jobs ends successfully
Warning (4)	One of the job in the chain ends with a return code of

	"Warning"
Non-Fatal Error (8)	One of the job in the chain ends with a return code of "Non-Fatal Error"
Failed (12)	One of the job in the chain ends with a return code of "Failed"
Terminated (16)	One of the job in the chain ends with a return code of "Terminated"
System Failure (20)	One of the job in the chain ends with a return code of "System Failure"

Problem Resolution

Please refer to the individual job "Problem Resolution" section for more details

Load Payment Hold by TIN Chain: SysManUtil

Chain or Job Name	SysManUtil
Recommended Frequency	Nightly
Single Instance Required	Yes
Can be restarted?	Yes
Reports generated	None

Overview

SysManUtil reads the interfaced file with vendor records that either have outstanding debts where holds should be applied or have settled their outstanding debts where holds need to be released and then loads the records to a temporary holding table (R_PYMT_HLD_TIN_TMP) with the table action set as *Table Import*. The holding table has only one unique key field. This is the Auto Generated Sequence Number.

The SysManUtil job validates the following parameters and throws error message and terminates the job in case the parameters do not get validated:

Restartability

Job cannot be restarted. If the job fails then the entire interface file will be read and then loaded into the temporary holding table. A new chain should be scheduled for the same File ID.

Major Input

- Interface XML File

Batch Parameters

Parameter	Description	Default Value
Parameter Directory AMSPARM	Parameter Directory location	$$AMSPARM$$
Chain Job ID CHAIN_JOB_ID	Chain Job ID	$$@CHAINJOBID@$$
Parameter File Name PARM_FILE_NM	Interface XML File Name	No Default
Commit Block Size COMMIT_SIZE	When the number of records that have been processed equals this param, a database commit is done.	100
Table Action TABLE_ACTN	The table action that needs to be done.	Table Import

Major Output

- Temporary Holding Table (R_PYMT_HLD_TIN_TMP): The temporary holding table.

Job Return Code

The following table shows the potential job return codes for the Inferences & Validations job:

Return Code	Condition
Successful (1)	All the records in the XML file are processed successfully
Warning (4)	This job does not issue this Return Code.
Non-Fatal Error (8)	This job does not issue this Return Code.
Failed (12)	The job fails under the following conditions: • Parameters are invalid • Issues in SysManUtil while discarding the transaction • Technical/System failure
Terminated (16)	This return code is issued when the job is terminated by the user. When this job ends with a return of code Terminated subsequent jobs in the chain will be set to inactive.
System	This return code is issued when the job is terminated

Return Code	Condition
Failure (20)	because of database server or network issues. When this job ends with a return of code System Failure, subsequent jobs in the chain are set to inactive.

Selection Criteria

N/A

Sort Sequence

N/A

Problem Resolution

The following table shows the possible return codes and recommendations for each processing step.

Step 1: SysManUtil

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All the records are successfully loaded. This is done by the SysManUtil “Table Import” action functionality.	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Job failed due to Fatal conditions	In this step, the job can fail under the following two conditions. 1) Encounters any runtime exceptions and 2) Failed during restart. If the job fails because of the runtime exceptions, investigate the exception reported by the process, resolve	Schedule a new job

Possible Return Codes	Condition	Recommendation	Other Instructions
		the error and restart the job.	
Terminated (16)	Job is terminated manually by the user.	Reason for the termination needs to be investigated. A new job can be scheduled	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.
System Failure (20)	When the job is terminated because of database server or network issues	Reason for the System Failure needs to be investigated. A new job can be scheduled.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.

Load Payment Hold by TIN Chain: LoadPaymentHoldbyTIN

Chain or Job Name	LoadPaymentHoldbyTIN
Recommended Frequency	Nightly
Single Instance Required	Yes
Can be restarted?	Yes
Reports generated	Payment Hold TIN Records Exclusion Report

Overview

The Load Payment Hold by TIN process reads the records from the temporary holding table and adds the valid records to the Payment Hold by TIN table. It takes into consideration the values on the Date Debt Added and Date Debt Removed to decide on whether the current record should be inserted or updated on the Payment Hold by TIN Table.

This method calls the validateParameters () to validate the parameters passed to the Load Payment by TIN process. If validation of parameters completes successfully then for each record in the temporary table, 'Date Debt Added' and 'Date Debt Removed' is read. Depending on the values in these two date fields, either a new record is added to the Payment Hold by TIN table or an existing record is updated on the Payment Hold by TIN table.

The process generates a Payment Hold TIN Records Exclusion Report showing details on the TIN, TIN Type, Vendor Name, Hold Type, Date Debt Added, Date Debt Removed and Exception

Comment for all the records which could not be added or updated to the Payment Hold by TIN table.

Process Steps	Messages
1. Validate the Parameters	• Started Validating the Parameters • If parameters are invalid, then write the invalid param to job log
2. Selection of records	• Started selection of records • If the selection returns 0 records, then the following message will be issued: “No eligible record found”.
3. Insert the record	• Inserting the record started. • Inserting the record completed.
4. Update the corresponding record	• Updating the record started. • Updating the record completed.
5. Select the records to be written to Exclusion Report	• Writing error records started. • Writing error records completed.

Restartability

Job can be restarted. If the job fails and then is restarted, the flag Row Processed is used to determine if the record has been processed or not. The chain job should start processing only those records which do not have this flag set to true.

Major Input

- Temporary Holding Table (R_PYMT_HLD_TIN_TMP)

Batch Parameters

Parameter	Description	Default Value
Parameter Directory AMSPARM	Parameter Directory location	$$AMSPARM$$
Chain Job ID CHAIN_JOB_ID	Chain Job ID	$$@CHAINJOBID@$$
Progression Message Block Size PROG_CTR_SZ	Required. This field provides the information about the record being processed.	No Default.

Commit Block Size COMMIT_SIZE	When the number of records that have been processed equals this param, a database commit is done.	100

Major Output

- Payment Hold by TIN table
- Payment Hold TIN Records Exclusion Report

Job Return Code

The following table shows the potential job return codes for the Inferences & Validations job:

Return Code	Condition
Successful (1)	All the selected records are processed successfully
Warning (4)	No eligible records found.
Non-Fatal Error (8)	N/A
Failed (12)	The job fails under the following conditions: • Parameters are invalid • Run time exceptions for unexpected situations. When this job ends with a return of code Failed, subsequent jobs in the chain will be set to inactive.
Terminated (16)	This return code will be issued when the job is terminated by the user. When this job ends with a return of code Terminated subsequent jobs in the chain will be set to inactive.
System Failure (20)	This return code will be issued when the job is terminated because of database server or network issues. When this job ends with a return of code System Failure, subsequent jobs in the chain will be set to inactive.

Step 1: Parameter Validation

- Verify the Fiscal Year.
- Verify the Default Hold Department Code
- Verify the Default Hold Unit Code
- If all the parameters are validated successfully then retrieve the records from the temporary holding table

Step 2: Selection Criteria

- Select all the records from the temporary holding table which do not have any value in the 'Row Processed' field.
- For each selected record
  - Get the 'Date Debt Added' and 'Date Debt Removed' fields.

If 'Date Debt Added' is NULL or Invalid

- Set the 'Exception Comment' field to 'EC2' on the temp table

If 'Date Debt Added' is Valid and 'Date Debt Removed' is Empty

- Inserts the record into the Payment Hold by TIN table

Else If 'Date Debt Added' is Valid and 'Date Debt Removed' is Not Empty

- Update the record on the Payment Hold by TIN table

Step 3: Inserting the record

- Insert the record on the Payment Hold By TIN Table with the corresponding values from the temporary holding table

Step 4: Updating the record

- Query the Payment Hold by TIN table by using the TIN, TIN Type, Hold Type and the Date Debt Added values.

If a record is found for the queried values

- Set the Date Debt Removed on the Payment Hold by TIN with the Date Debt Removed value from the Temporary Holding Table

Step 5: Selecting Error Records

Query the temporary table for these fields: TIN, TIN Type, Hold Type, Date Debt Added, Date Debt Removed and Exception Comment for records that have no value on the 'Action Performed' field and with some value in the 'Exception Comment' field and ordered by TIN, TIN Type and Hold Type

Sort Sequence

N/A

Problem Resolution

The following table shows the possible return codes and recommendations for each processing step.

Step 1: Parameter Validation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All the parameters are validated successfully.	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Job failed due to Fatal conditions	In this step, the job can fail under the following conditions. 1) Encounters any runtime exceptions and 2) Failed during restart. If the job fails because of the runtime exceptions, investigate the exception reported by the process, resolve the error and restart the job.	Schedule a new job
Terminated (16)	Job is terminated manually by the user.	Reason for the termination needs to be investigated. A new job can be scheduled	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.
System Failure (20)	When the job is terminated because of database server or network issues	Reason for the System Failure needs to be investigated. A new job can be scheduled.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.

Step 2: Selecting the records

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	N/A	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	There are no records in the Temporary Holding Table	N/A	Write a message to the job log “Log the message ‘No Record found into the interfaced file’”
	Failed because of runtime exceptions for an unexpected situation.	Failure reason needs to be investigated before scheduling a new job.	Failed because of runtime exceptions for an unexpected situation.
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated and a new job scheduled.	N/A
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated and a new job scheduled.	N/A

Step 3: Inserting the record

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	The record has been successfully inserted.	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	Errors that can be generated due to some failed validations while inserting into the Payment Hold by TIN Table	N/A	Get the error thrown by the Payment Hold by TIN table and then write it to the exception

			report.
Failed (12)	Failed due to issues in reading the data row	Investigate the reason for the datarow not being read and re-run the chain.	N/A
	Failed because of runtime exceptions for an unexpected situation.	Failure reason needs to be investigated before scheduling a new job.	N/A
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated and a new job scheduled.	N/A.
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated and a new job scheduled.	N/A.

Step 4: Updating the record

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	The record has been successfully updated.	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	Errors that can be generated due to some failed validations while updating the corresponding record on the Payment Hold by TIN Table	N/A	Get the error thrown by the Payment Hold by TIN table and then write it to the exception report.
Failed (12)	Failed due to issues in reading the data row	Investigate the reason for the datarow not being read and re-run the chain.	N/A
	Failed because of runtime exceptions for an unexpected situation.	Failure reason needs to be investigated before scheduling a new job.	N/A
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated and a new job scheduled.	N/A.

System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated and a new job scheduled.	N/A.

Step 5: Selecting error records

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	The Exclusion Report has been successfully created.	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	Errors that can be generated due to some failed validations while inserting into the Payment Hold by TIN Table	N/A	Get the error thrown by the Payment Hold by TIN table and then write it to the exception report.
Failed (12)	Failed due to issues in creating the report.	Investigate the reason for the report not being created and re-run the chain.	N/A
	Failed because of runtime exceptions for an unexpected situation.	Failure reason needs to be investigated before scheduling a new job.	N/A
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated and a new job scheduled.	N/A.
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated and a new job scheduled.	N/A.

2.1.34 Mass Cancellation

Description

The Mass Cancellation batch process cancels Checks (automated or manual), Warrants or Electronic Fund Transfers (EFT). It gives users the flexibility to Reschedule, Hold and Cancel Checks / Warrants / EFTs in bulk.

The user enters the Check Range to be cancelled, Cancellation Type, Cancellation Reason, Hold Type, Hold Request Description, Payment Hold Type Department, Payment Hold Type Unit, Bank Account Code and Reschedule Date in the Disbursement Cancellation Parameter (R_AP_DISBCAN_PARM) table and sets the Active Status to True. The process selects all records from the Disbursement Cancellation Parameter table that have the Active Status set to True and the Process Date set to null. The disbursement transaction to be cancelled is selected on the basis of the bank code and check number entered in the parameter table. For every check number in the check range entered on the parameter table, the process selects the Final disbursement transaction and creates a Cancellation version. The Cancellation Type, Cancellation Reason, Hold Type, Hold Request Description, Payment Hold Type Department, Payment Hold Type Unit, Reschedule Date and Comments entered by the user in the parameter table are set in the Cancellation version and the transaction is processed to Final. Finally, the Mass Cancellation process generates a report listing all of the checks that were successfully cancelled.

The Job return status is set to “Successful” if the disbursement transactions are successfully processed and is set to “Failed” if an exception is encountered. The job return status is set to Failed when there are not active records on the Disbursement Cancellation parameters table.

When to Run

The Mass Cancellation process can be run on demand.

Major Input

- Disbursement Cancellation Parameter (R_AP_DISBCAN_PARM)
- Check Reconciliation table (R_AP_CHK_RECON)
- MD Transaction
- AD Transaction

Output

- A report is generated which contains a list of all of the checks successfully cancelled along with the Cancellation Type, Cancellation Reason, Reschedule Date and Reason.
- A report is generated which contains a list of all of the cancelled checks that could not be submitted.
- Disbursement Cancellation Parameter (R_AP_DISBCAN_PARM)
- Check Reconciliation table (R_AP_CHK_RECON)
- MD Transaction
- AD Transaction

Parameters

Custom Parameters

Disbursement Cancellation Parameters (R_AP_DISBCAN_PARM)

Description (Caption)	Parameter (Attribute) Name	Data Type/Length
Bank Account Code	BANK_ACCT_CD	Text/4
Cancellation Type	CAN_TYP	Number/Integer
Cancellation Reason	CAN_REAS_CD	Text/50
Hold Type	HLD_TYP_DC	Text/6
Hold Type Description	HLD_RQST_DSCR_DC	Text/120
Payment Hold Type Department	PY_HLD_TYP_DEPT_DC	Text/4
Payment Hold Type Unit	PY_HLD_TYP_UNIT_DC	Text/8
Start Check	STRT_CHK	Text/15
End Check	END_CHK	Text/15
Active Status	ACTIVE_STA	Yes/No
Schedule Payment Date	SCHEDL_PMT_DT	Date
Process Date	PROCESS_DT	Date/Time

Batch Parameters

Job	Parameter	Description	Default Value
Mass Cancellation	Client Name (CLIENT_NM)	Client name for Report	No Default

Sort Sequence

Parameter Code

Selection Criteria

Select records from the Disbursement Cancellation Parameter table, where

Active Status is ‘true’ and Process Date is null

Select records from check reconciliation table with Outstanding Status for the Check Range specified in the parameter.

Select records from Disbursement Transaction.

Based on the Check range, only New or Modified Final versions of the Disbursement transaction are picked.

Select records from Cancelled Draft versions of the disbursement transaction.

If a Cancellation Draft version already exists then it picks up that version and submits the transaction.

Problem Resolution

This batch program produces new Advantage transactions, which are subject to the line limit functionality constraints. Sites should ensure that they run this job with parameters set to ensure that the created transactions are within the line limit controls.

If the job fails due to any reason, the job can be restarted.

2.1.35 Multi-Process Disbursement Printing

Job Name	MultiProcess Disb Print
Recommended Frequency	After one or more AD Chain runs
Single Instance Required	Yes
Can be restarted?	No
Reports generated	See Disbursement Print for Multi-Process Disb Print

Overview

The Multi-Process Disbursement Printing job is an alternative to the Disbursement Printing job with all the same parameters and capabilities with the additional capability of spawning multiple instances of the Disbursement Printing job for faster throughput. As delivered, this job spawns a different job than the standard Disbursement Print batch job. The Disbursement Print for Multi-Process Disbursement Printing job is a report job. The report is intended to verify the accuracy of the print across multiple print jobs. The Sub Process Catalog ID identifies the special report job. If the report is not desired, that parameter default can be changed on Batch Setup.

This run sheet is abbreviated for just the differences between the multi process parts. The Disbursement Printing run sheet should be consulted for all other aspects.

The following table is intended to break the job step major processing steps with any applicable job log messages.

Process Steps	Messages
1. Parameter Validation	• Validating Batch Parameters • Parameters are valid or invalid depending on the Validation. If the parameter is invalid, the invalid value is displayed in the log. • Batch Parameter validation completed
2. Selection and Evaluation of Records	The following messages are displayed only in offline mode. The following set of messages appear when the Disbursement Type parameter is *Check*. • Run Started • Total Number of Checks: <Number> • Check Key Range Up to: <Check No> • Check Key Range From: < Check No> to < Check No> • Check Key Range From: < Check No> • Job - <Job ID> - Spawned (number of spawned jobs based on multiple threads and total number of checks) • Print Status Indicator Updated successfully. • All XML files are renamed successfully.

Process Steps	Messages
	• Run Ended

Restartability Information

This job cannot be restarted. If the job failed due to any reason, a new job should be scheduled after correcting the errors that caused the job to fail.

Batch Parameters

Refer to the Disbursement Printing job for the description of the batch parameters other than the following:

Parameter	Description	Default Value
Temp Location at Disbursement Printing Job (AMSTEMP)	A required parameter for the location to write temporary output from the job spawned.	No Default
Number of Checks per file (CHECK_COUNT)	An optional selection parameter to process the number of checks in the spawned job. If left blank or any invalid value, 1000 is the default.	No Default
Report ID (EXPORT_REPORT_ID)	An optional value to be written in the report generated by MultiProcessDisbPrinting.	No Default
Sub Process Catalog ID (PROCESS_CATALOG_ID)	A required operational parameter for the Catalog ID for the job spawned. (Catalog ID for Disbursement Printing is 4206).	4206
Queue Class (QUEUE_CLASS)	An optional parameter. If left blank, 1 is the default.	No Default
Stagger time in seconds (STAGGER_TIME)	An optional parameter for waiting time between two spawned jobs' run. If left blank, 30 is the default	No Default
Thread Count (THREAD_COUNT)	A required number of jobs to be spawned when selection reveals multiple threads are necessary. If left blank, 1 is the default.	No Default

For all other run sheet details, refer to the Disbursement Printing run sheet.

2.1.36 Disbursement Printing for Multi-Process Disbursement Printing

Job Name	Disbursement Printing for MultiProcess Disb Print
Recommended Frequency	After one or more AD Chain runs
Single Instance Required	Yes
Can be restarted?	No
Reports generated	See Disbursement Print for Multi-Process Disb Print

Overview

The Disbursement Printing for Multi-Process Disbursement Printing is a copy of the Disbursement Printing functionality with the added capability to generate the Automated Disbursement Register for verification.

Batch Parameters

Refer the Disbursement Printing job and the Multi-Process Disbursement Printing run sheets for details.

Process Steps	Messages
1. Parameter Validation	The following messages will be displayed in offline mode. • Validating batch parameters. • Parameters are valid or invalid depending on the validation. If the parameter is invalid, then the message for invalid parameter will be displayed in the log. It will be followed by message “Batch parameter validation failed”. • Batch parameter validation completed.
2. Selection and Evaluation of Records	All of the following messages are displayed only in offline mode. The following set of messages appears when the Disbursement Type parameter is Check. • Start processing Check records • If the selection returns zero records, then this step will be completed and the following message will be issued: “No eligible Check records found”. • Total check records processed with valid check number: <Count> • Total Check records with 'Null' Check Number: <Total Count> • Total Check records with '000000000000000' Check Number: <Total Count> • Performing Print Status Indicator updates on Source table

	• Total number of records updated for Print Status Indicator: <Total Count> • If one or more records updated then the message displayed will be: “Total number of records updated for Print Status Indicator: <Total Count>” • <Count> will be in multiple of value provided for parameter PROG_SZ_CTR. • <Total Count> will be the total number of records processed.
3. XML Creation and Printing	The following messages will be displayed for XML generation and report creation. • Creating XML file and Printing Check records • Forms output folder mapped. • Creation of XML file and Printing of Check records completed. • Reports output folder mapped. • HTML report file path: /apps/CGIADV/RTFiles/fin/ReportOutput/Reports/DisbursementPrinting/Output_<Spawned Run ID>//<XML file Name>/Html/DisbursementPrinting.html • PDF report file path: /apps/CGIADV/RTFiles/fin/ReportOutput/Reports/DisbursementPrinting/Output_<Spawned Run ID>//<XML File Name>/PDF/DisbursementPrinting.pdf • Rendering report started. • Rendering report completed. • Printing process completed successfully.

Restartability Information

This job cannot be restarted. If the job failed due to any reason, a new Multi-Process Disbursement Printing job should be scheduled after correcting the errors that caused the job to fail.

2.1.37 Multi Process Disb Journal Posting

Chain or Job Name	Multi Process Disbursement (Disb) Journal Posting
Recommended Frequency	On Demand
Single Instance Required	No
Can be restarted?	Yes
Reports generated	No

Overview

The Multi Process Disbursement Journal Posting job is a job used when automatic disbursements use the *Asynchronous Posting* method of Journal Posting. Please see the Journal Posting Control field on Transaction Control for the delivered AD and EFT transaction codes. When the AD Chain is run with this posting method, the posting lines from disbursements are not automatically written to the Accounting Journal and other journals. Instead those disbursements are numbered by a separate job. When numbered and checks have been made for accuracy, then the posting lines need to be 'journalized'. That is where the Multi Process Disbursement Journal Posting job comes in.

The job first validates all batch parameters. The second step of the job selects all of the transactions with Transaction Type *AD* that have the Transaction Phase of *Final* and have a Journal Posting Indicator of *Not Ready to Post* on the Posting Catalog. For these records, the job updates the Journal Posting Indicator for those records to *Ready to Post*. This behavior is the same as that of the Journal Posting Initiator batch job.

The third step of the job is to spawn one or more Journal Engine jobs to perform the Journal Posting of the selected posting lines to the applicable journals.

Major Input

- Posting Catalog (PSTNG_LN_CAT)

Job Parameters

Parameter	Description	Default Value
JOB_MANAGER_NAME	Required Job Manager Name	advantage/AMSJobManager
SLEEP_INTERVAL	Optional polling frequency (in seconds) for internal controller thread for checking child processes. This parameter is not required, but 10,000 will default if left blank.	10000
LOAD_BAL_MODE	Required Load Balancing Mode	1 (True)
DOC_CODE	Required listing of transaction	AD,EFT

	code for selection. The possible values are AD or EFT.	
ITERATION_COUNT	Optional iteration count that defined the length of time the job will wait for the spawned job to complete. If not specified, the job assumes a value of 50.	1500
PROCESS_CATALOG_ID	Required Catalog ID of the Journal Engine to spawn.	86 (AMSOfflineJournalPosting)
QUEUE_CLASS	Required Queue Class that specifies the catalog ID needed to create child jobs.	1
STAGGER_TIME	Optional lag time in seconds, between the spawning of each child job. If left blank the job assumes a value of 30.	20
THREAD_COUNT	Optional number of child jobs spawned to journalize posting lines. If not specified the job assumes a value of 1.	10

Major Output

Journal/Ledger Control (JLCTRL) records specify the names of the journals updated by the Journal Engine job(s). Potential output journals (in baseline CGI Advantage Financial) include the following, but a given posting line may only update a subset:

- Accounting (JRNL_ACTG)
- Cost Accounting (JRNL_CA)
- Cash (JRNL_CASH)
- 1099 Reporting (JRNL_1099)
- Internal (JRNL_INT)
- Cross-year (JRNL_BFYNOTFY)
- Fixed Asset Accounting (JRNL_FA)

Job Return Codes

If any of the child jobs was not successful, the greatest Job Return code among the child jobs is returned.

Return Code	Condition
Successful (1)	All of the selected records are processed successfully.
Warning (4)	No eligible records found. This could be because of the following reasons:

Return Code	Condition
	• No Records found on Posting Catalog. • No records found on Posting Catalog with Journal Posting Indicator set to Ready to Post.
Non-Fatal Error (8)	Not Applicable for this job.
Failed (12)	The job fails under the following conditions: • Parameters are invalid. • A spawned Journal Engine fails. • Run time exceptions for unexpected situations.
Terminated (16)	This return code is issued when the job is terminated by the user.
System Failure (20)	This return code is issued when the job is terminated because of database server or network issues.

Sort Criteria

- Transaction Department Code
- Transaction Code
- Transaction ID
- Transaction Version Number

Selection Criteria

The Multi Process Disbursement Journal Posting job first selects all of the transactions of Transaction Type AD that have been finalized on the transaction and have a Journal Posting Indicator of *Not Ready to Post* on the posting catalog and updates the Journal Posting Indicator to *Ready to Post*.

The Multi Process Disb Journal Posting job then selects all of the transactions from the Posting Line Catalog with JRNL_PSTNG_IND as *2 (Ready to Post)*.

Problem Resolution

The job cannot be restarted in this step. If the job fails a new job should be scheduled after correcting the errors that caused the job to fail

The following tables show the possible return codes and recommendations for each processing step.

Step 1: Batch Parameter Validation:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the parameters are	N/A	N/A

Possible Return Codes	Condition	Recommendation	Other Instructions
	validated successfully.		
Warning (4)	N/A	This step does not issue this return code.	N/A
Non-Fatal Error (8)	N/A	This step does not issue this return code.	N/A
Failed (12)	Job failed due to fatal conditions.	In this step, the job can fail under the following two conditions. • Encounters any runtime exceptions and • Parameters are invalid. If the job fails because of the runtime exceptions, investigate the exception reported by the process, resolve the error, and schedule a new job. If the job fails because of the invalid parameters, verify the job log for invalid parameter messages and submit a new job with the correct parameters.	N/A
Terminated (16)	Job is terminated manually by the user.	Reason for the termination needs to be investigated before submitting a new instance of this job.	N/A
System Failure (20)	When the job is terminated because of database server or network issues.	Reason for the termination needs to be investigated before submitting a new instance of this job.	N/A

Step 2: Journal Posting Initiator:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	One or more posting lines were	N/A	N/A

Possible Return Codes	Condition	Recommendation	Other Instructions
	updated.		
Warning (4)	Job ended with a Warning because there are no records selected.	Verify the Transaction Code parameter is correct. If it is, then there were no records to process.	Verify the AD Chain ran as expected if there were supposed to be records to process.
Non-Fatal Error (8)	N/A	This step does not issue this return code.	N/A
Failed (12)	Job failed due to fatal conditions.	The job step encountered a runtime exception. Investigate the exception reported by the process, resolve the error, and schedule a new job.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted. A new job should be scheduled.
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can either be restarted or a new job can be scheduled.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted. A new job should be scheduled.
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the system failure needs to be investigated. The job can either be restarted or a new job can be scheduled.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted. A new job should be scheduled.

Step 3: Journal Posting:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	One or more Journal Engine jobs were spawned and finished successfully.	N/A	N/A
Warning (4)	N/A	This step does not issue this return code.	N/A
Non-Fatal Error (8)	N/A	This step does not issue this return code.	N/A
Failed (12)	Job failed due to fatal conditions.	In this step, the job can fail under the following two conditions. • Encounters any runtime exceptions and • A spawned Journal Engine failed Investigate any runtime exception reported by the process, resolve the error, and restart the job. When a Journal Engine failed, investigate the reason for that failure in the job log of that batch job.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted. A new job should be scheduled to journalize any remaining posting lines. If necessary, a single instance of the Journal Engine can be scheduled.
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can either be restarted or a new job can be scheduled.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted. A new job should be scheduled.
System Failure (20)	When the job is terminated because of database server or	The reason for the system failure needs to be investigated. The job can either be restarted or a new job	If another instance of the job has already been scheduled

Possible Return Codes	Condition	Recommendation	Other Instructions
	network issues.	can be scheduled.	and ran successfully, then this job should not be restarted. A new job should be scheduled.

2.1.38 Offline 1099 Process

The Offline 1099 Process consolidates domestic taxpayer income by 1099-S Form Number, 1099 Reporting Payer, Taxpayer ID Number (TIN), TIN Type, and Form Type and updates this consolidated income information on the 1099 Reported Income (1099R) table.

If the income is for foreign taxpayers, the Offline 1099 Process updates the consolidated income on the 1042-S Reported Income table (1042R) by 1099 Reporting Payer, Taxpayer ID Number and Type, Form Type, 1042-S Recipient Account Number, and IRS Country of Residence.

The 1099 feature in Advantage allows external agencies to interface 1099 data into Advantage Financial. External 1099 data is stored on the 1099 External Reported Income (1099ER) table. Data is loaded to 1099ER by the 1099 External Reported Income batch process (See run sheet for details). Records on 1099ER are also processed by the Offline 1099 Process the same way it processes 1099R records in the various Processing Modes. The only difference is that records on 1099ER are not removed when run in Testing mode.

The Offline 1099 Process also creates text files that may be used by a third party tool to print IRS forms and generate IRS transmittal files (original submission and corrections). Separate files are created for records on 1099R and 1099ER. Records on these tables are not consolidated together within Advantage. It may be consolidated in the third party tool or using the Advantage Forms Table Load Chain batch.

When to Run

This process can be run at any point in Processing Mode of 1099 Report and Testing selected on 1099 Processing Options and Control (1099P) page. To generate 1099 Forms, the process should be run in Processing Mode of Forms. Then, the process can be run in Magnetic Media Mode to process updated or new records and generate data files to submit to the IRS. Finally, the Offline 1099 Process can be run in Corrections Mode to process any corrections made to already transmitted form data.

Description

The major input to this process is the 1099 Journal, which records all payments to vendors and federal withholding throughout the calendar year. Based on the 1099 reportability of a Taxpayer Identification Number (TIN) and the 1099 Income Return Type set on the Object, Sub-Object, Balance Sheet Account, and Sub Balance Sheet Account records, the Offline 1099 process consolidates reportable vendor income and updates this information on the Reported Income (1099R) table and the 1042-S Reported Income (1042R) table regardless of the threshold amount established on the Type of Income (TINC) table. The 1099ER table will not be read when job is run in 1099 Report Mode.

For 1099, the Valid Vendor Organization and 1099 Reporting Class (VORGCL) table is read in conjunction with the Type of Income (TINC) table to determine the Reporting Classification of the associated Vendor/Taxpayer and if the combination is indeed reportable. For the Organization Type, 1099 Classification, and TIN Type defined on VORGCL, the 1099 Reporting Classification will determine if the combination is reportable. If the 1099 Reporting Classification corresponds to one of the classifications on TINC, then a lookup will be performed on TINC to determine if the classification is reportable (checked). If the 1099 Reporting Classification on VORGCL is not one of the classifications on TINC then the vendor is not reportable. There is one exception: If the Object, Sub-Object, Balance Sheet Account, or Sub Balance Sheet Account is marked as *Always Reportable* and 1099 Reporting Classification on VORGCL is set as *Corporation* and 1099 Indicator is set to *Yes* on VCUST then the income will be reportable regardless of the setting on TINC. Refer to the Selection Criteria section for detailed logic for 1099 and 1042-S.

Backup withholding (posting pair E) and contract withholding (posting pair K) are consolidated to specific Income Return Type based on IRS rules instead of the Income Return Type set on the Object, Sub-Object, Balance Sheet Account, or Sub Balance Sheet Account records. Any reportable income paid to a vendor is updated to the Reported Income (1099R, 1099ER, 1042R) tables, regardless of the threshold amount established on the Type of Income tables. The threshold will be evaluated when the output files are generated.

If the 1099-S Reporting field is set to *TIN/TIN Type* on 1099, the 1099-S records will be added to the 1099R table by the Offline 1099 Process. If it is set to *Property*, no 1099-S records will be added to the 1099R table, they will only appear on the 1099-S Transactions by Vendor Report.

Any records on the 1099R, 1042R tables and 1099ER table (if Read 1099ER is checked), that have federal withholding or have amounts that exceed the threshold amounts specified on the Type of Income (TINC) table will be exported from Advantage as export text files which can be imported into a third-party tool to print the forms and transmit data files to the IRS. When the process is run in Forms or Magnetic Media mode, all records on the 1099R, 1042R, and the 1099ER table (if the Read 1099ER is checked) that has federal withholding or amounts that exceed the threshold will be exported, along with data files for Payer and Payee records. When the process is run in Magnetic Media mode, after either Forms or Magnetic Media modes, only new records that exceed the threshold or updated records will be exported. In Corrections Mode, records that have the Processing Indicator of *New or Updated* will be exported.

If there are special characters, the Offline 1099 Process job still ends successfully, but the job log shows that the record with the special character was skipped, and that record has not been corrected / updated. The job moves onto the next record to be processed. Regardless of Processing Mode, export files will only be created if there are corresponding records for that Form Type on the 1099R, 1042R, or 1099ER tables.

Because all new and updated records are exported during the Corrected Processing Mode (bypassing threshold logic), if the dollar amount of the 1099 Reported Income (1099R) table needs to be corrected on a record that had an amount that does not meet the threshold established for the Type of Income, it will need to be done in one of two ways:

1. Change the value after you have run the process in Magnetic Media Mode so that it will be picked up and a corrected form is generated when the process is run in Corrected Mode.
2. Manually change the amount in the 3rd Party Forms generation tool.

Depending on the Processing Mode and settings on the 1099P table, three different reports may be produced by this process: The Forms Generation Report, the 1099-S Transactions by Vendor Report, and the Non TIN Miscellaneous Vendors Report. Note that the Non TIN Miscellaneous Vendors Report reports transactions only for the form types 1099-MISC, 1099-INT, 1099-G, 1099-S, and 1099-NEC. After each run, some fields are reset for the calendar year. The Last Processing Mode is set to the current Processing Mode.

The 1099 Journal as Input Flag (EXTC_INPT_FL), 1099-S Transactions by Vendor Report flag (TRNS_VEND_RPT_FL) and Non TIN Miscellaneous Vendors Report flag (MVEND_RPT_FL) are turned to False (if they were True and Processing Mode is not Testing mode and not 1099 Report Mode).

The Correction Processing flag (CORR_PROC_FL) is set to True after running in Magnetic Media and Corrected mode. The Offline Batch job cannot be run in Magnetic Media Mode with the Corrections Processing flag set to True as the job will fail.

Type 1 and Type 2 Correction

Any correction to income reported on 1099R can be made using direct 1099R table update or using M1099 transaction. Any correction to existing reported income will be either Type 1 or Type 2 correction. Type 1 and Type 2 corrections are distinguished as follows:

Type 1 – It includes changes in address fields, incorrect boxes (income should have been reported in box 2 instead of 3), incorrect money amount, incorrect payment amount codes, and if a form was submitted in error.

Type 2 – It includes change in TIN, TIN Type, Form Type, or Name. If Name and Address fields both are modified in the same submission, then it will also be considered as Type 2 correction. For example, when Name was modified, it will be considered as Type 2 correction. If any correction of the address fields, such as City, is corrected then it will be considered as a Type 1 correction unless they both were corrected in one submission to the IRS. This is true if both changes were made by the same user/time or different users/time. The net changes are determined at the time Offline 1099 Process was executed.

If a correction is made that is of Type 2 and then a subsequent change is made that is of Type 1, the net changes will be considered as a Type 2 correction. If changes are made to a record that is neither Type 1 nor Type 2 (i.e., Comments), then no corrected form should be produced. Also, changes to fields that are not classified as Type 1 or Type 2 should not impact the correction type. For example, an amount change is made and then a comment is added to the same record, the record will be classified as Type 1 correction. The addition of comments should not change the correction type. Based on whether it is Type 1 or Type 2 correction, the Correction Return Indicator and Corrected Transaction Indicator will be determined and updated to the reported income tables. The Correction Return indicator is also updated in the Export txt files after running the correction job.

Parameters

Job	Parameter	Description	Default Value
Offline1099	Client Name for Report (CLIENT_NM)	Optional field. Entry of a value in this field specifies the name that will appear on report.	No Default
	Calendar Year for which records is to be processed (CY)	Required Field. Enter the Calendar Year for which Vendor Income should be selected.	
	Compare Threshold (COMPARE_THRESHOLD)	Required Field. Entry to this field will determine threshold flag. Valid values are : 1=Yes 2=No	2
	Read 1099ER table (READ_1099ER)	Required Field. This field will determine whether	2

Job	Parameter	Description	Default Value
		1099ER table needs to be read or not. Valid values: 1 = Yes 2 = No	
	Report Sorting Criteria. (SORT_CRIT_1)	Required Field. This field determines the report sort. Valid Sorting Criteria: 1. By TIN 2. By Vendor Name 3. By Zip Code	1
	Exception Report Sorting Criteria. (SORT_CRIT_2)	Required Field. This field determines the sort of the Exception Report. Valid Sorting Criteria: 1. By 1042-S Recipient Account Number 2. By Vendor Name 3. By Zip Code	1
	Exclude Pcard Payment (EXCL_PCARD_PYMT)	Required Field. This field determines whether or not to report PCard Payments on 1099 Forms. Exclude PCard Payment (Y=Yes,	Y

Job	Parameter	Description	Default Value
		N=No)	
	View VSS Payment (VIEW_VSS_PYMT)	Required Field. This field determines whether 1099/1042-S vendor transactions are visible or not on the VSS Tax Information page. View VSS Payment (Y=Yes, N=No)	N

For 1042-S records pertaining to calendar year 2014 and greater, the system will verify that the Withholding Agent's Form 1042-S Chapter 3 Status Code box is populated on the 1099RP table. If it is not populated, the batch job will fail and an error message will be written for each Reporting Payer record that does not have this code. Chapter 3 Status Code is required while Chapter 4 Status Code is optional on 1099RP.

Output of the process ran in various Processing Modes

Processing Mode	Output	Comments
Reports	Selected Reports	1099 Journal as Input flag must be set to True.
Testing	• Records are inserted on 1099R and/or 1042-S tables • Magnetic Media Audit Report • 1042 Exception Report • 1099 Miscellaneous Vendor Report (if selected on 1099P) • 1099-S Transactions by Vendor Report (if selected on 1099P) • 1099 Exception Report	1099 Journal as Input flag must be set to True. The Magnetic Media Audit report displays all reported income records that have federal withholding or records that have one or more amount fields that exceed the threshold established on the Type of Income table. **Note: Non TIN Miscellaneous Vendors Report reports transactions only for the form types 1099-MISC, 1099-INT, 1099-G, 1099-S**

Processing Mode	Output	Comments
		and 1099-NEC.

Forms	• Records are inserted and/or updated on 1099R, 1099ER, and/or 1042-S tables
	• Forms Generation Report (if selected on 1099P)
	• 1099 External Reported Income Forms Generation Report
	• 1042 Exception Report
	• 1099 Exception Report
	• Export Text files (found in ReportOutput folder):
	  • Payer(date).txt
	  • Payee(date).txt
	  • Payee1042(date).txt
	  • FormMISC(date).txt
	  • FormNEC(date).txt
	  • FormINT(date).txt
	  • FormG(date).txt
	  • FormS(date).txt
	  • Form1042(date).txt
	  • ExtPayer(date).txt
	  • ExtPayee(date).txt
	  • ExtFormMISC(date).txt
	  • ExtFormNEC(date).txt
	  • ExtFormINT(date).txt
	  • ExtFormG(date).txt
	  • ExtFormS(date).txt
	  • ExtFormA(date).txt
	  • ExtFormC(date).txt
	  • ExtFormR(date).txt
		The Forms Generation Report displays all records that have federal withholding or records that have one or more amount fields that exceed threshold by Form Type.
		
		An export file will only be created if there are corresponding records for that Form Type that have federal withholding or records that have one or more amount fields that exceed threshold on the Reported Income table.

Magnetic Media	• Records are inserted and/or updated on 1099R, 1099ER, and/or 1042-S tables • Magnetic Media Audit Report • Magnetic Media Audit Report for	The Magnetic Media Audit report displays all reported income records that have federal withholding or records that have one or more amount fields that

Processing Mode	Output	Comments

	External Reported Income
	• 1042 Exception Report
	• 1099 Exception Report
	• Export Text files (found in ExportImport folder used by Forms Table Load chain process):
	  • Payer.txt
	  • Payee.txt
	  • Payee1042.txt
	  • FormMISC.txt
	  • FormNEC.txt
	  • FormINT.txt
	  • FormG.txt
	  • FormS.txt
	  • Form 1042.txt
	  • ExtPayer.txt
	  • ExtPayee.txt
	  • ExtFormMISC.txt
	  • ExtFormNEC.txt
	  • ExtFormINT.txt
	  • ExtFormG.txt
	  • ExtFormS.txt
	  • ExtFormA.txt
	  • ExtFormC.txt
	  • ExtFormR.txt
	• Export Text files (found in ReportOuput folder):
	  • Payer(date).txt
	  • Payee(date).txt
	  • Payee1042(date).txt
	  • FormMISC(date).txt
	  • FormNEC(date).txt
	  • FormINT(date).txt
	  • FormG(date).txt
		exceed the threshold established on the Type of Income table.
		An export file will only be created if there are corresponding records for that Form Type that have federal withholding or records that have one or more amount fields that exceed threshold on the Reported Income table.

Processing Mode	Output	Comments
	• FormS(date).txt • Form1042(date).txt • ExtPayer(date).txt • ExtPayee(date).txt • ExtFormMISC(date).txt • ExtFormNEC(date).txt • ExtFormINT(date).txt • ExtFormG(date).txt • ExtFormS(date).txt • ExtFormA(date).txt • ExtFormC(date).txt • ExtFormR(date).txt	

Corrections	• Records updated on 1099R, 1099ER, and/or 1042-S tables
	• Magnetic Media Audit Report
	• Magnetic Media Audit Report for External Reported Income
	• 1042 Exception Report
	• 1099 Exception Report
	• For Updated (Corrected) records generates:
	  • Export Text files (found in ExportImport folder used by Forms Table Load chain process):
	  • Payee.txt
	  • Payee1042.txt
	  • FormMISC.txt
	  • FormNEC.txt
	  • FormINT.txt
	  • FormG.txt
	  • FormS.txt
	  • Form1042.txt
	  • ExtPayee.txt
	  • ExtFormMISC.txt
		The Magnetic Media Audit report displays all reported income records that have federal withholding or exceed the threshold established on the Type of Income table.
		Only the new or corrected records are written to the export .txt files.
		An export file will be created if there are new records for that form type that have federal withholding or exceed threshold on the Reported Income table. Also, any records that were previously reported to the IRS (IRS Reported = Yes) will be written to the file.

Processing Mode	Output	Comments
	• ExtFormNEC.txt • ExtFormINT.txt • ExtFormG.txt • ExtFormS.txt • ExtFormA.txt • ExtFormC.txt • ExtFormR.txt • Export Text files (found in ReportOuput folder): • Payer(date).txt • Payee(date).txt • Payee1042(date).txt • FormMISC(date).txt • FormNEC(date).txt • FormINT(date).txt • FormG(date).txt • FormS(date).txt • Form1042(date).txt • ExtPayer(date).txt • ExtPayee(date).txt • ExtFormMISC(date).txt • ExtFormNEC(date).txt • ExtFormINT(date).txt • ExtFormG(date).txt • ExtFormS(date).txt • ExtFormA(date).txt • ExtFormC(date).txt • ExtFormR(date).txt	

Major Input

- 1099 Journal (J1099)
- 1099 Reporting Information (1099I)
- 1099 Reporting Information By Department (1099ID)
- 1042-S Reporting Information (1042I)

- 1099 Type of Income (TINC)
- 1042-S Type of Income (1042T)
- 1099 Reporting Class (1099CL)
- 1099 Reported Income (1099R)
- 1099 External Reported Income (1099ER)
- 1042-S Reported Income (1042R)
- 1099 Reporting Payers Information (1099RP)
- Valid Vendor Organization and 1099 Reporting Class (VORGCL)
- 1042-S Withholding Allowance (1042W)
- Object (OBJ)
- Sub Object (SOBJ)
- Balance Sheet Account (BSA)
- Sub Balance Sheet Account (SBSA)
- Vendor/Customer (VCUST)
- 1099 Processing Options and Control (1099P)
- 1099 Date and Transaction Parameters (1099D)
- Country (CTRY)
- Income Code (INCM)
- Exemption Code (EXMP)
- 1042-S Ch. 3 Recipient Code (RECP)
- 1042-S Chapter 4 Status Code (CHP4)
- 1042-S Limitation of Benefits (LOB)

Output

- 1099 Reported Income (1099R)
- 1099 External Reported Income (1099ER)
- 1042-S Reported Income (1042R)
- Payee data file
- Payer data file
- 1099 and 1042 S Form files

Reports:

- Forms Generation Report
- Forms Generation Report for External Reported Income
- Magnetic Media Report
- Magnetic Media Report for External Reported Income

- Non-TIN Miscellaneous Vendor Report (This report contains transactions only for the form types 1099-MISC, 1099-INT, 1099G, 1099S and 1099-NEC.)
- 1099 Transactions by Vendor Report
- 1099-S Transactions by Vendor Report
- 1099 External Reported Income Transactions by Vendor Report
- 1099 Exception Report
- 1042S Exception Report

Sort Criteria

None

Selection Criteria

- If the 1099 Journal as Input Flag (EXTC_INPT_FL) is checked on the 1099 Processing Options and Controls table for Testing, Forms or Magnetic Media processing modes then source of input for the process is Journal 1099 (JRNL_1099) else it is 1099 Reported Income (1099R) (AP_RPT_1099_INC) and 1099 External Reported Income (1099ER) (AP_RPT_1099_XINC). The 1099 Reporting Payer is selected based on the SOPT setting. If the Multiple 1099 Reporting Payers flag on SOPT is set to *No*, then select the 1099 Reporting Payer from SOPT for all records. If the Multiple 1099 Reporting Payers flag is set to *Yes* the process shall read the DEPTFY table to obtain the 1099 Reporting Payer. The Department and FY from the transaction's accounting line will be used to perform a lookup on the DEPTFY table to retrieve the Payer for that department.
- The selection criteria are specified dates between or equal to the date range defined for the Transaction Code and Calendar Year on the 1099 Date and Transaction Parameters Setup (1099D) table or if no record for the calendar year exists on the 1099D table and the record date is between or equal to the Calendar Year records (CY entered as Batch parameter).

The batch process selects records from the 1099 Journal (if it is the first time the 1099 process is executed for the calendar year) or the 1099 Reported Income (1099R) or 1042-S Reported Income (1042R) tables (if it is not the first time the 1099 process is executed for the calendar year and the 1099 Journal as Input is not checked) based on all of the following criteria:

1. Records that contain a posting pair type of A (posting to Cash) or a posting pair type of E (for 1099 Withholding or 1042-S Withholding payable).

   Once these records are selected, the 1099 process must then determine if the records are reportable for 1099 or 1042-S reporting.

2. If the record contains a Vendor/Customer code with the 1042-S Recipient Account Number populated on the VCUST table, the record will be considered for 1042-S reporting and forms creation.

3. If the record is for a PCard Payment (i.e. the PCard ID is populated in the J1099 table) and the batch parameter **Exclude PCard Payment** is set to Y (Yes), then the batch job will skip these records. Otherwise, if the value of the batch parameter **Exclude PCard Payment** is set to N (No), then records with PCard ID populated on the 1099 Journal will be considered for 1099 Reporting.

4. Otherwise, all other records on the 1099 Journal will be considered for 1099 Reporting and forms creation.

1099 Reporting

1. If the Vendor is not Miscellaneous, take the VCUST Code and retrieve the TIN/TIN Type. If there is no TIN/TIN Type, skip the record.
2. If the vendor is miscellaneous and the *Use New TIN and TIN Type from 1099I for Miscellaneous Vendors on 1099 Reported Income* Updates parameter on the Application Parameters (APPCTRL) table is set to *True*, the process uses the TIN and TIN Type from 1099I and the TIN Change section to update records on 1099R, if populated. If TIN and TIN Type in the TIN Change section are not entered, it uses the values from J1099. When this parameter is set to *False*, the process obtains the TIN and TIN Type from J1099.
3. The system then determines if the selected record is reportable for 1099. If the Manage TIN by Department parameter on the Application Parameters (APPCTRL) table is set to *True*, it is defined on the 1099ID record for TIN/TIN Type and Transaction Department Code combination, or else it is defined on the 1099I record for the TIN/TIN Type when this parameter is set to *False*.
4. Once a record is determined to be eligible for 1099 Reporting, the system determines if the Sub-Object, Object, Sub-Balance Sheet Account, or Balance Sheet Account on the transaction is associated with a Form Type and Income Type on its chart of account (COA) record. The next four steps will be completed for each COA element.
5. If the COA element is found to be reportable, the system then retrieves the Organization Type, 1099 Classification and TIN Type from the VCUST record for the Vendor. If the Organization Type and 1099 Classification are not populated, the vendor data is incomplete and cannot be reported upon so the record is skipped.
6. Take the 1099 Reporting Classification from VORGCL and look up the value on the 1099 Reporting Classification (1099CL) table. Determine what the reporting classification value it equates to (e.g. Individual, Sole Proprietor, Corporation, etc.).
7. Take the Reporting Classification from 1099CL and the Income Type from the COA element and go to the 1099 Type of Income (TINC) table to determine if the vendor is eligible for reporting for that type of income. For the Form Type and Income Box, if classification is yes, the record is considered reportable. If the Reporting Classification is not equal to any classifications on TINC, the income cannot be reportable.

    i. The TINC Threshold comparison is bypassed when the Batch Parameter for ‘Compare Thresholds’ is set to ‘2’ (Note: Threshold is not taken into account in initial 1099 processing. All records that include reportable income are selected by the process and updated to the 1099 Reported Income table. Only records on 1099R where backup withholding was taken or at least one box meets or exceed the IRS specified threshold amount will be included in the 1099 data files that are created by the batch process), or

    j. When 1099 Reporting Classification is Corporation on VORGCL and ‘Always 1099 Reportable’ checkbox on the COA table, for the COA associated with the J1099 record, is set to yes. If this is yes, the record is considered reportable.

8. Next, the system reads the 1099D table to determine if there are any impacts on record selection.

    - The logic compares the transaction Record Date to the 1099D “From Record Date” and “To Record Date” range. The logic uses any dates entered on 1099D as additional selection logic. If those fields on 1099D are blank, or a 1099D record is not found for the transaction, then 1099D does not impact that record’s selection.
    - The logic also compares the Acceptance Date of the transaction to the Acceptance Date Range. As with other 1099D fields, if the fields are blank then they do not impact 1099 Selection. If the Acceptance Dates are populated, then the Transaction’s Acceptance Date must occur within the From/To range.

- The logic compares the Check/EFT Issue Date to the Check/EFT Issue Date Range. As with other 1099D fields, if the fields are blank then they do not impact 1099 selection. If the Check/EFT Issue Dates are populated, then the Issue Date must occur within the From/To Range.

  For a Transaction Code that is associated with an "AD" or "MD" Transaction Type, the Check/EFT Issue Date is obtained from the AD Transaction Header or MD Transaction Header table.

  For a Transaction Code that is associated with the "DC" Transaction Type, the Reference Transaction Number on the 1099 Journal is used to obtain the Check/EFT Issue Date from the AD Transaction Header or MD Transaction Header table. The Check/EFT transaction will always be memo referenced on the DC transaction.

9. Next, the 1099 Reporting Payer is selected for each record. If the 'Multiple 1099 Reporting Payers' flag on SOPT is set to *No*, then select the 1099 Reporting Payer from SOPT for all records. If the 'Multiple 1099 Reporting Payers' flag is set to *Yes*, the process reads the DEPTFY table to obtain the 1099 Reporting Payer. The Department and FY from the transaction's accounting line are used to perform a lookup on the DEPTFY table to retrieve the 1099 Reporting Payer for that department. The obtained 1099 Reporting Payer(s) are used when reading the 1099RP table as well as when reading and/or updating the 1099R and 1042R tables. The 1099RP table is read to infer the TIN, TIN Type, Name and Address of the payer(s) as well as Box-3 of the 1099-S form.
10. Next, 1099-S reportability is determined. If the '1099-S Reporting' indicator on 1099P is set to *Property* then the process selects all 1099-S transactions to be written to the 1099-S Transactions by Vendor report. If the '1099-S Reporting' indicator is set to *TIN/TIN Type* then all 1099-S transactions are selected. The '1099-S Form Number' is set to '0' for all 1099 forms.
11. Next, verification for duplicate records is determined on 1099R and 1099ER tables. Records with the same Calendar Year, TIN, Form Type and 1099 Reporting Payer but different TIN Type are identified when the 1099 Journal is used as an input. The process writes the 1099 reportable records that are duplicates to the 1099 Exception report, along with a description of the message "Duplicate Record exists on 1099R table." Research is required to verify if the records on the report are incorrect and manually update them.
12. Next, the system determines if the selected record is for a PCard Payment or not. If the record is for a PCard Payment (i.e. the PCard ID is populated in the J1099 table) and the batch parameter **Exclude PCard Payment** is set to Y (Yes), then the batch job will skip these records and the 1099R table is not updated for such records. Otherwise, if the value of batch parameter **Exclude PCard Payment** is set to N (No), then the 1099R table is updated and reported for both types of records (those having a PCard ID and those not having a PCard ID) from the J1099 table.

    All consolidated records, including duplicates as defined in step above, are updated to the 1099 Reported Income table. 1099 Journal records that are deemed eligible for 1099 reporting are then consolidated by Calendar Year, 1099-S Form Number, 1099 Reporting Payer, TIN, TIN Type, and Form Type. The results of this consolidation are then updated on the 1099R table or to the 1099-S Transactions by Vendor Report.

1042-S Reporting

If records are considered eligible for 1042-S Reporting, the following record selection process is followed:

1. The system determines if the specific transaction is reportable for 1042-S processing.

- The system takes the Vendor/Customer code on the transaction and retrieves the 1042-S Recipient Account Number on the VCUST table.
- The system then performs a look up to the 1042I table by the 1042-S Recipient Account Number to verify if the 1042-S Reportable flag is True.

2. Next, the system determines if the Record Date of the transaction is between or equal to the From and To Date's for the 1099D table designated for the Transaction Code and Calendar Year. In addition, there can be date ranges for From and To Acceptance Dates and From and To Check/EFT Issue Dates that can be used in combination with the From and To Record Dates. If a record for the Transaction Code does not exist on the 1099D table for the designated calendar year, then the process will assume the From and To Date's set to January 1 and December 31 of the actual calendar year date range.
3. Once a record has been determined to be eligible for 1042-S reporting, the system determines if **any** of the COA fields, Sub-Object, Object, Sub-Balance Sheet Account, or Balance Sheet Account on the transaction is associated with a 1042-S Income Code/1042-S Income Type.
4. Then, the system takes the 1042-S Income Code/1042-S Income Type associated with the COA record AND the 1042-S Ch. 3 Recipient Code or the 1042-S Ch. 4 Status Code associated with the Vendor/Customer Code and determines if a matching record is defined on the 1042-S Type of Income table.
5. 1099 Journal records deemed eligible for 1042-S reporting are then consolidated by the following fields:
   - 1099 Reporting Payer - (If the 'Multiple 1099 Reporting Payers' flag on SOPT is set to *No*, then select the 1099 Reporting Payer from SOPT for all records. If the 'Multiple 1099 Reporting Payers' flag is set to *Yes* the process shall read the DEPTFY table to obtain the 1099 Reporting Payer.) The Department and FY from the transaction's accounting line will be used to perform a lookup on the DEPTFY table to retrieve the 1099 Reportable Payer for that department.
   - 1042-S Recipient Account Number (inferred from the VCUST table)
   - IRS Country of Residence (inferred from the 1099 Journal)
   - 1042-S Tax Rate (inferred from the 1099 Journal)
   - 1042-S Ch. 3 Recipient Code or 1042-S Ch. 4 Status Code (inferred from the VCUST table)
   - 1042-S Income Code (inferred from applicable COA code)
   - 1042-S Ch. 3 Exemption Code or 1042-S Ch. 4 Status Code (inferred from the 1042-S Income Code/Tax Rate table).
   - 1042-S Ch.3 Exemption Code (inferred from ICTX for Ch. 3 Recipients and it is inferred from ICTX4 for Ch. 4 Recipients). Also, Ch. 3 Exemption Code will be blank if Ch. 3 Tax Rate is equal to the 1042-S Backup Withholding Tax Rate on 1099P.
   - LOB Code (inferred from 1042I)
6. This process performs the following validations when the 1099 Journal is used as an input and writes the 1042-S Reportable records that do not meet any of those validations to the 1042-S Exception Report, along with a description of the error. In addition, all consolidated records, even those that cause exceptions, are updated to the new 1042-S Reported Income table. This report is necessary so a site can validate if any changes were made to the setup during the reporting period.
   - Exception 1: The 1042-S Income Code and IRS Country of Residence on the consolidated transactions have a matching combination on the 1042-S Income Code/Tax

Rate table and the rate that is specified is less than the rate specified on the 1042-S Withholding Rate on the 1099P table. If this exception is found, the transactions are written to the exception report with the exception "Tax Rate Less than 1099P."

- Exception 2: If the Exemption Code on the consolidated transactions is "04" (exempt under tax treaty) **OR** the Tax Rate is less than the 1042-S Withholding Rate specified on the 1099P table (see Exception 1), the system verifies that the IRS Country of Residence has the Treaty indicator set to *Yes* on the IRS Country of Residence table. If the treaty indicator is not selected in this situation, the transactions are written to the exception report with the exception "Treaty Indicator not Selected."
- Exception 3: If the Net Amount on Box 4 is negative the transactions are written to the exception report with the exception "Net Amount is Negative." (This scenario would only be possible when the Withholding Allowances is greater than the Gross Income.)
- Exception 4: If the Income Code consolidated transactions exists on the 1042-S Income Code/Recipient Code table, then the Income Code/Recipient Code combination of the transactions must be defined on that table. If this record does not exist, the transactions are written to the exception report with the exception "Invalid Income Code/Recipient Code Combination."

Troubleshooting

No database restore is required. Correct the problem and rerun the job executing the program. No restoration of datasets or files from backups is required for this program.

2.1.39 Payment Hold Archive Process

Chain or Job Name	Payment Hold Archive Process
Recommended Frequency	On Demand
Single Instance Required	No
Can be restarted?	Yes
Reports generated	None

Overview

This is a chain job and is run in 2 steps.

The first batch job (**Selection**) validates the parameters passed into the batch job. It then selects the applicable records to archive from the Payment Hold Maintenance, Payment Hold Activity, and Print Payment Hold Notice tables as well as Payment Hold Maintenance transactions and create a parameter file for the next batch job AMSFacilitator.

The **AMSFacilitator** (Table Archive) job then reads the parameter file and archives the records form the given tables based on an input parameter (Report Only). If this parameter is set to Yes, then the AMSFacilitator just creates an xml file for the records to be archived. If this parameter is set to No then the batch job creates the XML file and deletes the records.

This chain implements the Data Warehouse Archived Record Queue process, which is used to retain the table names and key values for records that are archived and deleted. The infoAdvantage reporting system uses this information to distinguish between records that have been archived (as valid historical data) before deletion and records that have simply been deleted from the system (as no longer needed). The process is enabled for most archiving processes when the value of Application Parameter Enable Data Warehouse Archived Record Queue (ENABLE_DW_ARCH_QUEUE) is *True*. Please see the *CGI Advantage System Administration Guide* for more information regarding this process and the Application Parameter.

Major Input

Data from the following tables:

- Print Payment Hold Maintenance Table (R_PYMT_HLD_MNTN)
- Print Payment Hold Notice Table (R_PRNT_PYMT_HLD_NTC)
- Payment Hold Activity Table (R_PYMT_HLD_ACT)

Major Output

Following are the output of the chain job:

- Run Number_Sequence Number_Label_YYYYMMDD.txt
- Run Number_Sequence Number_Label_YYYYMMDD.sql

Chain Return Code

The following table indicates the potential return codes with which the chain will end:

Return Code	Condition
Successful (1)	All the jobs ends successfully
Warning (4)	One of the job in the chain ends with a return code of "Warning"
Non-Fatal Error (8)	One of the job in the chain ends with a return code of "Non-Fatal Error"
Failed (12)	One of the job in the chain ends with a return code of "Failed"
Terminated (16)	One of the job in the chain ends with a return code of "Terminated"
System Failure (20)	One of the job in the chain ends with a return code of "System Failure"

Problem Resolution

Please refer to the individual job "Problem Resolution" section for more details

Payment Hold Archive Process: Selection Job

Chain or Job Name	Selection
Recommended Frequency	On Demand
Single Instance Required	No
Can be restarted?	Yes
Reports generated	None

Overview

The Payment Hold Archive batch process archives applicable records from the Payment Hold Maintenance, Payment Hold Activity, and Print Payment Hold Notice tables as well as Payment Hold Maintenance transactions. The Advantage 3 baseline System Maintenance Utility "Table Archive" and "Transaction Archive" commands are utilized to archive selected records from the target tables.

Process Steps	Messages
1. Validate the Parameters	• Validating Batch Parameters • Parameters are valid or invalid depending on the Validation. If the parameter is invalid, the invalid value will be displayed in the log. • Batch Parameter validation completed

Process Steps	Messages
2. Archiving the Payment Hold Maintenance Table	• In this step, the process will identify and archive eligible records from the Payment Hold Maintenance table. • The process shall identify and archive all the records on the Payment Hold Maintenance table where the Hold Level is “Vendor”, having no corresponding Vendor records on the VCUST table. Vendor of the Payment Hold Maintenance table shall be used to do look up on the VCUST table. The process shall not archive the Payment Hold Maintenance Table record if there is a matching record exists in the VCUST table. • The process shall identify and archive all Master Agreement records on the Payment Hold Maintenance table where the Hold Level is “Award”, having no corresponding Master Agreement transactions on the MA_DOC_HDR table. Transaction Code, Transaction Department, Transaction ID of the Payment Hold Maintenance table shall be used to do look up on the MA_DOC_HDR table. The process shall not archive the Payment Hold Maintenance table record if there is a matching record exists in the MA_DOC_HDR table. • The process shall identify and archive all Encumbrance records on the Payment Hold Maintenance table where the Hold Level is “Award”, having no corresponding Encumbrance transactions on the PO_DOC_HDR table (for commodity-based) or ABS_DOC_HDR table (for accounting-based). Transaction Code, Transaction Department, Transaction ID of the Payment Hold Maintenance table shall be used to do look up on the PO_DOC_HDR table. The process shall not archive the Payment Hold Maintenance table record if there is a matching record exists in the PO_DOC_HDR table. • The process shall identify and archive all Payment Request records on the Payment Hold Maintenance table where the Hold Level is “Payment Request”, having no corresponding Payment Request transactions on the PR_DOC_HDR table (for commodity-based) or ABS_DOC_HDR table (for accounting-based). Transaction Code, Transaction Department, Transaction ID of the Payment Hold Maintenance table shall be used to do look up on the PR_DOC_HDR or ABS_DOC_HDR tables. The process shall not archive the Payment Hold Maintenance table record if there is a matching record exists in the PR_DOC_HDR or ABS_DOC_HDR tables.
3. Archiving the Payment Hold Maintenance Transactions	• The process shall identify and archive all Payment Hold Maintenance transactions with no corresponding records on the Payment Hold Maintenance table. Auto Generated Number of the Payment Hold Maintenance transaction shall be used to do look up on the Payment Hold Maintenance table. The process shall not archive the Payment Hold Maintenance Transactions if there is a matching record exists in the Payment Hold Maintenance table.

Process Steps	Messages
4. Archiving the Payment Hold Activity table	• This process shall identify and archive all records on the Payment Hold Activity table with no corresponding Payment Request transactions on the PR_DOC_HDR table (for commodity-based) or ABS_DOC_HDR table (for accounting-based). Payment Request Transaction Code, Payment Request Transaction Department, Payment Request Transaction ID of the Payment Hold Activity table shall be used to do look up on the PR_DOC_HDR or ABS_DOC_HDR tables. The process shall not archive the Payment Hold Activity Table record if there is a matching record exists in the PR_DOC_HDR or ABS_DOC_HDR tables.
5. Archiving the Payment Hold Notice Table	• The process shall identify and archive all records on the Print Payment Hold Notice table with no corresponding records on the Payment Hold Activity table. Reference Printed Payment Notice Number of the Payment Hold Activity table shall be used to do look up with the Reference Printed Payment Notice Number on the Payment Hold Notice table. The process shall not archive the Payment Hold Notice Table record if there is a matching record exists in the Payment Hold Activity table.
6. Create parameter file.	• This method create new System Maintenance Utility parameter file. A new parameter file is created when "Archive Count" for a parameter file equals or exceeds the user specified Tolerance as a new BS Agent record is identified as eligible to be selected for archive (meaning a new set of logically related records are about to be processed). Before creating a new parameter file all logically related records for the current Agnt_ID being processed will be written to the old parameter file. Therefore the new parameter file will begin with the set of records for a new Agnt_ID.

Restartability

Job can be restarted.

Major Input

Data from the following tables:

- Print Payment Hold Maintenance Table (R_PYMT_HLD_MNTN)
- Print Payment Hold Notice Table (R_PRNT_PYMT_HLD_NTC)
- Payment Hold Activity Table (R_PYMT_HLD_ACT)

Batch Parameters

Parameter	Description	Default Value
File Location FILE_LOCATION	Required editable Field. This parameter defines the location to which the system generated parameter and SQL files will be written.	
Tolerance TOLERANCE	This parameter defines a benchmark for the approximate number of records to be written to each XML archive file.	1000
Report Only REPORT_ONLY	Required Field. This parameter defines the "Report_Only" parameter to define if records are to be purged in addition to reporting on the selected records.	N
Suppress Reports SUPPRESS_RPTS	Required Field. This parameter defines the "SUPPRESS_REPORTS" parameter in the parameter file. When this parameter is defined as *Y* a related SMU report is not generated for each processed parameter file. If set to *N,* one SMU report is generated for each parameter file.	N
Progression counter size PROG_CTR_SZ	Field used to indicate the progression with respect to the number of records that have been archived.	1000
Update Facilitator Status UPDATE_STATUS	Update Facilitator Status. If Report Only is *Y,* Update Status must be *N.*	Y

Major Output

Following are the output of the batch job:

- Run Number_Sequence Number_Label_YYYYMMDD.txt
- Run Number_Sequence Number_Label_YYYYMMDD.sql
- ARCH_DW_QUEUE_REC – The Data Warehouse Archive Record table keeps track of the values for all primary keys of each record being archived and deleted. The table name and attribute names are stored in ARCH_DW_QUEUE_TBL and linked to this record by unique ID.
- ARCH_DW_QUEUE_TBL – The Data Warehouse Archive table keeps track of the table name and primary key attribute names of each table being archived. If a record already exists for the table being archived, it will reuse the record. If it does not yet exist, it will be created.

Job Return Code

Return Code	Condition
Successful (1)	All the selected notice letters are printed successfully.
Warning (4)	No eligible records found. • If no eligible Payment Request records found on the Payment Hold Maintenance table for archiving, the job should issue the following message in the log “No eligible records found on the Payment Hold Maintenance table for archiving”. • If no eligible Payment Hold Maintenance Transactions found for archiving, the job should issue the following message in the log “No eligible Payment Hold Maintenance Transactions found for archiving”. • If no eligible records found on the Payment Hold Activity table for archiving, the job should issue the following message in the log “No eligible records found on the Payment Hold Activity table for archiving” • If no eligible records found on the Payment Hold Notice table for archiving, the job should issue the following message in the log “No eligible records found on the Payment Hold Notice table for archiving”
Non-Fatal Error (8)	We are not setting this return code.
Failed (12)	The job fails under the following conditions: • If validations of the parameters fail. • Run time exceptions for unexpected situations. • Restart failed because another instance of the process has already been run successfully
Terminated (16)	This return code is issued when the job is terminated by the user.
System Failure (20)	This return code is issued when the job is terminated because of database server or network issues.

Sort Criteria

None.

Selection Criteria

Archiving the Payment Hold Maintenance Table:

- The process shall identify and archive all the records on the Payment Hold Maintenance table where the Hold Level is “Vendor”, having no corresponding Vendor records on the VCUST table. Vendor of the Payment Hold Maintenance table shall be used to do look up on the VCUST table. The process shall not archive the Payment Hold Maintenance Table record if there is a matching record exists in the VCUST table.
- The process shall identify and archive all Master Agreement records on the Payment Hold Maintenance table where the Hold Level is “Award”, having no corresponding Master Agreement transactions on the MA_DOC_HDR table. Transaction Code, Transaction Department, Transaction ID of the Payment Hold Maintenance table shall be used to do look up on the MA_DOC_HDR table. The process shall not archive the Payment Hold Maintenance table record if there is a matching record exists in the MA_DOC_HDR table.
- The process shall identify and archive all Encumbrance records on the Payment Hold Maintenance table where the Hold Level is “Award”, having no corresponding Encumbrance transactions on the PO_DOC_HDR table (for commodity-based) or ABS_DOC_HDR table (for accounting-based). Transaction Code, Transaction Department, Transaction ID of the Payment Hold Maintenance table shall be used to do look up on the PO_DOC_HDR table. The process shall not archive the Payment Hold Maintenance table record if there is a matching record exists in the PO_DOC_HDR table.
- The process shall identify and archive all Payment Request records on the Payment Hold Maintenance table where the Hold Level is “Payment Request”, having no corresponding Payment Request transactions on the PR_DOC_HDR table (for commodity-based) or ABS_DOC_HDR table (for accounting-based). Transaction Code, Transaction Department, Transaction ID of the Payment Hold Maintenance table shall be used to do look up on the PR_DOC_HDR or ABS_DOC_HDR tables. The process shall not archive the Payment Hold Maintenance table record if there is a matching record exists in the PR_DOC_HDR or ABS_DOC_HDR tables.

Archiving the Payment Hold Maintenance Transactions:

- The process shall identify and archive all Payment Hold Maintenance transactions with no corresponding records on the Payment Hold Maintenance table. Auto Generated Number of the Payment Hold Maintenance transaction shall be used to do look up on the Payment Hold Maintenance table. The process shall not archive the Payment Hold Maintenance Transactions if there is a matching record exists in the Payment Hold Maintenance table.

Archiving the Payment Hold Activity table:

- This process shall identify and archive all records on the Payment Hold Activity table with no corresponding Payment Request transactions on the PR_DOC_HDR table (for commodity-based) or ABS_DOC_HDR table (for accounting-based). Payment Request Transaction Code, Payment Request Transaction Department, Payment Request Transaction ID of the Payment Hold Activity table shall be used to do look up on the PR_DOC_HDR or ABS_DOC_HDR tables. The process shall not archive the Payment Hold Activity Table record if there is a matching record exists in the PR_DOC_HDR or ABS_DOC_HDR tables.

Archiving the Payment Hold Notice Table:

- The process shall identify and archive all records on the Print Payment Hold Notice table with no corresponding records on the Payment Hold Activity table. Reference Printed Payment Notice Number of the Payment Hold Activity table shall be used to do look up with the Reference Printed Payment Notice Number on the Payment Hold Notice table. The process

shall not archive the Payment Hold Notice Table record if there is a matching record exists in the Payment Hold Activity table.

Problem Resolution

Step 1: Validate the Parameters

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	N/A	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	We are not setting this return code.		
Failed (12)	Job failed due to Fatal conditions	In this step, the job can fail under the following three conditions. 1) Encounters any runtime exceptions and 2) Failed during restart. 3) Parameter validation fails. If the job fails because of the runtime exceptions, investigate the Logger file, resolve the error and restart the job.	N/A
	Failed while restarting the job since another instance of the job has already been run successfully.	**Recommendation:** Schedule a new job.	
Terminated (16)	Job is terminated manually by the user.	Reason for the termination needs to be investigated. The job can either be restarted or schedule a new job.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.

Possible Return Codes	Condition	Recommendation	Other Instructions
System Failure (20)	When the job is terminated because of database server or network issues	Reason for the System Failure needs to be investigated. The job can either be restarted or schedule a new job.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.

Step 2: Archiving the Payment Hold Maintenance Table

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	N/A	N/A	N/A
Warning (4)	Job ended with a Warning because there are no eligible records found to archive **Sample Message:** No eligible records found on the Payment Hold Maintenance table for archiving	N/A	N/A
Non-Fatal Error (8)	We are not setting this return code.		
Failed (12)	Job failed due to Fatal conditions	In this step, the job can fail under the following three conditions. 1) Encounters any runtime exceptions and 2) Failed during restart. If the job fails because of the runtime exceptions, investigate the Logger file, resolve the error and restart the job.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.

Possible Return Codes	Condition	Recommendation	Other Instructions
	Failed while restarting the job since another instance of the job has already been run successfully.	**Recommendation:** Schedule a new job.	
Terminated (16)	Job is terminated manually by the user.	Reason for the termination needs to be investigated. The job can either be restarted or schedule a new job.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.
System Failure (20)	When the job is terminated because of database server or network issues	Reason for the System Failure needs to be investigated. The job can either be restarted or schedule a new job.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.

Step 3: Archiving the Payment Hold Maintenance Transactions

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	N/A	N/A	N/A
Warning (4)	Job ended with a Warning because there are no eligible records found to archive **Sample Message:** No eligible Payment Hold Maintenance Transactions found for archiving.	N/A	N/A
Non-Fatal Error (8)	We are not setting this return code.		

Possible Return Codes	Condition	Recommendation	Other Instructions
Failed (12)	Job failed due to Fatal conditions	In this step, the job can fail under the following three conditions. 1) Encounters any runtime exceptions and 2) Failed during restart. If the job fails because of the runtime exceptions, investigate the Logger file, resolve the error and restart the job.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.
	Failed while restarting the job since another instance of the job has already been run successfully.	**Recommendation:** Schedule a new job.	
Terminated (16)	Job is terminated manually by the user.	Reason for the termination needs to be investigated. The job can either be restarted or schedule a new job.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.
System Failure (20)	When the job is terminated because of database server or network issues	Reason for the System Failure needs to be investigated. The job can either be restarted or schedule a new job.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.

Step 4: Archiving the Payment Hold Activity table

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	N/A	N/A	N/A

Possible Return Codes	Condition	Recommendation	Other Instructions
Warning (4)	Job ended with a Warning because there are no eligible records found to archive **Sample Message:** No eligible records found on the Payment Hold Activity table for archiving.	N/A	N/A
Non-Fatal Error (8)	We are not setting this return code.		
Failed (12)	Job failed due to Fatal conditions	In this step, the job can fail under the following three conditions. 1) Encounters any runtime exceptions and 2) Failed during restart. If the job fails because of the runtime exceptions, investigate the Logger file, resolve the error and restart the job.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.
	Failed while restarting the job since another instance of the job has already been run successfully.	**Recommendation:** Schedule a new job.	
Terminated (16)	Job is terminated manually by the user.	Reason for the termination needs to be investigated. The job can either be restarted or schedule a new job.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.
System Failure (20)	When the job is terminated because of database server or network issues	Reason for the System Failure needs to be investigated. The job can either be restarted or schedule a new job.	If another instance of the job has already been scheduled and ran successfully, then this job should

Possible Return Codes	Condition	Recommendation	Other Instructions
			not be restarted – only new job should be scheduled.

Step 5: Archiving the Payment Hold Notice Table

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	N/A	N/A	N/A
Warning (4)	Job ended with a Warning because there are no eligible records found to archive **Sample Message:** No eligible records found on the Payment Hold Notice table for archiving.	N/A	N/A
Non-Fatal Error (8)	We are not setting this return code.		
Failed (12)	Job failed due to Fatal conditions	In this step, the job can fail under the following three conditions. 1) Encounters any runtime exceptions and 2) Failed during restart. If the job fails because of the runtime exceptions, investigate the Logger file, resolve the error and restart the job.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.
	Failed while restarting the job since another instance of the job has already been run successfully.	**Recommendation:** Schedule a new job.	

Possible Return Codes	Condition	Recommendation	Other Instructions
Terminated (16)	Job is terminated manually by the user.	Reason for the termination needs to be investigated. The job can either be restarted or schedule a new job.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.
System Failure (20)	When the job is terminated because of database server or network issues	Reason for the System Failure needs to be investigated. The job can either be restarted or schedule a new job.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.

Step 5: Create parameter file

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	N/A	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	We are not setting this return code.		
Failed (12)	Job failed due to Fatal conditions	In this step, the job can fail under the following three conditions. 1) Encounters any runtime exceptions and 2) Failed during restart. If the job fails because of the runtime exceptions, investigate the Logger file, resolve the error and restart the job.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.
	Failed while restarting the job	**Recommendation:** Schedule a new job.	

Possible Return Codes	Condition	Recommendation	Other Instructions
	since another instance of the job has already been run successfully.		
Terminated (16)	Job is terminated manually by the user.	Reason for the termination needs to be investigated. The job can either be restarted or schedule a new job.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.
System Failure (20)	When the job is terminated because of database server or network issues	Reason for the System Failure needs to be investigated. The job can either be restarted or schedule a new job.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.

Payment Hold Archive Process: Table Archive Job

Chain or Job Name	Table Archive
Recommended Frequency	On Demand
Single Instance Required	No
Can be restarted?	Yes
Reports generated	None

Overview

This is the second job in the chain. The Facilitator process has a set of standard parameters required for archive processing. These parameters are chain job level parameters. The Report Only and Suppress Reports parameters are entered at the same time as the Selection Process parameters.

The Archive Facilitator process reads the Archive Facilitator records inserted by the Selection job, and for each record found whose Facilitator Status = *Table ready for archive,* the Facilitator initiates a System Maintenance Utility process. The System Maintenance Utility is invoked with an action of *Table Archive* and archives all records identified in the related parameter file with each individual Archive Facilitator record. As the System Maintenance Utility is processing the file the Facilitator updates the **Facilitator Status** on the Archive Facilitator record being

processed. The status is set to *Table Archiving* while the archive is occurring, and once it completes the status changes to *Table Archive Complete*.

There are no reports or output files created from the archive process.

The following table shows the various steps that the AD XML Creation Job goes through and the messages issued at each step.

Process Steps	Messages
Validate the Parameters	• Validating Batch Parameters • Parameters are valid or invalid depending on the Validation. If the parameter is invalid, the invalid value will be displayed in the log. • Batch Parameter validation completed
Processing and archiving records	The following messages are issued each time a System Maintenance Utility process is spawned. • The Run Number for this archive/restore process = <job_No> • SMU Job - <Job_No> – Spawned • SMU Job - <Job_No> - Processing completed successfully • The job slept a total of <No.> times, for a Total Sleep Time of <No.> seconds

Restartability

If the job was discontinued for any reason then the chain has the ability to be restarted from the point it left off.

It is compulsory to reschedule the entire chain in place of just rescheduling this job alone; otherwise, the following error message will be logged, “No Facilitator records found for Run Number <Run_No>. Nothing to process.” Since this job processes the records based on the parameter file created in the first job, an entry is made into the facilitator table for the parameter file location along with the Run_No. In this case it takes the Run_No of the current Job and there will be no record present in the Facilitator table for this; therefore, it will not be able to process any facilitator records.

Major Input

Data from the following tables:

- Print Payment Hold Maintenance Table (R_PYMT_HLD_MNTN)
- Facilitator table
- System Maintenance Utility parameter files.
- Print Payment Hold Notice Table (R_PRNT_PYMT_HLD_NTC)
- Payment Hold Activity Table (R_PYMT_HLD_ACT)

Batch Parameters

Parameter	Description	Default Value
Archive Restore ID (1- Table Archive) (ARCHIVE_RESTORE_ID)	Required editable Field. This parameter defines the location to which the system generated parameter and SQL files will be written.	
Tolerance TOLERANCE	This parameter defines a benchmark for the approximate number of records to be written to each XML archive file.	1000
Report Only REPORT_ONLY	Required Field. This parameter defines the "Report_Only" parm to define if records are to be purged in addition to reporting on the selected records.	N
Suppress Reports SUPPRESS_RPTS	Required Field. This parameter defines the "SUPPRESS_REPORTS" parm in the parameter file. When this parameter is defined as *Y* a related SMU report is not generated for each processed parameter file. If *N,* one SMU report is generated for each parameter file.	N
Progression counter size PROG_CTR_SZ	Field used to indicate the progression with respect to the number of records that have been archived.	1000
Update Facilitator Status UPDATE_STATUS	Update Facilitator Status. If Report Only is *Y* Update Status must be *N.*	Y

Major Output

Following are the output of the batch job:

- Run Number_Sequence Number_Label_YYYYMMDD.txt
- Run Number_Sequence Number_Label_YYYYMMDD.sql

Job Return Code

Return Code	Condition
Successful (1)	All of the records archived correctly.
Warning (4)	No Facilitator records were found.
Non-Fatal	This job does not have a return code of Non-Fatal Error.

Return Code	Condition
Error (8)	
Failed (12)	The job will fail under the following conditions: • If any run time exception occurs • Parameters are invalid
Terminated (16)	This return code will be issued when the job is terminated by the user.

Sort Sequence

N/A

Selection Criteria

Records from the Facilitator table are selected for processing based on the following criteria:

Status should be “Table ready for archive”

Archive Type should be “EFT table Archive”

Problem Resolution

The following table shows the possible return codes and recommendations for each processing step.

Problem Resolution

Step 1: Batch parameter validation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All batch parameter validations are successful (that is, there are no errors raised during validations.)	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A

Possible Return Codes	Condition	Recommendation	Other Instructions
Failed (12)	Required parameters are not entered.		**Sample Messages:** ARCHIVE_RESTORE_ID is required to run the Facilitator
	The RUN_NO is required to run the Facilitator		PROCESSOR_NO is required to run the Facilitator
	COMMIT_BLOCK_SIZE is required to run the Facilitator		SLEEP_TIME is required to run the Facilitator
Reschedule the job and enter the required parameters		Entered Parameters are not valid.	

Step 2: Processing and archiving records

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful.	N/A	N/A
Warning (4)	When no Facilitator records are found.	Make sure the Facilitator record exists and schedule a new chain.	
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Failed because of runtime exceptions caused by unexpected conditions.	Failure reason needs to be investigated and a new chain scheduled.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated and a new chain scheduled.	

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful.	N/A	N/A
System Failure (20)	Job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated and a new chain scheduled.	

2.1.40 PCard Chain

A Procurement Card is a charge card (i.e., Visa, MasterCard) that may be used to purchase goods for business purposes. The card can be used with any merchant that accepts the charge card. Procurement Cards are used to establish a more efficient, cost-effective alternative for agencies to make small purchases. It saves them time and money by consolidating paperwork, suppliers will receive payments more quickly, and the agency will get the supplies needed without delays.

The PCard Chain consists of five batch processes that are chained together. Each step in the process is described below.

Description

PCard Flat to XML Process

The PCard Flat to XML process converts the flat file sent by the credit card company into XML format.

Unprocessed Procurement Card Load Process

This process loads the XML file generated from the PCard Flat File to XML process to the Unprocessed Procurement Card Activity, Unprocessed Procurement Card Activity - Commodity Lines and Unprocessed Procurement Card Activity - Accounting Lines tables. The process also deletes the records from these tables that are marked for deletion and creates a report of the deleted records. The Unprocessed Procurement Card Load Exception Report is created by the batch process. If the bank file is not loaded successfully for this job, an error message will be sent to the exception report for this job. The message will indicate that the file did not load successfully. In addition, if the PCard ID is not found on the PRCUA table, the system will write out the values from the bank file to the report and issue an error message.

Procurement Card Payment Creation

This process initiates payments to the credit card company for purchases made using a procurement card. This offline process generates only an XML file to create a Payment Request in the draft version. Payment Creation Exception report is added to display the Administrator ID, Account No, PCard ID, and Error Message.

This batch job also contains the process by which the Vendor/Customer Code corresponding to each PRCUU record is selected. The process will first look to match a Vendor/Customer Code based on Merchant ID. If the system finds a matching Merchant ID on the VCUST table corresponding to an Active VCUST record, the system will use that record for the Vendor/Customer Code in the XML file. If there is no Active VCUST record with a matching Merchant ID, the system will then attempt to match using TIN/TIN Type. As before, if the system finds an Active VCUST record with a matching TIN/TIN Type to the PRCUU record, it will use that record for the Vendor/Customer Code in the XML file. If there are no Active VCUST records with matching Merchant ID or TIN/TIN Type to the PRCUU record, the system will use the Miscellaneous Vendor code from the batch parameter to populate the Vendor/Customer Code in the XML file. The Disbursement Form from the PRCUU record will be used in creation of PR Transactions.

PCard Upload

This process loads the XML file into the system with a status of Ready or Draft as defined by batch parameters.

PCard Submit

The PCard Submit process submits the Payment Request transactions that were created in the Procurement Card Payment creation job.

When to Run

Monthly, weekly, or on demand

Major Input

PCard Flat to XML Process

Flat file sent by the credit card company.

Unprocessed Procurement Card Load Process

The XML file created in the Flat to XML process.

Procurement Card Payment Creation

Auto Transaction Number (AUTO_DOC_NO) table

Unprocessed Procurement Card Activity (R_AP_UPROC_PRCU) table

Unprocessed Procurement Card Activity - Commodity Lines (R_AP_UPROC_PRCU_CL) table

Unprocessed Procurement Card Activity - Accounting Lines (R_AP_UPROC_PRCU_AL) table

Procurement Card Administration (R_AP_PRCU_ADMIN) table

Vendor/Customer (R_VEND_CUST) table

Output

- Payment Request Transaction(s)
- Updates to the Unprocessed Procurement Card Activity table
- Updates to Unprocessed Procurement Card Activity - Commodity Lines table
- Updates to Unprocessed Procurement Card Activity - Accounting Lines table

Parameters

Job	Parameter	Description	Default Value
FlatFileToXml	Attribute List for Record Type H (ATTRIBUTE_LIST)	The process requires the Attribute List for Record Type H as a parameter; the sequence of the list must match the sequence in the flat file.	REC_TYP,ADNDUM_KEY,PCARD_ID,ACCT_NO,MRCHT_ID,MRCHT_NM,TRAN_DT,TRAN_TM,BANK_PSTNG_DT,TIN,TIN_TYP,TRAN_AM,TRAN_REF_NO

Job	Parameter	Description	Default Value
	Attribute List for Record Type C (ATTRIBUTE_LIST_1)	The process requires the Attribute List for Record Type C as a parameter; the sequence of the list must match the sequence in the flat file.	REC_TYP,ADNDUM_KEY,COMM_CD,BANK_COMM_DSCR,COMM_QTY,UNIT_AM,UNIT_MEAS_CD,TRAN_AM,PO_DOC_CD,PO_DOC_DEPT,PO_DOC_ID,COMM_LN_NO
	Dataobject Name (DATAOBJECT_NAME)	Required Field. This is used to create a Data Object Node in XML.	R_AP_UPROC_PRCU
	Dataobject Name 1 (DATAOBJECT_NAME_1)	Required Field. This is used to create a Data Object Node in XML.	R_AP_UPROC_PRCU_CL
	DELIMITER (DELIMITER)	Required Field. Defines what separates the fields in the bank's data file.	,
	Flat File (FLAT_FILE)	Required Field. The name of the text file that contains the banks data.	No Default
	XML File (XML_FILE)	Required Field. Defines the XML file name to be created.	ProcurementPR.xml
	Export Location at FlatFileToXml Job (AMSEXPORT)	Required (** **Refer to Note**: Assumptions for SWBP on page no. 7)	-
	Parameter Location at FlatFileToXml Job (AMSPARM)	Required (** **Refer to Note**: Assumptions for SWBP on page no. 7)	No Default

Job	Parameter	Description	Default Value
UnProcPcardLoad	Client name For Report (CLIENT_NM)	Optional field. Entry of a value in this field specifies the name that will appear on report.	No Default
	Payment Flag (PAY_FLAG)	Required Field. Defines how the payment flag will be set on the Unprocessed Procurement Card table.	Yes
	Reconciliation Status (RECON_STA)	Required Field. Defines how the Reconciliation Status flag will be set on the Unprocessed Procurement Card table.	Not Reconciled
	Default PCard Commodity Code (DFLT_PCARD_COMM_CD)	Optional Field. This parameter will be used to specify the default Commodity Code value that should be used in the event that the Bank file does not specify Commodity details, and there is no Commodity Code listed on VENDCOM	
	XML File (XML_FILE)	Required Field. Defines the name of the file to be uploaded.	ProcurementPR.xml
	Export Location at UnProcPcardLoad Job (AMSEXPORT)	Required (** **Refer to Note**: Assumptions for SWBP on page no. 7)	No Default

Job	Parameter	Description	Default Value
	The parameter when set to Y(Y=YES), the job return code will be Non-Fatal, when set to N(N=NO) the job return code will be Warning. (PCARD_JOB_RETURN_INVALID)	Required Field. Defines the job return code when the bank file has an invalid PRCUA record.	N
PaymentCrea	Administrator ID – Optional (ADMR_ID)	Optional Field. If entered, will be included on the header of the Payment Request transactions.	
	Transaction Code (DOC_CD)	Required Field. This field refers to the Transaction that will be generated by the Payment Creation process.	PRCC
	Department Code (DOC_DEPT_CD)	Optional Field. The transaction department to be used with the transaction code and prefix to find an Automatic Transaction Numbering entry. Will appear as the transaction department for all transactions created by the program.	
	Transaction Unit Code (DOC_UNIT_CD)	Optional Field. Provides security with a unit code enabling transaction access to be secured at a level below department.	No Default

Job	Parameter	Description	Default Value
	Event Type – Optional (EVNT_TYP_ID)	Optional Field. If entered, will be updated to each Accounting Line of the Payment Request.	AP01
	Prefix (PFX)	Optional Field. The transaction prefix to be used with the transaction code and department to find an Automatic Transaction Numbering entry. Will appear in the beginning of all transaction ID's for all transactions created by the program.	No Default
	Payee Vendor Code (PYEE_VEND_CD)	Required Field. Vendor Code to be used on the header of the Payment Request.	
	Scheduled Payment Date-(MM/DD/YYYY) (SCHED_PYMT_DT)	Optional Field. If entered, will be inferred onto each Vendor Line of the Payment Request. (** **Refer to Note**: Pivot Date/Year Validation, while entering the date)	
	Select Block Size (SELECT_BLOCK)	Optional Field. If not entered then the value will default to 1000. This is the number of records that should be selected at one time. Can be used for performance tuning.	1000

Job	Parameter	Description	Default Value
	Unit of Measure Code – Optional (UNIT_MEAS_CD)	Optional Field. If entered, will be inferred into the Unit of Measure field of each Commodity Line of the Payment Request.	
	Misc. Vendor Code (VEND_CUST_CD)	Required Field. Will be inferred on the Vendor Line of the Payment Request if the Merchant ID is not found on the Vendor/Customer table or a PO reference is not available on the PRCUU table.	
	Export Location at PaymentCrea Job (AMSEXPORT)	Required (** **Refer to Note**: Assumptions for SWBP on page no. 7)	
	Use the Dept and Unit from PRCUA table. Y/N (Y= Yes N=No) (PRCUA_DEPT_UNIT_CD)	Required Field. If the parameter is set to 'Y', using the PCard ID on the PRCUU table and the Administrator ID if they exist, the system will retrieve the record on the PRCUA table. If the record is found on PRCUA then the system will verify a record on ADNT using the current fiscal year and the Transaction Code and the Administration Department code. If the record exists, the system will use the	No Default

Job	Parameter	Description	Default Value
		Transaction Code from the batch parameters, the Administration Department as the Transaction Department and the Administration Unit, if one exists for the Transaction Unit code. If the value is set to 'N' the system will reference the Department Code and Transaction Unit Code currently entered on the batch parameters.	
	Cited Authority (CITED_CD)	Optional Field. The Cited Authority, if entered, exists on the Cited Authority table for the transaction code entered in the batch parameters.	No Default
	Misc Address 1 (MISC_ADDRESS_1)	Optional Field. The value will be used in the Address 1 field of the payment request transaction for miscellaneous vendors.	No Default
	Misc City (MISC_CITY)	Optional Field. The value will be used in the City field of the payment request transaction for miscellaneous vendors.	No Default

Job	Parameter	Description	Default Value
	Misc Country Code (MISC_CTRY_CD)	Optional Field. The value will be used in the Country field of the payment request transaction for miscellaneous vendors.	No Default
	Misc State Code (MISC_ST_CD)	Optional Field. The value will be used in the State field of the payment request transaction for miscellaneous vendors.	No Default
	Misc Zip Code (MISC_ZIP_CD)	Optional Field. The value will be used in the Zip field of the payment request transaction for miscellaneous vendors.	No Default
	Enter the Credit Card Posting Date (MM/DD/YYYY) (PST_DT)	Optional Field. This field will store information of the Posting Date used for the bank file.	No Default
	Misc Legal Name (MISC_LEG_NM)	Optional Field. The parameter to store the Misc Legal Name value.	No Default
PCard Upload	Action Code: Import (ACTN_CD)	Required (and maybe protected) action code which instructs the program to perform a certain action on its records.	171

Job	Parameter	Description	Default Value
	Transaction Status Code (1-Held,2-Ready) (DOC_STA_CD)	Required field that will be used by the program when loading transactions. Valid values are 1 - Held and 2 - Ready. Transactions loaded as held will not be selected by another process searching for transactions to submitting. Ready transactions will be selected by such a program. Held is often used when some user action will be required.	1
	File import from, export to (FILE_NM)	Required (and protected) field that defines what file will be uploaded.	$$AMSEXPORT$$/ProcurementPR.xml
PCard Submit	Exception Report File (EXCEP_REP_FILE_NM)	Required field that defines what file the program is to use to create an exception report.	$$AMSLOGS$$/PcardExep.txt
	Parameter File: (PARM_FILE)	Required (and protected) field that will be used to define the parameters for submitting the Payment Request transactions.	$$AMSPARM$$/SubmitPrcParm.txt

Sort Sequence

PCard Flat to XML Process

None

Unprocessed Procurement Card Load Process

Administrator ID

Card Number

Account Number

Procurement Card Payment Creation

If no record exists in the PCard Configuration table (R_AP_PCARD_CNFG) for the Current FY:

- Administrator ID
- Merchant ID

If a record exists in the PCard Configuration table (R_AP_PCARD_CNFG) for the Current FY and the Payment Request PCard Consolidation field on PCNFG is set to 'Administrator ID and PCard ID':

- Administrator ID
- Card Number
- Billing Cycle End Date

If a record exists in the PCard Configuration table (R_AP_PCARD_CNFG) for the Current FY and Payment Request PCard Consolidation field on PCNFG is set to 'Administrator ID':

- Administrator ID
- Billing Cycle End Date

Selection Criteria

PCard Flat to XML Process

None

Unprocessed Procurement Card Load Process

To delete records:

Select records where

Delete Flag = 'True' and

Process flag = 'True'

For the load process, all records are loaded from the XML file.

Procurement Card Payment Creation

Administrator ID

Merchant ID

Commodity Code

Use Dept and Unit from PRCUA table = Y

Using the PCard ID on the PRCUU and the Administrator ID if they exist, the system will retrieve the record on the PRCUA. If record found on PRCUA then system will verify a record on ADNT

using the current fiscal year and the Transaction Code and the Administration Department code. If the record exists, the system will use the Transaction Code from the batch parameters, the Administration Department as the Transaction Department and the Administration Unit, if one exists for the Transaction Unit code.

Use Dept and Unit from PRCUA table = N

The system will reference the Department Code and Transaction Unit Code currently entered on the batch parameters.

To create payments for a particular Administrator ID (you specify Administrator ID as a parameter):

Select records where:

Administrator ID = chosen value and

Payment Flag = ‘True’ and

Delete Flag = ‘False’ and

Processed Flag = ‘False’

To create payments for all Administrator IDs (you do not specify an Administrator ID as a parameter):

Select records where:

Payment Flag = ‘True’ and

Delete Flag = ‘False’ and

Processed Flag = ‘False’

Problem Resolution

PCard Flat to XML Process

If the process fails for any reason, check to see whether an XML file of the name of the flat file or the name you specified already exists in the Datafiles/ImportExport directory with the current system date and time. If the process did not create an XML file, or the created XML file is not in the proper format, correct the problem and rerun the process.

Unprocessed Procurement Card Load Process

Check the parameter record and verify that the file name is entered correctly. Correct the problem and rerun the process.

Procurement Card Payment Creation

No database restore is required. Correct the problem and rerun the job executing the program. No restoration of data sets or files from backups is required for this program.

This batch program produces new Advantage transactions, which are subject to the line limit functionality constraints. Sites should ensure that they run this job with parameters set to ensure that the created transactions are within the line limit controls.

2.1.41 Prenote Batch Process

Description

A prenote is a record containing electronic funds transfer (EFT) information about a vendor such as Bank Account Number and Bank Name. This information is sent along with nightly payments to a Financial Institution, and notifies the Financial Institution of the intent to electronically deposit payments to one of its accounts and verifies that all the account information is correct.

The process selects records from the Vendor/Customer (R_VEND_CUST) and Vendor Address (R_AD) tables based on EFT status, generates EFT files and reports, and updates tables accordingly. There are four sub-processes involved in Prenote Offline Batch Process. The processes create the following chain:

- Bank Reply Process -- The first job in the chain has a parameter called 'Use Bank Reply File' and the valid values are "Y" and "N". If the parameter is set to "Y" then it will read the bank's reply file (Financial Institution Reply) that contains a list of rejected vendors, and updates the Vendor/Customer table and Vendor Address table for rejected vendors found in banks reply file. This job also updates Vendor EFT Status for records that are eligible for EFT. If the parameter is set to "N", the batch program will bypass the processing of an input file, bypass the automated updates of prenote rejections and move directly to the remaining functionality which includes updating the appropriate vendor record's EFT Status to Eligible for EFT.
- Prenote Report -- Generates all the reports.
- Build XML File -- Creates the XML file.
- Build Flat File -- Creates the flat files to be sent to the bank.

When to Run

It is expected for the report to be run daily as part of the nightly cycle with other EFT jobs.

Major Input

- Vendor/Customer (R_VEND_CUST)
- Vendor Address (R_AD)
- System Options (R_EXP_SOPT)

Output

- Flat Files to send to the Financial Institution to notify them of vendors that wish to receive electronic payment.
- Reports:
  - Prenote Selection Report – Lists all Vendors who have an EFT status equal to " Prenote Requested" in the VCUST table and Vendor Address table.
  - Prenote Invalid EFT Format Report – Generated at the end of the Prenote Process from the VCUST table and Vendor Address table and displays all vendors with an invalid Disbursement Format. Vendors with an invalid Disbursement Format remain in Prenote Requested status until their Disbursement Format is corrected and the process is rerun
  - Prenote Exception Report – Generated at the end of the Prenote process and lists all the Vendors whose EFT status is equal to "Prenote Rejected" (rejected by the Bank) in the

VCUST table and Vendor Address table and the reason why the vendor was rejected for EFT payments.

Table Updates

- For Prenote Requested vendors:
  - Generate bank file(s) (one bank file for all vendors using the same bank).
  - Generate Selection Report

- For Prenote Pending vendors exceeding number of Prenote days set on the System Options:
  - Update the Vendor/Customer (R_VEND_CUST) and Vendor Address (R_AD) tables to Eligible for EFT.

Parameters

Job	Parameter	Description	Default Value
Banks Reply	Bank's Reply File Name with Extension (BANK_REPLY_FILE_NM)	Required Field when user selects *USE_BANK_REPLY_FILE* = 'Y'. Defines the name of the bank's data file.	BankReply.dat
	Use Bank Reply File (*USE_BANK_REPLY_FILE*)	Required Field *and the valid values are 'Y' and 'N'*.	Y
	Commit Block Size : (COMMIT_SIZE)	Commit Block Size. if not entered then defaulted to 100	100
	Import Location at Banks Reply Job (AMSIMPORT)	Required (** **Refer to Note**: Assumptions for SWBP on page no. 7)	No Default
Prenote Reports	Client Name for Report (CLIENT_NM)	Optional field. Entry of a value in this field specifies the name that will appear on report.	No Default

Job	Parameter	Description	Default Value
	Max. Number of records to be Processed before Flush (MAX_FLUSH_COUNTER)	Required Field.	100
	Report ID (REPORT_ID)	Optional field. Entry of a value in this field specifies the ID that will appear on report.	No Default
Build XML File	Process Date (CLR_DT)	Clear Date Or Effective Date (Valid format is MM/DD/YYYY) (** **Refer to Note**: Pivot Date/Year Validation, while entering the date)	If left blank, APPCTRL Date
	Payer Contact Code (CNTAC_CD)	Required Field.	No Default
	Payer Department Code (DEPT_CD)	Required Field.	No Default
	Payer Unit Code (UNIT_CD)	Required Field.	No Default
	Export Location at Build XML File Job (AMSEXPORT)	Required Field	No Default
	Identification Number (ID_NUM)	Identification Number: Valid Values are 1 (Routing ID Number) and 2 (Vendor Customer Code). 1 is the default value.	1

Job	Parameter	Description	Default Value
	Receiving Name (RCV_NM)	Receiving Name: Valid Values are 1 (Vendor Customer code) and 2 (Either the Vendor's Last Name or Company Name). 1 is the default value.	1
	Type of Prenote Transactions (TYP_PRENOTE_TRNSC)	Required Field. Valid values are 1 - Credit Prenotes only or 2 - Credit and Debit Prenotes	
	Generate Addenda Record (GEN_ADDENDA_REC)	Required Field. Valid values are *True* or *False*. Indicates whether or not the Addenda records should be created	
	Credit Detail Discretionary Data (CR_DET_DISCR_DATA)	Optional Field. If entered, the length should not exceed 2 characters.	
	Debit Detail Discretionary Data (DBT_DET_DISCR_DATA)	Optional Field. If entered, the length should not exceed 2 characters.	
	Company Entry Description (COMP_ENTR_DESC)	Optional Field. If entered, the length should not exceed 10 characters.	
	Lag Days (LAG_DY)	Optional field. This represents Business Days and not Calendar Days. If populated, the system will set the Process Date within the batch job to the Application System Date + the Lag Days.	No Default

Job	Parameter	Description	Default Value
	Settlement Date (SETTLEMENT_DT)	Required field. Valid values are: • 1 (sets value to 000), • 2 (sets value to 3 blank spaces), and • alphanumeric string of length 3.	1
Build Flat File	Prenote XML File Name (FILE_NAME)	Required Field. Defines the name of the file to be created by the process.	PrenoteACHXMLFormatFile.xml
	FLAT_FILE_EXTENSION	Flat file extension. Valid values are *DAT* or *txt*.	DAT
	FLAT_FILE_PREFIX	Flat file prefix	Prenote
	FLAT_FILE_SUFFIX	Flat file suffix	File
	Return Code when Flat files are not generated (NO_FLAT_FILE_GEN_RET_CODE)	An optional parameter to update Return Code when no flat files are generated post job completion with valid values: 1 - Success, 4 - Warning. • Value 1 returns the Return Code of Successful. • Value 4 returns the Return Code of Warning.	4

Sort Criteria

None

Selection Criteria for Batch process

Select records from the Vendor/Customer table (R_VEND_CUST) table

Where EFT Status = 1 (Prenote Requested)

Select records from the Vendor Address table (R_AD) table

Where EFT Status = 1 (Prenote Requested)

Select FY (Fiscal Year) information from R_CLDT table

Use current system date as input.

Select Number of Prenote Days from R_EXP_SOPT table

FY is the Fiscal Year obtained from R_CLDT table

Select records from the Vendor/Customer table (R_VEND_CUST) table

Where EFT Status = 2 (Prenote Pending)

Where the number of days this record is in the Prenote Pending status is more than the Number of Prenote Days obtained from R_EXP_SOPT table.

Select records from the Vendor Address (R_AD) table

Where EFT Status = 2 (Prenote Pending)

Where the number of days this record is in the Prenote Pending status is more than the Number of Prenote Days obtained from R_EXP_SOPT table.

Problem Resolution

No database restore is required. Correct the problem and rerun the job executing the program. No restoration of datasets or files from backups is required for this program.

2.1.42 Print Backup Withholding Notices

Chain or Job Name	Print Backup Withholding Notices
Recommended Frequency	Twice a year
Single Instance Required	Single
Can be restarted?	The job can be restarted
Reports generated	Four Reports: • TIN Not Found Exception • TIN Name Mismatch Exception • Backup Withholding B Notices Original Printing Report • Backup Withholding B Notices Re-Print Report

Overview

Print Backup Withholding Notices Batch Job selects records from the 1099 Reporting Information (1099I) and Backup Withholding Notice Printing History (BWNPH) tables for taxpayers, for whom the generation of the 1st or 2nd B Notice is required. Both tables are updated by the process, as well as serve as input for identifying records that require printing (as per '1099I table setup) or were scheduled for re-printing (as per BWNPH table setup) of a corresponding B Notice.

Three independent sets of selection logic look for Backup Withholding Notices to print:

1. If run with the IRS CP2100 ('IRS Backup Withholding') file, this batch job compares the 'TIN', 'TIN Type', and 'Name 1' on the IRS file to the Advantage 1099I table to identify Taxpayers with wrong or incorrect TIN, TIN Type, and/or Name information. TIN numbers on the IRS file that have a matching record on 1099I are selected for printing if they match other selection criteria. TIN numbers on the IRS file that do not match 1099I are written to an exception report.
2. The job always looks for B notices that were "manually" requested by users by checking print flags on 1099I. "Manual" 1099I print requests that match other selection criteria will be picked up for printing.
3. Finally, the job looks for reprint requests manually entered on the Backup Withholding Notice Printing History (BWNPH) page.

The job can be executed in two modes - 'Report' and 'Update'. The output of this process in 'Update' mode are files of 1st or 2nd B Notices (the two have different forms) and table updates to the 1099I and BWNPH records selected. Also, new notices printed (not reprints) inserts new records on BWNPH. BWNPH is updated only by this batch job. The table stores history information of printed 1st and 2nd B Notices. Reports are produced in both 'Report' and 'Update' mode.

In 'Update' mode, this job uses its parameter values, as well as values on the selected records on 1099I or BWNPH table, to populate the specific lines on the printed notices.

For stand-alone batch jobs (including the jobs in the chain):

- This process identifies taxpayers eligible for backup withholding or to print the 1st and 2nd B Notices for applicable taxpayers. This is an automated process to identify taxpayers whom the IRS has identified as having a missing or invalid Name and TIN combinations.
- The process involves the following steps.
  - Parameter Validation
  - Selection of records from IRS CP2100 File if the Use of the File is set as *yes.*
  - Comparison of Records from the file with 1099I table and selection of Records
  - Selection of manual records
  - Process all selected records and print 1st B or 2nd B Notices.
  - Update 1099I and BWNPH
  - Create the following reports:

TIN Not Found Exception

TIN Name Mismatch Exception

Backup Withholding B Notices Original Printing Report

Backup Withholding B Notices Re-Print Report.

Process Steps	Messages
1. Parameter Validation	• Validating Batch Parameters • Parameters are valid or invalid depending on the Validation. If the parameter is invalid, the invalid value is displayed in the log. • Batch Parameter validation completed
2. Selection of Records	• Selecting eligible records matching from the IRS CP2100 file and 1099I. • Select the manual records from 1099I and BWNPH. • If the selection returns 0 records, then the following message is issued: “No eligible record found”. • Number of records (count) selected are be displayed. • At the end, the following message is issued: Selection of records completed.
3. Process Records	• Create the XML for both 1st B and 2nd B Notices for Printing. • Create the XML for both 1st B and 2nd B Master Notices for Printing. • If the Print job fails, then that is logged to the Job Log • After a Print Job is complete, the following message is issued: Printing of Notices Completed.

Process Steps	Messages
4. Update Tables	• Update 1099I and BWNPH with the information from the records processed for printing. • If the update fails, the following message will be issued: "Update of the records in the table failed" • If the update is successful, the following message is issued: Records have been updated successfully.
5. Create Reports	• Create 4 reports. • If the creation of reports fails, the following message is issued: Creation of reports failed. • If reports are created successfully. the following message is issued: Reports have been created successfully.

The job has restart capability.

- Check pointing is done on the following steps.
  - Validate Parameters
  - Selection of Records
  - Print Notices
  - Update tables
  - Create reports

Major Input

The following are the input for this job:

- IRS CP2100 File
- 1099 Reporting Information (1099I / R_TIN_1099_INFO)
- Backup Withholding Notice Print History (BWNPH / R_BKWLD_NTC_PRHST)

Note: The default values listed are those delivered with the software. Actual values may vary based on your site's setup.

Parameter	Description	Default Value
TRNS_CTRL_CD	Transmitter Control Code. This is optional.	No
APND_ST_ZIP_TO_TIN_CITY_NM	Append State and Zip along with TIN City name in address. This is an optional parameter, when set to Y (Yes), the State and Zip code will append along with City Name to TIN City name	N

	in the address level. If set to N (No), it will not append State and Zip code to City Name.	
PRNT_MSTR_FILE	Print Master File. This is an optional parameter, when set to Y (Yes), the master file will generate for all the Vendors' 1st B and 2nd B notices in a single PDF file. When a single file is created with all notices, all notices can be printed with one print action.	N
PRNT_ONE_NTC	Print Only One Notice Per Year. This is optional.	Y
OPER_MODE	Operation Mode. This is required. This will represent if the batch is going to be run in a 'Report' (1) mode or 'Update' (2) mode.	1
CLNT_NM	Client Name for Report. This is optional.	No
IRS_NTC_TAX_YR	IRS Notice Tax Year. This is required. It indicates the tax year printed on the notice and to exclude records where a B Notice has already been sent to a taxpayer in the same tax year. See note below on this parameter.	No
PRNT_SCND_ATCH	Print 2nd Notice Attachment. This is optional.	No
NTC_LN_1	Notice Line 1. This is optional. This will represent the value for Notice Line 1.	No
NTC_LN_2	Notice Line 2. This is optional. This represents the value for Notice Line 2.	No
NTC_LN_3	Notice Line 3. This is optional. This represents the value for Notice Line 3.	No
NTC_LN_4	Notice Line 4. This is	No

	optional. This represents the value for Notice Line 4	
NTC_LN_5	Notice Line 5. This is optional. This represents the value for Notice Line 5.	No
NTC_LN_6	Notice Line 6. This is optional. This represents the value for Notice Line 6.	No
ATCH_LN_1	Attachment Line 1. This is optional. This represents the value for Attachment Line 1 on 2nd B Notice Attachment.	No
CNTC_PH_NO	Contact Phone Number. This is optional. This represents the value for Contact Phone Number that will appear on 1st and 2nd B Notices	No
FRST_B_NTC	First B Notice BIRT Form. This is optional. This identifies the corresponding BIRT form for printing of the 1st B Notice.	No
SCND_B_NTC	Second B Notice BIRT Form. This is optional. This identifies the corresponding BIRT form for printing of the 2nd B Notice	No
PRNT_RSRC_ID	Print Resource Identifier. This is optional. This represents the output device (Print Resource) for the Notices to be printed.	No
IRS_PNLTY_AM	This is an optional parameter. This will represent the value for IRS Penalty Amount.	No
USE_IRS_FILE	Use IRS File. This is required. This will represent if the IRS CP2100 File will also be used in selection of records. If Y, the batch will select records from the IRS file and process the matching records on 1099I	N

	table as well as select records on 1099I table that satisfy the job's selection criteria. If N, the batch will only process records selected from 1099I table.	
IRS_FL_LOC	IRS File Location. This parameter is conditionally required. If the USE_IRS_FILE is set as 'Y' then this parameter is required and denotes the file location of the IRS CP2100 file.	No
IRS_FL_NM	IRS File Name. This parameter is conditionally required. If the USE_IRS_FILE is set as 'Y' then this parameter is required and denotes the file name of the IRS CP2100 file.	No

IRS Notice Tax Year**:** IRS regulations state that B Notices should be sent once per year and multiple notices should **not** be sent for the same tax year.

- This parameter has no impact on Notices printed due to the IRS File or due to reprints selected from BWNPH.
- For records that are manually selected for printing on 1099I, the job will select records from 1099I table only if the value in 'IRS Notice Tax Year' field is less than the value specified in this parameter. This comparison does not apply to any record where 'TIN Notice Counter' is blank (i.e. B Notice has not been printed).

  (Explanation: If the 1099I record's 'IRS Notice Tax Year' equals this parameter, then that means a notice was already sent for the Tax Year and the record should not be selected again. 1099I records that had notices printed in later tax years should not be selected either. This would not be a normal situation, but it is possible.)

Major Output

The following are the output for this job:

- 1099I - The selected records from this table for Printing of Notices are updated with relevant information.
- BWNPH - The selected records from 1099I have a corresponding record inserted into BWNPH. The selected records from this table for reprinting, have relevant updates.
- 1st and 2nd B Notices
  - TIN Not Found Exception
  - TIN Name Mismatch Exception
  - Backup Withholding B Notices Original Printing Report

- Backup Withholding B Notices Re-Print Report

The following tables are used as Temp tables during this process:

- RS_CP2100_TEMP
- R_IRS_BKUP_REP_TMP
- R_BWPNH_TMP

Return Code	Condition
Successful (1)	All the selected records are processed successfully
Warning(4)	No eligible records found. This could be because of the following reasons: • No matching records found between 1099I and IRS CP2100 table and no manual records found in 1099I table and BWNPH table
Non-Fatal Error (8)	
Failed (12)	The job will fail under the following conditions: • Parameters are invalid. • Unable to Insert Records in 1099I or BWNPH. • Run time exceptions for unexpected situations. When this job ends with a return of code *Failed*, subsequent jobs in the chain will be set to *Inactive*.
Terminated (16)	This return code will be issued when the job is terminated by the user. When this job ends with a return of code *Terminated* subsequent jobs in the chain will be set to *Inactive*.
System Failure (20)	This return code will be issued when the job is terminated because of database server or network issues. When this job ends with a return of code *System Failure*, subsequent jobs in the chain will be set to *Inactive*.

Sort Criteria

N/A

Selection Criteria

If the *Transmitter Control Code* parameter is blank, select all IRS File records, where the '*Form Type'* on the file's record equals "92" or "95".

For each selected IRS File record, the system will perform a look-up to the 1099I table to find a matching record there based on the TIN, TIN Type and Name combination.

- If no matching record is found on 1099I for the TIN and TIN Type, then the information from this IRS File record will be written to the 'TIN Not Found Exception Report'. No additional statistic is updated for these records. No further processing occurs for this IRS File record.
- If a matching record exists on 1099I for the TIN Type, but the Names on both matching records are not the same (value in 'Name' field on the 1099I table does not equal the value in 'Name 1' field of the record in the 'IRS Backup Withholding' file), then the information from this IRS File record will be written to the 'TIN Name Mismatch Exception Report'. No additional statistic is updated for these records. No further processing occurs for this IRS File record.
- If a match is found on 1099I for the TIN, TIN Type, and Names but the 'New TIN' field is populated, this record will not be processed because the TIN is out of date, then the information from this IRS File record will be written to the "TIN Not Found Exception Report". No additional statistic is updated for these records. No further processing occurs for this IRS File record.
- If a match is found on 1099I for the TIN, TIN Type, and Names and the 'New TIN' field is blank, then the IRS File record is selected for printing consideration. There will be updates performed to 1099I table and new records will be inserted into BWNPH table. Note that BWNPH table records will use the UNUM table to create Notice reference number field. Records from the IRS File will be considered for groups 1A, 1B, 2A, and 2B based on the criteria. For more detail on each group can be found at the end of this run sheet as described in the functional design DD2145.

1A: 1099I Backup Withholding Status is empty AND TIN Notice Counter is empty AND IRS Notice Tax Year is empty or IRS Notice Tax Year is NOT empty and is less than the Tax Year on the Payer Record "A" of the IRS file

1B: 1099I Backup Withholding Status is "pending" or "eligible" AND TIN Notice Counter is empty AND IRS Notice Tax Year is empty or IRS Notice Tax Year is NOT empty and is less than the Tax Year on the Payer Record "A" of the IRS file

2A: 1099 Backup Withholding Status is empty AND TIN Notice Counter is "1" AND IRS Notice Tax Year is empty or IRS Notice Tax Year is NOT empty and is less than the Tax Year on the Payer Record "A" of the IRS file AND Print Only One Notice Per Year job parameter is "N" or Print Only One Notice Per Year job parameter is "Y" and Calendar Years of the TIN Notice Date and Application Control Date are not the same

2B: 1099 Backup Withholding Status is "pending" or "eligible" AND TIN Notice Counter is "1" AND IRS Notice Tax Year is empty or IRS Notice Tax Year is NOT empty and is less than the Tax Year on the Payer Record "A" of the IRS file AND Print Only One Notice Per Year job parameter is "N" or Print Only One Notice Per Year job parameter is "Y" and Calendar Years of the TIN Notice Date and Application Control Date are not the same

- Groups 1C and 2C will represent the manually entered records on the 1099I table which will be selected when the Use IRS File parameter is set to 'Y' or 'N'.
    - **1C**: TIN Notice Counter is empty AND Issue 1st Notice flag is checked AND IRS Notice Tax Year is empty or IRS Notice Tax Year is not empty and is less than the IRS Notice Tax Year parameter.
    - **2C**: Issue 2nd Notice is checked AND Print Only One Notice per Year job parameter is "N" or Print Only One Notice Per Year job parameter is "Y" and Calendar Years of the TIN Notice Date and Application Control Date are not the

same. AND IRS Notice Tax Year is “blank” or IRS Notice Tax Year is not “blank” and is less than the IRS Notice Tax Year parameter.

Groups 3 and 4 will represent the manually entered records on the BWNPH table which will be selected when the Use IRS File parameter is set to ‘Y’ or ‘N’.

- **3**: TINs scheduled for 1st Notice Reprint and Notice Printing Status is “Replacement Pending” AND Requested Notice Reprint Date is less than or equal to the Application Control Date
- **4:** TINs scheduled for 2nd Notice Reprint and Notice Printing Status is “Replacement Pending” AND Requested Notice Reprint Date is less than or equal to the Application Control Date.

Problem Resolution

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All the parameters are validated successfully	N/A	N/A
Warning (4)	No records were found eligible.	Check the parameter values.	N/A
Non-Fatal Error (8)	This return code will be issued when the validation fails in this step.	N/A	N/A
Failed (12)	Job failed due to Fatal conditions.	If the job fails because of the runtime exceptions, investigate the exception reported by the process, resolve the error and restart the job.	Schedule a new job
	Failed while restarting the job since another instance of the job has already been run successfully. Cannot restart the job since another instance of this job has already been run successfully.	**Recommendation:** Schedule a new job.	

Possible Return Codes	Condition	Recommendation	Other Instructions
Terminated (16)	Job is terminated manually by the user.	Reason for the termination needs to be investigated. The job can either be restarted or schedule a new job.	Schedule a new job
System Failure (20)	When the job is terminated because of database server or network issues	Reason for the System Failure needs to be investigated. The job can either be restarted or schedule a new job.	Schedule a new job

Detailed selection logic for initial printing of B Notices appears in the table below. There are six groups of records (1A, 1B, 1C, 2A, 2B, and 2C). Note that groups 1A, 1B, 2A, and 2B are only selected if the job is run with the 'Use IRS File' parameter = "Y". Groups 1C and 2C are **always** tested for matches.

The **1st B Notice** will need to be printed for these groups:

	Group Description	Selection Criteria	Group depends on "Use IRS File" parameter	Updates to the 1099I table when the Operation Mode is set to "update
1A	TINs that appear for the first time on the IRS File and have never been sent a notice before.	Record appears on the 1099 File and is matched to a 1099I record. **AND** 1099I Backup Withholding Status is "blank" **AND** TIN Notice Counter is "blank" **AND** IRS Notice Tax Year is "blank" **or** IRS Notice Tax Year is NOT "blank" and is <u>less than</u> the Tax Year on the Payer Record "A" of the IRS file. <u>Statistics updated:</u> 'Total number of 1099I records selected for	Only selected if "Use IRS File" is Yes.	▪ Set 1099 Backup Withholding Status to "Pending" ▪ Populate IRS Notice Tax Year from the Tax Year on the Payer Record "A" of the IRS file ▪ Populate the TIN Status from BWH TIN Status of respective IRS record ▪ Set Withholding Tracking Start Date to blank ▪ Set Withholding Tracking Stop Date to blank

	Group Description	Selection Criteria	Group depends on "Use IRS File" parameter	Updates to the 1099I table when the Operation Mode is set to "update
		initial printing 1st B Notice due to IRS File match'; 'Total number of 1099I records bypassed for printing 1st B Notice because notice was printed in the same Tax Year'. 'Total number of 1099I records with Pending Backup Withholding'		▪ Set Start 1099 Backup Withholding Reason to "CP2100" ▪ Set Stop 1099 Backup Withholding Reason to blank ▪ Set TIN Notice Date to the Application Control Date ▪ Set TIN Notice Counter to "1" (i.e. 1st Notice)
1B	TINs that are already subject to withholding (or are pending for withholding) and that may have appeared on a file for a different Tax Year, but the notice counter is blank.	Record appears on the 1099 File and is matched to 1099I record **AND** 1099 Backup Withholding Status is "Pending" or "Eligible" **AND** TIN Notice Counter is "blank" **AND** IRS Notice Tax Year is "blank" **or** IRS Notice Tax Year is NOT "blank" and is <u>less than</u> the Tax Year on the Payer Record "A" of the IRS file. <u>Statistics Updated:</u> 'Total number of 1099I records selected for initial printing 1st B Notice due to IRS File match'; 'Total number of 1099I records bypassed for printing 1st B Notice because notice was printed in the same Tax Year'.	Only selected if "Use IRS File" is Yes.	▪ Populate IRS Notice Tax Year from the Tax Year on the Payer Record "A" of the IRS file ▪ Populate the TIN Status from BWH TIN Status of respective IRS record ▪ Set TIN Notice Date to the Application Control Date ▪ Set TIN Notice Counter to "1" (i.e. 1st Notice) **Note:** for TINs updated in this group, the Start and Stop Reason & Date fields will not be updated to avoid overwriting of a manually-set value set previously by a user.

	Group Description	Selection Criteria	Group depends on "Use IRS File" parameter	Updates to the 1099I table when the Operation Mode is set to "update
		'Total number of 1099I records with Pending Backup Withholding'		
1C	TINs that are subject to withholding (or are pending for withholding) and have never been issued a notice, but for which someone has manually selected the "Issue 1st Notice" flag.	TIN Notice Counter is "blank" **and** Issue 1st Notice flag is checked **OR** TIN Notice Counter is NOT "blank" **and** Issue 1st Notice flag is checked **and** IRS Notice Tax Year is not "blank" **and** is less than the *IRS Notice Tax Year* parameter. Statistics Updated: 'Total number of 1099I records selected for initial printing 1st B Notice due to manual request'; 'Total number of 1099I records bypassed for printing 1st B Notice because notice was printed in the same Tax Year'. 'Total number of 1099I records with Pending Backup Withholding'	No – execute selection always.	▪ Set Issue 1ST Notice to "false" ▪ Set TIN Notice Date to the Application Control Date ▪ Set TIN Notice Counter to "1" (i.e. 1st Notice) ▪ Populate IRS Notice Tax Year with the IRS Notice Tax Year Parameter. ▪ Set the 1099 Backup Withholding Status to "Pending" if the Status was "blank".

The **2nd B Notice** will need to be printed for these groups:

	Group Description	Selection Criteria	Group depends on "Use IRS File" parameter	Updates to the 1099I table when the Operation Mode is set to "update

	Group Description	Selection Criteria	Group depends on "Use IRS File" parameter	Updates to the 1099I table when the Operation Mode is set to "update
2A	TINs that are not subject to withholding and that have appeared on a file for a different Tax Year, so the notice counter is set to "1".	1099 Backup Withholding Status is "blank" AND TIN Notice Counter is "1" AND IRS Notice Tax Year is "blank" or IRS Notice Tax Year is NOT "blank" and is less than the Tax Year on the Payer Record "A" of the IRS file AND *Print Only One Notice Per Year* job parameter is "N" or *Print Only One Notice Per Year* job parameter is "Y" and Calendar Years of the TIN Notice Date and Application Control Date are not the same Statistics Updated: - 'Total number of 1099I records selected for initial printing 2nd B Notice due to IRS File match'; - 'Total number of 1099I records bypassed for printing 2nd B Notice because notice was printed in the same Calendar Year'; - 'Total number of 1099I records bypassed for printing 2nd B Notice because notice was printed in the same Tax Year'; 'Total number of 1099I records with Pending Backup Withholding'	Only selected if "Use IRS File" is Yes.	▪ Set 1099 Backup Withholding Status to "Pending" ▪ Populate IRS Notice Tax Year from the Tax Year on the Payer Record "A" of the IRS file ▪ Populate the TIN Status from BWH TIN Status of respective IRS record ▪ Set TIN Notice Date to the Application Control Date ▪ Set TIN Notice Counter to "2" (i.e. 2nd Notice) ▪ Set Withholding Tracking Start Date to blank ▪ Set Withholding Tracking Stop Date to blank ▪ Set Start 1099 Backup Withholding Reason to "CP2100" ▪ Set Stop 1099 Backup Withholding Reason to blank
2B	TINs that are subject to withholding (or are pending for withholding) and have been issued a 1st B Notice.	1099 Backup Withholding Status is "Pending" or "Eligible" AND TIN Notice Counter is "1" AND IRS Notice Tax Year is "blank" or	Only selected if "Use IRS File" is Yes.	▪ Populate IRS Notice Tax Year from the Tax Year on the Payer Record "A" of the IRS file ▪ Populate the TIN Status from BWH TIN Status of

	Group Description	Selection Criteria	Group depends on "Use IRS File" parameter	Updates to the 1099I table when the Operation Mode is set to "update
		IRS Notice Tax Year is NOT "blank" and is <u>less than</u> the Tax Year on the Payer Record "A" of the IRS file AND *Print Only One Notice Per Year* job parameter is "N" or *Print Only One Notice Per Year* job parameter is "Y" AND Calendar Years of the TIN Notice Date and Application Control Date are not the same <u>Statistics Updated:</u> - 'Total number of 1099I records selected for initial printing 2nd B Notice due to IRS File match'; - 'Total number of 1099I records bypassed for printing 2nd B Notice because notice was printed in the same Calendar Year'; - 'Total number of 1099I records bypassed for printing 2nd B Notice because notice was printed in the same Tax Year'; 'Total number of 1099I records with Pending Backup Withholding'		respective IRS record ▪ Set TIN Notice Date to the Application Control Date ▪ Set TIN Notice Counter to "2" (i.e. 2nd Notice)
2C	TINs that subject to withholding (or are pending for withholding) and have been issued a 1st B Notice, but for which someone has manually selected the "Issue 2nd B Notice" flag.	Issue 2nd Notice is checked AND *Print Only One Notice Per Year* job parameter is "N" or *Print Only One Notice Per Year* job parameter is "Y" and Calendar Years of the TIN Notice Date and Application Control Date are not the same. AND IRS Notice Tax Year is "blank" or	No – execute selection always.	▪ Set Issue 2nd Notice to "false" ▪ Set TIN Notice Date to the Application Control Date ▪ Set TIN Notice Counter to "2" (i.e. 2nd Notice) ▪ Populate IRS Notice Tax Year with the IRS Notice Tax Year Parameter

	Group Description	Selection Criteria	Group depends on "Use IRS File" parameter	Updates to the 1099I table when the Operation Mode is set to "update
		IRS Notice Tax Year is not "blank" and is less than the *IRS Notice Tax Year* parameter. Statistics Updated: - 'Total number of 1099I records selected for initial printing 2nd B Notice due to manual request'; - 'Total number of 1099I records bypassed for printing 2nd B Notice because notice was printed in the same Calendar Year'; - 'Total number of 1099I records bypassed for printing 2nd B Notice because notice was printed in the same Tax Year'; 'Total number of 1099I records with Pending Backup Withholding'		

The process will insert a new record into the **BWNPH table** for each record that was selected from 1099I table to print the **First** OR **Second B Notice**. For each of these groups (1A, 1B, 1C, 2A, 2B, and 2C), the field values may be different.

Regardless of the setting in the *Use IRS File* job parameter, the process reads the **Backup Withholding Notice Printing History table** and identifies the following sets of conditions on the table to select records, for which:

Re-printing of the **B Notices** has been scheduled for these groups**:**

	Group Description	Selection Criteria for the 2 groups that require re-printing of B Notices	Group depends on "Use IRS File" parameter	Updates to the BWNPH table when the Operation Mode is set to "update
3	TINs that are scheduled for re-printing of the 1st B Notice	Notice Printing Status is "Replacement Pending" **AND** Requested Notice Reprint Date is less than or equal to the Application Control	No – execute selection always.	• Set Notice Printing Status to "Printed" • Set Requested Notice Reprint Date to "blank" • Set Last Notice Reprint Date to the Application Control

		Date <u>Statistics updated:</u> 'The total number of BWNPH records selected for reprinting 1st B Notice';		Date.
4	TINs that are scheduled for re-printing of the 2nd B Notice	Notice Printing Status is "Replacement Pending" **AND** Requested Notice Reprint Date is less than or equal to the Application Control Date <u>Statistics updated:</u> 'The total number of BWNPH records selected for reprinting 2nd B Notice';	No – execute selection always.	• Set Notice Printing Status to "Printed" • Set Requested Notice Reprint Date to "blank" • Set Last Notice Reprint Date to the Application Control Date.

2.1.43 Print Payment Hold Notice

Chain or Job Name	Print Payment Hold Notice
Recommended Frequency	Nightly
Single Instance Required	Yes
Can be restarted?	No
Reports generated	Payment Hold Notice Code Exception Report Address ID Exception Report

Overview

The Print Payment Hold Notice batch process prints Notice Letters for applicable held payment requests on the Payment Hold Activity table or for requested held payments on the Print Payment Hold Notice table. This process runs after executing the Automated Payment Hold batch process.

The Print Payment Hold Notice batch job first inserts valid records from the Payment Hold Activity table (R_PYMT_HLD_ACT) into the Print Payment Hold Notice History table (R_PRNT_PYMT_HLD_NTC). It will select records on the Print Payment Hold Notice History table that meet the following criteria:

1. Print Payment Hold Notice Status is set to *Ready for Reprint* and Scheduled Print Date is less than or equal to Application Control Date. OR
2. Print Payment Hold Notice Status is set to *Ready for Print*.

Then the Print Payment Hold Notice batch job creates an xml file for all the notices to be printed. Every notice to be printed goes in XML file for Form Printing and eventually a dat file is generated for printing. Two exception reports are also created listing the invalid or inactive Address ID and the other report lists the invalid Notice Code.

Process Steps	Messages
1. Validate the Parameters	• Validating Batch Parameters • Parameters are valid or invalid depending on the Validation. If the parameter is invalid, the invalid value is displayed in the log. • Batch Parameter validation completed
2. Selection and Grouping of records	• Group all the eligible records from the 'Payment Hold Activity' table to get a single record per group in the 'Print Payment Hold Notice' table. If no eligible records are found on the 'Payment Hold Activity' table then the process will be terminated with a return code of *Warning*. • If the last run of the batch job was not completed properly then none of the

Process Steps	Messages
	tables are updated and the next run will start with the first step again. Hence, at this step we only need to look at the activity table and see if there are any eligible records and if not we can stop the process.
3. Validate Address ID and Payment Hold Notice Codes / Generate Reports	• Records inserted into the Print Payment Hold Notice History table the Address ID and Payment Hold Notice Code are validated. If not valid the invalid Address ID records are written into the 'Address ID Exception Report' and invalid Payment Hold Notice Code record are written into the 'Payment Hold Notice Code Exception Report'. If the record is valid then it is inserted into the 'Payment Hold Activity' table.
4. Populate Reference Print Payment Notice Number	• This is the common connection between the two tables Payment Hold Activity table and Print Payment Hold Notice History table. This is an auto generated number (an entry will be made in R_IN_UNID which will store the last used number) and once it is generated for a record in Print Payment Hold Notice History table, the same number will be assigned to the respective records on the Payment Hold Activity table.
5. Retrieve record from the Payment Hold Activity table / Generate the Payment Hold Notice Letters	• Based on a Print Details flag on Payment Hold Notice Text table, the notice letters with the details lines or without the detailed line information. If the Print Details flag is set to *Yes* then all the associated records (same reference print payment notice number) on the Payment Hold Activity table will be grouped together and each grouped record will be printed as the detailed line on the notice letter. If the Print Detail flag is set to *No* on the Payment Hold Notice Text table then none of the details will be printed. BIRT Server would be used to generate the letter.

Restartability

Job cannot be restarted. A new job should be scheduled every time.

Major Input

Data from the following tables:

- Print Payment Hold Notice Table (R_PRNT_PYMT_HLD_NTC)
- Payment Hold Activity Table (R_PYMT_HLD_ACT)
- Payment Hold Notice Text Table (R_PYMT_HLD_NTC_TXT)

Batch Parameters

Parameter	Description	Default Value
Application Resource ID APPL_RSRC_ID	Required. The entered application resource ID should be a valid Application resource.	
Print Job Name PRINT_JOB_NAME	Required.	
Print Resource Identifier PRINT_RSRC	Required. The entered print resource should be a valid resource on Print resource table.	

Major Output

Following are the output of the batch job:

- Address ID Exception Report.
- Payment Hold Notice Code Exception Report.
- Hold Notice Letters.

Job Return Code

Return Code	Condition
Successful (1)	All the selected notice letters are printed successfully.
Warning (4)	No eligible records found. • No eligible records found in Payment Hold Activity table for printing Payment Hold Notice Letter
Non-Fatal Error (8)	We are not setting this return code.
Failed (12)	The job fails under the following conditions: • Print Payment Hold Notices flag on the Payment Hold Options table is NOT set to *Yes*.

Return Code	Condition
	• Run time exceptions for unexpected situations. • Restart failed because another instance of the process has already been run successfully
Terminated (16)	This return code will be issued when the job is terminated by the user.
System Failure (20)	This return code is issued when the job is terminated because of database server or network issues.

Sort Criteria

The process shall group selected records on the Payment Hold Activity table where the Vendor Address ID is NOT blank by the Payment Hold Notice Code, Vendor and Address ID to insert a single record to the Print Payment Hold Notice table for each grouped set of records with the same Payment Hold Notice Code, Vendor, and Address ID.

The process shall group selected records on the Payment Hold Activity table where the Vendor Address ID is blank by the Payment Hold Notice Code, Vendor Code, Department and Hold Type to insert a single record to the Print Payment Hold Notice table for each grouped set of records with the same Payment Hold Notice Code, Vendor, Department, and Hold Type.

Selection Criteria

The process shall select all the records from the Payment Hold Activity table where the **Eligible for Print** flag is set to *Yes*, the **Reference Printed Payment Notice Number** is *Blank* and the **Hold Removal Date** is *Blank*.

Problem Resolution

Step1: Validate Parameters

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	N/A	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	We are not setting this return code.		
Failed (12)	Job failed due to Fatal conditions	In this step, the job can fail under the following three conditions. 1) Encounters any runtime exceptions and	N/A

Possible Return Codes	Condition	Recommendation	Other Instructions
		2) Failed during restart. 3) Parameter validation failed. 4) **Print Payment Hold Notices** flag on the Payment Hold Options table is NOT set to *Yes*. If the job fails because of the runtime exceptions, investigate the Logger file, resolve the error and restart the job.	
	Failed while restarting the job since another instance of the job has already been run successfully.	**Recommendation:** Schedule a new job.	
Terminated (16)	Job is terminated manually by the user.	Reason for the termination needs to be investigated. The job can either be restarted or schedule a new job.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.
System Failure (20)	When the job is terminated because of database server or network issues	Reason for the System Failure needs to be investigated. The job can either be restarted or schedule a new job.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.

Step 2: Selection and Grouping of records

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	N/A	N/A	N/A

Possible Return Codes	Condition	Recommendation	Other Instructions
Warning (4)	Job ended with a Warning because there are no eligible records found in Payment Hold Activity table. **Sample Message:** No eligible records found in Payment Hold Activity table for printing Payment Hold Notice Letter	Try to run after executing the Automated Payment Hold batch process.	N/A
Non-Fatal Error (8)	We are not setting this return code.		
Failed (12)	Job failed due to Fatal conditions	In this step, the job can fail under the following three conditions. 1) Encounters any runtime exceptions and 2) Failed during restart. If the job fails because of the runtime exceptions, investigate the Logger file, resolve the error and restart the job.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.
	Failed while restarting the job since another instance of the job has already been run successfully.	**Recommendation:** Schedule a new job.	
Terminated (16)	Job is terminated manually by the user.	Reason for the termination needs to be investigated. The job can either be restarted or schedule a new job.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.

Possible Return Codes	Condition	Recommendation	Other Instructions
System Failure (20)	When the job is terminated because of database server or network issues	Reason for the System Failure needs to be investigated. The job can either be restarted or schedule a new job.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.

Step 3: Validate Address ID and Payment Hold Notice Codes / Generate Reports

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	N/A	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	We are not setting this return code.		
Failed (12)	Job failed due to Fatal conditions	In this step, the job can fail under the following three conditions. 1) Encounters any runtime exceptions and 2) Failed during restart. If the job fails because of the runtime exceptions, investigate the Logger file, resolve the error and restart the job.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.
	Failed while restarting the job since another instance of the job has already been run successfully.	**Recommendation:** Schedule a new job.	
Terminated (16)	Job is terminated manually by the user.	Reason for the termination needs to be investigated. The job can either be restarted	If another instance of the job has already been scheduled and ran successfully,

Possible Return Codes	Condition	Recommendation	Other Instructions
		or schedule a new job.	then this job should not be restarted – only new job should be scheduled.
System Failure (20)	When the job is terminated because of database server or network issues	Reason for the System Failure needs to be investigated. The job can either be restarted or schedule a new job.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.

Step 4: Populate Reference Print Payment Notice Number

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	N/A	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	We are not setting this return code.		
Failed (12)	Job failed due to Fatal conditions	In this step, the job can fail under the following three conditions. 1) Encounters any runtime exceptions and 2) Failed during restart. If the job fails because of the runtime exceptions, investigate the Logger file, resolve the error and restart the job.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.
	Failed while restarting the job since another instance of the job has already been run successfully.	**Recommendation:** Schedule a new job.	

Possible Return Codes	Condition	Recommendation	Other Instructions
Terminated (16)	Job is terminated manually by the user.	Reason for the termination needs to be investigated. The job can either be restarted or schedule a new job.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.
System Failure (20)	When the job is terminated because of database server or network issues	Reason for the System Failure needs to be investigated. The job can either be restarted or schedule a new job.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.

Step 5: Retrieve record from the Payment Hold Activity table / Generate the Payment Hold Notice Letters

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	N/A	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	We are not setting this return code.		
Failed (12)	Job failed due to Fatal conditions	In this step, the job can fail under the following three conditions. 1) Encounters any runtime exceptions and 2) Failed during restart. If the job fails because of the runtime exceptions, investigate the Logger file, resolve the error and restart the job.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.

Possible Return Codes	Condition	Recommendation	Other Instructions
	Failed while restarting the job since another instance of the job has already been run successfully.	**Recommendation:** Schedule a new job.	
Terminated (16)	Job is terminated manually by the user.	Reason for the termination needs to be investigated. The job can either be restarted or schedule a new job.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.
System Failure (20)	When the job is terminated because of database server or network issues	Reason for the System Failure needs to be investigated. The job can either be restarted or schedule a new job.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.

2.1.44 Retainage Payout

Description

Retainage amount can be paid out or forfeited over the life of the award or at the completion of the award depending on the terms of the award. There are six possible Payout methods: Payout to Vendor, Payout to Third Party, Collateral Substitution, Forfeiture to Expenditure, Forfeiture to Revenue, Forfeiture to Trust and Agency.

Retainage is generally paid out or forfeited using the First-In-First-Out (FIFO) method for transactions that took the retainage. However, forfeiture rules can vary depending on the APPCTRL setting for the Retainage Forfeiture Option (RET_FORF_OPT) field. For each forfeiture method, the user may specify on Retainage Fund Control (RTGFC) if funds may be forfeited from the current BFY or the prior BFY, or both. If the fund rules are not established for every forfeiture method or if the Fund has no rules for any forfeiture method, then the system will use the FIFO method to forfeit retainage. If the Retainage Forfeiture Option on APPCTRL is set to 2, the RTGFC rules will be read by the Retainage Summary by Commodity Line (RTGSUM) table on save when retainage forfeiture is setup.

The Retainage Payout process creates the Retainage Payout/Forfeiture (RTGPF) transactions as defined by the Transaction Code batch parameter for all of the payout methods. The RTGPF transaction is created for each record selected from the RTGSUM page.

For each record selected on RTGSUM, one accounting line or two accounting lines may be created for it on the RTGPF transaction. One accounting line will be created when the following two conditions are true:

- The AFY of the RTGPF transaction is the same as the FY of the transaction that withheld retainage.
- The Fund on the RTGPF transaction is the same as the Fund on the transaction that withheld retainage.

Two accounting lines will be created for each record selected from RTGSUM if either of the following two conditions is true:

- If the AFY of the RTGPF transaction is after the BFY of the transaction that withheld retainage.
- If the Fund on the RTGPF transaction is different than the Fund of the transaction that withheld retainage.

When to Run

On demand

Major Input

Retainage Summary by Commodity Line (R_RTG_SUM)

Retainage Detail (R_RTG_DET)

Retainage Fund Control (R_RTG_FC)

Posting Line Catalog (PSTNG_LN_CAT)

Output

- RTGPF transaction

- Updates to the Retainage Summary by Commodity Line (RTGSUM), Retainage Detail (RTGDET) table and the Retainage Released/Forfeited (RTGPAID) table.

Parameters

The following parameters are required for the Retainage Payout process.

Job	Parameter	Description	Default Value
Retainage Payout	Transaction Code (DOC_CD)	Required. Refers to the Transaction Code that will be generated by the Retainage Payout process.	RTGPF
	Department Code (DOC_DEPT_CD)	Conditionally required, if the Use Award Doc Department parameter is set to *No*. Defines the transaction department to be used with the transaction code and prefix to find an Automatic Transaction Numbering entry. It will appear as the transaction department for all transactions created by the program.	No Default
	Unit Code (DOC_UNIT_CD)	Optional. Provides security with a Unit Code enabling transaction access to be secured at a level below department.	No Default
	Event Type ID to Substitute Retainage in Lieu of Collateral (COLL_EVNT_TYP_ID)	Required. Defines the Event Type ID to be used, when the payout method selected is Collateral Substitution.	AP03

Job	Parameter	Description	Default Value
	Event Type ID for Retainage Payout to Vendor or Third Party (EVNT_TYP_ID)	Required. Defines the Event Type to be used, when the Payout Method is Return to Vendor or Third Party.	AP03
	Event Type ID for Retainage Forfeiture (EXP_EVNT_TYP_ID)	Required. Defines the Event Type ID to be used, when the Payout Method selected is Forfeiture to Revenue.	AP14
	Event Type ID to Forfeit Retainage to Expenditure (FRFT_EXP_EVNT_TYP_ID)	Required. Defines the Event Type ID to be used, when the Payout Method selected is Forfeiture to Expenditure.	AP14
	Event Type ID to Forfeit Retainage to Expenditure – RTGPF AFY > Disb BFY (FRFT_EXP_EVNT_TYP_ID_1)	Required. Defines the Event Type ID to be used, when Payout Method is Forfeiture to Expenditure and the AFY of the output transaction is greater than the BFY of the transaction that took the retainage.	AP14
	Event Type ID to Forfeit Retainage to Revenue (FRFT_REV_EVNT_TYP_ID)	Required. Defines the Event Type ID to be used, when the Payout Method selected is Forfeiture to Revenue.	AP21

Job	Parameter	Description	Default Value
	Event Type ID to Forfeit Retainage to Revenue – RTGPF AFY > Disb BFY (FRFT_REV_EVNT_TYP_ID_1)	Required. Defines the Event Type ID to be used, when the Payout Method is Forfeiture to Revenue and the AFY of the output transaction is greater than the BFY of the transaction that took the retainage.	AP22
	Event Type ID to Forfeit Retainage to Revenue – new Fund (FRFT_REV_FUND_EVNT_TYP_ID)	Required. Defines the Event Type ID to be used for the second accounting line needed to apply the forfeited retainage to revenue.	AP23
	Forfeit to Trust and Agency (FRFT_TAF_EVNT_TYP_ID)	Required. Defines the Event Type ID to be used, when the Payout Method selected is Forfeiture to Trust and Agency Fund.	AP22
	Event Type ID to Forfeit Retainage to Trust and Agency Fund – RTGPF AFY > Disb BFY (FRFT_TAF_EVNT_TYP_ID_1)	Required. Defines the Event Type ID to be used, when Payout Method is Forfeiture to Trust and Agency Fund and the AFY of the output transaction is greater than the BFY of the transaction that took the retainage.	AP22

Job	Parameter	Description	Default Value
	Event Type ID to Forfeit Retainage to Trust and Agency Fund – new Fund (FRFT_TAF_FUND_EVNT_TYP_ID)	Required. Defines the Event Type ID to be used for the second accounting line needed to apply the forfeited retainage to the trust and agency fund.	AP23
	Accounting Template ID to Forfeit Retainage to Revenue (FRFT_REV_ACC_TEM_ID)	Required. Defines the Accounting Template ID to be used, when the Payout Method selected is Forfeiture to Revenue.	
	Accounting Template ID to Forfeit Retainage to Trust and Agency Fund (FRFT_TAF_ACC_TEM_ID)	Required. Defines the Accounting Template ID to be used, when the Payout Method selected is Forfeiture to Trust and Agency Fund.	
	PreFix (PFX)	Optional. Defines the transaction prefix to be used with a Transaction Code and Transaction Department to find an Automatic Transaction Numbering entry. It will appear in the beginning of all Transaction ID's for all transactions created by the program.	No Default

Job	Parameter	Description	Default Value
	Event Type ID for Retainage Forfeiture to be used when the AFY of the payout transaction is greater than the BFY of the transaction that took Retainage (UNDT_EVNT_TYP_ID)	Required. Defines the Event Type to be used, when Payout Method is Retainage Forfeiture and the AFY of payout transaction is greater than the BFY of the transaction that took Retainage.	AP15
	Export Location at Retainage Payout Process Job (AMSEXPORT)	Required (** **Refer to Note**: Assumptions for SWBP on page no. 4)	-
	Parameter Location at Retainage Payout Process Job (AMSPARM)	Required (** **Refer to Note**: Assumptions for SWBP on page no. 4)	-
	Use Award Doc Department (USE_AWD_DOC_DEPT)	Optional. Valid values are *Yes* and *No*. If *Yes*, the Award Department value from RTGSUM is used instead of the specific Department Code batch parameter to populate the output transaction.	No
Transaction Upload	Action Code (ACTN_CD)	Required (and protected) action code which instructs the program to perform a certain action on its records.	171

Job	Parameter	Description	Default Value
	Commit Block Size (COMMIT_BLOCK)	Required. Controls how many records are committed by the application at one time. The size should be compatible with technical capabilities and performance guidelines.	10
	File Name To Be Imported (FILE_NM)	Required (and protected) file that defines what file will be uploaded.	$$AMSIMPORT$$ /RetainagePR.xml
Submit Retainage	Exception Report File (EXCEP_REP_FILE_ NM)	Required. Defines what field the program is to use to create an exception report.	$$AMSLOGS$$/RetainagePayoutExep.txt
	Parameter File (PARM_FILE)	Required (and protected) field that will be used to define the parameters for submitting the GAX transactions.	$$AMSPARM$$/SubmitRtgParm.txt

Sort Sequence

- Sort the Retainage Summary by Commodity Line (RTGSUM) records by the Vendor Customer Code.
- Sort the Retainage Detail records by the Auto Sequence No.

Selection Criteria

Select the awards for which the Scheduled Payment Date is entered where:

- Requested Retainage Release/Forfeiture Amount > ‘0.00’ and
- Payout Method has been entered in the Retainage Summary by Commodity Line (RTGSUM) table.

Select all the payment requests from the Retainage Detail table that refer to the award transactions selected from the Retainage Summary by Commodity Line table depending on the

option selected for Retainage Forfeiture on APPCTRL that is, either based on the F-I-F-O (First-In-First-Out) method or based on the Retainage Fund Control (RTGFC) table rule method.

NOTES

In order to run the Retainage Payout process successfully, complete the following steps:

1. Submit a purchase order with Retainage Terms entered on the commodity line.
2. Submit a payment request (PRC) referencing the purchase order. The PRC should calculate retainage amount based on retainage terms of the order.
3. Run the AD Chain Process. The Retainage Withheld Amount should be updated.
4. Go to the Retainage Summary by Commodity Line (RTGSUM) table and enter a Payout Method, a Requested Retainage Release/Forfeiture Amount, and a Scheduled Payment Date for each retainage record you wish to pay/forfeit.
5. Run the Retainage Payout process with preferred batch parameters.

Problem Resolution

No database restore is required.

This batch program produces new Advantage transactions, which are subject to the line limit functionality constraints. Sites should ensure that they run this job with parameters set to ensure that the created transactions are within the line limit controls.

2.1.45 Retiree Billing

Chain Name	Retiree Billing
Recommended Frequency	Daily as part of the nightly cycle or on demand
Single Instance Required	Yes
Can be restarted?	N/A
Reports generated	N/A

Overview

The Retiree Billing chain process identifies payments made to retirees from the 1099 Journal (J1099), calculates three types of fees, and creates Internal Exchange Transactions (IET) to record the fees associated with those payments. The process creates a file at the end with summary level payment data by TIN.

This chain has the following jobs:

- RETIRPAY XML Generation
- Load to RETIRPAY Table
- IET Transaction XML Generation
- IET Upload
- IET Submit
- RETIRPAY Summary

Note: Even though the above jobs in the chain can be run individually by disabling other jobs, it is recommended to always run the entire chain.

Major Input

- 1099 Reporting Journal (J1099 / JRNL_1099)
- Retiree TIN (RETIRTIN / R_RETIRE_TIN)
- Object (OBJ / R_OBJ)
- Retiree Payments (RETIRPAY / R_RETIRE_PYMTS)
- Vendor/Customer (VCUST / R_VEND_CUST)

Major Output

- Internal Exchange Transactions
- RETIRPAY Summary CSV Output File

Chain Return Codes

When an active job step in the chain ends with a Return Code of anything other than successful, any remaining job steps have a Run Status of *Inactive.* Please see documentation on individual job steps for possible Return Codes for those steps, as *Warning* and *Non-Fatal Error* are not always possible outcomes.

Problem Resolution

See the Problem Resolution sections of each step of the chain job to determine resolution.

Retiree Billing Chain: RETIRPAY XML Generation Job

Job Name	RETIRPAY XML Generation
Recommended Frequency	N/A
Single Instance Required	N/A
Can be restarted?	Yes
Reports generated	None

Overview

This process is used to produce a RetireePayLoad.xml file that contains records selected from J1099. The J1099 records are selected if all these conditions are true:

- The TIN/TIN Type of the J1099 record is found on RETIRTIN.
- The TIN/TIN Type is used on VCUST.
- Transaction Codes are AD, EFT, MD or Transaction Types are CR or JV.
    - DOC_CD equal 'AD', include posting pairs that impact Warrants Payable (posting code 'D007') for Disbursements (Posting Pair A), Retainage (Posting Pair C), Interest/Penalty (Posting Pair F), and Discounts (Posting Pair H)
    - DOC_CD equal 'EFT' or 'MD', include posting pairs that impact Cash (posting code 'A001') for Disbursements (Posting Pair A), Retainage (Posting Pair C), Interest/Penalty (Posting Pair F), and Discounts (Posting Pair H)
    - DOC_TYP equal 'CR' and 'JV', include posting pairs that impact Cash (posting code 'A001') only.
- Batch parameters values also specify additional selection criteria.

If a record is found that met the above criteria, the record is added to the RETIRPAY xml file. A parameter file, DateParam.txt, is also produced that is reused for the final step of the Retiree Billing chain job.

Process Steps	Messages
1. Parameter Validation	• Validating Batch Parameters • Parameters are valid or invalid depending on the validation.

Process Steps	Messages
	If the parameter is invalid, the invalid value is displayed in the log. • Batch Parameter validation completed
2. Selection of Records	• Selecting eligible J1099 records • If the selection returns 0 records, then the following message is issued: “No eligible J1099 records found”. • Else, the number of records processed is displayed at the end.
3. XML Output	• Selected J1099 record contents are checked for the values of the following fields: Transaction Type (DOC_TYP), Transaction Code (DOC_CD), Transaction Department (DOC_DEPT_CD), Transaction ID (DOC_ID), and Vendor/Customer Code (VEND_CUST_CD). • The only J1099 records allowed to be processed are those of DOC_TYP values of JV / CR and DOC_CD values of AD / EFT / MD. • Load the VCUST record from the selected J1099 record and get the values from the following fields: TIN (TIN), TIN Type (TIN_TYP). • Search for the RETIRTIN record referenced on the selected VCUST record by TIN and TIN_TYP. • If a RETIRTIN record is found, for all fields on the RETIRTIN record that also exist on a RETIRPAY record (except for RETIRE_ER_CONTR_AM / RETIRE_AED_AM / RETIRE_SAED_AM), copy the field values from RETIRTIN to the new RETIRPAY record and set the TIN and TIN_TYP values accordingly. • If a RETIRTIN record is not found, then search again but by Headquarters TIN (PNT_TIN) and Headquarters TIN Type (PNT_TIN_TYP). • If a RETIRTIN record is found, for all fields on the RETIRTIN record that also exist on a RETIRPAY record (except for RETIRE_ER_CONTR_AM / RETIRE_AED_AM / RETIRE_SAED_AM), copy the field values from RETIRTIN to the new RETIRPAY record and set the PNT_TIN and PNT_TIN_TYP values accordingly. • If a RETIRTIN record is still not found, then do nothing.
4. Parameter File	• Create a parameter file that contains the values of the Beginning Date (BGN_DT), Ending Date (END_DT), and Processed Date (PROC_DT) that is later used by the final step of the chain job only if the parameter validation step did not fail.

Major Input

1099 Reporting Journal (J1099 / JRNL_1099)

Retiree TIN (RETIRTIN / R_RETIRE_TIN)

Object (OBJ / R_OBJ)

Batch Parameters

The following are the delivered parameter values which may have been updated through Batch Setup to meet local needs.

Parameter	Description	Default Value
Export/Import Location (AMSEXPORT)	Required ExportImport location	$$AMSROOT$$/ExportImport
Parameter Location (AMSPARM)	Required Parameter location	$$AMSROOT$$/Parms
RETIRPAY XML Output File (XML_FILE)	Required name of the xml of payment transactions.	RetireePayLoad.xml
Date Parameters TXT Output File (DATE_PARAM)	Required name of the date parameter txt file.	DateParam.txt
J1099 Beginning Date (BGN_DT)	Required beginning date for J1099 records selection (mm/dd/yyyy).	(No Default)
J1099 Ending Date (END_DT)	Required ending date for J1099 records selection (mm/dd/yyyy).	(No Default)
J1099 Posting Code (PSTNG_CD_ID)	Required selection and output posting code for stale payable disbursement account.	D007
J1099 Cash Posting Code (CASH_POST_CD)	Required cash posting code for cash debits and credits.	A001
OBJ Income Type (INC_TYP_NO)	Required 1099 Type of Income comma-separated list of box numbers.	(No Default)
OBJ Tax Form Type (FORM_TYP_IND)	Required 1099 Form Type denoted by any but only one of the following characters: G: 1099-G I: 1099-INT	(No Default)

	M: 1099-MISC S: 1099-S A: 1099-A C: 1099-C E: 1099-NEC R: 1099-R	
Commit Block Size (COMMIT_BLOCK)	Commit Block Size. If blank or invalid, defaults to 100.	100
Select Block Size (SELECT_BLOCK)	Select Block Size. If blank or invalid, defaults to 1000.	1000
Progression Counter Size (PROG_CTR_SZ)	Progression Counter Size. If blank or invalid, defaults to 5000.	5000

Major Output

- RETIRPAY XML Output File (XML_FILE): RetireePayLoad.xml
  - Date Parameters TXT Output File (DATE_PARAM): DateParam.txt

Sort Criteria

None

Selection Criteria

From the 1099 Journal (J1099 / JRNL_1099):

- DOC_REC_DT that is within the J1099 Beginning Date and End date from the batch parameter.
- DOC_TYP of JV or CR
- OBJ_CD that are specified with Form Type (FORM_TYP_IND) and Income Type (INC_TYP_NO) in the batch parameter
- DOC_CD = AD, MD, EFT
- PSTNG_CD_ID
  - For DOC_CD equal ‘AD’, include posting pairs that impact Warrants Payable (posting code ‘D007’) for Disbursements (Posting Pair A), Retainage (Posting Pair C), Interest/Penalty (Posting Pair F), and Discounts (Posting Pair H)
  - For DOC_CD equal ‘EFT’ or ‘MD’, include posting pairs that impact Cash (posting code ‘A001’) for Disbursements (Posting Pair A), Retainage (Posting Pair C), Interest/Penalty (Posting Pair F), and Discounts (Posting Pair H)
  - For DOC_TYP equal ‘CR’ and ‘JV’, include posting pairs that impact Cash (posting code ‘A001’) only.
  - For DOC_CD of AD, MD, EFT, PSTNG_PR_TYP = A, C, F, H

- Retiree TIN (RETIRTIN / R_RETIRE_TIN)
  - The process selects TIN/TIN_TYP on RETIRTIN using the TIN found in J1099

Problem Resolution

The following table shows the potential job return codes for this job.

Return Code	Condition
Successful (1)	The job ends with this status when parameter validation is successful and at least one record was selected and processed.
Warning (4)	A warning results when no eligible J1099 records matched the selection criteria or no eligible RETIRTIN records were found from any of the eligible J1099 records that were checked.
Non-Fatal Error (8)	This job does not end with this return code.
Failed (12)	The job fails under the following conditions: • Parameters are invalid • Selection query execution failed • Commit of the transaction(s) to output failed • Run time exceptions for unexpected situations When this job fails, subsequent jobs in the chain are set to *Inactive*.
Terminated (16)	User manually terminates job.
System Failure (20)	Application is bounced while job is running.

Problem Resolution

Step 1: Parameter Validation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the parameters are validated successfully.	N/A	N/A
Failed (12)	One of the parameters failed validation.	Populate all required parameters with valid values.	N/A

Step 2: Selection of Records

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	At least one eligible J1099 record was	N/A	N/A

Possible Return Codes	Condition	Recommendation	Other Instructions
	selected that is linked to at least one RETIRTIN record and the data from the J1099 is successfully copied into a new RETIRPAY transaction in the XML output file.		
Warning (4)	No eligible J1099 records matched selection criteria	Check the system debug log for RetireePayLoadToXML and the selection query that was executed to observe the search criteria and conditions.	Perform a search on each of the records to confirm that criteria exist.
	No eligible RETIRTIN records found from any of the eligible J1099 records that were checked	Check the system debug log for RetireePayLoadToXML to observe the VCUST record that was loaded from the eligible J1099 record being read that also shows the TIN and TIN Type values.	Manually search out the RETIRTIN records for the TIN and TIN Type values in question to confirm if a matching record actually exists.
Failed (12)	Selection query execution failed	Check the system debug log for RetireePayLoadToXML to observe the technical error message.	Analyze the log error messages to derive a possible root cause to the issue.
	Run time exceptions for unexpected situations	Check the system debug log for RetireePayLoadToXML to observe the technical error message.	Analyze the log error messages to derive a possible root cause to the issue.

Step 3: XML Output

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All eligible J1099 records are successfully copied and saved into a new RETIRPAY file in XML format if an eligible RETIRTIN record was found.	N/A	N/A

Possible Return Codes	Condition	Recommendation	Other Instructions
Failed (12)	Commit of the transaction(s) to output failed	Check the system debug log for RetireePayLoadToXML to observe the error message.	Analyze the log error messages to derive a possible root cause to the issue.
	Run time exceptions for unexpected situations	Check the system debug log for RetireePayLoadToXML to observe the error message.	Analyze the log error messages to derive a possible root cause to the issue.

Step 4: Parameter File

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Parameter file is successfully saved with the BGN_DT, END_DT, and PROC_DT values.	N/A	N/A
Failed (12)	Run time exceptions for unexpected situations	Check the system debug log for RetireePayLoadToXML to observe the error messages.	Analyze the log error messages to derive a possible root cause to the issue.

Retiree Billing Chain: Load to RETIRPAY Table

Job Name	Load to RETIRPAY Table
Recommended Frequency	N/A
Single Instance Required	N/A
Can be restarted?	No
Reports generated	None

Overview

This job is used to process the RetireePayLoad.xml file that is produced in the previous job.

Major Input

- RETIRPAY XML Output File (XML_FILE): RetireePayLoad.xml

Batch Parameters

The following are the delivered parameter values which may have been updated through Batch Setup to meet local needs.

Parameter	Description	Default Value
Action Code (ACTN_CD)	Action Code: Table Import	201
File Name (FILE_NM)	File Name	$$AMSEXPORT$$/RetireePayLoad.xml

Major Output

- Retiree Payments (RETIRPAY / R_RETIRE_PYMTS)

Sort Criteria

None

Selection Criteria

None

Problem Resolution

The following table shows the potential job return codes for this job.

Return Code	Condition
Successful (1)	The job ends as successful when parameter validation is successful and all transactions in the XML file are loaded successfully.
Warning (4)	A warning results when the XML file is empty.
Non-Fatal Error (8)	This job does not end with this return code.
Failed (12)	The job fails under the following conditions: • Parameters are invalid • XML file not found When this job fails, subsequent jobs in the chain are set to *Inactive*.
Terminated (16)	User terminates batch job.
System Failure (20)	Application is bounced while job is running.

Problem Resolution

Step 1: Parameter Validation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the parameters are validated successfully.	N/A	N/A
Failed (12)	One of the parameters failed validation.	A batch parameter can fail if required parameter is not provided or user specified an invalid value.	N/A
	The input XML file not found.	Check that the file path to the input XML file is valid and/or file name is correct.	N/A

Step 2: Load Table

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the transactions were committed to the target table(s) of the database successfully.	N/A	N/A
Warning (4)	No transactions in the input XML file.	Check the input XML file to verify.	Analyze the job logs of the previous step in the chain job.
Failed (12)	Run time exceptions for unexpected situations	Check the system debug log for SysManUtil to observe the error messages.	Analyze the log error messages to derive a possible root cause to the issue.

Retiree Billing Chain: IET Transaction XML Generation Job

Job Name	IET Transaction XML Generation
Recommended Frequency	N/A
Single Instance Required	N/A
Can be restarted?	No
Reports generated	None

Overview

IET Transaction XML Generation generates Internal Exchange Transactions (IET) from RETIRPAY page for reimbursement from departments for the State's portion of the retirement payments. One IET transaction is created for each unique Department Code (DEPT_CD) on the RETIRPAY table. All records are summarized by the TIN and accounting string (Fund, Dept, Unit, Appropriation Unit, Program, Program Period, Phase, Activity, Location) combination.

The process creates three accounting lines for each unique TIN and accounting string combination. The accounting lines records a vendor's/TIN summarized total for each department by Posting Amount (PSTNG_AM) for each of these amounts:

- AL1 records the Employer Contribution Amount (RETIRE_ER_CONTR_AM),
- AL2 records the AED Amount (RETIRE_AED_AM),
- AL3 records the SAED Amount (RETIRE_SAED_AM).

The process will repeat the creation of the three accounting lines for each record in the summarized file. When a new department is encountered, it creates a new IET transaction.

The process produces an IETDOC.xml file. A pair of parameter files IETSubmit.txt and IETException.txt are also produced that is reused is later steps of the Retiree Billing chain process.

Process Steps	Messages
1. Parameter Validation	• Validating Batch Parameters • Parameters are valid or invalid depending on the validation. If the parameter is invalid, the invalid value is displayed in the log. • Application Parameters for this batch job are also set. • Batch Parameter validation completed
2. Selection of Records	• Selecting eligible RETIRPAY records • If the selection returns 0 records, then the following message is issued: "No eligible RETIRPAY records found". • Else the number of records processed is displayed at the end.
3. XML Output	• Selected RETIRPAY record contents is checked for the value of the following field: Department (DEPT_CD). • If this is the first DEPT_CD value being checked or if it is

Process Steps	Messages
	different from the DEPT_CD value of the previous RETIRPAY record checked, only then proceed to create the IET transaction header and vendor/customer components. • Create three IET transaction accounting lines for each unique TIN and accounting string.
4. Parameter File	• Create a parameter file that contains the values of the Exception Report Type (EXCEP_REP_IND), Exception Report File Name (EXCEP_REP_FILE_NM), and the following values for each IET header component created: Transaction Code (DOC_CD), Transaction Department (DOC_DEPT_CD), Transaction ID (DOC_ID), and Transaction Version Number (DOC_VERS_NO) of 1.

Major Input

- Retiree Payments (RETIRPAY / R_RETIRE_PYMTS)

Batch Parameters

The following are the delivered parameter values which may have been updated through Batch Setup to meet local needs.

Parameter	Description	Default Value
Export/Import Location (AMSEXPORT)	Required ExportImport location	$$AMSROOT$$/ExportImport
Parameter Location (AMSPARM)	Required Parameter location	$$AMSROOT$$/Parms
RETIRPAY Beginning Date (BGN_DT)	Required beginning date for RETIRPAY records selection (mm/dd/yyyy).	(No Default)
Commit Block Size (COMMIT_BLOCK)	Commit Block Size. If blank or invalid, defaults to 100.	100
Transaction Code (DOC_CD)	Required transaction code used when creating the transfer transactions.	(No Default)
Transaction Department (DOC_DEPT)	Required department used when creating the transfer transactions.	(No Default)
Transaction Description (DOC_DESC)	Required transaction description used when creating the transfer transactions.	(No Default)

Transaction Prefix (DOC_PREFIX)	Optional transaction code prefix used when creating the transfer transactions.	(No Default)
RETIRPAY Ending Date (END_DT)	Required ending date for RETIRPAY records selection (mm/dd/yyyy).	(No Default)
Exception Report Output File (EXCEP_REP_FILE_NM)	Required name of the Exception Report txt file.	$$AMSLOGS$$/IETException.txt
OBJ Tax Form Type (FORM_TYP_IND)	Required 1099 Form Type denoted by any but only one of the following characters: G: 1099-G I: 1099-INT M: 1099-MISC S: 1099-S A: 1099-A C: 1099-C E: 1099-NEC R: 1099-R	(No Default)
OBJ Income Type (INC_TYP_NO)	Required 1099 Type of Income comma-separated list of box numbers.	(No Default)
Progression Counter Size (PROG_CTR_SZ)	Progression Counter Size. If blank or invalid, defaults to 5000.	5000
Select Block Size (SELECT_BLOCK)	Select Block Size. If blank or invalid, defaults to 1000.	1000
IET Submit TXT Output File (SUBMIT_PARM_FILE)	Required name of the IET submission txt file.	IETSubmit.txt
IET XML Output File (XML_FILE)	Required name of the xml of internal exchange transactions.	IETDOC.xml

Application Parameters

All should be setup on APPCTRL before running this process.

Parameter	Description	Default Value
RETIREE Department (RETIRE_IET_DEPT_CD)	Department code used in first party accounting line for the RETIREE IET transaction	(No Default)
RETIREE Unit (RETIRE_IET_UNIT_CD)	Unit code used in first party accounting line for the RETIREE IET transaction	(No Default)
RETIREE Event Type ID (RETIRE_IET_EVNT_TYP_ID)	Event type ID used in Event Type in the Exchange Details of the RETIREE IET transaction	(No Default)
RETIREE Fund (RETIRE_IET_FUND_CD)	Fund code used in first party accounting line for the RETIREE IET transaction	(No Default)
RETIREE Balance Sheet Account (RETIRE_IET_BSA_CD)	Balance sheet account code used in first party accounting line for the RETIREE IET transaction	(No Default)
RETIREE Employer Contribution Object (RETIRE_ER_CONTR_OBJ_CD)	Object code for the RETIREE ER Contribution	(No Default)
RETIREE AED Object (RETIRE_AED_OBJ_CD)	Object code for the RETIREE AED	(No Default)
RETIREE SAED Object (RETIRE_SAED_OBJ_CD)	Object code for the RETIREE SAED	(No Default)

Major Output

- IET XML Output File (XML_FILE): IETDOC.xml
- IET Submit TXT Output File (SUBMIT_PARM_FILE): IETSubmit.txt
- Exception Report Output File (EXCEP_REP_FILE_NM): IETException.txt

Sort Criteria

- Selected RETIRPAY (R_RETIRE_PYMTS) records are sorted by the following fields:
  - DEPT_CD
  - TIN
  - VEND_CUST_CD
  - LGL_NM
  - FUND_CD
  - SFUND_CD
  - UNIT_CD
  - SUNIT_CD
  - APPR_CD
  - PROG_CD
  - PPC_CD
  - PHASE_CD
  - ACTV_CD
  - SACTV_CD
  - LOC_CD
  - SLOC_CD
  - FUNC_CD
  - SFUNC_CD
  - TASK_CD
  - STASK_CD
  - TASK_ORD_CD

- RPT_CD
- SOBJ_CD
- DOBJ_CD
- BSA_CD

Selection Criteria

- Records that are within the Beginning date and Ending date from the batch parameter as compared to the DOC_REC_DT on RETIRPAY.
- Records with Object codes (OBJ_CD) that belong to Form Type (FORM_TYP_IND) and Income Type (INC_TYP_NO) in the batch parameter.

Problem Resolution

The following table shows the potential job return codes for this job.

Return Code	Condition
Successful (1)	The job ends as successful when parameter validation is successful and at least one record was selected and processed.
Warning (4)	A warning results when no eligible RETIRPAY records matched the selection criteria.
Non-Fatal Error (8)	This job does not end with this return code.
Failed (12)	The job fails under the following conditions: • Parameters are invalid • Selection query execution failed • Commit of the transaction(s) to output failed • Run time exceptions for unexpected situations When this job fails, subsequent jobs in the chain are set to *Inactive*.
Terminated (16)	User terminates batch job.
System Failure (20)	Application is bounced while job is running.

Problem Resolution

Step 1: Parameter Validation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the parameters are validated successfully.	N/A	N/A
Failed (12)	One of the parameters failed validation.	A batch parameter can fail if a required parameter is not provided or user specified an invalid value.	N/A

Step 2: Selection of Records

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	At least one eligible RETIRPAY record was selected and the data from the record is successfully copied into a new IET transaction in the XML output file.	N/A	N/A
Warning (4)	No eligible RETIRPAY records matched selection criteria	Check the system debug log for RetireeIETLoad and the selection query that was executed to observe the search criteria and conditions.	Manually search out each of the table records being searched for by the selection query to confirm if those records actually exist.
Failed (12)	Selection query execution failed	Check the system debug log for RetireeIETLoad to observe the technical error messages.	Analyze the log error messages to derive a possible root cause to the issue.
	Run time exceptions for unexpected situations	Check the system debug log for RetireeIETLoad to observe the technical error messages.	Analyze the log error messages to derive a possible root cause to the issue.

Step 3: XML Output

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All eligible RETIRPAY records are successfully copied and saved into a new IET transaction in XML format if an eligible RETIRPAY record was found.	N/A	N/A
Failed (12)	Commit of the transaction(s) to	Check the system debug log for RetireeIETLoad to observe the error	Analyze the log error messages to derive a possible root cause to

Possible Return Codes	Condition	Recommendation	Other Instructions
	output failed	messages.	the issue.
	Run time exceptions for unexpected situations	Check the system debug log for RetireeIETLoad to observe the error messages.	Analyze the log error messages to derive a possible root cause to the issue.

Step 4: Parameter File

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Parameter file is successfully saved with the EXCEP_REP_IND, EXCEP_REP_FILE_NM, and for each IET header component the following values: DOC_CD, DOC_DEPT_CD, DOC_ID, and DOC_VERS_NO of 1	N/A	N/A
Failed (12)	Run time exceptions for unexpected situations	Check the system debug log for RetireeIETLoad to observe the error messages.	Analyze the log error messages to derive a possible root cause to the issue.

Retiree Billing Chain: IET Upload Job

Job Name	IET Upload
Recommended Frequency	N/A
Single Instance Required	N/A
Can be restarted?	No
Reports generated	None

Overview

This process uploads the IETDOC.xml file that is produced in step three of the Retiree Billing chain process.

Major Input

- IET XML Output File (XML_FILE): IETDOC.xml

Batch Parameters

The following are the delivered parameter values which may have been updated through Batch Setup to meet site needs.

Parameter	Description	Default Value
Action Code (ACTN_CD)	Action Code: Import	171
File Name (FILE_NM)	File Name	$$AMSEXPORT$$/IETDOC.xml
Apply Override (APPLY_OVERRIDES)	Apply Override (True/False)	True
Override Level (OVERRIDE_LVL)	Override Level	2
Bypass Auto Transaction Numbering (BYPS_ADNT_FL)	Bypass Auto Transaction Numbering (True/False)	True

Major Output

- Internal Exchange Transaction (IET / IET_DOC_HDR, IET_DOC_VEND, IET_DOC_ACTG)

Sort Criteria

None

Selection Criteria

None

Problem Resolution

The following table shows the potential job return codes for this job.

Return Code	Condition
Successful (1)	The job ends as successful when parameter validation is successful and all transactions in the XML file are loaded successfully.
Warning (4)	A warning results when the XML file is empty or when the maximum number of IET accounting lines is exceeded.
Failed (12)	The job fails under the following conditions: 1. Parameters are invalid 2. XML file not found

Return Code	Condition
	When this job fails, subsequent jobs in the chain are set to *Inactive*.
Terminated (16)	User terminates batch job.
System Failure (20)	Application is bounced while job is running.

Problem Resolution

Step 1: Parameter Validation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the parameters are validated successfully.	N/A	N/A
Failed (12)	One of the parameters failed validation.	A batch parameter can fail if a required parameter is not provided or user specified an invalid value.	N/A
	The input XML file not found.	Check that the file path to the input XML file is valid and/or file name is correct.	N/A

Step 2: Load Table

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the transactions were committed to the target table(s) of the database successfully.	N/A	N/A
Warning (4)	No transactions in the input XML file.	Check the input XML file to verify.	Analyze the job logs of the previous step in the chain job.
Failed (12)	Run time exceptions for unexpected situations	Check the system debug log for SysManUtil to observe the error messages.	Analyze the log error messages to derive a possible root cause to the issue.

Retiree Billing Chain: IET Submit Job

Job Name	IET Submit
Recommended Frequency	N/A
Single Instance Required	N/A
Can be restarted?	No
Reports generated	None

Overview

This process is used to process the IETSubmit.txt file that is produced in step three of the Retiree Billing chain job.

Major Input

- IET Submit TXT Output File (SUBMIT_PARM_FILE): IETSubmit.txt

Batch Parameters

The following are the delivered parameter values which may have been updated through Batch Setup to meet local needs.

Parameter	Description	Default Value
Parameter File (PARM_FILE)	Parameter File	$$AMSPARM$$/IETSubmit.txt

Major Output

Validation and submission of the IET transactions created.

Sort Criteria

None

Selection Criteria

None

Problem Resolution

The following table shows the potential job return codes for this job.

Return Code	Condition
Successful (1)	The job ends as successful when parameter validation is successful and the TXT file is processed successfully.

Return Code	Condition
Warning (4)	This job does not end with this return code.
Non-Fatal Error (8)	This job does not end with this return code.
Failed (12)	The job fails under the following conditions: • Parameters are invalid • TXT file not found When this job fails, subsequent jobs in the chain are set to *Inactive*.
Terminated (16)	User terminates batch job.
System Failure (20)	Application is bounced while job is running.

Problem Resolution

Step 1: Parameter Validation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the parameters are validated successfully.	N/A	N/A
Failed (12)	One of the parameters failed validation.	A batch parameter can fail if a required parameter is not provided or user specified an invalid value.	N/A
	The input TXT file not found.	Check that the file path to the input TXT file is valid and/or file name is correct.	N/A

Step 2: Parameter File

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the contents of the parameter file were processed successfully.	N/A	N/A
Failed (12)	Run time exceptions for unexpected situations	Check the system debug log for SysManUtil to observe the error messages.	Analyze the log error messages to derive a possible root cause to the issue.

Retiree Billing Chain: RETIRPAY Summary Job

Job Name	RETIRPAY Summary
Recommended Frequency	N/A
Single Instance Required	N/A
Can be restarted?	No
Reports generated	None

Overview

This process produces a CSV output file that contains summary level payment data by TIN obtained from combined RETIRPAY/VCUST record pairs based on date values from the DateParam.txt input file.

Data in the file includes the following:

Field Name	Field Length	Type	Logic
FIELD_1	1	Varchar	“D” (Always this value)
FIELD_2	2	Varchar	“86” (Always this value)
TIN	9	Varchar	TIN
RETIREE_NM	30	Varchar	Last name, first name
PROC_DT	4	Varchar	MMYY format (Use the END_DT batch parameters from the RETIRPAY XML Generation job. Map the month from the END DATE to MM. Map the year from the END_DT to the YY.)
SUM_PYMT_AM		Amount	Sum of Posting Amount of the selected records for the vendor.

Process Steps	Messages
1. Parameter Validation	• Validating Batch Parameters • Parameters are valid or invalid depending on the validation. If the parameter is invalid, the invalid value is displayed in the log. • Batch Parameter validation completed
2. Parameter File	• Load a parameter file that contains the values of the Beginning Date (BGN_DT), Ending Date (END_DT), and

Process Steps	Messages
	Processed Date (PROC_DT) only if the parameter validation step did not fail. • If any of the date values are missing, the following error message will appear: “Date values are missing in the parameter file”.

3. CSV Output	• Selected RETIRPAY and VCUST record are matched together by the Vendor/Customer Code (VEND_CUST_CD).
	• If eligible record pairs are found, then the following rows are added to a CSV output file (column / value):
	  • FIELD_1 / D
	  • FIELD_2 / 86
	  • TIN / R_RETIRE_PYMTS.TIN
	  • RETIREE_NM / LAST_NM, FRST_NM
	  • PROC_DT
	  • SUM_PYMT_AM / SUM(R_RETIRE_PYMTS.PSTNG_AM)
	• If no eligible record pairs are found, then do nothing and display message to job log.

Major Input

- Date Parameters TXT Output File (DATE_PARAM): DateParam.txt
- Retiree Payments (RETIRPAY / R_RETIRE_PYMTS)
- Vendor/Customer (VCUST / R_VEND_CUST)

Batch Parameters

The following are the delivered parameter values which may have been updated through Batch Setup to meet local needs.

Parameter	Description	Default Value
Export/Import Location (AMSEXPORT)	Required ExportImport location	$$AMSROOT$$/ExportImport
Date Parameters TXT Input File (DATE_PARAM)	Required name of the imported txt file.	DateParam.txt
Commit Block Size (COMMIT_BLOCK)	Commit Block Size. If blank or invalid, defaults to 100.	100

Select Block Size (SELECT_BLOCK)	Select Block Size. If blank or invalid, defaults to 1000.	1000
Progression Counter Size (PROG_CTR_SZ)	Progression Counter Size. If blank or invalid, defaults to 5000.	5000

Major Output

- RETIRPAY Summary CSV Output File: *ChainJobId*.csv

Sort Criteria

- R_RETIRE_PYMTS

TIN

- R_VEND_CUST

LAST_NM

FRST_NM

Selection Criteria

- R_RETIRE_PYMTS
  - TIN
  - PSTNG_AM
  - DOC_REC_DT
  - VEND_CUST_CD
- R_VEND_CUST
  - LAST_NM
  - FRST_NM
  - VEND_CUST_CD

Problem Resolution

The following table shows the potential job return codes for this job.

Return Code	Condition
Successful (1)	The job ends as successful when parameter validation is successful and at least one record was selected and processed.
Warning (4)	A warning results when no eligible RETIRPAY/VCUST record pairs matched the selection criteria.
Non-Fatal Error (8)	This job does not end with this return code.

Return Code	Condition
Failed (12)	The job fails under the following conditions: 3. Parameters are invalid 4. Parameter file failed to load 5. Selection query execution failed 6. Commit of the transaction(s) to output failed 7. Run time exceptions for unexpected situations When this job fails, subsequent jobs in the chain are set to *Inactive*.
Terminated (16)	This job does not end with this return code.
System Failure (20)	This job does not end with this return code.

Problem Resolution

Step 1: Parameter Validation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the parameters are validated successfully.	N/A	N/A
Failed (12)	One of the parameters failed validation.	A batch parameter can fail if a required parameter is not provided or user specified an invalid value.	N/A

Step 2: Parameter File

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	The parameter file and its contents successfully loaded.	N/A	N/A
Failed (12)	The parameter file failed to load.	Check the system debug log for RetireeBillingSummaryReport to observe the error messages.	Analyze the log error messages to derive a possible root cause to the issue.

Step 3: CSV Output

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All eligible RETIRPAY/VCUST record pairs are successfully saved to the CSV output file.	N/A	N/A
Warning (4)	No eligible RETIRPAY/VCUST record pairs matched selection criteria	Check the system debug log for RetireeBillingSummaryReport and the selection query that was executed to observe the search criteria and conditions.	Manually search out each of the table records being searched for by the selection query to confirm if those records actually exist.
Failed (12)	Commit of the transaction(s) to output failed	Check the system debug log for RetireeBillingSummaryReport to observe the technical error messages.	Analyze the log error messages to derive a possible root cause to the issue.
	Run time exceptions for unexpected situations	Check the system debug log for RetireeBillingSummaryReport to observe the error messages.	Analyze the log error messages to derive a possible root cause to the issue.

2.1.46 Retiree Update

Chain Name	Retiree Update
Recommended Frequency	Daily as part of the nightly cycle or on demand
Single Instance Required	Yes
Can be restarted?	N/A
Reports generated	N/A

Overview

This new chain job reads an external file with retiree taxpayer identification number (TIN), status, and retirement date from an external payroll system to validate the existences of those records against the Vendor Customer (VCUST) page. Validated data is then used to an XML file of retiree data that is loaded to the Retiree TIN (RETIRTIN) page.

This chain has the following jobs:

- Retiree Flat to Xml
- Load To RETIRTIN Table

Major Input

- RETIRTINUPDATE.txt
- Vendor Customer (VCUST / R_VEND_CUST)

Major Output

- RETIRTINUPDATE.xml
- Retiree TIN (RETIRTIN / R_RETIRE_TIN)

Chain Return Codes

When an active job step in the chain ends with a Return Code of anything other than successful, any remaining job steps have a Run Status of *Inactive.* Please see documentation on individual job steps for possible Return Codes for those steps, as *Warning* and *Non-Fatal Errors* are not always possible outcomes.

Problem Resolution

If any of the jobs in the chain failed due to application errors, it is advisable to restart the job after correcting the errors instead of rescheduling the job. Alternatively, a new chain can be submitted that starts with the job that previously failed.

Retiree Update Chain: Retiree Flat to Xml

Job Name	Retiree Flat to Xml
Recommended Frequency	N/A
Single Instance Required	N/A
Can be restarted?	No
Reports generated	No

Overview

The Retiree Flat to XML process reads an external input TXT file. If the record in the file has a status of “RETIREE”, the record is processed. Any other statuses are skipped. The selected records are used to match the TIN in the file to the first TIN record found on Vendor Customer. It validates the TIN Type is SSN/ITIN/ATIN and that the Legal Name exists. If either are incorrect or missing, a message is written the log. Each selected record that validates is written to an XML file for a later job step. When an invalid data is encountered in the file, the process writes a message to the log, skip record, and process the next record.

Steps in Running this Process:

1. **Parameter Validation**: First the process validates the batch parameters.
2. **Checking input File**: The job checks the input file whether it contains any records or not.
3. **Validate Records**: The job validates records in the input file.
4. **XML File Creation**: Creates an XML file with the name ‘RETIRTINUPDATE.xml’

The following table shows the various steps of the Retiree Flat to XML job and the messages issued at each step:

Process Steps	Messages
1. Parameter Validation	• If the parameters are valid, the job logs a message indicating all parameters are validated. • If the parameters are invalid, the job logs a message indicating parameter validations failed for the Retiree Flat to XML job.
2. Checking input File	• If the job finds records in the input file, it logs a message indicating that Started Reading Inbound Flat File. • If the job didn’t find a record, then a message is logged indicating no records found in the inbound flat file to process.
3. Validate Records	• If the TIN is not found in R_TIN_1099_INFO, the job logs a message indicating it could not find a TIN on the 1099I. • If the date is invalid, then log a message indicating the date should be in MM/YYYY format and month should

Process Steps	Messages
	be between 00 to 12, and the year should be between 1000 to 9999. • If the TIN is found but the Legal Name is not entered, the job logs a message indicating the Legal Name is not found for TIN. • If the TIN is found but the TIN Type is EIN, the job logs a message indicating the TIN/TIN Type combination is not found on VCUST table for TIN and TIN Type of SSN/ITIN/ATIN.
4. XML File Creation	• If the process is completed, then job logs messages indicating 1. Total # Records 2. Total # Valid Records 3. Total # Invalid Records

Major Input

- Vendor Customer (VCUST / R_VEND_CUST)
- Input TXT file

Field	Field Length	Data Type	Field
TIN	9	VarChar	TIN – SSN for Retiree
	1	VarChar	Space
STATUS	7	VarChar	Retiree Status = RETIREE
	1	VarChar	Space
RETIRE_DT	7	VarChar	Retirement Date; format MM/YYYY

Batch Parameters

The following are the delivered parameter values which may have been updated through Batch Setup to meet local needs.

Parameter	Description	Default Value
Flat file (FLAT_FILE)	Required name of the inbound flat file for processing.	RETIRTINUPDATE.txt
Progression Counter Size	Progression Counter Size. If blank or invalid, defaults to 5000.	5000

(PROG_CTR_SZ)		
Export Location (AMSEXPORT)	Required location where the flat file is written.	$$AMSROOT$$/ExportImport
XML Output File Name (XML_FILE)	A required output file name for the XML file.	RETIRTINUPDATE.xml

Major Output

- RETIRTINUPDATE.xml
  - Fields of the XML file:

Field Name	Field Length	From Date	To Date	Type	Data Source
HQ_TIN	9	1	9	VarChar	TIN from inbound txt file
HQ_TIN_TYP	1	10	10	VarChar	TIN_TYP from VCUST
HQ_LGL_NM	60	11	70	VarChar	Inferred from VCUST using TIN, TIN_TYP
TIN	9	71	79	VarChar	TIN from inbound txt file
TIN_TYP	1	80	80	VarChar	TIN_TYP from VCUST
LGL_NM	60	81	140	VarChar	Inferred from VCUST using TIN, TIN_TYP
RETIRE_DT	7	141	147	Date	RETIRE_DT from inbound txt file.

Sort Criteria

None

Selection Criteria

The process selects records from the input file with Retiree Status = Retiree.

Problem Resolution

The following table shows the possible job Return Codes for the Retiree Flat To Xml job:

Return Code	Condition
Successful (1)	The job ends as successful when the parameter validation is

Return Code	Condition
	successful and at least one record was selected and processed and written to an XML file.
Warning (4)	The job returns warning under the following conditions: 8. No records are found in the flat file to process. 9. Could not find valid records to process.
Non-Fatal Error (8)	This job does not issue this return code.
Failed (12)	The job fails under the following conditions: 10. Parameters are invalid. When this job fails, subsequent jobs in the chain are set to *Inactive*.
Terminated (16)	This return code is issued when the job is terminated by the user and subsequent jobs in the chain are set to *Inactive*.
System Failure (20)	A system failure is issued when the job is terminated because of the database server or network issues. When this job encounters a system failure, subsequent jobs in the chain are set to *Inactive*.

Problem Resolution

If the job ends with a return code of *Failed* and above, the job can be rescheduled.

The following table shows the various steps that the Retiree Flat to Xml job goes through and the messages issued at each step:

Step 1: Parameter Validation

Return Code	Condition	Recommendation	Other Instructions
Successful (1)	All of the parameter validations are successful.	N/A	N/A
Warning (4)	N/A	This step does not issue this return code.	N/A
Non-Fatal Error (8)	N/A	This step does not issue this return code.	N/A
Failed (12)	The job fails if the Parameters are invalid. **Sample Message:** “The value entered in the AMSEXPORT	If the required parameters are not entered (or) not entered correctly, enter the required parameters, and reschedule the job.	N/A

Return Code	Condition	Recommendation	Other Instructions
	parameter is not a valid directory." **Recommendation:** Enter Valid Directory.		
Terminated (16)	The job was terminated manually by the user.	The reason for the termination needs to be investigated. The job can be rescheduled. addressed.	N/A
System Failure (20)	The job was terminated because of the database server or network issues.	The reason for the System Failure needs to be investigated. The job can be rescheduled.	N/A

Step 2: Checking Input File

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	When the process finds records in the input file.	N/A	N/A
Warning (4)	When the process was not able to find records in the input file.	Add a few entries in the input file.	N/A
Non-Fatal Error (8)	N/A	This step does not issue this return code.	N/A
Failed (12)	N/A	This step does not issue this return code.	N/A
Terminated (16)	When the job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can be rescheduled.	N/A
System Failure (20)	When the job is terminated because of the database server or network issues.	The reason for the System Failure needs to be investigated. The job can be rescheduled.	N/A

Step 3: Validate Records

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	When the process validates all records.	N/A	N/A
Warning (4)	When the process was not able to find a valid record to process.	Add valid entries in input file with a Status of 'Retiree'.	N/A
Non-Fatal Error (8)	N/A	This step does not issue this return code.	N/A
Failed (12)	N/A	This step does not issue this return code.	N/A
Terminated (16)	When the job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can be rescheduled.	N/A
System Failure (20)	When the job is terminated because of the database server or network issues.	The reason for the System Failure needs to be investigated. The job can be rescheduled.	N/A

Step 4: XML File Creation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	When the process creates the XML file.	N/A	N/A
Warning (4)	N/A	This step does not issue this return code.	N/A
Non-Fatal Error (8)	N/A	This step does not issue this return code.	N/A
Failed (12)	N/A	This step does not issue this return code.	N/A

Possible Return Codes	Condition	Recommendation	Other Instructions
Terminated (16)	When the job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can be rescheduled.	N/A
System Failure (20)	When the job is terminated because of the database server or network issues.	The reason for the System Failure needs to be investigated. The job can be rescheduled.	N/A

Retiree Update Chain: Load To RETIRTIN Table

Job Name	Load To RETIRTIN Table
Recommended Frequency	N/A
Single Instance Required	N/A
Can be restarted	No
Reports Generated	No

Overview

The Load To RETIRTIN Table job loads the XML file generated in the previous step into RETIRTIN. The job uses the System Maintenance Utility (SMU) functionality to load the data.

The steps involved in this process are:

1. **Parameter Validation:** First, the process validates the batch parameters.
2. **Load Data:** This is achieved by the SysManUtil functionality.

Major Input

- XML file generated by the first step of chain job.

Job Parameters

Parameter	Description	Default Value
ACTN_CD	Required SysManUtil Action Code for Table Import.	201
FILE_NM	Required name of the XML input file of Retiree TIN	$$AMSEXPORT$$/RETIRTINUPDATE.xml

Parameter	Description	Default Value
	records.	

Major Output

- Retiree TIN (RETIRTIN / R_RETIRE_TIN)

Job Codes

Since this job uses the common SysManUtil functionality, the job Return Code and the exception handling are handled by the SysManUtil functionality and there is no specific requirement for this job.

Restartability

The job cannot be restarted. If the job fails in any of the above steps, a new job should be scheduled after correcting the errors that caused the job to fail.

Sort Criteria

N/A

Selection Criteria

N/A

Problem Resolution

If the job ends with a return code of *Failed* and above, the job can be rescheduled.

The following table shows the various steps that the Load To RETIRTIN Table process goes through and the messages issued at each step.

Step 1: Parameter Validation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Parameter validations are successful.	N/A	N/A
Warning (4)	N/A	This step does not issue this return code.	
Non-Fatal Error (8)	N/A	This step does not issue this return code.	
Failed (12)	All required parameters are not entered.	If the required parameters are not entered, enter the required parameter and	

	Sample Message: "XML File is required"	reschedule the job.	
	Entered parameters are not valid. **Sample Message:** "XML File is not found on the specified location."	If the parameters are invalid, enter the valid parameter and reschedule the job.	
Terminated (16)	The job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can be rescheduled.	
System Failure (20)	The job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can be rescheduled.	

Step 2: Load Data

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	This Return Code is issued under the following condition: • The input file is empty. **Sample Message:** "No data found on the input file"	Provide an input file with valid records.	N/A
Non-Fatal Error (8)	N/A	N/A.	N/A
Failed (12)	N/A	N/A	N/A
Terminated (16)	The job is terminated manually by the user.	The reason for the termination needs to be investigated before rescheduling the entire chain.	N/A
System Failure (20)	The job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated before rescheduling	N/A

		the entire chain.	

2.1.47 Retiree VCUST Extract

Job Name	Retiree VCUST Extract
Recommended Frequency	This job runs on a monthly basis.
Single Instance Required	Yes
Can be restarted?	No
Reports generated	This job creates a flat file in TXT format.

Overview

This new batch job is developed to extract Vendor Customer (VCUST) records to a text file for specific processing by a third-party application. It reads through the Vendor Customer and creates an extract file of vendor records that meet specific selection criteria, detailed in a later section.

Steps in Running this Process:

1. **Parameter Validation**: First the process validates the batch parameters.
2. **Selection of records**: Based on the criteria, the job selects records and if the Exclude Vendor Code Prefix has value, then it excludes those records while selecting.
3. **Creating the file**: The process creates a file with the name 'RETIREE_UPDATE.txt' then it writes records to it.

The following table shows the various steps that the Retiree VCUST Extract job goes through and the messages issued at each step:

Process Steps	Messages
1. Parameter Validation	• If the parameters are valid, the process logs a message indicating all parameters are validated. • If the parameters are not valid, the process logs a message indicating parameter validations failed for Retiree VCUST Extract job.
2. Record Selection	• Based on the criteria, the job extracts records from the VCUST table. • If the process found valid records, the job logs a message indicating Retiree VCUST Extract Batch started processing records. • If no records are found, the job logs a message indicating No eligible records found to process.
3. File Creation	• If the process completes writing records to the file, it logs a message indicating the Processed Output file name is

Process Steps	Messages
	RETIREE_UPDATE.txt.

Restartability

The job cannot be restarted. If the job fails in any of the above steps, a new job should be scheduled after correcting the errors that caused the job to fail.

Major Input

- Vendor Customer (VCUST / R_VEND_CUST)

Batch Parameters

The following are the delivered parameter values which may have been updated through Batch Setup to meet local needs.

Parameter	Description	Default Value
Export Location (AMSEXPORT)	Required location where the flat file is written.	$$AMSROOT$$/ExportImport
Exclude Vendor Code Prefix (EXC_VEND_CD_PFX)	Optional selection parameter of vendor codes by code prefix. Multiple values allowed is comma-separated.	(No Default)
Progression Counter Size (PROG_CTR_SZ)	A performance parameter used in progression messaging. If blank or invalid, the default is 5000.	5000
Select Block Size (SELECT_BLOCK)	A performance parameter that controls record selection for review. If blank or invalid, defaults to 1000.	1000

Major Output

This process creates a file with the name 'RETIREE_UPDATE.txt' that includes a list of Taxpayer Identification Numbers from VCUST that met the selection criteria.

Job Return Codes

If this job does not finish successfully, there is no restarting. A new of the job should be submitted after addressing the reason(s) for failure.

Return Code	Condition
Successful (1)	The job ends as successful when parameters are valid, record selection was successful, and text file creation was successful.

Return Code	Condition
Warning (4)	A warning results when no matching records were found.
Non-Fatal Error (8)	The job does not use this return code.
Failed (12)	The job fails under the following conditions: • Parameters are invalid. • Run time exceptions for unexpected situations.
Terminated (16)	The job is terminated by the user.
System Failure (20)	A system failure is issued when the job is terminated because of database server or network issues.

Sort Criteria

None

Selection Criteria

This job uses the following selection criteria:

- RETIRE_FL = 0,
- RETIRE_DT is blank.
- ORG_TYPE/ORG_CLS meets any condition below:
  - ORG_TYP equal 1 (Individual), or
  - ORG_TYP equal 2 (Company) and ORG_CLS equal 1 (Individual), or
  - ORG_TYP equal 2 (Company) and ORG_CLS equal 2 (Sole Prop), or
  - ORG_TYP equal 2 (Company) and ORG_CLS equal 18 (LLC Filing as Sole Prop), or
  - ORG_TYP equal 2 (Company) and ORG_CLS equal 20 (Personal Service), or
- ORG_CLS does not equal 13 (indicates active employee, which is to be excluded from processing).
- HLD_PYMT_FL = 0.
- If the batch parameter ‘EXC_VEND_CD_PFX’ has a value, the job excludes those vendor records from VCUST which begins with the specified prefix.

Problem Resolution

If the job ends with a return code of *Failed* and above, the job can be rescheduled.

The following table shows the various steps that the Retiree VCUST Extract job goes through, and the messages issued at each step:

.

Step 1: Parameter Validation

Return Code	Condition
Successful (1)	This step is Successful when all parameter validations are successful.
Warning (4)	This step does not issue this return code.
Non-Fatal Error (8)	This step does not issue this return code.
Failed (12)	The job fails under the following conditions: • If the parameters are invalid.
Terminated (16)	The job is terminated manually by the user.
System Failure (20)	A system failure is issued when the job is terminated because of the database server or network issues.

Step 2: Record Selection

Return Code	Condition
Successful (1)	This step is Successful when the job extracts the records from VCUST.
Warning (4)	A warning results when no eligible records are found in the selection.
Non-Fatal Error (8)	This step does not issue this return code.
Failed (12)	This step does not issue this return code.
Terminated (16)	The job is terminated manually by the user.
System Failure (20)	A system failure is issued when the job is terminated because of the database server or network issues.

Step 3: **File Creation**

Return Code	Condition
Successful (1)	This step is Successful when the job creates a file and writes records to it.
Warning (4)	This step does not issue this return code.
Non-Fatal Error (8)	This step does not issue this return code.
Failed (12)	The job fails under the following conditions: • A problem occurring while creating an output file.

Return Code	Condition
Terminated (16)	The job is terminated manually by the user.
System Failure (20)	A system failure is issued when the job is terminated because of database server or network issues.

2.1.48 Returned ACH Transaction Process

Job Name	Returned ACH Transaction Process
Recommended Frequency	This process runs before the AD Chain and Mass Cancellation batch processes during the nightly cycle
Single Instance Required	Yes.
Can be restarted?	No.
Reports Generated	Yes.

Overview

The Returned ACH Transaction process is a chain job that will initiate the cancellation of rejected EFT payments. The process will also mark the vendors associated with returned Prenotes, EFT payments and EFT reversals as ineligible for EFT at the Vendor Location level and Address level, where applicable. This batch process can be used with the supported ACH formats CCD, PPD and CTX. The bank can send a Returned ACH File with EFT payments, EFT reversals and/or Prenotes.

The chain has the following jobs:

- Flat to XML File
- Load XML File
- Process ACH Return
- Initiate Disbursement Cancellation

Major Input

- Returned ACH File (CTX, PPD or CCD)
- Check Recon table (R_AP_CHK_RECON)
- Paid Check table (AP_PD_CHK)
- AD Transaction Header table (AD_DOC_HDR)
- AD Transaction Vendor table (AD_DOC_VEND)
- Vendor Customer table (R_VEND_CUST)
- EFT Return table (R_EFT_RETURN)
- EFT Reversal table (R_EFT_RVRSL)

Major Output

- Returned EFT and Prenotes table (RET_EFT_PNOTE)
- EFT Return table (R_EFT_RETURN)
- EFT Reversal table (R_EFT_REVERSAL)

- Disbursement Cancellation Parameters table (R_AP_DISBCAN_PARM)
- Vendor Customer table (R_VEND_CUST)
- Payee/Vendor Prenote report
- Payee/Vendor Prenote Exception report
- Returned EFT Transactions report
- Returned EFT Transactions Exception report
- Initiated Cancellations of Returned EFT Payments report

Job Return Code

The following table shows the potential job return codes for Reversal ACH Transaction Batch Process:

Return Code	Condition
Successful (1)	All of the validations are performed successfully and no errors as reported.
Warning (4)	No records found under the selection criteria.
Non-Fatal Error (8)	N/A
Failed (12)	The job fails under the following conditions: • Parameters are invalid. • No records that meet existing criteria were found • Run time exceptions for unexpected situations. When this job ends with a Return Code of Failed, subsequent jobs in the chain are set to Inactive.
Terminated (16)	This return code is issued when the job is terminated by the user. When this job ends with a Return Code of Terminated, subsequent jobs in the chain are set to Inactive.
System Failure (20)	This return code is issued when the job is terminated because of database server or network issues.

Returned ACH Transaction Process: Flat to XML File

Job Name	Flat to XML File
Recommended Frequency	Nightly Cycle
Single Instance Required	Yes.
Can be restarted?	No.

Reports Generated	No

Overview

This job will read the input Returned ACH Bank file and create the output file in an XML format.

Job Parameters

Parameter	Description	Default Value
Export file location at Flat to XML Job	This is the location where the output file will be stored	No Default
Import file location at Flat to XML Job	This is the location where the job will look for the input file	No Default
Parameter Location at Flat to XML Job	This is the location where the job stores the parameter file for the subsequent jobs in the chain	No Default
Data Object Name	The table name of the Returned EFT and Prenotes Table that will be populated here. This is a required parameter	No Default
Attribute List	The list of attributes of the table (Returned EFT and Prenotes table) where the data from the XML file will be uploaded to. This is a required parameter	No Default
Flat File Name	The name of the ACH file received from the Bank. This file is the input for this job. This is a required parameter	No Default
XML File Name	The name of the output file. XML file will be created with this name and will be stored in the Export file location. This is a required parameter	No Default

Steps for running this process:

1. **Parameter Validation**: First the process will validate the batch parameters.
2. **Create XML:** The process will create a single record XML file with given Derivation logic. After creating the XML file, the process will rename the text file to Input file name followed by "_processed".

Restartability Information

The job cannot be restarted in this run mode. If the job fails in any of the above steps, a new job should be scheduled after correcting the errors that caused the job to fail.

Selection Criteria

N/A

Sort Sequence

N/A

Problem Resolution

If the job ends with a return code of Failed and above, the job can be restarted.

The following table shows the various steps that the Flat to XML File Process goes through and the messages issued at each step.

Step 1: Parameter Validation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the table validations are successful, that is, there are no errors raised during validations.	N/A	N/A
Warning (4)	N/A	This step does not issue this return code.	
Non-Fatal Error (8)	N/A	This step does not issue this return code.	
Failed (12)	All of the required parameters are not entered. Sample Message: “Data Object Name cannot be Null” Recommendation: The Data Object Name parameter must not be Null	If the required parameters are not entered, enter the required parameters and restart the job.	
	Entered Parameters are not valid. Sample Message: “Attribute xxxx (xxxx being the attribute name) is not	If the parameters are invalid, enter the valid parameter and restart the job.	

	valid on the Data object xxxx (Data object name)" Recommendation: Attribute List parameter shall be valid on the Data Object entered in the Data Object parameter		
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can either be restarted or a new job scheduled.	
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can either be restarted or a new job scheduled.	

Step 2: Create XML:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	Under the following conditions: If the input file is empty. If there are no detail records. Sample Message: "No data found on the input file" "No Detail records found on the input file"	Make sure that the parameter value entered is correct. If the parameter value is incorrect, then enter the correct parameter value and run the chain again.	N/A
Non-Fatal Error (8)	N/A	N/A.	N/A
Failed (12)	N/A	N/A	N/A
Terminated(16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated before rescheduling the entire chain.	N/A

System Failure(20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated before rescheduling the entire chain.	N/A

Returned ACH Transaction Process: Load XML File

Job Name	Load XML File
Recommended Frequency	nightly cycle
Single Instance Required	Yes.
Can be restarted?	No.
Reports Generated	No

Overview

This job loads the data from the file into the table. It will load the XML file created in the first job. This job will use the SysManUtil functionality to load the data into the Returned EFT and Prenotes Table.

Job Parameters

Parameter	Description	Default Value
Import file location at Flat to XML Job	This is the location where the job will look for the input file	No Default
Commit Block size	Commit Block size	No Default
XML File Name	This is the input file name. The job will load the data from this file only to load into the table. This is a required parameter.	No Default
Generate Statistics	Generate Statistics	True

Steps for running this process:

1: Parameter Validation: First the process will validate the batch parameters.

2: Load the Data into the table: This will be achieved by the SysManUtil functionality.

This process will set the Processed Flag and Processed Date to *False* and *Blank*, respectively, when inserting records into the Returned EFT and Prenotes table.

Job Codes

Since this job uses the common SysManUtil functionality, the job return code and the exception handling will be handled by the SysManUtil functionality and there is no specific requirement for this job.

Restartability Information

The job cannot be restarted in this run mode. If the job fails in any of the above steps, a new job should be scheduled after correcting the errors that caused the job to fail.

Selection Criteria

N/A

Sort Sequence

N/A

Problem Resolution

If the job ends with a return code of Failed and above, the job can be restarted.

The following table shows the various steps that the Load XML File Process goes through and the messages issued at each step.

Step 1: Parameter Validation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the table validations are successful, that is, there are no errors raised during validations.	N/A	N/A
Warning (4)	N/A	This step does not issue this return code.	
Non-Fatal Error (8)	N/A	This step does not issue this return code.	
Failed (12)	All of the required parameters are not entered. Sample Message: “XML File is required” Recommendation: The XML File is required.	If the required parameters are not entered, enter the required parameters and restart the job.	
	Entered Parameters are not valid. Sample Message:	If the parameters are invalid, enter the valid parameter and restart the job.	

	"XML File is not found on the specified location" Recommendation: Import File location Object entered in the Data Object parameter must be valid.		
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can either be restarted or a new job scheduled.	
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can either be restarted or a new job scheduled.	

Step 2: Load the Data into the table:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	Under the following conditions: If the input file is empty. Sample Message: "No data found on the input file"	Make sure that the parameter value entered is correct. If the parameter value is incorrect, then enter the correct parameter value and run the chain again.	N/A
Non-Fatal Error (8)	N/A	N/A.	N/A
Failed (12)	N/A	N/A	N/A
Terminated(16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated before rescheduling the entire chain.	N/A
System Failure(20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated before rescheduling the entire chain.	N/A

Returned ACH Transaction Process: Process ACH Return

Job Name	Process ACH Return
Recommended Frequency	nightly cycle
Single Instance Required	Yes
Can be restarted?	No
Reports Generated	Payee/Vendor Prenote report Payee/Vendor Prenote Exception report Returned EFT Transactions report Returned EFT Transactions Exception report

Overview

This job processes the returned EFT/Prenote transactions from the Bank as follows:

- If the transaction is a returned EFT Reversal, the job will look for a matching record on the EFTREV and CHREC pages based on EFT Number, Bank Account code and Amount combination. Then it will update the EFT Reversals and EFT Return tables respectively, and mark the associated Vendor as ineligible for EFT payments on the VCUST table.
    - If the associated Vendor is Miscellaneous Vendor on VCUST, the system will update only the EFT Reversal and EFT Return records and will not update the EFT eligible status on VCUST.
- If the transaction is a rejected EFT payment, the job will look for a matching record on CHREC based on EFT Number, Bank Account and Amount combination. Then it will update the EFT Return record, initiate the cancellation of the associated EFT transaction, and mark the associated Vendor as ineligible for EFT payments on VCUST.
    - If the associated Vendor is Miscellaneous Vendor on VCUST, the system will update the EFT Return record and initiate the cancellation of the associated EFT transaction and will not update the EFT eligible status on VCUST.
- If the transaction is Notification of Change, the job will look for a matching record on CHREC or PDCHK based on EFT Number and Bank Account combination. Then it will update the EFT Return record and mark the associated Vendor as ineligible for EFT payments on VCUST.
    - If the associated Vendor is Miscellaneous Vendor on VCUST and no Payee details on the AD Header, the system will update only the EFT Return record and will not update the EFT eligible status on VCUST.
- If the returned transaction is a Prenote, the job will mark the associated Vendor as ineligible for EFT payments on VCUST.

Major Input

- Returned EFT and Prenotes (RET_EFT_PNOTE)
- Check Recon (R_AP_CHK_RECON)

- Paid Check (AP_PD_CHK)
- AD Transaction Header (AD_DOC_HDR)
- AD Transaction Vendor (AD_DOC_VEND)
- Vendor Customer (R_VEND_CUST)
- EFT Reversal (R_EFT_REVERSAL)

Major Output

The job will update the following tables:

- Returned EFT and Prenotes (RET_EFT_PNOTE)
- EFT Return (R_EFT_RETURN)
- EFT Reversal (R_EFT_REVERSAL)
- Vendor Customer (R_VEND_CUST)

Job Parameters

N/A

Steps for running this process:

1. **Selection of records: :** The job shall select the records from the Returned EFT and Prenotes Table where the Processed Flag is false and the Processed date is null. Proceed to next step only if the job finds any eligible records on the table.
2. **Process EFT Returns:** It will process the returned / rejected EFT records. If there are no returned or rejected EFT records (amount greater than zero), then job will go to the next step of processing the rejected Prenotes.
3. **Process Returned Prenote records**: It will process the returned Prenote records. If the Returned EFT and Prenotes table record's Amount = $0 (i.e. returned Prenote), the job shall update the Prenote status of the associated Vendor on the Vendor Customer Table. The job also generates the Payee / Vendor Prenote Report.

Restartability Information

The job cannot be restarted in this run mode. If the job fails in any of the above steps, a new job should be scheduled after correcting the errors that caused the job to fail.

Selection Criteria

N/A

Sort Sequence

N/A

Problem Resolution

If the job ends with a return code of Failed and above, the job can be restarted.

The following table shows various Return codes for the step Selection of records in Process ACH Return job goes through and the messages issued at each step.

Selection of records:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	Under the following conditions: • If no records are selected from Returned EFT and Prenotes Table where the Processed Flag is false and the Processed date is null. **Sample Message:** • "No eligible records found on the Returned EFT and Prenotes table"	Make sure that the parameter value entered is correct. If the parameter value is incorrect, then enter the correct parameter value and run the chain again.	N/A
Non-Fatal Error (8)	N/A	N/A.	N/A
Failed (12)	Under the following conditions: If the Job cannot match a returned record with the application data.	N/A	N/A
Terminated(16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated before rescheduling the entire chain.	N/A
System Failure(20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated before rescheduling the entire chain.	N/A

Returned ACH Transaction Process: Initiate Disbursement Cancellation

Job Name	Initiate Disbursement Cancellation
Recommended Frequency	nightly cycle
Single Instance Required	Yes

Can be restarted?	No
Reports Generated	Initiated Cancellations of Returned EFT Payments report

Overview

This job initiates the cancellation of the EFT transactions associated with the rejected EFT payments. It selects the eligible records from the EFT Return Table and inserts records into the Disbursement Cancellation parameter table so that the Mass Cancellation process can cancel those EFT payments that were rejected by the bank.

This job can be run independently.

Major input

- EFT Return table (R_EFT_RETURN)

Major output

- EFT Return table (R_EFT_RETURN)
- Disbursement Cancellation Parameters table (R_AP_DISBCAN_PARM)

Job Parameters

Parameter	Description	Default Value
Cancellation Type	Required parameter. Valid values are Hold, Reschedule, Close and PR Cancellation.	No Default
Cancellation Reason	Required parameter. The entered value must be valid on the Cancellation Reason CVL.	No Default
Rescheduled Payment Date	Conditionally Required. Required only when the Cancellation Type is equal to “Reschedule”. If entered, the date must be greater than or equal to the Application control date.	No Default
Hold Type Code	Conditionally Required. Required only when the Cancellation Type is equal to Hold. If entered it must be valid on the Payment Hold Type table for the Payment Hold Type Department and Payment Hold Type Unit parameters.	No Default
Payment Hold Type Department	Conditionally Required. Required only when the Cancellation Type is equal to Hold.	No Default

Payment Hold Type Unit	Conditionally Required. Required only when the Cancellation Type is equal to Hold.	No Default
Hold Request Description	Conditionally Required. Required only when the Cancellation Type is equal to Hold.	No Default

Steps for running this process:

1. **Parameter Validation:** First the process validates the batch parameters.
2. **Selection of records:** The job shall select all the records from EFT Return with the Reversal Status set to “EFT Cancel Initiated” or “Reversal Failed by the Bank” and Date EFT Cancel initiated is blank. The job will proceed to the next step only if it selects the eligible records.
3. **Initiate Disbursement Cancellation:** It will insert records into Disbursement Cancellation Parameters for the selected EFT Return record.
    - Insert a record to Disbursement Cancellation Parameter as shown below:
        - Bank Account Code – Populated with Bank Account code from the EFT Return record.
        - Cancellation Type – Populated with the Cancellation Type batch parameter value.
        - Cancellation Reason – Populated with the Cancellation Reason batch parameter value.
        - Hold Type – Populated with the Hold Type code from the batch parameter value.
        - Hold Type Description – Populated with the Hold Type Description batch parameter value.
        - Payment Hold Type Department – Populated with the Payment Hold Type Department parameter value.
        - Payment Hold Type Unit – Populated with the Payment Hold Type Unit parameter value.
        - Start Check Number – Populated with the Check/EFT Number from the EFT Return table.
        - End Check Number – Populated with the Check/EFT Number from the EFT Return table.
        - Active flag – It will be set to True.
        - Scheduled Payment Date – If the Cancellation Type is set to “Reschedule”, it will be populated with the Rescheduled Payment Date batch parameter value if it is populated. If the Cancellation Type is set to “Reschedule”, it will be set to the Application Control Date if the Rescheduled Payment Date batch parameter is blank.
        - Comments – Populated with the text “Inserted by the Returned ACH Transaction Process”.
        - Write the Bank Account Code, EFT Number, Payee/Vendor Code, Payee/Vendor Legal Name and Amount to the report “Initiated Cancellations of Returned EFT Payments Report”.
4. Table Updates:

After inserting the records into Disbursement Cancellation Parameter, the job shall make the following updates to the EFT Return record:

- Set the Date EFT Cancel Initiated to the Application Control Date; and
- Set the System Message field to “Updated by the Returned ACH Transaction Process” on the EFT Reversal record

5. Generate Reports:

   After performing all the above steps successfully, the job will generate the Initiated Cancellations of Returned EFT Payments Report.

Job Codes

Since this job uses the common SysManUtil functionality, the job return code and the exception handling will be handled by the SysManUtil functionality and there is no specific requirement for this job.

Restartability Information

The job cannot be restarted in this run mode. If the job fails in any of the above steps, a new job should be scheduled after correcting the errors that caused the job to fail.

Selection Criteria

N/A

Sort Sequence

N/A

Problem Resolution

If the job ends with a return code of Failed and above, the job cannot be restarted. A new job has to be scheduled again.

The following table shows the various steps that the Initiate Disbursement Cancellation job goes through and the messages issued at each step.

Step 1: Parameter Validation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the table validations are successful, that is, there are no errors raised during validations.	N/A	N/A
Warning (4)	N/A	This step does not issue this return code.	
Non-Fatal Error (8)	N/A	This step does not issue this return code.	

Failed (12)	All of the required parameters are not entered. **Sample Message:** "Cancellation Type is required" **Recommendation:** Cancellation type parameter is should not be entered blank	If the required parameters are not entered, enter the required parameters and restart the job.	
	Entered Parameters are not valid. **Sample Message**: "Entered Cancellation Type is not a valid cancellation type" **Recommendation:** Cancellation Type must be valid.	If the parameters are invalid, enter the valid parameter and restart the job.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can either be restarted or a new job scheduled.	
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can either be restarted or a new job scheduled.	

Step 2: Selection of Records

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	Under the following conditions: If there are no records from the EFT Return Table with the Reversal Status set to "EFT Cancel Initiated" or "Reversal Failed by the Bank" and Date EFT Cancel initiated is blank. Sample Message:	Make sure that the parameter value entered is correct. If the parameter value is incorrect, then enter the correct parameter value and run the chain again.	N/A

	"No eligible records found on the EFT Return Table"		
Non-Fatal Error (8)	N/A	N/A.	N/A
Failed (12)	N/A	N/A	N/A
Terminated(16)	Job is terminated manually by the user	The reason for the termination needs to be investigated before rescheduling the entire chain.	N/A
System Failure(20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated before rescheduling the entire chain.	N/A

2.1.49 Scheduled Invoice Generation

Description

Scheduled Invoice Generation is a schedule-based process, established when a Recurring Payment Order is entered in the system, that will generate payments to vendors according to the predefined schedule. The process generates Recurring Invoices for the recurring amount and on the schedule identified. The Matching Manager chain job will then match the invoice to the order and generate Payment Request transactions.

Scheduled Invoice Generation:

The complete Scheduled Invoice Generation process is divided into the Update SIG Paid Amount process, the Create Recurring Invoices process, and the Add Subsequent records to SIG process.

Update SIG Paid Amount Process:

The process calculates the total payments processed against the Commodity line of a Recurring Payment Order during the reconciling period and updates the Paid Amount field on the Scheduled Invoice Generation (SIG) table.

Create Recurring Invoices Process has three steps:

1.Generate the Recurring Invoice transaction XML File

The process selects SIG records that are not yet invoiced and whose Scheduled Payment Date is less or equal to job run date. The selected SIG records are sorted by Recurring Order Transaction Code, Order Transaction Department, Order Transaction ID, Order Transaction Vendor Line, and Scheduled Payment Date and for each group a Recurring invoice transaction is generated.

These selected records are then processed and a RIN-XML file is generated.

2.Load Recurring Invoice (RIN) Transactions

The Recurring Invoice Load process uploads the XML File and creates Recurring Invoices Transactions.

3.Submit Recurring Invoice (RIN) Transactions

The Recurring Invoice Submit process submits all the Recurring Invoices transactions that were created in the previous step.

Add Subsequent records to SIG Process:

Once the Recurring Invoice transactions have been submitted to Final, the Add Subsequent records to SIG process adds a record to the SIG table for the next Schedule Sequence of the chosen Schedule on the Recurring Payment Order.

When to Run

The complete Scheduled Invoice Generation process can be run as part of the nightly cycle on the Scheduled payment dates. It can also be run on demand.

Steps to Run the Scheduled Invoice Generation Process (Job Sequence)

Step 1 – Enter reconciling information for the SIG records wherever applicable

Step 2 – Run Scheduled Invoice Generation Process

Major Input

Update SIG Paid Amount Process

Scheduled Invoice Generation Table (R_SCHED_INV_GEN)

PO Paid History Table (R_PO_PAID_HIST)

Create Recurring Invoices Process

Scheduled Invoice Generation Table (R_SCHED_INV_GEN)

Add Subsequent records to SIG Process

Scheduled Invoice Generation Table (R_SCHED_INV_GEN)

Recurring Payment Schedule Table (R_RECUR_SCHED_PYMT)

Schedule Sequence Table (R_SCHED_SEQ)

Output

Update SIG Paid Amount Process

Paid Amount on Scheduled Invoice Generation Table (R_SCHED_INV_GEN)

Create Recurring Invoices Process

Recurring Invoice (RIN) Transactions

Add Subsequent records to SIG Process

New records on Scheduled Invoice Generation Table (R_SCHED_INV_GEN) for next Scheduled Payment Date of the selected Schedule

Parameters

Create Recurring Invoices Process

The Recurring Invoice process is comprised of 3 batch jobs that are chained together. Each step in the process is described below.

Job	Parameter	Description	Default Value
Create Recurring Invoices	Transaction Code (DOC_CD)	Required Field. Defaulted to RIN. This serves as a Transaction Code for the XML	RIN
	Transaction Department Code (DOC_DEPT_CD)	Required Field if 'Use Transaction Department and Unit from Referenced Order (Y/N)' is set to "N". The Transaction Department to be used to set as Transaction Department on generated Recurring Invoice Transactions.	No Default
	Transaction Prefix (DOC_PFX)	The transaction prefix to be used with the transaction code and department to find an Automatic Transaction Numbering entry. Will appear in the beginning of all transaction ID's for all transactions created by the program.	No Default
	Transaction Status Code (DOC_STA_CD)	Required Field. Transaction Status Code (1-Held, 2-Ready) is set.	No Default
	Transaction Unit Code (DOC_UNIT_CD)	Required Field if Department Code is entered and 'Use Transaction Department and Unit from Referenced Order (Y/N)' is set to "N". The Transaction Unit to be used to set as Transaction Unit on generated Recurring Invoice Transactions	No Default

Job	Parameter	Description	Default Value
	Condition (COND)	Required Field. Condition to update the condition field on commodity Line of generated Recurring Invoice Transaction	No Default
	Reason (REAS)	Required Field Reason to update the condition field on commodity Line of generated Recurring Invoice Transaction	No Default
	Export File Name (EXP_FILE_NM)	Required Field. Export File Name is the name that will be used to create XML file	INDocs.xml
	Load File Name (LOAD_FILE_NM)	Required Field. Load Parameter file to contain all the load parameters required for loading the created XML file. This parameter value should be set to same as PARM_FILE parameter value of Load Transactions job.	INLoad.txt
	Submit File Name (SUBMIT_FILE_NM)	Required Field. Submit File to contain all the submit parameters required for submitting of the generated RIN Transactions which are in ready state. This parameter value should be set to same as PARM_FILE parameter value of Submit Transactions job.	INSubmit.txt

Job	Parameter	Description	Default Value
	Use Transaction Department and Unit from Referenced Order (USE_FROM_RFED_DOC)	Required Field, Use Transaction Department and Unit from Referenced Order(Y/N), When this parameter is set to 'N" then the User has to provide Transaction Department and Transaction Unit. When set to "Y" then Transaction Department and Transaction Unit are inferred from referencing Recurring Payment Order transaction.	No Default
	Export Location at Create Recurring Invoices Job (AMSEXPORT)	Required (** **Refer to Note**: Assumptions for SWBP on page no. 7)	No Default
	Parameter Location at Create Recurring Invoices Job (AMSPARM)	Required (** **Refer to Note**: Assumptions for SWBP on page no. 7)	No Default

Job	Parameter	Description	Default Value
Load Transactions	Load Parameter file (PARM_FILE)	Required Field. This parameter file is written by Create Recurring Invoices batch job at runtime. It contains load parameters to load the generated Recurring Invoice transactions from the Transaction XML.	$$AMSPARM$$/INLoad.txt

Job	Parameter	Description	Default Value
Submit Transactions	Submit Parameter File (PARM_FILE)	Required Field. This parameter file is written by Create Recurring Invoices batch job at runtime. It contains submit parameters to submit the generated Recurring Invoice transactions.	$$AMSPARM$$/INSubmit.txt

Add Subsequent records to SIG Process

Job	Parameter	Description	Default Value
Add Subsequent records to SIG	Commit block size (COMMIT_SIZE)	Optional Field. It is the number of SIG records committed at a time, which also determines the checkpoint update frequency. If value is not entered, the process defaults it to 100. This parameter can be used for performance tuning of the job depending upon server processing capability.	100

Sort Sequence

The XML file is created in the following order:

- Order Transaction Code
- Order Transaction Dept Code
- Order Transaction ID
- Order Transaction Vendor Line Number
- Schedule Payment Date
- Record Number of SIG record

Selection Criteria

The selection criteria for Create Recurring Invoices Process is:

Select all SIG records where Invoiced Status (INVD_STA_CD) = Not Invoiced And

Scheduled Payment Date <= job run date And

Hold Indicator is No And

Vendor Invoice Number is populated And

((Reconciling Period Type is Populated And

Reconcile From is populated And

Reconcile To is populated And

Reconcile Amount is populated And

Paid Amount is populated) OR

(Reconciling Period Type is blank))

The selection criteria for Update SIG Paid Amount Process is:

Select all SIG records where Invoiced Status (INVD_STA_CD) = Not Invoiced And

Reconciling Period Type (RECON_PRD_TYP) is populated And

Reconciling From Date (RECON_FRM) is populated And

Reconciling To Date (RECON_TO) is populated And

Vendor Invoice Number (VEND_INV_NO) is populated And

Paid Amount (PAID_AM) is blank.

The selection criteria for Add Subsequent records to SIG Process is:

Select all SIG records where Invoiced Status (INVD_STA_CD) = Recently Invoiced.

Problem Resolution

The whole process is implemented with checkpoints. If the process fails for any reason (such as the network is down or the server is down), then the process restarts from the last point where it stopped.

This batch program produces new Advantage transactions, which are subject to the line limit functionality constraints. Sites should ensure that they run this job with parameters set to ensure that the created transactions are within the line limit controls.

2.1.50 Select Payment Hold by TIN

Chain or Job Name	Select Payment Hold by TIN
Recommended Frequency	Nightly
Single Instance Required	Yes
Can be restarted?	Yes
Reports generated	• Payment Hold TIN Record Exclusion report • Payment Hold TINs with Matching Vendors report

Overview

The Select Payment Hold by TIN batch job selects records from the Payment Hold by TIN table to insert or update the corresponding records on the Payment Hold Maintenance table. This process runs after the Load Payment Hold by TIN batch process which inserts or updates records on the Payment Hold by TIN table.

Step 1: Parameter Validation

- This step verifies the batch job parameters: Commit Block Size, Select Block Size and Progression Counter. If all parameters are valid then the batch job moves ahead with the next step.

Step 2: Deleting the records

- The second step deletes records from the Payment Hold by TIN Table where the Updated by Hold Process field is set to *Hold Removed* and the Date Debt Removed is not null. The batch job also writes the count of records deleted message to the job log.

Step 3: Updating the records

- This step selects records from the Payment Hold by TIN table where the Date Debt Removed field contains a value and the Updated by Hold Process field is set to *Hold Applied*.
- For each selected record from the Payment Hold by TIN table, the batch job retrieves the associated Payment Hold Maintenance table record. The Payment Hold Maintenance record is updated as follows:
  - Set the Payment Hold Action to *Remove Hold*.
  - Set the Hold Removed By to Select Payment Hold by TIN Batch Process.
  - Set the Hold Removal Description to Remove by the Select Payment Hold by TIN Batch Process.
  - Set the Last Hold Approved By to Select Payment Hold by TIN Batch Process.
  - Set the Updated by Hold Process field to *Hold Removed* for associated records on the Payment Hold by TIN table using the Taxpayer ID Number (TIN) and TIN Type.

- The batch job writes the Payment Hold by TIN table record to the Payment Hold TINs with Matching Vendors report if the associated Payment Hold Maintenance table record is found or else the Payment Hold by TIN table record will be written to Payment Hold Records Exclusion Report.

Step 4: Inserting the records

- This step retrieves records from the Payment Hold by TIN table where Date Debt Added field contains a value, the Date Debt Removed field is blank and the Updated by Hold Process field is equal to *No Hold Applied*.
- For each selected record, the batch job infers the Vendor Code if the vendor code is blank. If multiple vendor codes are found for a combination of TIN and TIN Type, then multiple Payment Hold Maintenance records will be inserted.
- The batch job will not insert a record on the Payment Hold Maintenance table if the Vendor Code is excluded on the Disbursement Request Exclusion table.
- The following field values will be set on Payment Hold Maintenance table while inserting a new record on it:
    - Auto Generated Number with the automatically system generated sequential number.
    - Set the Hold Level to *Vendor*.
    - Set the Payment Hold Action to *Request Hold*.
    - Set the Vendor with the Vendor from the Payment Hold by TIN table record.
    - Set the Taxpayer Identification Number with the Taxpayer Identification Number (TIN) from the Payment Hold by TIN table record.
    - Set the TIN Type with the TIN Type from the Payment Hold by TIN table record.
    - Set the Legal Name with the Taxpayer Name from the Payment Hold by TIN table record.
    - Set the Hold Type with the Hold Type from the Payment Hold by TIN table record.
    - Set the Payment Hold Type Department with the Payment Hold Type Department from the Payment Hold by TIN table record.
    - Set the Payment Hold Type Unit with the Payment Hold Type Unit from the Payment Hold by TIN table record.
    - The Priority Hold Order is inferred from the Payment Hold Type by Department table based on Department, Unit, and Hold Type combination.
    - Set the Automated Payment Hold Source to *Select Payment Hold by TIN Batch Process*.
    - Set the Hold Requested By to Select Payment Hold by TIN Batch Process.
    - Set Hold Request Date to the Date Debt Added from the Payment Hold by TIN table record.
    - Set Hold Request Description to Inserted by the Select Payment Hold by TIN Batch Process.
    - Set the Last Hold Approved By to Select Payment Hold by TIN Batch Process.
    - Set the Street 1 with the Street 1 from the Payment Hold by TIN table record.
    - Set the Street 2 with the Street 2 from the Payment Hold by TIN table record.
    - Set the City with the City from the Payment Hold by TIN table record.

- Set the State/Province with the State/Province from the Payment Hold by TIN table record.
- Set the Zip Code/Postal Code with the Zip Code/Postal Code from the Payment Hold by TIN table record.
- Set the Country with the Country from the Payment Hold by TIN table record.
- Set the Phone Number with the Phone Number from the Payment Hold by TIN table record.
- Set the Email Address ID with the Email Address ID from the Payment Hold by TIN table record.
- The Print Payment Hold Notice Status will be set according to the Payment Hold Type by Department table value.

- Writes the selected Payment Hold by TIN table records to Payment Hold TINs with Matching Vendors Report.

The following are the process steps and progression messages for the Select Hold by TIN batch job:

Process Steps	Messages
1. Validate the Parameters	• At start of the step, “Validating batch parameters” and at the end of the step “Parameter validation completed” will be printed. Following messages will be printed based on different parameter inputs: • Invalid Value for Commit Block size parameter. • Using the default value 100 for commit block size parameter. • Invalid value for Select Block Size parameter. • Using the default value 100 for Select Block Size parameter. • Invalid value for progression counter parameter. • Using the default value 100 for progression counter parameter. • If parameters are invalid, then the job writes “Parameter validation failed“ to the job log.
2. Deletion of records	• At the start of this step, “Deleting record from the Payment Hold TIN Table” will be printed. • If the records to be deleted return 0 records, then “No records deleted.”

Process Steps	Messages
	will be printed. • If some records are deleted then "No. of records deleted:" will be printed with the deleted records number. • "Delete Completed." will be printed on completion of the step.
3. Update the record	• At the start of this step, "Updating records into Payment Hold Maintenance Table/Payment Hold Tin Table." will be printed. • If no records are found to be updated then "No records found on the Payment Hold by TIN table to be updated" • "Update Completed" will be printed on completion of the step.
4. Insert the record	• At the start of this step, "Inserting records into Payment Hold Maintenance Table" will be printed to the job log. • "No records found on the Payment Hold by TIN table with Updated by Hold Process field set to No Hold Applied" will be printed if no records found. • "Insert Completed" will be printed at the end of insert step. • Inserting the record completed.

Restartability

Job can be restarted. If the job fails, then on restart the check point needs to be read. Based on the checkpoint value, the job will restart the deletion of records or updating the records or inserting the records based on the step value which indicates where the last processing was when the process terminated.

Major Input

Data from the following tables:

- Payment Hold by TIN table (R_PYMT_HLD_TIN)
- Payment Hold Maintenance table (R_PYMT_HLD_MNTN)
- Vendor Customer table (R_VEND_CUST)
- Payment Hold Exclusion table (R_DISB_MGMT_RSTR)

- Payment Hold Type Department table (R_PYMT_HLD_TYP_DPT)

Batch Parameters

Parameter	Description	Default Value
Progression Message Block Size COMMIT_ SZ	Required. This field provides the information about the record being processed based on which a commit is done.	100
Progression Message Block Size SELECT_SZ	Required. The Select Block Size field provides the information about the select block size for queries.	100
Progression Message Block Size PROG_CTR_SZ	Required. This field provides the information about the record being processed.	100

Major Output

Following are the output of the batch job:

- Payment Hold Maintenance table (R_PYMT_HLD_MNTN)
- Payment Hold by TIN table (R_PYMT_HLD_TIN)
- Payment Hold TIN Records Exclusion report
- Payment Hold TINs with Matching Vendors report

Job Return Code

The following table shows the potential job return codes:

Return Code	Condition
Successful (1)	All the records in the Payment Hold by TIN Table are processed successfully.
Warning (4)	If no records were found on the Payment Hold by TIN Table that are eligible for deletion.
Non-Fatal Error (8)	This job does not issue this Return Code.
Failed (12)	The job fails under the following conditions: • Parameters are invalid • Technical/System failure
Terminated	This return code will be issued when the job is terminated by

Return Code	Condition
(16)	the user. When this job ends with a return of code *Terminated* subsequent jobs in the chain will be set to *Inactive*.
System Failure (20)	This return code will be issued when the job is terminated because of database server or network issues. When this job ends with a return of code *System Failure*, subsequent jobs in the chain will be set to *Inactive*.

Sort Sequence

The records will be ordered by TIN, TIN Type and Hold Type.

Selection Criteria

- Delete step: Delete records of Payment Hold by TIN table if Update Hold Process flag is set to *Hold Removed* and Date Debt Removed is not null.
- Update step: Select records of Payment Hold by TIN table if Update Hold Process flag is set to *Hold Applied* and Date Debt removed is not null.
    - Select the associated Payment Hold Maintenance table records from the Payment Hold by TIN table by matching TIN, TIN Type, and Vendor Code of the two tables.
    - Match Hold Request Date of the Payment Hold Maintenance record with Date Debt Added of the Payment Hold by TIN record.
    - Match Hold Level on Payment Hold Maintenance record to be *Vendor* level.
    - Match Action code on Payment Hold Maintenance record to be *Request Hold*.
    - Match Auto Generation flag to be *Select Payment Hold by TIN Batch Process*.
- Insert step: Select records from the Payment Hold by TIN table if the Update Hold Process flag is set to *No Hold Applied* and Date Debt Added is not null and Date Debt Removed is null.

Problem Resolution

The following table shows the possible return codes and recommendations for each processing step:

Step 1: Parameter Validation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All the parameters are validated successfully.	N/A	N/A

Possible Return Codes	Condition	Recommendation	Other Instructions
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Job failed due to fatal conditions.	In this step, the job can fail under the following conditions. 1) Encounters any runtime exceptions. 2) Failed during restart. If the job fails because of the runtime exceptions, investigate the exception reported by the process, resolve the error and restart the job.	Schedule a new job.
Terminated (16)	Job is terminated manually by the user.	Reason for the termination needs to be investigated. A new job can be scheduled.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.
System Failure (20)	When the job is terminated because of database server or network issues.	Reason for the System Failure needs to be investigated. A new job can be scheduled.	If another instance of the job has already been scheduled and ran successfully, then this job should not be restarted – only new job should be scheduled.

Step 2: Deleting the records

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	N/A	N/A	N/A

Warning (4)	If the number of records to delete from the Payment Hold by TIN Table is 0.	N/A	Write the error message “No eligible records found on the Payment Hold by TIN Table” to job log.
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Failed because of runtime exceptions for an unexpected situation.	Failure reason needs to be investigated before scheduling a new job.	Failed because of runtime exceptions for an unexpected situation.
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated and since the job can be restarted, the records to be deleted will be selected based on the Check Point value.	N/A
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated and since the job can be restarted, the records to be deleted will be selected based on the Check Point value.	N/A

Step 3: Updating the record

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	The record has been successfully updated.	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	Errors that can be generated due to some failed validations while updating the corresponding record on the Payment Hold Maintenance Table.	N/A	Get the error thrown by the Payment Hold by Maintenance table and then write it to the exception report.

Failed (12)	Failed due to issues in reading the data row.	Investigate the reason for the data row not being read and re-run the chain.	N/A
	Failed because of runtime exceptions for an unexpected situation.	Failure reason needs to be investigated before scheduling a new job.	N/A
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated and since the job can be restarted, the records to be updated will be selected based on the Check Point value.	N/A
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated and since the job can be restarted, the records to be updated will be selected based on the Check Point value.	N/A

Step 4: Inserting the record

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	The record has been successfully inserted.	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	Errors that can be generated due to some failed validations while inserting into the Payment Hold Maintenance Table.	N/A	Get the error thrown by the Payment Hold Maintenance table and then write it to the exception report.
Failed (12)	Failed due to issues in reading the data row.	Investigate the reason for the data row not being read and re-run the chain.	N/A
	Failed because of runtime exceptions for an unexpected situation.	Failure reason needs to be investigated before scheduling a new job.	N/A

Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated and since the job can be restarted, the records to be inserted will be selected based on the Check Point value.	N/A
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated and since the job can be restarted, the records to be inserted will be selected based on the Check Point value.	N/A

2.1.51 Stale Escheat Process

Chain or Job Name	Stale Escheat Process
Recommended Frequency	On demand
Single Instance Required	Yes
Can be restarted?	No
Reports generated	No

Overview

Once a check is issued, an entity usually establishes a certain number of days in which the check must be redeemed. A check is stale when it has not been redeemed within the established number of days. When this happens, an entity has the right to reclaim the funds in question until such time as the holder of the check claims the funds.

An escheat is the succession of abandoned property to the state. It is commonly associated with property that comes from the estate of a person dying without a will and without any known heirs. However, this concept has been broadened to include the recovery of any property that results from the failure of a person legally entitled to that property to make a valid claim against the holder of the property within a prescribed period of time. Consequently, the terms escheat and unclaimed property are used interchangeably. The types of property that are often unclaimed may include checking and savings accounts and even uncashed checks.

There are two methods of this post-disbursement activity. The first is that only one of the two methods is used. The second is where Stale is the first step and Escheating is the second step. Determining which of the two methods is controlled by two fields on the Expense tab of System Options. If only one of Stale Date Disbursements or Escheat Disbursements is set to *Yes,* then method 1 is in place. If both are *Yes,* then method 2 is in place. The Stale Escheat Process job only works with method 1. If using method 2, then look to the Stale Process, Escheat Pre-Selection and Escheat Update run sheets.

Description

This process is used to cancel the checks issued by the automated disbursement process. If the issued checks are unclaimed, they could undergo the following cancellation actions.

- STALE
- ESCHEAT

When to Run

On demand

Major Input

Check Reconciliation (CHREC - R_AP_CHK_RECON)

System Options - Expense (SOPT - R_GEN_SOPT)

Output

- Disbursement Cancellation Transaction (DC)
- Check Reconciliation (CHREC - R_AP_CHK_RECON)

Parameters

Parameter	Description	Default Value
Transaction Code (DOC_CD)	A required output parameter for the transaction code generated.	No Default
Department (DOC_DEPT_CD)	A required output parameter for transaction creation and reading Automatic Transaction Numbering.	No Default
Prefix (DOC_ID)	A required output parameter for transaction creation in the reading of Automatic Transaction Numbering.	No Default
Unit (DOC_UNIT_CD)	An optional output parameter for transaction creation, namely for security purposes.	No Default
Bank (BANK_ACCT_CD)	An optional selection parameter for selecting records from CHREC. If left blank, then all banks are selected. Multiple values allowed if separated by commas.	No Default
Commit Block Size (COMMIT_BLK)	A required performance parameter controlling the number of records processed in a block.	1000
Prefetch Block Size (PREFETCH_BLK)	A required performance parameter used in record selection blocks.	1000

Sort Sequence

None

Selection Criteria:

Select records from Check Reconciliation where the following are true:

- Status is *Disbursed*, *Warranted*, or *Undelivered*.
- Check/EFT Number is not all zeros.
- Issue Date is not null.

- Transaction Code is not ‘CW’ or ‘CE’.
- The difference between the Current Date and the Issue Date is greater than the number of days specified on System Options (SOPT) for Stale Days or Escheat Days
- Optionally – Bank matches the Bank(s) entered in the job parameter.

Problem Resolution

The process has a rerun procedure as follows:

The Stale Escheat process picks up those records from the Check Reconciliation table with the status of Outstanding and the difference between Current Date and Issue Date greater than the number of days specified in the SOPT table for Stale days or Escheat Days. Then it uses Copy Corresponding to create a DC Transaction in a draft version and submits the transaction.

If there are any errors while creating a DC draft version or on submitting the draft version, the process leaves that particular check and it moves forward to process the rest of the checks. The check that has errors is rectified, the process is rerun, and then the process successfully creates the DC Transaction.

If both Stale Date Disbursements and Escheat Disbursements are *Yes* on SOPT, the job will stop at the parameter editing stage. Run the Stale Process, Escheat Pre-Selection, and/or Escheat Update batch jobs instead.

If a DC transaction is created successfully in draft but there are problems in submitting the transaction, the process leaves that transaction and moves forward to process the rest of the checks. When you rerun the process, a new DC transaction is created.

This batch program produces new Advantage transactions, which are subject to the line limit functionality constraints. Sites should ensure that they run this job with parameters set to ensure that the created transactions are within the line limit controls.

2.1.52 Stale Process

Chain or Job Name	Stale Process
Recommended Frequency	Nightly Cycle or can be run on demand
Single Instance Required	Single
Can be restarted?	Yes
Reports generated	Yes

Overview

When using both stale and escheat functionality (the Stale Dated Disbursements and Escheat Disbursements are set to *Yes* on System Options: Expense) and not just one or the other (one field is *Yes* and the other is *No* where the Stale Escheat batch process is used), the Stale Process job is run to change the Status on Check Reconciliation (CHREC) to *Stale Dated* where the number of days in the Stale Days on System Options has passed since the Issue Date. There is no Disbursement Reclassification (DC) transaction posted to update the Stale Dated Payable and Stale Dated Cash balance sheets. It is just a change of Status on CHREC for reporting reasons. The Cleared Date is also set on the CHREC record to the current application date.

At a later date, if still outstanding, the Escheat Pre-Selection and Escheat system processes will select the CHREC for potential escheat processing and a DC transaction.

This job consists of the following processes and executed in the following sequence:

1. Parameter Validation
2. Check Reconciliation Selection/Update and Reporting

Process Steps	Messages
1. Parameter Validation	• Run Started • Parameter is listed • If the parameter is invalid, an error is logged
2. Check Reconciliation Selection/Update and Reporting	• Reports output folder mapped • Paths listed • Rendering report started • Rendering report completed • Number of Check Records updated check status to Stale Dated: ### • Run Ended

Restartability

After parameter validation, if the job fails it can be restarted.

Major Input

- Check Reconciliation (CHREC / R_AP_CHK_RECON)

Note: The default values listed are those delivered with the software. Actual values may vary based on your site's setup.

Parameter	Description	Default Value
Select Block (Select Block)	A required performance parameter that controls the number of CHREC records selected for processing in one pass. There is no default if left blank.	1000

Major Output

- Check Reconciliation (CHREC / R_AP_CHK_RECON)
- Stale Process Report

Return Codes

The following return codes are issued for the Collection Payment process.

Return Code	Condition
Successful (1)	One or more records were found to mark as State Dated.
Warning (4)	No records were found for processing.
Non-Fatal Error (8)	The job does not use this return code.
Failed (12)	Failed parameter validations.
Terminated (16)	The job was terminated by the user.
System Failure (20)	The job was terminated because of database server or network issues.

Sort Criteria

Bank (BANK_ACCT_CD) and Check/EFT Number (CHK_EFT_NO)

Selection Criteria

- Status is Disbursed or Warranted
- Check/EFT Number is not all zeros
- Transaction Type is AD or MD
- Issue Date is not null and the difference between the Application Date – Issue Date = > Stale Days and Status does <> Stale Dated.

Problem Resolution

No database restore is required. Correct the problem and rerun the job executing the program. No restoration of datasets or files from backups is required for this program.

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the parameters are validated successfully and records selected and processed successfully.	N/A	N/A
Warning (4)	Job ended with a Warning because there are no eligible records found "No records selected for the file":	Verify the records based on the selection logic	
Non-Fatal Error (8)	NA	NA	
Failed (12)	Job failed due to parameter validation failure or any Runtime Exception.	In this step, the job can fail under the following two conditions. • Encounters any runtime exceptions • Parameter Validation • Stale Pre-Selection are not applicable as stale Flag is not set in SOPT Verify parameters and restart the job. If the job fails because of runtime exceptions, investigate the exception reported by the process, resolve the error, and restart the job.	
	Failed while restarting the job since another instance of the job has already been run successfully. **Sample Message**: Cannot restart the job since another instance of this job has already been run successfully.	**Recommendation:** Schedule a new job.	

Possible Return Codes	Condition	Recommendation	Other Instructions
Terminated (16)	Job is terminated manually by the user.	Reason for the termination needs to be investigated. Schedule a new job.	Job should be rescheduled after correcting the problem
System Failure (20)	When the job is terminated because of database server or network issues.	Reason for the System Failure needs to be investigated. Schedule a new job.	Job should be rescheduled after correcting the problem.

2.1.53 Escheat Pre-Selection Process

Job Name	Escheat Pre-Selection
Recommended Frequency	The job is most likely done on demand after running the Stale Process
Single Instance Required	Yes
Can be restarted?	Yes
Reports generated	Escheat Pre-Selection Report

Overview

When using both stale and escheat functionality (the Stale Dated Disbursements and Escheat Disbursements are set to *Yes* on System Options: Expense) and not just one or the other (one field is *Yes,* one is and the other is *No,* and the Stale Escheat process is used), the Stale Process job is first run to change the Status on Check Reconciliation (CHREC) to *Stale Dated* where the number of days in the Stale Days on System Options has passed since the Issue Date. At a later date and if still outstanding, this Escheat Pre-Selection process is used to load the Escheat Pre-Selection (ESCH) page for manual review and then the Escheat Update process is used to create a Disbursement Reclassification (DC) for those selected records that were automatically and manually approved for escheating.

This job consists of the following processes and executed in the following sequence:

1. Parameter Validation
2. Escheat Pre-Selection Processing
3. Report Generation

Process Steps	Messages
1. Parameter Validation	Run Started • Parameters are listed • If the parameter is invalid, an error is logged
2. Escheat Pre-Selection Processing	• Number of Check Records that were pre-selected : ###
3. Report Generation	The following messages will be displayed: • Reports output folder mapped • HTML report file path: <path> • PDF report file path: <path> • Rendering report started. • Rendering report completed.

Process Steps	Messages
	• If the selection returns 0 records, the following message is issued: “No records found for report”.

Major Input

- System Options – Expense (SOPT / R_EXP_SOPT)
- Check Reconciliation (CHREC / R_AP_CHK_RECON)
- Automatic Disbursement Accounting Lines for Exception Fund(s) (AD_DOC_ACTG)

Parameters

Note: The default values listed are those delivered with the software. Actual values may vary based on your site’s setup.

Parameter	Description	Default Value
Auto Approval Threshold (AUTO_AMNT)	An optional output parameter that is a dollar amount threshold whereby checks or warrants less than or equal to the specified dollar amount have the Approval Status set to *Approved* on ESCH. Those not marked as *Approved* have a status of *Eligible*.	0
Fund Code Excluded (FUND_CD)	An optional selection parameter of one or more funds to exclude a check or warrant when at least one accounting line contains the fund on an Automated Disbursement or Manual Disbursement transaction type. Multiple values must be comma-separated.	<blank>
Select Block Size (SELECT_BLK)	A required performance parameter to controls the number of CHREC records selected.	1000

Major Output

- Escheat Pre-Selection (R_AP_CHK_ESCH)
- Escheat Pre-Selection Report

Sort Sequence

None

Selection Criteria:

Select records from Check Reconciliation where the following are true:

- Transaction Type = AD or MD

- Status = Stale Dated
- Application Date - Issue Date > Escheat Days
- No Accounting Line of a check or warrant has an excluded fund

Problem Resolution

The process can be restarted on failure. If failure is due to parameter validation, then a new job must be submitted.

The following table shows the possible return codes and recommendations for each processing step:

Step 1: Parameter Validation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the parameters are validated successfully.	N/A	N/A
Warning (4)	N/A	This step does not issue this return code.	This step does not issue this return code.
Non-Fatal Error (8)	N/A	This step does not issue this return code.	This step does not issue this return code.
Failed (12)	The job failed because of runtime exceptions for an unexpected situation.	The reason for the failure needs to be investigated before re-scheduling the job.	Reschedule job with appropriate parameter values.
Terminated (16)	The job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can be rescheduled.	Schedule a new job.
System Failure (20)	The job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can be rescheduled.	Schedule a new job.

Step 2: Escheat Pre-Selection Processing

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful	Records are selected	N/A	N/A

(1)	successfully.		
Warning (4)	No records were found that matched selection criteria	N/A	N/A
Non-Fatal Error (8)	N/A	This step does not issue this return code.	This step does not issue this return code.
Failed (12)	The job fails if the Escheat Disbursements is not *Yes* on SOPT for the current FY. **Sample message:** Escheat are not applicable as Flags are not set in the SOPT	Ensure that an entry is made in SOPT for the current FY where the Escheat Disbursements flag is enabled.	Restart the job or schedule a new job.
	The job failed because of runtime exceptions for an unexpected situation.	The reason for the failure needs to be investigated before rescheduling the job.	Reschedule job with appropriate parameter values.
Terminated (16)	The job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can be rescheduled.	Schedule a new job.
System Failure (20)	The job is terminated because of database server or network issues.	The reason for the system failure needs to be investigated. The job can be rescheduled.	Schedule a new job.

2.1.54 Escheat Update

Chain or Job Name	Escheat Update
Recommended Frequency	This process can be run on demand and only after running the Escheat Pre-Selection process
Single Instance Required	Yes
Can be restarted?	No
Reports generated	Escheat Update report and Escheat Update Exception report

Overview

The Escheat Update process escheats Automatic Disbursement (AD) and Manual Disbursement (MD) transaction types for certain records present on Escheat Pre-Selection (ESCH), based on settings defined on System Options (SOPT). This process must be run after the Escheat Pre-Selection process, which loads records into ESCH for manual review. Once that review is performed and all updates have been made, this process can be run to select the records that were automatically or manually approved for escheating and to create transactions to record the escheat.

This job consists of the following processes and executed in the following sequence:

1. Parameter Validation
2. Selection and Update of Records

Process Steps	Messages
1. Parameter Validation	Run Started • Parameters are listed • If the parameter is invalid, an error is logged.
2. Selection and Update of Records	• Number of DCs submitted ## • Number of DCs rejected ##

Major Input

Calendar Date (CLDT/ R_CLDT)

System Options (SOPT/ R_EXP_SOPT)

Escheat Pre-Selection (ESCH/ R_AP_CHK_ESCH)

Parameters

Note: The default values listed are those delivered with the software. Actual values may vary based on your site's setup.

Parameter	Description	Default Value
Commit Block Size (COMMIT_BLK)	A required performance parameter that defines the number of records committed in an instance.	1000
Transaction Code (DOC_CD)	A required parameter to define the documents that are to be created. For example, 'DC' is the recommended value so that Disbursement Reclassification transactions are created to record the escheat.	(no default)
Department Code (DOC_DEPT_CD)	A required parameter used to locate the Automatic Transaction Numbering record for transaction creation.	(no default)
DocumentID Prefix (DOC_ID)	An optional parameter used to locate the Automatic Transaction Numbering record for transaction creation.	(no default)
Transaction Unit Code (DOC_UNIT_CD)	An optional parameter to be used in transaction creation for security purposes.	(no default)
Pre-Fetch Block size (PREFETCH_BLK)	A required performance parameter that defines the number of records selected in one group processing. Default is 1000. If left blank, a null value will be used by the job.	1000

Major Output

Disbursement Cancellation (DC)

Check Reconciliation (CHREC/ R_AP_CHK_RECON)

Escheat Pre-Selection (ESCH/ R_AP_CHK_ESCH)

Escheat Update Exception report

Escheat Update report

Job Return Code

The following return codes are issued for the process:

Return Code	Condition
Successful (1)	When selected ESCH records were processed successfully
Warning (4)	When generated transactions were rejected
Non Fatal Error (8)	This job doesn't issue this return code
Failed (12)	When failed to commit transactions and failed parameter validations
Terminated (16)	The job was terminated by the user
System Failure (20)	The job was terminated because of database server or network issues

Sort Criteria

Check Number (CHK_EFT_NO)

Selection Criteria

Check Status is 'Approved'

Check Number is not '000000000000000'

Issue Date is not null

Transaction Code is 'MD' or 'AD'

Problem Resolution

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the parameters are validated successfully and records selected are processed successfully.	N/A	N/A
Warning (4)	Job ended with a Warning • DC transactions created were failed to submit	Review AD or MD and ESCH data for the transaction created.	

Possible Return Codes	Condition	Recommendation	Other Instructions
Non Fatal Error (8)	N/A	N/A	
Failed (12)	Job failed due to parameter validation failure or any Runtime Exception.	In this step, the job can fail under the following two conditions. 11. Encounters any runtime exceptions 12. Parameter Validation If the job fails because of the runtime exceptions, investigate the exception reported by the process, resolve the error and restart the job.	
	Failed while restarting the job since another instance of the job has already been run successfully. **Sample Message**: Cannot restart the job since another instance of this job has already been run successfully.	Schedule a new job.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated.	Job should be rescheduled after correcting the problem
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated.	Job should be rescheduled after correcting the problem.

2.1.55 Tax Form Printing Process

Chain or Job Name	Tax Form Printing Process
Recommended Frequency	On Demand
Single Instance Required	Yes
Can be restarted?	No
Reports generated	Yes

Overview

The Tax Form Printing Process job will print IRS tax forms for data captured in Advantage Financial for tax forms supported by Advantage. The Tax Form Printing Process job is used to print single forms from each of the tax form reporting tables (the tables are listed under Major Input). The job is also used for bulk printing of tax forms. The Tax Form Printing Process job can be executed in one of two modes, online and offline.

- **Online Mode** - From one of the tax form reporting tables, the user selects the tax form record to be printed and selects the Print Tax Form action. User is transitioned to the Online Tax Form Printing Parameters page to enter the parameters required to print the form. On transition to the Online Tax Form Printing Parameters (XX196) page, the system uses the Tax Year and the Form Type from the selected tax form record and infers the printing parameters from the Tax Form Options and Parameters (TAXOPT) record corresponding to the Tax Year and Form Type. User may override the inferred parameter values such as the Print Resource ID to specify a local printer in which to print the tax form. On verifying the correct print parameters, the user will select the Print action. On selection of the action, the system executes the Tax Form Printing Process in the background to print the tax form using the parameter specified on the Online Tax Form Printing Parameters page. Note, in the Online Mode, the process does not check the form for thresholds.
- **Offline Mode** - In the Offline Mode, the Tax Form Printing Process job selects the Tax Form Options and Parameters records for the Tax Year and Form Type specified in the job parameters of the batch process. Selected TAXOPT records are used to select and process tax form reporting table records for each of the tax forms specified on the selected TAXOPT records. The process checks forms for thresholds, if the option is specified on the TAXOPT record, and creates a form xml file that is used by BIRT to generate the appropriate tax forms.

Process Steps	Messages
1. Parameter Validation	• Offline Mode: • If Tax Year is not entered, the following message will be displayed on the job log: “Tax Year is required”. • If Tax Year entered is not valid on FY table, following message will be displayed on the job log: “Tax Year must be valid on the FY table”. • If no record exists on TAXOPT table for the Tax Year entered, the following error message will be displayed on the job log: “At least one

Process Steps	Messages
	record must exist on the TAXOPT table for the Tax Year". • If Print Copy A is not empty and value is not equal to *Yes* or *No*, the following error message will be displayed on the job log: "Print Copy A must be equal to *Yes* or *No* or blank". • If Print Copy A value is *Yes* and Form Type is not null and has more than one value, the following error message will be displayed on the job log: "Form Type is required and must be a single value. Multiple Form Types are not allowed since Copy A is printed on preprinted form stock". • If Print Copy A value is *Yes* and Tax Year is not null and corresponding record on TAXOPT for Tax Year and Form Type entered does not have Copy A set to *Yes*, the following error message will be displayed on the job log: "Copy A must be set to Yes on the TAXOPT record associated with the Tax Year and Form Type". • If no Form Type is entered, the following message will be displayed on the job log: "Job will be run for all Form Types of the Tax Year." • If Form Type is not null, each of the tax forms specified in the comma delimited field must be valid on the TAXOPT table for the Tax Year, else the following message will be displayed on the job log: "No 'Active for Printing' TAXOPT record found for the tax year <<Tax Year>> and form type <<Form Type>>". • If Commit block size is less than *0*, the following message will be displayed on the job log: "Commit block size should be greater than zero". • If Commit block size is non-numeric, the following message will be displayed on the job log: "Commit block size should be a positive integer". • If Progression message block size is less than *0*, the following message will be displayed on the job log: "Progression message block size should be greater than zero". • If Progression message block size is non-numeric, the following message will be displayed on the job log: "Progression message block size should be a positive integer". • If Tax forms to be printed per file is non-numeric, the following message will be displayed on the job log: "Tax forms per file should be a positive integer". • Online Mode: • No messages are issued when job runs in Online Mode.
2. Selection of Records	• The following message is displayed in the job log to display the number of records selected: "Total number forms selected for printing for Form Type<<Form Type>> = <<Count>>". • If No records are selected for processing, the following message will be displayed on the job log: "No tax form reporting records were

Process Steps	Messages
	selected Form Type<<Form Type>>".
3. Process Records	• After the successful creation of XML file, the following message will be displayed: "Creation of Form Type<<Form Type>> XML completed successfully." • Total count of records printed will be displayed on the job log: "Total number of selected forms printed for Form Type<<Form Type>> = <<Count>>" • Total count of records bypassed will be displayed on the job log: "Total number of selected forms bypassed for Form Type<<Form Type>> = <<Count>>"

Major Input

Online Mode

- Form Reporting tables:

R_FORM_1042S_RPT

R_FORM_1099A_RPT

R_FORM_1099C_RPT

R_FORM_1099G_RPT

R_FORM_1099INT_RPT

R_FORM_1099MIS_RPT

R_FORM_1099NEC_RPT

R_FORM_1099R_RPT

R_FORM_1099S_RPT

- Online Tax Form Printing Parameters (ONLN_TAXF_PRN_PARM)
- 1099 Reporting Payer table (R_1099_RPT_PYRINFO)
- Tax Forms Options and Parameters (R_TAX_OPT_PARM)

Offline Mode

- Form Reporting tables (tables listed under Online Mode)
- Tax forms options and parameters R_TAX_OPT_PARM
- 1099 Reporting Payer table (R_1099_RPT_PYRINFO)

Batch Parameters

Online Mode

Parameter	Description	Default Value
Tax Year	The year for which the form needs to be printed.	Inferred from the selected record to be printed.

Parameter	Description	Default Value
(TAX_YR)	Required. Should be valid on FY table.	
Form Type (FORM_TYP)	The Form Type for the Forms to be printed. Required.	Inferred from the selected record to be printed.
Form Name (FORM_NM)	The Name of the form to be printed. Required.	Inferred from the selected record to be printed.
Print Job Code (PRNT_JOB_CD)	This field specifies the job code that is being printed. Required.	Inferred from the TAXOPT for the Taxy Year and Form Type for the selected record to be printed.
Print Resource ID (PRNT_RSRC_ID)	This field is used to specify which Print Resource ID from the Print Resource Setup should be used for a request. Required.	Inferred from the TAXOPT for the Taxy Year and Form Type for the selected record to be printed.
View Forms (VIEW_FORMS)	Optional. This parameter specifies how to produce the output. The value of *Y* is used to send forms output to job status inquiries (must use a PDF print resource). The value of *N* is used to send forms output directly to the print resource destination. Value values are Y or N.	N

Offline Mode

Parameter	Description	Default Value
Export Location (AMSEXPORT)	Required. Specifies the export location of the printing job.	$$AMSROOT$$/ExportImport

Parameter	Description	Default Value
Commit Block Size (COMMIT_SIZE)	Required. Specifies the number of records after which the job should perform a commit. It should be a positive integer.	1000
Tax Year (TAX_YR)	Required. The year for which the forms are to be printed. It should be valid on FY table and should have at least one corresponding record on TAXOPT.	No Default
Form Type (FORM_TYP)	Optional. The Form Type for which the Forms are to be printed. If entered, each of the form types specified in the comma delimited field must be valid on the TAXOPT table for the Tax Year. The value must be one of the following in upper case: • 'N' – 1042S • 'E' – 1099N • 'G' – 1099G • 'I' – 1099I • 'M' – 1099M • 'S' – 1099S • 'A' – 1099A • 'C' – 1099C • 'R' – 1099R If null, all form types are eligible for selection. If Print Copy A is set to Yes, only one form type can be specified.	No Default
Print Copy A	Optional.	No Default

Parameter	Description	Default Value
(PRNT_COPY_A)	Specifies that only Copy A will be printed for the form type(s). When Copy A is printing, other Copies may not be printed in the same run because the data is printed on a paper stock by form type. Another job must be scheduled to print other Copies. Valid values are Yes and No.	
Progression Message Block Size (PROG_CTR_SZ)	Optional. Specifies the number of records after which the job should log progression messages. It should be a positive integer.	1000
View Forms (VIEW_FORMS)	Optional. Specifies how to produce the output. Value values are Y or N. The value of *Y* is used to send forms output to job status inquiries (must use a PDF print resource). The value of *N* is used to send forms output directly to print resource destination.	N
Include Account Numbers That Start With (ACCT_NUM_INCL)	Optional. Specifies the prefix of the Account Numbers that will be selected. This parameter is in addition to configuration on TAXOPT. Multiple values can be entered comma delimited. Letters must capitalize and end with an asterisk. Examples: A*	No Default

Parameter	Description	Default Value
	A*,BC* A*,BC*,123*	
Exclude Account Numbers That Start With (ACCT_NUM_EXCL)	Optional. Specifies the prefix of the Account Numbers to exclude during record selection. This parameter is in addition to configuration on TAXOPT. Multiple values can be entered if comma delimited. Letters must capitalize and end with an asterisk. Examples: A* A*,B01* A*,B01*,123*	No Default
Tax forms to be printed per file (TAX_FORMS_PER_FILE)	Required. Specifies the number of records which will be accommodated by job in tax forms. It should be a positive integer.	3000

Major Output

XML files for the tax records

JSON files for the tax records

Form Reporting tables (see all tables listed above in Input section)

Tax Form Printing report

Job Return Code

The following table shows the potential job Return Codes for the Tax Form Printing Process:

Return Code	Condition
Successful (1)	All the selected records are processed successfully.
Warning (4)	No eligible records found. This could be because of the following reasons: No valid TAXOPT record found for the tax year <<Tax Year>>
Non-Fatal Error (8)	N/A

Failed (12)	The job may fail under the following conditions: • Parameters validation failed. • Tax Year is required. • Failed to initialize additional member variables • Run time exceptions for unexpected situations.
Terminated (16)	This return code will be issued when the job is terminated by the user.
System Failure (20)	This return code will be issued when the job is terminated because of database server or network issues.

Sort Criteria

For Online Mode, no sorting is required.

For Offline Mode, the system sorts the selected joined records by Tax Year and Form Type. Within each Tax Year and Form Type group, the system uses the sorting criteria and the sorts by the precedence specified on the respective TAXOPT record for the Tax Year and Form Type. A sort order precedence value of *0* means the field is not used in sorting. A sort order precedence other than *0* indicates that the records are sorted in descending order. This means a sort order of *1* would have a higher precedence than a sort order of *4*. When all sort order precedence values are *0*, the default sort order is Reporting Payer, TIN, and TIN Type. Note: the logic on the TAXOPT page does not allow two or more fields with the same sort order precedence unless the value is *0*.

Selection Criteria

Online Mode

The Tax Form Printing Process batch job uses the following logic in the Online Mode to select tax form reporting records for printing:

1. Select Tax Form Report record – The system performs a lookup for records on the tax form reporting table (Form Table) and Reporting Payer (1099RP) table where the following conditions are true:
   - Tax Year (Form Table) equals Tax Year on the Online Tax Form Printing Parameters (XX196) page.
   - Form Type (Form Table) equals Form Type on the Online Tax Form Printing Parameters (XX196) page.
   - Reporting Payer (Form Table) equals Reporting Payer on 1099 Reporting Payer Information table (1099RP), Payee TIN (Form Table) equals Payee TIN on the selected tax form reporting record.
   - Payee TIN Type (Form Table) equals Payee TIN Type on the selected tax form reporting record.
   - Account Number (Form Table) equals Account Number on the selected tax form reporting record.
   - Current Record flag (Form Table) equals to *Yes.*

- Last Print Date (Form Table) is Null or Last Print Date (Form Table) is not Null and the Reprint flag (Form Table) equals *Yes.*
- Exclude Print flag (Form Table) equals *No.*
- Reporting Payer (Form Table) is equal to Reporting Payer (1099RP).

Offline Mode

The Tax Form Printing Process batch job uses the following logic in the Offline Mode to select tax form reporting records for printing:

1. Retrieve Tax Form Printing Options - The process performs a look up to the TAXOPT table using the Tax Year and Form Type parameters as record selection criteria. If the Form Type parameter is null, then the process uses only the Tax Year to perform lookup of the TAXOPT table. The system retrieves all records matching the TAXOPT selection criteria where the Active for Printing is set to *Yes*.
   - If no TAXOPT records are found, the system displays the following message, “No TAXOPT record found for the tax year <tax year> and form type <form type displayed value>”, in the job log and terminate job with a Return Code of *Warning*.
2. Select Tax Form Reporting Records - For each TAXOPT record retrieved, the system performs a lookup for records on the tax form reporting table (Form Table) and Reporting Payer (1099RP) table (SQL JOIN) where the following conditions are true:
   - Tax Year (Form Table) equals Tax Year (TAXOPT); AND
   - Form Type (Form Table) equals Form Type (TAXOPT); AND
   - Reporting Payer (Form Table) is in the list of Reporting Payers on the Include Reporting Payer(s) parameter on the TAXOPT record OR Include Reporting Payer(s) parameter is Null; AND
   - Current Record flag (Form Table) equals *Yes*; AND
   - Last Print Date (Form Table) is Null) OR Last Print Date (Form Table) is not Null and the Reprint flag (Form Table) equals *Yes*; AND
   - Exclude Print flag (Form Table) equals *No*; AND
   - Reporting Payer (Form Table) equals Reporting Payer (1099RP)
   - If the Threshold Checking flag (TAXOPT) equals *Yes* AND Threshold Met equals *Yes* OR if the Threshold Checking flag (TAXOPT) equals *No*.
   - Account Number (Form Table) starts with the value(s) from the ACCT_NUM_INCL parameter (examples; A*,BC*,D123)
   - Account Number (Form Table) does not start with the value(s) from the ACCT_NUM_EXCL parameter (examples; E*,FG*,H123*)
3. If no tax form reporting records are found, the system issues the following message, “No tax form reporting records were selected” in the job log and terminates the job with a Return Code of *Warning*.

Problem Resolution

The process cannot be restarted on failure; however new job can be scheduled by fixing the issue.

The following table shows the possible return codes and recommendations for each processing steps:

Step 1: Parameter Validation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	N/A	This step does not issue this return code.	This step does not issue this return code.
Non-Fatal Error (8)	N/A	This step does not issue this return code.	This step does not issue this return code.
Failed (12)	Required Parameters are not valid. **Sample Message:** "Tax Year must be valid on the FY table.	Enter valid Tax Year.	Reschedule job with appropriate parameter values.
	The job failed because of runtime exceptions for an unexpected situation.	The reason for the failure needs to be investigated before re-scheduling the job.	Reschedule job with appropriate parameter values.
Terminated (16)	The job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can be rescheduled.	Schedule a new job.
System Failure (20)	The job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can be rescheduled.	Schedule a new job.

Step 2: Selection of Records

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful	Successful	N/A	N/A

(1)			
Warning (4)	If no TAXOPT records are found for the parameters entered: **Sample Message:** "No valid TAXOPT record found for the tax year <<Tax Year>>"	Verify the parameters entered and reschedule the job.	Reschedule job with appropriate records on CHREC.
Non-Fatal Error (8)	N/A	This step does not issue this return code.	This step does not issue this return code.
Failed (12)	The job failed because of runtime exceptions for an unexpected situation.	The reason for the failure needs to be investigated before rescheduling the job.	Reschedule job with appropriate parameter values.
Terminated (16)	The job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can be rescheduled.	Schedule a new job.
System Failure (20)	The job is terminated because of database server or network issues.	The reason for the system failure needs to be investigated. The job can be rescheduled.	Schedule a new job.

Step 3: Processing Of Records

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All records are processed successfully.	N/A	N/A
Warning (4)	N/A	This step does not issue this return code.	This step does not issue this return code.
Non-Fatal Error (8)	N/A	This step does not issue this return code.	This step does not issue this return code.
Failed (12)	If any exception occurs while processing records. **Sample Message:**	Ensure the Tax Report Records are valid.	Reschedule job after corrections.

Possible Return Codes	Condition	Recommendation	Other Instructions
	"Exception in processRecords(): <<Error Message>>"		
	The job failed because of runtime exceptions for an unexpected situation.	The reason for the failure needs to be investigated before rescheduling the job.	Reschedule job with appropriate parameter values.
Terminated (16)	The job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can be rescheduled.	Schedule a new job.
System Failure (20)	The job is terminated because of database server or network issues.	Reason for the system failure needs to be investigated. The job can either be restarted or a new job can be scheduled.	Schedule a new job.

2.1.56 Treasury Offset Process

Chain or Job Name	Treasury Offset Process
Recommended Frequency	Nightly before the AD Chain
Single Instance Required	Yes
Can be restarted?	No
Reports generated	No

Overview

The Treasury Offset Process identifies scheduled payment requests eligible for offset, retrieves any outstanding debt from an external debt database for the vendor (taxpayer ID) and loads that debt into Advantage for intercepting in the next disbursement run.

This process depends on the ability to make SOAP requests. SOAP is a lightweight protocol based on XML used for data interchange between applications. Sending a SOAP request over HTTP allows web services to talk to client applications.

The processing steps of the job are:

1. Validate batch parameters
2. Purge Temporary Debt table (R_TEMP_DEBT)
3. Retrieve ADV30params.ini for web service user ID and password
4. Select eligible combinations of TIN/TIN Type with eligible payments for intercept from Disbursement Request (DISRQ)
5. Create SOAP Request Messages for selected combinations of TIN/TIN Type
6. Call the external collections Web Service real time to return debt information, encrypting the data transmission using a SSL certificate
7. If the external collections system sends a SOAP Response Message successfully to Advantage:
    a) Insert debts received from the external collections system into a Temporary Debt table
    b) Increment the counter tracking the number of debts inserted to the Temporary Debt table
    c) Use the Temporary Debt table to update existing debts on Intercept Request (INTR)
    d) Use the Temporary Debt table to insert new debts to INTR
8. If the external collections system does not respond to Advantage within 60 seconds, then Advantage attempts to check for a response from the external collections system again. Advantage repeats this attempt up to number specified in a batch parameter.
9. If Advantage does not receive any response from external collections system after the specified attempts, a SOAP error in the job log indicating that the Web Service call timed out and stops the job.

Process Steps	Messages
Parameter Validation	• A list of parameters are displayed (followed by each associated value) • Batch Parameter Validation Completed
Delete Temporary Table Records	<number> Debt records deleted from R_TEMP_DEBT table
Selection of Records	<number> TINs selected for processing
Create Outbound Stream	<number> TINs sent to external collections system via Web Call

Process Inbound Stream	• Response received from external collections system via Web Call
	• If SOAP fault occurs, the following displays:
	  • Soap Fault has occurred
	  • Soap Fault, Calling soap request again after 60 seconds. Call count <count>
	• If the external Web Service returns an error, the following may display:
	  • External Web Service returned an error. The fault code is: <value>
	  • External Web Service returned an error. The fault string is: <value>
	  • External Web Service returned an error. The Node Name is: <value>
	  • External Web Service returned an error. The Node Value is: <value>

Update Temporary Table	Total Inserts to R_TEMP_DEBT table: <number>
Updates to Intercept Request	• Started updating debts on the INTR table if the debt exists • Completed updating debts on the INTR table • Started inserting new debts into the INTR table • Completed inserting debts into the INTR table • Total Updates to INTR table: <number> • Total Inserts to INTR table: <number>

Restart Information

If the job failed due to any reason, schedule a new job after correcting the errors that caused the job to fail.

Major Input

Tables

The following is a list of tables used as input for the Treasury Offset Process:

- Unit (UNIT / R_UNIT)
- Department (DEPT / R_DEPT)
- Disbursement Parameters (DISPA / R_AP_DISB_PARM)
- Disbursement Request (DISRQ / R_AP_DISB_RQST)
- Intercept Request (INTR / R_AP_INCT_RQST)
- Intercept Disbursement Exception (INTDBEX / R_AP_DISB_EXCP)
- Vendor/Customer (VCUST / R_VEND_CUST)
- Application Parameters (APPCTRL / IN_APP_CTRL)

1. TRSRY_INTRCPT_DEBT_ENTY_SYS_1
2. TRSRY_INTRCPT_DEBT_ENTY_SYS_2
3. TRSRY_INTRCPT_DEBT_ENTY_SYS_3
4. TRSRY_INTRCPT_SOAP_HDR_URL,
5. TRSRY_INTRCPT_ENVLP_NMS_URL,
6. TRSRY_INTRCPT_BODY_NMS_URL,
7. TRSRY_INTRCPT_XSD_NMS_URL,
8. TRSRY_INTRCPT_SOAP_SECURITY_URL,
9. TRSRY_INTRCPT_SOAP_USER_TOKEN_URL,
10. TRSRY_INTRCPT_SOAP_PSWD_URL

Other

- Two ADV30Params.ini file parameters are required to execute this process: TRS_OFST_WS_USERID and TRS_OFST_WS_PSWD.
- Response Debt Input Stream - The contents of the inbound stream include TIN, TIN Type, Claim Status, Comments, Debt Type Code, Debtor Name (mapped to Last Name of INTR), Due Date, Entity Code, Fixed Amount, Greater/Lesser value, Percentage, Outstanding Amount, Third Party Name (mapped to First Name on INTR), Treasury Debt Identifier.

Batch Parameters

Note: The default values listed are those delivered with the software. Actual values may vary based on your site's setup.

Parameter	Description	Default Value
Application Control Parameter Name (APPCTRL_PARM_ENTY_PRFL)	Required APPCTRL parameter names that detail the various external debt system(s) used. See Application Parameters in the input section: TRSRY_INTRCPT_DEBT_ENTY_SYS_# for valid values.	(No Default)

	Specify multiple values with commas.	
INTR Referenced Account Line Default (AR_RF_DOC_AL_NO)	Required Accounting Line Number value populated on INTR records created.	(No Default)
INTR Referenced Transaction Code (AR_RF_DOC_CD)	Required Transaction Code value populated on INTR records created.	(No Default)
Number of Times External Call is Made (MAX_CALLS_PER_MSG)	Required number of calls to the external collections Web Service in case of a SOAP fault.	(No Default)
Maximum Requests per Call (MAX_REQ_PER_MSG)	Required performance parameter for the number of Taxpayer requests per a Web Service message submission.	500
Special Payment Indication (SPEC_PMT_TYP)	Optional indication, based on one or more Transaction Unit codes, used in the SOAP request for matching external debts that indicates the payment as 'special' instead of a 'normal.' Specify multiple values with commas.	(No Default)
External Debt URL (TREASURY_WEBSERVICES_URL)	Required URL for external debt Web Service.	(No Default)
File Location (FILE_LOC)	Required location where the process stores the reporting file: FIN_TREASURY_OFFSET_DISRQ_DATA.txt file.	$$AMSROOT$$/ExportImport
DISPA Transaction Codes Selected (DISB_PARM_CD)	Required list of transaction codes used when reading DISPA.	AD,EFT
Commit Block Size (COMMIT_BLOCK)	Required performance parameter to control the number of INTR records committed in a save. If left blank, 100 defaults.	100
Select Block Size (SELECT_BLOCK)	Required parameter to control DISRQ and Temporary Debt Table record selection for a processing round. If left blank, 1000 defaults.	1000
Progression Counter (PROG_CTR_SZ)	Required parameter to control when the job log updates with a progress status. If left blank, 5000 defaults.	5000

Major Output

Tables

- Intercept Request (INTR / R_AP_INCT_RQST)

- Temporary Debt Table (R_TEMP_DEBT) - Attributes of the Temporary table include the following:

Attribute	Data Details
Taxpayer Identification Number (TIN)	Variable Character (9)
Taxpayer Identification Type (TIN_TYP)	Variable Character (1)
First Name (FRST_NM)	Variable Character (40)
Last Name (LAST_NM)	Variable Character (30)
Outstanding Amount (OUT_AM)	Number (17,2)
Due Date (DUE_DT)	Date
Entity Code (ENTY_CD)	Variable Character (4)
Debt Type Code (DEBT_TYP_CD)	Variable Character (4)
Claim Status (CLM_STA)	Number (10)
Fixed Intercept Amount (FIX_AM)	Number (17,2)
Intercept Percentage (INCT_PC)	Number (7,4)
Greater/Lesser (GTR_LSR)	Number (10)
Comments (CMNT)	Variable Character (250)
Referenced Transaction Code (AR_RF_DOC_ID)	Variable Character (20)

Other

FIN_TREASURY_OFFSET_DISRQ_DATA.txt – TIN, TIN_Type, Legal Name, Alias Name, Address Line 1, First Name, Last Name, Payment Type, Total Payment Amount

Payment Request Output Stream - TIN, TIN Type, Legal Name, Alias Name, Address Line 1, First Name, Last Name, Transaction Unit, and Line Amount.

Job Return Code

The following table shows the potential return codes.

Return Code	Condition	Recommendation
Successful (1)	Successful	N/A
Warning (4)	Job ends with a return code of Warning. No eligible records found. This could be because of the following reasons: No active Disbursement Parameter record exists No records found on Disbursement Request for the parameters entered on the Disbursement Parameters selected.	Verify that the AD and EFT records on DISPA are active and are set with the proper date ranges.
Non-Fatal Error	This job returns the Non-Fatal Error Code	Notify the external collection

Return Code	Condition	Recommendation
(8)	if: • External collection system sends a duplicate debt. • External collection system sends a debt with a Debt Type not defined in Advantage • External collection system's Web Call is down	system when any of these errors occur.
Failed (12)	Job ends returns the Failed Error Code if: • Parameters are invalid data types, invalid values, or missing. • Login information not found or not valid in the ini-file • Run time exceptions for unexpected situations.	Investigate and remedy the reason for the job failure before scheduling a new job.
Terminated (16)	Job ends with a return code of Terminated if a user terminates the job.	Investigate the reason for the termination before scheduling a new job.
System Failure (20)	Job ends with a return code of System Failure when the job terminates because of database server or network issues.	Investigate the reason for the System Failure before scheduling a new job.

Sort Criteria

- Outbound stream is in the order found on DISRQ.
- Inbound stream is in the order sent by the external debt systems.
- Reporting text file is in the order sent by the external debt systems.
- Updates to INTR are in the order sent by the external debt systems.

Selection Criteria

The Treasury Offset Process selects the records from Disbursement Request (DISRQ) based on the criteria specified on one or more selected Disbursement Parameters (DISPA) records. The steps for selection are:

1. Identify active DISPA records that match the DISB_PARM_CD batch parameter. For those DISPA records, the process selects applicable payment requests that are present on Disbursement Request (DISRQ).

2. For each active DISPA record, use the following criteria to identify payment requests on DISRQ that are eligible for disbursement:

   a) Transaction Code specified in DISB_PARM_CD (such as AD or EFT) will be used to lookup records on DISPA.

b) Disbursement Type (Check, Warrant or EFT) on the DISPA record is equivalent to the Disbursement Type on the DISRQ record.

c) Bank Account Code on the DISPA record equals the Bank Account Code on the DISRQ record.

d) From Date on the DISPA record <= the Scheduled Payment Date on the DISRQ record.

e) To Date on the DISPA record >= to the Scheduled Payment Date on the DISRQ record.

3. For selected DISRQ records, exclude the selected DISRQ records that meet the following condition:

   - The Disbursement Management Hold on the DISRQ record = 1 (true) or User Hold = 1 (true).

4. For selected DISRQ records that passed the above conditions, exclude the DISRQ record if the TIN / TIN Type are null on DISRQ and VCUST are null.

5. Using the FY, TIN, and TIN Type from DISRQ, verify if an exception exists on Intercept Disbursement Exception (INTDBEX / R_AP_DISB_EXCP). If the TIN/TIN Type for the given FY and at least one of the following fields Fund, Department, Unit, Object, Sub-Object, Appr Unit, Transaction Department Code, Transaction Unit Code, or Transaction ID on the payment matches a record on INTDBEX then exclude the payment from selection (or) process the offset amount based on the Offset Percentage and prioritize the record based on the Rank field.

6. For selected DISRQ records that passed the above conditions, validate VCUST as follows:

   a) An active vendor has an Active Status other than Not Active on VCUST. If Not Active, then exclude selected DISRQ record.

   b) An approved vendor has a Vendor Approval Status other than Incomplete on VCUST. If Incomplete, then exclude selected DISRQ records.

   c) A held vendor has Hold Payment indication is true. If true, then exclude selected DISRQ records.

7. The DISRQ records that pass all the above conditions are used to create the Request Message by TIN and TIN Type combination as follows:

   a) If the Unit code is listed in the SPEC_PMT_TYP batch parameter, then Advantage assigns ‘N’ as the Payment Type for non-regular payments. Otherwise, Advantage assigns ‘R’ as the Payment Type for regular payments. The SOAP Request uses the N or R in the PMT_TYP field.

   a) The Treasury Offset process changes the values for TIN_TYP from 1 (EIN) and 2 (SSN/ITIN/ATIN) to F (EIN) and S (SSN/ITIN/ATIN).

   b) For each selected TIN/TIN Type combination, the process sums the Accounting Line Amount (LN_AM) fields for selected respective DISRQ records to populate the Total Payment Amount (TOT_PMT_AMT).

   c) For each selected record that has TIN/TIN Type on VCUST, the process populates the applicable Legal Name (LGL_NM), Alias (ALIAS_NM), First Name (FRST_NM) and Last Name (LAST_NM) from VCUST.

   d) For each selected record that has TIN/TIN Type on DISRQ, the process populates the applicable Legal Name (LGL_NM) and Alias (ALIAS_NM) from DISRQ.

2.1.57 Upload 1099 External Reported Income Process

Chain or Job Name	Upload 1099 External Reported Income Process
Recommended Frequency	On Demand
Single Instance Required	Yes
Can be restarted?	No
Reports generated	1099 External Transaction Exception report

Overview

The Upload 1099 External Reported Income Process is responsible for uploading external 1099 reported income by 1099 Reporting Payer, Taxpayer ID Number and Type, updating the consolidated income on the 1099 External Reported Income Table (1099ER) which can then be used by the Offline 1099 Process to create text files. The files may be used by a third party tool or Advantage Tax Reporting to generate IRS forms and transmit data files (original submission and corrections) to the IRS. Interfaced 1099 transactions with the Correction Return Indicator of *Blank* are inserted onto the 1099 External Reported Income table as new records. The 1099 transactions with matching records on the 1099 External Reported Income table (i.e. same Calendar Year, Reporting Payer, Form Type, TIN and TIN Type combination) are written to an Exception Report.

The records will validate values upon save. The records uploaded to the 1099ER table will have a Processing Indicator of *New*. The batch parameters are used for validation. The TIN does not have to exist on the 1099I table. TIN Type should be a value of 1 (EIN) or 2 (SSN).

Interfaced 1099 transactions with the Correction Return Indicator of “C” or “G” will only replace non-key fields of respective records on the 1099 External Reported Income table. These 1099 transactions will not be inserted onto the 1099 External Reported Income table as new records if there are no matching records (i.e. same Calendar Year, Reporting Payer, Form Type, TIN and TIN Type combination). These records will be written to an Exception Report.

The exception report is generated by populating Reporting Payer, TIN, TIN Name, TIN Type, Department, Form Type, and Amounts from the input file.

The chain job involves the following Steps:

1. Convert Flat File
2. Flat to XML
3. Load XML
4. Upload_1099_External_Reported_Income

Major Input

- Input file (File to Upload)
- AP_RPT_1099_XINC_TMP Table (records uploaded from input file)

Major Output

- AP_RPT_1099_XINC Table

- Upload 1099 External Reported Income exception report

Job Return Code

The following table shows the potential return codes for the batch job.

Return Code	Condition
Successful (1)	All the selected records are successfully added on report.
Warning (4)	No eligible records found.
Non-Fatal Error (8)	If records are added on report with few exceptions i.e. not all selected records are added on report.
Failed (12)	The job will fail under the following conditions: • Parameters are invalid • Run time exceptions for unexpected situations. When this job ends with a return of code *Failed*, subsequent jobs in the chain will be set to *Inactive*.
Terminated (16)	This return code will be issued when the job is terminated by the user.
System Failure (20)	This return code will be issued when the job is terminated because of database server or network issues.

Problem Resolution

In case job fails, fix the issue and schedule a new job.

Upload 1099 External Reported Income Process: Convert Flat File

Job Name	Convert Flat File
Recommended Frequency	On Demand.
Single Instance Required	Yes.
Can be restarted?	No.
Reports Generated	No.

Overview

This process will convert the data file to a flat file with delimiter separated values which will be an input to the next job, Flat to XML.

Major Input

- Input file (File to convert to comma-separated values). Sample file layout fields and positions can be found in the "Sample Input File for the Convert Flat File batch job" section of this run sheet.

Major Output

- Output file (Flat file with delimiter separated values)

Process Steps	Messages
1. Parameter Validation	• Validating Batch Parameters • Parameters are valid or invalid depending on the Validation. If the parameter is invalid, the invalid value will be displayed in the log. • If the required parameters are not provided then <<PARM_NM>> is required message is added on Job Log. • Parameter validation completed.
2. Processing Data File	• Read the data file provided by the client and create a flat file with delimiter separated values compatible.

Batch Parameters

Parameter	Description	Default Value
AMSEXPORT	Optional Export directory	$$AMSROOT$$/ExportImport
AMSIMPORT	Optional Import directory	$$AMSROOT$$/ExportImport
ATTRIBUTE_LIST	Attribute list	256
ATTRIBUTE_LIST_1	Attribute list	256
ATTRIBUTE_LIST_2	Attribute list	256
INPUT_FILE_NM	Input File Name	N/A
OUTPUT_FILE_NM	Output File Name	N/A
DELIMITER	Delimiter	^

Parameter Validation:

In this step, the process will verify the parameters. If the parameter validation is successful, the job will go to the next step. Otherwise, the job will end with a return code of *Failed*.

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	N/A	This step doesn't issue this return code.	This step doesn't issue this return code.
Non-Fatal Error (8)	N/A	This step doesn't issue this return code.	This step doesn't issue this return code.
Failed (12)	Required Parameters are not entered **Sample Message:** File To Upload cannot be blank.	Enter the File To Upload.	A new job can be scheduled after correcting the parameters.
	Failed because of runtime exceptions for an unexpected situation.	The reason for the failure needs to be investigated before scheduling the new job.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The new job can be scheduled.	
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The new job can be scheduled.	

Processing Data File:

Based on the size of the columns, the data file provided by the external agency will be read and written to another data file as comma separated values. The Flat file created from this job will be an input to the next job in the chain.

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful	All the records from input file are	N/A	N/A

Possible Return Codes	Condition	Recommendation	Other Instructions
(1)	successfully written to another flat file as delimiter separated values.		
Warning (4)	No eligible records were found.	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Job failed due to fatal conditions.	In this step, the job can fail under the following condition. 1) Encounters any runtime exceptions and If the job fails because of the runtime exceptions, investigate the exception reported by the process, resolve the error and schedule a new job.	Schedule a new job.
Terminated (16)	Job is terminated manually by the user.	Reason for the termination needs to be investigated. A new job can be scheduled.	Schedule a new job.
System Failure (20)	When the job is terminated because of database server or network issues.	Reason for the System Failure needs to be investigated. A new job can be scheduled.	Schedule a new job.

Sample Input File for the Convert Flat File batch job:

Fields in Sample File			Positions from the 1st Job	
Used by 1st Chain Job	**Field Names**	**Length**	**Start**	**End**
CALENDAR YEAR	CY	4	1	4
TAXPAYER IDENTIFICATION NUMBER (TIN)	TIN	9	5	13
TIN TYPE	TIN_TYP	1	14	14

RECIPIENT NAME 1	TIN_NM_1	40	15	54
RECIPIENT NAME 2	TIN_NM_2	40	55	94
RECIPIENT ADDRESS	TIN_AD	40	95	134
RECIPIENT CITY NAME	TIN_CITY_NM	30	135	164
RECIPIENT STATE	TIN_ST	2	165	166
RECIPIENT ZIP	TIN_ZIP	10	167	176
FORM TYPE INDICATOR	FORM_TYP_IND	1	177	177
IRS REPORTED FLAG	IRS_RPT_FL	1	178	178
COMMENTS	CMNT	100	179	278
1099 REPORTING PAYER	RPT_1099_PYR	4	279	282
DATE OF PAYMENT	DT_PYMT_1099R	10	283	292
REPORTING DEPT CODE	RPT_DEPT_CD	4	293	296
REPORTING UNIT CODE	RPT_UNIT_CD	8	297	304
REPORTING LOCATION CODE	RPT_LOC_CD	4	305	308
CORRECTED RETURN INDICATOR	CORR_RET_IND	1	309	309
CORRECTED TRANSACTION INDICATOR	CORR_TRAN_IND	1	310	310
DEPARTMENT	DEPT_CD	4	311	314
BOX 1	BOX_1_INC_AM	20	315	334
BOX 2	BOX_2_INC_AM	20	335	354
BOX 2A	BOX_2A_INC_AM	20	355	374
BOX 3	BOX_3_INC_AM	100	375	474
BOX 4	BOX_4_INC_AM	20	475	494
BOX 5	BOX_5_INC_AM	20	495	514
BOX 6	BOX_6_INC_AM	20	515	534
BOX 7	BOX_7_INC_AM	20	535	554

BOX 8	BOX_8_INC_AM	20	555	574
BOX 9	BOX_9_INC_AM	20	575	594
BOX 9B	BOX_9B_INC_AM	20	595	614
BOX 10	BOX_10_INC_AM	20	615	634
BOX 10A	BOX_10A_INC_AM	20	635	654
BOX 10B	BOX_10B_INC_AM	20	655	674
BOX 11	BOX_11_INC_AM	20	675	694
BOX 12	BOX_12_INC_AM	20	695	714
BOX 13	BOX_13_INC_AM	20	715	734
BOX 14	BOX_14_INC_AM	20	735	754
BOX 15	BOX_15_INC_AM	60	755	814
BOX 15A	ADDL_1_INC_AM	17	815	831
BOX 15B	ADDL_2_INC_AM	17	832	848
BOX 16	BOX_16_INC_AM	20	849	868
BOX 17	BOX_17_INC_AM	20	869	888
BOX 18	BOX_18_INC_AM	20	889	908

Note: If the Manage TIN by Department parameter on the Application Parameters (APPCTRL) table is set to *True*, the Department Code can be provided in the positions 311 through 314, otherwise left blank.

The valid Form Type Indicator (FORM_TYP_IND) values are as per below chart:

Forms	**Valid Form Type value**
1099-G	G
1099-INT	I
1099-MISC	M
1099-S	S
1099-A	A
1099-C	C
1099-R	R
1099-NEC	E

Upload 1099 External Reported Income Process: Flat to XML

Job Name	Flat to XML
Recommended Frequency	On Demand.
Single Instance Required	Yes.
Can be restarted?	No.
Reports Generated	No.

Overview

This process will convert the flat file generated by the previous job to an XML file based on the attribute list provided in the parameter.

Major Input

- Input file (File with delimiter separated values)

Major Output

- Output file (XML file)

Process Steps	Messages
1. Parameter Validation	• Validating Batch Parameters. • Parameters are valid or invalid depending on the Validation. If the parameter is invalid, the invalid value will be displayed in the log. • If the required parameters are not provided then <<PARM_NM>> is required message is added on Job Log. • Parameter validation completed.
2. Processing Flat File	• Read the data file created by the previous job and based on the attribute provided in the parameters create an XML file.

Batch Parameters

Parameter	Description	Default Value
AMSEXPORT	Optional Export Location at Flat To XML Job	$$AMSROOT$$/ExportImport
AMSIMPORT	Optional	$$AMSROOT$$/ExportImport

	Import Location at Flat To XML Job	
AMSPARM	Optional Parameter Location at Flat To XML Job	$$AMSROOT$$/Parms
ATTRIBUTE_LIST	Attribute list	256
ATTRIBUTE_LIST_1	Attribute list	256
ATTRIBUTE_LIST_2	Attribute list	256
DATAOBJECT_NAME	Data Object Name	N/A
DELIMITER	Delimiter	^
FLAT_FILE	Optional. Flat File Name	N/A
XML_FILE	Optional. XML File Name	N/A

Parameter Validation:

In this step, the process will verify the parameters. If the parameter validation is successful, the job will go to the next step. Otherwise, the job will end with a return code of *Failed*.

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	N/A	This step doesn't issue this return code.	This step doesn't issue this return code.
Non-Fatal Error (8)	N/A	This step doesn't issue this return code.	This step doesn't issue this return code.
Failed (12)	Required Parameters are not entered. **Sample Message:** File To Upload cannot be blank.	Enter the File To Upload.	A new job can be scheduled after correcting the parameters.
	Failed because of runtime exceptions for an	The reason for the failure needs to be investigated before scheduling the new	

	unexpected situation.	job.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The new job can be scheduled.	
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The new job can be scheduled.	

Processing Flat File:

Based on the attribute list provided as a parameter, this job will create an XML file which will be used as an input to the next job in the chain.

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All the records from input file are successfully written to an XML file.	N/A	N/A
Warning (4)	No eligible records were found.	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Job failed due to fatal conditions.	In this step, the job can fail under the following condition. 1) Encounters any runtime exceptions and If the job fails because of the runtime exceptions, investigate the exception reported by the process, resolve the error and schedule a new job.	Schedule a new job.
Terminated (16)	Job is terminated manually by the user.	Reason for the termination needs to be investigated. A new job can be scheduled.	Schedule a new job.

Possible Return Codes	Condition	Recommendation	Other Instructions
System Failure (20)	When the job is terminated because of database server or network issues.	Reason for the System Failure needs to be investigated. A new job can be scheduled.	Schedule a new job.

Upload 1099 External Reported Income Process: Load XML

Job Name	Load XML
Recommended Frequency	On Demand.
Single Instance Required	Yes.
Can be restarted?	No.
Reports Generated	No.

Overview

This step will upload the entries provided in the XML file to the 1099ER Temp table.

Major Input

- Input file (XML File)

Major Output

- Output file (1099ER Temp table)

Process Steps	Messages
1. Parameter Validation	• Validating Batch Parameters. • Parameters are valid or invalid depending on the Validation. If the parameter is invalid, the invalid value will be displayed in the log. • If the required parameters are not provided then <<PARM_NM>> is required message is added on Job Log. • Parameter validation completed.
2. Processing XML File	• Upload the entries in the 1099ER temp table.

Batch Parameters

Parameter	Description	Default Value
ACTN_CD	Action Code	N/A
AMSIMPORT	Optional Import Location at Flat To XML Job	$$AMSROOT$$/ExportImport
CLIENT_NM	Optional Client name for Report	N/A
COMMIT_BLOCK	Optional. Commit Block Size	2000
XML_FILE	Optional. XML File Name	N/A

Parameter Validation:

In this step, the process will verify the parameters. If the parameter validation is successful, the job will go to the next step. Otherwise the job will end with a return code of *Failed*.

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	N/A	This step doesn't issue this return code.	This step doesn't issue this return code.
Non-Fatal Error (8)	N/A	This step doesn't issue this return code.	This step doesn't issue this return code.
Failed (12)	Required Parameters are not entered. **Sample Message:** File To Upload cannot be blank.	Enter the File To Upload.	A new job can be scheduled after correcting the parameters.
	Failed because of runtime exceptions for an unexpected situation.	The reason for the failure needs to be investigated before scheduling the new job.	
Terminated	Job is terminated manually	The reason for the termination needs to be	

(16)	by the user.	investigated. The new job can be scheduled.	
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The new job can be scheduled.	

Processing XML File:

In this step the System Maintenance Utility will perform the Insert table action in the 1099ER Temp table for the records present in the XML file generated from the previous job.

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All the records from input file are successfully inserted on 1099ER temp table.	N/A	N/A
Warning (4)	No eligible records were found.	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Job failed due to fatal conditions.	In this step, the job can fail under the following condition. 1) Encounters any runtime exceptions and If the job fails because of the runtime exceptions, investigate the exception reported by the process, resolve the error and schedule a new job.	Schedule a new job.
Terminated (16)	Job is terminated manually by the user.	Reason for the termination needs to be investigated. A new job can be scheduled.	Schedule a new job.
System Failure (20)	When the job is terminated because of database server or network issues.	Reason for the System Failure needs to be investigated. A new job can be scheduled.	Schedule a new job.

Upload 1099 External Reported Income Process: Upload 1099 External Reported Income

Job Name	Upload 1099 External Reported Income
Recommended Frequency	On Demand.
Single Instance Required	Yes.
Can be restarted?	No.
Reports Generated	Yes. Exception report.

Overview

This job will insert records into the 1099ER table based on the Correction Indicator.

Major Input

- Input file (1099ER Temp Table)

Major Output

- Output file (1099ER table)

Process Steps	Messages
Parameter Validation	• Validating Batch Parameters. • Parameters are valid or invalid depending on the Validation. If the parameter is invalid, the invalid value will be displayed in the log. • If the required parameters are not provided then <<PARM_NM>> is required message is added on Job Log. • Parameter validation completed.
Process Records	• If SOPT setup is for Single Reporting Payer and the records from 1099ER temporary table have *Reporting Payer* value different than the one specified on SOPT. • Records are added on exception report as Reporting Payer is not matching with SOPT value.
Summary of records added on report	• Total number of records processed: <<count>>
Deletion of records from temporary table	• All 1099ER temp records are deleted.

Batch Parameters

Parameter	Description	Default Value
CLIENT_NM	Optional Client Name	N/A
AMSEXPORT	Optional Export directory	$$AMSROOT$$/ExportImport
COMMIT_SIZE	Optional. Commit Block Size	2000
PROG_CTR_SIZE	Optional. Progression Counter Size	5000
SEL_BLK_SIZE	Optional. Select Block Size	5000

Parameter Validation:

In this step, the process will verify the parameters. If the parameter validation is successful, the job will go to the next step. Otherwise the job will end with a return code of *Failed*.

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	N/A	This step doesn't issue this return code.	This step doesn't issue this return code.
Non-Fatal Error (8)	N/A	This step doesn't issue this return code.	This step doesn't issue this return code.
Failed (12)	Required Parameters are not entered **Sample Message:** File To Upload cannot be blank.	Enter the File To Upload.	A new job can be scheduled after correcting the parameters.
	Failed because of runtime exceptions for an unexpected situation.	The reason for the failure needs to be investigated before scheduling the new job.	

Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The new job can be scheduled.	
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The new job can be scheduled.	

Process Records:

If the Correction Return Indicator is blank

And if matching record for Calendar Year, Reporting Payer, Form Type, TIN and TIN Type combination is found on 1099ER table then

Add the record on exception report

Else the record will be inserted on 1099ER table

If the Correction Return Indicator is “C” or “G”

And if matching record for Calendar Year, Reporting Payer, Form Type, TIN and TIN Type combination is found on 1099ER table then

Update the 1099ER record by replacing all non-key attributes.

Else the record will be added on exception report.

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All the selected records are inserted or updated on 1099ER table.	N/A	N/A
Warning (4)	No eligible records were found.	N/A	N/A
Non-Fatal Error (8)	Not all records are inserted or updated on 1099ER successfully.	N/A	N/A
Failed (12)	Job failed due to fatal conditions.	In this step, the job can fail under the following condition. 1) Encounters any runtime exceptions and If the job fails because of the runtime exceptions, investigate the exception reported	Schedule a new job.

Possible Return Codes	Condition	Recommendation	Other Instructions
		by the process, resolve the error and schedule a new job.	
Terminated (16)	Job is terminated manually by the user.	Reason for the termination needs to be investigated. A new job can be scheduled.	Schedule a new job.
System Failure (20)	When the job is terminated because of database server or network issues.	Reason for the System Failure needs to be investigated. A new job can be scheduled.	Schedule a new job.

Summary of records added on report

The report will be generated by listing Reporting Payer, TIN, TIN Type, Form Type, Correction Return Indicator and amounts

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Exception report was successfully generated.	N/A	N/A
Warning (4)	No eligible records were found.	N/A	N/A
Non-Fatal Error (8)	Not all records are inserted or updated on 1099ER successfully.	N/A	N/A
Failed (12)	Job failed due to fatal conditions.	In this step, the job can fail under the following condition. 1) Encounters any runtime exceptions and If the job fails because of the runtime exceptions, investigate the exception reported by the process, resolve the error and schedule a new job.	Schedule a new job.

Possible Return Codes	Condition	Recommendation	Other Instructions
Terminated (16)	Job is terminated manually by the user.	Reason for the termination needs to be investigated. A new job can be scheduled.	Schedule a new job.
System Failure (20)	When the job is terminated because of database server or network issues.	Reason for the System Failure needs to be investigated. A new job can be scheduled.	Schedule a new job.

Deletion of records from temporary table

This step will delete all the records present in the 1099ER Temp table.

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	The 1099ER Temp table entries are successfully deleted.	N/A	N/A
Warning (4)	No eligible records were found.	N/A	N/A
Non-Fatal Error (8)	The database could not be connected.	N/A	N/A
Failed (12)	Job failed due to fatal conditions.	In this step, the job can fail under the following condition. 1) Encounters any runtime exceptions and If the job fails because of the runtime exceptions, investigate the exception reported by the process, resolve the error and schedule a new job.	Schedule a new job.
Terminated (16)	Job is terminated manually by the user.	Reason for the termination needs to be investigated. A new job can be scheduled.	Schedule a new job.

Possible Return Codes	Condition	Recommendation	Other Instructions
System Failure (20)	When the job is terminated because of database server or network issues.	Reason for the System Failure needs to be investigated. A new job can be scheduled.	Schedule a new job.

2.1.58 Update Print Status of Zero Dollar Checks

Job Name	Update Print Status of Zero Dollar Checks
Recommended Frequency	After Automated Disbursement Chain and before Disbursement Printing.
Single Instance Required	Yes
Can be restarted?	No
Reports generated	Yes – Audit Report

Overview

When a disbursement that was fully intercepted in the AD Chain, resulting in a zero-dollar check/warrant, should be printed but other zero-dollar checks should not be (credit memo applications), this Update Print Status of Zero-Dollar Checks job will update the Print Status (PRN_STA_IND) on the Automatic Disbursement header to *Printed.* This ensures the Disbursement Printing job will only print zero-dollar checks/warrants when there was a full intercept. The print is essential to inform the entity of the event.

- If the status is *Ready for Printing*, change the status to *Printed*
- If the status is *Ready for Reprinting*, change the status to *Reprinted*
- If the status is *Ready for Generation*, change the status to *Generated*

An Audit Report is created listing those transactions and check numbers where the Print Status was updated to suppress printing. When there are no such records, the report contains a single line: No Records found matching the selection criteria to update the status.

Note: when using this process, the Print $0 Amount Check indication is *true* on System Options.

Major Input

- AD Transaction Header (AD_DOC_HDR)

Batch Parameters

Parameter	Description	Default Value
Transaction Code (PRNT_ZERO_DOC_CD)	A required selection parameter used to select AD header records. Multiple values are allowed if comma-separated.	AD, EFT

Major Output

- AD Transaction Header (AD_DOC_HDR)
- Audit Report (PrintZeroChecks)

Job Return Code

The following table shows the potential job return codes for the job Submit.

Return Code	Condition
Successful (1)	All of the validations are performed successfully and no errors as reported. Update Print Status of Zero Dollar Checks Job completed successfully.
Warning (4)	No records found matching the selection criteria to update the status
Non-Fatal Error (8)	Not Applicable for this job.
Failed (12)	Parameter validation failed, Parameter: Transaction Code The above is encountered when an invalid Transaction code is entered apart from DOC_TYP = AD.
Terminated (16)	This Return Code is issued when the job is terminated by the user. When the job ends with a Return Code of Terminated, subsequent jobs in the chain are set to inactive.
System Failure (20)	This Return Code is issued when the job is terminated because of database server or network issues. When this job ends with a Return Code of System Failure, subsequent jobs in the chain are set to inactive.

Sort Sequence

N/A

Selection Criteria

For the given disbursement transaction codes, the Check Amount must be $0 and Total Intercepted Amount must equal $0.

Problem Resolution

The job cannot be restarted. If the job fails in any of the above steps, a new job should be scheduled after correcting the errors that caused the job to fail.

The following table shows the possible return codes and recommendations for each processing step specific to the job in the chain. For general errors and recommendations, refer to the “SMU Transaction Submit Job” run sheet in the *CGI Advantage Financial – Utilities Run Sheets* guide.

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	At least one AD DOC HDR was found to update.	N/A	N/A

Possible Return Codes	Condition	Recommendation	Other Instructions
Warning (4)	No records were found to update.	Verity that there were no zero-dollar checks from the credit memo application.	N/A
Non-Fatal (8)	The job does not use this return code.	N/A	N/A
Failed (12)	Failed in parameter validation for transaction code.	Provide a valid transaction code and schedule a new job.	NA
Terminated (16)	The job was terminated manually by the user.	The reason for the termination needs to be addressed.	If another instance of the chain has already been scheduled and run successfully, then this job should not be restarted.
System Failure (20)	The job was terminated because of database server or network issues.	The reason for the System Failure needs to be investigated.	If another instance of the chain has already been scheduled and run successfully, then this job should not be restarted.

2.1.59 Vendor Payment Check Printing Process Chain

Chain or Job Name	Vendor Payment Check Printing Process
Recommended Frequency	Nightly
Batch Schedule Name	FIN Nightly Batch Cycle
Can be restarted?	Refer to the Restart Information section
Reports generated	The following reports are generated: • Summary Report • Detail Report

Overview

The Vendor Payment Check Printing process is an alternative to the generated fixed-length files of checks to be printed by an external print center. These files represent disbursements to local and foreign addresses for vendor checks. The process does not include BIRT form designs and assumes the print center will have the forms.

The Vendor Payment Check Printing process is composed of three job steps that perform the following:

1. Selection Process
    - Validates parameters
    - Initializes temporary tables
    - Selects disbursement records to be printed and updates 2 temporary tables
    - Selects intercepted debts and updates a temporary table
    - Selects free-form remittance advice uploaded and updates a temporary table
2. File Creation
    - Validates parameters
    - Selects records from the 4 temporary tables and creates print files according to format type.
3. Report Generation
    - Validates parameters
    - Creates the Summary Report
    - Creates the Detail Report

The process selects Automated Disbursement (AD) transactions with the Print Status of *Ready for Original Printing* or *Ready for Reprint*. The selected transactions are written to a group of temporary tables. The temporary table data is then used to create print files. The print files specify how the data will be printed on each line of the remittance advice.

The maximum characters per line along with the maximum stub lines per remittance advice for each of the two disbursement formats impacts the output that the print centers will use to generate the checks. There are two Disbursement Formats printed for vendor checks within this process: *STD* (Standard Format Remittance Advice (RA) and *FRE* (Free Format Remittance Advice (RA)). Define these on Disbursement Format (DISF). The print centers should use generated check files by Advantage to print the checks based on each of the disbursement formats. The print center should also use the control reports to ensure that all checks print from the files received.

The differences between the Free Format Check Layouts and the Standard Format Check Layouts are that the Free Format's stub content can vary across remittance advices and will be obtained from a separate database table (R_FREE_FRMT_RA). The R_FREE_FRMT_RA table must be loaded through an interface as there is no online page for it. The Standard Format's stub content will always contain the following columns: Invoice Number, Invoice Date, Reference Transaction, and Amount derived from the AD transactions.

Departments that choose to submit their payment requests via interfaces by using the Disbursement Format of *FRE* should set the Single Check indication to *true* on the vendor component of interfaced payment requests. These payment requests will not be consolidated with other applicable payment requests upon disbursement. This allows the process to main the stub information updated to the R_FREE_FRMT_RA table.

Intercepted debts associated with an Entity with the Intercept Remittance Advice Format of *CASE* ((Case-based Remittance Advice), e.g. an Entity for garnishments or levies), will be limited to one remittance advice page regardless of the number of debts that could be intercepted.

Intercepted debts associated with Entity with the Intercept Remittance Advice Format of *ASMT* ((Assessment-based Remittance Advice), for example, an Entity for a Treasury Accounts Receivable, Motor-Fuel or Tobacco Tax)), will be limited to one remittance advice page regardless of the number of debts that could be intercepted.

The chain process generates print files only for warrants (AD transactions). Manual warrants issued on demand during the day, (Manual Disbursement (MD) transaction), should be printed by using the Disbursement Printing job. The Transaction Version Number (DOC_VERS_NO) for approved AD transactions used in this printing process will always be 1. Consolidation in this process uses the same Consolidation Objects Configuration (COCNFG) used for internal check printing. Fields assigned to the Consolidation Objects used in this process cannot exceed four characters. While some fields can have values up to eight characters in length, the Consolidation Object fields on the Vendor Checks Summary (R_VEND_CHK_SMRY) table only allows up to four characters.

For more details on the tables and output files, refer to the “Advanced Batch Processing” section in the *CGI Advantage – Accounts Payable User Guide*.

Process Steps	Messages
1. Selection Process	The following messages will be displayed: • A list of parameters will be displayed (followed by each associated value) • STUB_LINES_FRST_PG; value: <value> • DISB_TYP_PARM; value: <value> • APPL_RSRC_ID; value: <value> • EXTEND_RA_PG; value: <value> • ASMT_FRMT_ENTITY_CD; value: <value> • FILE_LOC; value: <value>

Process Steps	Messages
	• CASE_FRMT_ENTITY_CD; value: <value> • COMMIT_SIZE; value: <value> • SELECT_BLOCK; value: <value> • PROG_CTR_SZ; value: <value> • Using Disbursement Type of <value> • From and To Run IDs found. • Clearing table R_VEND_CHK_SMRY. <number> records deleted • Clearing table R_VEND_CHK_DET_STD. <number> records deleted • Clearing table R_VEND_CHK_DET_FRE. <number> records deleted • Clearing table R_VEND_CHK_OFST. <number> records deleted • Clearing table R_ TEMP_VEND_CHK. <number> records deleted • <number> records inserted to Summary table • Processing Detail STD format • <number> Standard Format Detail record(s) processed • Processing Detail FRE format • <number> Free Format Detail record(s) processed • Check Sequences No Added <number> records in the Summary TEMP table. • Check Sequences No Set. <number> records in the Summary table updated. • Process Complete
2. File Creation	The following messages will be displayed: • A list of parameters will be displayed (followed by each associated value) • STUB_LINES_ADD_PG; value: <value> • STUB_LINES_FRST_PG; value: <value> • PRINT_FILE_LOC; value: <value> • COMMIT_SIZE; value: <value> • SELECT_BLOCK; value: <value> • PROG_CTR_SZ; value: <value> • Batch Parameters validated Parameters are valid or invalid depending on the validation. If the parameter is invalid, then the message for invalid parameters will be displayed in the log. It will be followed by the message “Batch parameter validation failed.” • Summary Data fetch completed

Process Steps	Messages
	• Print file formatting started • <number> records from summary added to files • A list of files created will be displayed. Possible options are: • STD_ZC_ YYYYMMDD_HHMM • STD_RS_YYYYMMDD_HHMM • STD_RM_YYYYMMDD_HHMM • STD_ER_YYYYMMDD_HHMM • FRE_ZC_ YYYYMMDD_HHMM • FRE _RS_YYYYMMDD_HHMM • FRE _RM_YYYYMMDD_HHMM • FRE_ER_YYYYMMDD_HHMM • Total Number of each type of check will be displayed followed by a count. Possible options are: • Total Number of STD Zero checks in file: <count> • Total Number of STD Single Page checks in file: <count> • Total Number of STD Multiple Page checks in file: <count> • Total Number of STD Extended Remittance checks in file: <count> • Total Number of FRE Zero checks in file: <count> • Total Number of FRE Single Page checks in file: <count> • Total Number of FRE Multiple Page checks in file: <count> • Total Number of FRE Extended Remittance checks in file: <count> • Total Check Amount of each type of check will be displayed followed by the sum of the check amounts. Possible options are: • Total Check Amount in STD Zero file: $<sum> • Total Check Amount in STD Single Page file: $<sum> • Total Check Amount in STD Multiple Page file: $<sum> • Total Check Amount in STD Extended Remittance file: $<sum> • Total Check Amount in FRE Zero file: $<sum> • Total Check Amount in FRE Single Page file: $<sum> • Total Check Amount in FRE Multiple Page file: $<sum> • Total Check Amount in FRE Extended Remittance file: $<sum> • Updated AD_DOC_HDR print status.
3. Report Generation	The following messages will be displayed: • A list of parameters will be displayed (followed by each associated value) • CLIENT_NM; value: <value>

Process Steps	Messages
	• PRINT_FILE_LOC; value: <value> • COMMIT_SIZE; value: <value> • SELECT_BLOCK; value: <value> • PROG_CTR_SZ; value: <value> • Reports output folder mapped • HTML report file path: <path> • PDF report file path: <path> • Summary Report Data fetched • <number> record(s) added to Summary Report • Rendering report started. • Rendering report completed. • Summary Report Generated. • Reports output folder mapped • HTML report file path: <path> • PDF report file path: <path> • Detail Report data fetched • <number> record(s) added to Detail Report • Rendering report started. • Rendering report completed. • Detail Report Generated.

Restart Information

If the first or second steps within the chain job fail due to any reason, schedule a new job from the beginning after correcting the errors that caused the job to fail.

If the third step in the chain job fails, inactivate the first and second steps in the chain job when scheduling a new job after correcting the errors that caused the third step to fail.

Major Input

Tables

The following is a list of tables used as input:

- AD Transaction Header (AD_DOC_HDR)
- AD Transaction Vendor (AD_DOC_VEND)
- AD Transaction Accounting (AD_DOC_ACTG)
- Unit (UNIT / R_UNIT)
- Bank (BANK / R_AP_BANK_ACCT)
- Country (CTRY / R_CTRY)
- Free Format Remittance Advice (R_FREE_FRMT_RA)

- Intercept External Allocation (INTEA / R_INT_EA)
- Intercept Activity (INTA / AP_INCT_ACTV)

Batch Parameters

Selection Process

Parameter	Description	Default Value
Entities for ASMT Format (ASMT_FRMT_ENTITY_CD)	One or more required Entities where the Remittance Advice for respective debts will be printed using the Assessment-based Format. Separate multiple values from Entity (ENTY) with commas.	No Default
Contact Message for ASMT Format (ASMT_FRMT_CNTAC_MSG)	Required message for contact information used on Assessment-based Remittance Advice. It must be left justified and not to exceed 255 characters. It prints in the bottom box of intercepted debts, split into 4 lines with a maximum of 75 characters each.	No Default
Entities for CASE Format (CASE_FRMT_ENTITY_CD)	One or more required Entities where the Remittance Advice for respective debts will be printed using the Case-based Format. Separate multiple values from Entity (ENTY) with commas	No Default
Contact Message for CASE Format (CASE_FRMT_CNTAC_MSG)	Required message for contact information used on Case-based Remittance Advice. Required. It must be left justified and not to exceed 255 characters. It prints in the bottom box of intercepted debts, split into 4 lines with a maximum of 75 characters each.	No Default
Commit Block Size (COMMIT_SIZE)	Required parameter for performance tuning on the number of records written to temporary tables in a single update. If left blank, 500 defaults.	500
Disbursement Type	Optional selection parameter. If left blank, then all records are	No Default

Parameter	Description	Default Value
(DISB_TYP_PARM)	selected. Valid Values are: 1 – Check, 2 – Warrant, 3 – Remittance Advice, and 4 – EFT.	
Extended RA Page Minimum (EXTEND_RA_PG)	Required number of stub pages necessary to be an Extended Remittance Advice.	7
From AD Chain Run ID (FRM_RUN_ID)	Optional selection parameter of an AD Chain instance (Job ID) to identify a specific group of disbursements. If not entered, all eligible records are selected. This parameter is required if the To AD Chain Run ID parameter is used. Both can be the same ID to select from just a single run.	No Default
Progression Counter (PROG_CTR_SZ)	Required parameter to control the issuance of progression messages into the job log. If left blank, 5000 defaults.	5000
Select Block Size (SELECT_BLOCK)	Required parameter for performance tuning that controls the number of records selected from the AD catalogs in a single instance to write to the temporary tables. If left blank, 1000 defaults.	1000
Number of Overflow Stub Lines per Page (STUB_LINES_ADD_PG)	Required number of stub lines on overflow pages.	66
Number of 1st Page Stub Lines (STUB_LINES_FRST_PG)	Required number of stub lines for the first check page.	37
To AD Chain Run ID (TO_RUN_ID)	Optional selection parameter of an AD Chain instance (Job ID) to identify a specific group of disbursements. If not entered, all eligible records are selected. This parameter is required if the From AD Chain Run ID parameter is used. Both can be the same ID to select from just a single run.	No Default

File Creation

Parameter	Description	Default Value
Total Offset Message for ASMT Format (ASMT_RA_TOT_OFST_MSG)	Required message associated with the Total Offset Amount printed using the Assessment-based Format on the Remittance Advice. It must be left justified and not to exceed 70 characters. It prints at the bottom of the intercepted debts section. Must match the parameter in the selection job.	None
Total Offset Message for CASE Format (CASE_RA_TOT_OFST_MSG)	Required message associated with the Total Offset Amount printed using the Case-based Format on the Remittance Advice. It must be left justified and not to exceed 70 characters. It prints at the bottom of the intercepted debts section. Must match the parameter in the selection job.	None
Commit Block Size (COMMIT_SIZE)	Required parameter for performance tuning on the number of records written to the output files in a single update. If left blank, 500 defaults.	500
Print File Output Location (PRINT_FILE_LOC)	Required file location to write the print files.	$$AMSROOT$$/ExportImport
Progression Counter (PROG_CTR_SZ)	Required parameter to control the issuance of progression messages into the job log. If left blank, 5000 defaults.	5000
Select Block Size (SELECT_BLOCK)	Required parameter for performance tuning that controls the number of records selected from the temporary tables in a single instance to write to an output file. If left blank, 1000 defaults.	1000

Parameter	Description	Default Value
Number of Overflow Stub Lines per Page (STUB_LINES_ADD_PG)	Required number of stub lines on overflow pages.	66
Number of 1st Page Stub Lines (STUB_LINES_FRST_PG)	Required number of stub lines for the first check page.	37

Report Generation

Parameter	Description	Default Value
Client Name (CLIENT_NM)	Optional name to appear as the first line in the header of reports.	None
Commit Block Size (COMMIT_SIZE)	Required parameter for performance tuning on the number of records written to a report in a single update. If left blank, 500 defaults.	500
Select Block Size (SELECT_BLOCK)	Required parameter for performance tuning that controls the number of records selected from the temporary tables in a single instance before writing to a report. If left blank, 1000 defaults.	1000
Progression Counter (PROG_CTR_SZ)	Required parameter to control the issuance of progression messages into the job log. If left blank, 5000 defaults.	5000

Major Output

Selection Process

- Vendor Checks Summary (R_VEND_CHK_SMRY)
- Vendor Checks Detailed for Standard Remittance Advice (R_VEND_CHK_DET_STD)
- Vendor Checks Detailed for Free Format Remittance Advice (R_VEND_CHK_DET_FRE)
- Intercepted Vendor Checks for Standard and Free Format Remittance Advice (R_VEND_CHK_OFST).

File Creation

- Free Format, $0 - FRE_ZC_YYYYMMDD_HHMM
- Free Format, Single Page - FRE_RS_YYYYMMDD_HHMM
- Free Format, Multi-Page - FRE_RM_YYYYMMDD_HHMM
- Free Format, Extended Remittances - FRE_ER_YYYYMMDD_HHMM

- Standard, $0 - STD_ZC_YYYYMMDD_HHMM
- Standard, Single Page - STD_RS_YYYYMMDD_HHMM
- Standard, Multi-Page - STD_RM_YYYYMMDD_HHMM
- Standard, Extended Remittances - STD_ER_YYYYMMDD_HHMM

Report Generation

- Summary Report
- Detail Report

Job Return Codes

The following table shows the potential return codes for the Vendor Payment Check Printing Process. Note that the Chain job will end with the highest return code across all of the jobs.

Return Code	Condition	Recommendation
Successful (1)	All of the jobs end successfully. A job completes successfully when all of the selected records are processed successfully.	N/A
Warning (4)	One of the jobs in the chain ends with a return code of Warning. Jobs finish with warnings when no eligible record is found for specified parameters.	Verify that AD transaction records meet the criteria specified in the parameters and are marked for printing or re-printing.
Non Fatal Error (8)	None of the jobs return a Non Fatal Error in any condition.	N/A
Failed (12)	One of the jobs in the chain ends with a return code of Failed. Jobs will fail under the following conditions: • Parameters are invalid. • Run time exceptions for unexpected situations. • The PDF and/or HTML file paths are specified incorrectly.	The reason for the job failure needs to be investigated before scheduling a new job.
Terminated (16)	One of the jobs in the chain ends with a Return Code of Terminated. This return code will be issued when the job is terminated by the user.	The reason for the termination needs to be investigated before scheduling a new job.
System Failure (20)	One of the jobs in the chain ends with a Return Code of System Failure. This return code will be issued when the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated before scheduling a new job.

Selection Criteria

Selection Process

- The Selection process selects all AD transactions within the specified From and To AD Chain Run IDs, if specified. If not specified, then all AD Transactions are selected. In either case, only those with the Printing Status Indicator (PRN_STA_IND) on the AD Transaction Header table is set to *2 - Ready for Original Printing* or *3 - Ready for Re-Printing*.
- Inserts a record to the Vendor Check Summary (R_VEND_CHK_SMRY) table for each check selected on the AD Transaction Header table.
- For selected checks with the *STD* Disbursement Format, inserts the stub lines for selected checks to Vendor Checks Detailed (R_VEND_CHK_DET_STD).
- Establishes records on Free Format Remittance Advice (R_FREE_FRMT_RA) using the Reference Transaction Code, Department, Unit, and Transaction ID from the first accounting line of the payment transaction. Adds matching records if they exist to Free Format Remittance Advice (R_VEND_CHK_DET_FRE), in order based on the Stub Line Number.
- For selected checks with the *FRE* Disbursement Format, inserts the stub lines for selected checks to Vendor Checks Detailed (R_VEND_CHK_DET_FRE).
- For selected checks where the Total Intercepted Amount for an AD Transaction Header record is greater than $0, inserts the stub lines to Vendor Offset (R_VEND_CHK_OFST).
- For each selected check, insert a record to R_TEMP_VEND_CHK, which helps the process increase performance.

File Creation

The second step of the Vendor Payment Check Printing Process selects all records from the temporary tables in the first step of the chain job to create the print files. Print files for the Standard Format checks comes from the Summary table and the Standard Format Detailed table. Likewise, print files for the Free Format checks comes from the Summary table and the Free Format Detailed table. To see the specifications of the output files, refer to the “Advanced Batch Processing” section in the *CGI Advantage – Accounts Payable User Guide*.

The following are technical details on file creation.

- Selects records from the temporary tables and uses records from the Summary and Detailed tables to create the Standard and Free Format files.
- Uses File Type and Check Sequence Number to fill the check specification files.
- Sorts by Check Sequence Number.
- Generates page 1 with lines as they are in the temporary tables.
- Determines whether intercepts are present and whether they come next or if overflow pages are needed.
- If overflow pages are needed, executes logic to generate overflow pages, moving the lines as they are specified in the temporary tables.
- Executes logic to generate intercepts, moving the lines as they are specified in the temporary table.

- Breaks the check files into eight, individual, fixed-length files by file type and disbursement format to be sent to the print center. If no data exists for a specific Disbursement Format, there will be no file created for that Disbursement Format during that run of the Java process.
- All fields within the check print files are left-justified.
- Updates the Print Status on the AD transaction to Printed or Reprinted, as applicable.

Report Generation

Selects all records from all of the tables in the first step of the chain job to create the Summary Control report. Detail records for the Free Format Detail reports come from the Detail table - Free Format in addition to the Summary table. Likewise, detail records for the Standard Format Detail reports come from the Detail table.

Sort Criteria and Check Sequencing

Selection Process

Process the free format checks first.

a. For the "ZC" File Type within the Free Format Disbursement Format, no additional sorting needs to be done. Assigns the check number beginning with Check Sequence Number “0000001” for the first sorted check then increments it by “1” for each subsequent check on the temporary tables.

b. Process the remaining File Types in descending order.

   i. For each Disbursement Format/File Type grouping, when the Handling Code is populated, sorts the records by Consolidated Object 1 (CNSD_1_OBJ) followed by Handling Code. Assigns the check number beginning where the previous grouping left off.

   ii. For each Disbursement Format/File Type grouping, when the Handling Code is not populated, puts the foreign records at the top and sorts the remaining records by zip code. Assigns check numbers beginning where the previous grouping left off.

Process the standard format checks next.

a. For the "ZC" File Type within the Standard Format Disbursement Format, no additional sorting needs to be done. Assign check numbers beginning where the previous grouping left off.

b. Sort remaining records by Bank Code.

c. Process the remaining File Types in descending order.

   i. For each Disbursement Format/File Type grouping, when the Handling Code is populated, sort the records by Consolidated Object 1 (CNSD_1_OBJ) followed by Handling Code. Assign check numbers beginning where the previous grouping left off.

   ii. For each Disbursement Format/File Type grouping, when the Handling Code is not populated, put the foreign records at the top and sort the remaining records by zip code. Assign check numbers beginning where the previous grouping left off.

File Creation

Sort by Check Sequence Number.

Problem Resolution

It is always a good practice to look at the log of each job for warnings and errors even if the job ran successfully.

If the first job in the chain fails, correct any errors and reschedule the chain job. In the case that the second job in the chain fails, the chain can be rescheduled with the first job set as disabled. This will make sure the data is not purged from the temporary tables. If the report job fails, the errors can be corrected and it can be scheduled by itself.

2.2 Check Writer Batch and Chain Processes

Descriptions of the Check Writer Processes are organized in this section in alphabetical order.

2.2.1 Automatic CW EFT Reversal Confirmation

Job Name	Automatic CW EFT Reversal Confirmation
Recommended Frequency	On demand. Alternatively, the job can be run before the CW EFT Reversal Chain, or the job can be configured to be part of the CW EFT Reversal Chain between the Build Flat File job and the Initiate CW Payment Cancellation job.
Single Instance Required	Yes
Can be restarted?	No
Exception reports generated	Yes

Overview

The Automatic CW EFT Reversal Confirmation process automatically confirms the Check Writer EFT reversals sent to the bank that are eligible to be confirmed automatically based on the lag days.

The steps involved in this process are:

1. **Parameter Validation**: First the process validates the batch parameters.
2. **Selection of Records**: Once the parameter validation is successful, the job calculates the cutoff date and selects records from the Check Writer EFT Reversal (CWEFTREV) table. The job calculates the cutoff date by subtracting the number of business days specified by the Cancel EFT Reversal Lag Days (CAN_EFT_REV_LAG_DAYS) parameter on the Application Parameters (APPCTRL) table from the Application Date. The CWEFTREV records whose Reversal Status is equal to *Reversal sent to the Bank* and Date ACH Reversal File Created is before the calculated cutoff date are selected.
3. **Process Records**: Update the CWEFTREV record Reversal Status to *Reversal Confirmed*. The updated records are also written to the Automatic CW EFT Reversal Confirmation Report.

Process Steps	Messages
1. Parameter Validation	• Validating Batch Parameters • Parameters are valid or invalid depending on the validation. If the parameter is invalid, the invalid value is displayed in the log. • Parameter validation completed
2. Selection of Records	• After calculating the cutoff date, the following message is issued: • CW EFT Reversal records with Reversal sent to the Bank status and Date ACH Reversal File Created before <<date>> will be processed. • If the selection returns 0 records, then the following message is

Process Steps	Messages
	issued: • No records meet selection criteria.
3. Process Records	• Number of records confirmed: 'n'

Restartability

The job cannot be restarted. If the job fails in any of the above steps, a new job should be scheduled after correcting the errors that caused the job to fail.

Major Input

- Check Writer EFT Reversal (CWEFTREV)
- Application Parameters (APPCTRL)

Job Parameters

Name	Description	Value and Edits
CLIENT_NM	Optional. Client Name	No default

Major Output

- Check Writer EFT Reversal (CWEFTREV) table
- Automatic CW EFT Reversal Confirmation Report

Job Return Code

The following table shows the potential job Return Codes for the Automatic CW EFT Reversal Confirmation process:

Return Code	Condition
Successful (1)	All selected records are processed successfully.
Warning (4)	No eligible records found. This could be because of the following reason: • No records meet selection criteria.
Non-Fatal Error (8)	N/A
Failed (12)	The job will fail under the following conditions: • Parameters are invalid • Run time exceptions for unexpected situations.

Terminated (16)	This Return Code is issued when the job is terminated by the user.
System Failure (20)	This Return Code is issued when the job is terminated because of database server or network issues.

Sort Criteria

The selected records are sorted by Bank Account Code, EFT Number, CW Department, and CW File ID.

Selection Criteria

The process selects all CWEFTREV table records that meet the following selection criteria:

- Reversal Status is equal to *Reversal sent to the Bank*, and
- Date ACH Reversal File Created is before the calculated cutoff date.

Problem Resolution

The process cannot be restarted on failure; however, a new job should be scheduled after resolving the issue.

The following table shows the possible Return Codes and recommendations for each processing steps:

Step 1: Parameter Validation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful **Sample Message:** "Parameter validation is completed"	N/A	N/A
Warning (4)	N/A	This step does not issue this return code.	This step does not issue this return code.
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	The job failed because of runtime exceptions or the job has invalid parameters. **Sample Message:** "The Cancel EFT	The reason for the failure needs to be investigated before re-scheduling the job.	Reschedule the job with appropriate parameter values.

Possible Return Codes	Condition	Recommendation	Other Instructions
	Reversal Lag Days on the APPCTRL table must be an integer greater than zero"		
Terminated (16)	The job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can be rescheduled.	Schedule a new job.
System Failure (20)	The job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can be rescheduled.	Schedule a new job.

Step 2: Selection of Records

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	If the selection returns 0 records, then the following message is issued: **Sample Message:** "No records meet selection criteria."	Verify CWEFTREV records against selection criteria and set up records accordingly before re-scheduling the job.	Reschedule job with appropriate records on CWEFTREV.
Non-Fatal Error (8)	N/A	N/A	N/A
Failed(12)	The job failed because of runtime exceptions for an unexpected situation.	The reason for the failure needs to be investigated before re-scheduling the job.	Reschedule the job with appropriate parameter values.
Terminated (16)	The job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can be rescheduled.	Schedule a new job.
System Failure (20)	The job is terminated because of database	The reason for the System Failure needs to	Schedule a new job.

Possible Return Codes	Condition	Recommendation	Other Instructions
	server or network issues.	be investigated. The job can be rescheduled.	

Step 3: Processing of Records

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All the records were processed successfully.	N/A	N/A
Warning (4)	N/A	This step does not issue this return code.	This step does not issue this return code.
Non-Fatal Error (8)	N/A	N/A	N/A
Failed(12)	The job failed because of runtime exceptions for an unexpected situation.	The reason for the failure needs to be investigated before re-scheduling the job.	Reschedule job with appropriate parameter values.
Terminated (16)	The job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can be rescheduled.	Schedule a new job.
System Failure (20)	The job is terminated because of database server or network issues	The reason for the System Failure needs to be investigated. The job can either be restarted or a new job can be scheduled.	Schedule a new job.

2.2.2 CW 1099 Posting Process

Job Name	CW 1099 Posting Process
Recommended Frequency	Daily as part of the nightly cycle or On Demand. Can be run On Demand for a small set of records.
Single Instance Required	Yes.
Can be restarted?	Yes, see the individual jobs for more details.
Exception reports generated	Yes.

Overview

The CW 1099 Posting process builds 1099 reportable posting lines for each payment within a selected CW File ID where the CW Accounting transaction generated 1099 reportable posting lines. The amount on the CW Accounting transaction 1099 reportable posting line is distributed proportionally to the posting lines generated for payment. Each of the generated posting lines are posted to the Posting Line Catalog, where they can be used as input to the Journal Posting routine to create 1099 journal entries.

The CW 1099 Posting Chain posts the accounting impacts for each CW payment via a 2-step job:

1. 1099 Posting Line Generation
2. 1099 Journal Posting

Major Input

- Check Writer Options (CWOPT) R_AP_CW_OPT
- Check Writer Header (CWHDR) R_AP_CW_OPT
- Check Writer Payment (CWPYMT) R_AP_CW_PYMT
- Check Writer Accounting (CWACTG) R_AP_CW_ACTG
- Check Writing Accounting Transaction (CWA)
- Posting Code (PSCD) R_RSCD
- Posting Line Catalog PSTNG_LN_CAT

Major Output

- Check Writer 1099 Posting Report
- 1099 Reporting Journal (J1099)
- 1099 Reporting Information (1099I)
- Posting Line Catalog

Chain Return Code

The following table indicates the potential return codes with which the chain will end:

Return Code	Condition
Successful (1)	All of the jobs end successfully.
Warning (4)	One of the jobs in the chain ends with a return code of "Warning."
Non-Fatal Error (8)	One of the jobs in the chain ends with a return code of "Non-Fatal Error."
Failed (12)	One of the jobs in the chain ends with a return code of "Failed."
Terminated (16)	One of the jobs in the chain ends with a return code of "Terminated."
System Failure (20)	One of the jobs in the chain ends with a return code of "System Failure."

Problem Resolution

Please refer to the individual job "Problem Resolution" section for more details.

CW 1099 Posting Process Chain: 1099 Posting Line Generation

Job Name	1099 Posting Line Generation
Recommended Frequency	Daily as part of the nightly cycle or On Demand. Can be run On Demand for a small set of records.
Single Instance Required	Yes.
Can be restarted?	Yes.
Exception reports generated	Yes.

Overview

The purpose of the 1099 Posting Line Generation step is to build 1099 reportable posting lines for each payment within a selected CW File ID where the CW Accounting transaction generated 1099 reportable posting lines. These posting lines are posted to the Posting Line Catalog, where they can be used as input to the 1099 Journal Posting step.

For each selected CW Header record (Department/CW File ID), the 1099 Posting Line Generation batch job reads the Posting Line Catalog for posting lines for the CW Accounting transaction referenced on the posting line and builds 1099 reportable posting lines for each of the associated CW payments (with Cancellation Status as *N/A*), where applicable.

The posting lines of the CW Accounting transaction containing posting codes that have the Journal Type for 1099 Reporting set to 'Yes' are used as the source to create 1099 reportable posting lines for each of the payments within the CW File ID. In addition, when the 1099 Reportable Funding Validation flag on the CW Options table is set to 'Yes', only selected posting lines with Miscellaneous 3 field with a value of '1099' are considered for further CW 1099 processing. The Miscellaneous 3 field on a posting line is set to '1099' when the CW Accounting transaction is submitted to final and the posting line has a COA element (Object, Sub Object, BSA or Sub BSA) that is 1099 reportable.

The 1099 Posting Line Generation updates the following tables:

- Posting Line Catalog.
- CW Payment
- CW Header
- 1099 Reporting Information

Note: Updates to the Posting Line Catalog, 1099I and CWPYMT for a selected CW Payment line are subject to the commit block interval specified in the job parameters. If an update to the Posting Line Catalog, 1099I table or CW Payment record fails for a specific CW Payment, updates for all payments within the commit block will be backed out, not just the selected CW Payment that failed. The selected CW payments within the commit block will be bypassed from processing. The 1099 Posting Line Generation process will continue processing with the next available CW Payment line within the selected Dept/CW File ID, if any. Records bypassed may be reprocessed in a subsequent execution of the CW 1099 Posting Chain.

Restartability Information

If the job fails during one of the above steps, all changes made after the last commit are backed out. Once the problem is fixed, the job can be restarted and will skip the records that were previously processed. If the job cannot be restarted, then a new chain can be scheduled for the same CW File ID. The new chain only picks up the remaining records and does not select the previously processed records.

Major Input

- CW Options (CWOPT) table - This provides user-defined "Posting Code for 1099 Posting" and "1099 Reportable Funding Validation" options.
- CW Header (CWHDR) table - This provides general information regarding the associated payments within the CW file record.
- CW Accounting (CWACTG) table - This provides accounting-level information within the CW file record.
- CW Payment (CWPYMT) table - This provides payment-level information within the CW file record, such as payment amount, Check/EFT number, and vendor information.
- Posting Line Catalog – The Catalog provides posting lines from which journal records are generated.
- Posting Code (PSCD) table - It provides information about which posting codes need to post to the 1099 Journal.

Batch Parameters

Parameter	Description	Default Value
File location of parameter file (AMSPARM)	Optional. Location of the parameter file to be used in subsequent job steps within the chain. (** **Refer to Note:** Assumptions for SWBP on page no. 7)	$$AMSPARM$$
CW File ID(s) to be processed. (CW_FILE_ID)	Optional. Parameter for the user to specify the CW File ID's to be selected for processing. If multiple File IDs are to be processed, the file ID's should be comma delimited.	<blank>
Department of CW File ID(s) (CW_DEPT_CD)	Optional. This field refers to the CW Department Code in the CWHDR table. This is mainly used for selecting the record in CWHDR.	<blank>
Unit of CW File ID(s) (CW_UNIT_CD)	Optional. This field refers to the CW Unit Code in the CWHDR table. This is mainly used for selecting the record in CWHDR.	<blank>
Transaction Code of generated posting lines. (DOC_CD)	Required. Parameter for the user to specify the Transaction Code to use on the generated posting lines.	<blank>
Transaction Prefix for Generated posting lines (DOC_PFX)	Optional. Parameter for the user to specify the prefix used to generate the Transaction ID of the generated posting lines.	<blank>
Commit Block Size. (COMMIT_SIZE)	Optional. Number of payment lines grouped together for committing updates.	10
Client Name (CLIENT_NM)	Optional. Client Name for Report.	<blank>
Filename of parameter text file (PARM_FILE)	Required. Filename of journal posting parameter file.	CW1099Parm.txt
Progression Message Block Size (PROG_CTR_SZ)	Required. This field provides the information about the record being processed.	No Default

Major Output

- A parameter file is generated which contains a list of the generated posting lines with the Transaction Code and the starting and ending Transaction IDs of the generated posting lines. This file becomes the input to the 1099 Journal Posting step.

- Check Writer 1099 Posting Report
- 1099 Reporting Information (1099I)
- R_AP_CW_HDR (update PROC_1099_FL)
- R_AP_CW_PYMT (update PROC_1099_FL)

Job Return Code

The following table shows the potential job return codes for the CW 1099 Posting Line Generation Process job.

Return Code	Condition
Successful (1)	All of the validations are performed successfully and no errors as reported.
Warning (4)	No records on CW Header meet the selection criteria.
Non-Fatal Error (8)	N/A
Failed (12)	The job fails under the following conditions: • Parameters are invalid. • No records that meet existing criteria were found • No record could be found for a specified File ID. • Must be a valid Posting code for 1099 Posting on CW options table for the given FY. • The valid posting code must post ONLY to the 1099 Journal. • Run time exceptions for unexpected situations. When this job ends with a Return Code of Failed, subsequent jobs in the chain are set to Inactive.
Terminated (16)	This return code is issued when the job is terminated by the user. When this job ends with a Return Code of Terminated, subsequent jobs in the chain are set to Inactive.
System Failure (20)	This return code is issued when the job is terminated because of database server or network issues.

Selection Criteria

Records are selected for processing based on the following criteria:

1. The process selects the CW Header record for each CW File ID specified on the job parameter using the following selection criteria:
   - Department Code from CWHDR matches Department Code from Job Parameter (if entered), if Department Code is not given, then it selects all CWHDR records.

- CW File ID from CWHDR matches Department Code from Job Parameter (if entered), if Department Code is not given, then it selects all CWHDR records.
- Unit Code from CWHDR matches Unit Code from Job Parameter (if entered), if Unit Code is blank, then it selects all CWHDR records.
- CW Run Status from CWHDR is 'Pending 1099 Posting.'

2. To select all relevant posting lines for the above selected CW Header record:
   - If the 1099 Reportable Funding Validation flag on the CW Options is set to ‘Yes’, select all Posting Line Catalog (PLC) entries where the following are true:

Transaction Type/Transaction Sub Type of Transaction Code (PLC) = ‘CH/CW’, AND

Transaction Department (PLC) = Department (CWHDR), AND

Transaction ID (PLC) = CW File ID (CWHDR), AND

Transaction Accounting Line (PLC) <= Number of Accounting Lines (CWHDR), AND

Posting Code (PLC) or Offset Posting Code (PLC) has Journal Type for 1099 Reporting (PSCD) = ’Yes’, AND

Journal Posting Indicator = 3 (‘Posted’), AND

Miscellaneous 3 (PLC) = ‘1099’

   - If the 1099 Reportable Funding Validation flag on the CW Options is set to ‘No’, select all Posting Line Catalog (PLC) entries where the following are true:

Transaction Type/Transaction Sub-Type of Transaction Code (PLC) = ‘CH/CW’, AND

Transaction Department (PLC) = Department (CWHDR), AND

Transaction ID (PLC) = CW File ID (CWHDR), AND

Transaction Accounting Line (PLC) <= Number of Accounting Lines (CWHDR), AND

Posting Code (PLC) or Offset Posting Code (PLC) has Journal Type for 1099 Reporting (PSCD) = ’Yes’, AND

Journal Posting Indicator = 3 (‘Posted’)

Note: If no source posting lines are selected, bypass selected CWHDR record from CW 1099 Posting processing.

3. Selection of CWPYMT:
   - Select all payments (CWPYMT) associated with the selected CW Header record where the Payment Amount is greater than zero and the “Processed for 1099” flag is ‘No’.

Sort Sequence

N/A

Problem Resolution

The following table shows the possible return codes and recommendations for each processing step.

Step 1: Parameter Validation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the table validations are successful, that is, there are no errors raised during validations.	N/A	N/A
Warning (4)	1. At least 1 record exists on CWHDR and none meet selection criteria.	Correct the data and then run the job to make it successful.	N/A
	2. All CW File IDs specified exist on CWHDR but not all are selected.	Correct the data and then run the job to make it successful.	N/A
Non-Fatal Error (8)	N/A	This step does not issue this return code.	
Failed (12)	All of the required parameters are not entered. **Sample Message:** CWP Transaction Code parameter cannot be blank. **Recommendation:** Enter the CWP Transaction Code.	If the required parameters are not entered, enter the required parameters and restart the job.	
	Entered Parameters are not valid. **Sample Message**: User ID is not valid within the application. **Recommendation:** Enter a valid User ID.	If the parameters are invalid, enter the valid parameter and restart the job.	
	Failed because of runtime exceptions for an	Failure reason needs to be investigated before	

Possible Return Codes	Condition	Recommendation	Other Instructions
	unexpected situation.	restarting the job.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can either be restarted or a new job scheduled.	
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can either be restarted or a new job scheduled.	

Step 2: Select eligible Posting Line Catalog records from CWHDR

Possible Return Codes	Condition	Recommendation	Other instructions
Successful (1)	Eligible records found.	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	
Failed (12)	Failed because of runtime exceptions for an unexpected situation.	Failure reason needs to be investigated before restarting the job.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can either be restarted or a new job scheduled	
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can either be restarted or a new job scheduled.	

Step 3: Process each selected CWPYMT record

For each selected CWPYMT record, the system shall create a corresponding Posting Line Catalog line for each posting line selected for the CWA transaction, update 1099I records if applicable, and update CWPYMT record.

Possible Return Codes	Condition	Recommendation	Other instructions
Successful (1)	The system created the corresponding Posting Line Catalog line successfully.	N/A	
Warning (4)	The 1099I update fails.	Investigate the reason for the issue and re-run the job.	
	The Posting Line Catalog “Save” fails.	Investigate the reason for the issue and re-run the job.	
Non-Fatal Error (8)	N/A	This step does not issue this return code.	
Failed (12)	Failed because of runtime exceptions for unexpected situation	Failure reason needs to be investigated before restarting the job.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can either be restarted or a new job scheduled.	
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can either be restarted or a new job scheduled.	

Step 4: Update CWHDR record

If the all of the 1099 posting lines were successfully inserted for the payments within the selected CWHDR, the system shall update the CWHDR.

Possible Return Codes	Condition	Recommendation	Other instructions
Successful (1)	The system successfully updates the CWHDR.	N/A	

Possible Return Codes	Condition	Recommendation	Other instructions
Warning (4)	N/A	N/A	
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Failed because of runtime exceptions for an unexpected situation.	Failure reason needs to be investigated before restarting the job.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can either be restarted or a new job scheduled.	
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can either be restarted or a new job scheduled.	

Step 5: Create parameter file

The system shall create the parameter file after all of the selected CWHDR records have been processed.

Possible Return Codes	Condition	Recommendation	Other instructions
Successful (1)	1 Parameter file is created.	N/A	N/A
Warning (4)	N/A	This step does not issue this return code.	
Non-Fatal Error (8)	N/A	This step does not issue this return code.	
Failed (12)	Failed because of issues in creating parameter files, for example, parameter file location not found.	Investigate the reason of failure and then re-run the job. If any location issues correct the same.	
	Failed because of runtime exceptions for an unexpected situation.	Failure reason needs to be investigated before restarting the job.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can either be restarted or a new job scheduled.	
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can either be restarted or a new job scheduled.	

CW 1099 Posting Process Chain: 1099 Journal Posting

Job Name	1099 Journal Posting
Recommended Frequency	Daily as part of the nightly cycle or On Demand. It can be run On Demand for a small set of records.
Single Instance Required	Yes.
Can be restarted?	Yes.

Exception reports generated	Yes.

Overview

The purpose of the 1099 Journal Posting step is to leverage the Advantage Journal Posting routine to post 1099 journal entries based on posting lines generated by the 1099 Posting Line Generation step.

Restartability Information

If the job fails during one of the above steps, all changes made after the last commit are backed out. Once the problem is fixed, the job can be restarted and will skip the records that were previously processed. If the job cannot be restarted, then new job can be scheduled by disabling the first job in the chain. This new job will journalize the remaining records.

Major Input

- Posting Line Catalog

Batch Parameters

Parameter	Description	Default Value
File location of parameter file (AMSPARM)	Optional. The location of the parameter file to be used in subsequent job steps within the chain. (**** Refer to Note:** Assumptions for SWBP on page no. 7)	$$AMSPARM$$
Filename of parameter text file (PARM_FILE)	Required. The Filename of the journal posting parameter file.	CW1099Parm.txt
Progression Counter Size (PROG_CTR_SZ)	Required. This field provides the information about the record being processed.	No Default

Major Output

- JRNL_1099

Job Return Code

The following table shows the potential job return codes for the CW 1099 Journal Posting Process job.

Return Code	Condition
Successful (1)	All of the validations are performed successfully and no errors are reported.
Warning (4)	N/A
Non-Fatal Error (8)	N/A
Failed (12)	The job fails under the following conditions: • Parameters are invalid. • Run time exceptions for unexpected situations.
Terminated (16)	This return code is issued when the job is terminated by the user.
System Failure (20)	This return code is issued when the job is terminated because of database server or network issues.

Selection Criteria

Records are selected based on the following criteria:

- Transaction Code from the Posting Line Catalog matches the Transaction Code from the Parameter file. AND
- Transaction ID from Posting Line Catalog >= FROM_DOC_ID and <= TO_DOC_ID of Parameter file. AND
- DOC_PHASE_CD is final or historical final phase AND
- Journal Posting Indicator of Posting Line Catalog should be ready for Journal Posting.

Sort Sequence

N/A

Problem Resolution

Step 1: Parameter Validation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the validations are successful, that is, there were no errors raised during validations.	N/A	N/A
Warning (4)	N/A	N/A	N/A

Possible Return Codes	Condition	Recommendation	Other Instructions
	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	This step does not issue this return code.	
Failed (12)	All of the required parameters are not entered.	If the required parameters are not entered, enter the required parameters and restart the job.	
	Entered Parameters are not valid.	If the parameters are invalid, enter the valid parameter and restart the job.	
	Failed because of runtime exceptions for an unexpected situation.	Failure reason needs to be investigated before restarting the job.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can either be restarted or a new job scheduled.	
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can either be restarted or a new job scheduled.	

Step 2: Update 1099 Journal:

After selecting the records that are ready to post from the Posting line catalog, Step 2 is performed.

Possible Return Codes	Condition	Recommendation	Other instructions
Successful (1)	The system successfully updates the JRNL_1099.	N/A	
Warning (4)	N/A	N/A	
Non-Fatal Error (8)	N/A	N/A	N/A

Possible Return Codes	Condition	Recommendation	Other instructions
Failed (12)	Failed because of runtime exceptions for an unexpected situation.	Failure reason needs to be investigated before restarting the job.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can either be restarted or a new job scheduled.	
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can either be restarted or a new job scheduled.	

2.2.3 CW Accounting Transaction Generation

Job Name	CW Accounting Transaction Generation
Recommended Frequency	On Demand. The CW Accounting Transaction Generation chain should be run after the CW Table Load and CW Pre-Edit chain jobs.
Single instance required	No
Can be restarted	No
Reports generated	No

Overview

The CW Accounting Transaction Generation Chain creates CW Accounting (CWA) transactions for selected CW files to post accounting lines to the applicable Advantage journals. When applicable, the process posts records to the CW Vendor Intercept (CWVINCT) table for each payment record that is intercepted. These records are used by the CW Intercept Transaction Generation process to create CW Intercept Transfer (CWI) transactions to post accounting lines to applicable journals for intercepted payments as well as updating applicable intercept tables.

Once a CW file has been loaded, validated using the CW Pre-Edit process, and certified by the department and central office, the file is eligible for selection by the CWA Generation process. If not certified by both the department and central office or the file has not passed the CW Pre-Edit process, the CW file will be bypassed by the chain job for processing. The CW Accounting Transaction Generation process selects the CW File ID records based on user parameter input to determine if the files are eligible for CWA transaction generation. The process generates a single CWA transaction for each Department/CW File ID combination selected.

The following is a list of the job steps that make up the CW Accounting Transaction Generation Chain followed by a more in-depth description of the jobs:

1. Accounting Transaction XML Generation
2. Upload
3. Submit

Major Input

- Check Writer Header (CWHDR) table
- Check Writer Payment (CWPYMT) table
- Check Writer Accounting (CWACTG) table
- Intercept Request (INTR) table

Major Output

- CWA transaction is uploaded and submitted to final

- Check Writer Vendor Intercept (CWVINCT) table is updated if payments are intercepted
- Intercept Request (INTR) table is updated if payments are intercepted
- INTA / INTP
- Check Writer tables

Chain Return Code

The following table shows the potential return codes for the CWA Generation Chain. Note that the Chain job ends with the highest return code across all of the jobs.

Return Code	Condition
Successful (1)	All of the jobs end successfully.
Warning (4)	One of the jobs in the chain ends with a return code of "Warning."
Non-Fatal Error (8)	One of the jobs in the chain ends with a return code of "Non-Fatal Error."
Failed (12)	One of the jobs in the chain ends with a return code of "Failed."
Terminated (16)	One of the jobs in the chain ends with a return code of "Terminated."
System Failure (20)	One of the jobs in the chain ends with a return code of "System Failure."

Problem Resolution

Please refer to the individual job "Problem Resolution" section for more details.

CW Accounting Transaction Generation Chain: Accounting Transaction XML Generation Job

Job Name	Accounting Transaction XML Generation
Recommended Frequency	On Demand
Single Instance Required	No
Can be restarted	No
Reports generated	No

Overview

The Accounting Transaction XML Generation job creates CWA transactions to post accounting entries for the payments. A single CWA transaction is created for each of the Department/CW File ID records meeting the selection criteria. When applicable, the process posts entry/entries to the CW Vendor Intercept (CWVINCT) table for each payment that is intercepted. On execution of

the CW Accounting Transaction Generation Chain, the process performs standard edits on the parameters entered. Each of the job step parameters are validated.

Restartability Information

This job cannot be restarted. If the job fails for some reason, then the reason for failure should be verified and the new job should be scheduled. The new job is scheduled irrespective of its previous jobs that failed.

Major Input

- Check Writer Header (CWHDR) table
- Check Writer Payment (CWPYMT) table
- Check Writer Accounting (CWACTG) table
- Intercept Request (INTR) table

Batch Parameters

Parameter	Description	Default Value
Parameter Directory AMSPARM	Parameter Directory location (**** Refer to Note:** Assumptions for SWBP on page no. 7)	$$AMSPARM$$
Export Directory AMSEXPORT	Export Directory location (**** Refer to Note:** Assumptions for SWBP on page no. 7)	$$AMSEXPORT$$
Apply Override APPLY_OV	Apply override on transaction import	<blank>
Commit block size COMMIT_BLOCK	Commit Block Size	<blank>
CW Department Code CW_DEPT_CD	CW Department Code	<blank>
CW File ID CW_FILE_ID	CW File ID(s) to be processed. Multiple files can be entered, separated by commas.	<blank>
CW Unit Code CW_UNIT_CD	CW Unit Code	<blank>
CWA Transaction Code DOC_CD	CW Accounting Transaction Code	<blank>
Intercept Selection	Intercept Selection values that will	1,2,3,4,5,6,7

Parameter	Description	Default Value
INT_SEL	be used in the selection of debt records.	
Override Level OVERRIDE_LEVEL	Override level to be used in transaction submission in the Submit job step.	<blank>
Parameter file 1 PARM_FILE_1	File name of upload parameter text file.	UploadCWAParm.txt
Parameter file 2 PARM_FILE_2	File name of submit parameter text file.	SubmitCWAParm.txt
User ID USER_ID	User ID	<blank>
Progression Message Block Size PROG_CTR_SZ	Required. This field provides the information about the record being processed.	No Default

Major Output

- XML file that is uniquely named by the CWA Transaction Code and the job's Chain ID.
- Upload parameter file (default name will be UploadCWAParm.txt) that is used as input to the second job of the CW Accounting Transaction Generation chain.
- Submit parameter file (default name will be SubmitCWAParm.txt) that is used as input to the third job of the CW Accounting Transaction Generation chain.

Job Return code

Return Code	Condition
Successful (1)	All of the validations are performed successfully.
Warning (4)	• No CW records meet selection criteria. • All of the CW File ID's specified are valid but not all were selected for processing. • When any of the files failed to process when multiple files are entered. • When the Intercept selection value is not numeric. • When the Intercept selection value is not between 0 & 7.
Non-Fatal Error (8)	All of the records failed to load into the Transaction Catalog.

Failed (12)	The job will fail under the following conditions: • Parameters are invalid • If a single CW File ID was specified during the run and it did not meet selection criteria • Technical failure When this job ends with a Return Code of Failed, subsequent jobs in the chain are set to Inactive.
Terminated (16)	This Return Code is issued when the job is terminated by the user. When this job ends with a Return Code of Terminated, subsequent jobs in the chain are set to Inactive.
System Failure (20)	This Return Code is issued when the job is terminated because of database server or network issues. When this job ends with a Return Code of System Failure, subsequent jobs in the chain are set to Inactive.

Selection Criteria

Selection Criteria for CWHDR records:

Records are selected for processing based on the following criteria:

- [(Department (CWHDR) = Department (Parameter)) or if Department (Parameter) = blank, select all CWHDR records], AND
- CW Run Status (CWHDR) = 'Pending CWA Generation', AND
- [(CW File ID (CWHDR) = CW File ID (Parameter)) or if CW File ID (Parameter) = blank, select all CWHDR records], AND
- Hold (CWHDR) = 'No', AND
- [(Unit (CWHDR) = Unit (Parameter)) or if Unit (Parameter) = blank, select all CWHDR records]

Selection Criteria for INTR records:

Intercept Request records that meet the following condition are **not** included in the payment intercept processing:

- If the INTR record contains a Claim Status not equal to Active; OR
- If {(the transaction type for the debt record = RE and the INTR Outstanding Amount - INTR Intercepted Amount is equal to $0) OR (the transaction type for the debt record <> RE and the INTR Outstanding Amount - INTR Intercepted Amount - INTR Transferred Amount is equal to $0)}; OR
- If the Intercept Selection batch parameter value(s) does not equal at least one of the Intercept Selection values that is set to true for the Entity; OR
- If a record on the INTR table with values of RE Transaction Code, RE Transaction Dept, RE Transaction Unit, RE Transaction ID, Customer Code, TIN and TIN Type matches any combination of RE Transaction Code, RE Transaction Dept, RE Transaction Unit, RE

Transaction ID, Customer Code or TIN and TIN Type on the Receivable Intercept Exception (INTREX) table.

Sort Criteria

Sorting Criteria for INTR records:

Intercept records on the Intercept Request (INTR) table are sorted by TIN, TIN Type, and Intercept Priority (in ascending order) and Due Date.

Sorting Criteria for CWPYMT records:

The process sorts Payment (CWPYMT) records associated with the CW Header record by TIN and TIN Type.

Problem Resolution

When the job fails for any reason, a new chain should be scheduled with the same parameters. Though this job makes updates to the CWVINCT table, the updates are backed out when the new chain is scheduled for the same ID; therefore, there is no need to back out any updates made to the CWVINCT table.

The following table shows the possible return codes and recommendations for each processing step.

Step 1: Batch Parameter Validation:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All batch parameter validations are successful (that is, there are no errors raised during validations.)	N/A	N/A
Warning (4)	Intercept selection value is not numeric.	Make sure that the data is correct before rescheduling the job.	N/A
Non-Fatal Error (8)	N/A	This step does not issue this return code.	N/A
Failed (12)	Required parameters are not entered. **Sample Message**: CWA Transaction Code cannot be blank.	Reschedule the job and enter the required parameters.	N/A

Possible Return Codes	Condition	Recommendation	Other Instructions
	Recommendation: Enter CWA Transaction Code.		
	Entered Parameters are not valid. **Sample Message:** CW Accounting Transaction Code does not exist on the Transaction Control table. **Recommendation:** Enter a valid CW Accounting Transaction Code.	Enter valid parameters and reschedule the job.	N/A
	Failed because of runtime exceptions caused by unexpected conditions.	If the job fails with fatal conditions on encountering unknown exceptions, then investigate the exception reported by the process, resolve the error and schedule the job.	N/A
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated before rescheduling the job.	N/A
System Failure (20)	Job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated before rescheduling the job.	N/A

Step 2: Inference and Validation Processing:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Data inferences and validation on the CW File records are successful.	N/A	N/A
Warning (4)	All of the CW File IDs specified are valid but not	Make sure that the selected records from the CWHDR	N/A

Possible Return Codes	Condition	Recommendation	Other Instructions
	all were selected.	table have valid File ID's for processing before rescheduling the job.	
	None of the selected records on CW Header meet the selection criteria.	Make sure that the selected records from the CWHDR table meet the selection criteria before rescheduling the job.	N/A
	If CW Accounting Transaction Code requires Transaction Unit, exclude CWHDR from processing if Unit is blank on CWHDR.	Make sure that the selected record from CWHDR is has a valid value before rescheduling the job.	N/A
	If the data passed the Pre-Edit job's validation step but is no longer valid or it has been changed.	Re-run the Pre-Edit chain on the CW File to get a listing of all of the errors. Once the errors are resolved and Pre-Edit runs successfully, schedule a new job.	N/A
Non-Fatal Error (8)	N/A	This step does not issue this return code.	N/A
Failed (12)	If a Single File ID was specified and it was not selected.	Make sure that the records meet the selection criteria.	N/A
	The CW Options record was not found for the FY. **Sample Message:** CW Options record not found for FY. **Recommendation:** Make sure that the CW Options table has the record for the FY associated with the Application Control table.	Make sure that the CW Option record exists for the selected Fiscal year before rescheduling the job.	N/A

Possible Return Codes	Condition	Recommendation	Other Instructions
	Failed because of runtime exceptions caused by unexpected conditions.	If the job fails with fatal conditions on encountering unknown exceptions, then investigate the exception reported by the process, resolve the error and schedule the job.	N/A
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated before rescheduling the job.	N/A
System Failure (20)	Job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated before rescheduling the job.	N/A

Step 3: CWA Transaction XML Generation Processing:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	CWA Transaction XML generated successfully.	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	This step does not issue this return code.	N/A
Failed (12)	Failed because of runtime exceptions caused by unexpected conditions.	If the job fails with fatal conditions on encountering unknown exceptions, then investigate the exception reported by the process, resolve the error and schedule the job.	N/A
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated before rescheduling the job.	N/A
System Failure (20)	Job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated before rescheduling the job.	N/A

CW Accounting Transaction Generation Chain: Upload Job

Job Name	Upload
Recommended Frequency	On Demand
Single Instance Required	No
Can be restarted	No
Reports generated	No

Overview

This job step executes a SysManUtil Import to load the CWA transaction created by the Accounting Transaction XML Generation job. An upload CWA parameter text file was created by the Accounting Transaction XML Generation job step specifying what CWA XML file to load.

On a successful upload of a CWA transaction, the transaction will perform the following:

- If the CW Run Status (CWHDR) = “Pending CWA Generation”, perform the following updates:
  - Set CWA Generated (CWHDR) to ‘Yes’.
  - Set CWA Generation Date/Time Stamp to application server date and time.
  - Set CWA Generation User ID to User ID of user executing CW Accounting Transaction Generation Chain job.
  - The Pre-condition Return code for this job is Warning.

Restartability Information

This job cannot be re-started. If the job fails then the failure reason needs to be verified and the entire CWA chain should be re-run again.

Major Input

- Upload parameter file created during the first step of this chain.

Batch Parameters

Parameter	Description	Default Value
Parameter file name PARM_FILE	File location and name of the upload parameter file.	$$AMSPARM$$/UploadCWAParm.txt

Major Output

- CWA transaction is uploaded to the Transaction Catalog in the Draft Phase.

Job Return code

The following table shows the potential job return codes for the SMU Transaction Upload job.

Return Code	Condition
Successful (1)	All of the records are loaded into the Transaction Catalog successfully.
Warning (4)	This return code is issued under the following conditions: • The input file is empty • Some records failed to load into the Transaction Catalog.
Non-Fatal Error (8)	All records failed to load into the Transaction Catalog.
Failed (12)	The job fails under the following conditions: • Parameters are invalid • The input file is not found in the specified directory • If the error messages exceed the maximum number specified in the parameter • Runtime exceptions encountered for any unexpected situations When the job ends with a Return Code of Failed, subsequent jobs in the chain are set to Inactive.
Terminated (16)	This Return Code is issued when the job is terminated by the user. When the job ends with a Return Code of Terminated, subsequent jobs in the chain are set to Inactive.
System Failure (20)	This Return Code is issued when the job is terminated because of database server or network issues. When this job ends with a Return Code of System Failure, subsequent jobs in the chain are set to Inactive.

Sort Sequence

N/A

Selection Criteria

N/A

Problem Resolution

If the job fails to load some or all of the transactions from the XML file into the Transaction Catalog, then a new instance of the chain should be executed with the same input parameters after correcting the issues. Before running the new chain, updates made to the CW Header table should be backed out.

If the job failed after loading a few transactions into the Transaction Catalog and those transactions were already committed, then before scheduling the new job those transactions should be discarded manually. The transaction can be identified by the CW File ID since the Transaction ID is equivalent to the CW File ID. Otherwise, any new chain for the CW File ID will be rejected when the job tries to load it into the Transaction Catalog.

The following table shows the possible return codes and recommendations for each processing step.

Upload the CWA transaction:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All parameters are valid and all the transactions are loaded successfully.	N/A	N/A
Warning (4)	The input file is empty. **Sample message:** Could not locate valid records in the table.	Make sure that the Accounting Transaction XML Generation job creates the input file with records.	N/A
	This return code is issued when the job fails to load some of the transactions. **Sample Message:** Cannot find transaction(s) to be restored.	The reason for the failure needs to be investigated and then load the rejected transactions.	N/A
Non-Fatal Error (8)	This return code will be issued when the job failed to load all of the transactions.	The reason for this error needs to be investigated and then load all of the rejected transactions.	N/A
Failed (12)	This return code is issued when the parameters are not valid. **Sample Message:** Job failed because no valid parameter input file was specified.	Make sure the parameter file exits in the specified folder and reschedule the job with prior jobs disabled.	N/A

Possible Return Codes	Condition	Recommendation	Other Instructions
	Failed because of runtime exceptions for an unexpected condition.	If the job fails with fatal conditions on encountering unknown exceptions, then investigate the exception reported by the process, resolve the error and schedule the job with the prior jobs disabled.	N/A
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated before rescheduling the job.	
System Failure (20)	Job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated before rescheduling the job.	

CW Accounting Transaction Generation Chain: Submit Job

Job Name	Submit
Recommended Frequency	On Demand
Single Instance Required	No
Can be restarted	No
Reports generated	No

Overview

This job step will execute a SysManUtil Submit to submit the CWA transaction created by the Accounting Transaction XML Generation job. A submit CWA parameter text file, created in the Accounting Transaction XML Generation process, specifies which transactions should be submitted.

The Pre-condition Return Code for this job is Successful.

Major Input

- SMU job parameter file
- Draft transactions in the Transaction Catalog

Restartability information:

This job cannot be re-started. The transactions can be manually submitted. The Transaction ID can be found on the processing log. If the transaction cannot be processed to final, then it should be discarded manually so that any instance of this chain with the rejected transaction's CW File ID will pick up that record for processing.

Major Output

- Transactions in Pending / Final / Rejected on the Transaction Catalog
- Check Writer Header (CWHDR) table
- Check Writer Accounting (CWACTG) table
- Check Writer Intercept (CWVINCT) table
- Pending Intercept Payment (INTP) table

Batch Parameters

Parameter	Description	Default Value
Parameter File PARM_FILE	File location and name of the submit parameter file.	$$AMSPARM$$/SubmitCWAParm.txt

Job Return Code

The following table shows the potential job return codes for the SMU Transaction Submit job.

Return Code	Condition
Successful (1)	All of the transactions submitted successfully.
Warning (4)	N/A
Non-Fatal Error (8)	N/A
Failed (12)	The job fails under the following conditions: • Parameters are invalid • Parameter file is not found • Runtime exceptions encountered for any unexpected situations
Terminated (16)	This Return Code is issued when the job is terminated by the user.
System Failure (20)	This Return Code is issued when the job is terminated because of database server or network issues.

Sort Sequence

N/A

Selection Criteria

N/A

Problem Resolution

If this job ends with a Return Code of Failed, Terminated or System Failure, then the loaded CWA transaction should be submitted manually. If the transaction cannot be processed to final, then the transaction should be discarded so that the File ID associated with the rejected transaction can be picked up by the subsequent run of this chain.

The following table shows the possible return codes and recommendations for each processing step.

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the transactions submitted successfully.	N/A	
Warning (4)	N/A	The job does not issue this return code.	
Non-Fatal Error (8)	N/A	The job does not issue this return code.	
Failed (12)	This return code is issued when the input parameter is not found in the specified directory. **Sample Message:** Parameter file could not be located/read.	Make sure that the parameter file exits in the specified folder and schedule a new job.	
	Failed because of runtime exceptions for an unexpected situation.	If the job fails with fatal conditions on encountering unknown exceptions, then investigate the exception reported by the process, resolve the error and submit the transaction manually. The Transaction ID can be found on the process logs. If the issue cannot be resolved, then a new chain should be scheduled after discarding the Draft transaction from the Transaction Catalog.	

Possible Return Codes	Condition	Recommendation	Other Instructions
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. If the issue is resolved then the loaded transaction can be submitted manually. The Transaction ID can be found on the Transaction Catalog. If the issue cannot be resolved, then a new chain should be scheduled after discarding the Draft transaction from the Transaction Catalog.	
System Failure (20)	Job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. If the issue is resolved then the loaded transaction can be submitted manually. The Transaction ID can be found on the Transaction Catalog. If the issue cannot be resolved, then a new chain should be scheduled after discarding the Draft transaction from the Transaction Catalog.	

2.2.4 CW ACH File Generation

Job Name	CW ACH File Generation
Recommended Frequency	Daily, as part of the nightly cycle, or On Demand.
Single Instance Required	Yes
Can be restarted	No
Reports Generated	Yes

Overview

The CW ACH File Generation process runs after the CW Check/EFT Generation process. The XML output acts as the source for creating the four ACH flat files in the ACH Formatting process. The four flat files are created in accordance with NACHA rules for EFT payments. The CW ACH File Generation process has the following jobs (each of the jobs listed below is described in subsequent sections):

1. Build XML File
2. Build Flat File

If the APPCTRL parameter USE_CWACH_CONFIG is set to *True*, the process reads through the ACH Configuration pages for information to create the ACH output file(s). For any missing Record Types in the ACH Configuration pages, the process uses existing code to create the ACH output file(s).

Note:

- *Even though the above jobs in the chain can be run individually by disabling other jobs, it is recommended to always run the entire chain.*
- Refer to the “ACH Configuration Pages” topic in the *Disbursement User Guide* for more details on the 3 ACH Configuration pages.

Acceptable return codes for both jobs are Successful, Warning, and Non-Fatal Error. If the first job in the chain ends with a Return Code of Failed, Terminated or System Failure, the second job will be set to Inactive.

Major Input

- Check Writer Header (CWHDR / R_AP_CW_HDR)
- Check Writer Payment (CWPYMT / R_AP_CW_PYMT)
- Check Writer Addendum (CWADNM / R_AP_CW_ADNM)
- ACH XML File Configuration (ACHXML / R_ACH_XML_CNFG)
- ACH Flat File Configuration (ACHFF / R_ACH_FILE_CNFG)

Major Output

- ACH PPD Flat File

- ACH CCD Flat File
- ACH CTX Flat File
- ACH IAT Flat File

Chain Return Code

The following table indicates the potential return codes with which the chain will end:

Return Code	Condition
Successful (1)	All of the jobs end successfully.
Warning (4)	One of the jobs in the chain ends with a return code of “Warning.”
Non-Fatal Error (8)	One of the jobs in the chain ends with a return code of *Non-Fatal Error*.
Failed (12)	One of the jobs in the chain ends with a return code of “Failed.”
Terminated (16)	One of the jobs in the chain ends with a return code of “Terminated.
System Failure (20)	One of the jobs in the chain ends with a return code of “System Failure.”

Problem Resolution

If any of the jobs in the chain encounters a Non-Fatal Error, it is advisable to review and correct the errors as appropriate, and reschedule the job as needed. When a Non-Fatal Error occurs in the Build XML File job, some records may still be processed successfully. Therefore, any generated ACH files should be allowed to follow normal processing to send to the bank. It is recommended to configure the chain job Pre Condition Return Code as follows to allow the successfully processed records to complete processing.

Job Name	Pre Condition Return Code
Build XML File	Successful
Build Flat File	Non-Fatal Error

Only the first Job updates the table, if it fails you must reschedule the chain. If any batch job fails, it is necessary to reschedule the entire chain. In order to do this, the user should take the following steps:

1. Open the log of the first job in the chain (Build XML File) and note which CW header records were processed. You will need to know the Department and the File ID of each record.
2. Navigate to the CW Header page (CWHDR).
3. Find the records that were processed and change their ACH File Generation to “Ready for Regeneration” if the status is Generated.
4. Reschedule the job, making sure to enter “3” (Ready for Regeneration) for the ACH XML Status parameter.

CW ACH File Generation Chain: Build XML File Job

Job Name	Build XML File
Recommended Frequency	Daily, as part of the nightly cycle, or On Demand.
Single Instance Required	Yes.
Can be restarted	No
Report Generated	No

Overview

The objective of the Build XML File process is to generate an ACH XML file. This file includes the information required to construct the actual ACH files with the ACH format of PPD+, CCD+, CTX and IAT. This XML file is the single source for the different ACH file formats that are generated in the next step (Build Flat File).

If the APPCTRL parameter USE_CWACH_CONFIG is set to *True*, the Build XML File job reads through the ACH XML Configuration page (ACHXML) for information to create the ACH XML file(s). For any missing record types in the ACH XML Configuration page, the process uses existing code to create the ACH XML file(s).

The following table shows the various steps that the Build XML File Job goes through and the messages issued at each step.

Process Steps	Messages
1. Parameter Validation	• Validating the batch parameters. • Parameter validation completed.

2. Record Selection	The following messages are displayed
	• Selecting eligible records
	• Records with EFT Format CCD found.
	• Creating File Header record for CCD
	• Started processing File ID <CW_File_id>
	  • Creating Batch Header record for CW File ID <CW_File_id>
	  • Creating Entry detail Credit record for CW File ID: <CW_File_id>
	  • Creating Addendum record for CW File ID: <CW_File_id>
	  • All 1 payment records processed successfully for CW File ID: <CW_File_id>
	  • Creating Entry detail Debit record for CW File ID:

Process Steps	Messages

	<CW_File_id>
	• Creating Batch Control record for CCD
	• Creating File Control record for CCD
	• CCD records processed successfully
	
	The above format of the message is written inside the log for the EFT format “CTX” and “PPD.” In case of “CTX” and “PPD”, the following is written:
	• CTX records processed successfully
	• PPD records processed successfully
	• IAT records processed successfully
	
	If not all existing records meet the selection criteria, the log will display: “Not all existing records were selected for processing.”
	• The following message is written in the log when no records are selected:
	  • No records on CW Header meet selection criteria.
	  • No record found for CCD Format, no CCD file generated
	  • No record found for CTX Format, no CTX file generated
	  • No record found for PPD Format, no PPD file generated
	  • No record found for IAT Format, no IAT file generated

3. XML Generation	If the Bank record cannot be found, the system writes the following log message: • “File ID <CWHDR Check Writer File ID> is skipped because the Bank record <CWHDR Bank> cannot be found.” If the Bank record is found but does not have ACH Company ID populated, and the Disbursement Format record corresponding to the CW Header EFT Format does not have Company Identification populated, the system writes the log message: • “File ID <CWHDR Check Writer File ID> is skipped because the Company ID cannot be determined from the bank or disbursement format.” If the system encounters one of the above mentioned errors, after writing the log message, the system skips the rest of the processing for the current Check Writer File and resumes processing at the next Check Writer File.

Process Steps	Messages
	At the end of this step, the following message is written in the log: • "ACH XML Generation file successfully generated."

Restartability

This job does not support restart ability. It does not update any table; therefore, no back-out of data is required. Please see the steps mentioned in the "Problem Resolution" section of the chain.

Major Input

- Check Writer Header (CWHDR / R_AP_CW_HDR)
- Check Writer Payment (CWPYMT / R_AP_CW_PYMT)
- Check Writer Addendum (CWADNM / R_AP_CW_ADNM)
- ACH XML File Configuration (ACHXML / R_ACH_XML_CNFG)

Batch Parameters

Parameter	Description	Default Value
ACH XML Status (ACH_XML_STATUS)	An optional selection parameter for the ACH XML Status: • 2 - Ready for Original Generation • 3 - Ready for Re-Generation If no value is entered, then both 2 and 3 are selected.	(No Default)
ACH XML File Location (AMSEXPORT)	The required file location for saving XML files.	$$AMSEXPORT$$
Parameter File Location (AMSPARM)	The required file location for the parameter file.	$$AMSPARM$$
Client Name (CLIENT_NM)	An optional name for report headers.	(No Default)
Check Writer File ID (CW_FILE_ID)	An optional selection parameter for one or more file IDs (comma separated). If left blank, all unprocessed IDs are processed.	(No Default)

Check Writer File Department (CW_DEPT_CD)	A selection parameter that is optional unless the Unit parameter is used.	(No Default)
Output File Prefix (PREFIX)	A required prefix for the XML file created that will also include the Chain Job ID.	CWACH
Check Writer File Unit (CW_UNIT_CD)	An optional selection parameter.	(No Default)
Progression Counter Size (PROG_CTR_SZ)	A required size for issuing job log messages about record processing.	(No Default)
Add Bank Account Code to Company Identification (ADD_BANK_CODE_TO_CMPNY_ID)	An optional output parameter that overwrites the trailing characters of Company Identification attribute with Bank Account code on the output ACH XML file generated. This is useful when using an external printing process.	(No Default)
Get Company Name and Entry Description from Bank (CMPNY_NM_DESC_FROM_BANK)	An optional parameter that decides the data source of Company Name and Company Entry Description for the Batch Header Record (Record Type 5) of the generated .dat file. When set to *Yes*, the process writes the first 16 characters of the Bank Name (BANK_NM) from the BANK table to Company Name in the file and the first 10 characters of the Bank Account Description (ACCT_DSCR) from the BANK to Company Entry Description in the file. When set to *No*, the process writes the first 16 characters of the Company Name (COMP_NM) from the DISF table to Company Name in the file and the first 10 characters of the Company Entry Description (COMP_ENTRY_DESC) from the DISF table to	(No Default)

	Company Entry Description in the file. The default value is *No*.	
Skip reading Company ID from BANK (SKIP_BANK_ACH_COMP_ID)	An optional output parameter that when set to *Y*, skips the overriding logic to fetch ACH Company ID from the BANK page. The COMP_ID (Company identification) attribute in Record Type 5 and 8 of the ACH XML file is updated with the Company Identification value from Disbursement Format (DISF). When set to *N*, logic to fetch the ACH Company ID value from BANK applies. The process fails in the following two cases: a. Both BANK and DISF pages do not have Company ID and the parameter is set to *N*. b. Company ID is populated in BANK page and blank in DISF with parameter set to *Y*.	(N-No Default)

Major Output

An XML file (Prefix_ChainID.xml, where Prefix is determined by the User entered parameter and the Chain ID is assigned by the job) is created. A parameter file (CWACHParm_ChainID.txt, where Chain ID is assigned by the job) is created for use in the Build Flat File job.

Job Return Code

The following table shows the potential job return codes for the Build XML File job:

Return Code	Condition
Successful (1)	All of the selected payment records are processed successfully.
Warning (4)	This Return Code is issued under the following conditions: • Not all existing records were selected for processing. • When multiple files are entered and only a few files processed successfully while a few failed due to issues other

Return Code	Condition
	than the Bank or Company ID information.
Non-Fatal Error (8)	This Return Code is issued when at least one Check Writer File is processed successfully, and at least one Check Writer File is skipped due to issues with the Bank or Company ID information.
Failed (12)	The job fails under the following conditions: • Parameters are invalid • No records that meet existence criteria were found. • No record could be found for a specified File ID • All Check Writer Files are skipped due to issues with the Bank or Company ID information. • Run time exceptions for unexpected situations When this job ends with a Return Code of Failed, subsequent jobs in the chain are set to Inactive.
Terminated (16)	This return code is issued when the job is terminated by the user. When this job ends with a Return Code of Terminated, subsequent jobs in the chain are set to Inactive.
System Failure (20)	This return code is issued when the job is terminated because of database server or network issues. When this job ends with a Return Code of System Failure, subsequent jobs in the chain are set to Inactive.

Sort Sequence

- Records from R_AP_CW_PYMT are selected by LN_NO in ascending order for each File ID

Selection Criteria

Records are selected for processing based on the following criteria:

- Department Code (if entered)
- Unit Code (if entered)
- CW File ID (if entered)
- CW ACH Status (if entered) - Note CW ACH Status must be either blank or have a value of 2 or 3, where 2 indicates “Ready for Original Generation”, 3 indicates “Ready for Re-Generation”, and <blank> indicates all records where the status is either “Ready for Original Generation” or “Ready for Re-Generation'”

Problem Resolution

The following table shows the possible return codes and recommendations for each processing step.

Step 1: Parameter Validation:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Required Parameters are not entered **Sample Message:** Department Code required when Unit is Entered.	Reschedule the job and enter the required parameters.	N/A
	Entered Parameters are not valid.	Reschedule the job and enter valid parameters.	N/A
	Failed because of runtime exceptions for an unexpected situation.	The reason for failure needs to be investigated before rescheduling the job.	N/A
Terminated (16)	Job is terminated manually by the user. **Sample message**: Job terminated by the user.	The reason for the termination needs to be investigated before rescheduling the job.	N/A
System Failure (20)	When the job is terminated because of database server or network issues. **Sample message**: Inadequate disk space.	The reason for the System Failure needs to be investigated before rescheduling the job.	N/A

Step 2: Selection of records:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	Under the following conditions: • If CW File ID(s) are specified and either Department or Unit is NOT blank, and all records are found on the CW Header table and less than all records are selected for processing. • If only one parameter is entered between CW File ID, Department and Unit and at least one exists on the CW Header, but none are selected for processing. • If no parameter values are entered between CW File ID, Department and Unit and nothing is selected for processing. **Sample Message:** No record found for CCD Format, no CCD file generated. No record found for CTX Format, no CTX file generated. No record found for PPD Format, no PPD file generated. No record found for IAT Format, no IAT file generated.	Make sure that the parameter value entered is correct. If the parameter value is incorrect, then enter the correct parameter value and run the chain again.	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Failed because of runtime exceptions for an unexpected situation.	This step is performed only if the parameter validation is successful. It can fail with fatal conditions	N/A

Possible Return Codes	Condition	Recommendation	Other Instructions
		only on encountering unknown exceptions. If that happens, investigate the exception reported by the process, resolve the error and reschedule the entire chain.	
Terminated(16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated before rescheduling the entire chain.	N/A
System Failure(20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated before rescheduling the entire chain.	N/A

Step 3: XML Generation:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	At least one Check Writer File is processed successfully, and at least one Check Writer File is skipped. Sample Messages: • Dept <CWHDR Department> File ID <CWHDR Check Writer File ID> is skipped because the Bank record cannot be	• Verify that a valid Bank record exists for each Check Writer file that was skipped. • Verify that the ACH Company ID has been provided on the Bank (BANK) table or in the Company Identification field in the ACH Information section of the Disbursement Format (DISF) table for each Check Writer file that was skipped.	N/A

Possible Return Codes	Condition	Recommendation	Other Instructions
	found. • Dept <CWHDR Department> File ID <CWHDR Check Writer File ID> is skipped because the Company ID cannot be determined from the bank or disbursement format.		
Failed (12)	• Job failed due to Fatal conditions • Failed because all Check Writer Files were skipped.	• The reason for the failure needs to be investigated before rescheduling the job. • Verify why all Check Writer files were skipped. For example, the ACH Company ID may be blank for all Check Writer files on the BANK and DISF tables or a BANK record may not exist for the Check Writer files.	N/A
Terminated(16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated before rescheduling the job.	N/A
System Failure(20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated before rescheduling the job.	N/A

Step 4: CW Header table update:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A

Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Job failed due to Fatal conditions.	The reason for the failure needs to be investigated before rescheduling the job.	N/A
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated before rescheduling the job.	N/A
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated before rescheduling the job.	N/A

CW ACH File Generation Chain: Build Flat File Job

Job Name	Build Flat File
Recommended Frequency	Daily, as part of the nightly cycle, or On Demand.
Single Instance Required	Yes
Can be restarted	No
Report Generated	No

Overview

The objective of the second job, the Build Flat File process, is to generate the ACH file following the NACHA rules for supported ACH formats (that is, PPD+, CCD+, CTX and IAT) to electronically deposit the Check Writer EFT payments. The Check Writer records for EFT payment in the ACH XML file created above are separated by the EFT format. Separate formatted ACH flat files are created for each of the CCD, CTX, PPD and IAT formats. The parameter for this job is the ACH Flat File Location directory, where the four different ACH flat files are stored.

If the APPCTRL parameter USE_CWACH_CONFIG is set to *True*, the Build Flat File job reads through the ACH Flat File Configuration page (ACHFF) for information to create the ACH Flat file(s). For any missing record types in the ACH Flat File Configuration page, the process uses existing code to create the ACH Flat file(s).

Restartability

This Job does not support the restart ability. It does not update any table; therefore, no back-out of data is required. If this batch job fails, it is necessary to reschedule the entire chain. Please see the steps mentioned in the "Problem Resolution" section of the chain.

Major Input

- The XML file created from the Build XML File job.
- The parameter file created from the Build XML File job.
- ACH Flat File Configuration (ACHFF / R_ACH_FILE_CNFG))

Batch Parameters

Parameter	Description	Default Value
ACH Flat File Location (AMSEXPORT)	Required field. Directory where flat files are created. (**** Refer to Note:** Assumptions for SWBP on page no. 7)	$$AMSEXPORT$$
Parameter File Location (AMSPARM)	Required field. Directory where the parameter file is located. (**** Refer to Note:** Assumptions for SWBP on page no. 7)	$$AMSPARM$$

Major Output

Four flat files are created for each of the ACH formats where the Prefix is determined from the user entered parameter on the Build XML File job and the ChainID is assigned by the chain job:

- Prefix_CCD_ChainID.dat
- Prefix_CTX_ChainID.dat
- Prefix_PPD_ChainID.dat
- Prefix_IAT_ChainID.dat

Job Return Code

The following table shows the potential job return codes for the CW ACH File Generation job.

Return Code	Condition
Successful (1)	Flat Files were created successfully.
Warning (4)	This return code is issued if one of the input XML files contains no records.
Non-Fatal Error (8)	N/A
Failed (12)	• Parameters are invalid

Return Code	Condition
	• The parameter file is not found in the specified directory. • Runtime exceptions encountered for any unexpected situations.
Terminated (16)	This return code is issued when the job is terminated by the user.
System Failure (20)	This return code is issued when the job is terminated because of database server or network issues.

Sort Sequence

N/A

Selection Criteria

N/A

Problem Resolution

If the job fails for any reason, then a new chain should only be scheduled. Before scheduling the new chain job, updates made by the Build XML job should be backed out. The CW File ID can be obtained from the failed job's log. On the CW Header table, ACH File Generation Status should be reset to Ready for Generation. Also the ACH File Generation Date / Time Stamp and ACH File Generation User ID values should be blanked out.

The following table shows the possible return codes and recommendations for each processing step.

Step 1: Parameter Validation:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Required Parameters are not entered.	If the required parameters are not entered, reschedule the job with valid parameters.	N/A

Possible Return Codes	Condition	Recommendation	Other Instructions
	Entered Parameters are not valid.	If the parameters are invalid, reschedule the job with valid parameters.	
	Failed because of runtime exceptions for an unexpected situation.	The reason for the failure needs to be investigated before rescheduling the job. Please refer to the "Problem Resolution" section above on rescheduling the job.	
Terminated (16)	Job is terminated manually by the user. **Sample message**: Job terminated by the user.	The reason for the termination needs to be investigated before rescheduling the job.	
System Failure (20)	Job is terminated because of database server or network issues. **Sample message**: Inadequate disk space.	The reason for the System Failure needs to be investigated before rescheduling the job.	

Step 2: Flat File Creation:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Failed because of runtime exceptions for an unexpected situation.	The reason for the failure needs to be investigated before rescheduling the job.	Please refer to the "Problem Resolution"

Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated before rescheduling the job.	section above on rescheduling the job.
System Failure (20)	Job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated before rescheduling the job.	

2.2.5 CW Archive Chain

Job Name	CW Archive
Recommended Frequency	On Demand
Single Instance Required	No
Can be restarted	Yes
Reports generated	Yes

Overview

The CW Archive process archives records on the Check Writer tables for CW files with no outstanding checks or EFT's from corresponding CW tables over a user-specified age. The archive process archives records where all associated payments have been reconciled (that is, either paid or escheated) for the CW tables (that is, Pre-Assigned Check Number, Header, Accounting, Payment, ACH Addendum, and Vendor Intercept). The process also produces output that can be used to restore records back to the corresponding Check Writer tables for user-specified CW files that were previously archived.

The CW Archive functionality utilizes the System Maintenance Utility (SMU) to perform the archive (export/delete).

This process consists of 2 individual processes:

1. **Table Record Selection** (custom selection logic) - The Selection step is the first job in the chain. The main purpose of this job is to select records for archiving from applicable Check Writer tables as per selection criteria. Using the key information from the selected check writer records, as the records are selected, the process builds SMU parameter files that are used to archive the selected records. For each generated parameter file, an Archive Facilitator table record is created. Finally, the selection process generates a report documenting all records selected for archive. The report that is generated can be viewed from the Advantage Job Manager in HTML and PDF format.
2. **Process Facilitation** (initiates SMU processes to perform archive) - The Archive step is a Facilitator process that uses a set of standard parameters required for archive processing. These parameters are chain job level parameters. The Report Only and Suppress Reports parameters are entered at the same time as the Selection Process parameters. Among all the job parameters in this job of the chain, only the "Report Only" parameter is user entered; all other parameters have default values.

This chain implements the Data Warehouse Archived Record Queue process, which is used to retain the table names and key values for records that are archived and deleted. The infoAdvantage reporting system uses this information to distinguish between records that have been archived (as valid historical data) before deletion and records that have simply been deleted from the system (as no longer needed). The process is enabled for most archiving processes when the value of Application Parameter Enable Data Warehouse Archived Record Queue (ENABLE_DW_ARCH_QUEUE) is *True*. Please see the *CGI Advantage System Administration Guide* for more information regarding this process and the Application Parameter.

Major Input

- Check Writer Pre-Assigned Check Number (CWCHK) table
- Check Writer Header (CWHDR) table
- Check Writer Accounting (CWACTG) table
- Check Writer Payment (CWPYMT) table
- Check Writer ACH Addendum (CWADNM) table
- Check Writer Vendor Intercept (CWVINCT) table

Major Output

- System Maintenance Utility parameter files with Check Writer tables record keys
- Facilitator records storing the SMU parameter filenames
- CW Archive Report
- XML file with Check Writer record information

Chain Job Return code

The following table shows the potential return codes for the Check Writer Archive Chain. Note that the Chain job ends with the highest return code across all of the jobs.

Return Code	Condition
Successful (1)	All of the jobs end successfully.
Warning (4)	One of the jobs in the chain ends with a return code of “Warning”.
Non-Fatal Error (8)	The chain job does not end with a return code of “Non-Fatal Error”.
Failed (12)	One of the jobs in the chain ends with a return code of “Failed”.
Terminated (16)	One of the jobs in the chain ends with a return code of “Terminated”.
System Failure (20)	One of the jobs in the chain ends with a return code of “System Failure”.

Problem Resolution

Please refer to the individual job “Problem Resolution” section for more details.

Check Writer Archive Chain: Selection Job

Job Name	Selection
Recommended Frequency	On Demand

Single Instance Required	No
Can be restarted	Yes
Reports generated	Yes

Overview

At a high level the selection process consists of the following steps:

5. Validate online selection process parameters - The selection process has the following standard parameters: Tolerance, File Location, Client Name, Report Only, Suppress Reports and Number of days for Archive. These chain level parameters are set by the user in the first step of the Check Writer Archive job chain.
    - Select eligible records
    - Generate input parameter files for the System Maintenance Utility identifying the actual records to be archived. The Tolerance parameter is used to control the approximate number of records to be specified in a parameter file. A new parameter file is created in the following way.
    - A new parameter file is created if archiving the next set of records because the "Archive Count" for a parameter file equals the tolerance or exceeds it by 150% of the tolerance. The new parameter file is used by a new BS Agent record identified as eligible to be selected for archiving.
    - Before creating a new parameter file, all logically related records for the current AGNT_ID being processed are written to the old parameter file. Therefore, the new parameter file begins with the set of records for a new AGNT_ID.
    - The naming convention of the file is:
      "Run Number_Sequence Number_Label_YYYYMMDD.txt"

      Where Run Number is -- Job ID of the chain

      Sequence Number -- Starting at "1" for the first parameter file within an individual run number, incremented by "1" for each new parameter file

      Label -- "CheckWriter"

      YYYYMMDD -- The system date in YYYYMMDD format

6. Update the Archive Facilitator table with one record per parameter file generated. Once a parameter file has been generated, a record is written to the Archive Facilitator table with the “Archive Type” set to “Check Writer files”. One record is written for each parameter file created.
7. Generate a custom report detailing all records selected for archive.
    - The selection process generates a report called the Check Writer Archive Selection report. The report has a standard header on every page, with fields for the Client Name, Page Number, Run Date, Run Time, and Run Number. This report lists all archived Check Writer files from the Check Writer Header table, with columns for each of the key fields (Department Code and Check Writer File ID), plus the Unit Code. There are five additional columns, one each for the Check Writer Accounting, Check Writer Payment,

Check Writer ACH Addendum, Check Writer Vendor Intercept, and Check Writer Pre-Assigned Next Number tables, populated with a count of the records archived from those tables for each Check Writer File ID.

- If the records do not meet the selection criteria, the report is still generated with a standard header and displays “No records eligible for archive selection.”
- The report that is generated can be viewed from the Advantage Job Manager in HTML and PDF format.

Process Steps	Messages
1. Batch Parameter Validation	• Validating the Batch Parameters • Parameter validation completed.
2. Selection of Records.	• Selecting eligible records • If the selection returns 0 records, then the following message is issued: “No records eligible for archive selection.” • If some records get selected the following message will be displayed: x number of records archived. x number of records selected for archival process • Processing complete
3. Report Generation.	• Rendering report started • Rendering report completed

Restartability Information

If the job fails it can be restarted after resolving the error. On restarting the job, it will start from the step where it had stopped execution earlier.

Before scheduling a new job, the record with the chain ID of the previous failed job should be removed from the FACILITATOR table. This step will reduce the redundant data in the FACILITATOR table.

The job can be rescheduled without deleting a record from FACILITATOR. No need to back out any data.

Note: The value of the chain ID for the failed job can be found by looking at the details in the Job log.

Major Input

- Check Writer Pre-Assigned Check Number (CWCHK) table.
- Check Writer Header (CWHDR) table.
- Check Writer Accounting (CWACTG) table.

- Check Writer Payment (CWPYMT) table
- Check Writer ACH Addendum (CWADNM) table
- Check Writer Vendor Intercept (CWVINCT) table

Batch Parameters

Parameter	Description	Default Value
Number Of Days ARCH_DAYS	Required Field. Number of Days for Archive must be a positive number.	No Default
Client Name CLIENT_NM	Optional. Client Name to be displayed on report.	No Default
File Location. FILE_LOCATION	Required Field. Location where the parameter file is created.	No Default
Report Only REPORT_ONLY	Required Field. Valid values are: Y – Create report only, do not perform archive. N – Perform archive and create report.	No Default
Suppress SMU Reports SUPPRESS_RPTS	Required Field. Valid values are: Y – Do not generate SMU archive reports. N – Generate SMU archive reports.	N
Tolerance TOLERANCE	Required. Tolerance must be a positive number.	1000
Progression Counter Size (PROG_CTR_SZ)	Required. This field provides the information about the record being processed.	No Default

Major Output

- Facilitator records.
- System Maintenance Utility parameter files.
- CW Archive Report.
- ARCH_DW_QUEUE_REC – The Data Warehouse Archive Record table keeps track of the values for all primary keys of each record being archived and deleted. The table name and attribute names are stored in ARCH_DW_QUEUE_TBL and linked to this record by unique ID.
- ARCH_DW_QUEUE_TBL – The Data Warehouse Archive table keeps track of the table name and primary key attribute names of each table being archived. If a record already

exists for the table being archived, it will reuse the record. If it does not yet exist, it will be created.

Job Return Code

Return Code	Condition
Successful (1)	All of the parameter validations are performed successfully and the report is generated successfully.
Warning (4)	This job does not issue a return code of Warning.
Non-Fatal Error (8)	This job does not issue a return code of Non-Fatal Error.
Failed (12)	The job will fail under the following conditions: • Parameters are invalid • Error during creation of parameter file • Run time exceptions for unexpected situations When this job ends with a return code of Failed, subsequent jobs in the chain will be set to inactive.
Terminated (16)	This return code will be issued when the job is terminated by the user. When this job ends with a return code of Terminated, subsequent jobs in the chain will be set to inactive
System Failure (20)	This return code will be issued when the job is terminated because of database server or network issues. When this job ends with a return code of System Failure, subsequent jobs in the chain will be set to inactive.

Sort Sequence

- CW Department Code
- CW File ID

Selection Criteria

Records from the CW Header table will be selected for processing based on the following criteria:

- Check Writer run status is set to "Processing Completed."
- Check Writer Intercept status is set to “Not Applicable” or “Processing Complete.”
- Payment Date is earlier than or equal to the current Application Control Date minus the Number of Days for Archive.
- No payments having a Transaction Code of CW or CE and the same Check Writer File ID and Department on the Check Reconciliation table have a status of “Disbursed.”

For all of the files selected from the Check Writer Header table, all of the records with the same CW File ID and CW Department Code will be archived from Check Writer Accounting, Check Writer Payment, Check Writer ACH Addendum, Check Writer Vendor Intercept, and Check Writer Pre-Assigned Check Number tables.

Problem Resolution

If the job fails at any point of time due to a technical issue, no table updates are necessary. The job can be rescheduled once the issue is resolved.

Step 1: Batch Parameter Validation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All batch parameter validations are successful.	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Required parameters are not entered. **Sample Message:** A file location must be specified for the Check Writer Archive process. Number of days is required The Tolerance is required.	Reschedule the job and enter the required parameters.	N/A
	Entered Parameters are not valid **Sample Message:** The user supplied Tolerance parameter value is not numeric and positive. Enter a positive numeric value and resubmit the job. The user supplied Number of Days for the parameter value	Reschedule the job and enter valid parameters.	N/A

	is not numeric and positive. Enter a positive numeric value and resubmit the job. The Suppress Report parameter can only be “Y” or “N”. Change the value and resubmit. The current value entered in the Report Only parameter is invalid. Change to “Y” or “N” and resubmit the job.		
	Failed because of runtime exceptions caused by unexpected conditions	Failure reason needs to be investigated before rescheduling the job.	N/A
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job needs to be rescheduled.	N/A
System Failure (20)	Job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job needs to be rescheduled.	N/A

Step 2: Selection of Records

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	The records meet selection criteria.	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Failed because of runtime exceptions caused by unexpected conditions.	Failure reason needs to be investigated before rescheduling the job.	N/A
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job	N/A

		needs to be rescheduled.	
System Failure (20)	Job terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job needs to be rescheduled.	N/A

Step 3: SMU parameter file creation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful.	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Failed because of issues in creating the parameter file. **Sample Message:** The value entered in the File Location parameter is not a valid directory.	Investigate the reason for failure and then re-run the job. Correct any file location issues.	N/A
	Failed because of runtime exceptions caused by unexpected conditions.	Failure reason needs to be investigated before rescheduling the job.	N/A
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job needs to be rescheduled.	N/A
System Failure (20)	Job terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job needs to be rescheduled.	N/A

Step 4: Report Generation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Report generated	N/A	N/A

	without any errors.		
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Failed because of runtime exceptions caused by unexpected conditions.	Failure reason needs to be investigated before rescheduling the job.	N/A
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job needs to be rescheduled.	N/A
System Failure (20)	Job terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job needs to be rescheduled.	N/A

Check Writer Archive Chain: Archive Job

Job Name	Archive Facilitator process
Recommended Frequency	On Demand
Single Instance Required	No
Can be restarted	Yes
Reports generated	No

Overview

This is the second job in the chain. The Facilitator process has a set of standard parameters required for archive processing. These parameters are chain job level parameters. The Report Only and Suppress Reports parameters are entered at the same time as the Selection Process parameters.

The Archive Facilitator process reads the Archive Facilitator records inserted by the Selection job, and for each record found whose Facilitator Status = “Table ready for archive” the Facilitator initiates a System Maintenance Utility process. The System Maintenance Utility is invoked with an action of “Table Archive” and archives all records identified in the related parameter file with each individual Archive Facilitator record. As the System Maintenance Utility is processing the file the Facilitator updates the “Facilitator Status” on the Archive Facilitator record being processed. The status is set to “Table Archiving” while the archive is occurring, and once it completes the status changes to “Table Archive Complete.”

There are no reports or output files created from the archive process.

The following table shows the various steps that the AD XML Creation Job goes through and the messages issued at each step.

Process Steps	Messages
1. Batch parameter validation.	• Validating Batch Parameters. • Batch Parameter validation completed.
2. Processing and archiving records	• The following messages will be issued each time a System Maintenance Utility process is spawned. • The Run Number for this archive/restore process = <Job_No> • SMU Job - <Job_No> – Spawned • SMU Job - <Job_No> - Processing completed successfully • The job slept a total of <No.> times, for a Total Sleep Time of <No.> seconds

Restartability Information

If the job was discontinued for any reason then the chain has the ability to be restarted from the point it left off.

It is compulsory to reschedule the entire chain in place of just rescheduling this job alone; otherwise, the following error message will be logged, “No Facilitator records found for Run Number <Run_No>. Nothing to process.” Since this job processes the records based on the parameter file created in the first job, an entry is made into the facilitator table for the parameter file location along with the Run_No. In this case it takes the Run_No of the current Job and there will be no record present in the Facilitator table for this; therefore, it will not be able to process any facilitator records.

Major Input

- Facilitator table
- System Maintenance Utility parameter files.
- Check Writer Pre-Assigned Check Number (CWCHK) table
- Check Writer Header (CWHDR) table
- Check Writer Accounting (CWACTG) table
- Check Writer Payment (CWPYMT) table
- Check Writer ACH Addendum (CWADNM) table
- Check Writer Vendor Intercept (CWVINCT) table

Batch Parameters

Parameter	Description	Default Value
Archive Restore ID (1-Table Archive) (ARCHIVE_RESTORE_ID)	Required non editable field. Entry of a value in this field specifies the System Maintenance Utility command to be executed during the Archive. Must be “1” for Archive.	1
Commit Block Size (COMMIT_BLOCK_SIZE)	Required field. Controls how many records are committed by the application at one time. The size should be compatible with technical capabilities and performance guidelines.	1000
Number of Processors (PROCESSOR_NO)	Required Field. Defines the number of SMU processes to be executed simultaneously.	2
Sleep Time (Frequency to pool for free processing slots) SLEEP_TIME	Defines the frequency to pool for free processing slots when initiating SMU processes.	5
Update Facilitator Status UPDATE_STATUS	If Report Only is Y, Update Status must be N.	Y

Job Return Code

Return Code	Condition
Successful (1)	All of the records archived correctly.
Warning (4)	No Facilitator records were found.
Non-Fatal Error (8)	This job does not have a return code of Non-Fatal Error.
Failed (12)	The job will fail under the following conditions: • Parameters are invalid • If any run time exception occurs
Terminated (16)	This return code will be issued when the job is terminated by the user.
System Failure (20)	This return code will be issued when the job is terminated because of database server or network issues.

Sort Sequence

N/A

Selection Criteria

Records from the Facilitator table are selected for processing based on the following criteria:

- Status should be “Table ready for archive”
- Archive Type should be “Check Writer Tables”

Problem Resolution

The following table shows the possible return codes and recommendations for each processing step.

Step 1: Batch parameter validation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All batch parameter validations are successful (that is, there are no errors raised during validations.)	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Required parameters are not entered. **Sample Messages:** ARCHIVE_RESTORE_ID is required to run the Facilitator The RUN_NO is required to run the Facilitator PROCESSOR_NO is required to run the Facilitator COMMIT_BLOCK_SIZE is required to run the Facilitator SLEEP_TIME is required to	Reschedule the job and enter the required parameters.	

	run the Facilitator		
	Entered Parameters are not valid. **Sample Messages:** The RUN_NO must be an integer greater than zero The PROCESSOR_NO must be a positive integer > 0 Invalid value received for parameter SMU_CTLG_ID (expected value >1) The SLEEP_TIME must be an integer and > 0 Invalid UPDATE_STATUS. Valid values are either “Y” or “N”	Enter valid parameters and restart the job.	
	Failed because of runtime exceptions caused by unexpected conditions.	Failure reason needs to be investigated and a new chain scheduled.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated and a new chain scheduled.	
System Failure (20)	Job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated and a new chain scheduled.	

Step 2: Processing and archiving records

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful.	N/A	N/A
Warning (4)	When no Facilitator records are found.	Make sure the Facilitator record exists and	

		schedule a new chain. .	
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Failed because of runtime exceptions caused by unexpected conditions.	Failure reason needs to be investigated and a new chain scheduled.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated and a new chain scheduled.	
System Failure (20)	Job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated and a new chain scheduled.	

2.2.6 CW Cancellation

Chain or Job Name	CW Cancellation
Recommended Frequency	On Demand.
Single Instance Required	Single. The job is configured as a Single instance due to low volume and can be run in multiple instances, if needed.
Can be restarted?	No
Reports generated	Yes. CW Transaction Summary Exception report and CW Transaction Detail Exception report.

Overview

The CW Cancellation Chain creates Check Writer Cancellation (CWC) transactions to cancel the Check Writer Payment (CWPYMT) records that are marked for cancellation (Cancellation Status is *Cancellation Requested*).

The process generates a single CWC transaction for each selected Check Writer Payment (CWPYMT) record. The number of accounting lines on the CWC transaction is based on the number of records on the Check Writer Accounting (CWACTG) table for the CW Payment record. The Event Type on the CWC transaction is inferred from the DC Event Type Crosswalk (DCXWLK) table based on the Event Type specified on the corresponding record on the CWACTG table and the Reclassification Type (that is, Cancel).

The process changes the Cancellation Status of the processed CWPYMT records as *Selected for Processing* after generating the CWC transaction XML file for the CW Payment record. On submission of the CWC transaction, the Cancellation Status of the corresponding CWPYMT record is changed to *Cancellation Processed*.

The following is a list of the job steps that make up the CW Cancellation Chain followed by an in-depth description of the jobs:

1. CWC XML Creation
2. CWC Upload
3. CWC Submit
4. Generate Reports

Major Input

- Check Writer Header (CWHDR) table
- Check Writer Payment (CWPYMT) table
- Check Writer Accounting (CWACTG) table
- Check Reconciliation (CHREC) table
- DC Event Type Crosswalk (DCXWLK) table

Major Output

- CWC transaction is uploaded and submitted to Final
- Check Writer Payment (CWPYMT) table
- CW Transaction Summary Exception report or CW Transaction Detail Exception report is generated based on the parameter value.

Chain Return Code

The following table shows the potential Return Codes for the CW Cancellation chain. The chain job ends with the highest return code across all of the jobs.

Return Code	Condition
Successful (1)	All of the jobs end successfully.
Warning (4)	One of the jobs in the chain ends with a return code of *Warning*.
Non-Fatal Error (8)	One of the jobs in the chain ends with a return code of *Non-Fatal Error*.
Failed (12)	One of the jobs in the chain ends with a return code of *Failed*.
Terminated (16)	One of the jobs in the chain ends with a return code of *Terminated*.
System Failure (20)	One of the jobs in the chain ends with a return code of *System Failure*.

Problem Resolution

Please refer to the individual job “Problem Resolution” sections for more details.

CW Cancellation Chain: CWC XML Creation

Job Name	CWC XML Creation
Recommended Frequency	On Demand
Single Instance Required	No
Can be restarted	No
Reports generated	No

Overview

The CWC XML Creation job creates CWC transactions to cancel the CWPYMT records that are marked for cancellation (Cancellation Status is *Cancellation Requested*).

The process performs following steps:

- Parameter Validations
- Selection of Records
- Generate XML file and Table Updates

Parameter Validations:

The process validates all of the batch parameters. If the parameter validation is successful then the process will go to the next step of selecting records; otherwise, the job fails by logging the appropriate message(s).

Selection of Records:

Once the parameter validation is successful, the process selects CWPYMT records marked for cancellation (Cancellation Status is *Cancellation Requested*). If CW File ID and/or CW Department parameters are entered then CWPYMT records are filtered accordingly.

The process validates the selected records on the Check Reconciliation (CHREC) table to verify the Status is *Disbursed* or *Warranted* along with matching the Payment Code (Transaction Code on CHREC), CW Department (Transaction Department on CHREC), CW File ID (Transaction ID on CHREC) and Check Number on the CWPYMT record.

If the CHREC validation is successful then the process will go to the next step of generating the XML file and updating the table for the selected CWPYMT record.

Generate XML file and Table Updates:

For the selected record, the process generates a CWC transaction to the XML file. By successfully adding the record to the XML file, the process updates the Cancellation Status to *Selected for Processing* on the CWPYMT table.

Process Steps	Messages
1. Parameter Validation	• Validating batch parameters • Parameters are valid • Parameter validation failed
2. Selection of Records	• Selecting eligible records • If the selection returns 0 records, then the following message is issued: “No eligible records found”. If CHREC record is not found for any selected CWPYMT record, and: • If Check No is present on CWPYMT, then the following message is issued: Record not found on the CHREC table for the <<CW_FILE_ID>>, <<CW_DEPT_CD>> and <<CHK_EFT_NO>>. • If Check No is blank on CWPYMT then

Process Steps	Messages
	the following message is issued: Record not found on the CHREC table for the <<CW_FILE_ID>>, <<CW_DEPT_CD>> and <<PYTM _AM>>.
3. Generate XML file and Table Updates	• <<Progression_Count>> CWPYMT records are processed. • Total <<total_count>> CWPYMT records are processed. • <<doc_count>> CWC Transactions are added to XML. • Failed to commit the transaction.

Major Input

- Check Writer Header (CWHDR) table
- Check Writer Payment (CWPYMT) table
- Check Writer Accounting (CWACTG) table
- Check Reconciliation (CHREC) table

Batch Parameters

Parameter	Description	Default Value
DOC_CD	Transaction Code. Required. The Transaction Code of the transaction that will be generated by the CW Cancellation process. If the Transaction Code is not blank, the system validates that the Transaction Code has a Transaction Type of *DC* and a Sub-type of *CW* on the Transaction Control (DCTRL) table.	CWC
DOC_DEPT_CD	Transaction Department Code. Conditionally required when the Use Transaction Department and Unit from CW Header table parameter is “N”.	No Default

	Should be valid on the Department (DEPT) table.	
DOC_PREFIX	Transaction Prefix. Required. This will be the Prefix for the CWC Transaction ID. The value should be available on the ADNT table for the Transaction Code and Transaction Department Code for the current Fiscal Year. When the Use Transaction Department and Unit from CW Header table parameter is “N”, Transaction Code and Transaction Department batch parameters will be used to retrieve the Automatic Transaction Numbering (ADNT) record. When the Use Transaction Department and Unit from CW Header table parameter is “Y”, Transaction Code and Transaction Department from the corresponding CW Header record will be used to retrieve the ADNT record.	No Default
DOC_UNIT_CD	Transaction Unit Code. Conditionally required when the Require Transaction Unit Code flag is set to *true* for the Transaction Code on the DCTRL table; otherwise, it is an optional field. If entered, it should be valid on the Unit table for the entered Transaction Department and for the current application date’s (APPCTRL table) Fiscal Year.	No Default
USE_CWHDR_DEPT_UN	Use Transaction Department and Unit from CW Header	Y

IT	table (Y/N). Required. When this parameter is set to 'N" then the User should enter the Transaction Department. When set to "Y" the Transaction Department and Transaction Unit (if entered) from the respective CW Header record is used to generate the Transaction ID.	
CW_DEPT_CD	CW Department. Optional. CW Department associated with the CW File ID.	No Default
CW_FILE_ID	CW File ID. Optional.	No Default
AMSPARM	Location where the parameter file is created. Required.	$$AMSPARM$$
AMSEXPORT	Location where output XML files are created. Required.	$$AMSEXPORT$$
USER_ID	User ID. Optional.	No Default
OVERRIDE_LVL	Override Level. Optional. Override level used in the Submit job step. Numeric value. If entered, must be an integer between 0 and 10. Cannot be specified if the Apply Overrides parameter is	No Default

	False.	
APPLY_OVERRIDES	Apply Override. Required. Apply override on transaction import and submit. Must be entered and valid values are *True* or *False*.	False
CWC_XML_FILE_NM	CWC Transaction XML File Name. Required. The name with which the CWC XML Creation job generates the CWC Transaction XML File.	CWCDocument_<<CHAINID>>.xml
UPLOAD_FILE_NM	Upload Parameter File Name. Required. Upload parameter file that is used as input to the second job of the CW Cancellation chain.	UploadCWCParm_<<CHAINID>>.txt
SUBMIT_FILE_NM	Submit Parameter File Name. Required. Submit parameter file that is used as input to the third job of the CW Cancellation chain.	SubmitCWCParm_<<CHAINID>>.txt
EXCP_RPT_TYP	Report Mode (1 – Summary, 2 – Detail). Required. Report mode for Exception Reports.	1
COMMIT_SIZE	Commit Block Size. Required. If the parameter is not entered then the value is defaulted to 10.	10
PROG_CTR_SZ	Progression Message Counter.	100

	Optional. If the parameter is not entered then the value is defaulted to 100.	

Major Output

- CWC Transaction XML File to load transactions through the second job
- CWC Upload parameter File for second job
- CWC Submit parameter File for third job
- CWC Exception Report File for fourth job
- Check Writer Payment (CWPYMT) table

Job Return Code

Return Code	Condition
Successful (1)	All of the validations are performed successfully.
Warning (4)	No eligible records found.
Non-Fatal Error (8)	When commit transaction fails for any of the block • Failed to commit transaction.
Failed (12)	The job will fail under the following conditions: • Parameters are invalid • Run time exceptions for unexpected situations. When the job ends with a Return Code of *Failed*, subsequent jobs in the chain are set to *Inactive*.
Terminated (16)	This return code is issued when the job is terminated by the user. When the job ends with a Return Code of Failed, subsequent jobs in the chain are set to Inactive.
System Failure (20)	This return code is issued when the job is terminated because of database server or network issues. When the job ends with a Return Code of Failed, subsequent jobs in the chain are set to Inactive.

Sort Criteria

The selected CWPYMT records are ordered by CW Department (CW_DEPT_CD) and CW File ID (CW_FILE_ID).

Selection Criteria

The process selects all Check Writer Payment (R_AP_CW_PYMT) records that meet the following selection criteria:

Cancellation Status is *Cancellation Requested* and

CW Department matches the CW Department batch job parameter (if populated) and

CW File ID matches the CW File ID batch job parameter (if populated)

Problem Resolution

This job cannot be restarted. If the job fails for some reason, then the reason for failure should be verified and a new job should be scheduled. The new job is scheduled irrespective of its previous jobs that failed.

Parameter Validations:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the parameters are validated successfully.	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Required parameters are not entered. **Sample Message:** "Transaction Code is required and cannot be blank"	Enter a valid Transaction Code with a Transaction Type of *DC* and Transaction Sub-Type of *CW*.	N/A
	Parameter value is not valid. **Sample Message:** "Use Transaction Department and Unit from the CW Header is required and cannot be blank. It should be either Y or N."	Enter a valid value as Y or N.	N/A

Possible Return Codes	Condition	Recommendation	Other Instructions
	Failed because of runtime exceptions for an unexpected situation.	The reason for the failure needs to be investigated before re-scheduling the job.	N/A
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can be rescheduled.	N/A
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can be rescheduled.	N/A

Selection of Records:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	If the selection returns 0 records, then the following message will be issued: **Sample Message:** "No eligible records found."	Verify CWPYMT records against selection criteria and setup records accordingly before re-scheduling the job.	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	N/A	N/A	N/A
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can be rescheduled.	N/A
System Failure (20)	When the job is terminated because of database server or	The reason for the System Failure needs to be investigated. The job	N/A

Possible Return Codes	Condition	Recommendation	Other Instructions
	network issues.	can be rescheduled.	

Generate XML file and Table Updates:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful.	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Committing processed records failed. **Sample Message:** "Failed to commit the transaction."	The reason for the failure needs to be investigated before re-scheduling the job. Manually update processed CW Payment records to meet the selection criteria so that when re-scheduling the job; all records are selected again for processing and transaction creation.	N/A
	Failed because of runtime exceptions for an unexpected situation.	The reason for the failure needs to be investigated before re-scheduling the job.	N/A
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can be rescheduled.	N/A
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can be rescheduled. Manually update processed CW Payment records to meet the selection criteria so that when re-scheduling the	N/A

Possible Return Codes	Condition	Recommendation	Other Instructions
		job; all records are selected again for processing and transaction creation.	

CW Cancellation Chain: CWC Upload

Job Name	CWC Upload
Recommended Frequency	On Demand
Single Instance Required	No
Can be restarted	No
Reports generated	No

Overview

This job step executes a SysManUtil Import to load the CWC transaction created by the CW Cancellation chain job. An upload CWC parameter text file was created by the CWC XML Creation job step specifying what CWC XML file to load.

Major Input

- Upload parameter file created during the first step of this chain.

Batch Parameters

Parameter	Description	Default Value
PARM_FILE	Parameter file. Required. Parameter file to load transactions.	$$AMSPARM$$/UploadCWCParm<<CHAINID>>.txt

Major Output

- CWC transaction is uploaded to the Transaction Catalog in the *Draft* Phase.

Job Return code

The following table shows the potential job Return Codes for the CWC Upload job.

Return Code	Condition
Successful (1)	All of the records are loaded into the Transaction Catalog successfully.
Warning (4)	This return code is issued under the following conditions: • The input file is empty • Some records failed to load into the Transaction Catalog.
Non-Fatal Error (8)	All records failed to load into the Transaction Catalog.
Failed (12)	The job fails under the following conditions: • Parameters are invalid • The input file is not found in the specified directory • Runtime exceptions encountered for any unexpected situations When the job ends with a Return Code of Failed, subsequent jobs in the chain are set to Inactive.
Terminated (16)	This Return Code is issued when the job is terminated by the user. When the job ends with a Return Code of Terminated, subsequent jobs in the chain are set to Inactive.
System Failure (20)	This Return Code is issued when the job is terminated because of database server or network issues. When this job ends with a Return Code of System Failure, subsequent jobs in the chain are set to Inactive.

Sort Sequence

N/A

Selection Criteria

N/A

Problem Resolution

If the job fails to load some or all of the transactions from the XML file into the Transaction Catalog, then a new instance of the chain should be executed with the same input parameters after correcting the issues. The transactions loaded by this job should be either processed or discarded before rescheduling the job. To process the loaded transactions, submit the CWC Submit job in that chain. This job would be set to *Inactive* since the CWC Upload failed. The CWC Submit job will submit the transactions that were generated during that CW Cancellation chain. To discard the transactions, manually identify transactions on the Transaction Catalog and perform the Discard action. After discarding transactions manually, reset the Cancellation Status on the CW Payment records to *Cancellation Requested* to match selection criteria and then records can be selected again for processing.

The following table shows the possible return codes and recommendations for each processing step.

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All parameters are valid and all of the transactions are loaded successfully.	N/A	N/A
Warning (4)	The input file is empty. **Sample message:** Could not locate valid records in the table.	Make sure that the CWC XML Creation job creates the input file with records.	N/A
	This return code is issued when the job fails to load some of the transactions. **Sample Message:** Cannot find transaction(s) to be restored.	Analyze the reason that records failed to load to the Transaction Catalog and schedule a new job using the same Parameter File (PARM_FILE).	N/A
Non-Fatal Error (8)	This return code will be issued when the job failed to load all of the transactions.	Analyze the reason that records failed to load to the Transaction Catalog and schedule a new job using the same Parameter File (PARM_FILE).	N/A
Failed (12)	This return code is issued when the parameters are not valid. **Sample Message:** Job failed because no valid parameter input file was specified.	Make sure the parameter file exits in the specified folder and reschedule the job with prior jobs disabled.	N/A
	Failed because of runtime exceptions for an unexpected condition.	If the job fails with fatal conditions on encountering unknown exceptions, then investigate the exception reported by the process, resolve the error and schedule the job with the prior jobs disabled.	N/A

Possible Return Codes	Condition	Recommendation	Other Instructions
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated before rescheduling the job.	N/A
System Failure (20)	Job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated before rescheduling the job.	N/A

CW Cancellation Chain: CWC Submit

Job Name	CWC Submit
Recommended Frequency	On Demand
Single Instance Required	No
Can be restarted	No
Reports generated	No

Overview

This job step will execute a SysManUtil Submit to submit the CWC transaction created by the CWC XML Creation job. A submit CWC parameter text file, created in the CWC XML Creation process, specifies which transactions should be submitted.

The Pre-condition Return Code for this job is *Successful*.

Major Input

- SMU job parameter file
- Draft transactions in the Transaction Catalog

Major Output

- Transactions in Pending / Final / Rejected on the Transaction Catalog
- Check Writer Payment (CWPYMT) table

Batch Parameters

Parameter	Description	Default Value
PARM_FILE	Parameter file. Required. The Parameter file to submit the transactions.	$$AMSPARM$$/SubmitCWCParm<<CHAINID>>.txt

Job Return Code

The following table shows the potential job Return Codes for the CWC Submit job.

Return Code	Condition
Successful (1)	All of the transactions submitted successfully.
Warning (4)	N/A
Non-Fatal Error (8)	N/A
Failed (12)	The job fails under the following conditions: • Parameters are invalid • Parameter file is not found • Runtime exceptions encountered for any unexpected situations • When the job ends with a Return Code of Failed, subsequent jobs in the chain are set to Inactive.
Terminated (16)	This Return Code is issued when the job is terminated by the user. When the job ends with a Return Code of *Failed*, subsequent jobs in the chain are set to *Inactive*.
System Failure (20)	This Return Code is issued when the job is terminated because of database server or network issues. When the job ends with a Return Code of *Failed*, subsequent jobs in the chain are set to *Inactive*.

Sort Sequence

N/A

Selection Criteria

N/A

Problem Resolution

If this job ends with a Return Code of *Failed*, *Terminated* or *System Failure*, then the loaded CWC transaction should be submitted manually or discarded manually depending on the issues. After discarding transactions manually, reset the Cancellation Status on the CW Payment records to *Cancellation Requested* to match selection criteria and records can be selected again for processing.

The following table shows the possible return codes and recommendations for each processing step.

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the transactions submitted successfully.	N/A	N/A
Warning (4)	N/A	The job does not issue this return code.	N/A
Non-Fatal Error (8)	N/A	The job does not issue this return code.	N/A
Failed (12)	This return code is issued when the input parameter is not found in the specified directory. **Sample Message:** Parameter file could not be located/read.	Make sure that the parameter file exits in the specified folder and schedule a new job.	N/A
	Failed because of runtime exceptions for an unexpected situation.	If the job fails with fatal conditions on encountering unknown exceptions, then investigate the exception reported by the process, resolve the error and submit the transaction manually. The Transaction ID can be found on the process logs. If the transaction cannot be submitted then discard the transaction and also reset the Cancellation Status on the CW Payment records to *Cancellation Requested* to match selection criteria so that records can be selected again for processing.	N/A
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. If the issue is resolved then the loaded transaction must be submitted manually.	N/A
System Failure (20)	Job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. If the issue is resolved then the loaded	N/A

Possible Return Codes	Condition	Recommendation	Other Instructions
		transaction must be submitted manually.	

CW Cancellation Chain: Generate Reports

Job Name	Generate Reports
Recommended Frequency	On Demand
Single Instance Required	No
Can be restarted	No
Reports generated	Yes. CW Transaction Summary Exception report and CW Transaction Detail Exception report.

Overview

The final job in the CW Cancellation Chain generates an exception report that lists all of transactions that did not submit to the *Final* Transaction Phase. When run in the *Detail* mode, the errors are also listed for each transaction from earlier submit job steps with reference to a line within the transaction.

Major Input

- CW Cancellation Exception Report parameter file (CWCancelExcepRepParm_<<CHAINID>>.txt)

Batch Parameters

Parameter	Description	Default Value
CLIENT_NM	Client Name for Report. Optional.	No Default
AMSPARM	Parameter File Location. Required. The parameter file location where the parameter file is available.	$$AMSPARM$$

Major Output

- CWC Transaction Summary Exception Report
- CWC Transaction Detail Exception Report

Job Return Code

The following table shows the potential job Return Codes for the Generate Reports job.

Return Code	Condition
Successful (1)	All of the validations are performed successfully and there are no records eligible for exception. • Generating Summary Exception Report • <<Count>> transactions are added to Summary Exception Report or • Generating Detailed Exception Report • <<Count>> transactions are added to Detailed Exception Report
Warning (4)	N/A
Non-Fatal Error (8)	N/A
Failed (12)	• Exception Parameter File is not available at Parameter File Location. • Runtime exceptions encountered for any unexpected situations.
Terminated (16)	This Return Code is issued when the job is terminated by the user.
System Failure (20)	This Return Code is issued when the job is terminated because of database server or network issues.

Sort Sequence

N/A

Selection Criteria

N/A

Problem Resolution

Job cannot be restarted in this step. If the job fails, then a new job should be scheduled after correcting the errors that caused the job to fail.

The following table shows the possible Return Codes and recommendations for each processing step.

Parameter Validations:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the parameters are validated successfully.	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Required parameters are not entered or invalid. **Sample Message:** "AMSPARM Directory does not exist on the server"	Enter a valid parameter directory.	N/A
	Failed because of runtime exceptions for an unexpected situation.	The reason for the failure needs to be investigated before re-scheduling the job. If the reason for failure lies within the current job then only a single job needs to be re-scheduled (for example, parameter file location was incorrect). Otherwise, if there is an issue with selected records then re-schedule the chain job.	N/A
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can be rescheduled.	N/A
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can be rescheduled.	N/A

Generate Exception Reports:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Exception reports are generated successfully. **Sample Message:** • Generating Summary Exception Report • <<Count>> transactions are added to Summary Exception Report or • Generating Detailed Exception Report • <<Count>> transactions are added to Detailed Exception Report	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Failed because of runtime exceptions for an unexpected situation.	The reason for the failure needs to be investigated before re-scheduling the job.	N/A
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can be rescheduled.	N/A
System Failure (20)	When the job is terminated because of database server or	The reason for the System Failure needs to be investigated. The job	N/A

Possible Return Codes	Condition	Recommendation	Other Instructions
	network issues.	can be rescheduled.	

2.2.7 CW Check/EFT Generation

Job Name	CW Check/EFT Generation
Recommended Frequency	Daily as part of the nightly cycle or On Demand. Can be run On Demand for a small set of records.
Single Instance Required	Yes
Can be restarted	Yes
Reports Generated	No

Overview

The CW Check/EFT Generation is a single batch job, which assigns check numbers for CW check payments without pre-assigned check numbers and EFT tracking numbers for CW EFT payments.

Multiple instances are not allowed for this job. If the job is run in multiple instances after un-checking the flag on the batch setup, then the data on the resultant tables may be corrupted.

The CW Check/EFT Generation process can only be run by authorized users after the CWA and CWI Generation processes and before the 1099 posting process. The system executes the Check/EFT Generation process with restrictions to verify, for each specified Check Writer File ID, the “CWA Transaction Final” flag on the CW Header table is set to “Yes” or the “Total Payment Amount” on CW Header table is 0, and the “CW Run Status” field on CW Header table is set to “Pending Check/EFT Generation.”

The following table shows the various steps that the CW Check EFT Generation Job goes through and the messages issued at each step.

Process Steps	Messages
1. Batch Parameter Validation	• Validating Batch Parameters. • Batch Parameter validation completed.
2. Generate the Check/EFT Number.	The following messages are issued when the job processes each CW file: • Selecting eligible records • Started processing File ID<CW_FILE_ID> • Assigning Check/EFT number • <No.> of records found with Payment type CW for processing • Processing Payment records with Payment type CW • Total <No.> CW records processed • Processing for Payment records with Payment type CW completed

Process Steps	Messages
	• Department: <Dept_Code> CW File ID: <File_ID> processing is complete. The same message format as above is written in the log if the payment type is CE as well. In case there are no records to process, the following message is logged: • No eligible records found. Check Writer Check/EFT Generation Process Successful.

Restartability Information

It is advisable that this job be restarted if it fails. The job reserves a block of contiguous check numbers for each Check Writer payment with a Payment Code of "CW" and a Check Number that has not been Pre-Assigned. If the job fails before all of the payments for a CW file ID are processed, those check numbers will remain reserved and if it is restarted, it will start processing the payments from where the process was stopped.

If a new instance of the job is run for the same file ID, it will still be able to use the correct check numbers which will be the next available check number, since the bank table is also updated with the last check number after the payment table is updated. Processing of the payments is based on those check numbers that are empty, which means check numbers for these payment lines have not yet been assigned and as a result there will not be any corruption of output data.

As a result there is no need to back out any updates even if the job fails. If the job fails after updating the payment table but before updating the bank table, then all updates are rolled back.

Major Input

- Check Writer Header (CWHDR / R_AP_CW_HDR)
- Check Writer Payment (CWPYMT / R_AP_CW_PYMT)
- Check Writer Vendor Intercept (CWVINCT / R_AP_CW_VINCT)
- Bank (BANK / R_AP_BANK)
- Disbursement Format (DISF / R_AP_DISB_FRMT)

Batch Parameters

Parameter	Description	Default Value
Check Writer File ID (CW_FILE_ID)	An optional selection parameter to identify a single file on Check Writer Header. Multiple values are allowed if comma-separated.	No Default
Commit Block	A required performance parameter to control how many records are committed by the	100

Parameter	Description	Default Value
(COMMIT_BLOCK)	application at one time.	
Company Name and Entry Description from Bank (CMPNY_NM_DESC_FROM_BANK)	An optional output parameter where a value of *Yes* results in the first 10 characters of the bank name and bank account description to COMP_NM (Company Name) and COMP_ENTRY_DESC (Company Entry Description) fields on the Batch Header Record (Record Type 5), respectively. With a setting of *No,* the information comes from Disbursement Format.	No
CWHDR Department (CW_DEPT_CD)	An optional selection parameter to identity records on Check Writer Header. Multiple values are not allowed.	No Default
Progression Message Block Size (PROG_CTR_SZ)	A required performance parameter used to provide information in the job log about record processing.	100
CWHDR Unit (CW_UNIT_CD)	An optional selection parameter to identity records on Check Writer Header along with the CWHDR Department. Multiple values are not allowed.	No Default

Major Output

The process updates the following tables:

- CW Payment (CWPYMT / R_AP_CW_PYMT)
- CW Pre-Assigned Check Numbers (CWCHK / R_AP_CW_CHK_NO)
- CW Header (CWHDR / R_AP_CW_HDR)
- Check Reconciliation (CHREC / R_AP_CHK_RECON)
- Paid Checks (PDCHK / AP_PD_CHK)
- Intercept Activity (INTA / AP_INCT_ACTV)

Job Return Code

The following table shows the potential job return codes for the CW Check/EFT Generation batch job.

Return Code	Condition
Successful (1)	All of the selected payment records are processed successfully.
Warning (4)	This Return Code is issued under two conditions: • Single or multiple CW File ID's are provided and all CW

Return Code	Condition
	records exist on the CW Header table, but at least one was not selected for processing • A CW File ID is not provided and one or more CW records exist on CW Header table, but none were selected for processing.
Non-Fatal Error (8)	N/A
Failed (12)	The job fails under the following conditions: • Parameters are invalid. • CW File IDs are specified and at least one CW record does not exist on the CW Header table. • CW File IDs are not specified and no CW records exist on the CW Header table for the specified parameters. • Run time exceptions for unexpected situations.
Terminated (16)	Job is terminated manually by the user.
System Failure (20)	This return code is issued when the job is terminated because of database server or network issues.

Sort Sequence

- Records are selected from R_AP_CW_HDR in ascending order of CW_DEPT_CD and CW_FILE_ID.
- Records are selected from R_AP_CW_PYMT in ascending order of CW_DEPT_CD, CW_FILE_ID and LN_NO.

Selection Criteria

Records are selected for processing based on the following criteria:

- Department Code from CWHDR matches Department Code from Job Parameter (if entered)
- Unit Code from CWHDR matches Unit Code from Job Parameter or Unit Code from Job Parameter is blank
- CW File ID from CWHDR matches Department Code from Job Parameter (if entered)
- CW Run Status from CWHDR is 'Pending Check/EFT Generation'
- CWA Transaction Final from CWHDR is 'Yes' or Total Payment Amount from CWHDR is $0.00

Problem Resolution

- If the process fails for any reason, verify if the entered parameter values are correct. The job log also gives some indication of the problem or the condition that could cause failure.

- If the job fails for any data setup reasons, correct the data setup and schedule a new job.
- Please refer the “Restartability Information” section for any data updates that need to be backed out.

The following table shows the possible return codes and recommendations for each processing step.

Step 1: Parameter validation:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Entered Department Code, Check Writer File ID, and Unit Code do not exist on CWHDR.	Enter the valid Department Code, Check Writer File ID and Unit Code.	N/A
	Department is not entered when Unit is provided. **Sample Message:** Department Code Required when Unit is entered.	Enter the Valid Department Code.	N/A
Terminated (16)	Job is terminated manually by the user. **Sample message**: Job terminated by the user.	The reason for the termination needs to be investigated before the job is rescheduled.	N/A
System Failure (20)	Issued when the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated before the job is rescheduled.	N/A

Step 2: Generate the Check/EFT Number:

This step is performed only if the eligible records selected.

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	The Check/EFT number is created successfully.	N/A	N/A
Warning (4)	At least 1 eligible record is not selected from the CWHDR table.	Confirm that selected records from the CWHDR table are valid for processing and schedule a new job if necessary.	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	If the next check number does not exist for the bank code on the Bank table of the selected records of the CWHDR.	Confirm that the next check number exists for the bank code on the Bank table of the selected records of the CWHDR. The job must be restarted.	
	The job can fail because of runtime exceptions for unexpected situations.	The reason for failure needs to be investigated before the job is rescheduled.	
Terminated (16)	Job is terminated manually by the user. **Sample message**: Job terminated by the user.	The reason for the termination needs to be investigated before the job is rescheduled.	
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated before the job is rescheduled.	

2.2.8 CW Check Printing Generation

Job Name	CW Check Printing Generation
Recommended Frequency	Daily, as part of the nightly cycle, or On Demand.
Single Instance Required	Yes
Can the job be restarted	No
Reports generated	No

Overview

The CW Check Printing Generation job in CGI Advantage Financial generates Check and Remittance Advice XML files. The XML files include the information required to support printing checks or remittance advice. Also this xml file can be used to generate any output file. The CW Check XML file contains information about check payments and the CW RA XML file includes Remittance Advice information for the EFT payments. Both XML files may contain information for multiple CW records, which can be determined by the number of Header sections in the XML file. The records posted to the Check XML File and RA XML Files are sorted by the Format Code, Department Code, Payment Code and Check/EFT Number.

This job requires that the PRINT_CW_CHECK parameter on APPCTRL is always set to Yes for achieving the baseline functionality in Advantage Financial. Checks will be printed only if the parameter is set to True. If the parameter is false, no check will be printed. Instead the job will end after generating the xml file.

Advantage BIRT Printing APIs convert these XML files from Physical Files to BIRT compatible XML Files. These BIRT compatible XML files are copied to the folder (folder named 'input') that is configured to poll continuously by the BIRT server. The BIRT server then maps the necessary field of the XML file with the form design (*.rptdesign). The Name of the form to be picked up is mentioned in the json file. If the mapping of fields with corresponding form is completed then XML files are moved to another folder (generally to a folder named 'backup'). If there are any issues while mapping the field, XML files are moved to a 'failure' folder. The BIRT server then generates a corresponding log report for the XML file.

Restartability Information

This Job does not support restart ability. If the job fails for any reason, only a new job should be scheduled. Please see the "Problem Resolution" section for more information about the data restore.

Major Input

Data from the following tables:

- Application control Parameter table
- Check Writer Header (CWHDR) table
- Check Writer Payment (CWPYMT) table
- Check Writer Vendor Intercept (CWVINCT) table

Batch Parameters

Parameter	Description	Default Value
Check/RA XML File Directory Name (AMSEXPORT)	Required Field. Location where output XML files are created. (** **Refer to Note:** Assumptions for SWBP on page no. 7)	$$AMSEXPORT$$
Parameter File Location (AMSPARM)	Required Field. Location where the parameter file is created. (** **Refer to Note:** Assumptions for SWBP on page no. 7)	$$AMSPARM$$
Application Resource Identifier (APPL_RSRC_ID)	Required Field.	CW_PRN
CW Check XML File Prefix (CHECK_FL_NM_PRE)	Required Field. Prefix for the output Check XML file.	Check
Check Printing Status (CHK_PRN_STA)	Optional. Valid values are: 2 - Ready for Original Printing 3 - Ready for Re-Printing If no value is entered, then both 2 and 3 are selected.	2
Client Name (CLIENT_NM)	Optional. Client Name.	No Default
Check Writer File ID (CW_FILE_ID)	Optional. Multiple File IDs can be entered by separating them with commas.	No Default
Department Code (DEPT_CD)	Optional. Required if Unit Code is entered.	No Default
Print Job Name (PRINT_JOB_CD)	Required Field.	No Default
Print Resource (PRINT_RSRC)	Required Field.	No Default
Progression Message Block Size (PROG_CTR_SZ)	Required Field. This field provides the information about the record being processed.	No Default

Parameter	Description	Default Value
CW RA XML File Prefix (RA_FL_NM_PRE)	Required Field.	RA
Skip Mailing Address (SKIP_MAIL_ADD)	Required Field. It should be either Yes or No	YES
Unit Code (UNIT_CD)	Optional.	No Default
User Name 1 (USER_NAME_1)	Optional. User Name 1.	No Default
User Name 2 (USER_NAME_2)	Optional. User Name 2.	No Default
Prevent PYMT_REMT_ADV white space trimmings (PYMT_REMT_ADV_NO_TRIM)	An optional output parameter that prevents trimming of leading and trailing spaces in the PYMT_REMT_ADV attribute on the output XML file generated. Valid values are • Y- Prevents trimming of spaces and • N- Trims leading and trailing spaces, which is the default if left blank.	N

Major Output

- Check XML File named as CWCheckPrinting.xml
- RA XML File named as CWRAPrinting.xml
- XML File compatible with the BIRT server for printing check and/or remittance advice

The following table shows the various steps that the Check Printing Generation Job goes through and the messages issued at each step.

Process Steps	Messages
1. Parameter Validation	Validating Batch Parameters. • If the parameter is invalid, the error messages will be displayed in the log. • Batch Parameter validation completed

2. Check for record existence	The following message is issued if no records are found: • No record found for specified File ID
3. XML Generation	A message is written to the log every time the job finishes processing a CW record. The following messages are logged in the log: • Processing Header records • Department <Dept_Code>, Unit <Unit>,CW File ID <File_ID> Processing is Complete. • "N" Header records processed • Rendering report started. • Rendering report completed • "N" indicates the Number of Records.

Job Return Code

The following table shows the potential job return codes for the Check Printing Generation job.

Return Code	Condition
Successful (1)	All of the selected payment records are processed successfully.
Warning (4)	Not all of the existing records were selected for processing.
Non-Fatal Error (8)	N/A
Failed (12)	The job fails under the following conditions: • Parameters are invalid. • No records that meet existing criteria were found. • No record could be found for a specified File ID. • Run time exceptions for unexpected situations.
Terminated (16)	This return code is issued when the job is terminated by the user.
System Failure (20)	This return code is issued when the job is terminated because of database server or network issues.

Sort Sequence

The records posted to the Check XML File and RA XML File are sorted by:

- Format Code
- Department Code
- Payment Code
- Check/EFT Number

Selection Criteria

Records are selected for processing based on the following criteria:

- Department Code (if entered)
- Unit Code (if entered)
- CW File ID (if entered)
- CW Print status (if entered)

If no CW Print Status is entered, records with a Status of either “Ready for Original Printing” or “Ready for Re-printing” are selected.

Problem Resolution

If the Job fails due to some reason, then updates to the table are rolled back. The job can be rescheduled without any need of backing out data.

If job is successful but no check/RA gets printed, then some of the probable causes are:

1. Verify that the Printer is online and on network and has sufficient papers in it.
2. Verify whether the XML file generated by the process is copied under the correct folders in the BIRT server. If not, copy the XML file manually into the folder that is continuously polled by the server.
3. Verify the BIRT server is up and running; if it is not, start the server up.
4. Verify the BIRT server is configured to poll the correct folder where XML file is copied by the process. If not, then configure the BIRT server.
5. Verify whether the XML file is not rejected because of some reason. If it is rejected then resolve the issue mentioned in the log file of the BIRT server.

In a situation where the issue is because of a printer error (for example, a paper Jam), the BIRT server fails the printing and moves the XML file into a “failure” folder. It generates an associated log for error.

Users should resolve the printer specific error and copy the same XML file into the “input” folder (the folder which is continuously polled by the server). Restart the server and it will print the check again. It is important to note that this printing will start printing the check from the beginning. In this case, if pre-printed stationary is used then the user needs to re-run the job by backing out data as listed below:

1. Set the “Check Print Status” on CWHDR to “Ready for Re-Printing” for records where ‘Check XML Generate Date’ = <date of running the job> and ‘Check XML Generate User’ = <user who ran the job>.
2. Job run after setting the value mentioned above ensures that it will reprocess the records which were spoiled during the previous printing on pre-printed stationary.

Note: No data back-out is required if the printing is done with plain stationary.

The following table shows the possible return codes and recommendations for each processing step.

Step 1: Parameter Validation:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	N/A.	N/A
Failed (12)	Required Parameters are not entered **Sample Message:** Department Code required when Unit is Entered.	Reschedule the job and enter the required parameters.	N/A
	Entered Parameters are not valid. **Sample Message**: Invalid Check Printing Status parameter.	Reschedule the job and enter valid parameters.	
	Failed because of runtime exceptions for an unexpected situation.	The reason for the failure needs to be investigated before the job is rescheduled.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated before the job is rescheduled.	N/A
System Failure (20)	When the job is terminated because of database server or network issues.	The reason of System failure needs to be investigated before the job is rescheduled.	N/A

Step 2: Check for Record Existence

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	N/A	N/A.	N/A
Non-Fatal Error (8)	N/A	N/A	N/A.
Failed (12)	Job failed due to No records existing.	Schedule a job with different parameters if needed.	N/A
	Job failed due to no records being found for a specified CW File ID.	Schedule a job with different parameters, if needed.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated before the job is rescheduled.	N/A
System Failure (20)	Issued when the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated before the job is rescheduled.	

Step 3: XML Generation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Job failed due to Fatal conditions.	Job will fail if it encounters any runtime exceptions.	N/A

Possible Return Codes	Condition	Recommendation	Other Instructions
		Investigate the exception reported by the process, resolve the error and restart the job.	
Terminated(16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated before the job is rescheduled.	N/A
System Failure(20)	Issued when the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated before the job is rescheduled.	N/A

2.2.9 CW Check Printing Process

Chain or Job Name	Check Writer Check Printing Process
Recommended Frequency	Nightly after Check Writer processes have run
Can be restarted?	Refer to the Restart Information section
Reports generated	Yes. The following reports are generated: • Summary Report • Detail Report

Overview

The Check Writer Check Printing Process is an alternate process to the CW Check Printing Generation process to create output files in a different layout for external printing. This alternative process consists of two batch jobs and a report, which generate fixed-length files of checks to be printed. The first job step of the alternate process is the same as the primary printing job, but then the two differ.

The Check Writer Check Printing process consists of the following job steps:

Check Writer Check Printing Selection

Validate the existing batch parameters to be used and key inputs to the process before proceeding to the initialization and record selection steps.

Initialize the staging table by purging any existing records.

Select records for processing from Check Writer Header and Payment.

With each Check Writer Header record processed, updates are made for the following:

- Check Printing Generation User ID with the value corresponding to the User ID that initiated the batch job.
- Check Printing Generation Process Date/Time Stamp to the server date at the time of processing.
- Check Print Status to *Printed* for a record printed for the first time. If the record was already printed then set to *Ready for Re-printing*.

Update staging table:

- Set Address File Type to “D” (department pickup) if the Payment Department on the respective CWPYMT record matches one of the department codes specified on the Payment Department Code batch parameter.
- Set Address File Type to “L” (local address) if the Country Code is “USA” or "US" or blank.
- Set Address File Type to “F” (foreign address) if the Country Code does not equal “USA” or does not equal "US" or is not blank.
- Assign a File ID based on the following naming convention:

  CW_[Check Format]_YYYYMMDD_HHMM

Date comes from the application system date. Time stamp comes from the server time. For example, a file for payroll records for the batch process being run on May 19, 2015 at a 10:30 am server time would display the following in the File ID field:

CW_PAY_20150519_1030

Establish the written amount by running a routine against the Net Payment Amount field and by populating the Amount in Words field (PYMT_AM_WRD). This routine captures the part to the left of the decimal point of the Net Payment Amount and displays it in words. Next, it appends "&" and then appends the two characters after the decimal and then appends "/100".

Sort the records on the staging table using the Sort Criteria defined below.

Job Steps	Messages
Check Writer Check Printing Selection	The following messages will be displayed: • Validating batch parameters • Parameter validation completed Parameters are valid or invalid depending on the validation. If the parameter is invalid, then the message for invalid parameters will be displayed in the log. It will be followed by the message "Batch parameter validation failed." • Initializing Staging Table Complete • List of Department/Unit/CW File ID being written to the staging table where each message will look like the following: Department XXX, Unit XXXX, CW File ID XXXXXXXXXXXX Processing is Complete • Not all existing records were selected for processing • Updating staging table • Updating staging table complete

Check Writer Print File Creation Selection

Validate the batch parameters before proceeding to the record selection steps.

Select all records from the staging table to create the print files.

Create three fixed-length check files using several batch parameters.

All fields except for the amount fields within the check print files are left justified. The amount fields in the check print files are right justified.

If no data exists for a specific Check Format, there will be no file created for that Check Format during that run of the process.

Job Steps	Messages
Check Writer Print File Creation	The following messages will be displayed: • Validating batch parameters • Parameter validation completed Parameters are valid or invalid depending on the validation. If the parameter is invalid, then the message for invalid parameters will be displayed in the log. It will be followed by the message "Batch parameter validation failed." • Begin processing IIT files • Completed processing IIT files • Begin processing PAY files • Completed processing PAY files • Begin processing RET files • Completed processing RET files • Files successfully created

Check Writer Control Report Generation

Validate the batch parameters before proceeding to the record selection steps.

Select all records from the staging table to create the reports.

Create the Summary Report.

Create the Detail Report.

Job Steps	Messages
Check Writer Control Report Generation	The following messages will be displayed: • Validating batch parameters • Parameter validation completed Parameters are valid or invalid depending on the validation. If the parameter is invalid, then the message for invalid parameters will be displayed in the log. It will be followed by the message "Batch parameter validation failed." • Reports output folder mapped • HTML report file path: <URL for path> • PDF report file path: <URL for path> • Rendering report started • Rendering report completed • Summary Report Generated Successfully • Report output folder mapped • HTML report file path: <URL for path>

Job Steps	Messages
	• PDF report file path: <URL for path> • Rendering report started • Rendering report completed • Detail Report Generated Successfully • Control Report Generated Successfully

Restart Information

If the first or second steps within the chain job fail due to any reason, a new job should be scheduled from the beginning after correcting the errors that caused the job to fail.

If the third step in the chain job fails, the first and second steps in the chain job that completed successfully will need to be inactivated when scheduling a new job after correcting the errors that caused the third step to fail.

Major Input

Tables

The following is a list of tables that are primary input for the Check Writer Check Printing Process:

Check Writer Header (CWHDR): R_AP_CW_HDR

Check Writer Payment (CWPMT): R_AP_CW_PYMT

Batch Parameters

Check Writer Check Printing Selection

Parameter	Description	Default Value
Check Print Status (CHK_PRN_STA)	Required choice to control printing and record selection: • 2 – Ready for Original Printing • 3 – Ready for Re-Printing, allows the file to be recreated in the event it was previous compromised. If blank, both 2 and 3 are selected.	2
Commit Block Size (COMMIT_BLOCK)	Required performance parameter to control how often database transactions will be committed. If left blank 1000 will default.	1000
Check Writer File ID (CW_FILE_ID)	Optional parameter for record selection. Multiple File IDs allowed if separated with commas. If left blank, then all files that match the Check Print Status, Department, and Unit parameters will be selected.	No Default
Selection	Optional selection parameter that allows	No Default

Parameter	Description	Default Value
Department (DEPT_CD)	multiple values if comma delimited.	
Select Block Size (SELECT_BLOCK)	Required parameter to control check writer record selection for a processing round. If left blank 1000 will default.	1000
Progression Counter (PROG_CTR_SZ)	Required parameter to control when the job log updates with a progress status. If left blank, 1000 will default.	1000
Payment Document Code (PYMT_DEPT_CD)	Optional parameter used to specify one or more department codes (delimited by commas) that will match to the Payment Department of Check Writer Payment records and set the Address Flat File Type to D (Department Pickup).	No Default
Selection Unit (UNIT_CD)	Optional selection parameter that allows multiple values if comma delimited. If used, the Selection Department parameter is required.	No Default

Check Print File Creation

Parameter	Description	Default Value
File Location (AMSEXPORT)	Required location where the fixed length files will be written.	$$AMSROOT$$/ExportImport
Tax Refunds File Name (IIT)	Required file name for the Treasury Tax Refund records. The value supplied is appended as specified in the Overview section.	CW_IIT
Payroll File Name (PAY)	Required file name for the Payroll records. The value supplied is appended as specified in the Overview section.	CW_PAY
Retirement File Name (RET)	Required file name for the Retirement records. The value supplied is appended as specified in the Overview section.	CW_RET
Select Block Size (SELECT_BLOCK)	Required parameter to control how many check writer records are selected for processing at a time. If left blank 1000 will default.	1000

Check Writer Control Report Generation

Parameter	Description	Default Value
Client Name (CLIENT_NM)	Optional name to appear on report headers.	None
Report Type (RPT_TYP)	Required parameter to control report output. • 1 – Summary • 2 – Detail to show vendor and payment number details. • 3 – Both	3

Major Output

Check Writer Check Printing Selection

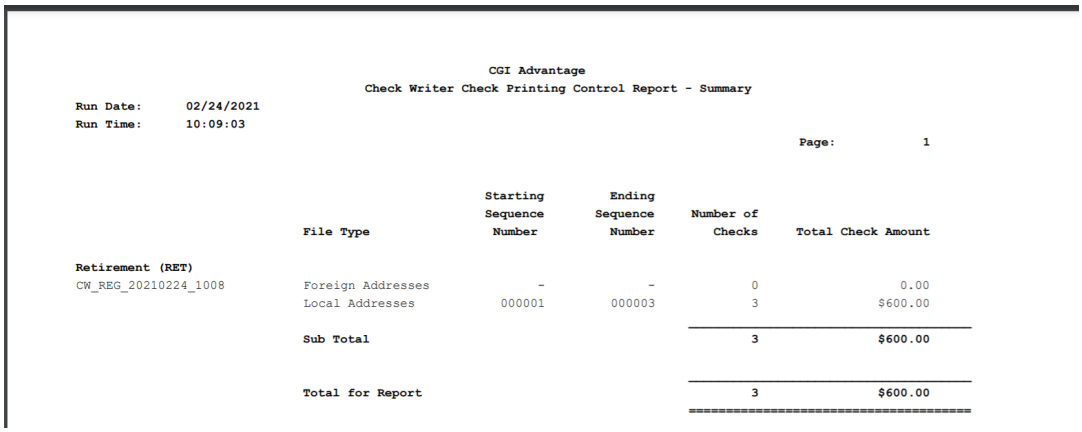
- Check Writer Header (CWHDR): R_AP_CW_HDR
- Staging table: R_CW_CHK_PRNT

Check Print File Creation

- IIT: CW_IIT_YYYYMMDD_HHMM.txt
- PAY: CW_PAY_YYYYMMDD_HHMM.txt
- RET: CW_RET_YYYYMMDD_HHMM.txt

Check Writer Control Report Generation

- Summary Report



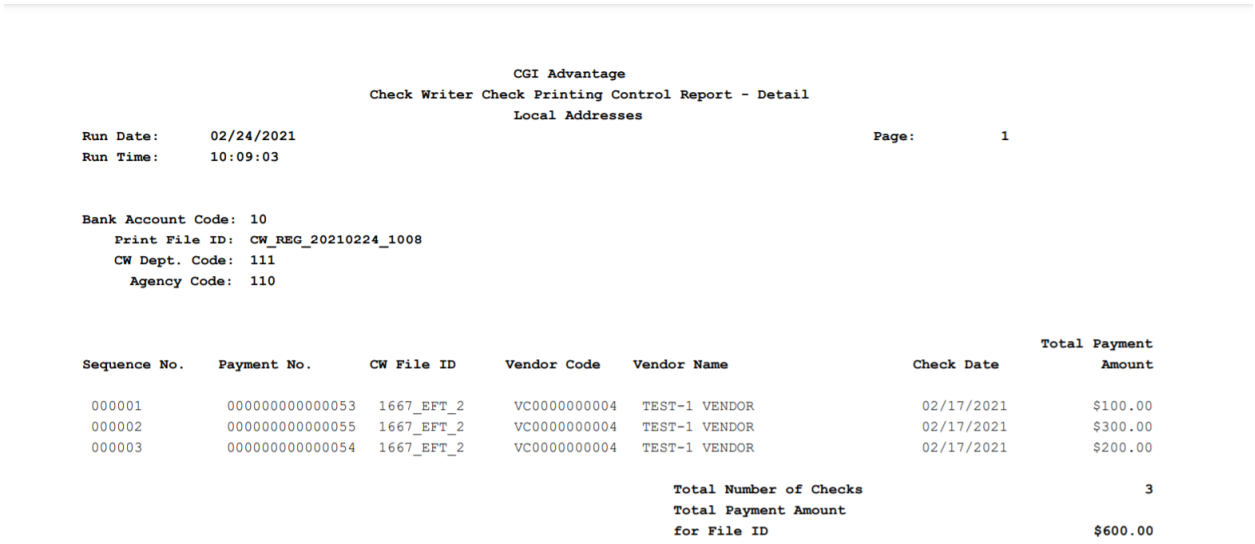
CGI Advantage
Check Writer Check Printing Control Report - Summary

Run Date: 02/24/2021
Run Time: 10:09:03

Page: 1

File Type	Starting Sequence Number	Ending Sequence Number	Number of Checks	Total Check Amount
Retirement (RET)				
CW_REG_20210224_1008	Foreign Addresses	-	-	0	0.00
	Local Addresses	000001	000003	3	$600.00
	Sub Total			3	$600.00
	Total for Report			3	$600.00

- Detail Report



CGI Advantage
Check Writer Check Printing Control Report - Detail
Local Addresses

Run Date: 02/24/2021 Page: 1
Run Time: 10:09:03

Bank Account Code: 10
Print File ID: CW_REG_20210224_1008
CW Dept. Code: 111
Agency Code: 110

Sequence No.	Payment No.	CW File ID	Vendor Code	Vendor Name	Check Date	Total Payment Amount
000001	000000000000053	1667_EFT_2	VC0000000004	TEST-1 VENDOR	02/17/2021	$100.00
000002	000000000000055	1667_EFT_2	VC0000000004	TEST-1 VENDOR	02/17/2021	$300.00
000003	000000000000054	1667_EFT_2	VC0000000004	TEST-1 VENDOR	02/17/2021	$200.00

Total Number of Checks 3
Total Payment Amount for File ID $600.00

Job Return Codes

The following table shows the potential return codes for the Check Writer Check Printing Process. Note that the Chain job will end with the highest return code across all of the jobs.

Check Writer Check Printing Selection

Return Code	Condition	Recommendation
Successful (1)	Job completes successfully when all of the selected records are processed successfully to the staging tables and updates are made to selected records.	N/A
Warning (4)	Job finishes with a warning when no eligible records are found for processing.	Verify that records exist on CWHDR that meet the criteria specified in the parameters and are marked Ready for Original Printing or Ready for Re-printing.
Non-Fatal Error (8)	This job step does not use this return code	N/A
Failed (12)	Job failed under the following conditions: • Parameters are invalid. • Run time exceptions for unexpected situations.	The reason for the job failure needs to be investigated before scheduling a new job.
Terminated (16)	This return code will be issued when the job is terminated by the user.	The reason for the termination needs to be investigated before scheduling a new job.
System Failure (20)	This return code will be issued when the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated before scheduling a new job.

Check Print File Creation

Return Code	Condition	Recommendation
Successful (1)	Job completes successfully when all of the selected records are processed successfully to one or more output files.	N/A
Warning (4)	The job does not end with this return code.	N/A
Non-Fatal Error (8)	The job does not end with this return code.	N/A
Failed (12)	Job failed under the following conditions: • Parameters are invalid. • Run time exceptions for unexpected situations.	The reason for the job failure needs to be investigated before scheduling a new job.
Terminated (16)	This return code will be issued when the job is terminated by the user.	The reason for the termination needs to be investigated before scheduling a new job.
System Failure (20)	This return code will be issued when the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated before scheduling a new job.

Check Writer Control Report Generation

Return Code	Condition	Recommendation
Successful (1)	Job completes successfully when all of the selected records are processed successfully to one or more output files.	N/A
Warning (4)	The job does not end with this return code.	N/A
Non-Fatal Error (8)	The job does not end with this return code.	N/A
Failed (12)	Job failed under the following conditions: • Parameters are invalid. • Run time exceptions for unexpected situations.	The reason for the job failure needs to be investigated before scheduling a new job.
Terminated (16)	This return code will be issued when the job is terminated by the user.	The reason for the termination needs to be investigated before scheduling a new job.
System Failure (20)	This return code will be issued when the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated before scheduling a new job.

Sort Criteria

The following steps are used to sort and sequence the records.

1. Sort the Address File Type in the following order per each Check Format (IIT, PAY and RET):

    a. First, departments designated for manual pickup (Address File Type 'D') – these are sorted by Payment Department Code in ascending order.

    b. Next checks for Foreign addresses (Address File Type "F") – these are sorted by Country Code in ascending order.

    c. Lastly checks for Local addresses (Address File Type "L") – these are sorted by Zip Code in ascending order.

2. Set Check Sequence Number for the first sorted check with XXX Check Format to "000001" then increment it by "1" for subsequent checks with the same Check Format. There will be no gaps in sequence numbers.
3. Repeat the sequencing beginning with "000001" for the next sorted check with the next XXX Check Format then increment it by "1" for subsequent checks with the same Check Format.
4. Repeat the sequencing beginning with "000001" for the third sorted check with the next XXX Check Format then increment it by "1" for subsequent checks with the same XXX Check Format.

Selection Criteria

Records are selected for processing based on the following criteria:

- Department Code (if entered)
- Unit Code (if entered)
- CW File ID (if entered)
- CW Print status (if entered)

Problem Resolution

Because this chain job is a part of a larger Check Writer process, there are certain restrictions that apply to the Check Printing Generation, in terms of when it should be run:

- The chain job can be run only after the Check/EFT Generation process has been completed. The system executes the job with those restrictions by verifying that for each specified Check Writer File ID:

Check Print Status field on CW Header is *Ready for Original Printing* when the Check Printing Status parameter is *Ready for Original Printing* or blank

Check Print Status field on CW Header is *Ready for Re-printing* when the Check Printing Status parameter is *Ready for Re-printing* or blank.

The system also determines if the CW File Includes Check/EFT with Remittance Advice field on CW Header is set to *Only Checks* for specified CW file(s). Therefore, the batch job only selects CW records for which the conditions described above are true.

If the Check Printing Status field on CW Header is set to *Ready for Original Printing* and the CW File Includes Check/EFT with Remittance Advice field on CW Header is set to *Only Checks* for the specified CW record(s) to be processed, it means that CW record(s) contain check payments only. Therefore, the process creates staging table records based on these CW records if the Cancellation Status on the CW Payment is set to *N/A* or is blank. CW warrants with other Cancellation Statuses (i.e. *Cancellation Requested, Selected for Processing,* and *Cancellation Processed*) should not be selected for printing.

If the Check Printing Status field on CW Header is set to *Ready for Original Printing* and the CW File Includes Check/EFT with Remittance Advice field on the CW Header table is set to *Only RA*

for the specified CW record(s) to be processed, it means that CW record(s) contain EFT payments with Remittance Advice only. Therefore, the process does not select the CW records to be added to the staging table. If specified CW record(s) do not contain any check payments, then the process generates an empty staging table that does not contain any information.

2.2.10 CW Clean Up

Job Name	CW Clean Up
Recommended Frequency	On Demand CW Clean Up job should be run after the CW Table Load chain job or CW Pre-Edit chain job, as needed.
Single Instance Required	No
Can be restarted	No
Reports Generated	No

Overview

The CW Clean Up job is run by authorized users who are able to delete only their respective department's CW records. The CW records are deleted allowing the user to make changes to the CW file and resubmit. The CW records are selected for deletion by the Clean Up parameters. These parameters include CW Department Code, Unit Code, CW File ID, and Delete All. Only the CW Department field is required. The CW File ID can be entered for a single file, or multiple files when separated by commas. The Delete All field is set to "Yes" to delete all records that are un-certified. The Delete All field may be used in combination with other parameter values to delete all records within the user specified CW Department and or Unit codes that are un-certified. The Clean Up job only selects CW records that have been un-certified by both the central and department offices and that have not been through any of the succeeding batch jobs.

The following table shows the various steps that the CW Clean Up job goes through and the messages issued at each step.

Process Steps	Messages
1. Parameter Validation	Validating Batch Parameters. • If all of the parameters are validated successfully, the following messages are displayed in the log: "Parameters are valid." • Parameter validation completed.
2. Clean Up Process	Selection of the CW records. • The following message is displayed if no records are selected for deletion: "No records selected for deletion." and the process completes with a Non-Fatal Error as the Return Code. • The following messages are displayed if the records are deleted successfully: • "Number of CW Accounting Records deleted for CW File id: xxxx and CW Dept Cd:yyyy = z." • "Number of CW Payment Records deleted for CW File id: xxxx and CW Dept Cd:yyyy = z."

Process Steps	Messages
	• "Number of CW Addendum Records deleted for CW File id: xxxx and CW Dept Cd:yyyy = z." • "Number of CW Header Records deleted for CW File id: xxxx and CW Dept Cd:yyyy = z." (Where xxxx is the CW File ID and yyyy is the CW Dept Code and z is the count of records deleted.) And the process completes with a Return Code of Successful.

Restartability

This job cannot be restarted. If the job fails due to any reason, the user can schedule a new job.

If there are multiple files entered, and the job failed after processing a few files, then a new job can be scheduled to process the remaining files.

Major Input

Data from the following tables:

- Check Writer Header (R_AP_CW_HDR) table
- Check Writer Payment (R_AP_CW_PYMT) table
- Check Writer Accounting (R_AP_CW_ACTG) table
- Check Writer Addendum (R_AP_CW_ADNM) table

Batch Parameters

Parameter	Description	Default Value
CW Department Code CW_DEPT_CD	CW Department Code CW Department Code is required when Delete All is set to No, and it is Optional when Delete All is set to Yes.	<blank>
CW File ID CW_FILE_ID	The CW File ID(s) to be processed in the Clean Up. Multiple File IDs can be entered, separated by a comma.	<blank>
CW Unit Code CW_UNIT_CD	Unit	<blank>
Delete All parameter DELETE_ALL	Delete All Possible values are Yes or No.	No

Parameter	Description	Default Value
Progression Message Block Size PROG_CTR_SZ	Required. This field provides the information about the record being processed.	No Default.

Major Output

Delete records from the following tables for the specified CW File ID:

- Check Writer Header (R_AP_CW_HDR) table
- Check Writer Payment (R_AP_CW_PYMT) table
- Check Writer Accounting (R_AP_CW_ACTG) table
- Check Writer Addendum (R_AP_CW_ADNM) table

Job Return Code

The following table shows the potential job return codes for the CW Clean Up job:

Return Code	Condition
Successful (1)	All of the CW File ID's are deleted successfully.
Warning (4)	When multiple files are entered and a few of them are processed, a Return Code of Warning is issued.
Non-Fatal Error (8)	When multiple files are entered and No records are processed for deletion a return code of Non-Fatal Error is issued.
Failed (12)	The job will fail under the following conditions: • Parameters are invalid • Technical/System failure
Terminated (16)	This Return Code is issued when the job is terminated by the user.
System Failure (20)	This Return Code is issued when the job is terminated because of database server or network issues.

Sort Sequence

N/A

Selection Criteria

Records are selected for processing based on the following criteria:

- CW File ID with the Department Certification Flag set to "No"

Problem Resolution

If the process fails for any reason, verify that the parameters entered are correct. The job log also gives some indication of the problem or the condition that could cause failure.

Step 1: Parameter Validation:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All parameter validations are successful (that is, there are no errors raised during validations.)	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	This step does not issue this Return Code.	N/A
Failed (12)	Required parameters are not entered. **Sample Message**: Department is required when Unit is entered. **Recommendation:** Enter a Department when Unit is entered.	Enter missing required parameters and reschedule the job.	N/A
	Entered Parameters are not valid. **Sample Message:** CW Clean Up: Invalid Batch Parameters **Recommendation:** Enter a valid Department.	Enter valid parameters and reschedule the job.	N/A
	Failed because of runtime exceptions caused by unexpected conditions.	Failure reason needs to be investigated before rescheduling the job.	N/A
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can then be rescheduled.	N/A
System Failure (20)	Job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can then be rescheduled.	N/A

Step 2: CleanUp process:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	The selected records deleted successfully from the CWHDR, CWPYMT, CWACTG and CWADNM tables.	N/A	N/A
Warning (4)	When multiple files are entered and a few files are yet to be processed for deletion, a Warning is issued.	Schedule a new job for the remaining files.	N/A
Non-Fatal Error (8)	When multiple files are entered and no files are processed for deletion, a Non-Fatal Error is issued.	Verify whether the File IDs are available for deletion.	N/A
Failed(12)	Failed because of runtime exceptions caused by unexpected conditions.	Failure reason needs to be investigated before rescheduling the job.	N/A
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can then be rescheduled.	N/A
System Failure (20)	The job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can then be rescheduled.	N/A

2.2.11 CW EFT Return Process

Chain Name	CW EFT Return Process
Recommended Frequency	It is recommended this process be run daily before the Check Writer Cancellation chain during the nightly cycle.
Single Instance Required	Yes
Can be restarted	No
Reports Generated	Yes

Overview

The CW EFT Return Process is a chain job that processes the CW EFT Returns from the bank and initiates the cancellation for returned CW EFT payments. The process also marks the vendors as ineligible for EFT at the Vendor Location level and Address level, if applicable.

The chain job is comprised of the following four batch jobs:

- Flat to XML File
- Load XML File
- Process CW EFT Returns
- Initiate CW Payment Cancellation

Major Input

- Return ACH file (CTX, PPD, or CCD)
- Check Reconciliation table
- Paid Check table
- Check Writer Payment table
- Check Writer EFT Returns table
- Check Writer EFT Reversal table

Major Output

- Check Writer EFT Returns table
- Check Writer EFT Reversal table
- Vendor Customer table
- Address table
- Check Writer Payment table
- Returned CW EFT Report
- Returned CW EFT Exception Report

- Initiated Cancellations of Returned CW EFT Payments Report
- Cannot Initiate Cancellations of Returned CW EFT Payments Exception Report

Chain Job Return Code

The following table shows the potential job Return Codes for the CW EFT Return Process:

Return Code	Condition
Successful (1)	All of the jobs end successfully.
Warning (4)	One of the jobs in the chain ends with a Return Code of *Warning*.
Non-Fatal Error (8)	One of the jobs in the chain ends with a Return Code of *Non-Fatal Error*.
Failed (12)	One of the jobs in the chain ends with a Return Code of *Failed*.
Terminated (16)	One of the jobs in the chain ends with a Return Code of *Terminated*.
System Failure (20)	One of the jobs in the chain ends with a Return Code of *System Failure*.

Problem Resolution

If any of the jobs in the chain encounters a Non-Fatal Error, it is advisable to review and correct the errors as appropriate, and reschedule the job as needed. When a Non-Fatal Error occurs in the CW EFT Return Process, some records may still be processed successfully. It is recommended to configure the chain job Pre Condition Return Code as follows to allow successfully processed records to complete processing.

Job Name	Pre Condition Return Code
Flat to XML File	Successful
Load XML File	Warning
Process CW EFT Returns	Successful
Initiate CW Payment Cancellation	Non-Fatal Error

Please refer to the individual jobs for details regarding the specific job processes and problem resolution.

CW EFT Return Process: Flat to XML File

Job Name	Flat to XML File
Recommended Frequency	Daily as part of the nightly cycle or on demand.

Single Instance Required	Yes
Can be restarted?	No
Reports Generated	No

Overview

This job converts the input ACH file received from the Bank into an XML file containing formatted data that is ready to be uploaded to the system. The bank can send a Returned ACH File with CW EFT payments, CW EFT reversals and/or CW EFT Prenotes ($0 Payment Amount on CWPYMT table).

The steps involved in this process are:

4. **Parameter Validation**: First the process validates the batch parameters.
5. **Create XML:** The process extracts specific information from each pair of returned detail records (record type 6) and associated returned addenda records (record type 7) and creates a single record on the output XML file.

The following table shows the various steps that the Flat to XML File job goes through and the messages issued at each step:

Process Steps	Messages
1. Parameter Validation	• If a parameter is invalid, a message indicating the error will be displayed in the log.
2. Create XML	• Reading Flat File ‘x’ • If no returned detail records and returned addenda record pairs are found, or if a returned detail record is missing a returned addenda record, a message will be displayed in the log.

Restartability

The job cannot be restarted. If the job fails in any of the above steps, a new job should be scheduled after correcting the errors that caused the job to fail.

Major Input

- Return ACH file (CTX, PPD, or CCD)

Job Parameters

Parameter	Description	Default Value
AMSEXPORT	Required and overrideable This is the location where the output file will be stored.	$$AMSEXPORT$$

Parameter	Description	Default Value
AMSIMPORT	Required and overrideable This is the location where the job will search for the input file.	$$AMSIMPORT$$
FLAT_FILE_NM	(Required) The name of the ACH file received from the Bank. Must be a text file with a ".txt" extension.	No default
XML_FILE_NM	Output XML File Name. Not overrideable.	ReturnedCWACH_$$@CHAINJOBID@$$.xml

Major Output

- XML file containing records that are ready to be imported to the CW EFT Unprocessed Returns table by the Load XML File job.

Job Return Code

The following table shows the possible job Return Codes for the Flat to XML File job:

Return Code	Condition
Successful (1)	The input file has been processed successfully and no error was reported.
Warning (4)	The input ACH file does not have any pairs of returned detail records (record type 6) and returned addenda records (record type 7) to process.
Non-Fatal Error (8)	N/A
Failed (12)	The job fails under the following conditions: • Parameters are invalid. • A returned detail record (record type 6) is not immediately followed by a returned addenda record (record type 7) in the input file. • Run time exceptions for unexpected situations. When this job ends with a Return Code of *Failed*, subsequent jobs in the chain are set to *Inactive*.
Terminated (16)	This return code is issued when the job is terminated by the user. When this job ends with a Return Code of *Terminated*, subsequent jobs in the chain are set to *Inactive*.
System Failure (20)	This return code is issued when the job is terminated because of database server or network issues. When this job ends with a Return Code of *System Failure*, subsequent jobs in the chain are set to *Inactive*.

Sort Criteria

N/A

Selection Criteria

N/A

Problem Resolution

If the job ends with a return code of *Failed* and above, the job can be rescheduled.

The following table shows the various steps that the Flat to XML File job goes through and the messages issued at each step:

Step 1: Parameter Validation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the parameter validations are successful.	N/A	N/A
Warning (4)	N/A	This step does not issue this return code.	N/A
Non-Fatal Error (8)	N/A	This step does not issue this return code.	N/A
Failed (12)	All of the required parameters are not entered. **Sample Message:** “Export Location is required” **Recommendation:** Enter AMSEXPORT parameter.	If the required parameters are not entered, enter the required parameters and reschedule the job.	N/A
	Entered Parameters are not valid. **Sample Message:** “The name of the ACH file received from the bank must be a text file with a '.txt' extension." **Recommendation:** Enter a valid FLAT_FILE_NM parameter that ends with ‘.txt’ extension.	If the parameters are invalid, enter valid values and reschedule the job.	N/A

Terminated (16)	The job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can be rescheduled.	N/A
System Failure (20)	The job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can be rescheduled.	N/A

Step 2: Create XML

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	This Return Code is issued under the following conditions: • If the input file is empty. • If there is not any detail record and addenda record pairs. **Sample Message:** "No records found" "No returned detail record (record type 6) and returned addenda record (record type 7) pair to process"	Make sure that the input file is not empty and has records in the expected format. Verify the file and run the chain again. If the file is correct, the chain can be rescheduled by disabling all jobs except the Initiate CW Payment Cancellation job.	N/A
Non-Fatal Error (8)	N/A	N/A.	N/A
Failed (12)	Input file has invalid records. **Sample Message:** "Invalid input file: returned detail record (record type 6) is not immediately followed by returned addenda record (record type 7)"	If the file is incorrect, contact the bank to get a correct file and reschedule the chain. If there is no correct file to process, the chain can be rescheduled by disabling all jobs except the Initiate CW Payment Cancellation job.	N/A
Terminated(16)	The job is terminated manually by the user.	The reason for the termination needs to be investigated before rescheduling the entire	N/A

		chain.	
System Failure(20)	The job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated before rescheduling the entire chain.	N/A

CW EFT Return Process: Load XML File

Job Name	Load XML File
Recommended Frequency	Daily as part of the nightly cycle or on demand.
Single Instance Required	Yes
Can be restarted	No
Reports Generated	No

Overview

The Load XML File job loads the XML file generated in the previous step into the CW EFT Unprocessed Returns table. The job uses the System Maintenance Utility (SMU) functionality to load the data.

The steps involved in this process are:

3. **Parameter Validation:** First the process validates the batch parameters.
4. **Load the Data into the table:** This is achieved by the SysManUtil functionality.

Restartability

This job does not support restartability. If the job fails it has to be rescheduled again.

Major Input

- XML file generated by the Flat to XML File job

Job Parameters

Parameter	Description	Default Value
ACTN_CD	Action Code: Table Import. Not overrideable.	201
AMSIMPORT	Import file location	$$AMSIMPORT$$

Parameter	Description	Default Value
COMMIT_SIZE	Commit Block Size. This is optional.	No default
FILE_NM	Input XML File Name. Not overrideable	ReturnedCWACH_$$@CHAINJOBID@$$.xml
GENERATE_STATS	Generate Statistics. Overrideable	TRUE

Major Output

- CW EFT Unprocessed Returns table

Job Codes

Since this job uses the common SysManUtil functionality, the job Return Code and the exception handling are handled by the SysManUtil functionality and there is no specific requirement for this job.

Restartability

The job cannot be restarted. If the job fails in any of the above steps, a new job should be scheduled after correcting the errors that caused the job to fail.

Sort Criteria

N/A

Selection Criteria

N/A

Problem Resolution

If the job ends with a return code of *Failed* and above, the job can be rescheduled.

The following table shows the various steps that the Load XML File process goes through and the messages issued at each step.

Step 1: Parameter Validation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the parameter validations are successful.	N/A	N/A

Warning (4)	N/A	This step does not issue this return code.	
Non-Fatal Error (8)	N/A	This step does not issue this return code.	
Failed (12)	All of the required parameters are not entered. **Sample Message:** "XML File is required" Recommendation: The XML File is required.	If the required parameters are not entered, enter the required parameters and reschedule the job.	
	Entered Parameters are not valid. **Sample Message:** "XML File is not found on the specified location" Recommendation: Import File location Object entered in the Data Object parameter must be valid.	If the parameters are invalid, enter the valid parameter and reschedule the job.	
Terminated (16)	The job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can be rescheduled.	
System Failure (20)	The job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can be rescheduled.	

Step 2: Load the Data into the table

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	This Return Code is issued under the following condition: • The input file is empty. **Sample Message:** "No data found on the input file"	Make sure that the parameter value entered is correct. If the parameter value is incorrect, then enter the correct parameter value and run the	N/A

		chain again.	
Non-Fatal Error (8)	N/A	N/A.	N/A
Failed (12)	N/A	N/A	N/A
Terminated(16)	The job is terminated manually by the user.	The reason for the termination needs to be investigated before rescheduling the entire chain.	N/A
System Failure(20)	The job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated before rescheduling the entire chain.	N/A

CW EFT Return Process: Process CW EFT Returns

Job Name	Process CW EFT Returns
Recommended Frequency	Daily as part of the nightly cycle or on demand.
Single Instance Required	Yes
Can be restarted	No
Reports Generated	Returned CW EFT Report Returned CW EFT Exception Report

Overview

The Process CW EFT Returns job initiates the cancellation for returned CW EFT payments and for non-miscellaneous vendors, marks the vendors associated with returned CW EFT payments and CW EFT reversals as ineligible for EFT at the Vendor Location level and Address level.

Steps in this process:

1. **Parameter Validation:** First the process validates the batch parameters. If the parameter validation fails, the process will end with the Return Code of *Failed*.
2. **Deletion of processed records:** The job deletes records from the Check Writer EFT Unprocessed Returns table where the Processed Flag is *Yes*.

3. **Selection of records:** The job selects all the records from the Check Writer EFT Unprocessed Returns table with the Processed Flag equal to No. If no records are found, the job will end with a Return Code of *Warning*.

4. **Process Records:** For each record selected, processing is performed to validate the input data, update CW EFT Reversal record (if applicable), insert CW EFT Return record (if applicable), write to report, update vendor EFT eligibility (if applicable) and finally marking the CW EFT Unprocessed Return record as processed. The job determines whether each selected CW EFT Unprocessed Returns record corresponds to a returned CW EFT Reversal, returned CW EFT Payment, Notification of Change, or returned Prenote as follows:

   - If the CW EFT Unprocessed Returns record Transaction Code is equal to *26* or *36* and Amount > $0, then the record corresponds to a returned CW EFT Reversal.
   - If the CW EFT Unprocessed Returns record Transaction Code is equal to *21* or *31* and Amount > $0 and Addenda Type Code = *99*, then the record corresponds to a returned CW EFT Payment.
   - If the CW EFT Unprocessed Returns record Transaction Code is equal to *21* or *31* and 26 or 36 for the same EFT Number and Bank Account combination (2 records for the same EFT Number and Bank Account combination), Amount > $0 and Addenda Type Code = *99*, then the record corresponds to a returned CW EFT Payment.
   - If the CW EFT Unprocessed Returns record does not correspond to a returned CW EFT Reversal or a returned CW EFT Payment, and Transaction Code is equal to *21* or *31* and Addenda Type Code = *98*, then the CW EFT Unprocessed Returns record corresponds to a Notification of Change.
   - If the CW EFT Unprocessed Returns record does not correspond to a returned CW EFT Reversal, returned CW EFT Payment, or Notification of Change, the CW EFT Unprocessed Returns record will be processed as a returned Prenote.

5. **Preliminary Validations**: If the CW EFT Unprocessed Returns record is a duplicate of a record that has been processed before, it will be written to the Returned CW EFT Exception Report and marked as processed. However, if the CW EFT Unprocessed Returns record is a duplicate of a record that has been processed before with the same EFT Number and Bank Account combination, but with a different transaction code then it will be written to the Returned CW EFT Exception Report but the status on CWEFTRET and CWEFTREV will still be updated. If the CW EFT Unprocessed Returns record has a Return Reason Code that is not valid on the Return Reason Codes (RETREAS) table, it will be written to the Returned CW EFT Exception Report and marked as unprocessed. Processing will then skip to the next CW EFT Unprocessed Returns record.

6. **CW EFT Reversal and CW EFT Return handling:**

   - Returned CW EFT Reversal

Use the EFT Number, Bank Account and Amount to look for the CW EFT Reversal record with the same EFT Number, Bank Amount and Net EFT Amount.

If exactly one CW EFT Reversal record is found and the Reversal Status is equal to *Reversal sent to the Bank*, the record is processed successfully. The corresponding CW EFT Reversal record's Reversal Status is set to *Reversal Returned by the Bank*. A CW EFT Return record with Return Status equal to *Reversal Failed by the Bank* is created to record this processed returned CW EFT reversal. The record is also written to the Returned CW EFT Report.

If there are two records for the same EFT Number, Bank Account and Amount in the return ACH file with different transaction codes (for example, 26 and 21). If transaction code 26 is received 1st and 21 is received 2nd, then the record will be inserted on CWEFREV with a Reversal Status of *Reversal Returned by the Bank*.

If the record is not processed successfully, the record is written to the *Returned CW EFT Exception Report*.

- Returned CW EFT Payment

Use the EFT Number, Bank Account and Amount to look for the Check Reconciliation (CHREC) record with the same Check/EFT Number, Bank Account, and Amount.

If exactly one matching CHREC record with Status equal to *Disbursed* or *Warranted* is found, and a matching CWPYMT record with CW Department equal to CHREC Transaction Dept, Check Writer File ID equal to CHREC Transaction ID, and Check/EFT Number equal to CHREC Check/EFT Number is found with the Cancellation Status equal to *N/A*, the record is processed successfully. A CW EFT Returns record with Return Status equal to *EFT Returned* is created to record this processed returned CW EFT payment. The record is also written to the Returned CW EFT Report.

If there are two records for the same EFT Number, Bank Account, and Amount in the return ACH file with different transaction codes (for example, 26 and 21). If transaction code 26 is received 1st and 21 is received 2nd, then after performing the above match on CHREC, CWPYMT, and so forth, the record will be inserted on CWEFRET with a Return Status of *Reversal Failed by the Bank*.

If the record is not processed successfully, the record is written to the Returned CW EFT Exception Report. In addition, a CW EFT Returns record with Return Status equal to *EFT Cancel Request Failed* is created to record this failed record if the matching CHREC record was found.

- Notification of Change

Use the ID Number to look for the Check Reconciliation (CHREC) or Paid Check (PDCHK) record with the same Check/EFT Number and Bank Account code combination.

If exactly one matching CHREC or PDCHK record is found, and a matching CWPYMT record with CW Department equal to CHREC/PDCHK Transaction Dept, Check Writer File ID equal to CHREC/PDCHK Transaction ID, and Check/EFT Number equal to CHREC/PDCHK Check/EFT Number is found, the record is processed successfully. A CW EFT Returns record with Return Status equal to *Bank Notified a Change* is created to record this processed Notification of Change. The record is also written to the Returned CW EFT Report.

If the record is not processed successfully, the record is written to the Returned CW EFT Exception Report. In addition, a CW EFT Returns record with Return Status equal to *EFT Cancel Request Failed* is created to record this failed record if the matching CHREC or PDCHK record was found.

7. **EFT Eligibility handling:**

- Identify the Vendor / Payee Code and Address Code. For returned reversal, identify the information from the CW EFT Reversal record. For returned payment or Notification of Change, identify the information from the CW Payment record. For returned prenote, if one matching CW Payment record where the original payment amount is $0 and the Check/EFT Number matches the ID Number on the CW EFT Unprocessed Returns record is found, identify the information from the CW Payment record found.

If the Vendor / Payee Code and Address Code cannot be determined because the corresponding CW EFT Reversal record or CW Payment record cannot be found, a message is written to the Returned CW EFT Exception Report.

- Get the Vendor/Customer (VCUST) record for the identified Vendor / Payee Code.

If the VCUST record is found and it is not a miscellaneous record, update the EFT Eligibility setup at the Vendor level. Then get the Address record with Address Type equal to Payment for the identified Address Code. If the Address record is found, update the EFT Eligibility setup at the payment address level.

If the VCUST record or the Address record is not found, a message is written to the Returned CW EFT Exception Report.

8. **CW EFT Unprocessed Returns handling:** The job marks the CW EFT Unprocessed Returns record as processed.

The following table shows the various steps that the Build XML File job goes through and the messages issued at each step:

Process Steps	Messages
1. Parameter Validation	• If a parameter is invalid, a message indicating the error will be displayed in the log.
2. Deletion of processed records	• 'n' Records were deleted from Check Writer EFT Unprocessed Returns.
3. Selection of Records	• Acquire returned records to be processed
4. Process records	• Number of records processed: 'n' • Number of EFT Reversal records: 'n' • Number of EFT Reversal records written to Exception: 'n' • Number of Returned EFT Payment without Notification: 'n' • Number of Returned EFT Payment without Notification written to Exception: 'n' • Number of Notification of Change: 'n' • Number of Notification of Change written to Exception: 'n' • Number of Return EFT Pre-Note records processed for Updating Returned Pre-Note: 'n' • Number of Return EFT Pre-Note records processed for Updating Returned Pre-Note written to Exception: 'n'

Restartability

The job cannot be restarted. If the job fails in any of the above steps, a new job should be scheduled after correcting the errors that caused the job to fail.

Major Input

- Check Writer EFT Unprocessed Returns table
- Check Writer EFT Reversal table
- Check Reconciliation table
- Paid Check table
- Check Writer Payment table

Job Parameters

Parameter	Description	Default Value
COMMIT_SIZE	Commit Size. Overrideable. Must be an integer greater than 0.	1000
CLIENT_NM	Client Name for Report. Overrideable and optional	No default
PROG_CTR_SZ	Progression Counter Size. Overrideable. Must be an integer greater than 0.	1000

Major Output

- Check Writer EFT Unprocessed Returns table
- Check Writer EFT Reversal table
- Check Writer EFT Returns table
- Vendor table
- Address table
- Returned CW EFT Report
- Returned CW EFT Exception Report

Job Return code

The following table shows the possible job return codes for the Process CW EFT Returns job:

Return Code	Condition
Successful (1)	All of the selected Check Writer EFT Unprocessed Returns records are processed successfully and no error was reported.
Warning (4)	No records found under the selection criteria.
Non-Fatal Error (8)	The job encountered an error and wrote the exceptions to the Returned CW EFT Exception Report. This could be because of the following reasons: • This is a duplicate record. • Reason Code does not exist on the RETREAS table • CW EFT Reversal record not found • CWEFTREV not Reversal sent to the Bank • Multiple CWEFTREV records match returned reversal

Return Code	Condition
	• Multiple CHREC records match returned EFT • CHREC Status not Disbursed or Warranted • EFT Transaction not found on CHREC • EFT Transaction not found on the CWPYMT table • CWPYMT Cancellation Status is not N/A • Multiple CHREC match returned EFT with Notification • Multiple PDCHK match returned EFT with Notification • Cannot identify Vendor/Address to update EFT Setup • Cannot find VCUST for Vendor <vendor code> • Pymt addr <address id> for Vend <vendor code> not found
Failed (12)	The job fails under the following conditions: • Parameters are invalid. • Run time exceptions for unexpected situations. When this job ends with a Return Code of *Failed*, subsequent jobs in the chain are set to *Inactive*.
Terminated (16)	This Return Code is issued when the job is terminated by the user. When this job ends with a Return Code of *Terminated*, subsequent jobs in the chain are set to *Inactive*.
System Failure (20)	This Return Code is issued when the job is terminated because of database server or network issues. When this job ends with a Return Code of *System Failure*, subsequent jobs in the chain are set to *Inactive*.

Sort Criteria

Vendor Bank Account and ID Number

Selection Criteria

Refer to the “Overview” section for this job.

Problem Resolution

If the job ends with a Return Code of *Failed* and above, the job can be rescheduled.

The following table shows various steps that the Process CW EFT Returns job goes through and the messages issued at each step:

Step 1: Parameter Validation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the parameter validations are successful.	N/A	N/A
Warning (4)	N/A	This step does not issue this return code.	N/A
Non-Fatal Error (8)	N/A	This step does not issue this return code.	N/A
Failed (12)	A required parameter is not entered. **Sample Message:** Parameter Commit Size is required	If a required parameter is not entered, enter the required parameter and reschedule the job.	N/A
	Entered Parameters are not valid. **Sample Message**: Progression Counter Size must be an integer greater than zero	If the parameters are invalid, enter the valid parameters and reschedule the job.	N/A
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can be rescheduled.	N/A
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can be rescheduled.	N/A

Step 2: Deletion of Processed Records

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the processed records are deleted successfully.	N/A	N/A
Warning (4)	N/A	This step does not issue this return code.	N/A

Non-Fatal Error (8)	N/A	This step does not issue this return code.	N/A
Failed (12)	Run time exceptions for unexpected situations.	The reason for the failure needs to be investigated. A new chain can be scheduled.	N/A
Terminated (16)	The job is terminated manually by the user.	The reason for the termination needs to be investigated. A new chain can be scheduled.	N/A
System Failure (20)	The job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. A new chain can be scheduled.	N/A

Step 3: Selection of records

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	This Return Code is issued under the following condition: • If no records are selected from CW EFT Unprocessed Returns Table. **Sample Message:** • "No records meet selection criteria."	N/A	N/A
Non-Fatal Error (8)	N/A	N/A.	N/A
Failed (12)	Run time exceptions for unexpected situations.	The reason for the failure needs to be investigated. A new job can be scheduled.	N/A
Terminated(16)	The job is terminated manually by the user.	The reason for the termination needs to be investigated before rescheduling the entire	N/A

		chain.	
System Failure(20)	The job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated before rescheduling the entire chain.	N/A

Step 4: Process records

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	N/A	N/A.	N/A
Non-Fatal Error (8)	The job encountered errors and wrote exceptions to the Returned CW EFT Exception Report.	Verify the exception report for the details. For a returned payment, if the error can be resolved so that the CW Payment cancellation can proceed, fix the error and update the corresponding CWEFTRET record with the Return Status of *EFT Returned* so that it can be processed during the next Initiate CW Payment Cancellation job.	N/A
Failed (12)	Run time exceptions for unexpected situations.	The reason for the termination needs to be investigated before rescheduling the entire chain.	N/A
Terminated(16)	The job is terminated manually by the user.	The reason for the termination needs to be investigated before rescheduling the entire chain.	N/A
System Failure(20)	The job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated before rescheduling	N/A

		the entire chain.	

CW EFT Return Process: Initiate CW Payment Cancellation

Job Name	Initiate CW Payment Cancellation
Recommended Frequency	Daily as part of the nightly cycle or on demand.
Single Instance Required	Yes
Can be restarted	No
Reports Generated	Initiated Cancellations of Returned CW EFT Payments Report Cannot Initiate Cancellations of Returned CW EFT Payments Exception Report

Overview

The Initiate CW Payment Cancellation job selects the returned CW EFT Payments from the Check Writer EFT Returns table and updates the corresponding CW Payment record's Cancellation Status to *Cancellation Requested* so that the Check Writer Cancellation chain can find these CW Payment records and create CWC transactions to cancel them.

The steps involved in this process are:

1. Selection of records: The job selects all records from the CW EFT Returns table with the Return Status set to EFT Returned or Reversal Returned by the Bank. The job will proceed to the next step only if it selects the eligible records.
2. **Initiate CW Payment Cancellation:** For each record to be processed, the job looks up the Check Reconciliation (CHREC) table to find the related record with a Status of *Disbursed* or *Warranted* that has the same Bank Account Code, EFT Number, Transaction Code of CE, Transaction Dept from CW Department, and Transaction ID from Check Writer File ID as the CWEFTRET record. If a matching CHREC record is found, the job looks up the CW Payment table to find the related record that has the same CW Department, Check Writer File ID, and EFT Number as the CWEFTRET record. If the CWPYMT record is found and has Cancellation Status equal to *N/*A, the job updates the CWEFTRET Return Status to *EFT Cancel Initiated*, updates the CWPYMT Cancellation Status to C*ancellation Requested* and Cancellation Reason to *Cancel*, and writes the record to the Initiated Cancellations of Returned CW EFT Payments Report. If the matching CHREC or CWPYMT is not found or has a different status, the job updates the CWEFTRET Return Status to *EFT Cancel Request Failed* and writes the record to the Cannot Initiate Cancellations of Returned CW EFT Payments Exception report.
3. Generate Reports: After performing all the above steps successfully, the job generates two reports:
    - The Initiated Cancellations of Returned CW EFT Payments Report lists records that were processed successfully and had their Return Status updated to *EFT Cancel Initiated*.

- The Cannot Initiate Cancellations of Returned CW EFT Payments Exception Report lists records that encountered errors and had their Return Status updated to *EFT Cancel Request Failed*.

The following table shows the various steps that the Initiate CW Payment Cancellation File job goes through and the messages issued at each step:

Process Steps	Messages
1. Selection of Records	• If the selection returns 0 records, then the following message is issued: "No records meet selection criteria."
2. Initiate CW Payment Cancellation	• Number of records processed: 'n'

Restartability

The job cannot be restarted. If the job fails in any of the above steps, a new job should be scheduled after correcting the errors that caused the job to fail.

Major input

- Check Writer EFT Returns table
- Check Writer Payment table

Job Parameters

Parameter	Description	Default Value
CLIENT_NM	Client Name for Report. Overrideable and optional.	No default
COMMIT_SIZE	Commit Block Size Optional.	100

Major Output

- Check Writer EFT Returns table
- Check Writer Payment table
- Initiated Cancellations of Returned CW EFT Payments Report
- Cannot Initiate Cancellations of Returned CW EFT Payments Exception Report

Job Return Code

The following table shows the potential job return codes for the Initiate CW Payment Cancellation job:

Return Code	Condition
Successful (1)	All of the selected Check Writer EFT Returns records are processed successfully and no error was reported.
Warning (4)	No records found under the selection criteria.
Non-Fatal Error (8)	The job encountered an error and wrote the exceptions to the Cannot Initiate Cancellations of Returned CW EFT Payments Exception Report. This could be because of the following reasons: • EFT not Disbursed/Warranted • EFT not found on CWPYMT table • CWPYMT Cancellation Status is not N/A
Failed (12)	The job fails under the following condition: • Run time exceptions for unexpected situations.
Terminated (16)	This Return Code is issued when the job is terminated by the user.

Sort Criteria

Bank Account Code, EFT Number, CW Department, and Check Writer File ID

Selection Criteria

Refer to the “Overview” section of this job for details.

Problem Resolution

If the job ends with a Return Code of *Failed* and above, the job should be rescheduled again.

The following table shows the various steps that the Initiate CW Payment Cancellation job goes through and the messages issued at each step:

Step 1: Selection of Records

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	If there are no records from the CW EFT Returns table with the Return Status equal to *EFT*	N/A	N/A

	Returned. **Sample Message:** "No records meet selection criteria."		
Non-Fatal Error (8)	N/A	N/A.	N/A
Failed (12)	N/A	N/A	N/A
Terminated(16)	The job is terminated manually by the user	The reason for the termination needs to be investigated before rescheduling the entire chain.	N/A
System Failure(20)	The job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated before rescheduling the entire chain.	N/A

Step 2: Initiate CW Payment Cancellation and Report generation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	The job is completed successfully.	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	If any record is written to the exception report.	Verify the exception report, fix the data and reschedule.	N/A
Failed (12)	The job failed due to fatal conditions.	If the job fails because of the runtime exceptions, investigate the exception reported by the process, resolve the error and reschedule.	N/A
Terminated(16)	The job is terminated manually by the user	The reason for the termination needs to be investigated before rescheduling the entire chain.	N/A

System Failure(20)	The job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated before rescheduling the entire chain.	N/A

2.2.12 CW EFT Reversal Process

Chain Name	CW EFT Reversal Process
Recommended Frequency	It is recommended this process be run daily before the Check Writer Cancellation chain during the nightly cycle.
Single Instance Required	Yes
Can be restarted	No
Reports Generated	Yes

Overview

The CW EFT Reversal Process performs two main functions: sending a reversal or reclamation request regarding a Check Writer payment to the bank, or initiating cancellation for the corresponding Check Writer payment when such a request has been processed successfully by the bank. The chain job is comprised of the following three batch jobs:

- Build XML File
- Build Flat File
- Initiate CW Payment Cancellation

These jobs can be run as a chain or alternatively, the first two jobs can be run without the third job or vice versa if only one main function is desired to be executed. This is because the first two jobs are responsible for generating the ACH files to send to the bank while the last job is responsible for initiating cancellation of Check Writer payments.

Major Input

- Check Writer EFT Reversal table
- Bank table
- Disbursement Format table
- Check Writer Header table
- Check Writer Payment table
- Check Writer ACH Addendum table

Major Output

- Check Writer EFT Reversal table
- Requested CW EFT Reversals Report
- CW EFT Reversal Exception Report
- ACH CTX file with EFT Reversals
- ACH CCD file with EFT Reversals

- ACH PPD file with EFT Reversals
- Check Writer Payment table
- Initiated Cancellations of Reversed CW EFT Payments Report
- Cannot Initiate Cancellations of Reversed CW EFT Payments Exception Report

Chain Job Return Code

The following table shows the potential Return Codes for the CW EFT Reversal job. Note that the chain will end with the highest Return Code across all of the jobs.

Return Code	Condition
Successful (1)	All of the jobs end successfully.
Warning (4)	One of the jobs in the chain ends with a Return Code of *Warning*.
Non-Fatal Error (8)	One of the jobs in the chain ends with a Return Code of *Non-Fatal Error*.
Failed (12)	One of the jobs in the chain ends with a Return Code of *Failed*.
Terminated (16)	One of the jobs in the chain ends with a Return Code of *Terminated*.
System Failure (20)	One of the jobs in the chain ends with a Return Code of *System Failure*.

Problem Resolution

If any of the jobs in the chain encounters a Non-Fatal Error, it is advisable to review and correct the errors as appropriate, and reschedule the job as needed. When a Non-Fatal Error occurs in the CW EFT Reversal chain, some records may still be processed successfully. Therefore, any generated ACH files should be allowed to follow normal processing to send to the bank and any cancellation initiated for CW payments should be allowed to proceed. It is recommended to configure the chain job Pre Condition Return Code as follows to allow the successfully processed records to complete processing.

Job Name	Pre Condition Return Code
Build XML File	Successful
Build Flat File	Non-Fatal Error
Initiate CW Payment Cancellation	Warning

Please refer to the individual jobs for details regarding the specific job processes and problem resolution.

CW EFT Reversal Process: Build XML File

Job Name	Build XML File
Recommended Frequency	Daily as part of the nightly cycle or On Demand.
Single Instance Required	Yes
Can be restarted?	No
Reports Generated	• CW EFT Reversal Exception Report • Requested CW EFT Reversals Report

Overview

The Build XML File job processes selected Check Writer EFT Reversal (CWEFTREV) records that have a Reversal Status of *Request* Reversal and generates one XML file containing the reversal details needed by the next job in order to create separate formatted ACH Files in PPD+, CCD+, and CTX format.

The steps involved in this process are:

1. Parameter Validation: First the process validates the batch parameters.

2. Selection of Records: Once the parameter validation is successful, the job selects all of the records from the Check Writer EFT Reversal table satisfying all of the following criteria:
   - If Department Code parameter is specified and CW Department is equal to the parameter value, and
   - If Check Writer File ID parameter is specified, and Check Writer File ID is equal to the parameter value, and
   - If Unit parameter is specified, and CW Unit is equal to the parameter value, and
   - Records with the Active flag set to *Yes*, and
   - Reversal Status is equal to *Request Reversal*

3. Processing the records and creating the Reversal XML File:

   For the selected Check Writer EFT Reversal records, the job will create the XML file containing the reversal details needed by the next job in order to create separate formatted ACH Files in PPD+, CCD+, and CTX format to be sent to the bank.

4. **Table updates:**
   - If the selected Check Writer EFT Reversal record is processed successfully, the following updates are applied to the record:

     Update CW EFT Reversal table:

     UPDATE R_CW_EFT_RVRSL

     SET ACT_FL=”False”;

UPDATE R_CW_EFT_RVRSL

SET RVRSL_STA = “Reversal sent to the Bank”

UPDATE R_CW_EFT_RVRSL

SET ACH_REVRS_CREAT_DT = “Application date”

- If the selected Check Writer EFT Reversal record encounters an exception that is not due to Bank or Company ID issue, the following updates are applied to the record:

  Update CW EFT Reversal table:

  UPDATE R_CW_EFT_RVRSL

  SET

  RVRSL_STA = “EFT Cancel Request failed”

  CONFRM_CMNT = “CW EFT Reversal <chain ID> Build XML File”

  SYS_MSG = <applicable exception message>

- If the selected Check Writer EFT Reversal record encounters an exception due to Bank or Company ID issue, the following updates are applied to the record:

  Update CW EFT Reversal table:

  UPDATE R_CW_EFT_RVRSL

  SET

  CONFRM_CMNT = “CW EFT Reversal <chain ID> Build XML File”

  SYS_MSG = <applicable exception message>

5. **Generate Reports:** Successfully processed records are shown on the Requested CW EFT Reversals Report. Records with exceptions are shown on the CW EFT Reversal Exception Report.

The following table shows the various steps that the Build XML File job goes through and the messages issued at each step:

Process Steps	Messages
1. Parameter Validation	• If a parameter is invalid, a message indicating the error is displayed in the log.
2. Selection of Records	• If the selection returns 0 records, then the following message is issued: “No records meet selection criteria.”
3. Processing the records and creating the Reversal XML File	• Creating File Header record for ‘x’ • Records with EFT Format ‘x’ found. • Creating Batch Header record for CW File ID: ‘x’ • All ‘n’ payment records processed successfully for CW File ID: ‘x’

Process Steps	Messages
	• Creating File Control record for 'x'

Restartability

The job cannot be restarted. If the job fails in any of the above steps, a new job should be scheduled after correcting the errors that caused the job to fail.

Major Input

- Check Writer EFT Reversal table
- Bank table
- Disbursement Format table
- Check Writer Header table
- Check Writer Payment table
- Check Writer ACH Addendum table

Batch Parameters

Parameter	Description	Default Value
AMSEXPORT	Required. ACH XML File Location.	$$AMSEXPORT$$
AMSPARM	Required. Parameter File Location	$$AMSPARM$$
CLIENT_NM	Optional. Client Name.	No Default
CW_DEPT_CD	Conditionally Required. Department Code. Required if Unit Code is entered	No Default
CW_FILE_ID	Optional. Check Writer File ID. Only one value can be entered. The value in combination with the Department and Unit parameter value(s) will be verified against the CWHDR table.	No Default
CW_UNIT_CD	Optional. Unit	No Default
PREFIX	Required. ACH XML File Prefix	CWACHREV

COMP_ENTR_DESC	Required. Company Entry Description. The entered value will be populated on the ACH Batch Header record for reversals.	REVERSAL
COMP_ENTR_DESC_RECLAIM	Required. Company Entry Description for Reclamations. The entered value will be populated on the ACH Batch Header record for reclamations.	RECLAIM
LAG_DY	Required. Lag days. Must be a positive numeric value. Represents the number of business days to be added to the Application Date to calculate the Effective Entry Date.	No Default

Major Output

- XML file containing information to be used by the Build Flat File job
- Parameter file (CWACHParam_<chain ID>.txt) to be used by the Build Flat File job
- Check Writer EFT Reversal table
- Requested CW EFT Reversals Report
- CW EFT Reversal Exception Report

Job Return Code

The following table shows the possible job Return Codes for the Build XML File job:

Return Code	Condition
Successful (1)	All of the selected Check Writer EFT Reversal records are processed successfully and no error was reported.
Warning (4)	No records found under the selection criteria.
Non-Fatal Error (8)	The job encountered an error and wrote the exceptions to the CW EFT Reversal Exception Report. This could be because of the following reasons: • CWHDR not found • File ID <CWHDR Check Writer File ID> is skipped because the Bank record cannot be found. • File ID <CWHDR Check Writer File ID> is skipped because the Company ID cannot be determined from the bank or disbursement format. • CWPYMT not found • CWPYMT Cancellation Status not equal to N/A
Failed (12)	The job fails under the following conditions: • Parameters are invalid. • Run time exceptions for unexpected situations. When this job ends with a Return Code of *Failed*, subsequent jobs in the chain are set to *Inactive*.
Terminated (16)	This Return Code is issued when the job is terminated by the user. When this job ends with a Return Code of *Terminated*, subsequent jobs in the chain are set to *Inactive*.
System Failure (20)	This Return Code is issued when the job is terminated because of database server or network issues. When this job ends with a Return Code of *System Failure*, subsequent jobs in the chain are set to *Inactive*.

Sort Criteria

Disbursement Format, Bank Account Code, CW File ID, CW Department Code, Reclamation flag, and Check/EFT number

Selection Criteria

Refer to the “Overview” section of this job for details.

Problem Resolution

If the job ends with a Return Code of *Failed* and above, the job can be re-scheduled.

The following table shows the various steps that the Build XML File job goes through and the messages issued at each step:

Step 1: Parameter Validation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the parameter validations are successful.	N/A	N/A
Warning (4)	N/A	This step does not issue this return code.	N/A
Non-Fatal Error (8)	N/A	This step does not issue this return code.	N/A
Failed (12)	A required parameter is not entered. **Sample Message:** Department Code Required when Unit is entered. **Recommendation:** Enter the Department Code parameter if Unit Code parameter is entered.	If a required parameter is not entered, enter the required parameter and reschedule the job.	N/A
	Entered Parameters are not valid. **Sample Message**: “CW File ID <<CW_FILE_ID>> does not exist.” **Recommendation:** Enter a valid CW_FILE_ID parameter.	If the parameters are invalid, enter the valid parameters and reschedule the job.	N/A

Terminated (16)	The job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can be rescheduled.	N/A
System Failure (20)	The job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can be rescheduled.	N/A

Step 2: Selection of Records

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	This Return Code is issued under the following condition: • If no eligible records are found for processing. **Sample Message:** "No records meet selection criteria."	Make sure that the parameter value entered is correct. If the parameter value is incorrect, then enter the correct parameter value and run the chain again.	N/A
Non-Fatal Error (8)	N/A	N/A.	N/A
Failed (12)	The job failed because of runtime exceptions for an unexpected situation.	This step is performed only if the parameter validation is successful. The job can fail with fatal conditions only on encountering unknown exceptions. If that happens, investigate the exception reported by the process, resolve the error and reschedule the entire chain.	N/A
Terminated(16)	The job is terminated manually by the user.	The reason for the termination needs to be investigated before rescheduling the entire chain.	N/A
System Failure(20)	The job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated before rescheduling the entire	N/A

		chain.	

Step 3: Create XML file and Update Check Writer EFT Reversal table

For the selected Check Writer EFT Reversal records, the job will create the XML file containing the reversal details needed by the next job in order to create separate formatted ACH Files in PPD+, CCD+, and CTX format to be sent to the bank. After generating the ACH file for the selected records, the process will update the CW EFT Reversal table.

Possible Return Codes	Condition	Recommendation	Other instructions
Successful (1)	The system successfully creates and updates the CW EFT Reversal table.	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	This Return Code is issued under the following condition: The job encountered an error and wrote the exceptions to the CW EFT Reversal Exception Report.	Verify the Exception report for the details. Issues due to Bank or Company ID need to be fixed before re-running the job. For other exceptions, if the reversal is still required, it must be requested manually through the bank outside of the application.	N/A
Failed (12)	The job failed because of runtime exceptions for an unexpected situation.	The reason for the failure needs to be investigated before rescheduling the job.	N/A
Terminated (16)	The job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can be rescheduled.	N/A
System Failure (20)	The job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can be rescheduled.	N/A

CW EFT Reversal Process: Build Flat File

Job Name	Build Flat File
Recommended Frequency	Daily as part of the nightly cycle or On Demand.
Single Instance Required	Yes

Can be restarted?	No
Reports Generated	No

Overview

The Build Flat File job generates the Reversal ACH Files for the supported ACH formats of PPD+, CCD+, and CTX. Separate formatted ACH Files will be generated for each of the valid formats.

Restartability

This job does not support restartability. If the job fails it has to be rescheduled again.

Major Input

- XML file generated by the Build XML File job
- Parameter file generated by the Build XML File job

Batch Parameters

Parameter	Description	Default Value
AMSEXPORT	ACH Flat File Location	$$AMSEXPORT$$
AMSPARM	Parameter File Location	$$AMSPARM$$

Major Output

Three Flat Files are generated based on the ACH format the job encounters.

- ACH CTX File with EFT Reversals
- ACH CCD File with EFT Reversals
- ACH PPD File with EFT Reversals

The output flat files are named according to the following format:

<prefix>_<disbursement format>_<chain id>.dat

Job Return Code

The following table shows the potential job Return Codes for the Build Flat File job:

Return Code	Condition
Successful (1)	The input XML file was processed successfully into output flat files and no error was reported.
Warning (4)	N/A

Non-Fatal Error (8)	N/A
Failed (12)	The job fails under the following conditions: • Parameters are invalid. • Run time exceptions for unexpected situations. When this job ends with a Return Code of *Failed*, subsequent jobs in the chain are set to *Inactive*.
Terminated (16)	This return code is issued when the job is terminated by the user. When this job ends with a Return Code of *Terminated*, subsequent jobs in the chain are set to *Inactive*.
System Failure (20)	This return code is issued when the job is terminated because of database server or network issues. When this job ends with a Return Code of *System Failure*, subsequent jobs in the chain are set to *Inactive*.

Sort Criteria

N/A

Selection Criteria

N/A

Problem Resolution

If this job fails, then this job has to be rescheduled. All cases of failure have to be investigated before rescheduling the job. In case of Parameter validation failures, provide valid parameters while rescheduling the job.

The following table shows the various steps that the Build Flat File job goes through and the messages issued at each step:

Step 1: Parameter Validation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the parameters are valid.	N/A	N/A
Warning (4)	N/A	This step does not issue this return code.	
Non-Fatal Error (8)	N/A	This step does not issue this return code.	

Failed (12)	A required parameter is not entered.	If a required parameter is not entered, enter the required parameter and reschedule the job.	
	Entered Parameters are not valid.	If the parameters are invalid, enter the valid parameters and reschedule the job.	
Terminated (16)	The job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can either be rescheduled or a new job can be scheduled.	
System Failure (20)	The job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can either be rescheduled or a new job can be scheduled.	

Step 2: Flat File Creation

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	N/A	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	The job failed because of runtime exceptions for an unexpected situation.	The reason for the failure needs to be investigated before rescheduling the job.	N/A
Terminated (16)	The job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can either be rescheduled or a new job can be scheduled.	N/A
System Failure (20)	The job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can either be rescheduled or a new job can be scheduled.	N/A

CW EFT Reversal: Initiate CW Payment Cancellation

Job Name	Initiate CW Payment Cancellation
Recommended Frequency	Daily as part of the nightly cycle or on demand.
Single Instance Required	Yes
Can be restarted?	No
Reports Generated	• Cannot Initiate Cancellations of Reversed CW EFT Payments Exception Report • Initiated Cancellations of Reversed CW EFT Payments Report

Overview

The Initiate CW Disbursement Cancellation job selects the successfully reversed or reclaimed CW EFT Payments from the CW EFT Reversal table and updates the corresponding CW Payment record's Cancellation Status to *Cancellation Requested* so that the Check Writer Cancellation chain can find these CW Payment records and create CWC transactions to cancel them.

The Steps involved in this process are:

1. **Selection of Records**: The job selects the eligible records from the Check Writer EFT Reversal table with Reversal Status set to Reversal Confirmed.
2. Processing Initiate CW Payment Cancellation:

   The Initiate CW Payment cancellation job updates the CW EFT Reversal table and the CW Payment table.

   If a matching record is found on the Check Reconciliation (CHREC) table with the Status of *Disbursed* or *Warranted* and a matching record is also found on CW Payment (CWPYMT) table with Cancellation Status = N/A, then the following updates are done on the CW EFT Reversal (CWEFTREV) table and the CW Payment (CWPYMT) table:

   CW Payment Table:

   UPDATE R_AP_CW_PYMT

   SET CAN_STA = "Cancellation Requested",

   CAN_REAS_CD = "Cancel"

   CW EFT Reversal table:

   UPDATE R_CW_EFT_RVRSL

   SET RVRSL_STA = "EFT Cancellation Initiated",

   EFT_CAN_INIT_DT = Application Date,

   SYS_MSG = "EFT Cancel Initiated by the CW EFT Reversal Process"
3. **Generate Reports:** After performing the above steps, the job generates two reports:

- Successfully processed records are displayed on the Initiated Cancellations of Reversed CW EFT Payments Report
- Records with exceptions are displayed on the Cannot Initiate Cancellations of Reversed CW EFT Payments Exception Report.

The following table shows the various steps that the Initiate CW Payment Cancellation File job goes through and the messages issued at each step:

Process Steps	Messages
1. Selection of Records	• If the selection returns 0 records, then the following message is issued: “No records meet selection criteria.”
2. Processing Initiate CW Payment Cancellation	• Number of records processed: ‘n’

Restartability

The job cannot be restarted. If the job fails in any of the above steps, a new job should be scheduled after correcting the errors that caused the job to fail.

Major Input

- Check Writer EFT Reversal table
- Check Writer Payment table

Batch Parameters

Parameter	Description	Default Value
CLIENT_NM	Client Name for Report. Optional.	No Default
COMMIT_SIZE	Commit Block Size Optional.	100

Major Output

- Check Writer EFT Reversal table
- Check Writer Payment table
- Initiated Cancellations of Reversed CW EFT Payments Report
- Cannot Initiate Cancellations of Reversed CW EFT Payments Exception Report

Job Return Code

The following table shows the potential job return codes for the Initiate CW Payment Cancellation job:

Return Code	Condition
Successful (1)	All of the selected Check Writer EFT Reversal records are processed successfully and no error was reported.
Warning (4)	No records found under the selection criteria.
Non-Fatal Error (8)	The job encountered an error and wrote the exceptions to the Cannot Initiate Cancellations of Reversed CW EFT Payments Exception Report. This could be because of the following reasons: • Check Reconciliation Status is not *Disbursed* or *Warranted* • EFT Transaction is not found on CHREC table • EFT Transaction is not found on Check Writer Payment table • Check Writer Payment Cancellation Status is not N/A.
Failed (12)	The job fails under the following condition: • Run time exceptions for unexpected situations.
Terminated (16)	This Return Code is issued when the job is terminated by the user.
System Failure (20)	This Return Code is issued when the job is terminated because of database server or network issues.

Sort Criteria

Bank Account Code, EFT Number, CW Department, and Check Writer File ID.

Selection Criteria

Refer to the “Overview” section of this job for details.

Problem Resolution

If the job ends with a Return Code of *Failed* and above, the job should be rescheduled.

The following table shows the various steps that the Initiate CW Payment Cancellation Process goes through and the messages issued at each step:

Step 1: Selection of records

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	This Return Code is issued under the following condition: If there are no records found on the CW EFT Reversal Table with the status *Reversal Confirmed*. **Sample Message:** "No records meet selection criteria".	N/A	N/A
Non-Fatal Error (8)	N/A	N/A.	N/A
Failed (12)	The job failed because of runtime exceptions for an unexpected situation.	This step is performed only if the parameter validation is successful. It can fail with fatal conditions only on encountering unknown exceptions. If that happens, investigate the exception reported by the process, resolve the error and reschedule the entire chain.	N/A
Terminated(16)	The job is terminated manually by the user.	The reason for the termination needs to be investigated before rescheduling the entire chain.	N/A
System Failure(20)	The job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated before rescheduling the entire chain.	N/A

Step 2: Processing of records

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	This Return Code is issued under the following condition: • If there is any record for which the CW payment cancellation could not be initiated. **Sample Message:** "EFT Transaction not found on CHREC table."	Verify the exception report for the details. Fix the setup accordingly and reschedule the entire chain. If the exception cannot be resolved, the user may delete the CWEFTREV record manually or set the Reversal Status to *EFT Cancel Request failed* by temporarily downgrading the corresponding error(s).	N/A
Failed (12)	The job failed because of runtime exceptions for an unexpected situation.	This step is performed only if the parameter validation is successful. The job can fail with fatal conditions only on encountering unknown exceptions. If that happens, investigate the exception reported by the process, resolve the error and reschedule the entire chain.	N/A
Terminated(16)	The job is terminated manually by the user.	The reason for the termination needs to be investigated before rescheduling the entire chain.	N/A
System Failure(20)	The job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated before rescheduling the entire chain.	N/A

2.2.13 CW Intercept Transaction Generation

Job Name	CW Intercept Transaction Generation
Recommended Frequency	On Demand. The CW Intercept Transaction Generation chain should be run after the CW Accounting Transaction Generation chain job.
Single Instance Required	No
Can the job be restarted	No
Exception reports generated	No

Overview

The CW Intercept Transfer (CWI) Transaction is a clone of the Internal Exchange Transaction (IET) transaction. The CWI uses the same user interface as the IT Transaction Code. The CWI transaction has a Transaction Sub-Type of 'CW' and is used to post accounting lines to the respective Advantage journals for Check Writer payments that have been intercepted. The CW Intercept Transaction Generation Chain creates a single CW Intercept Transfer transaction for each payment that was intercepted within the selected Department/CW File ID. Once a CWI transaction has been submitted to Final, it cannot be cancelled or modified. Correcting or modifying the accounting effects of a CW Intercept Transfer transaction must be performed using Advantage journal voucher transactions.

The following is a list of the jobs steps that make up the CW Intercept Transaction Generation Chain followed by a more in-depth description of the jobs:

1. Intercept Transaction XML Generation
2. Upload
3. Submit

Major Input

- Check Writer Header (CWHDR) table
- Check Writer Payment (CWPYMT) table
- Check Writer Accounting (CWACTG) table
- Check Writer Vendor Intercept (CWVINCT) table

Major Output

- XML file that will be uniquely named by the combination of CWI Transaction Code and the job's Chain ID.
- Upload parameter file (default name will be UploadCWIParm.txt) that will be used as input to the second job of the CW Intercept Transaction Generation chain,

- Submit parameter file (default name will be SubmitCWIParm.txt) that will be used as input to the third job of the CW Intercept Transaction Generation chain.
- Updates to the Check Writer Header (CWHDR) table.

Chain Return Code

For chain jobs, use the following table to determine the potential return codes with which the chain can end:

Return Code	Condition
Successful (1)	All of the jobs end successfully.
Warning (4)	One of the jobs in the chain ends with a return code of "Warning."
Non-Fatal Error (8)	One of the jobs in the chain ends with a return code of "Non-Fatal Error."
Failed (12)	One of the jobs in the chain ends with a return code of "Failed."
Terminated (16)	One of the jobs in the chain ends with a return code of "Terminated."
System Failure (20)	One of the jobs in the chain ends with a return code of "System Failure."

Problem Resolution

Please refer to the individual job "Problem Resolution" section for more details.

CW Intercept Chain: Intercept Transaction XML Generation Job

Job Name	Intercept Transaction XML Generation
Recommended Frequency	On Demand or as part of the nightly cycle. In the nightly cycle, this should be run after running the CWA Transaction Generation Chain.
Single Instance Required	No
Can the job be restarted	No
Exception reports generated	No

Overview

The Intercept Transaction XML Generation job step selects intercepted payment records from the CW Vendor Intercept page based on user entered job parameters.

On execution of the CW Intercept Transaction Generation chain, the Intercept Transaction XML Generation job step performs standard edits on the parameters entered. Each of the job parameters is validated. If any parameter fails in the validation, the job is terminated with a Return Code of Failed and subsequent job steps within the CW Intercept Transaction Generation Chain will not be executed.

Restartability Information

This job cannot be restarted. If the job fails for some reason, then the reason for failure should be verified and a new chain should be scheduled.

Major Input

- Check Writer Header (CWHDR) table
- Check Writer Payment (CWPYMT) table
- Check Writer Accounting (CWACTG) table
- Check Writer Vendor Intercept (CWVINCT) table

Batch Parameters

Parameter	Description	Default Value
File location of parameter files. AMSPARM	Parameter Directory location (** **Refer to Note:** Assumptions for SWBP on page no. 7)	$$AMSPARM$$
Export Directory AMSEXPORT	Export Directory location (** **Refer to Note:** Assumptions for SWBP on page no. 7)	$$AMSEXPORT$$
Apply Override APPLY_OV	Apply override on transaction import	<blank>
Commit block size COMMIT_BLOCK	Commit Block Size	<blank>
CW Department Code CW_DEPT_CD	CW Department Code	<blank>
CW File ID CW_FILE_ID	CW File ID(s) to be processed. Multiple files must be comma delimited.	<blank>
CW Unit Code CW_UNIT_CD	CW Unit Code	<blank>
CWI Transaction Code	CW Intercept Transfer Transaction Code	CWI

DOC_CD		
CWI Transaction Department Code DOC_DEPT_CD	CW Intercept Transfer Transaction Department Code	<blank>
Transaction Prefix DOC_PREFIX	CW Intercept Transfer Transaction Prefix	AUTO
CWI Transaction Unit Code DOC_UNIT_CD	CW Intercept Transfer Transaction Unit	<blank>
Override Level OVERRIDE_LEVEL	Override level to be used in transaction submission during the Submit job step.	<blank>
Parameter file 1 PARM_FILE_1	File name of upload parameter text file.	UploadCWIParm.txt
Parameter file 2 PARM_FILE_2	File name of submit parameter text file.	SubmitCWIParm.txt
Progression Message Block Size PROG_CTR_SZ	Required. This field provides the information about the record being processed.	No Default
User ID USER_ID	User ID to be used in transaction submission during the Submit job step.	<blank>

Major Output

- XML file that is uniquely named by the combination of CWI Transaction Code and the job's Chain ID.
- Upload parameter file (default name is UploadCWIParm.txt) that is used as input to the second job of the CW Intercept Transaction Generation chain.
- Submit parameter file (default name is SubmitCWIParm.txt) that is used as input to the third job of the CW Intercept Transaction Generation chain.
- Updates to the Check Writer Header (CWHDR) table.

Job Return Code

Return Code	Condition
Successful (1)	All of the validations are performed successfully.

Warning (4)	• No CW records meet the selection criteria. • All of the CW File ID's specified are valid but not all were selected for processing. • When any of the files failed to process when multiple file ID's are entered.
Non-Fatal Error (8)	This job does not issue this return code.
Failed (12)	The job fails under the following conditions: • Parameters are invalid • Technical failure When this job ends with a Return Code of Failed, subsequent jobs in the chain are set to Inactive.
Terminated (16)	This Return Code is issued when the job is terminated by the user. When this job ends with a Return Code of Terminated, subsequent jobs in the chain are set to inactive.
System Failure (20)	This Return Code is issued when the job is terminated because of database server or network issues. When this job ends with a Return Code of System Failure, subsequent jobs in the chain are set to Inactive.

Sort Sequence

- For Header Records → Department, Check Writer File ID.
- For Payment Records → Line Number

Selection Criteria for CWHDR records:

Records are selected for processing based on the following criteria:

- [(Department (CWHDR) = Department (Parameter)) or if Department (Parameter) = blank, select all CWHDR records], AND
- [(CW File ID (CWHDR) = CW File ID (Parameter)) or if CW File ID (Parameter) = blank, select all CWHDR records], AND
- [(Unit (CWHDR) = Unit (Parameter)) or if Unit (CWHDR) = blank, select all CWHDR records], AND
- CW Intercept Status (CWHDR) = 'Pending generation' or 'Generated', AND
- Intercept Amount (CWPYMT) > 0, AND
- CWI Transaction Code (CWPYMT) = Blank, AND
- CWA Transaction Final (CWHDR) = 'Yes'.

For each Department/CW File ID passing the selection criteria, the Intercept Transaction XML Generation process will begin generating a separate CWI transaction for each payment associated with the CWHDR record.

Problem Resolution

If the job fails for any reason, then a chain should be scheduled after correcting the errors. Since no intermediate commit Transaction occurs in this job, there is no need to restore any database updates. All of the updates are rolled back if this step is not successful.

The following table shows the possible return codes and recommendations for each processing step.

Step 1: Batch Parameter Validation:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All batch parameter validations are successful (that is, there are no errors raised during validations.)	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Required parameters are not entered. **Sample Message**: CWI Transaction Code cannot be blank	Enter the missing required parameters and schedule a new job. Enter the CWI Transaction Code	N/A
	Entered Parameters are not valid. **Sample Message:** CW Intercept Transfer Transaction Code does not exists on the Transaction Control table	Enter valid parameters and schedule a new job. Enter a valid CW Intercept Transfer Transaction Code and schedule a new job	N/A

	Sample Message: CW Intercept Transfer Transaction Code is not a valid IT Transaction Type/ Transaction Sub-Type of IET/CW.	Enter a valid CW Intercept Transfer Transaction Code and schedule a new job	N/A
	Sample Message: AMSPARM Directory does not exist on the server.	Enter a valid AMSPARM Directory and schedule a new job	N/A
	Failed because of runtime exceptions caused by unexpected conditions.	Failure reason needs to be investigated before rescheduling the job.	N/A
Terminated (16)	Job is terminated manually by the user.	The reason for the Termination needs to be investigated. A new job can be rescheduled.	N/A
System Failure (20)	Job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. A new job can be rescheduled.	N/A

Step 2: CWI Transaction XML Generation Processing:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	CWI Transaction XML generated successfully.	N/A	N/A
Warning (4)	All of the CW File IDs specified are valid but not all were selected.	Enter the valid CW File IDs that meet the selection criteria.	N/A
	No records on CW Header meet selection criteria.	Enter at least one valid CW File ID that meets the selection criteria.	N/A
Non-Fatal Error (8)	N/A	N/A	N/A

Failed (12)	Failed because of runtime exceptions caused by unexpected conditions.	Failure reason needs to be investigated before rescheduling the job.	N/A
Terminated (16)	Job is terminated manually by the user.	The reason for the Termination needs to be investigated. A new job can be scheduled.	N/A
System Failure (20)	Job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. A new job can be scheduled.	N/A

Check Writer Intercept Chain: Upload Job

Job Name	Upload
Recommended Frequency	On Demand
Single Instance Required	No
Can the job be restarted	No
Exception reports generated	No

Overview

This job step uses SysManUtility (SMU) to load the CWI transactions into the transactions using the xml file created by the Intercept Transaction XML Generation job. The CWI parameter text file generated in the Intercept Transaction XML Generation job contains the SMU transaction import parameters and the CWI Transaction XML File IDs.

On a successful insert of the CWI transaction into the Transaction Catalog, the system shall set the CWI Transaction identifier fields (Transaction Code, Transaction Department, Transaction Unit and Transaction ID) on the CWPYMT record equal to the corresponding fields on the CW Intercept Transfer transaction. If all other CWPYMT records have a non-blank CWI Transaction Code value, the system shall set the CW Intercept Status equal to ‘Generated’.

The following fields are updated on the selected CW Payment record:

- CWI Transaction Code = Transaction Code (CW Intercept Transfer)
- CWI Transaction Department = Transaction Department (CW Intercept Transfer)
- CWI Transaction Unit = Transaction Unit (CW Intercept Transfer)
- CWI Transaction ID = Transaction ID (CW Intercept Transfer)

Restartability Information

SysManUtil rolls back the data if it fails; therefore, there is no need to back out the data.

Major Input

- Upload parameter file created during the first step of the chain.

Batch Parameters

Parameter	Description	Default Value
Parameter file name PARM_FILE	The file location and name of the upload parameter file.	$$AMSPARM$$/UploadCWIParm.txt

Major Output

- CWI transaction will be uploaded to the Transaction Catalog.
- CW Payment Line records get updated when the job loads the transactions successfully into the Transaction Catalog.
- The CWI Generation Status on the CW header table gets updated.

Job Return Code

The following table shows the potential job return codes for the SMU Transaction Upload job.

Return Code	Condition
Successful (1)	All of the records are loaded into the Transaction Catalog successfully.
Warning (4)	This Return Code is issued under the following conditions: • The input file is empty • Some records fail to load into the Transaction Catalog.
Non-Fatal Error (8)	All of the records fail to load into the Transaction Catalog.
Failed (12)	The job will fail under the following conditions: • Parameters are invalid • The input file is not found in the specified directory • If the error messages exceed the maximum number specified in the parameter • Runtime exceptions encountered for any unexpected situations When the job ends with a Return Code of Failed, subsequent jobs in the chain are set to Inactive.
Terminated (16)	This Return Code is issued when the job is terminated by the user. When the job ends with a Return Code of Terminated, subsequent

	jobs in the chain are set to Inactive.
System Failure (20)	This Return Code is issued when the job is terminated because of database server or network issues. When this job ends with a Return Code of System Failure, subsequent jobs in the chain are set to Inactive.

Sort Sequence

N/A

Selection Criteria

N/A

Problem Resolution

If the job fails to load some or all of the transactions from the XML file into the Transaction Catalog, then new instance of the chain should be executed with the same input parameters after correcting the issues. Before running the new chain, updates made to the CW Header table should be backed out.

If the job failed after loading a few transactions in the Transaction Catalog and those transactions were already committed, then before scheduling the new job those transactions should be discarded manually. The transaction can be identified by the CW File ID since the Transaction ID is equivalent to the CW File ID. Otherwise, any new chain for the CW File ID will be rejected when the job tries to load it into the Transaction Catalog.

The following table shows the possible return codes and recommendations for each processing step.

Upload the CWI transaction:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All parameters are valid and all of the transactions are loaded successfully.	N/A	N/A
Warning (4)	The input file is empty. **Sample message:** No records are found on the input file.	Make sure that the Intercept Transaction XML Generation job created the input file with records.	N/A

	This return code is issued when the job fails to load some of the transactions. **Sample Message:** Unable to load some of the transactions into the Transaction Catalog.	Analyze the reason for the failure, resolve the issue and load all of the rejected transactions.	N/A
Non-Fatal Error (8)	This return code is issued when the job failed to load all of the transactions.	Analyze the reason for the failure, resolve the issue and load all of the rejected transactions.	N/A
Failed (12)	This return code is issued when the parameters are not valid. **Sample Message:** Parameter file could not be located/read.	Make sure that the parameter file exits in the specified folder and schedule a new chain job.	N/A
	Failed because of runtime exceptions for unexpected conditions.	The reason for the failure needs to be investigated before rescheduling the job.	N/A
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated before rescheduling the job.	N/A
System Failure (20)	Job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. A new job can be scheduled.	N/A

Check Writer Intercept Chain: Submit Job

Job Name	Submit
Recommended Frequency	On Demand

Single Instance Required	No
Can the job be restarted	No
Exception reports generated	No

Overview

This job step will execute a SysManUtil Submit to submit the CWI transaction created by the Intercept Transaction XML Generation job. A submit CWI parameter text file created in the Intercept Transaction XML Generation process, specifies which transactions should be submitted.

Restartability information:

This job cannot be re-started. The transactions can be manually submitted. The transaction ID can be found on the processing log. If the transaction cannot be processed to final, then it should be discarded manually so that any instance of this chain with the rejected transaction's CW file ID will pick up that record for processing.

Major Input

- SMU job parameter file
- Draft Transactions in the Transaction Catalog

Batch Parameters

Parameter	Description	Default Value
Parameter File PARM_FILE	File location and name of submit parameter file.	$$AMSPARM$$/SubmitCWIParm.txt

Major Output

- The draft CWI transactions are submitted.

Restartability information:

SysManUtil rolls back the data if it fails; therefore, there is no need to back out the data.

Job Return Code

The following table shows the potential job return codes for the SMU Transaction Submit job.

Return Code	Condition
Successful (1)	All of the transactions submitted successfully.

Warning (4)	N/A
Non-Fatal Error (8)	N/A
Failed (12)	The job will fail under the following conditions: • Parameters are invalid • Parameter file is not found • Runtime exceptions encountered for any unexpected situations.
Terminated (16)	This return code is issued when the job is terminated by the user.
System Failure (20)	This return code is issued when the job is terminated because of database server or network issues.

Sort Sequence

N/A

Selection Criteria

N/A

Problem Resolution

If this job ends with a Return Code of Failed, Terminated or System Failure, then the loaded CWI transaction should be submitted manually. If the transaction cannot be processed to final, then the transaction should be discarded so that the CW payment records associated with the rejected transaction can be picked up by the subsequent run of this chain.

The following table shows the possible return codes and recommendations for each processing step.

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the transactions submitted successfully.	N/A	
Warning (4)	N/A	The job does not issue this return code.	
Non-Fatal Error (8)	N/A	The job does not issue this return code.	
Failed (12)	This return code is issued when the input parameter is not found in the	Make sure that the parameter file exits in the specified folder	

	specified directory. **Sample Message:** Parameter file could not be located/read.	and schedule a new job.	
	Failed because of runtime exceptions for an unexpected situation.	If the job fails with fatal conditions on encountering unknown exceptions, then investigate the exception reported by the process, resolve the error and submit the transaction manually. The transaction ID can be found on the process logs. If the issue cannot be resolved, then a new chain should be scheduled after discarding the draft transaction from the Transaction Catalog.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. If the issue is resolved then the loaded transaction can be submitted manually. The Transaction ID can be found on the Transaction Catalog. If the issue cannot be resolved, then a new chain should be scheduled after discarding the Draft transaction from the Transaction Catalog.	
System Failure (20)	Job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. If the issue is resolved then the loaded transaction can be submitted manually. The Transaction ID can be found on the Transaction Catalog. If the issue cannot be resolved, then a new chain should be scheduled after discarding the draft transaction from the Transaction Catalog.	

2.2.14 CW Pre-Edit

Job Name	CW Pre-Edit
Recommended Frequency	On Demand. The CW Pre-Edit chain job should be run after the CW Table Load chain job and before the CW Accounting Transaction Generation chain job.
Single Instance Required	No
Can be restarted	No
Reports Generated	Check Writer Pre-Edit Accounting Report Check Writer Pre-Edit Non-Accounting Report

Overview

The CW Pre-Edit chain job performs the validations for all information provided in the Check Writer files and infers the data from the various tables. This chain job consists of five individual jobs as explained below.

1. **Inferences & Validations** - This is the first job of the CW Pre-Edit chain. The main purpose of this job is to infer values to the fields in the CW tables from various related tables and validate the additional fields. The fields for validation include all vendor and payment constraints as set by the CW Options table (CWOPT). Errors found during the validations will be written to the Non-Accounting Report. This report can be viewed in HTML or PDF format from the Advantage Job Manager. The Pre Condition Return Code for the Inferences & Validations job is Successful. This job creates 4 parameter files:
   - CreateCWAXML.txt
   - LoadValidateCWATrans.txt
   - CWActgExpRep.txt
   - DiscardCWATrans.txt

2. **Accounting Transaction XML Generation** - The purpose of this job is to create an XML file which will contain CWA transaction Header and Accounting component details by using the CreateCWAXML.txt file as an input. The xml file generated is used as an input to the next job in the chain. The XML file is uniquely named using the CWA Transaction Code and the job's Chain ID. The Pre Condition Return Code for the Accounting Transaction XML Generation job is Successful which indicates that this job is scheduled in situations where the Inferences & Validations job has exited with a Return Code of Successful.

3. **Load & Validate** - The purpose of this job is to Load the CWA transaction and validate the same by using the LoadValidateCWATrans.txt file created in the first job of the Pre-Edit chain. Loading and validation of the CWA transaction is done by SysManUtil. This job creates an exception file (.txt file). The errors generated during validation of the CWA

transaction are logged in this file. This file is then used as an input for generating the Accounting Report. The Pre Condition Return Code for the Load & Validate job is Successful.

4. **Generate Accounting Report** - The purpose of this job is to create a report which will contain all accounting errors generated during the validation of the CWA transaction. This job uses CWActgExpRep.txt generated by the Inferences & Validations job and the exception file generated by the Load & Validate job of the chain to generate the report. This report can be viewed in HTML or PDF format from the Advantage Job Manager. The Pre Condition Return Code for the Generate Accounting Report job is Successful.

5. **Discard the CWA transaction** - The purpose of this job is to discard the CWA transaction which has been loaded during this chain. This job uses DiscardCWATrans.txt created in the Inferences & Validations step as an input which contains details of the transaction to be discarded. The Pre Condition Return Code for the Discard job is Warning which indicates that this job will be scheduled in one of two situations:
   - When the Generate Accounting Report job has exited with a Return Code of Successful.
   - When the Generate Accounting Report job has exited with a Return Code of Warning.

Major Input

- Data from the following tables:
  - Check Writer Options (R_AP_CW_OPT) table
  - Check Writer Header (R_AP_CW_HDR) table
  - Check Writer Payment (R_AP_CW_PYMT) table
  - Check Writer Accounting (R_AP_CW_ACTG) table
  - Check Writer Addendum (R_AP_CW_ADNM) table

Major Output

- 2 reports will be generated: Non-Accounting report and Accounting report
- CWA transaction will be created and discarded

Chain Return Code

The following table indicates the potential return codes with which the chain will end:

Return Code	Condition
Successful (1)	All of the jobs end successfully.
Warning (4)	One of the jobs in the chain ends with a return code of “Warning.”
Non-Fatal Error (8)	One of the jobs in the chain ends with a return code of “Non-Fatal Error.”
Failed (12)	One of the jobs in the chain ends with a return code of “Failed.”

Terminated (16)	One of the jobs in the chain ends with a return code of “Terminated.”
System Failure (20)	One of the jobs in the chain ends with a return code of “System Failure.”

Problem Resolution

Please refer to the individual job “Problem Resolution” section for more details.

CW Pre-Edit Chain: Inferences & Validations

Job Name	Inferences & Validations
Recommended Frequency	On Demand
Single Instance Required	No
Can be restarted	No
Reports Generated	Check Writer Pre-Edit Non-Accounting Report

Overview

The CW Pre-Edit chain begins with the Inferences & Validations. This step infers values to the fields in the CW Header, Accounting, and Payment tables from the various related tables. These inferences are required to ensure that all of the necessary fields are populated before any validations occur against the CWA transaction or CW tables. The validations that occur during this step include all vendor and payment constraints as set by the CW Options table (CWOPT). Errors found during this part of the Pre-Edit chain are written to the Non-Accounting Report. If errors occur during validations the Return Code of the job is set to Warning. When the job completes without any errors in validation the return code is set to Successful.

Once all inferences and validations are done a Non-Accounting Report is generated. The report lists all errors, severity and the location of the error found from the checks performed. The report may be viewed in HTML or PDF format from the Advantage Job Manager.

CW File ID’s that have no Accounting Lines or the sum of the Accounting Lines equals $0 will have the CW Pre-Edit Processing User ID, CW Pre-Edit Processing Date/Time Stamp and Pre-Edit Successful flag updated on the CW Header table during this step because they will not be processed in subsequent steps. The Pre-Edit Successful flag is set to “True” if no hard errors are found on the Non-Accounting Report. If there are hard errors on the Non-Accounting Report, the Pre-Edit Successful flag is set to “False”.

Process Steps	Messages
1. Parameter Validation	• Validating the Batch Parameters. • Parameter validation completed.
2. Performing Inferences &	• ‘N’ Accounting Records Inferred.

Validations.	• 'N' Payment Records Inferred. • • Where 'N' = no. of Records.
3. Creation of the required Parameter files.	• Started creating Parameter files. • Parameter files creation completed.
4. Creation of Non-Accounting Report	• Rendering report started. • Rendering report completed. •

Restartability

Job cannot be restarted. If the job fails then no Database restore is required on any of the tables. A new chain should be scheduled for the same File ID.

Major Input

- Check Writer Header (CWHDR) table
- Check Writer Payment (CWPYMT) table
- Check Writer Accounting (CWACTG) table
- Check Writer Options table (CWOPT)
- Check Writer Addendum table (CWADNM)

Batch Parameters

Parameter	Description	Default Value
Parameter Directory AMSPARM	Parameter Directory location (** **Refer to Note:** Assumptions for SWBP on page no. 7)	$$AMSPARM$$
Export Directory AMSEXPORT	Export Directory location (** **Refer to Note:** Assumptions for SWBP on page no. 7)	$$AMSEXPORT$$
Log Directory AMSLOGS	Log Directory Location (** **Refer to Note:** Assumptions for SWBP on page no. 7)	$$AMSLOGS$$
Chain Job ID CHAIN_JOB_ID	Chain Job ID	$$@CHAINJOBID@$$

Client Name CLIENT_NAME	Client Name	<blank>
CW Department Code CW_DEPT_CD	CW Department Code. Multiple Department Code values are not allowed.	<blank>
CW File ID CW_FILE_ID	CW File ID. Multiple File IDs are not allowed.	<blank>
CWA Transaction Code CWA_DOC_CD	CWA Transaction Code	<blank>
User ID USER_ID	User ID	<blank>
Parameter File Name PARM_FILE_NM	Accounting Transaction XML Generation Parameter File Name	CreateCWAXML.txt
Parameter File Name PARM_FILE_NM	Load & Validate Parameter File Name	LoadValidateCWATrans.txt
Parameter File Name PARM_FILE_NM	Generate Accounting Report Parameter File Name	CWActgExpRep.txt
Parameter File Name PARM_FILE_NM	Discard Parameter File Name	DiscardCWATrans.txt
Progression Message Block Size PROG_CTR_SZ	Required. This field provides the information about the record being processed.	No Default.
Overridable Error Check CHK_OVERRIDDEN_ERR	A required parameter that decides whether the overridden errors will be considered as hard errors by the CW Pre-Edit job to mark the "Pre-Edit Successful" status as Yes or No on Check Writer Header (CWHDR) based on the validation performed. When set to Y, the job checks if all the errors raised during validation have a severity of Error, Severe, or Override, and if it is any one of them, then marks the status of the "Pre-Edit Successful" as *No* on	Y

	Check Writer Header (CWHDR). When set to N, the job should check only errors raised during validation with a severity of Error, Severe and exclude Override severity, and if none of them, then mark the status of the "Pre-Edit Successful" as *Yes* on Check Writer Header (CWHDR).	

Major Output

4 parameter files are created by this job:

- **CreateCWAXML.txt** - used as input by the Accounting Transaction XML Generation job of the CW Pre-Edit chain.
- **LoadValidateCWATrans.txt** - used as input by the Load & Validate job of the CW Pre-Edit chain.
- **CWActgExpRep.txt** - used as input by the Generate Accounting Report job of the CW Pre-Edit chain.
- **DiscardCWATrans.txt** - used as input by the Discard job of the CW Pre-Edit chain.

Job Return Code

The following table shows the potential job return codes for the Inferences & Validations job:

Return Code	Condition
Successful (1)	All of the validations are performed successfully and no errors are reported.
Warning (4)	Validations result in errors. If the severity in the CW Non-Accounting report is Error/Severe, the Return Code of this job is set to Warning and the process moves to the next step in the Pre-Edit chain.
Non-Fatal Error (8)	This Return Code is issued if there are no records in the CWACTG table for the given CW Department and CW File ID combination OR the sum of the Accounting Line Amount = $0.
Failed (12)	The job will fail under the following conditions: • Parameters are invalid • No CW Header records are selected for processing • Technical/System failure When this job ends with a Return Code of Failed, subsequent jobs in the chain are set to Inactive.
Terminated (16)	This Return Code is issued when the job is terminated by the user. When this job ends with a Return Code of Terminated,

	subsequent jobs in the chain are set to Inactive.
System Failure (20)	This return code is issued when the job is terminated because of database server or network issues. When this job ends with a Return Code of System Failure, subsequent jobs in the chain are set to Inactive.

Selection Criteria

Records are selected for processing based on the following criteria:

- CW File ID with a run status of "Pending Department Certification" or "Pending Central Office Certification"

Sort Sequence

N/A

Problem Resolution

The following table shows the possible return codes and recommendations for each processing step.

Step 1: Parameter Validation:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	N/A	N/A	N/A
Warning (4)	If the validations on CWHDR, CWPYMT and CWACTG result in errors.	Correct the data and then run the job to make it successful.	N/A
Non-Fatal Error (8)	If there are no records in the CWACTG table for the given CW Department and CW File ID combination OR the sum of Accounting Line Amount = $0.	N/A	N/A
Failed (12)	All of the required parameters are not entered. **Sample Message:** CWA Transaction Code parameter cannot be	If the required parameters are not entered, enter the required parameters and Schedule a new job.	

Possible Return Codes	Condition	Recommendation	Other Instructions
	blank. **Recommendation:** Enter the CWA Transaction Code.		
	Entered Parameters are not valid. **Sample Message**: User ID is not valid within the application. **Recommendation:** Enter a valid User ID.	If the parameters are invalid, enter the valid parameter and Schedule a new job.	
	Failed because of runtime exceptions for unexpected situations.	Failure reason needs to be investigated before scheduling a new job. When the job fails, inferences to CW Header, Payment or Accounting tables are not made and no database back out is required.	
Terminated (16)	Job is terminated manually by the user.	Reason for the termination needs to be investigated. When the job terminates, the new job can be rescheduled and all of the record values are re-inferred on the respective CW tables.	N/A
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. A new job can be rescheduled. All of the records are re-inferred on the respective CW tables when the job is rescheduled.	N/A

Step 2: Performing Inferences & Validations:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the inferences and validations performed successfully.	N/A	N/A
Warning (4)	Not all validations are successful, that is, there are errors found in the non-accounting report.	Correct the data so that all of the validations are performed successfully.	N/A
Non-Fatal Error (8)	If the CW record contains no Accounting Lines or the sum of all Accounting Lines Amounts is equal to $0.	N/A	N/A
Failed (12)	No eligible records are selected for Pre-Edit with a run status of “Pending Central Office Certification” or “Pending Department Certification”.	Correct the data and re-run the chain. A valid run status for CW Pre-Edit chain is Pending Department Certification or Pending Central Certification.	N/A
	Failed because of runtime exceptions for an unexpected situation.	Failure reason needs to be investigated before rescheduling the job. When the job fails, inferences to CW Header, Payment or Accounting tables are not made and no database back out is required.	N/A
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated and a new job scheduled.	N/A
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated and a new job scheduled.	N/A

Step 3: Creation of the required Parameter files:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	N/A	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Failed because of issues in creating parameter files, for example, parameter file location not found.	Investigate the reason of failure and then re-run the job. If any location issues, correct the same.	Parameter files will be located in the AMSPARM directory.
	Failed because of runtime exceptions for an unexpected situation.	Failure reason needs to be investigated before scheduling a new job. If the CW Run Status is updated then change it to the valid one. A valid run status for the CW Pre-Edit chain is Pending Department Certification or Pending Central Certification. When the job fails, inferences to CW Header, Payment or Accounting tables are not made and no database back out is required.	Failed because of runtime exceptions for an unexpected situation.
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated and a new job scheduled.	N/A
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated and a new job scheduled.	N/A

Step 4: Creation of Non-Accounting Report:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	The Non-Accounting report generated successfully.	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Failed due to issues in creating the report.	Investigate the reason for the report not being created and re-run the chain.	N/A
	Failed because of runtime exceptions for an unexpected situation.	Failure reason needs to be investigated before rescheduling a new job. Change the CW Run Status, in case it is already updated. A valid run status for the CW Pre-Edit chain is Pending Department Certification or Pending Central Certification.	N/A
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated and a new job scheduled.	N/A.
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated and a new job scheduled.	N/A.

CW Pre-Edit Chain: Accounting Transaction XML Generation

Job Name	Accounting Transaction XML Generation
Recommended Frequency	On Demand
Single Instance Required	No
Can be restarted	No. Refer Restartability Section.
Reports Generated	No

Overview

The next step in the CW Pre-Edit chain is Accounting Transaction XML Generation, for records with Accounting Lines and Accounting Line Amounts greater than $0, records with “Pending Central Office Certification” or “Pending Department Certification”, and records that have not already generated a CWA transaction. This step creates an XML file that is used to populate the Check Writer Accounting (CWA) transactions. The CWA transaction is a clone of the Cost Accounting Transaction, and is created during the CW Pre-Edit chain. The input parameters for this step include CW Department Code, CW File ID and the CWA Transaction Code. The CWA XML file is created, in transaction XML format, based on the input parameters. Data is pulled from the CW Header and Accounting tables. A single CWA XML file is created for all of the records being processed in the CW Pre-Edit job, and is uniquely named by the CWA Transaction Code and the job’s Chain ID.

Restartability

The job cannot be restarted. If the job is failed for any reason, a new chain should be scheduled for the same File ID.

Major Input

- CreateCWAXML.txt generated in the first step of the CW Pre-Edit chain
- Data from the CWHDR and CWACTG tables to generate the CWA Transaction

Batch Parameters

Parameter	Description	Default Value
Parameter Directory location PARM_FILE	The file location and file name of the parameter file created in the Inferences & Validations job.	$$AMSPARM$$/CreateCWAXML$$ @CHAINJOBID@$$.txt
Progression Message Block Size PROG_CTR_SZ	Required. This field provides the information about the record being processed.	No Default

Major Output

- The CWA XML file, which will be uniquely named by the CWA Transaction Code and the job’s Chain ID.

Job Return Code

The following table shows the potential job return codes for the Accounting Transaction XML Generation job:

Return Code	Condition
Successful (1)	CWA XML file generated successfully.

Warning (4)	This job does not issue this Return Code.
Non-Fatal Error (8)	This job does not issue this Return Code.
Failed (12)	The job will fail under the following conditions: • Parameters are invalid • Issues in creating the xml file • Technical/System failure When this job ends with a Return Code of Failed, subsequent jobs in the chain are set to inactive.
Terminated (16)	This Return Code is issued when the job is terminated by the user. When this job ends with a Return Code of Terminated subsequent jobs in the chain are set to Inactive.
System Failure (20)	This Return Code is issued when the job is terminated because of database server or network issues. When this job ends with a Return Code of System Failure, subsequent jobs in the chain are set to Inactive.

Selection Criteria

- Records are selected for processing based on the following criteria:
- CW File ID with a run status of "Pending Department Certification" or "Pending Central Office Certification
- CW Accounting Lines and Accounting Line Amounts > $0
- Records that have not already generated a CWA transaction

Sort Sequence

N/A

Problem Resolution

The following table shows the possible return codes and recommendations for each processing step.

Step 1: Creation of XML File:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	If the CWA XML File is created successfully.	N/A	N/A
Warning (4)	N/A	N/A	N/A

Possible Return Codes	Condition	Recommendation	Other Instructions
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Issues in creating XML file.	Investigate the issue and re-run the job. Change the CW Run Status in case it is already updated and re-run the job so that the records are picked up.	
	Failed because of runtime exceptions for an unexpected situation.	Failure reason needs to be investigated before scheduling a new job. When the job fails, inferences to CW Header, Payment or Accounting tables are not made and no database backout is required.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated and a new job scheduled.	N/A
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated and a new job scheduled.	N/A

CW Pre-Edit Chain: Load & Validate Job

Job Name	Load & Validate
Recommended Frequency	On Demand
Single Instance Required	No
Can be restarted	No
Reports Generated	No

Overview

The Load & Validate job is the third step in the CW Pre-Edit chain. The load uses the existing Advantage SysManUtil function with the "Transaction Import" action to create the CWA transactions from the CWA XML file. The CWA transaction created in this step is a temporary transaction used to validate data from the CW tables. The Advantage SysManUtil function with the "Transaction Validate" action validates the accounting elements in the CWA transaction. This step has an additional validation for the TIN and TIN Type presence when COA elements are 1099 reportable. This validation checks the Object, Sub Object, BSA, and Sub BSA for 1099 reportability on the respective tables. If any of those elements contain a 1099 income code, and the TIN & TIN Type for Miscellaneous Vendors is not Optional on the CWOPT table, the TIN and TIN Type are required on the CW Payment table for Miscellaneous vendors. This validation is done at this time, and is not done during the Inferences and Validations step of the CW Pre-Edit chain, due to the potential presence of an Accounting Template and the way COA elements are inferred.

Process Steps	Messages
1. Upload and Validate CWA transaction	• Rows Processed : N • Rows Saved : N • Rows in Error : N • Total # Processed N : Failed n : Succeeded N-n • N = No. of records.

Restartability

Job cannot be restarted. If the job failed for any reason, a new chain should be scheduled for the same File ID. If the job is failed after loading the transaction into the transaction, then before scheduling a new chain, loaded transaction should be discarded manually. This is because the CW File ID is used as the CWA Transaction ID.

Major Input

- LoadValidateCWATrans.txt parameter file generated in the first step of the CW Pre-Edit chain
- CWA xml file generated in the second step of the CW Pre-Edit chain

Major Output

- CWA Transaction is created with Transaction ID = CW File ID
- Exception file is created which contains the accounting errors as a result of CWA validation and it is uniquely named as CWExcepRep.txt

Batch Parameters

Parameter	Description	Default Value
Parameter File Name PARM_FILE	File location and the file name of the parameter file created in the Inferences & Validations job.	$$AMSPARM$$/LoadValidateCWADoc$$ @CHAINJOBID@$$.txt

Job Return Code

The following table shows the potential job return codes for the Load & Validate job:

Return Code	Condition
Successful (1)	The CWA transaction is uploaded to the transaction catalog successfully and validated successfully. An Exception file is created for the errors encountered during validation.
Warning (4)	This job does not issue this Return Code.
Non-Fatal Error (8)	This job does not issue this Return Code.
Failed (12)	The job will fail under the following conditions: • Parameters are invalid • Issues in reading the input files and creating the CWA transaction • Technical failure When this job ends with a Return Code of Failed, subsequent jobs in the chain will be set to Inactive.
Terminated (16)	This return code is issued when the job is terminated by the user. When this job ends with a Return Code of Terminated, subsequent jobs in the chain are set to Inactive.
System Failure (20)	This return code is issued when the job is terminated because of database server or network issues. When this job ends with a Return Code of System Failure, subsequent jobs in the chain are set to Inactive.

Sort Sequence

N/A

Problem Resolution

The following table shows the possible return codes and recommendations for each processing step.

Step 1: Upload and Validate CWA transaction:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	The CWA Transaction is uploaded successfully to the Transaction Catalog. The CWA transaction validated successfully.	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Issues in reading the parameter file. **Sample issue:** Parameter File not found in the AMSPARM directory. **Recommendation:** Place the file in the AMSPARM directory, where it can be read by the SMU job.	Investigate the issue and schedule a new job.	N/A
	Failed because of runtime exceptions for an unexpected situation.	Failure reason needs to be investigated before rescheduling the job. Change the CW Run Status in case it is already updated. The Valid run status for CW Pre-Edit chain is Pending Department Certification or Pending Central Certification.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated and a new job scheduled.	N/A
System Failure (20)	When the job is terminated because of database server or network issues	The reason for the System Failure needs to be investigated and a new job scheduled.	N/A

CW Pre-Edit Chain: Generate Accounting Report Job

Job Name	Generate Accounting Report
Recommended Frequency	On Demand
Single Instance Required	No
Can be restarted	No
Reports Generated	Check Writer Pre-Edit Accounting Report

Overview

This next step of the CW Pre-Edit chain is Generate Accounting Report. This report is created from the validations performed by the SysManUtil "Transaction Validate" action on the CWA transaction. If errors occur during validations the Return Code of the job is set to Warning. When the job completes without any errors in validation the Return Code is set to Successful.

The errors generated from the CWA transaction are put in the report, which can be viewed in HTML or PDF format from the Advantage Job Manager. At this time, the CW Header table is updated for the completion of the CW Pre-Edit chain job. The CW Pre-Edit Processing User ID, CW Pre-Edit Processing Date/Time Stamp and Pre-Edit Successful flag are updated. The Pre-Edit Successful flag is set to "True" if no hard errors were found on the Accounting Exception Report and the Non-Accounting Successful flag in the Generate Accounting Report parameter file is set to "True". If either the Accounting Report is found with hard errors or the Non-Accounting Successful flag is set to "False", the Pre-Edit Successful flag will be set to "False" on the CW Header table. These updates are only performed when the record contains Accounting Lines and the sum of the Accounting Line Amounts is greater than $0, as stated in the Inferences and Validations step above.

Process Steps	Messages
1. Batch Parameter validation.	• Validating the Batch Parameters • Parameter validation completed
2. Generation of Accounting Report.	• Reports output folder mapped • Rendering report started • Rendering report completed

Restartability

This job cannot be restarted. If this job fails, the new chain can be re-run for the same File ID. Before running the chain, any previous loaded transaction, should be deleted from the Transaction Catalog manually, since the CWA transaction created in the previous steps exists on the Transaction Catalog.

Major Input

- Exception file, which is uniquely named as CWExcepRep.txt (CWExcepRep.txt is generated in the Load & Validate Job step of the CW Pre-Edit chain)

Batch Parameters

Parameter	Description	Default Value
Exception file EXCEP_FILE_NM	The exception file generated in the Load & Validate Job step of the CW Pre-Edit chain	CWExcepRep$$ @CHAINJOBID@$$.txt
Parameter File PARM_FILE	The file location and file name of the parameter file created in the Inferences & Validations job.	$$AMSPARM$$/CWActgExpRep$$ @CHAINJOBID@$$.txt

Major Output

- Accounting Report that can be viewed in HTML and PDF format from the Advantage Job Manager

Job Return Code

The following table shows the potential job return codes for the Generate Accounting Report job:

Return Code	Condition
Successful (1)	Accounting report generated successfully.
Warning (4)	Validations result in errors.
Non-Fatal Error (8)	This job does not issue this Return Code.
Failed (12)	The job will fail under the following conditions: • Parameters are invalid • Issues in creating the report • Technical/System failure When this job ends with a Return Code of Failed, subsequent jobs in the chain are set to Inactive.
Terminated (16)	This Return Code is issued when the job is terminated by the user. When this job ends with a Return Code of Terminated, subsequent jobs in the chain are set to Inactive.
System Failure (20)	This Return Code is issued when the job is terminated because of database server or network issues. When this job ends with a Return Code of System Failure, subsequent jobs in the chain are set to Inactive.

Sort Sequence

N/A

Problem Resolution

The following table shows the possible return codes and recommendations for each processing step.

Step 1: Parameter Validation:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	The Accounting Report generated successfully.	N/A	N/A
Warning (4)	The validations on CWHDR and CWACTG result in errors.	Correct the data and then run the job to make it successful.	
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Failed due to issues in creating the report.	Investigate the reason for the report not being created and re-run the chain. Change the CW Run Status, in case it is already updated, and re-run the job so that the records are picked. A Valid run status for CW Pre-Edit chain is Pending Department Certification or Pending Central Certification.	N/A
	Failed because of runtime exceptions for an unexpected situation.	The Failure reason needs to be investigated before rescheduling the job. Change the CW Run Status on the CW Header table for this CW File ID, in case it is already updated, and re-run the chain so that the records are picked up.	

Possible Return Codes	Condition	Recommendation	Other Instructions
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated and a new job scheduled.	N/A
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated and a new job scheduled.	N/A

CW Pre-Edit Chain: Discard Job

Job Name	Discard
Recommended Frequency	On Demand
Single Instance Required	No
Can be restarted	No
Reports Generated	No

Overview

After the generation of the accounting report, the next step of the CW Pre-Edit chain is the Discard. This step uses the SysManUtil function with the "Transaction Discard" action to delete loaded CWA transactions. CWA transactions are selected for deletion based the Transaction ID and Transaction Type parameters for this step.

The CW Pre-Edit chain terminates with a “Successful” job Return Code if the steps of the chain are processed correctly. If any of the steps in the chain fails to process successfully, the job will terminate. If the CW Pre-Edit chain does not process beyond the parameter validations the job Return Code is set to “Failed”.

After the CW Pre-Edit chain, users can view the generated reports to determine the necessary updates to the CW files. The CW records with errors should be un-certified by the authorized departments allowing them to be deleted in the CW Clean Up job.

Restartability

This job cannot be restarted. If the job failed in this step, then a new chain should be scheduled for the same File ID. Alternatively, the CWA transaction can be manually discarded from the Transaction Catalog.

Major Input

- DiscardCWADOC.txt parameter file created in the first step of the CW Pre-Edit chain

Batch Parameters

Parameter	Description	Default Value
Parameter File. PARM_FILE	The CWA Discard step has a parameter file created from the user input parameters during the Inferences step. This parameter file has a default file name of DiscardCWADOC.txt	$$AMSPARM$$/DiscardCWADOC.txt

Major Output

- CWA transaction is permanently discarded from the Transaction Catalog

Job Return Code

The following table shows the potential job return codes for the Discard job:

Return Code	Condition
Successful (1)	CWA transaction discarded successfully.
Warning (4)	This job does not issue this Return Code.
Non-Fatal Error (8)	This job does not issue this Return Code.
Failed (12)	The job will fail under the following conditions: • Parameters are invalid • Issues in SysManUtil while discarding the transaction • Technical/System failure
Terminated (16)	This return code is issued when the job is terminated by the user.
System Failure (20)	This return code is issued when the job is terminated because of database server or network issues.

Sort Sequence

N/A

Problem Resolution

The following table shows the possible return codes and recommendations for each processing step.

Step 1: CWA Transaction Discard:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	CWA transaction is discarded from the Transaction Catalog. This is done by the SysManUtil “Transaction Discard” action functionality.	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Failed because of runtime exceptions for an unexpected situation.	Failure reason needs to be investigated.	N/A
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated.	N/A
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated.	N/A.

2.2.15 CW Restore

Job Name	CW Restore
Recommended Frequency	On Demand
Single Instance Required	No. If multiple users schedule this job, then all of the requests will be processed simultaneously and the job will end with appropriate return codes, depending on the records available for restore.
Can the job be restarted	Yes
Reports generated	No

Overview

The Facilitator process reads the Archive Facilitator table for a specified Run Number. For each record found whose Facilitator Status = "Table Archive Complete", the Facilitator initiates a System Maintenance Utility process. The System Maintenance Utility is invoked with an action of "Table Import" and restores all records found within the XML file associated with each individual Archive Facilitator record, restoring the records to the Check Writer Vendor Intercept (CWVINCT), Check Writer ACH Addendum (CWADMN), Check Writer Payment (CWPYMT), Check Writer Accounting (CWACTG), Check Writer Header (CWHDR), and Check Writer Pre-Assigned Check Number (CWCHK) tables in that order, which is the reverse order specified for archiving. As each XML file restore completes, the Facilitator changes the Status on the Archive Facilitator table to "Table ready for archive" for the current record being processed. There are no reports or output files created from the restore process.

Process Steps	Messages
1. Processing and restoring records.	The following message is saved to the log, each time a SMU process is spawned (where XXXX is the Job ID, 'x' is the time in seconds, and 'n' is the number of times the job slept): • SMU Job XXXX - Spawned • SMU Job - XXXX - Processing completed successfully • The job slept a total of n times, for a Total Sleep Time of x seconds If the CW Restore job is restarted then the following message is logged (where XXXX is the Job ID): • Restarted Job XXXX

Restartability Information

If the job was discontinued for any reason then the chain has the ability to be restarted from the point it left off.

Major Input

- Facilitator Table
- Parameter files created by the Check Writer Archive Chain process

Batch Parameters

Parameter	Description	Default Value
Archive Restore ID (ARCHIVE_RESTORE_ID)	Required non editable field. Entry of a value in this field specifies the System Maintenance Utility command to be executed during the Archive Restore. Must be “2” for Restore.	2
Commit Block Size (COMMIT_BLOCK_SIZE)	Required field. Controls how many records are committed by the application at one time. The size should be compatible with technical capabilities and performance guidelines.	<blank>
Number of Processors (PROCESSOR_NO)	Required Field. Defines the number of SMU processes to be executed simultaneously.	<blank>
Run Number (RUN_NO)	Required Field. This field is required to specify the Run Number to restore.	<blank>
Sleep Time (SLEEP_TIME)	Required Field. Defines the frequency to pool for free processing slots when initiating SMU processes. Any integer number greater than Zero.	<blank>

Major Output

- Check Writer Pre-Assigned Check Number(CWCHK) table
- Check Writer Header(CWHDR) table
- Check Writer Accounting (CWACTG) table
- Check Writer Payment(CWPYMT) table
- Check Writer ACH Addendum(CWADNM) table
- Check Writer Vendor Intercept(CWVINCT) table

- Archive Facilitator table

Job Return Code

Return Code	Condition
Successful (1)	All of the records restored correctly.
Warning (4)	This job returns Warning when: • No Facilitator records are found. • SMU job parameter file not found.
Non-Fatal Error (8)	N/A
Failed (12)	The job fails under the following conditions: • Parameters are invalid • Technical/System failure
Terminated (16)	This return code is issued when the job is terminated by the user.
System Failure (20)	This return code is issued when the job is terminated because of database server or network issues.

Sort Sequence

N/A

Selection Criteria

The Facilitator reads the records from the Facilitator table for the Run Number indicated by the Run Number parameter. The Facilitator generates SMU processes to import the records that have been identified as "archive complete".

Problem Resolution

The job can be restarted if it fails for any reason. If the job fails for some reason and a new job is scheduled instead of restarting the failed job, then the new job runs as a new independent job for the Run Number mentioned to process the job. No database restore is required if the job fails for any reason and cannot be restarted.

The following table shows the possible return codes and recommendations for each processing step.

Step 1: Processing and restoring records:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful.	N/A	N/A
Warning (4)	Returns Warning when: • No Facilitator records are found. • SMU job parameter file not found.	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Failed because of runtime exceptions caused by unexpected conditions.	Failure reason needs to be investigated before restarting the job	N/A
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can be restarted or rescheduled.	N/A
System Failure (20)	Job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can be restarted or rescheduled.	N/A

2.2.16 CW Stale Escheat

Job Name	CW Stale Escheat Process
Recommended Frequency	On Demand
Single Instance Required	No
Can the job be restarted	No
Exception reports generated	No

Overview

Once a check is issued, an entity usually establishes a certain number of days in which the check must be redeemed. A check is stale when it has not been redeemed within the established number of days. When this happens, an entity has the right to reclaim the funds in question until such time as the holder of the check claims the funds.

An escheat is the succession of abandoned property to the state. It is commonly associated with property that comes from the estate of a person dying without a will and without any known heirs. However, this concept has been broadened to include the recovery of any property that results from the failure of a person legally entitled to that property to make a valid claim against the holder of the property within a prescribed period of time. Consequently, the terms escheat and unclaimed property are used interchangeably. The types of property that are often unclaimed may include checking and savings accounts and even uncashed checks.

This process is used to cancel the checks issued by the check writer process. If the issued checks are incorrect they could undergo the following cancellation actions.

- Stale
- Escheat

The Stale Escheat process picks up those records from the Check Reconciliation table with the status of Disbursed, Warranted, or Undelivered and the difference between Current Date and Issue Date greater than the number of days specified in the SOPT table for Stale days or Escheat Days, depending on whether Stale Dating or Escheating method is being used. Then it builds the CWC transaction from the CHREC using information from the CW Header, CW Payment and CW Accounting records associated with the CW File ID specified in the Transaction ID field of the CHREC record and submits the transaction.

The following table shows the various steps that the AD XML Creation Job goes through and the messages issued at each step.

Process Steps	Messages
1. Parameter Validation	• Validating Batch Parameters • Batch Parameter validation completed
2. Selection of Records	• If the selection returns 0 records, then the following message will be issued: “No records present in the Check reconciliation screen for the process”

Process Steps	Messages
3. Process Records	• A message is written to the log each time the job attempts to submit a transaction, indicating if it was successful or not • The Following message is displayed when the transaction submits successfully: • Transaction <Transaction_No.> submitted without errors

Restartability Information

This job cannot be restarted. If the job is failed for any reason, the new job should be scheduled with the same parameters. There is no need to back out any updates since no tables updates are performed in this job. All of the updates are triggered from the CWC transaction when it goes to final.

Major Input

- Check Reconciliation table (R_AP_CHK_RECON)
- System Options (R_GEN_SOPT)
- Check Writer Tables

Batch Parameters

Parameter	Description	Default Value
Transaction Code (DOC_CD)	Required Field. This field refers to the Transaction that will be generated by the CW Stale Escheat process.	No Default
Department Code (DOC_DEPT_CD)	Required Field. Defines the Transaction Department to be used with the Transaction Code and Prefix to find an Automatic Transaction Numbering entry.	No Default
Transaction prefix (DOC_ID)	Required Field. The transaction prefix to be used with the Transaction Code and Department Code, to find an Automatic Transaction Numbering entry.	No Default
Transaction Unit Code (DOC_UNIT_CD)	Optional Field. Provides security with a unit code enabling transaction access to be secured at a level below department.	No Default
Bank Code	Optional Field. If left blank, executes the process for all banks. To execute the process for a	No Default

Parameter	Description	Default Value
(BANK_ACCOUNT_CD)	subset of banks, enter one or more bank codes, separated by commas. Enter only valid Bank codes or the process will fail.	

Major Output

- Disbursement Cancellation Transaction (CWC)
- Update to the Check Reconciliation Table

Job Return code

The following table shows the potential job return codes for the CW Stale Escheat job.

Return Code	Condition
Successful (1)	All of the selected payment records are processed successfully.
Warning (4)	N/A
Non-Fatal Error (8)	N/A
Failed (12)	The job will fail under the following conditions: • Parameters are invalid • Run time exceptions for unexpected situations • No eligible records found
Terminated (16)	This return code is issued when the job is terminated by the user.
System Failure (20)	This return code is issued when the job is terminated because of database server or network issues.

Sort Sequence

None

Selection Criteria

Select records from Check Reconciliation where the following are true:

- Status of Disbursed, Warranted, or Undelivered
- Transaction Code of ‘CW” or ‘CE’
- The difference between Current Date and Issue Date greater than the number of days specified on System Options (SOPT) for Stale days or Escheat Days
- Optionally - Bank matches the Bank entered in the job parameter.

Problem Resolution

If there are any errors while creating a CWC draft version, the process leaves that particular record and moves forward to process the rest of the records. Rejected records can be reprocessed by running the job again with the same parameters after correcting the errors.

If a CWC transaction is created successfully in Draft but there are problems in submitting the transaction, the process leaves that transaction and moves forward to process the rest of the transaction. Failed transactions are written to the logs and the users can process the transaction manually after correcting the errors.

This batch program produces new Advantage transactions, which are subject to the line limit functionality constraints. Sites should ensure that they run this job with parameters set to ensure that the created transactions are within the line limit controls.

The following table shows the possible return codes and recommendations for each processing step.

1. Parameter Validation:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the parameters validated successfully.	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Required Parameters are not entered. **Sample Messages:** Transaction Code cannot be null. Transaction Department Code cannot be null. Invalid Transaction ID prefix.	Reschedule the job and enter the required parameters.	N/A

Possible Return Codes	Condition	Recommendation	Other Instructions
	Parameters are Invalid. **Sample Messages:** Entered Transaction Code is not found on R_GEN_DOC_CTRL. Entered Department Code is not found on R_DEPT. Entered Transaction Unit Code is not found on R_UNIT. Invalid Transaction ID prefix. One or more values entered in the Bank Code parameter is not a valid value on BANK.	Reschedule the job and enter valid parameters.	N/A
	Job failed due to Fatal conditions	The reason for failure needs to be investigated before rescheduling the job.	N/A.
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated before rescheduling the job.	N/A
System Failure (20)	Job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated before rescheduling the job.	N/A

2. Selection of records:

Possible Return Codes	Condition	Recommendation	Other instructions
Successful (1)	Eligible records found.	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Stale and Escheat are not applicable as Flags are not set in the SOPT.	Ensure that on SOPT, required flags are checked for stale/escheat.	N/A.
	No records present in the Check reconciliation screen for the process.	Schedule a job with different parameters if needed.	
	Job failed due to Fatal conditions	The reason for failure needs to be investigated before rescheduling the job.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated before rescheduling the job.	N/A
System Failure (20)	Job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated before rescheduling the job.	N/A

2.2.17 CW Table Load Chain

Job Name	CW Table Load
Recommended Frequency	On Demand
Single Instance Required	No.
Can be restarted	No.
Exception reports generated	No.

Overview

The CW Table Load Chain allows users to load check files to the Advantage application for the generation of check and EFT payments. The CW files are loaded by authorized users who execute the CW Table Load Chain job against a single or multiple CW files that have been previously loaded to the Advantage application server. The CW Table Load Chain is made up of 2 steps:

1. Header Load Job

   The purpose of this job is to load the CW Header information into the R_AP_CW_HDR table. It also creates a component XML file and a parameter file which serve as inputs for the next job in the chain

2. Component Load Job

   The purpose of this job is to load the CW components, that is, the CW Accounting, CW Payment and CW Addendum information into the R_AP_CW_ACTG, R_AP_CW_PYMT, and R_AP_CW_ADNM tables respectively.

   The Pre Condition Return Code for the CW Component Load job is Successful, which indicates that this job will be scheduled only if the CW Header Load job has exited with a Return Code of Successful.

Major Input

- CW XML input file

Major Output

Records are inserted into the following tables:

- R_AP_CW_HDR
- R_AP_CW_ACTG
- R_AP_CW_PYMT and
- R_AP_CW_ADNM

Chain Return Code

The following table indicates the potential return codes with which the chain will end:

Return Code	Condition
Successful (1)	All of the jobs end successfully.
Warning (4)	One of the jobs in the chain ends with a Return Code of "Warning."
Non-Fatal Error (8)	One of the jobs in the chain ends with a return code of "Non-Fatal Error."
Failed (12)	One of the jobs in the chain ends with a return code of "Failed."
Terminated (16)	One of the jobs in the chain ends with a return code of "Terminated."
System Failure (20)	One of the jobs in the chain ends with a return code of "System Failure."

Problem Resolution

Please refer to the individual job "Problem Resolution" section for more details.

CW Table Load Chain: Header Load Job

Job Name	Header Load
Recommended Frequency	On Demand
Single Instance Required	No
Can be restarted	No
Reports Generated	No

Overview

The Header Load job inserts the records into the CW Header table from the input file. The job first validates the batch parameters. If the batch parameters are valid, the job tries to insert header records directly to the R_AP_CW_HDR table from the input file. If the header record is successfully inserted, the job writes the remaining records into an XML file after performing the following:

- Populates the Header information on the Accounting and Payment records
- Populates Header and Payment line information on the Addendum records

In addition to that, the job also populates the CW header information on all of the records. If the header record is not inserted successfully, appropriate error messages are written to the Header Load job log and Advantage stops processing the remaining steps.

The following table shows the various steps that the Header Load Job goes through and the messages issued at each step.

Process Steps	Messages
1. Parameter Validation	Validating the Batch Parameters • If the batch parameter validation fails, the following message is displayed on the log, ”Parameter validation failed.” • If the batch parameters pass validation, the following message is displayed” Parameters are Valid.” • Parameter validation completed.
2. Processing of input CW XML file	• Processing the input XML file • Importing the CW Header record into the R_AP_CW_HDR table • If the Header imported successfully, the following message is displayed in the log” CW Header Record Department=<deptName>, CW File ID= <fileId> processed” • If the Header does not get imported successfully then the following message is displayed in the log“ CW Header Record Department=<deptName>, CW File ID= <fileId> failed to load.Remaining components bypassed” • Processing complete • If all of the records do not get imported then the following message is displayed in the log ”All header records failed to load”

Restartability information

The job cannot be restarted. If the job fails for any reason, only a new chain should be scheduled for the same File ID after correcting the errors that caused the job to fail.

Major Input

- Check Write XML file (Note: Multiple files can be entered separated by comma.)

Batch Parameters

Parameter	Description	Default Value
Import Directory AMSIMPORT	Required. CW Input File Directory (** **Refer to Note:** Assumptions for SWBP on page no. 7)	$$AMSIMPORT$$
Export Directory AMSEXPORT	Required. CW Split Component File Directory (** **Refer to Note:** Assumptions for SWBP on page no. 7)	$$AMSEXPORT$$
Input XML File Name FILE_NM	Required. CW Input XML File Name(s). Multiple files can be entered in the comma delimited format.	<blank>
Component XML File Name COMP_FILE_NM	Required. CW Component XML File Name	CWCompUpload.xml
Parameter Directory AMSPARM	Required. The File location for the parameter file created for use in the Component Load job step.	$$AMSPARM$$
User ID USER_ID	Optional. The User ID to be used for loading component records in the Component Load job step.	<blank>
Parameter File Name PARM_FILE_NM	Required. The File name of the parameter file created for use in the Component Load job step.	LoadComponentParm.txt
Progression Message Block Size (PROG_CTR_SZ)	Mandatory. This field provides the information about the record being processed.	No Default

Major Output

- R_AP_CW_HDR table – records will be inserted
- CW Component XML .file

Job Return Code

The following table shows the potential job return codes for the Header Load job.

Return Code	Condition
Successful (1)	All of the selected CW input XML files are processed successfully.
Warning (4)	• At least one, but not all, Header records failed to load successfully.

Return Code	Condition
	• If one of the input files does not exist in the specified location.
Non-Fatal Error (8)	No CW header record inserted successfully.
Failed (12)	The job will fail under the following conditions: • Parameters are invalid • Technical/System failure • All of the input CW Files do not exist in the specified location. When this job ends with a Return Code of Failed, subsequent jobs in the chain are set to Inactive.
Terminated (16)	This return code is issued when the job is terminated by the user. When this job ends with a Return Code of Terminated, subsequent jobs in the chain are set to Inactive.
System Failure (20)	This Return Code is issued when the job is terminated because of database server or network issues. When this job ends with a Return Code of System Failure, subsequent jobs in the chain are set to Inactive.

Sort Sequence

Not applicable for this job.

Selection Criteria

Not applicable, as this job parses the XML we supply as an input.

Problem Resolution

If the job fails at any point in time due to a technical issue, a new job should only be scheduled. If only one CW file is entered as the input for this job, then there is no need to back out any db updates since the updates occur only at the end of the job.

If multiple files are entered as input parameters and the job failed after processing a few files, then updates made to the CW Header table should be backed out before scheduling a new job for loading the same CW files. The CW File IDs can be taken from the processing log and the Header records can be deleted by running the CW Clean Up job for those files.

Step 1: Parameter Validation:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the parameters are validated	N/A	N/A

Possible Return Codes	Condition	Recommendation	Other Instructions
	successfully.		
Warning (4)	N/A	This step does not issue this return code.	This step does not issue this return code.
Non-Fatal Error (8)	N/A	This step does not issue this return code.	This step does not issue this return code.
Failed (12)	Required Parameters are not entered. **Sample Message:** CW File Name parameter cannot be blank. **Recommendation:** Enter the CW File Name.	If the required parameters are not entered, enter the required parameters and reschedule the job.	N/A
	Entered Parameters are not valid. **Sample Message**: User ID is not valid within the application. **Recommendation:** Enter a valid User ID.	If the parameters are invalid, enter the valid parameter and reschedule the job.	N/A
	Failed because of runtime exceptions for an unexpected situation.	Failure reason needs to be investigated before scheduling a new job.	N/A
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated and a new job scheduled.	N/A
System Failure	When the job is terminated because	The reason for the System Failure needs to be	N/A

Possible Return Codes	Condition	Recommendation	Other Instructions
(20)	of database server or network issues.	investigated and a new job scheduled.	

Step 2: Processing of input CW XML file:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All of the Header records are imported successfully.	N/A	N/A
Warning (4)	At least one, but not all, Header records failed to load successfully. **Sample Message:** CW Header Record Dept=<CW Department Code>, CW File ID= <CW File ID> failed to load. Remaining components bypassed.	**Recommendation:** Correct CW file(s) that failed and schedule a new chain only for the failed CW file(s).	N/A
	A CW file contained no header record. **Sample Message**: CW Input File <Name of the CW Input file> contained no header records. CW File bypassed.	**Recommendation:** Correct the CW file(s) that failed and schedule a new chain only for the failed CW file(s).	N/A
Non-Fatal Error (8)	No CW header record inserted successfully.	Correct CW header records in CW file(s) and re-run chain.	N/A
Failed (12)	Failed because of runtime exceptions for an unexpected situation.	Failure reason needs to be investigated before scheduling a new job.	N/A

Possible Return Codes	Condition	Recommendation	Other Instructions
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated and a new job scheduled.	N/A
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated and a new job scheduled.	N/A

Step 3: Creation of Component File:

Possible Return Codes	Condition	Recommendation	Other instructions
Successful (1)	Component XML file is created successfully.	N/A	N/A
Warning (4)	N/A	This step does not issue this return code.	
Non-Fatal Error (8)	N/A	This step does not issue this return code.	
Failed (12)	Failed because of runtime exceptions for an unexpected situation.	Failure reason needs to be investigated before scheduling a new job.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated and a new job scheduled.	N/A
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated and a new job scheduled.	N/A

Step 4: Creation of the Component Load Parameter file:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Component Load Parameter file is created successfully.	N/A	N/A
Warning (4)	N/A	This step does not issue this return code.	
Non-Fatal Error (8)	N/A	This step does not issue this return code.	
Failed (12)	Failed because of runtime exceptions for an unexpected situation.	Failure reason needs to be investigated before scheduling a new job.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated and a new job scheduled.	N/A
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated and a new job scheduled.	N/A

CW Table Load Chain: Component Load Job

Job Name	Component Load
Recommended Frequency	On Demand
Single Instance Required	No
Can be restarted	No
Reports Generated	No

Overview

The Component Load job uses SysManUtil (SMU) functionality to load the records into the respective tables from the input file. The input file name is specified in the parameter file supplied to this job.

The Component Load job processes each CW component record in the file sequentially, regardless of the order of the components (accounting, payment or ACH addendum). If the component fails to load, and if an addendum record comes before the payment line record, all of the addendum records will be rejected since the Payment line is part of the PKEY on the addendum records. If a component record fails to load into the corresponding CW table, the process will issue an error, bypass the failed record, and then the CW File Name of the failed record will be specified in the log and the job will continue to process the remaining records. Component records may fail loading if data provided is invalid or required data is missing from the XML file record. As an example, the CW Accounting component requires that a department be provided as part of the chart of account elements. If a value is not provided in the XML file, the load of the record will fail. The department is required for accounting level security.

Process Steps	Messages
1. Upload the Parameter file.	• Parameter file will be uploaded to the corresponding CW table.

Restartability Information

The job cannot be restarted. If the job fails due to some reason then reschedule a new chain job. SysMainUtil rolls back the changes if it is not successful; therefore, there is no need to back out any data while rescheduling the chain. Since the Header Load job inserted the records into the CW header table, those records should be deleted before scheduling the new chain job. The Header records can be deleted by running the CW Clean Up job for the failed CW File IDs.

Major Input

- Parameter file

Batch Parameters

Parameter	Description	Default Value
Parameter Directory AMSPARM	The file location and file name of the parameter file created in the Header Load job step. (** **Refer to Note:** Assumptions for SWBP on page no. 7)	$$AMSPARM$$/LoadComponentParm.txt

Major Output

Records are inserted into the following tables:

- R_AP_CW_ACTG,
- R_AP_CW_PYMT and
- R_AP_CW_ADNM

Job Return Code

The following table shows the potential job return codes for the Component Load job:

Return Code	Condition
Successful (1)	All of the components, as specified in the input XML, were imported successfully.
Warning (4)	CW component failed to load.
Non-Fatal Error (8)	Not Applicable for this job.
Failed (12)	• Input parameter file is not found • Runtime exceptions encountered for any unexpected situations
Terminated (16)	This return code is issued when the job is terminated by the user.
System Failure (20)	This return code is issued when the job is terminated because of database server or network issues.

Sort Sequence

Not applicable for this job.

Selection Criteria

Not applicable for this job.

Problem Resolution

The following table shows the possible return codes and recommendations for each processing step.

Step 1: Upload the Parameter file

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	When all of the components have been imported.	N/A	N/A
Warning (4)	This return code is issued when the job fails to load some of the transactions.	Analyze the reason for the failure, resolve the issue and reschedule a	

Possible Return Codes	Condition	Recommendation	Other Instructions
	Sample Message: Unable to load all of the transactions into the Transaction Catalog.	new chain job.	
Non-Fatal Error (8)	N/A	This step does not issue this return code.	
Failed (12)	Parameter file not found.	Failure reason needs to be investigated before scheduling a new chain job.	
	Failed because of runtime exceptions for an unexpected situation.	Failure reason needs to be investigated before scheduling a new chain job.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated and a new job scheduled.	
System Failure (20)	Job is terminated because of database server or network issues	The reason for the System Failure needs to be investigated and a new job scheduled.	

2.2.18 CWWR Transaction Generation

Chain or Job Name	CWWR Transaction Generation
Recommended Frequency	On Demand
Single Instance Required	Single
Can be restarted?	No
Reports generated	No

Overview

The CWWR Transaction Generation batch job creates Auto Check Writer Warrant Reconciliation (CWWRA) transactions to reclassify warrant funds to cash for the warranted records (Transaction Code is *CW* or *CE*) that were reconciled by the Check Reconciliation process. The process selects Recently Reconciled records with Check Status as Warranted from the Check Reconciliation (CHREC) table.

The process uses the CW Accounting (CWA) transaction information (Transaction Department Code and Transaction ID) on the selected records to retrieve the Check Writer Accounting (CWA) transaction. The process will then perform a Copy Forward action to create the CWWRA (Check Writer Warrant Reconciliation Auto) transaction.

Once the CWWRA transactions are created the process will submit all transactions and will write all messages to the job logs.

Process Steps	Messages
1. Parameter Validation	• Validating Batch Parameters • Parameters are valid or invalid depending on the validation. If the parameter is invalid, the invalid value will be displayed in the log. • Parameter validation completed

2. Selection of Records	• If the selection returns 0 records, then the following message is issued:
	  • No eligible records found on the CHREC table.
	• If CHREC record is found but the corresponding CWA transaction is not available on the Transaction Catalog (for all selected records), then the following message is issued:
	  • Matching transaction is not found on the Transaction Catalog for the CHREC record with Transaction Dept <<Transaction Dept Cd>> and Transaction ID <<Transaction ID>>.
	• If the selected CWA transaction is not valid, (that is, the Transaction Type is not CH and the Transaction Sub Type is not CW), then the following message is issued:
	  • The Transaction ID: <<DOC_ID>> and Transaction Department: <<DOC_DEPT_CD>> is not valid transaction of type CH and

Process Steps	Messages
	transaction subtype as CW.
3. Process Records	• CWWRA Draft version not found. • Override could not be applied to Transaction No: <<DOC_CD>>-<<DOC_DEPT_CD>>-<<DOC_ID>>. • Transaction No: <<DOC_CD>>-<<DOC_DEPT_CD>>-<<DOC_ID>> could not be submitted. • Transaction No: <<DOC_CD>>-<<DOC_DEPT_CD>>-<<DOC_ID>> successfully submitted. • <<Progression Counter>> CHREC records are processed.

Major Input

- Check Reconciliation table (R_AP_CHK_RECON)
- CW Accounting Transaction (CWA Transaction)

Batch Parameters

Parameter	Description	Default Value
Transaction Code	The Transaction Code of the transaction that will be generated by the CWWR Transaction Generation process. Required. Should be valid (Transaction Type/ Sub Type as CH/ CWWRA) on Transaction Control (DCTRL) table.	CWWRA
Use Transaction Department and Unit from CWA Transaction	Use Transaction Department and Unit from CWA Transaction (Valid values - Y/N). Required. When this parameter is set to 'N' then the User has to provide Transaction Department and	Y

	Transaction Unit (if Transaction Unit Code Required = True on DCTRL for the Transaction Code). When set to 'Y', Transaction Department and Transaction Unit are inferred from corresponding CWA Transaction.	
Transaction Department Code	The Transaction Department used to create the CW Warrant Reconciliation transactions. Required if Use Transaction Department and Unit from CWA Transaction parameter is set to 'N'. Should be valid on the Department table.	No Default
Transaction Prefix	This will be the Prefix for the CW Warrant Reconciliation Transaction ID. Required. Value should be available on the ADNT table for the Transaction Code and Transaction Department Code for the current Fiscal Year.	No Default
Transaction Unit Code	The Transaction Unit used to create the CW Warrant Reconciliation transactions. Conditionally Required. Transaction Unit is a required field if 'Require Transaction Unit Code' flag is set to true for the transaction	No Default

	code on DCTRL table and the Use Transaction Department and Unit from CWA Transaction parameter is set to 'N'; otherwise, it is an optional field. If entered, it should be valid on the Unit table for the entered Transaction Department and current Fiscal Year.	
Apply Overrides	Apply override on Submit of the CWWRA transaction. Required; Valid values are False or True.	False
Override Level	Override level used in the Submit job step. Conditionally Required. If Apply Overrides is set to True, Override Level is required. The value must be an integer between 0 and 10.	No Default
Commit Block Size	Commit Block Size. Specifies the number of records after which the job should perform a commit. If the Commit Block parameter is blank or not an integer greater than 0, the system will set the Commit Block job parameter equal to 1000.	100

Major Output

- Auto Check Writer Warrant Reconciliation (CWWRA) Transaction.
- Check Reconciliation table (R_AP_CHK_RECON)

Job Return Code

The following table shows the potential job Return Codes for the CWWR Transaction generation job:

Return Code	Condition
Successful (1)	All of the selected records are processed successfully.
Warning (4)	No eligible records found. This could be because of the following reasons: • No eligible records found on the CHREC table
Non-Fatal Error (8)	Out of selected records from the CHREC table, at least for one record, the corresponding CWWRA transaction could not submit to final: • Transaction No: <<DOC_CD>>-<<DOC_DEPT_CD>>-<<DOC_ID>> could not be submitted. • Failed to commit the transaction.
Failed (12)	The job will fail under the following conditions: • Parameters are invalid • Matching transaction is not found on the Transaction Catalog for the CHREC record with Transaction Dept <<Transaction Dept Cd>> and Transaction ID <<Transaction ID>> • Run time exceptions for unexpected situations.
Terminated (16)	This return code is issued when the job is terminated by the user.
System Failure (20)	This return code is issued when the job is terminated because of database server or network issues.

Sort Criteria

The selected CHREC records are ordered by Transaction Code (DOC_CD), Transaction Department Code (DOC_DEPT_CD) and Transaction ID (DOC_ID).

Selection Criteria

The process selects all Check Reconciliation (R_AP_CHK_RECON) table records that meet the following selection criteria:

- Transaction Code is *CW* or *CE*, and

- Check Status is equal to *Warranted*, and
- Recently Reconciled Flag is *True*

Problem Resolution

The process cannot be restarted on failure; however, a new job should be scheduled after resolving the problems.

The following table shows the possible return codes and recommendations for each processing steps.

Parameter Validation:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful **Sample Message:** "Parameter validation is successfully completed"	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Required Parameters are not entered **Sample Message:** "Transaction Code cannot be blank."	Enter a valid Transaction Code with a Transaction Type of *CH* and a Sub-Type of *CWWRA*.	.
	Failed because of runtime exceptions for an unexpected situation.	The reason for the failure needs to be investigated before re-scheduling the job.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can be rescheduled.	
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can be rescheduled.	

Selection of Records:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	If the selection returns 0 records, then the following message will be issued: **Sample Message:** "No eligible records found on the CHREC table."	Verify CHREC records against selection criteria and setup records accordingly before re-scheduling the job.	
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	If CHREC record is found but the corresponding CWA transaction is not available on the Transaction Catalog (for all selected records), then the following message is issued: **Sample Message:** "Matching transaction is not found on the Transaction Catalog for the CHREC record with Transaction Dept <<Transaction Dept Cd>> and Transaction ID <<Transaction ID>>"	Ensure appropriate CWA transactions exist for further processing.	
	Failed because of runtime exceptions for an unexpected situation.	The reason for the failure needs to be investigated before scheduling a new job.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can be rescheduled.	
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can be rescheduled.	

Processing of Records:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All transactions were processed and submitted successfully.	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	At least one CWWRA transaction fails to submit to Final. **Sample Message:** "Transaction No: <<DOC_CD>>-<<DOC_DEPT_CD>>-<<DOC_ID>> could not be submitted."	The reason for the failure needs to be investigated before scheduling a new job.	
Failed (12)	If the CHREC record is found but the corresponding CWA transaction is not available on the Transaction Catalog (for all selected records), then the following message is issued: **Sample Message:** "Matching transaction is not found on the Transaction Catalog for the CHREC record with Transaction Dept <<Transaction Dept Cd>> and Transaction ID <<Transaction ID>>"	Ensure appropriate CW Accounting transactions exist for further processing.	
	Failed because of runtime exceptions for an unexpected situation.	The reason for the failure needs to be investigated before scheduling a new job.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job	

Possible Return Codes	Condition	Recommendation	Other Instructions
		can be rescheduled.	
System Failure (20)	When the job is terminated because of database server or network issues	Reason for the System Failure needs to be investigated. A new job can be scheduled.	

2.2.19 EFT Payment and Prenote NOC

Chain or Job Name	EFT Payment and Prenote NOC
Recommended Frequency	Nightly
Single Instance Required	Yes
Can be restarted?	No
Reports generated	No

Overview

This batch process supports the interface needs required for an external payroll system and the use of Check Writer functionality within Advantage. As Advantage Financial is the system of record for payment and other bank interactions (i.e. EFT status), information must be interfaced back to the external payroll system to keep EFT information there up-to-date. The EFT Payment and Prenote Notice of Changes (NOCs) batch job reads an input bank file and Check Writer Payment data to create a sequential text file for transmission to the external payroll system.

Please note that the output file created should be sent or stored in an alternative location as the next run of the process will overlay the file from the previous run.

Process Steps	Messages
Parameter Editing	• Run Started • A list of parameters will be displayed (followed by each associated value) • Parameters validated • Output File EFT_PYMT_PRENOTE_NOC.txt created • <n> records written to output file EFT_PYMT_PRENOTE_NOC.txt
File Creation	• Generating the output file • Output File EFT_PYMT_PRENOTE_NOC.txt created • <n> records written to output file EFT_PYMT_PRENOTE_NOC.txt

Restartability Information

This job does not support restartability. If the job fails for any reason, a new job should be scheduled after correcting the errors that caused the job to fail.

Major Input

Tables

- Check Writer Payment (CWPYMT) - R_AP_CW_PYMT

Files

- Returned_CW_ACH.txt

Batch Parameters

Parameter	Description	Default Value
Bank (BANK_CD)	Required bank code that must match the bank code inserted into the ACH file on the 5 line in place of the Company ID for processing	No default
Department (DEPT_CD)	Required department code for Check Writer Payment record selection.	No default
Input File Location (INPUT_FILE_LOC)	Required location where the input file will be retrieved.	$$AMSROOT$$/ExportImport
Input File Name (INPUT_FILE_NM)	Required name of the input bank file. If not entered it will default as Returned_CW_ACH.txt.	Returned_CW_ACH.txt
Output File Location (OUTPUT_FILE_LOC)	Required location where the output file will be saved.	$$AMSROOT$$/ExportImport
Output File Name (OUTPUT_FILE_NM)	Required output file name. If not entered it will default as EFT_PYMT_PRENOTE_NOC.txt	EFT_PYMT_PRENOTE_NOC.txt
Unit (UNIT_CD)	Required unit code for Check Writer Payment record selection.	No default

Major Output

EFT_PYMT_PRENOTE_NOC.txt

Job Return Code

The following table shows the potential return codes for the EFT Payment and Prenote NOC process.

Return Code	Condition
Successful (1)	A job completes successfully when it reads all records from the input file and writes it to the output file without any errors.
Warning (4)	Job ends with this return code under following the following conditions: • No records are written to the output file. • When the input file is empty. • When no matching Check Writer Payment records are found.

Return Code	Condition
Non-Fatal Error (8)	This job does not use this return code.
Failed (12)	Jobs will fail under the following conditions: • Parameters are invalid. • Run time exceptions for unexpected situations. • When input file is not found
Terminated (16)	This return code will be issued when the job is terminated by the user.
System Failure (20)	This return code will be issued when the job is terminated because of database server or network issues.

Selection Criteria/Processing Logic

If ACH Return/NOC file exists in the specified location, start reading the file line by line where record type (position 1) is ‘6’ or ‘7’.

Read Record Type ‘6’ (position 1) from the ACH Return/NOC file and select records with the Amount of ‘0000000000’ - positions 30-39. Skip if the amount <> ‘0000000000’ (i.e. EFT Returns).

Positions 30 – 39: Amount – It will always be 0000000000 for Prenotes. It will be 0000000000 for EFT payments with NOCs. It will NOT be 0000000000 for EFT payments with Returns.

Positions 40 – 54: 15-digit EFT Number - Use the 15-digit EFT Number on Record Type ‘6’ to match it against CHK_EFT_NO on R_AP_CW_PYMT table:

- R_AP_CW_PYMT record with PYMT_AM = 0, then the transaction is a Prenote.
- R_AP_CW_PYMT record with PYMT_AM > 0, then the transaction is an EFT payment.

For selected Record Type ‘6’, retrieve the following fields from respective Record Type ‘7’ (position 1):

a. Addenda Type Code – positions 2-3:

If the code is ‘98’, then associated Record Type ‘6’ is a NOC

If the code is ‘99’, then associated Record Type ‘6’ is a Return

b. Reason Code – positions 4-6

If the code starts with a ‘C’, then the Reason Code represents an ACH NOC Code

If the code starts with a ‘R’, then the Reason Code represents an ACH Return Code

c. Addenda Information – positions 36-79

Create the EFT_PYMT_PRENOTE_NOC.txt file.

Problem Resolution

The following table shows the possible return codes and recommendations for each processing step.

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Job ends successfully. A job completes successfully when it reads all records from the input file and writes it to the output file without any errors.	N/A	N/A
Warning (4)	Job ends with a return code of Warning under the following conditions: • When no records are written to the output file • When the file is empty. • When no matching records are found.	N/A	Review Returned_CW_ACH.txt to confirm records are present on file. Check the input has proper data
Non-Fatal Error (8)	This job will not return a Non-Fatal Error.	N/A	N/A
Failed (12)	Job ends with a return code of Failed under the following conditions: • Parameters are invalid. • Run time exceptions for unexpected situations. • When file is not found at the location	The reason for the job failure needs to be investigated before scheduling a new job.	Confirm parameters are valid See if the input file is located in the same location as mentioned in the job parameter.
Terminated (16)	Job ends with a return code of Terminated. This return code will be issued when the job is terminated by the user.	The reason for the termination needs to be investigated before scheduling a new job.	N/A
System Failure (20)	This return code will be issued when the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated before scheduling a new job.	N/A

2.2.20 Initiate CW Mass Cancellation

Chain or Job Name	Initiate CW Mass Cancellation
Recommended Frequency	On Demand
Single Instance Required	Single
Can be restarted?	No
Reports generated	Yes. CW Mass Cancellation Generation Register report and CW Mass Cancellation Exception report.

Overview

The Initiate CW Mass Cancellation batch job initiates the cancellation of Check Writer payments.

The process selects all active records from the CW Cancellation Parameter (CWCPA) table. For each selected record, Department, CW File ID, Bank Account Code and Check Number range is validated on the Check Reconciliation (CHREC) table. If a record is found on the CHREC with the Status of *Disbursed* or *Warranted*, then the process performs a validation against Check Writer Payment (CWPYMT) by matching Department, Unit (if provided), CW File ID and Check Number range to verify whether a record exists for that combination. If either of the above validations fails then the CW Cancellation Parameter record is skipped from further processing and added to the CW Mass Cancellation Exception report.

If selected records (check range) are valid on CHREC and CWPYMT, then corresponding CWPYMT records are updated to set the Cancellation Status to *Cancellation Requested* and the Cancellation Reason Code with the value from the selected CWCPA record.

The process also updates the CWCPA record to indicate it is processed, (that is, the Processed Date is populated with the APPCTRL date and the Active flag is set to false). All successfully processed records (check range) are listed in the CW Mass Cancellation Generation Register report.

The batch process generates two reports:

- CW Mass Cancellation Generation Register report
- CW Mass Cancellation Exception report

The CW Mass Cancellation Generation Register lists all of the successful records, (that is, records for which cancellation has been initiated). The CW Mass Cancellation Exception Report lists all of the exceptions.

The Initiate CW Mass Cancellation batch job performs following steps:

1. Selection of Records
2. Table Updates
3. Generate Report

Process Steps	Messages

1. Selection of Records	• If the selection returns 0 records, then the following message is issued:
	  • No Active records found for processing.
	• If CHREC record is not found for any (if range is provided) of the Check Numbers provided on CWCPA, then the following message is issued:
	  • No matching record found on the CHREC table for Department: <<CW_DEPT_CD>>, CW File ID: <<CW_FILE_ID>>, Bank Account: <<BANK_ACCT_CD>>, Check Number: <<CHK_NO>>.

2. Table Updates	• If the CWPYMT record is not found for any (if range is provided) of the Check Numbers provided on CWCPA, then the following message is issued: • No matching record found on the CW Payment table for Department: <<CW_DEPT_CD>>, Unit: <<CW_UNIT_CD>>, CW File ID: <<CW_FILE_ID>>, Check Number: <<CHK_NO>>. • **Note:** Unit is added only if available.
3. Generate Report	• Total exceptions for the CW File ID <<CW_FILE_ID>>: <<Exception_Count>> • Total Cancellations for the CW File ID <<CW_FILE_ID>>: <<Cancellation_Count>> • Total exceptions: <<total_exception_count>> • Total Cancellations: <<total_cancellation_count>>

Major Input

- CW Cancellation Parameter (R_AP_CWCAN_PARM)
- Check Reconciliation (R_AP_CHK_RECON)
- Check Writer Payment (R_AP_CW_PYMT)

Batch Parameters

Parameter	Description	Default Value
CLIENT_NM	Client Name for Report. Optional.	No Default

Major Output

- Check Writer Payment (R_AP_CW_PYMT)
- CW Cancellation Parameter (R_AP_CWCAN_PARM)

Job Return Code

The following table shows the potential job Return Codes for the Initiate CW Mass Cancellation job:

Return Code	Condition
Successful (1)	All of the selected records are processed successfully
Warning (4)	No eligible records found. This could be because of the following reasons: • No Active records found for processing
Non-Fatal Error (8)	N/A
Failed (12)	The job will fail under the following conditions: • Failed to commit the transaction • Run time exceptions for unexpected situations.
Terminated (16)	This Return Code is issued when the job is terminated by the user.
System Failure (20)	This return code is issued when the job is terminated because of database server or network issues.

Sort Criteria

The selected CW Cancellation Parameter records are ordered by CW File ID (CW_FILE_ID) and Department (CW_DEPT_CD).

Selection Criteria

The process selects all CW Cancellation Parameter (R_AP_CWCAN_PARM) table records that meet the following selection criteria:

- Active flag is selected and
- Processed Date is blank

Problem Resolution

The process cannot be restarted on failure; however, a new job should be scheduled after resolving the problems.

The following table shows the possible return codes and recommendations for each processing step.

Selection of Records:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful.	N/A	N/A
Warning (4)	If the selection returns 0 records, then the following message will be issued: **Sample Message:** "No Active records found for processing"	Verify CWCPA records against selection criteria and setup records accordingly before re-scheduling the job.	
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Failed because of runtime exceptions for an unexpected situation.	The reason for the failure needs to be investigated before re-scheduling the job.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can be rescheduled.	
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can be rescheduled.	

Table Updates:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All CWPYMT records were processed and updated successfully.	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Failed because of runtime exceptions for	The reason for the failure needs to be	

Possible Return Codes	Condition	Recommendation	Other Instructions
	an unexpected situation.	investigated before re-scheduling the job.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can be rescheduled.	
System Failure (20)	When the job is terminated because of database server or network issues	The reason for the System Failure needs to be investigated. The job can be rescheduled.	

Generate Reports:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Reports were generated successfully.	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Failed because of runtime exceptions for an unexpected situation.	The reason for the failure needs to be investigated before re-scheduling the job.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can be rescheduled.	
System Failure (20)	When the job is terminated because of database server or network issues	The reason for the System Failure needs to be investigated. The job can be rescheduled.	

2.3 Accounts Payable Report Processes

The Accounts Payable report run sheets included in this section are:

2.3.1 1099/1042-S Transaction Report

Description

The 1099/1042-S Transaction Report captures all 1099 transactions, including Journal Voucher transactions and Disbursement Cancellations, for 1099 Reportable Vendors.

When to Run

Yearly, or as needed.

Major Input

- Journal 1099 (JRNL_1099)
- Vendor/Customer (R_VEND_CUST)
- 1099 Reporting Information (R_TIN_1099_INFO)

Other Input

- 1099 Type of Income (R_TYP_OF_INC)
- 1099 Date and Transaction Parameters (R_1099_PARM)
- Tin Type (CVL_TIN_TYP)
- 1099 Return Types (CVL_RET_TYP)
- Object (R_OBJ)
- Sub Object (R_SOBJ)
- Balance Sheet Account (R_BSA)
- Sub Balance Sheet Account (R_SBSA)

Output

- 1099 Transactions by Vendor Report

Parameters

Batch Parameters

Job	Parameter	Description	Default Values
1099/1042-S Transaction Report	Calendar Year	Required Field. Enter the Calendar Year for which Vendor Income should be selected.	
	Report Type	Required Parameter. If not entered then defaulted to 1. Entry of a value specifies the type of report that is produced.	1 - Report will include all transactions 2 – Report will include

			1099 reportable transactions only 3 - Report will include 1099 non reportable transactions only
	Client Name for Report	Optional field. Entry of a value in this field specifies the name that will appear on report.	
	Event Types to Exclude (Optional, separate multiple values by commas)	Optional field. Entry of a value in this field will exclude the specified Event Type(s).	
	Report ID	Optional field. Entry of a value in this field specifies the ID that will appear on report.	
	Range From Date (Valid format is MM/DD/YYYY)	Optional field. Entry of a value in this field limits the report to transactions with a record date greater than or equal to the entered value.	
	Range To Date (Valid format is MM/DD/YYYY)	Optional field. Entry of a value in this field limits the report to transactions with a record date less than or equal to the entered value.	

Sort Sequence

- Vendor Customer Code (VEND_CUST_CD)
- Transaction Code (DOC_CD)
- Transaction ID (DOC_ID)
- Transaction Accounting line number (DOC_ACTG_LN_NO)
- Record Number (REC_NO)

Selection Criteria

- Select records from 1099 Journal table JRNL_1099 where
- Year component of Run_tmdt = Calendar Year as specified as batch parameter and
- If (Report Type is 1099 Reportable) then JRNL_1099 records with TIN with 1099 Reportable Flag for Vendor = True are selected.

- If (Report Type is non 1099 Reportable) then JRNL_1099 records with TIN with 1099 Reportable Flag for Vendor = false are selected.
- Check the 1099 reportability of the records.
- If the records are 1099 reportable then display on the report if not go to the next records.
- Categorize the transactions into four different 1099 Return Types of G, INT, MISC, and S and 1042 Return Type of S and display these for each Vendor.
- Compute the totals for each 1099 Return Type for each Vendor and display them.

Problem Resolution

No database restore is required. Correct the problem and rerun the job executing the program. No restoration of datasets or files from backups is required for this program.

2.3.2 1099 Miscellaneous Vendor Report

Description

The 1099 Miscellaneous Vendor Report details Miscellaneous Vendor records on the 1099 Journal that contain reportable income and no TIN information. This report will allow a site to Audit these transactions and manually process corrections to the 1099 Reported Income table to issue forms to these vendors if they choose to do so. The report consolidates miscellaneous vendor data by Type of Income, as well as by Vendor Legal Name and Address. Both Legal Name and Address fields are required to prevent two miscellaneous vendors with the same name from being reported together. This report does not take 1099 Type of Income Thresholds into account. Instead this report will show any miscellaneous vendor income that is reportable, regardless of threshold. Transactions for the form types 1099-MISC, 1099-INT, 1099-G, 1099-S and 1099-NEC are reported in this report.

When to Run

This report should be produced when the 1099 Process is run, regardless of the processing mode.

Major Input

Primary input: 1099 Journal

Other Input

- COA tables
  - Object OBJ_CD
  - Sub Object SOBJ_CD
  - Balance Sheet Account BSA_CD
  - Sub Balance Sheet Account SBSA_CD
- Type of Income table (TINC)
- Date and Transaction Parameters table (1099D)
- Income Code table (INCM)
- Exemption Code table (EXMP)
- Recipient Code table (RECP)
- Chapter 4 Status Code table (CHP4)

Output Fields

- Type of Return
- Legal Name
- Address (including all Address, City, State and Zip)
- Income Amounts for Box 1 through 18 depending on the Form Type.

Parameters

Batch Parameters

Job	Parameter	Description	Default Values
1099 Miscellaneous Vendor Report	7_EXCL_EVNT_TYP	Added 7_EXCL_EVNT_TYP parameter to 1099 reports.	

Sort Criteria

- Sort Order
  - Type of Return, Vendor Legal Name
- Break Totals
  - Totals by Boxes 1-14 within Type of Return

Selection Criteria

All miscellaneous vendor records in the 1099 Journal that contain a reportable income (as defined by the COA element and Type of Income table) or federal withholding (posting pair type E or K), no value in the TIN field and the record date is between or equal to the date range defined for the transaction code and calendar year on the 1099 Date and Transaction Parameters Setup (1099D)] OR [if no record for the calendar year exists on the 1099D and the record date is between or equal to January 1 and December 31 for the calendar year.

Problem Resolution

In case of a problem, and the job needs to be restarted.

2.3.3 AD Check Register

Description

This process generates a Disbursement Register, which is a report of all Automated Disbursements or Manual Disbursements transactions that have been successfully processed. This report does not display direct deposit payments, which are payments made using an electronic funds transfer (EFT) transaction.

When to Run

- On-demand
- At the end of each Automated Disbursements Run

Major Input

- Bank Account table
- AD and MD transactions

Output

Disbursement Register Report

Parameters

Job	Parameter	Description	Default Value
ADCheckRegister	Bank Account Code (BANK_ACCT_CD)	A required selection parameter for a bank account.	No Default
	Bank Name (BANK_NM)	An optional output parameter for when a Bank Name should be written to the report other than the value from the Bank reference page.	No Default
	Client Name for Report (CLIENT_NM)	An optional output parameter for a name appearing in the report header.	No Default

Job	Parameter	Description	Default Value
	Transaction Code (DOC_CD)	An optional selection parameter for a transaction code to limit the report output.	No Default
	Disbursement Type (1-Check, 2-Warrant, 4-EFT) (DISB_TYP)	An optional selection parameter to limit the report output.	No Default
	End Date (END_DT)	A required selection parameter to limit the report output to transactions with a Record Date equal to or less than a date. Please enter in the MM/DD/YYY format.	No Default
	Run ID (RUN_ID)	An optional parameter for the Run ID of a single AD Chain to limit the report output.	No Default
	Start Date (STRT_DT)	A required selection parameter to limit the report output to transactions with a Record Date equal to or greater than a date. Please enter in the MM/DD/YYY format.	No Default

Sort Sequence

- Bank Account
- Disbursement Type
- Transaction Type
- Check Number

Selection Criteria

Date of Record (DOC_REC_DT) is between start date (STRT_DT) and end date (END_DT), and

Check number is not null, and

Bank Account Code on the batch parameters equal the Bank Account Code (BANK_ACCT_CD = BANK_ACCT_CD), and

Transaction Type (DOC_TYP) = 'AD' or ‘MD’

Problem Resolution

No database restore is required. Correct the problem and rerun the job executing the program. No restoration of datasets or files from backups is required for this program.

2.3.4 AD Exception Report

Description

The AD Exception Report will list all the Disbursement Request Table (DRT) records which are on HOLD – due to rejection in the Automated Disbursement Process or during the Transaction Processing.

When to Run

The AD Exception Report can be run at any time.

Major Input

Disbursement Request table (R_AP_DISB_RQST)

Output

Report (AD Exception)

Parameters

Job	Parameter	Description	Default Value
AD Exception Report	Client Name for Report (CLIENT_NM)	Optional field. Entry of a value in this field specifies the name that will appear on report.	No Default

Sort Order/Break Totals

- Bank Account Code, Disbursement Type, Exception Code, Schedule Date & Vendor Number Control Totals
  Vendor Total: Sum of line amounts by Vendor
  Total by Exception Type: Sum of line amounts for each Exception Type
  Total by Bank Account: Sum of all Exception Types by Bank Account
  Report Total – Sum of all Exceptions
- Page Breaks
- Page Breaks by Bank Account

Selection Criteria

- The Disbursement Exception report will include records that failed during the most recent Disbursement process, and have a system Hold Flag as TRUE.

2.3.5 Backup Withholding Detail Report by Accounting Distribution

Description

The Withholding Detail by Accounting Distribution report provides detail information by accounting distribution for all withholding offsets within a specified date range. This information will be used as an organization's withholding offset activity audit trail.

When to Run

This process will be run on demand

Major Input

- 1099 Journal (JRNL_1099 table)
- 1099 Reporting Information (R_TIN_1099_INFO)
- Sub Balance Sheet Account (R_SBSA)
- Balance Sheet Account (R_BSA)
- Sub Object (R_SOBJ)
- Object (R_OBJ)
- Type of Income (R_TYP_OF_INC)

Output

Report (Backup Withholding Detail by Accounting Distribution)

Parameters

Job	Parameter	Description	Default Value
BkpWith By Actg Dist	Client Name for Report (CLIENT_NM)	Optional field. Entry of a value in this field specifies the name that will appear on report.	No Default
	Range From Date (Valid Date format is MM/DD/YYYY) (RANG_FRM_DT)	Required Field.	No Default
	Range To Date (Valid Date format is MM/DD/YYYY) (RANG_TO_DT)	Required Field.	No Default

Job	Parameter	Description	Default Value
	Report ID (REPORT_ID)	Optional field. Entry of a value in this field specifies the ID that will appear on report.	

Sort Criteria

- Cabinet (CAB_CD)
- Department (DEPT_CD)
- Division (DIV_CD)
- Government Branch (GOVT_BRN_CD)
- Unit (UNIT_CD)
- Taxpayer Identification Number (TIN table)

Selection Criteria

Transaction Date of Record is between the dates specified, and the Posting Pair is Type E.

Problem Resolution

No database restore is required. Correct the problem and rerun the job executing the program. No restoration of datasets or files from backups is required for this program.

2.3.6 Backup Withholding Detail Report by Tax Identification Number

Description

The Backup Withholding Detail by TIN report provides detail information by TIN and 1099 Form Type for all withholding offsets within a specified date range.

This Report will be used as a Vendor Backup Withholding Offset activity audit trail.

When to Run

This process will be run on demand

Major Input

- 1099 Journal (JRNL_1099 table)
- 1099 Reporting Information (R_TIN_1099_INFO)
- Sub Balance Sheet Account (R_SBSA)
- Balance Sheet Account (R_BSA)
- Sub Object (R_SOBJ)
- Object (R_OBJ)
- Type of Income (R_TYP_OF_INC)

Output

Report (Backup Withholding Summary Report by TIN)

Parameters

Job	Parameter	Description	Default Value
BkpWith By TIN	Client Name for Report (CLIENT_NM)	Optional field. Entry of a value in this field specifies the name that will appear on report.	No Default
	Range From Date (Valid Date format is MM/DD/YYYY) (RANG_FRM_DT)	Required Field.	No Default
	Range To Date (Valid Date format is MM/DD/YYYY) (RANG_TO_DT)	Required Field.	No Default

	Report ID (REPORT_ID)	Optional field. Entry of a value in this field specifies the ID that will appear on report.	

Sort Criteria

Taxpayer Identification Number

Selection Criteria

Transaction Date of Record is between the dates specified, and the Posting Pair is Type E.

Problem Resolution

No database restore is required. Correct the problem and rerun the job executing the program. No restoration of datasets or files from backups is required for this program.

2.3.7 Backup Withholding Summary Report by Reporting Period

When to Run

This process will be run on demand

Description

The Withholding Summary Report by Reporting Period provides summary information by date for all withholding offset amounts within a given date range. This report will be used for providing information relating to remittance schedule to (IRS) Internal Revenue Service.

Major Input

- 1099 Journal (JRNL_1099 table)
- 1099 Reporting Information (R_TIN_1099_INFO)
- Sub Balance Sheet Account (R_SBSA)
- Balance Sheet Account (R_BSA)
- Sub Object (R_SOBJ)
- Object (R_OBJ)
- Type of Income (R_TYP_OF_INC)

Output

Report (Backup Withholding Summary Report by Reporting Period)

Parameters

Job	Parameter	Description	Default Value
BkpWith By Rep Period	Client Name for Report (CLIENT_NM)	Optional field. Entry of a value in this field specifies the name that will appear on report.	No Default
	Range From Date (Valid Date format is MM/DD/YYYY) (RANG_FRM_DT)		No Default
	Range To Date (Valid Date format is MM/DD/YYYY) (RANG_TO_DT)		No Default

	Report ID (REPORT_ID)	Optional field. Entry of a value in this field specifies the ID that will appear on report.	

Sort Criteria

- Transaction Record Date (DOC_REC_DT)
- Income Type Number (INC_TYP_NO)

Selection Criteria

Transaction Date of Record is between the dates specified, and the Posting Pair is Type E.

Problem Resolution

No database restore is required. Correct the problem and rerun the job executing the program. No restoration of datasets or files from backups is required for this program.

2.3.8 Contract Withholding Detail Report By Accounting Distribution

Job Name	Contract Withholding Detail Report By Accounting Distribution
Recommended Frequency	On demand
Single Instance Required	Yes
Can be restarted?	No
Reports Generated	Yes

Overview

The Contract Withholding Detail Report by Accounting Distribution provides detailed information by accounting distribution for all Contract Withholding offset amounts within a specified date range. This information can be used as an organization's Contract Withholding offset activity audit trail.

When to Run

This process may be run on demand

Description

The Contract Withholding Detail Report by Accounting Distribution provides detailed information by accounting distribution for all Contract Withholding offset amounts within a given date range.

The report reads 1099 Journal records where the Transaction Date of Record is between the dates specified in the job parameters; Event Category is ADIS, MDIS, HRD, or DRCL; and the Posting Pair is Type K.

This information can be used as an organization's Contract Withholding offset activity audit trail. The report is sorted by Cabinet, Department, Division, Government Branch, Section, Unit, TIN, TIN Type, Vendor Code, Vendor Name, and Record Date.

Major Input

- JRNL_1099

Major Output

- Contract Withholding Detail Report by Accounting Distribution

Parameters

Batch Parameters

Parameter	Description	Default Values
Report ID (REPORT_ID)	Optional field. Entry of a value in this field specifies the ID that will appear on the report.	No Default
Client Name for Report (CLIENT_NM)	Optional field. Entry of a value in this field specifies the name that will appear on the report.	No Default
Range From Date (Valid format is MM/DD/YYYY) (RANG_FRM_DT)	Required field. Entry of a value in this field specifies the From Date range for record selection.	No Default
Range To Date (Valid format is MM/DD/YYYY) (RANG_TO_DT)	Required field. Entry of a value in this field specifies the To Date range for record selection.	No Default
COMMIT_SIZE	Optional field. Commit block size.	1000
RUN_MODE	Required non-overrideable field. The value in this field specifies the report to be generated by this process. This is necessary because the same underlying report process is responsible for all of the Contract Withholding reports. This parameter value must be set to “Accounting Distribution” to produce the Contract Withholding Detail Report by Accounting Distribution.	Accounting Distribution

Sort Criteria

- Cabinet (CAB_CD)
- Department (DEPT_CD)
- Division (DIV_CD)
- Government Branch (GOVT_BRN_CD)
- Section (SECT_CD)
- Unit (UNIT_CD)
- TIN (TIN)
- TIN Type (TIN_TYP)

- Vendor Code (VEND_CUST_CD)
- Vendor Name (LGL_NM)
- Transaction Record Date (DOC_REC_DT)

Selection Criteria

- Selection criteria for obtaining the eligible records from JRNL_1099:

  SELECT

  Records from JRNL_1099

  WHERE

  Transaction Date of Record is between the dates specified,

  Event Category is ADIS/MDIS/HRD/DRCL and the Posting Pair is Type K

  GROUP BY

  Cabinet, Department, Division, Government Branch, Section, Unit, TIN, TIN Type, Vendor Code, Vendor Name, Transaction Record Date, Transaction Code, Transaction Department Code and Transaction ID

  ORDER BY

  Cabinet, Department, Division, Government Branch, Section, Unit, TIN, TIN Type, Vendor Code, Vendor Name, Transaction Record Date

Process Steps	Messages
Parameter Validation	Validating Batch Parameters • If the parameter is invalid, the error statement will be logged.
Set Report Parameters	The name of the output file is set.
Generate Report	• Select the Reporting Period records for the Report as per the selection criteria. • Generate the Report for the records selected. • When the number of records reaches the Commit Size they are saved. If the commit action fails, the job return code is failed and the below message appears in the job logs. "Error while committing the transaction" • If no eligible records are found then the following message appears in the job logs.

Process Steps	Messages
	"No data to report." • This report gives the summarized record and consists of the following details: • The system writes the Organization section at the beginning of each page and when the Cabinet, Department, Division, Government Branch, Section, or Unit changes. • The system writes one report line per selected record with TIN, TIN Type, Vendor Number, Vendor Name, Record Date, Transaction Code, Transaction Dept, and Transaction ID and Offset Amount from the selected record. • It also displays the TIN Total, Unit Total, Section Total, Branch Total, Division Total, Department Total and Cabinet Total whenever the respective Organization fields are changed. • The report also displays total the Posting Amount of all selected records as the Grand Total on the last line of the report.

Problem Resolution

- It is a good practice to look at the log of the job even if the job has run successfully.

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Parameter validation was successful and the report was generated successfully.	N/A	N/A
Warning (4)	The job does not end with this return code.	N/A	N/A
Non-Fatal Error (8)	The job does not end with this return code.	N/A	N/A
Failed (12)	The job failed due to Fatal conditions.	In this step, the job can fail under the following conditions.	N/A

Possible Return Codes	Condition	Recommendation	Other Instructions
		1) Encounters any runtime exceptions 2) Invalid parameter found 3) The commit transaction of the job fails.	
Terminated (16)	The job is terminated manually by the user.	The reason for the termination needs to be investigated. Reschedule a new job after the reason for termination is resolved.	N/A
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. Reschedule a new job after the reason for termination is resolved.	N/A

2.3.9 Contract Withholding Detail Report By Tax Identification Number (TIN)

Job Name	Contract Withholding Detail Report By Tax Identification Number (TIN)
Recommended Frequency	On demand.
Single Instance Required	Yes
Can be restarted?	No
Reports Generated	Yes

Overview

The Contract Withholding Detail Report by Tax Identification Number (TIN) provides detailed information by TIN for all contract withholding offsets within a specified date range.

When to Run

This process may be run on demand.

Description

The Contract Withholding Detail Report by Tax Identification Number (TIN) provides detailed information by TIN for all contract withholding offsets within a specified date range.

The report reads 1099 Journal records where the Transaction Date of Record is between the dates specified; Event Category is ADIS, MDIS, HRD, or DRCL; and the Posting Pair is Type K.

This information may be used as a vendor Contract Withholding offset activity audit trail. The report is sorted by TIN, TIN Type, Vendor Code, Vendor Name, and Transaction Record Date.

Major Input

- JRNL_1099 (1099 Journal)
- R_VEND_CUST (VCUST - Vendor/Customer)
- R_TIN_1099_INFO (1099I – 1099 Reporting Information)

Major Output

- Contract Withholding Detail Report by Tax Identification Number (TIN)

Parameters

Batch Parameters

Parameter	Description	Default Values
Report ID (REPORT_ID)	Optional field. Entry of a value in this field specifies the ID that will appear on the report.	No Default
Client Name for Report (CLIENT_NM)	Optional field. Entry of a value in this field specifies the name that will appear on the report.	No Default
Range From Date (Valid format is MM/DD/YYYY) (RANG_FRM_DT)	Required field. Entry of a value in this field specifies the From Date range for record selection.	No Default
Range To Date (Valid format is MM/DD/YYYY) (RANG_TO_DT)	Required field. Entry of a value in this field specifies the To Date range for record selection.	No Default
COMMIT_SIZE	Optional field. Commit block size.	1000
RUN_MODE	Required non-overrideable field. The value in this field specifies the report to be generated by this process. This is necessary because the same underlying report process is responsible for all of the Contract Withholding reports. This parameter value must be set to “TIN” to produce the Contract Withholding Detail Report by Tax Identification Number (TIN).	TIN

Sort Criteria

- Taxpayer Identification Number (TIN)
- TIN Type (TIN_TYP)
- Vendor/Customer Code (VEND_CUST_CD)
- Vendor/Customer Name (LGL_NM)
- Transaction Record Date (DOC_REC_DT)

Selection Criteria

- Selection criteria for obtaining the eligible records from JRNL_1099:

  SELECT

  Records from JRNL_1099

  WHERE

  Transaction Date of Record is between the dates specified,

  Event Category is ADIS/MDIS/HRD/DRCL and the Posting Pair is Type K

  GROUP BY

  TIN, TIN Type, Vendor Code, Vendor Name, Transaction Record Date, TIN Name, Transaction Code, Transaction Department Code and Transaction ID

  ORDER BY

  TIN, TIN Type, Vendor Code, Vendor Name, Transaction Record Date

Process Steps	Messages
Parameter Validation	Validating Batch Parameters • If the parameter is invalid, the error statement will be logged.
Set Report Parameters	The name of the output file is set.
Generate Report	• Select the TIN records for the Report as per the selection criteria. • Generate the Report for the records selected. • When the number of records reaches the Commit Size they are saved. If the commit action fails, the job return code is failed and the below message appears in the job logs. "Error while committing the transaction" • If no eligible records are found then the following message appears in the job logs. "No data to report." • This report gives the summarized record and consists of the following details: • The report displays one report line per

	selected record and displays TIN, TIN Type, Vendor Number, Vendor Name, Record Date, Transaction Code, Transaction Dept, and Transaction ID and Offset Amount from the Posting Amount. • The report also displays total Posting Amount of all selected records as Total Offset Amount on the last line of the report.

Problem Resolution

- It is a good practice to look at the log of the job even if the job has run successfully.

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Parameter validation was successful and the report was generated successfully.	N/A	N/A
Warning (4)	The job does not end with this return code.	N/A	N/A
Non-Fatal Error (8)	The job does not end with this return code.	N/A	N/A
Failed (12)	The job failed due to Fatal conditions.	In this step, the job can fail under the following conditions. • Encounters any runtime exceptions • Invalid parameter found • The commit transaction of the job fails.	N/A
Terminated (16)	The job is terminated manually by the user.	The reason for the termination needs to be investigated. Reschedule a new job after the reason for the termination is resolved.	N/A
System Failure (20)	When the job is terminated because of database server or	The reason for the System Failure needs to be investigated.	N/A

Possible Return Codes	Condition	Recommendation	Other Instructions
	network issues.	Reschedule a new job after the reason for the termination is resolved.	

2.3.10 Contract Withholding Summary Report By Reporting Period

Job Name	Contract Withholding Summary Report By Reporting Period
Recommended Frequency	On demand.
Single Instance Required	Yes
Can be restarted?	No
Reports Generated	Yes

Overview

The Contract Withholding Summary Report by Reporting Period provides summary information, by date, for all Contract Withholding offset amounts within a specified date range.

When to Run

This process may be run on demand.

Description

The Contract Withholding Summary Report by Reporting Period provides summary information by date for all Contract Withholding offset amounts within a given date range.

The report reads 1099 Journal records where the Transaction Date of Record is between the dates specified in the job parameters; and Event Category is ADIS, MDIS, HRD, or DRCL; and the Posting Pair is Type K.

This report is used for providing information relating to remittance schedule to (IRS) Internal Revenue Service. The report is sorted by Transaction Record Date.

Major Input

- JRNL_1099

Major Output

- Contract Withholding Summary Report by Reporting Period

Parameters

Batch Parameters

Parameter	Description	Default Values
Report ID (REPORT_ID)	Optional field. Entry of a value in this field specifies the ID that	No Default

Parameter	Description	Default Values
	will appear on the report.	
Client Name for Report (CLIENT_NM)	Optional field. Entry of a value in this field specifies the name that will appear on the report.	No Default
Range From Date (Valid format is MM/DD/YYYY) (RANG_FRM_DT)	Required field. Entry of a value in this field specifies the From Date range for record selection.	No Default
Range To Date (Valid format is MM/DD/YYYY) (RANG_TO_DT)	Required field. Entry of a value in this field specifies the To Date range for record selection.	No Default
COMMIT_SIZE	Optional field. Commit block size.	1000
RUN_MODE	Required non-overrideable field. The value in this field specifies the report to be generated by this process. This is necessary because the same underlying report process is responsible for all of the Contract Withholding reports. This parameter value must be set to “Reporting Period” to produce the Contract Withholding Summary Report by Reporting Period.	Reporting Period

Sort Criteria

- Transaction Record Date (DOC_REC_DT)

Selection Criteria

- Selection criteria for obtaining the eligible records from is JRNL_1099:

  SELECT

  Records from JRNL_1099

  WHERE

  Transaction Date of Record is between the dates specified,

  Event Category is ADIS/MDIS/HRD/DRCL and the Posting Pair is Type K.

  GROUP BY

  Transaction Record Date

  ORDER BY

  Transaction Record Date

Process Steps	Messages
Parameter Validation	Validating Batch Parameters • If the parameter is invalid, the error statement will be logged.
Set Report Parameters	The name of the output file is set.
Generate Report	• Select the Reporting Period records for Report as per the selection criteria. • Generate the Report for the records selected. • When the number of records reaches the Commit Size they are saved. If the commit action fails, the job return code is failed and the below message appears in the job logs. "Error while committing the transaction" • If no eligible records are found then the following message appears in the job logs. "No data to report." • This report gives the summarized record. The report line consists of the following fields: • Record Date from summarized record in MM-DD-YYYY format. • Summarized Posting Amount for the Record Date. • The report also displays the total Posting Amount of all selected records as Summary Offset on the last line of the report.

Problem Resolution

- It is a good practice to look at the log of the job even if the job has run successfully.

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Parameter validation was successful and the report was generated successfully.	N/A	N/A
Warning (4)	The job does not end with this return code.	N/A	N/A
Non-Fatal Error (8)	The job does not end with this return code.	N/A	N/A
Failed (12)	The job failed due to Fatal conditions.	In this step, the job can fail under the following conditions. 1) Encounters any runtime exceptions 2) Invalid parameter found 3) The commit transaction of the job fails.	N/A
Terminated (16)	The job is terminated manually by the user.	The reason for the termination needs to be investigated. Reschedule a new job after the reason for termination is resolved.	N/A
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. Reschedule a new job after the reason for termination is resolved.	N/A

2.3.11 Discount Penalty Report

Description

This report lists all the disbursement transactions for a specified date range that were eligible for discounts, interest calculation, and penalties.

When to Run

On demand

At the end of the Disbursement Run

Major Input

- AD Transaction Vendor Line (AD_DOC_VEND)
- AD Transaction Accounting Line (AD_DOC_ACTG)

Major Output

Discount, Interest and Penalty Report

Parameters

Job	Parameter	Description	Default Value
Discount/Penalty	Client Name for Report (CLIENT_NM)	Optional field. Entry of a value in this field specifies the name that will appear on report.	No Default
	End Date(Valid format is MM/DD/YYYY) (END_DT)	Required Field. Defines the end range of the report.	No Default
	Start Date Valid format is MM/DD/YYYY) (STRT_DT)	Required Field. Defines the beginning range of the report.	No Default

Sort Sequence

- Vendor/Customer (VEND_CUST_CD)
- Referenced Transaction Code (RFED_DOC_CD)
- Referenced Transaction Department (RFED_DOC_DEPT_CD)
- Referenced Transaction Identifier (RFED_DOC_ID)

Selection Criteria

Select all the AD transactions where the Transaction Creation Date is within the date range specified in the parameter.

At least one amount from the AD Transaction Accounting Lines (discount, interest, and penalty) should be non-zero.

Table Updates

None.

Troubleshooting

No database restore is required. Correct the problem and rerun the job executing the program. No restoration of datasets or files from backups is required for this program.

2.3.12 Non-Reconciled Procurement Card Transaction by Status

Description

The Non-reconciled Procurement Card Transaction by Status report displays procurement card transaction activities that have not been reconciled with a cardholder receipt.

When to Run

On-demand or monthly

Major Input

R_AP_PD_PRCU_TRAN table

Output

Report (Non- Reconciled Procurement Card Transaction by Status)

Parameters

Job	Parameter	Description	Default Value
Non-Reconciled Pcard	Client Name for Report (CLIENT_NM)	Optional field. Entry of a value in this field specifies the name that will appear on report.	No Default
	From Date (MM/DD/YYYY) (FROM_DT)	Required Field. Defines the beginning range of the report.	No Default
	To Date (MM/DD/YYYY) (TO_DT)	Required Field. Defines the end range of the report.	No Default
	Administration ID (ADMR_ID)	Optional Field. If entered, the record exists on the Procurement Administration table with the Administrator ID entered in the batch parameters.	No Default

Sort Sequence

- Administrator ID (ADMR_ID)

- Reconciliation Status (RECON_STA)

Selection Criteria

All transactions on Paid Procurement Card Transaction table that do NOT have a Reconciliation Status = ‘Reconciled’.

If the Administrator ID is entered, the system shall select records with a non reconciled status for the report from the Paid Procurement Card Transaction table based on the Administrator ID entered. If the value is blank, the system shall select all records meeting the criteria specified on the batch parameters.

Problem Resolution

No database restore is required. Correct the problem and rerun the job executing the program. No restoration of datasets or files from backups is required for this program.

2.3.13 Retainage Suspect After Encumbrance Correction

When to Run

On-demand

Description

The Retainage Suspect After Encumbrance Correction report exists because the chart of accounts elements (COA) on the accounting line of a purchase order move to a referencing payment request, then to a disbursement transaction, where retainage is recorded. If the COA, namely the Fund should have been different on the disbursement because the wrong purchase order was referenced, retainage will be recorded into the Retainage Liability account under a different Fund. That money liability will sit there until the retainage is paid out to the vendor or forfeited by the vendor.

Purchase orders with retainage terms create retainage table entries where subsequent payment requests and disbursements make updates to track retainage from being withheld until being returned or forfeited. When an encumbrance correction is done for an encumbrance commodity line that has retainage terms, there may be retainage table entries that are not as accurate as they could have been initially. As a result of this retainage issue, Retainage Suspect After Encumbrance Correction report is provided that will identify encumbrance corrections that may have had retainage impacts so that those users responsible for retainage management can be notified and act accordingly.

The job selects Final and Historical Final records in the Payment Request accounting line catalog that have the transaction code specified in the parameters. If a Start Date is specified, the job will select records in the Payment Request accounting line catalog with a Transaction Last Date value equal to or after the Start Date. The Transaction Last Date is the date the payment request accounting line went to Final status. If a Start Date is not specified, the job will select all Final and Historical Final records in the Payment Request accounting line catalog.

The job compares the referenced purchase order transaction information from these selected Payment request accounting lines to the Retainage Detail Table. If the referenced purchase order line that was corrected is listed in the retainage detail table, then that record is selected for the report.

Major Input

- Payment Request Accounting (PR_DOC_ACTG) table – Records selected are those with a transaction phase of Final (3) or Historical (5)
- Retainage Detail (R_RTG_DET) table

Output

Retainage Correction Report

Parameters

Batch Parameters

Description (Caption)	Parameter Name	Default Value
Client Name for report	CLIENT_NM	
Correction transaction code(s) - multiple values are to be separated by commas.	DOC_CD	CEC
Optional start date for record selection. Please enter as mm/dd/ccyy.	START_DT	

- Client Name for report. The name appears as the first line in the header of the report. This is an optional field with no default. The system wide default for Client Name will be used when established.
- Correction transaction code(s) is a required field and records the encumbrance correction transaction codes that will be selected for processing. More than one code can be entered as a parameter. The default value is Commodity Based Correction (CEC).
- Start Date is an optional field. When not supplied, all lines for the transaction code parameter that are Final and Historical Final will be selected. When supplied, only those lines for the transaction code parameter that are Final and Historical Final will be selected where the transaction last date is equal to or greater than the parameter. The transaction last date in such a case is what is commonly referred to as the acceptance date.

Sort Sequence

The report is ordered by PR Transaction code, Department, Transaction ID, Version, Vendor line number, Commodity line number, and Accounting line number.

Selection Criteria

The payment request accounting records with a transaction Phase of Final or Historical Final and with a transaction code as one of the transaction codes specified in the job parameter are selected. Additional selection is done if a Start Date is supplied; only records with a transaction last date equal to or after the Start Date are selected. Retainage Detail records that have referenced purchase order information matching the previously selected Payment Request accounting line's referenced transaction information is selected.

Problem Resolution

No database restore is required. Rerun the job.

2.3.14 Review Payment for Offset Report

Job Name	Review Payment for Offset Report
Recommended Frequency	On-demand or can be run prior to the Disbursement process.
Single Instance Required	Yes
Can be restarted?	No
Report Generated	Yes

Overview

The Review Payment for Offset report allows a site to review payments that may potentially be offset based on the active records on the Disbursement Parameters (DISPA) table. This report compares transactions on the Disbursement Request (DISRQ) table with records on the Intercept Request (INTR) table to determine if the DISRQ records are eligible for intercept based on the active records on the Disbursement Parameters (DISPA) table. The Review Payments for Offset report is run prior to the Disbursement process in order to determine the volume of payments that may be intercepted during the disbursement process. This report is sorted by Vendor and then by Bank Account.

There are three important assumptions to keep in mind when running this report:

The report is a match based on the Vendor on DISRQ having a TIN/TIN Type that matches a record on INTR. A record on this report is not an indication that the payment will be intercepted during the next disbursement run, only that it is eligible for intercept based on a matching record on INTR.

In order for the report to accurately reflect payments that might be intercepted, INTR must be updated daily.

The report does not consider the Intercept Disbursement Exception table.

Major Input

Disbursement Parameters (R_AP_DISB_PARM)

Disbursement Request (R_AP_DISB_RQST)

Intercept Request (R_AP_INCT_RQST)

Vendor/Customer (R_VEND_CUST)

Batch Parameters

Parameter	Description	Default Value
Client Name for Report (CLIENT_NM)	Optional field. Entry of a value in this field specifies the name that will appear on the report.	No Default

Vendor Code (VEND_CUST_CD)	Optional field. Entry of a value in this field specifies the Vendor(s) whose records will appear on the report. If left blank the report will be generated for all vendor codes that meet the selection criteria.	No Default

Major Output

- Review Payments for Offset report which shows the eligible payments that may be intercepted during the disbursement process.

Batch Job Return Codes

The following table shows the potential job return codes for the Review Payments for Offset Report.

Return Code	Condition
Successful (1)	All of the validations are performed successfully and report generation is successful.
Warning (4)	No valid/eligible records eligible for the report.
Non-Fatal Error (8)	Not Applicable for this job.
Failed (12)	13. Runtime exceptions encountered for any unexpected situations. 14. Technical failure
Terminated (16)	This return code will be issued when the job is terminated by the user.
System Failure (20)	This return code will be issued when the job is terminated because of database server or network issues.

Sort Sequence

The report is sorted by Vendor, then Bank Account.

Within the Payments Eligible for Debt Offset section, the selected records are sorted alphabetically by the Referenced Payment Transaction (Transaction Code, Transaction Department, Transaction ID and Vendor Line Number).

Within the Debts Associated with Vendor section, the selected records are sorted alphabetically by Associated Debt Record (Debt Transaction Code, Debt Transaction Department and Debt Transaction ID).

Selection Criteria

The Review Payments for Offset report job selects the payment lines from the Disbursement Request table based on the following selection criteria to form the Payment section of the report:

- The Active Flag for the DISPA record is *True*.
- The Scheduled Payment Date of the DISRQ record is between or equal to the beginning and end date of the DISPA record.
- The Transaction Department Code, Bank Account Code, Disbursement Type, Disbursement Category, and Disbursement Priority of the DISRQ record are equal to the Transaction Department Code, Bank Account Code, Disbursement Type, Disbursement Category and Disbursement Priority of the DISPA record.
- DISRQ records are not on HOLD.
- DISRQ records are not of the Miscellaneous Vendor Type.

The Debts section of the report selects the Intercept lines from the Intercept Request and Vendor Customer table based on the following selection criteria:

- The Vendor Code from the selected DISRQ is equal to the Vendor Code of VCUST.
- The TIN and TIN Type of INTR are equal to the TIN and TIN Type of VCUST.
- The Status of the INTR record is *True* and the Outstanding Debt Amount is greater than zero.

Problem Resolution

No database restore is required. If the job fails a new job should be scheduled after correcting the errors that caused the job to fail.

The following table shows the possible return codes and recommendations for each processing step.

Step 1: Batch Parameter Validation:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	N/A	This step does not issue this return code.	N/A
Warning (4)	N/A	This step does not issue this return code.	N/A
Non-Fatal Error (8)	N/A	This step does not issue this return code.	N/A
Failed (12)	The Vendor Code does not exist on VCUST or is a Miscellaneous Vendor.	Enter a Vendor which is valid on VCUST and is not a Miscellaneous Vendor.	N/A
	Failed because of runtime exceptions caused by unexpected conditions.	Failure reason needs to be investigated before rescheduling the job.	N/A

Terminated (16)	Job is terminated manually by the user.	A new job can be rescheduled.	N/A
System Failure (20)	Job is terminated because of database server or network issues.	Reason for the system failure needs to be investigated. A new job can be rescheduled.	N/A

Step 2: Report Generation:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	N/A	This step does not issue this return code.	N/A
Warning (4)	No valid/eligible records found for the report.	Ensure records that meet the report selection criteria are present.	N/A
Non-Fatal Error (8)	N/A	This step does not issue this return code.	N/A
Failed (12)	Failed because of runtime exceptions caused by unexpected conditions.	Failure reason needs to be investigated before rescheduling the job.	N/A
Terminated (16)	Job is terminated manually by the user.	A new job can be rescheduled.	N/A
System Failure (20)	Job is terminated because of database server or Network issues.	Reason for the system failure needs to be Investigated. A new job can be rescheduled.	N/A

2.3.15 Scheduled and Unscheduled Payment Report

Description

The Scheduled Payment report process lists the payments that will be picked up by Automated Disbursement process.

The Unscheduled Payment report process lists the payments that will not be picked up by Automated Disbursement process.

When to Run

Before the Automated Disbursement process

Major Input

- Disbursement Request table (R_AP_DISB_RQST)
- Disbursement Parameters (R_AP_DISB_PARM)
- Application Control (R_IN_APP_CTRL)

Output

Report (Schedule / Un-Schedule)

Parameters

Job	Parameter	Description	Default Value
Schedule/UnSchedule	Client Name for Report (CLIENT_NM)	Optional field. Entry of a value in this field specifies the name that will appear on report.	No Default
	Report ID (REPORT_ID)	Optional field. Entry of a value in this field specifies the ID that will appear on report.	No Default

Job	Parameter	Description	Default Value
	Type of Report (PARM_VL)	Optional field. Entry of a value specifies the type of report that is produced 1: Schedule Report 2: Unschedule Report 3: Both Blank on this field will produce a scheduled as well as an unscheduled report.	No Default

Sort Sequence

- Bank Account Code
- Disbursement Type
- Schedule Date
- Vendor Number
- Control Totals

  A) Vendor Total: Sum of line amounts by vendor

  B) Total by Type: Sum of line amounts for each Disbursement Type

  C) Total Scheduled: Sum of all Disbursement Types (EFT, Check, Warrant)
- Page Breaks

  Page Breaks by Bank Account

Selection Criteria

Scheduled Payment Report

Select records where:

Hold Status in the Disbursement Request table is not = 'User Request' or 'Cash Management', and

Records meet the criteria of each active entry in the Disbursement Parameter table for scheduled payment date range, bank account code, disbursement type, organizational element, disbursement category, priority, and disbursement limit.

Unscheduled Payment Report

Select records where:

Hold Status in the Disbursement Request table = ‘User Request’ or ‘Cash Management’, or

Records do not meet the criteria specified in active Disbursement Parameter entries.

Problem Resolution

No database restore is required. Correct the problem and rerun the job executing the program. No restoration of datasets or files from backups is required for this program.

2.3.16 Split Check Summary Report Process

Chain or Job Name	Split Check Summary Report
Recommended Frequency	On Demand
Single Instance Required	Yes
Can be restarted?	No
Reports generated	Yes. Split Check Summary Report.

Overview

The Split Check Summary Report performs the selection of all the eligible records from the Check Reconciliation table that need to be processed to generate the report.

The report is generated by populating the Bank Account, Check Date, Status, From Check Number, To Check Number and Count values from job parameters and the Check Reconciliation table.

The output generated by the job is the Split Summary Check Report.

Process Steps	Messages
Parameter Validation	• Validating Batch Parameters. • Parameters are valid or invalid depending on the validation. If the parameter is invalid, the invalid value is displayed in the log. • Parameter validation completed.
Selection of records	• Selecting the records for report generation. • Selection of records for generating report completed. • If the selection returns 0 records then the following message is issued: “No records selected for Report generation”.
Summary of records added on report.	• Total number of records processed successfully: <<count1>> • Total number of records failed in processing: <<count2>> • Number of Records added on Report: <<count1 – count2>>

Batch Parameters

Parameter	Description	Default Value
BANK_ACCT_CD	• Required. • Bank Account (single value)	N/A
COMMIT_BLOCK	• Optional • Commit Block Size	2000
PROG_CTR_SZ	• Optional • Progression Counter Size	5000
SEL_BLK_SIZE	• Optional • Select Block Size	5000

Major Input

- R_AP_CHK_RECON (Check Reconciliation table)

Major Output

- Split Check Summary Report.

Job Return code

The following table shows the potential return codes for the Split Check Summary Report batch job.

Return Code	Condition
Successful (1)	All the selected records are successfully added on report.
Warning (4)	No eligible records found.
Failed (12)	The job fails under the following conditions: • Parameters are invalid • Run time exceptions for unexpected situations. When this job ends with a return of code *Failed*, subsequent jobs in the chain will be set to *Inactive*.
Terminated (16)	This return code is issued when the job is terminated by the user.

Return Code	Condition
System Failure (20)	This return code is issued when the job is terminated because of database server or network issues.

Parameter Validation

In this step, the process verifies the parameters. If the parameter validation is successful, the job will go to the next step. Otherwise the job will end with a return code of *Failed*.

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	N/A	This step doesn't issue this return code.	This step doesn't issue this return code.
Non-Fatal Error (8)	N/A	This step doesn't issue this return code.	This step doesn't issue this return code.
Failed (12)	Required Parameters are not entered **Sample Message:** Bank Account cannot be blank.	Enter the Bank Account.	A new job can be scheduled after correcting the parameters.
	Failed because of runtime exceptions for an unexpected situation.	The reason for the failure needs to be investigated before scheduling the new job.	
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The new job can be scheduled.	
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The new job can be scheduled.	

Selection of Records

In this step based on the input parameter Bank Account, the job selects records from the input table R_AP_CHK_RECON.

- Based on the value entered in the Bank Account job parameter field, all of the transactions with an AD subtype and with the same Bank Account Code are selected together.
- In addition to the above, all the transactions with status of *Disbursed* and an Issue Date equal to *Application Control Date* or with status as *Void* and Cleared Date equal to *Application Control Date* are selected.
- The records are sorted based on the status and the check numbers.
- The number of checks between the From Check Number and the To Check Number determine the value of the Count.

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All the selected records are successfully added on report.	N/A	N/A
Warning (4)	No eligible records were found.	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Job failed due to fatal conditions, such as a runtime exception.	If the job fails due to a runtime exception, investigate the exception reported by the job, resolve the exception, and schedule a new run of the job.	Schedule a new job.
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. A new run of the job can be scheduled.	Schedule a new job.
System Failure (20)	When the job is terminated because of database server or network issues	The reason for the system failure needs to be investigated. A new run of the job can be scheduled.	Schedule a new job.

Write Records to Report

- A record is added on the report with Bank Account, Check Date (Application Control Date), Status, From Check Number, To Check Number and Count.
- The count is calculated as:
  - To Check Range Number - From Check Range Number + 1

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	All the selected records are added on report.	N/A	N/A
Warning (4)	No eligible records were found.	N/A	N/A
Non-Fatal Error (8)	Not all records are added on report successfully.	N/A	N/A
Failed (12)	Job failed due to Fatal conditions	If the job fails, due to a runtime exception, investigate the exception reported by the job, resolve the exception, and schedule a new run of the job.	Schedule a new job.
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. A new run of the job can be scheduled.	Schedule a new job.
System Failure (20)	When the job is terminated because of database server or network issues	The reason for the system failure needs to be investigated. A new run of the job can be scheduled.	Schedule a new job.

2.4 Check Writer Report Processes

The Check Writer Report Process in this section is:

- Check Writer Register Report

2.4.1 Check Writer Register Report

Job Name	Check Writer Register Process
Recommended Frequency	Daily as part of the nightly cycle or On Demand. Can be run On Demand for a small set of records. The Check Writer Register Report batch job should be run at least after the Check/EFT Generation process has been completed successfully.
Single Instance Required	Yes.
Can be restarted?	Yes.
Reports Generated	Yes.

Overview:

The purpose of the CW Register Report is to produce a register report for disbursed CW check and EFT payments.

The following table shows the various steps that the CW Check Register job goes through and the messages issued at each step.

Process Steps	Messages
1. Parameter validation	• Validating the batch parameters. • Parameter validation completed.
2. Selection of records	• A message will be written to the log every time the job finishes processing a CW record. The following messages are written to the Log: • Selecting eligible records from Header • Writing the records to the reports. • "N" CW records processed. • Department <Dept_No.> , Unit <Unit> , CW File ID <CW_File_ID> processing is complete. • Rendering report started. • Rendering report completed. Where "N" = no. of Records.

Restartability Information

This job makes no table updates, so it can be restarted without any recovery steps.

Major Input

- CWHDR (R_AP_CW_HDR)
- CWPYMT (R_AP_CW_PYMT)

Batch Parameters

Parameter	Description	Default Value
Client Name (CLIENT_NM)	Optional. Allows a user to specify the Client Name of the Report, which is displayed at the Header section of the report.	No Default
Commit Block Size (COMMIT_BLK)	Optional. Number of payment/posting lines grouped together for committing updates.	100
Check Writer File ID (CW_FILE_ID)	Optional. This field refers to the Check Writer File ID in CWHDR table. This is mainly used for selecting the record in CWHDR.	No Default
Department Code. (DEPT_CODE)	Optional. This field refers to the CW Department Code in CWHDR table. This is mainly used for selecting the record in CWHDR.	No Default
Check Writer End Date (END_DT , Valid format is MM/DD/YYYY)	Optional. Allows a user to filter the records by specifying the latest date for the system to select the CW records. The field is required if the CW Start Date is populated and the CW File ID parameter is left blank.	No Default
Check Writer Start Date (STRT_DT ,Valid format is MM/DD/YYYY)	Optional. Allows user to filter the records by specifying the earliest date for the system to select the CW records. The field is required if the CW File ID parameter is left blank.	No Default
Unit Code (UNIT_CD)	Optional. This field refers to the CW Unit code in the CWHDR table. This is mainly used for selecting the record in CWHDR.	No Default
Progression Message Block Size (PROG_CTR_SZ)	Required. This field provides the information about the record being processed.	No Default

Major Output

- Check Writer Register Report

Job Return Code

The following table shows the potential job return codes for Check Writer Check/EFT Generation batch job.

Return Code	Condition
Successful (1)	All of the selected payment records are processed successfully.
Warning (4)	• No records found in CWHDR for a given input. • No eligible records found.
Non-Fatal Error (8)	N/A
Failed (12)	The job fails under the following conditions: • Parameters are invalid. • No records that meet existing criteria were found. • No record could be found for a specified File ID. • Run time exceptions for unexpected situations.
Terminated (16)	This return code will be issued when the job is terminated by the user
System Failure(20)	This Return Code is issued when the job is terminated because of database server or network issues.

Sort Sequence

The records in the report are sorted by:

- Department Code
- Check Writer File ID

Selection Criteria

Records are selected for processing based on the following criteria:

- Department Code from CWHDR matches Department Code from Job Parameter (if entered)
- Unit Code from CWHDR matches Unit Code from Job Parameter or Unit Code from Job Parameter is blank
- CW File ID from CWHDR matches Department Code from Job Parameter (if entered)
- CW Run Status from CWHDR is 'Pending 1099 Posting' OR 'Processing Complete'
- Start Date(Job Parameter) <= Payment Date(CWHDR) <= End Date(Job Parameter)

Problem Resolution

- If the process fails for any reason, check whether the parameters entered are correct or not. The job log will also give some indication of the problem of the condition that could cause failure.

- If the job fails for any data setup reasons then correct the data setup and schedule a new job.

Step 1: Parameter validation:

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Successful	N/A	N/A
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	Required Parameters are not entered. **Sample Message:** CW End Date must be entered when CW Start Date is entered.	Enter the required parameters and reschedule the job.	N/A
	Entered Parameters are not valid. **Sample Message:** Invalid Check Writer End Date.	Enter the required parameters and reschedule the job.	N/A
	Entered Department Code, Check Writer File ID, and Unit Code does not exist in CWHDR.	Enter a valid Department Code, Check Writer File ID and Unit Code and reschedule the job.	N/A
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated before restarting the job. Alternatively, a new job can also be scheduled with the same parameters.	N/A
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated. The job can either be restarted or	

Possible Return Codes	Condition	Recommendation	Other Instructions
		rescheduled.	

Step 2: Selection of records:

This step will be performed only if the parameter validation is successful.

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful (1)	Issued when eligible records are selected from the CWHDR and CWPYMT.	N/A	N/A
Warning (4)	Issued when at least 1 eligible record is not selected from the CWHDR table.	Reschedule the job and specify new parameters so that the job finds eligible records.	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	In this step, the job can fail only because of runtime exceptions for an unexpected situation.	Failure reason needs to be investigated before scheduling a new job.	N/A
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can be rescheduled.	N/A
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated before scheduling a new job.	N/A

Step 3: Check Register Report Generation:

This step is performed only if eligible records are selected.

Possible Return Codes	Condition	Recommendation	Other Instructions
Successful	Issued if Report is	N/A	N/A

(1)	generated Successfully.		
Warning (4)	N/A	N/A	N/A
Non-Fatal Error (8)	N/A	N/A	N/A
Failed (12)	In this step, the job can fail only because of runtime exceptions for an unexpected situation.	Failure reason needs to be investigated before scheduling a new job.	N/A
Terminated (16)	Job is terminated manually by the user.	The reason for the termination needs to be investigated. The job can be rescheduled.	N/A
System Failure (20)	When the job is terminated because of database server or network issues.	The reason for the System Failure needs to be investigated before scheduling a new job.	N/A