This page intentionally left blank

Objectives

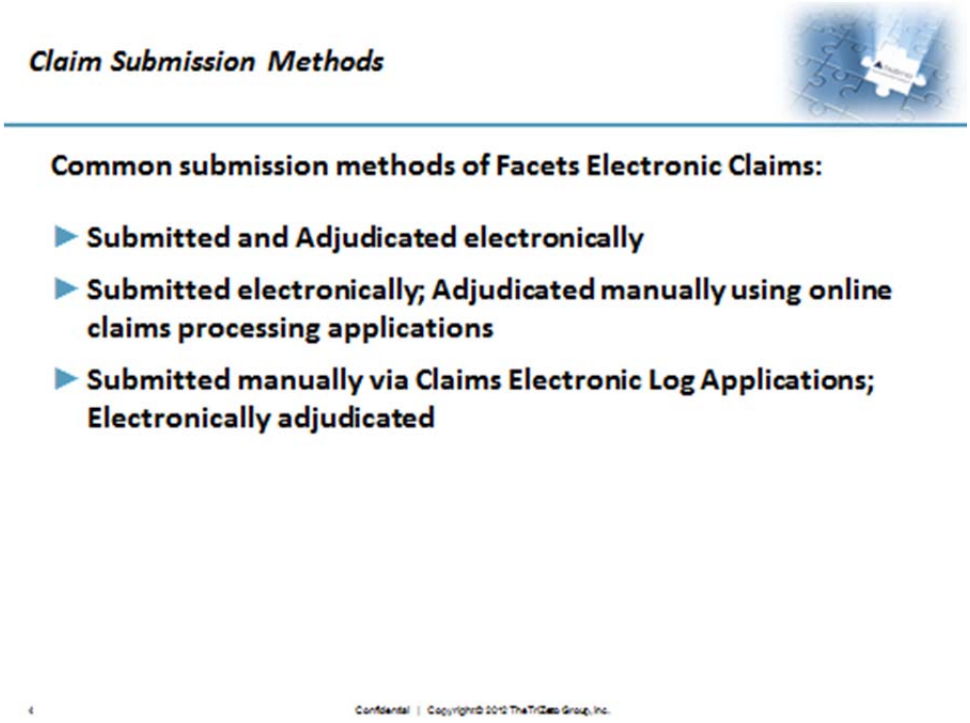
Upon successful completion of this chapter, you will be able to:

- Explain the purpose of the SYIN_INST ID
- Explain the purpose and structure of the XC database
- Explain the difference between electronic adjudication (CLMU) and claim payment batch processes (CKMM)
- Explain the workflow and relevant batch parameters for each run file/batch process:
  - Erccsrun837i
  - Ercmcrunxc00
  - Ercmcrunclmu

Upon successful completion of this chapter, you will be able to (continued):

- Explain the workflow and relevant batch parameters for each run file/batch process:
  - Ercmcrunckmm
  - Ercmcruncpc0

Claim Submission Methods

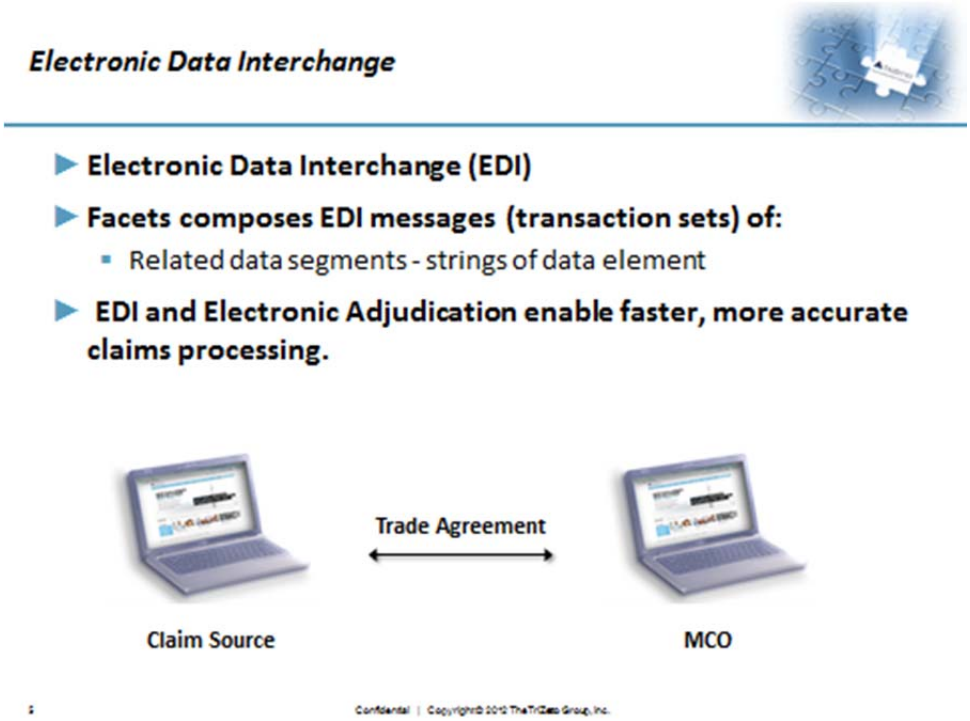


Claims submission and adjudication methods in Facets electronic claims processing include the following most common options:

- Electronically submitted and adjudicated
- Electronically submitted, but adjudicated by users in the online claims processing applications
- Manually submitted through the online Claims Electronic Log applications and electronically adjudicated

Ultimately, clients base their chosen method on their business needs.

Electronic Data Interchange

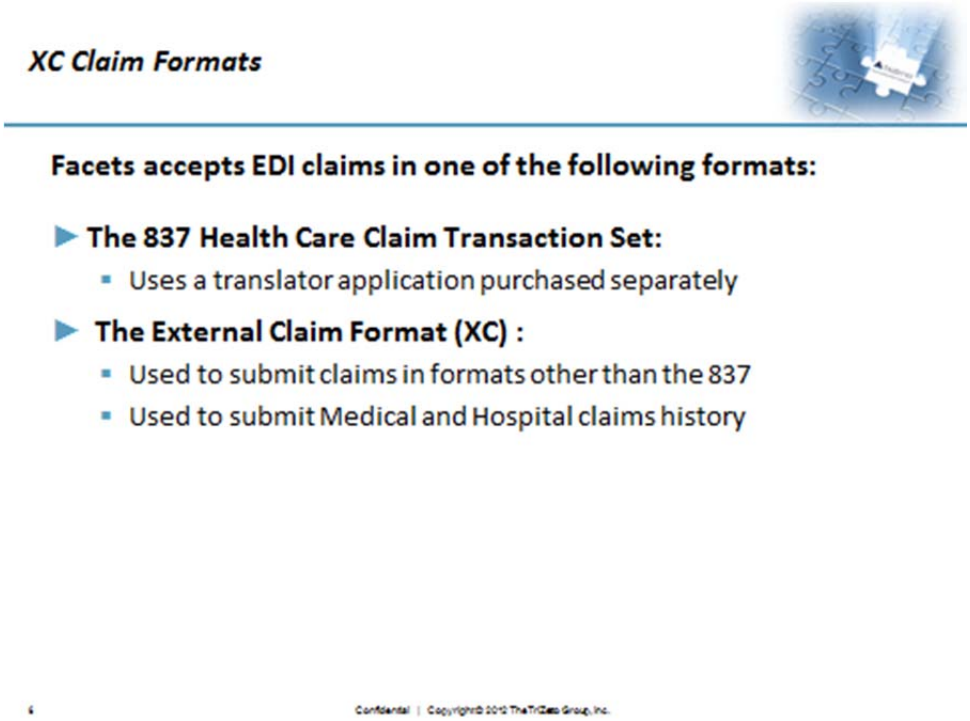


The industry standard format of Electronic Data Interchange (EDI) transmits electronic business data over networks.

An EDI message (called a transaction set) comprises of related data segments (strings of data elements) with each element separated by a delimiter. The data elements each represent a business parameter; i.e. a Member ID, Group, Provider ID, or Procedure Code.

EDI and Electronic Adjudication enable faster, more accurate claims processing.

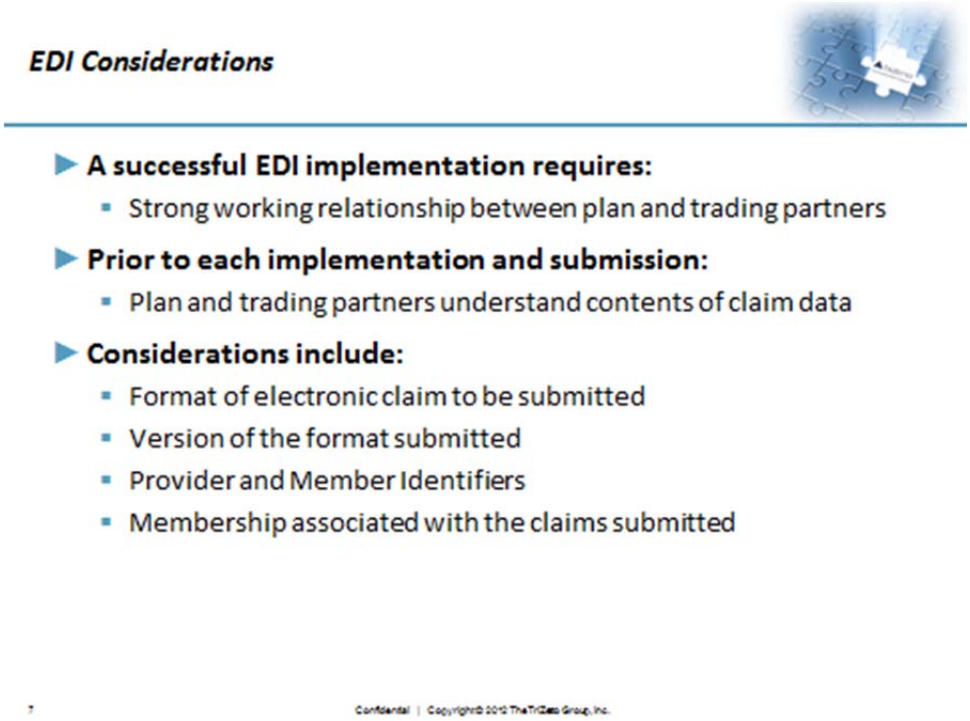
XC Claim Formats



Facets accepts EDI health care claims in one of the following formats:

- The American National Standards Institute (ANSI) charted the 837 Health Care Claim Transaction Set - The Accredited Standards Committee (ASC) X12 to develop uniform electronic data interchange standards. TriZetto uses the standard format developed for health care claims submission through electronic data interchange; i.e. the 837 Transaction Set. Facets accepts the 837 Transaction Set in version 4010A or 5010
- The 837 claims submission process uses a translator application clients must purchase separately
- TriZetto developed the External Claim Format (XC) - A proprietary External Claim (XC) or "True XC" format for Facets clients. Clients may use this to submit claims in formats other than the 837
- Clients may also use the XC when submitting Medical and Hospital claims history

EDI Considerations



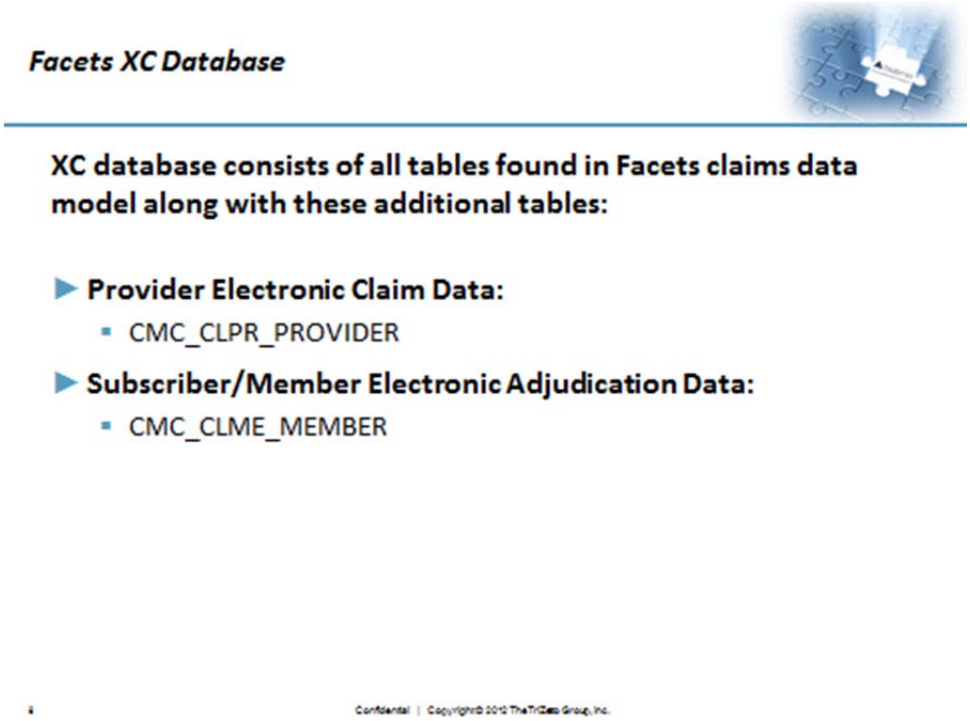
A successful EDI implementation requires the health plan to secure strong working relationships with their trading partners.

Prior to each implementation, the health plan and its trading partners must obtain a clear and common understanding of the contents of the claim data for submitting to Facets.

To obtain that understanding, some considerations include:

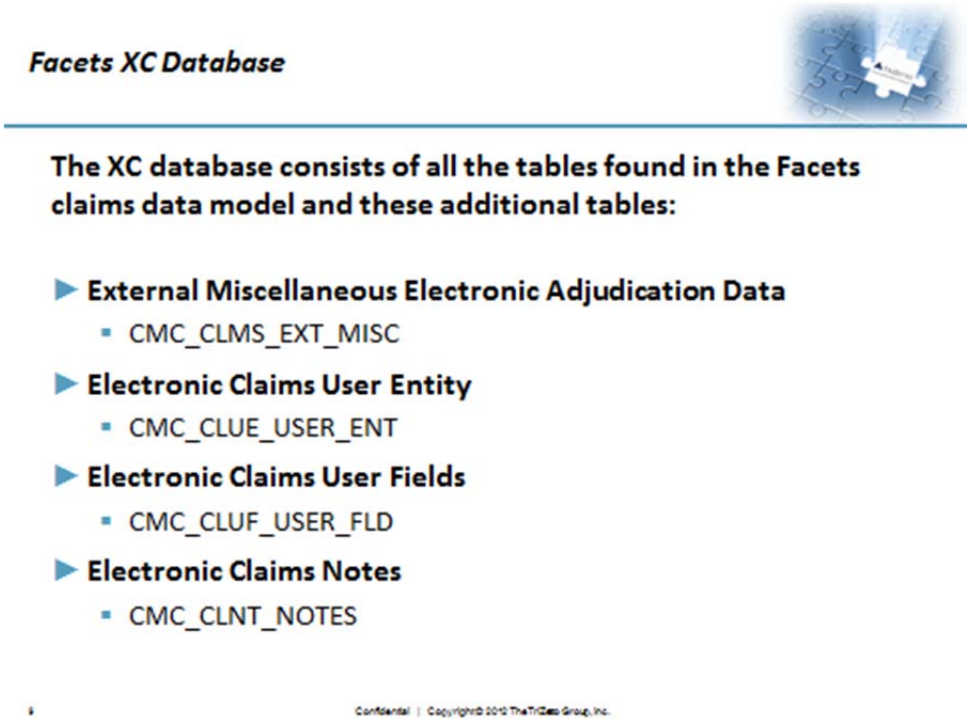
- The format of the electronic claim
  - The client must convert the claim to the Facets XC format if the 837 format does not support the original claim.
- The version of the format submitted
  - Facets contains several versions of the 837 format. Each version simply represents a different claim file layout or structure. TriZetto enables Facets to accept the 837 in version 4010A or 5010. The client must ensure submission to Facets of only the correct version of these claim formats.
- Provider and Member Identifiers
  - TriZetto recommends that clients attempt to receive electronically submitted claims containing a valid Facets Provider ID, Subscriber ID, Group ID, and Member Suffix.
- The membership associated with the claims submitted
  - In many cases, clients convert their membership data piecemeal to Facets by line of business. While Facets EDI contains the capability to validate member data submitted with the claims, clients hold the responsibility to ensure that claims submitted to Facets correspond with the converted lines of business.

Facets XC Database



The XC database consists of all the tables found in the Facets claims data model along with these additional tables:

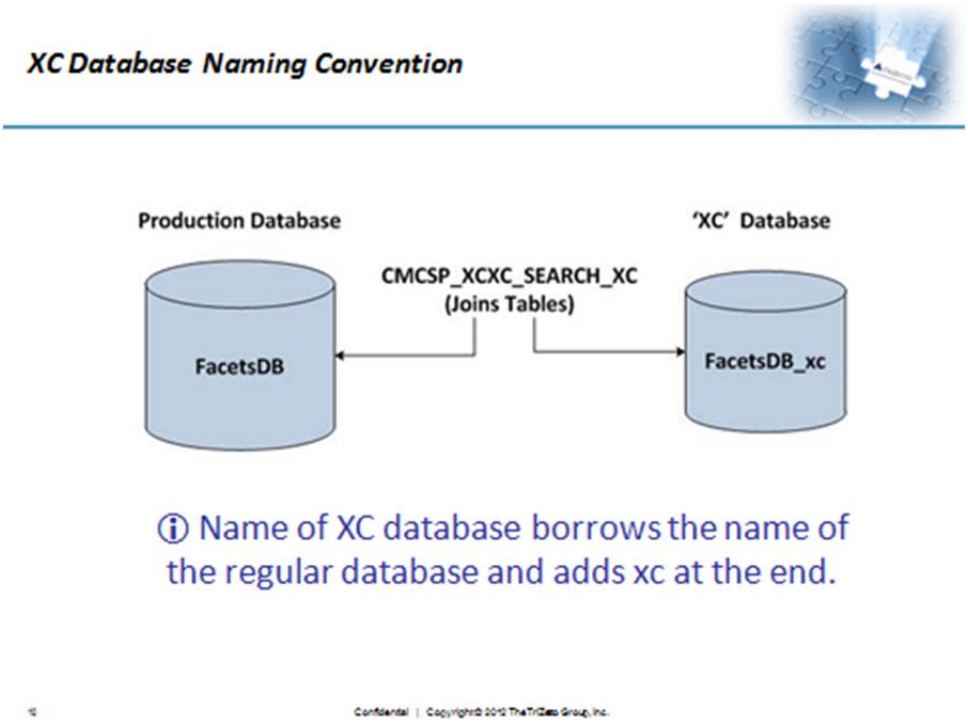
- The **Provider Electronic Claim Data** table (**CMC_CLPR_PROVIDER**) contains submitted provider data for all providers associated with a claim. Facets value-added routines optionally use the data stored in this table to find the service provider for claims submitted without a valid Facets Provider ID. Users may also view this data for error claims (status 15) in the External Claims Editing application. Facets only stores this data until identifying a provider for the claim and adjudicating the claim in Facets.
- The **Subscriber/Member Electronic Adjudication Data** table (**CMC_CLME_MEMBER**) contains all submitted data for the subscriber and member associated with a claim. Facets value-added routines optionally use the data stored in this table to find the member for claims submitted without a valid Facets Subscriber ID, Group ID and Member Suffix.



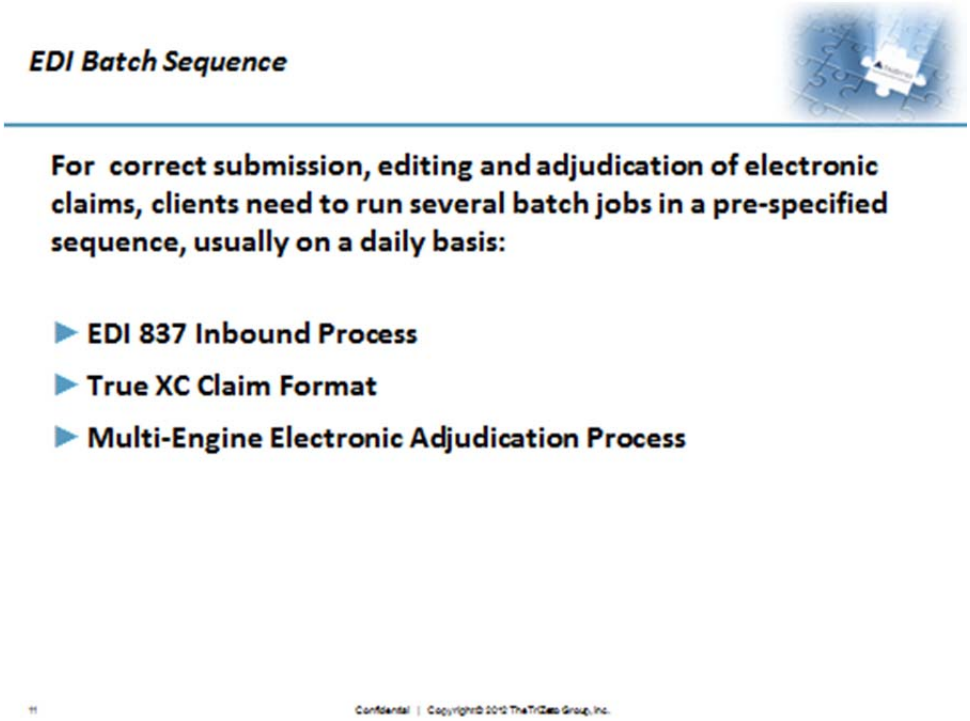
The XC database consists of all the tables found in the Facets claims data model along with these additional tables:

- The **External Miscellaneous Electronic Adjudication Data** table (**CMC_CLMS_EXT_MISC**) contains submitted miscellaneous information associated with the claim. During adjudication, Facets moves the **CLMS_MEMO** column to claims as the memo and **CLMS_EXT_REF** and **CLMS_TRAD_PARTNER** columns move to the **Claims EDI** table (**CMC_CLED_EDI_DATA**).
- The **Electronic Claims User Entity** table (**CMC_CLUE_USER_ENT**) contains user entity attachment data for electronic claims. Facets converts the data in this table to claim level contact attachments after loading XC claims.
- The **Electronic Claims User Fields** table (**CMC_CLUF_USER_FLD**) contains text, date, number, and money user field attachment data for electronic claims. Facets converts the data in this table to claim level user field attachments after loading XC claims.
- The **Electronic Claims Notes** table (**CMC_CLNT_NOTES**) contains claim note attachment data for electronic claims. Facets converts the data in this table to claim level note attachments after loading XC claims.

XC Database Naming Convention



EDI Batch Sequence



Facets requires clients to run several batch jobs in a pre-specified sequence, usually on a daily basis, in order to submit, edit and adjudicate electronic claims.

EDI 837 Inbound Process:

- Converts and loads ANSI X12 Standard 837 formatted Claims to the Facets XC database for processing. Only clients receiving 837 claims need to run this process.

True XC Claim Format:

- Converts and loads true XC Formatted Claims to the XC database for processing. Facets uses this for non-837 claims.

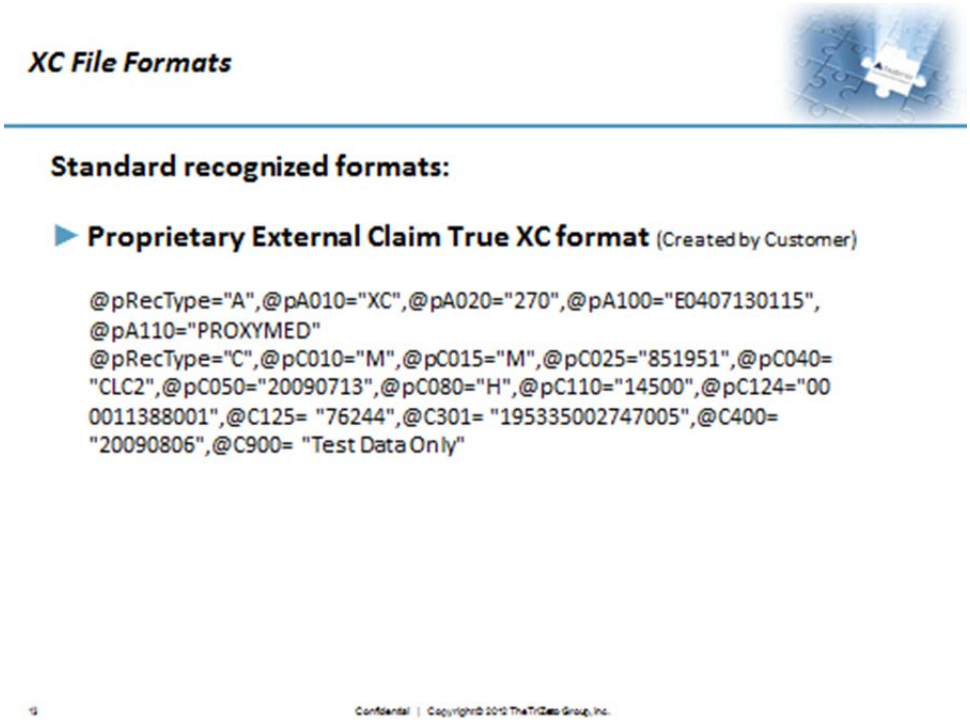
Multi-Engine Electronic Adjudication Process:

- This multi-engine enabled process (ErCmcRunClmu) executes an enhanced version of the electronic adjudication application for medical and hospital claims.

XC File Formats

Facets XC database accepts EDI Health Care Claims in one of two standard formats:

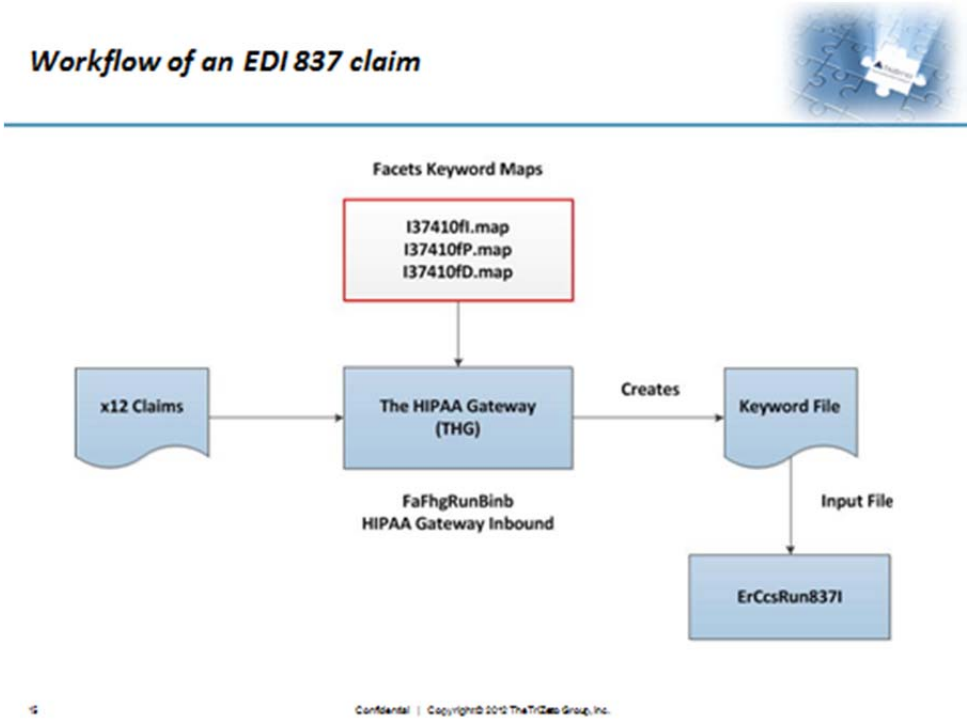
1. FACE_37IA_XXX_0.kwd format - Clients using the HIPAA Gateway to load claims use this format:
   - Face - The base system utilized (Facets)
   - 37IA - 837 Institutional Addenda
   - XXX - SYIN_INST from the batch run
   - 0 - Sequential number of the keyword file
   - Kwd - The extension that defines the keyword



Facets XC database accepts EDI Health Care Claims in one of these standard formats:

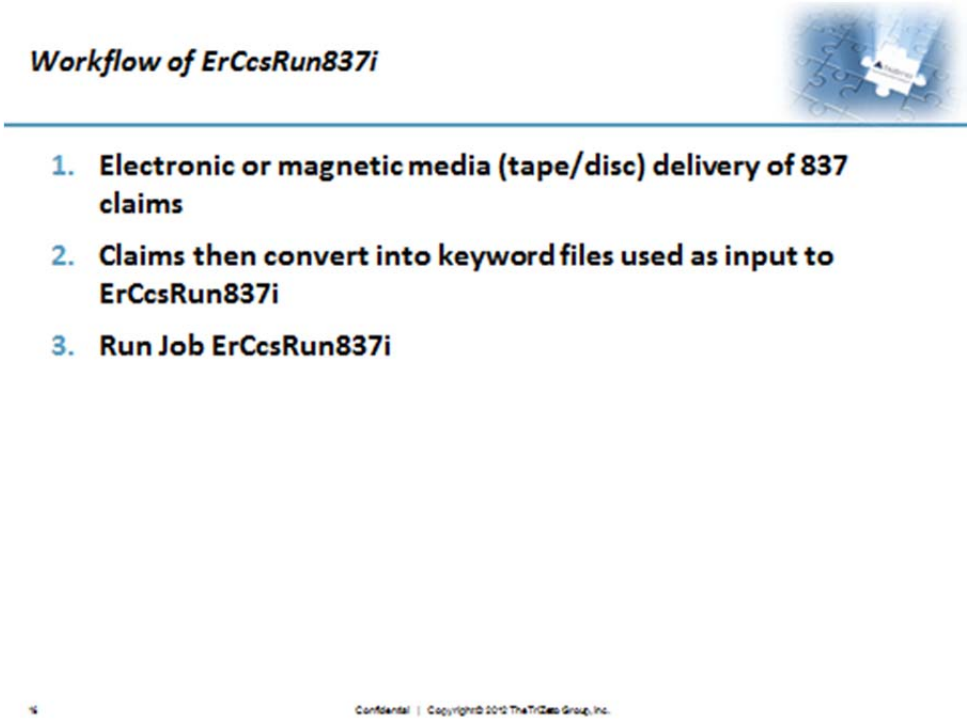
2. Proprietary External Claim True XC format - TriZetto developed this format for Facets clients who do not use the HIPAA Gateway, or who need to submit claims in a format other than the 837.

Workflow of an EDI 837 claim



Note: The HIPAA Gateway (THG) subsystem then reads through the x12 input file and runs either the I37410FI.Map for Institutional claims, the I37410FP.Map for professional claims, or the I37410FD Map for dental claims, and creates an 837 keyword formatted input file for the 837 inbound batch.

Workflow of ErCcsRun837i



Delivery of the 837 claims to the client occurs electronically or on magnetic media (tape or disk). The claims then convert into keyword file used as input to ErCcsRun837i.

Clients must then run Job ErCcsRun837i. This job executes a program coded in the translation software that converts and loads 837 formatted claims to an intermediate XC work file format identifiable by Facets. The intermediate file then loads to the Facets XC database tables.

The 837 process:

- Reads the file containing 837 formatted claims
- Writes a Control Split file used in the next step
- Reads the Control Split file created in the previous step, as well as the 837 formatted claims
- Maps the 837 fields to Facets fields
- Writes claims to an intermediate work file
- Reports on errors and run controls
- Reads the intermediate work file created in the previous step
- Optionally generates claim numbers
- Imports the intermediate workfile into the Facets XC (temporary) database
- Generates Facets Claims Acknowledgments (optionally, through the use of the CreateAcknowledgmentFile run file parameter)
- Reports on errors and run controls

Facets uses the Control Split file in cases where a claim located in an 837 transaction set contains more than one service provider or has dates that span a calendar year. Facets adjudication cannot accept claims containing either condition.

Note: The EDI 837 Inbound Process makes extensive use of .ini file options in lieu of run book parameters.

ErCcsRun837i Runbook Parameters

ErCcsRun837i Runbook Parameters

- **RunDate**
- **NumberOfEnginesClaim**
- **NumberOfEnginesFiles**
- **ClaimsPerEngine**
- **Received Date**

Parameter	Meaning
ReceivedDate	This optional parameter identifies a valid date in MM/DD/YYYY format that overrides the received date for all claims.
RunDate	This identifies the date in mm/dd/yyyy hh:mm:ss.000 format.
NumberOfEnginesClaim	This identifies the number of EDI processes starting simultaneously in the multi-engine job Step 5000. The procedure book defaults to 1.
NumberOfEnginesFiles	This identifies the number of processes that start simultaneously in the pre-processor Step 3000. This value equals, or contains fewer than, the number of files processed by the job. The procedure book defaults to 1.
ClaimsPerEngine	This identifies the number of claims processed by a single engine before proceeding to the next set of claims. It contains fewer than the total number of claims in the input file divided by the number of engines. The procedure book defaults to 1,000. Even if the number of claims exceeds 1,000, all claims enter into a single queue if this number is not specified.

ErCcsRun837i Runbook Parameters

- InputDir
- OutputDir
- HistoryDir
- DeleteDir
- AltIndexDir

Parameter	Meaning
InputDir	This required parameter identifies the directory where 837 Dental, Institutional, and/or Professional keyword files read.
OutputDir	This required parameter identifies the directory where acknowledgment (.ack), error (.err), and external (XC) files write.
HistoryDir	This required parameter identifies the directory where backup (.bak) 837 Dental, Institutional, and/or Professional keyword files write.
DeleteDir	This specifies the directory that stores output files containing rollback SQL that rollbacks 837i keyword files from the XC database.
AltIndexDir	This required parameter identifies the directory where Gateway Alternate Index (.alt) files write.

ErCcsRun837i Runbook Parameters

- InputFileExtension
- MultipleInputFiles
- NoInputFilesError
- InputFile
- DeadlockRetryAttempts

Parameter	Meaning
InputFileExtension	This identifies the extension of the 837 Dental, Institutional, or Professional keyword files for processing. It defaults to txt.
MultipleInputFiles	This identifies whether or not Facets processes multiple input files. Y indicates multiple input files process. N indicates no multiple file processing.
NoInputFilesError	The user uncomments, or sets the NoInputFilesError switch to Y to force a return code of 8 when processing multiple input files and no input files exist for the specified input file extension. It defaults to N, which returns a 0 return code.
InputFile	This identifies the name of the 837 Dental, Institutional, or Professional keyword file without the file extension. Facets uses this when MultipleInputFiles equal N.
DeadlockRetryAttempts	This indicates the desired number of retries attempted for a deadlocked transaction.

ErCcsRun837i Runbook Parameters

- DeadlockRetryWaitSeconds
- EnableConcurrentExecution
- MaxClaimDeadlocks
- ClclUserId
- CompressClclId

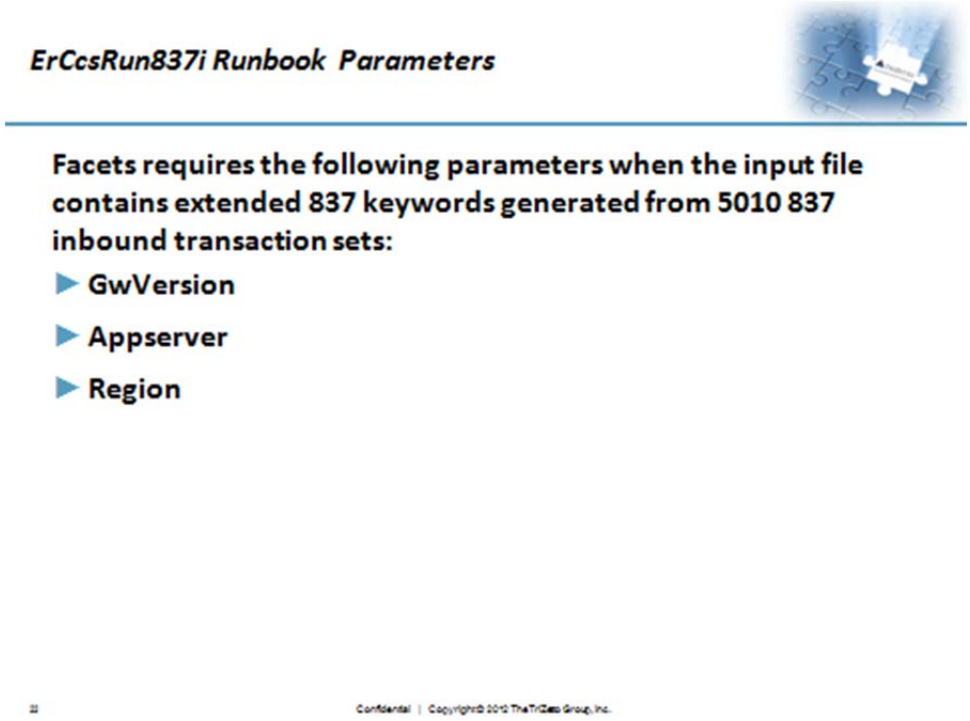
Parameter	Meaning
DeadlockRetryWaitSeconds	This specifies the number of seconds to wait between retry attempts. Facets uses the DeadlockRetryAttempts and DeadlockRetryWaitSeconds options to tune this application for optimal performance based on the client's specific environment and processing volumes. Setting these values inappropriately can adversely affect job performance. For example, by default the application makes three retry attempts to update a claim (if the transaction is deadlocked) by waiting 3 seconds between retries, making it a maximum of 9 seconds the application waits before bypassing the claim for the clean-up engine. Changing the value of DeadlockRetryAttempts to 6 and DeadlockRetryWaitSeconds to 10 raises the maximum retry time before bypassing to 70 seconds, which may not be desirable.

Parameter	Meaning
EnableConcurrentExecution	The user enters Y, allowing this job to run when other EDI 837 Inbound or True XC Claims jobs continue running. If runbook parameter MultipleInputFiles indicates Y, the InputDir should not indicate the same as the other jobs already running. This job can't restart after a fatal error when this switch sets.
MaxClaimDeadlocks	This identifies the number of deadlock errors needed to occur before the job terminates. By default, this job does not terminate when it encounters a deadlock error.
ClclUserId	This identifies the USUS_ID for the related KEYG numbering scheme. When running more than one engine, this value along with engine number, helps build the USUS_ID. If no USUS_ID passes, then Facets uses a USUS_ID of spaces for all engines. The first engine starts with the number 1.
CompressClclId	This option allows the Claim IDs generated from the **KEYG** table to compress to a 10 position Claim ID before appending the two position Segment ID. This occurs by converting the sequential number portion of the claim number from a base 10 to a larger base. This logic only occurs if KEYG_GEN_LEN indicates greater than 10.

ErCcsRun837i Runbook Parameters

- ExcludeVowels
- InsertOption
- CreateAcknowledgementFile
- HipaaRepositoryInstance
- HipaaBaseSystemInstance
- AmbulanceMileage
- MapObstetricAnesthesiaUnits

Parameter	Meaning
ExcludeVowels	Facets uses this entry in conjunction with the CompressClclId runbook option. This option only applies to CompressClclId values of 1 and 2. Entering N excludes uppercase letters I and O. Entering Y excludes uppercase letters A, E, I, O, U, and Y. It defaults to Y.
InsertOption	Entering Y inserts records to the database. Otherwise, entering N applies the records to the database. For example, if a Claim ID already exists on the XC database and the InsertOption is sets to N (apply records), the new claim loaded with the same Claim ID overwrites/updates the existing claim. If the InsertOption sets to Y (insert records), the new claim loaded errors as a duplicate claim. It defaults to N (apply the records to the database).
CreateAcknowledgementFile	The user enters M for Medical, D for Dental, or B for Both. It defaults to blank if no acknowledgement file gets created.
HipaaRepositoryInstance	The user enters the name of the repository instance. It defaults to Facets.
HipaaBaseSystemInstance	The user enters the base system instance. It defaults to Facets.
AmbulanceMileage	The user enters Y to assign Ambulance Mileage Units as an override for Ambulance Mileage Claims. It defaults to N.
MapObstetricAnesthesiaUnits	The user enters Y to convert the Obstetric Anesthesia Units into minutes. It defaults to N.



Facets requires the following parameters when the input file contains extended 837 keywords generated from the 5010 837 inbound transaction sets:

Parameter	Meaning
GwVersion	This identifies the gateway version (must equal 3 bytes and contain numeric characters only).
Appserver	This identifies the gateway server name used to install the gateway.
Region	This identifies the database name used to install the gateway.

ErCcsRun837i Runbook Parameters

- ProviderSearchConfigFile
- RFOtherProvEdit
- FAOtherProvEdit
- OtherProvEdit
- RendProvEdit
- SuppressSearchErrors
- SearchDebug
- WorkFlow

Parameter	Meaning
ProviderSearchConfigFile	The user optionally enters the full path and file name of the Provider Search Configuration file to activate the provider search.
RFOtherProvEdit	The user enters Y to edit the Referring Provider ID. The user enters N when not editing the Referring Provider ID. It defaults to N.
FAOtherProvEdit	The user enters Y to edit the Facility Provider ID. The user enters N when not editing the Facility Provider ID. It defaults to N.
OtherProvEdit	The user enters Y to edit the Admitting Provider (AD), Operating Provider (OP), Other Provider 1 (01), and Other Provider2 (02) IDs. The user enters N when not editing the Admitting Provider (AD), Operating Provider (OP), Other Provider1 (01), and Other Provider 2 (02) IDs. It defaults to N.

Parameter	Meaning
RendProvEdit	The user enters Y to validate the Rendering Provider ID at the line item level (CDSD) using the Tax ID and/or NPI. The user enters N when not validating the Rendering Provider ID at the line item level (CDSD). It defaults to N.
SuppressSearchErrors	The user enters Y when not writing search errors to the log. The user enters N when writing search errors to the log. It defaults to N.
SearchDebug	The user enters Y when writing the search debug information to the log file. The user enters N when not writing search debug information to the log file. It defaults to N.
WorkFlow	Facets uses this indicator in conjunction with the Member/Provider value added searches. When implementing workflow, set the switch to Y. This causes any claims, where member/provider keys don't resolve, to write to the XC database with a status 16, which pulls them into the adjudication/workflow process. The user should set this switch to N if not implementing value add searches and/or workflow.

Facets Acknowledgement Record File Layout

Facets Acknowledgement Record File Layout

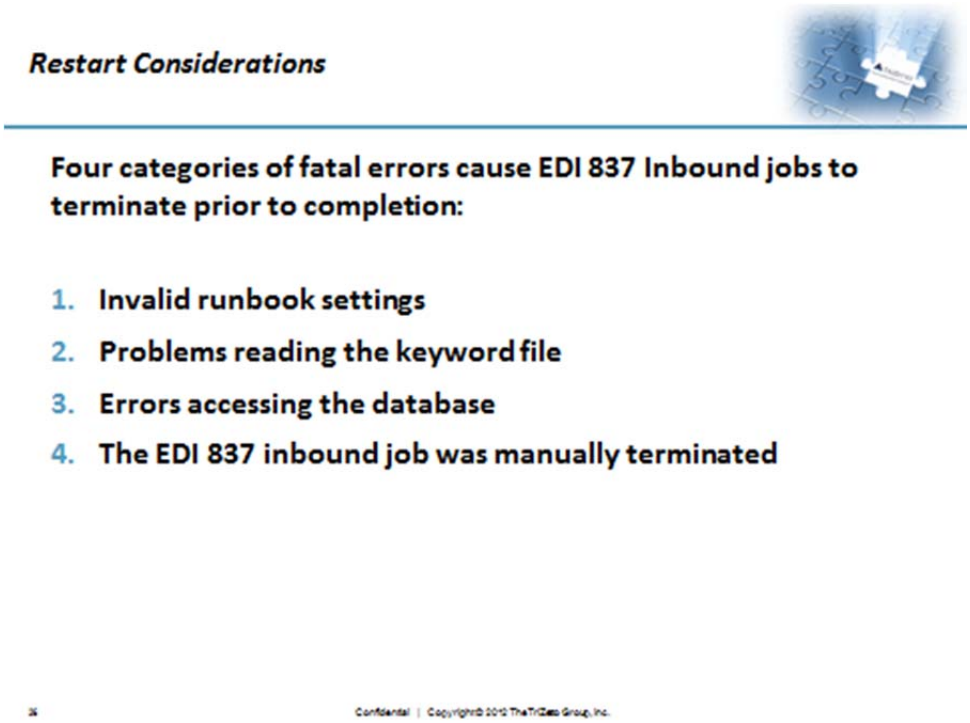
Field	Field Code	Size	Facets Field	Comments
Record Type	RecType	3	None	Move 'ACK'
Claim Number	CLCL_ID	12	CLCL_ID	Move value of CLCL_ID
Claim Acknowledge-ment Status	CLCL_ACK_STS	1	None	Move a value of 'A' (Accepted) if the claim is successfully applied to the XC database. Move a value of 'R' (Rejected) if the claim is not successfully applied to the XC database.
Claim Acknowledge-ment Date	CLCL_ACK_DT	8	None	Move current system date and time in CCYYMMDD format.
Claim Input Method	CLCL_INPUT_METH	1	CLCL_INPUT_METH	Move value of CLCL_INPUT_METH
Claim Type	CLCL_CL_TYPE	1	CLCL_CL_TYPE	Move value of CLCL_CL_TYPE
Claim Sub Type	CLCL_CL_SUB_TYPE	1	CLCL_CL_SUB_TYPE	Move value of CLCL_CL_SUB_TYPE
Claim External Refer-ence Number	CLMS_EXT_REF	15	CLMS_EXT_REF	Move value of CLMS_EXT_REF
Claim Trading Partner	CLMS_TRAD_PARTNER	15	CLMS_TRAD_PARTNER	Move value of CLCL_TRAD_PARTNER

Facets Alternate Index Record File Layout

Facets Alternate Index Record File Layout

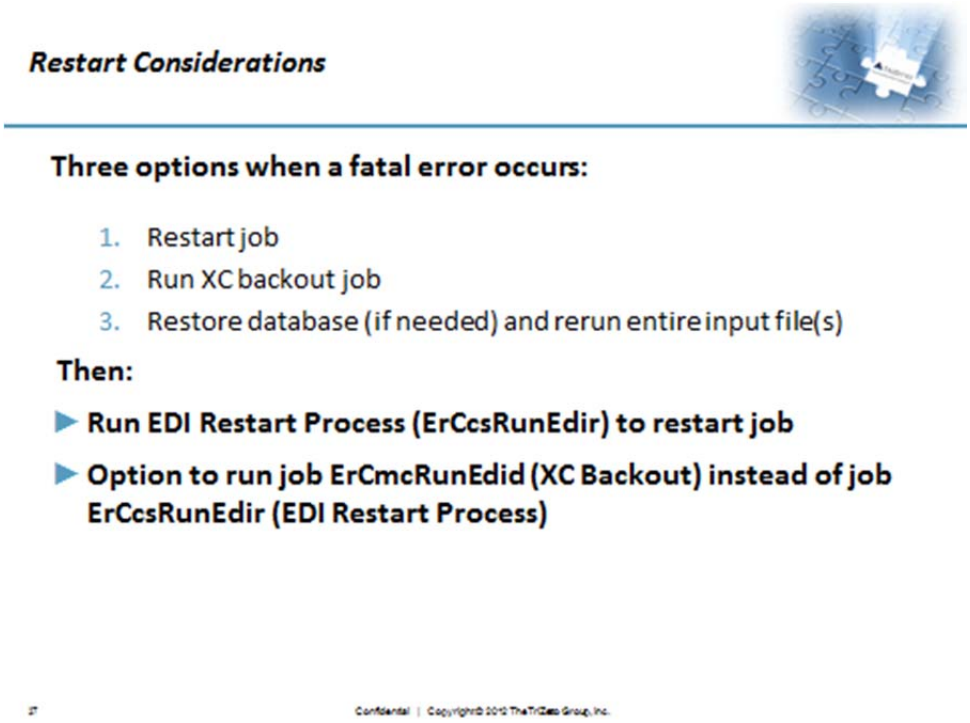
Field Code	Size	Facets Field	Comments
RecType	4	None	Move 'ALTX'
GWID	32	GWID	Move value XC keyword @pGWID
TRANSID	4	None	If the CLCL_CL_SUB_TYPE = 'M' 'Move 837P'. If the CLCL_CL_SUB_TYPE = 'D' 'Move 837D'. If the CLCL_CL_SUB_TYPE = 'H' 'Move 837I'.
INDEX_NAME	21	None	Move 'basePayerClaim-Number'
INDEX_DATA	12	CLCL_ID	Move value of CLCL_ID
HIPAA_REPOSITORY_INSTANCE (from run file parameter)		None	Move 'FACETS'
HIPAA_BASE_SYSTEM_INSTANCE (from run file parameter)		None	To be determined

Restart Considerations



Four main categories of fatal errors cause the EDI 837 Inbound job to terminate prior to completion:

3. Invalid runbook settings
4. Problems reading the keyword file. An invalid or missing record type in the keyword file causes the EDI 837 inbound to terminate.
5. Errors accessing the database. The EDI 837 Inbound terminates if an error occurs when writing to the database. The only write errors that won't cause this job to terminate contain errors caused by deadlocks.
6. The EDI 837 inbound job manually terminated. This job can manually terminate by executing the EDI Termination job (ErCcsRunEdit).



The three options available when a fatal error occurs include:

1. Restarting this job, processing the claims in the keyword file(s) that did not process. This is not an option when runbook parameter EnableConcurrentExecution equals Y.
2. Running the XC backout job, deleting the claims that wrote to the XC database, and clearing the queue rows.
3. Restoring the database (if necessary) and rerunning the entire input file(s).

To restart this job:

- First, run the EDI Restart Process (ErCcsRunEdir).
- Then run the EDI 837 Inbound job without making any changes to the runbook. The user should not use the BypassStep, RestartStep, or StopStep runbook parameters in an attempt to restart this job.
- It is always an option to run job ErCmcRunEdid (XC Backout) instead of running job ErCcsRunEdir (EDI Restart Process), as indicated, if the user wants to start ErCcsRun837i from the beginning after a fatal error.

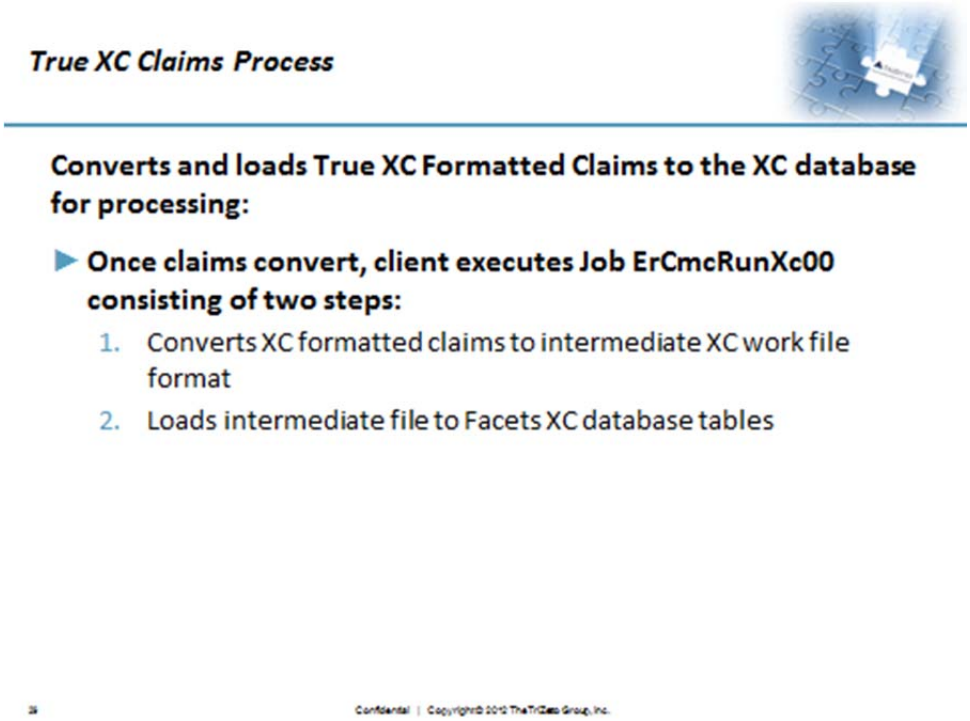
True XC

This process converts and loads True XC Formatted Claims to the XC database for processing.

This format submits all other types of claims using a keyword file format, and gets used to process M/H/D and historical claims load.

Note: Facets uses both jobs to populate the claim data in the XC database. If Facets finds no errors on the claim, the claim enters the XC database as a status 16. If the claim errors, it remains in the error file for re-submission after the correction occurs.

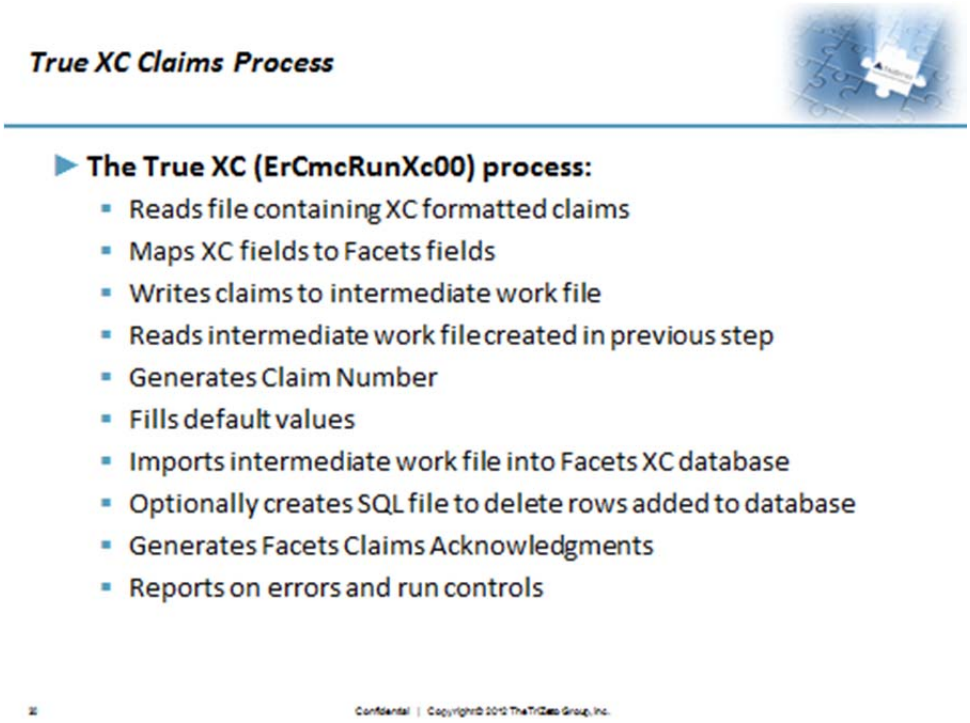
True XC Claims Process



This process converts and loads True XC formatted claims to the XC database for processing. The client must ensure that claims not in an 837 format convert to the True XC format.

Once the claims convert, the client must then execute Job ErCmcRunXc00.

This job consists of two steps that convert the XC formatted claims to an intermediate XC work file format identifiable by Facets, and then loads the intermediate file to the Facets XC database tables.

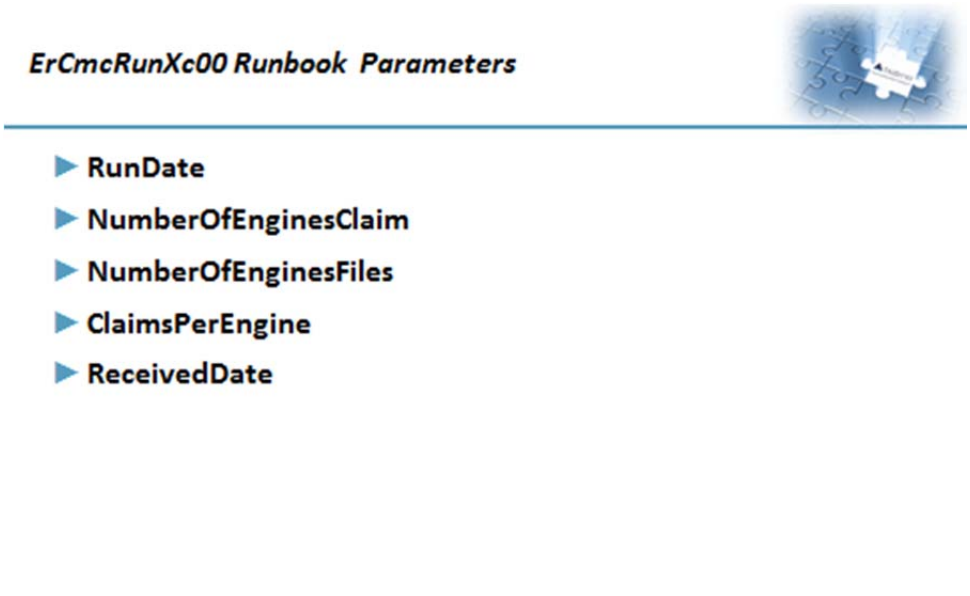


The True XC (ErCmcRunXc00) process:

- Reads the file containing XC formatted claims.
- Maps the XC fields to Facets fields.
- Writes claims to an intermediate work file.
- Reads the intermediate work file created in the previous step.
- Generates a Claim Number, if necessary.
- Fills the default values, if necessary.
- Imports the intermediate work file into the Facets XC (Temporary) Database.
- Optionally creates a SQL file to delete rows added to the database as part of the XC back out process.
- Generates Facets Claims Acknowledgments (optionally, through the use of the CreateAcknowledgmentFile run file parameter.).
- Reports on errors and run controls.

Note: The batch action code is a field on the XC file layout. Values include: H (Pend Claim), P (Pay Claim), E (Pre-price Claim), D (Predetermine Dental Claim), Q (History), R (Recall and Adjudicate), and S (Recall and Pre-price).

ErCmcRunXc00 Runbook Parameters



Parameter	Meaning
RunDate	The user enters the date in mm/dd/yyyy hh:mm:ss.000 format.
NumberOfEnginesClaim	The user enters the number of EDI processes to start simultaneously in the multi-engine job Step 5000. The parameter defaults to 1.
NumberOfEnginesFiles	The user enters the number of processes to start simultaneously in the pre-processor Step 3000. This value should be less than or equal to the number of files processing by the job. The parameter defaults to 1.
ClaimsPerEngine	The user enters the number of claims to process by a single engine before proceeding to the next set of claims. This number should equal less than the total number of claims in the input file divided by the number of engines. The procedure book defaults to 1,000. If the user did not specify this number, all claims enter into a single queue, even if the number of claims exceeds 1,000.
ReceivedDate	The user optionally enters a valid date in MM/DD/YYYY format that overrides the received date for all claims.

ErCmcRunXc00 Runbook Parameters

- InputDir
- OutputDir
- HistoryDir
- DeleteDir
- AltIndexDir
- InputFileExtension

Parameter	Meaning
InputDir	The user enters the directory where 837 Dental, Institutional, and/or Professional keyword files are read. Facets requires this entry.
OutputDir	The user enters the directory where acknowledgment (.ack), error (.err) and external (XC) files get written. This is required.
HistoryDir	The user enters the directory where backup (.bak) 837 Dental, Institutional, and/or Professional keyword files get written. Facets requires this entry.
DeleteDir	The user specifies the directory that store output files containing rollback SQL that rolls back 837i keyword files from the XC database.
AltIndexDir	The user enters the directory where Gateway Alternate Index (.alt) files get written. Facets requires this entry.
InputFileExtension	The user enters the extension of the 837 Dental, Institutional, and/or Professional keyword files that you wish to process. The parameter defaults to txt.

ErCmcRunXc00 Runbook Parameters

- MultipleInputFiles
- NoInputFilesError
- InputFile
- DeadlockRetryAttempts
- DeadlockRetryWaitSeconds

Parameter	Meaning
MultipleInputFiles	The user enters Y to process multiple input files. Otherwise, the user enters N.
NoInputFilesError	The user uncomments, or sets the NoInputFilesError switch to Y, which forces a return code of 8 when multiple input files process and no input files exist for the specified input file extension. The default sets to N, which returns a 0 return code.
InputFile	The user enters the name of the 837 Dental, Institutional, and/or Professional keyword file without the file extension. Facets uses this file name when MultipleInputFiles set to N.
DeadlockRetryAttempts DeadlockRetryWaitSeconds	The user specifies the desired number of retries attempted for a deadlocked transaction. DeadlockRetryWaitSeconds - The user specifies the number of seconds to wait between retry attempts. Facets uses the DeadlockRetryAttempts and DeadlockRetryWaitSeconds options to tune this application for optimal performance based upon the client's specific environment and processing volumes. Setting these values inappropriately can adversely affect job performance.

ErCmcRunXc00 Runbook Parameters

- **EnableConcurrentExecution**
- **MaxClaimDeadlocks**
- **ClclUserId**
- **CompressClclId**

Parameter	Meaning
EnableConcurrentExecution	Entering Y allows this job to run when other EDI 837 Inbound or True XC Claims jobs are already running. If the user set the runbook parameter MultipleInputFiles to Y, the InputDir should not be the same as the other jobs that already ran. The user cannot restart this job after a fatal error when this switch is set.
MaxClaimDeadlocks	The user enters the number of deadlock errors that need to occur before the job terminates. By default, this job won't terminate when it encounters a deadlock error.
ClclUserId	The user enters the USUS_ID for the related KEYG numbering scheme. When running more than one engine, Facets uses this value, along with the engine number, to build the USUS_ID. If no USUS_ID passes, Facets then uses a USUS_ID of spaces or all engines. The first engine starts with the number 1.
CompressClclId	This option allows Claim IDs generated from the **KEYG** table to compress to a 10 position Claim ID before appending the two position Segment ID. Facets completes this process by converting the sequential number portion of the claim number from a base 10 to a larger base. This logic only performs when KEYG_GEN_LEN is greater than 10.

ErCmcRunXc00 Runbook Parameters

- InsertOption
- CreateAcknowledgementFile
- MemberSearchConfigFile
- ProviderSearchConfigFile

Parameter	Meaning
InsertOption	Entering Y inserts records to the database. Entering N applies the records to the database. For example, if a Claim ID already exists on the XC database and the user set the InsertOption to N (apply records), the new claim loading with the same claim ID overwrites/updates the existing claim. If the user set the InsertOption to Y (insert records), the new claim loading errors as a duplicate claim. Facets defaults to N (apply the records to the database).
CreateAcknowledgementFile	This user enters M for Medical, D for Dental, or B for Both. Facets defaults to 'blank' if an acknowledgement file doesn't get created.
MemberSearchConfigFile	The user optionally enters the full path and file name of the Member Search Configuration file in order to activate the member search.
ProviderSearchConfigFile	The user optionally enters the full path and file name of the Provider Search Configuration file in order to activate the provider search.

ErCmcRunXc00 Runbook Parameters

- RFOtherProvEdit
- FAOtherProvEdit
- OtherProvEdit
- RendProvEdit
- SuppressSearchErrors
- SearchDebug
- WorkFlow

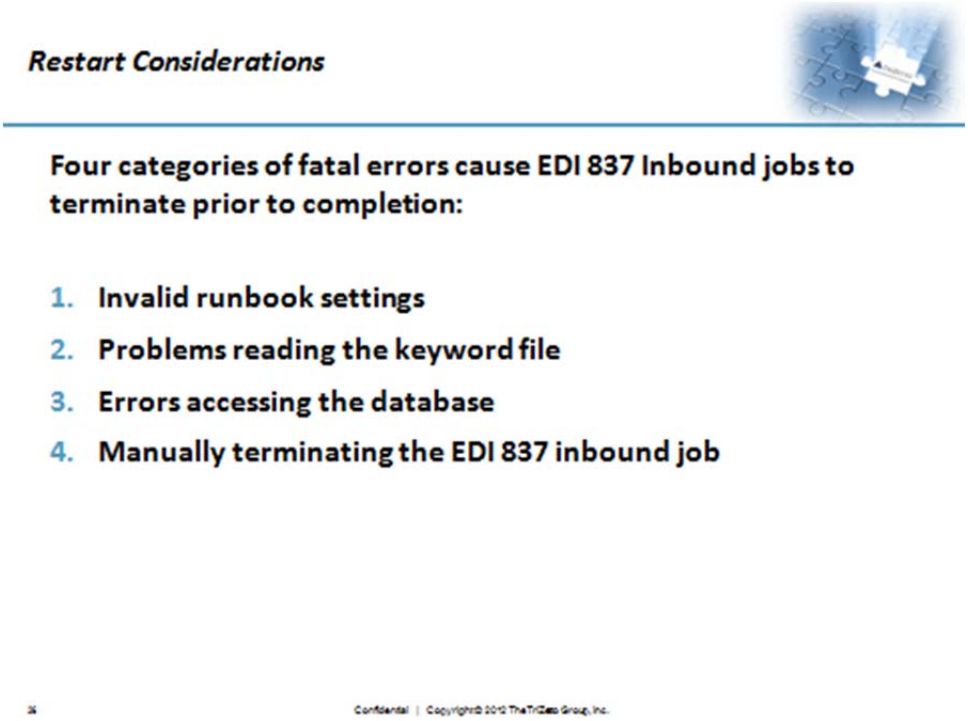
Parameter	Meaning
RFOtherProvEdit	Enter Y to edit the Referring Provider ID. Enter N to not edit the Referring Provider ID. Defaults to N.
FAOtherProvEdit	Enter Y to edit the Facility Provider ID. Enter N to not edit the Facility Provider ID. Defaults to N.
OtherProvEdit	Enter Y to edit the Admitting Provider (AD), Operating Provider (OP), Other Provider 1 (01), and Other Provider2 (02) IDs. Enter N to not edit the Admitting Provider (AD), Operating Provider (OP), Other Provider1 (01), and Other Provider 2 (02) IDs. Defaults to N.
RendProvEdit	Enter Y to validate the Rendering Provider ID at the line item level (CDSD) using Tax ID and/or NPI Enter N to not validate the Rendering Provider ID at the line item level (CDSD). Defaults to N.
SuppressSearchErrors	Enter Y to not write search errors to the log. Enter N to write search errors to the log. Defaults to N.
SearchDebug	Enter Y to write the search debug information to the log file. Enter N to not write search debug information to the log file. Defaults to N.
WorkFlow	Facets uses this indicator in conjunction with the Member/Provider value added searches. When implementing workflow, set the switch to Y. This causes any claims where member/provider keys could not resolve, to write to the XC database with status 16 so they can enter into the adjudication/workflow process. Set this switch to N if not implementing value add searches and/or workflow.

ErCmcRunXc00 Runbook Parameters

- **ExcludeVowels**
- **AmbulanceMileage**
- **EnableConcurrentExecution**
- **Step 3000**
  - ErrorFileInd
    - Set to Y or N

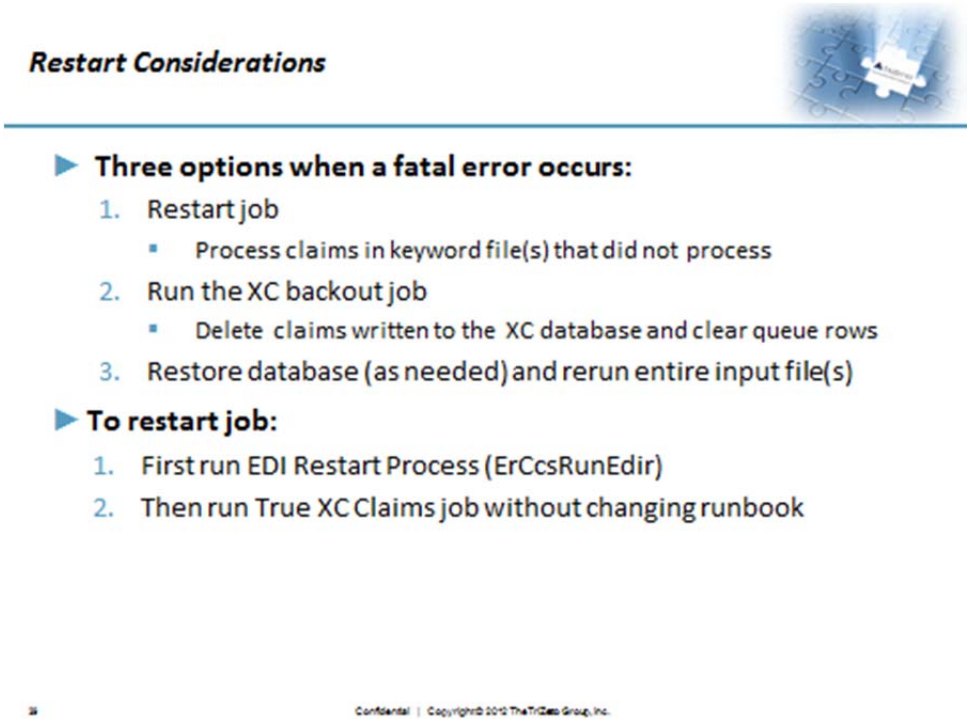
Parameter	Meaning
ExcludeVowels	Facets uses this entry in conjunction with the CompressClclId runbook option. This option only applies to CompressClclId values of 1 and 2. The user enters N to exclude uppercase letters I and O. The user enters Y to exclude uppercase letters A, E, I, O, U and Y. This parameter defaults to Y.
AmbulanceMileage	The user enters Y to assign Ambulance Mileage Units as an override for Ambulance Mileage Claims. This parameter defaults to N.
EnableConcurrentExecution	The user enters Y, which allows this job to run while already running other EDI 837 Inbound or True XC Claims jobs. If the user sets the runbook parameter MultipleInputFiles to Y, the InputDir should not be the same as the other jobs currently running. The user cannot restart this job after a fatal error when this switch is set.
Step 3000 / ErrorFileInd	This parameter sets to Y or N. Y produces the Error File. N does not produce the Error File.

Restart Considerations



Facets contains four main categories of fatal errors causing the True XC job to terminate prior to completion:

1. Invalid runbook settings
2. Problems reading the keyword file. An invalid or missing record type in the keyword file causes the True XC Claims job to terminate.
3. Errors accessing the database. The True XC Claims job terminates if an error occurs when writing to the database. Errors caused by deadlocks won't cause this job to terminate.
4. True XC Claims jobs terminated manually. This job manually terminates by executing the EDI Termination job (ErCcsRunEdit).



The user has three options when a fatal error occurs:

1. Restart this job processing the claims in the keyword file(s) that did not process originally. If the user set the runbook parameter, EnableConcurrentExecution, to Y, the user cannot use this option.
2. Run the XC backout job deleting the claims written to the XC database and clearing the queue rows.
3. Restore the database (if necessary) and rerun the entire input file(s).

Restarting this job:

- First run the EDI Restart Process (ErCcsRunEdir).
- Then run the True XC Claims job without making any changes to the runbook. The user should not use the BypassStep, RestartStep, or StopStep runbook parameters in an attempt to restart this job.

Facets Acknowledgement Record File Layout

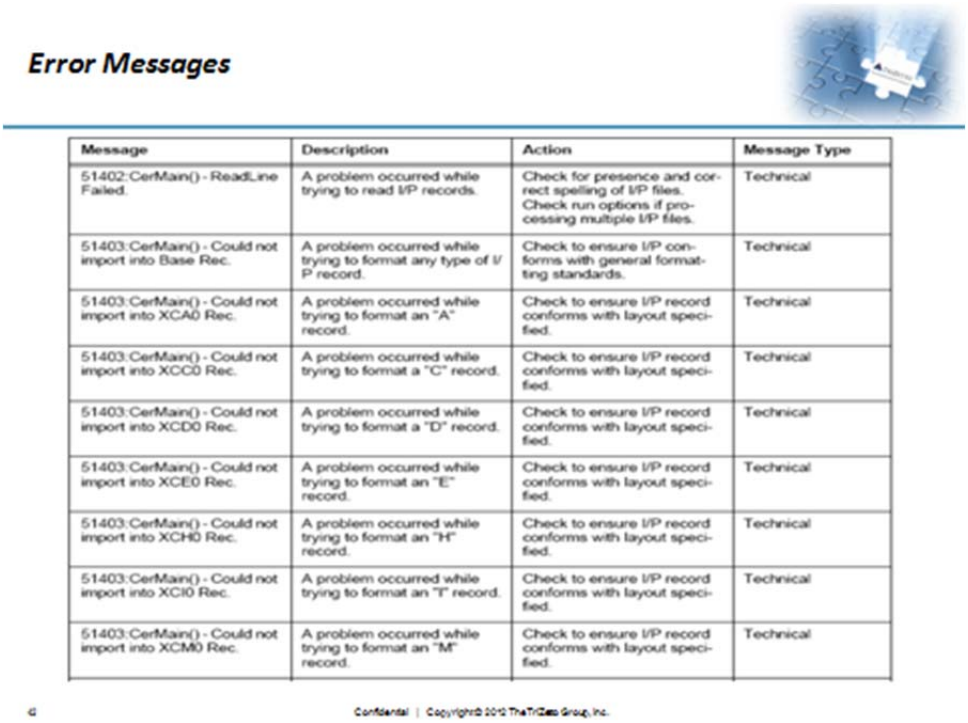
Facets Acknowledgement Record File Layout

Field	Field Code	Size	Facets Field	Comments
Record Type	RecType	3	None	Move 'ACK'
Claim Number	CLCL_ID	12	CLCL_ID	Move value of CLCL_ID
Claim Acknowledge-ment Status	CLCL_ACK_STS	1	None	Move a value of 'A' (Accepted) if the claim is successfully applied to the XC database. Move a value of 'R' (Rejected) if the claim is **not** successfully applied to the XC database.
Claim Acknowledge-ment Date	CLCL_ACK_DT	8	None	Move current system date and time in CCYYMMDD format.
Claim Input Method	CLCL_INPUT_METH	1	CLCL_INPUT_METH	Move value of CLCL_INPUT_METH
Claim Type	CLCL_CL_TYPE	1	CLCL_CL_TYPE	Move value of CLCL_CL_TYPE
Claim Sub Type	CLCL_CL_SUB_TYPE	1	CLCL_CL_SUB_TYPE	Move value of CLCL_CL_SUB_TYPE
Claim External Refer-ence Number	CLMS_EXT_REF	15	CLMS_EXT_REF	Move value of CLMS_EXT_REF
Claim Trading Partner	CLMS_TRAD_PARTNER	15	CLMS_TRAD_PARTNER	Move value of CLCL_TRAD_PARTNER

Facets Acknowledgement Record File Layout

Field Code	Size	Facets Field	Comments
RecType	4	None	Move 'ALTX'
GWID	32	GWID	Move value XC keyword @pGWID
TRANSID	4	None	If the CLCL_CL_SUB_TYPE = 'M' 'Move 837P' . If the CLCL_CL_SUB_TYPE = 'D' 'Move 837D'. If the CLCL_CL_SUB_TYPE = 'H' 'Move 837I'.
INDEX_NAME	21	None	Move 'basePayerClaim-Number'
INDEX_DATA	12	CLCL_ID	Move value of CLCL_ID
HIPAA_REPOSITORY_INSTANCE (from run file parameter)		None	Move "FACETS"
HIPAA_BASE_SYSTEM_INSTANCE (from run file parameter)		None	To be determined

Error Messages

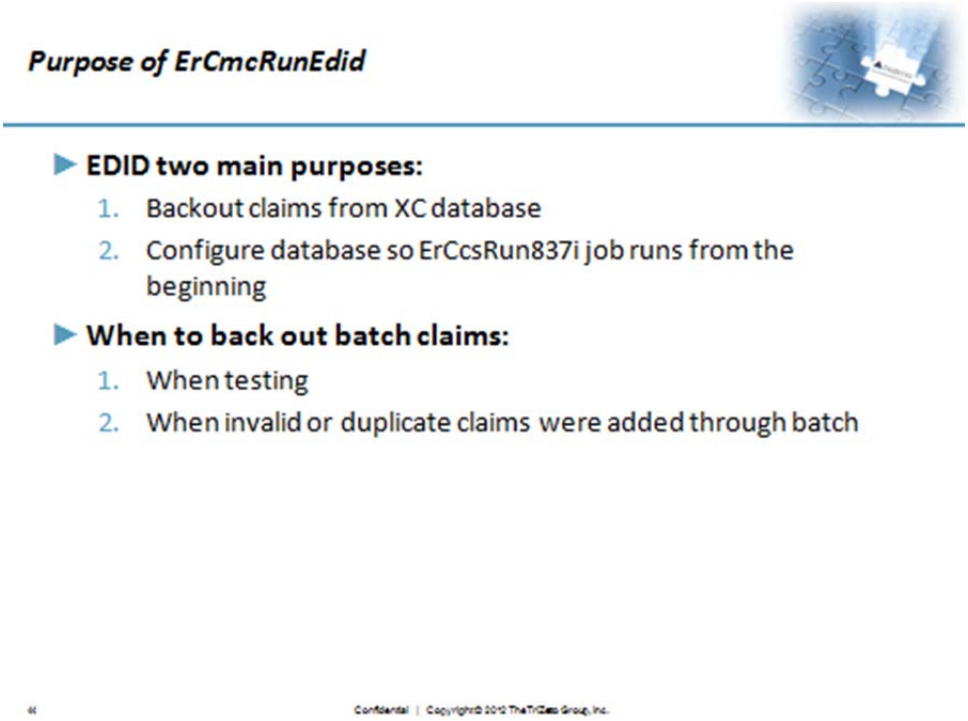


Error Messages

Message	Description	Action	Message Type
51402:CerMain() - ReadLine Failed.	A problem occurred while trying to read I/P records.	Check for presence and correct spelling of I/P files. Check run options if processing multiple I/P files.	Technical
51403:CerMain() - Could not import into Base Rec.	A problem occurred while trying to format any type of I/P record.	Check to ensure I/P conforms with general formatting standards.	Technical
51403:CerMain() - Could not import into XCA0 Rec.	A problem occurred while trying to format an "A" record.	Check to ensure I/P record conforms with layout specified.	Technical
51403:CerMain() - Could not import into XCC0 Rec.	A problem occurred while trying to format a "C" record.	Check to ensure I/P record conforms with layout specified.	Technical
51403:CerMain() - Could not import into XCD0 Rec.	A problem occurred while trying to format a "D" record.	Check to ensure I/P record conforms with layout specified.	Technical
51403:CerMain() - Could not import into XCE0 Rec.	A problem occurred while trying to format an "E" record.	Check to ensure I/P record conforms with layout specified.	Technical
51403:CerMain() - Could not import into XCH0 Rec.	A problem occurred while trying to format an "H" record.	Check to ensure I/P record conforms with layout specified.	Technical
51403:CerMain() - Could not import into XCI0 Rec.	A problem occurred while trying to format an "I" record.	Check to ensure I/P record conforms with layout specified.	Technical
51403:CerMain() - Could not import into XCM0 Rec.	A problem occurred while trying to format an "M" record.	Check to ensure I/P record conforms with layout specified.	Technical

Facets users find Error Messages documented in the ORM.

XC Backout / Purpose of ErCmcRunEdid



The two main purposes in EDID include:

4. Backing out claims from the XC database.

Place the appropriate delete files in the input directory for the EDID job. To backout claims from the XC database without deleting the QWK0 rows in step 2 below, set runbook parameter StopStep to 5000 in the EDID job.

5. Configuring the database so the ErCcsRun837i job runs from the beginning by deleting the queue (QWK0) rows.

If the ErCcsRun837i job terminated in any step other than 5000 or 9000, set the runbook parameter RestartStep to 5000 in the ErCmcRunEdid job. Uncomment the ErCcsRun837i section in the EDID runbook when running EDID after 837i failed.

A TriZetto best practice includes backing out batch claims when testing or when invalid or duplicate claims accumulated through batch.

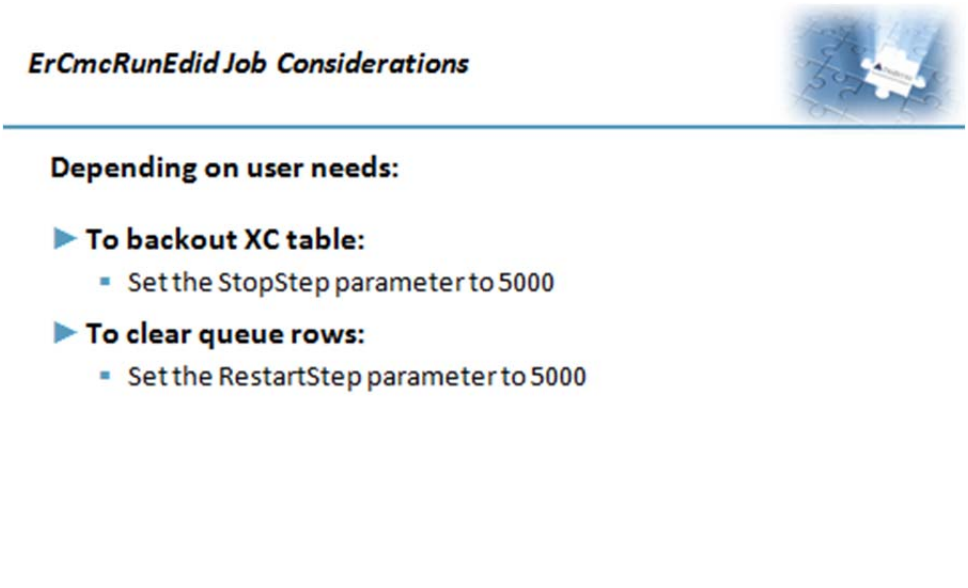
ErCmcRunEdid Runbook Parameters

ErCmcRunEdid Runbook Parameters

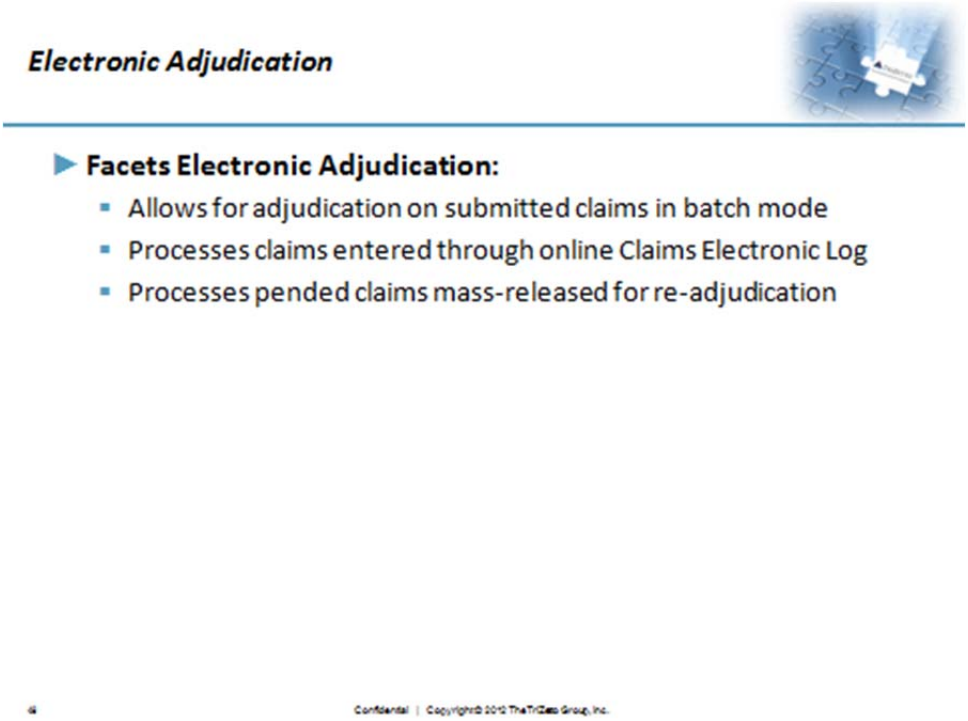
- **InputFileExtension**
- **InputDir**
- **MultipleInputFiles**
- **InputFile**
- **PreProcessorPzapAppId**
  - Enter PZAP_APP_ID of Step 3000 of failed job
    - Ex: <!--<Item name="PreProcessorPzapAppId">EDIP</Item>--> <!--<Item name="PreProcessorPzapAppId">XC00</Item>-->
- **Step 3000**
  - HistoryDir {directory name}
    - Stores backups of deleted files as they process

Parameter	Meaning
InputFileExtension	The user enters the appropriate input file extension, which Facets requires for both single and multiple file processing.
InputDir	The user enters the appropriate input directory for the .del file, which Facets requires.
MultipleInputFiles	The user enters Y to process multiple input files. The parameter defaults to N.
InputFile	The user enters the name of the files with no extension in order to process a single file. To process multiple files, leave commented.
PreProcessorPzapAppId	The user enters the PZAP_APP_ID of Step 3000 for the job that failed. Ex: <!--<Item name="PreProcessorPzapAppId">EDIP</Item>--> <!--<Item name="PreProcessorPzapAppId">XC00</Item>-->
Step 3000	The HistoryDir {directory name} stores backups of deleted files as they process. The user ensures that the ErCmcProcEdid_3000_HistoryDir does not point to the same directory as the HistoryDir from a previous job (i.e. ErCcsRun837i, ErCmcRunXc00).

ErCmcRunEdid Job Considerations



Electronic Adjudication



Facets Electronic Adjudication:

- Allows clients to adjudicate medical, hospital and dental electronically submitted claims in batch mode
- Processes claims entered through the online Claims Electronic Log applications
- Processes pended claims mass-released for re-adjudication

Electronic Adjudication

- **Facets Electronic Adjudication:**
  - Uses features and considerations regarding:
    - Batch run options
    - Error handling
    - Reporting
    - Security
    - Workflow
  - Uses adjudication routines to process claims similar to other online applications

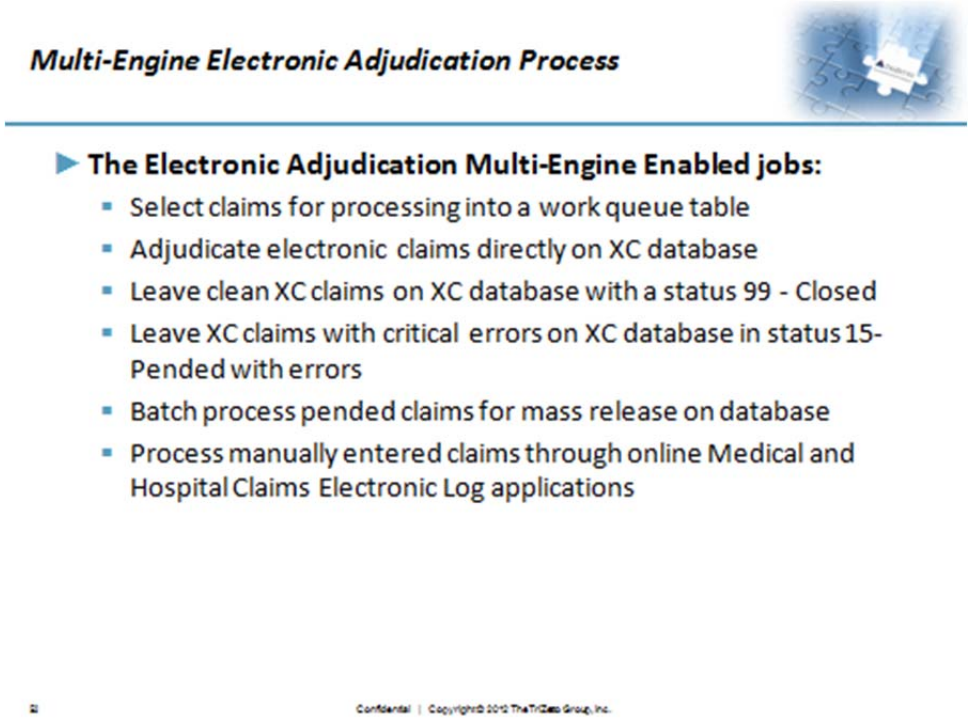
Electronic Adjudication

- **System components:**
  - Edit submitted claim data for accuracy
  - Reduce user entry error
  - Improve productivity by automating processing of tens of thousands of claims per day

Multi-Engine Electronic Adjudication Process

Multi-Engine Electronic Adjudication Process

The Medical Electronic Adjudication, Multi-Engine Enabled process (ErCmcRunClmu) executes an enhanced version of the electronic adjudication application for medical and hospital claims.



The Electronic Adjudication Multi-Engine Enabled jobs:

- Select claims for processing into a work queue table (**CMC_CLWQ_WORK_QUE**)
- Adjudicate electronic (XC and/or 837) claims directly on the XC database. Prior to processing, these XC claims contain a current claim status of 16 – Pended following batch entry. When adjudicating XC claims, multi-engine electronic adjudication optionally screens for critical errors. In the run file, the user can bypass the job step that selects these claims for processing.
- Leave clean XC claims (i.e., those without critical errors) on the XC database with a status of 99 - Closed after validating for member and provider, and subsequently (and optionally) mass deleting them at the end of the job
- Leave XC claims with critical errors on the XC database in a claim status of 15-Pended with errors
- Batch process pended claims for mass release on the Facets database. The user may bypass this optional step that selects these claims for processing in the run book.

Note: When Facets mass releases pended claims, the claims fully re-adjudicated. Facets considers any changes or updates to member eligibility or plan components after the claim originally pended during re-adjudication.

- Process manually entered claims through the online Medical and Hospital Claims Electronic Log applications. The user may bypass this optional feature that selects these claims for processing in the runbook.

Runbook Parameters

Runbook Parameters

- **BatchId - Specify Batch ID parameter for the job**
- **DeleteXcClaims - Job level override allows deletion of pended XC claims from XC database**
- **SetLastActionDatetimeToJobBeginDatetime - Determines CLCL_LAST_ACTION_DTM value used in adjudication during a given batch run**

Parameter	Meaning
BatchId	The user specifies the Batch ID parameter for the job. If the user enters a value for this variable, only claims with a matching value on the **CLCL_BATCH_ID** field on the **CLCL** table get processed.
DeleteXcClaims	The user may enter a job level override to allow the deletion of pended XC claims from the XC database. A value of M deletes status 99 Medical claims only at the end of the job. A value of Y deletes all XC claims in Status 99. If left commented, the job level default also deletes all medical and dental XC claims in Status 99 at the end of the job. A NULL value (comments removed but no value entered), or any other value such as N, won't delete XC claims at the end of the job.

Parameter	Meaning
SetLastActionDatetimeToJobBeginDatetime	This parameter determines the CLCL_LAST_ACTION_DTM value used in adjudication during a given batch run. It defaults to N, which uses the system datetime when claims update. Passing a Y value sets the value to the batch's SYIN_CREATE_DTM for all claims in a given run. This allows claims processing to run overnight (before and after midnight in a single run) in order to all have the same value in CLCL_LAST_ACT_DTM, which can potentially impact whether or not they get picked up for payment by their corresponding payment batch.

ErCmcRunClmu Runbook Parameters

ErCmcRunClmu Runbook Parameters

- NumberOfEnginesClmu – Specifies desired number of electronic adjudication processes to run simultaneously in step 4000
- NumberOfQuesClmu - Indicates total number of work queues allocated and assigned across specified number of engines
- MaxClaimsPerQue – Indicates maximum number of claims allocated to each queue

Parameter	Meaning
NumberOfEnginesClmu	The user specifies the desired number of electronic adjudication processes run simultaneously in step 4000. Facets recommends 2 engines per available processor (CPU). The parameter defaults to 2.
NumberOfQuesClmu	The user indicates the total number of work queues allocated and assigned across the specified number of engines. The procedure book defaults to 6. This number represents the total number of queues run, not queues per engine. Additionally, when selecting claims from both the XC and the Facets database, this represents the number of queues created per database (if the user selects 6, 6 queues create for each database for a total of 12).
MaxClaimsPerQue	The user indicates the maximum number of claims allocated to each queue. The user uses this optional parameter to override the number of claims assigned to each queue versus using the default method of #Claims/#Queues. The user does not need to enable the MaxClaimsPerQue parameter if using the NumberOfQuesClmu runbook parameter.

If CLMU processes less than 1,000 claims of one claim type (XC claims only or online claims only), Facets allocates only one queue regardless of the number of queues specified in the runbook.

Note: Users who create more queues with fewer claims don't necessarily improve the overall batch runtime since this requires more overhead.

ErCmcRunClmu Runbook Parameters

- QueCapture - Displays QWK0 rows in Step 3600
- BypassBCBSProcessITS - Job level override to process ITS claims for BCBS
- ItsCppaDebugCd – Allows settings for recon records to be written for transactions
- ItsApicReconId – Allows a way to trace content of CPPR (Cross Platform Passing Area) from Facets
- ItsCppaDebugPath – Allows users to specify directory for trace file icpa.txt

Parameter	Meaning
QueCapture	This parameter displays the QWK0 rows in Step 3600. It always displays as active and does not need setting. If left uncommented and set to N, the QWK0 rows don't display in the CLMU batch logs for Step 3600 (identified as Work Queue Display).
BypassBCBSProcessITS	The job level override, BypassBCBSProcessITS, processes ITS claims for BCBS. It defaults to Y, bypass ITS claims processing (do not process). N means do not bypass ITS claims (or process ITS). To process both ITS claims and regular claims, uncomment the following line: <item name = “BypassBCBSProcessITS”> “N” </item>
ItsCppaDebugCd	For clients running ITS claims processing, this parameter allows settings for recon records written for transactions.
ItsApicReconId	For clients running ITS claims processing, this parameter allows a way to trace the content of the CPPA (Cross Platform Passing Area) from Facets.
ItsCppaDebugPath	For clients running ITS claims processing, this parameter allows users to specify the directory where the trace file icpa.txt gets written.

ErCmcRunClmu Runbook Parameters

- **ItsXith0IniPath - Allows Microsoft NT platform users with ITS Home Middleware Interface to trace content of CPPA (Cross Platform Passing Area) from Facets**
- **ItsCloseEngineLilError - Provides run file option allowing engine to complete current work queue**
- **ApplicationPzapAppId - Provides ability to override job PzapAppId with different product configuration**
- **WorkflowReprocess - Determines whether or not Workflow Reprocess Claims Steps 2250 and 5450 are run**

Parameter	Meaning
ItsXith0IniPath	For clients running ITS claims processing, this parameter allows users with the Microsoft NT platform and the ITS Home Middleware Interface, a way to trace the content of the CPPA (Cross Platform Passing Area) from Facets.
ItsCloseEngineLilError	This value provides a run file option that allows the engine to complete the current work queue and close when a severe LIL error occurs. This limits the amount of ITS Home claims bypassed by this engine.
ApplicationPzapAppId	The user enters an alternate PZAP ID, which provides the ability to override the job, PzapAppId, with a different product configuration. Additional product data needs to be set to run the job with this PZAP. This optional parameter defaults to the supplied job PZAP.
WorkflowReprocess	This parameter determines whether or not the Workflow Reprocess Claims Steps 2250 and 5450 run. It defaults to N - Do Not Run the Steps. If using Mass Claim Adjustments or Workflow Itinerary processing to bring claims into the adjudication process, uncomment this tag and set the value to Y.

ErCmcRunClmu Runbook Parameters

- MinTotalClaimsRequiredForQues
- DeadlockRetryAttempts
- DeadlockRetryWaitSeconds
- JobRetCdThreshold
- WarningRetCd
- ErrorRetCd
- CaptureWorkTable

Parameter	Meaning
MinTotalClaimsRequiredForQues	This specifies the minimum amount of claims required to create a single queue.
DeadlockRetryAttempts	This specifies the desired number of retries to attempt for a deadlocked transaction.
DeadlockRetryWaitSeconds	This specifies the number of seconds to wait between retry attempts. The DeadlockRetryAttempts and DeadlockRetryWaitSeconds options work with this application for optimal performance based on the client's specific environment and processing volumes. Setting these values inappropriately may negatively impact job performance.
JobRetCdThreshold	This identifies the highest return code allowed for subsequent steps.
WarningRetCd	This parameter identifies the return code used for warning conditions identified by the CheckQues step that follows the multi-engine step.

Parameter	Meaning
ErrorRetCd	This parameter identifies the return code used for error conditions identified by the CheckQues step that follows the multi-engine step.
CaptureWorkTable	This parameter displays the CLWQ rows in Steps 2400 (pre-adjudication) and 5300 (post-adjudication). It defaults to N, so it only gets uncommented if wanting to capture the queue table. Anything entered besides Y gets treated as N.

Claims Adjudication Error Handling

Claims Adjudication Error Handling

Critical Errors:

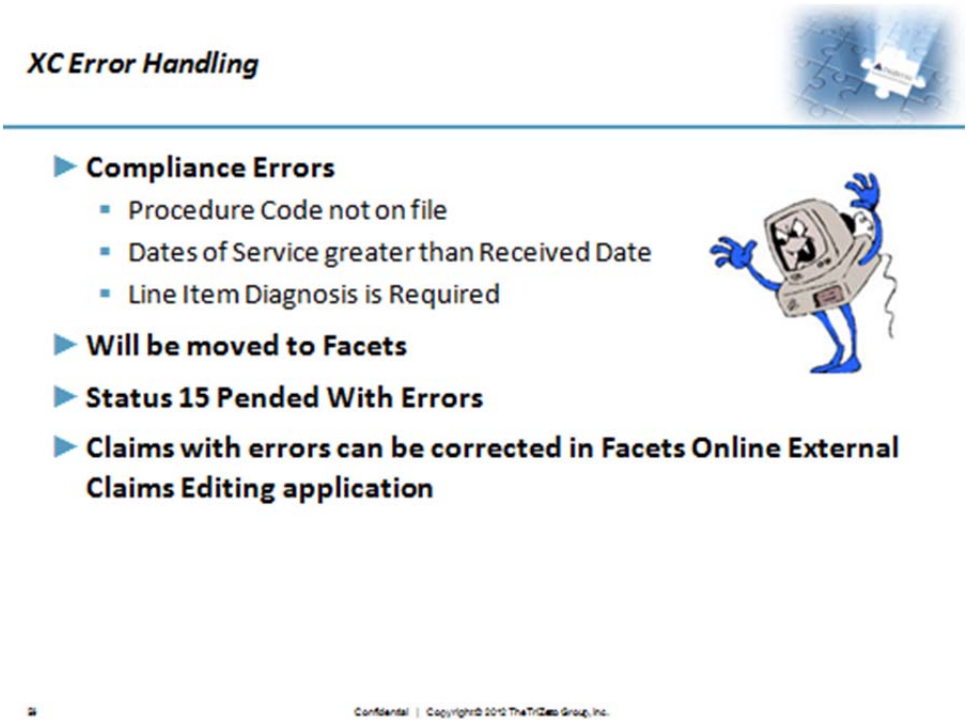
- **Member not found - Member on submitted claim not found on database**
- **Provider not found - Provider on submitted claim not found on database**
- **Duplicate Claim ID - Claim ID submitted for the claim already exists on database**
- **Invalid Recall Transaction - Recall logic attempts to recreate actions of a user recalling a claim and reprocessing it with new inputs**

Critical Errors	Meaning
Member not found	This error indicates that Facets can't locate the member submitted with the claim on the database. Facets may have found a discrepancy between the submitted member data and the data stored for the member in Facets, or the member may simply not exist in the database.
Provider not found	This error indicates that Facets can't locate the provider submitted with the claim on the database. Facets may have found a discrepancy between the submitted provider data and the data stored for the provider in Facets, or the provider may simply not exist in the database.
Duplicate Claim ID	This error indicates the Claim ID submitted for the claim already exists in the database. These errors typically result from an incorrect claims auto numbering scheme.
Invalid Recall Transaction	This error means that Facets didn't find the original claim segment on the adjusted claim. It indicates that recall logic in batch processing attempted to recreate the actions of a user recalling a claim in the online application and reprocessed it with new inputs. Recall logic assumes the submission of the original claim segment, and that a recall segment containing the original Claim ID subsequently submitted with new information for reprocessing. This error prevents claims from being loaded into Facets with Status 15 - Pended with Errors.

The above errors prevent claims from loading from the XC database into Facets. If a critical error exists, the claim remains in XC with a status of 15 – Pended with Errors.

The External Claims Editing application corrects the errors. The user opens these claims and manually corrects or resubmits them.

XC Error Handling



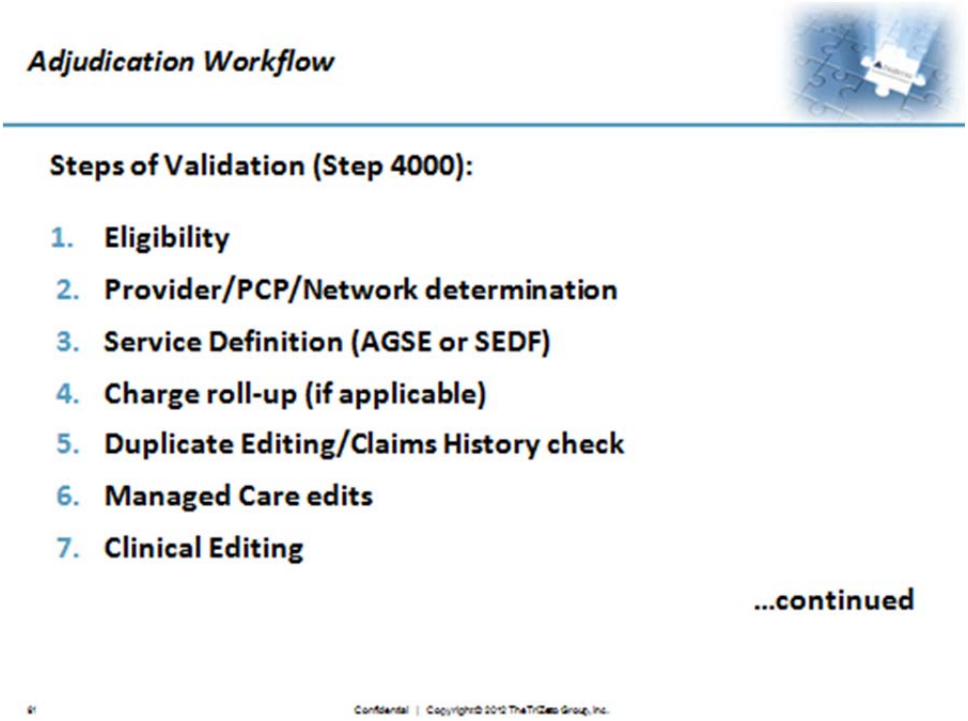
The XC database removes claims with adjudication errors and moves them to Facets, but the claim with the error gets assigned a claim status of 15 (Pended with Errors). The user must recall these claims manually, review, and correct them in the online Facets claims processing applications.

Medical Electronic Adjudication

Medical Electronic Adjudication

- ErCmcRunClmu
- Multi-engine batch process
- Data selected by CLCL_TYPE = M and CLCL_CURR_STS = 16
- Worktable CMC_CLWQ_WORK_QUE

Adjudication Workflow

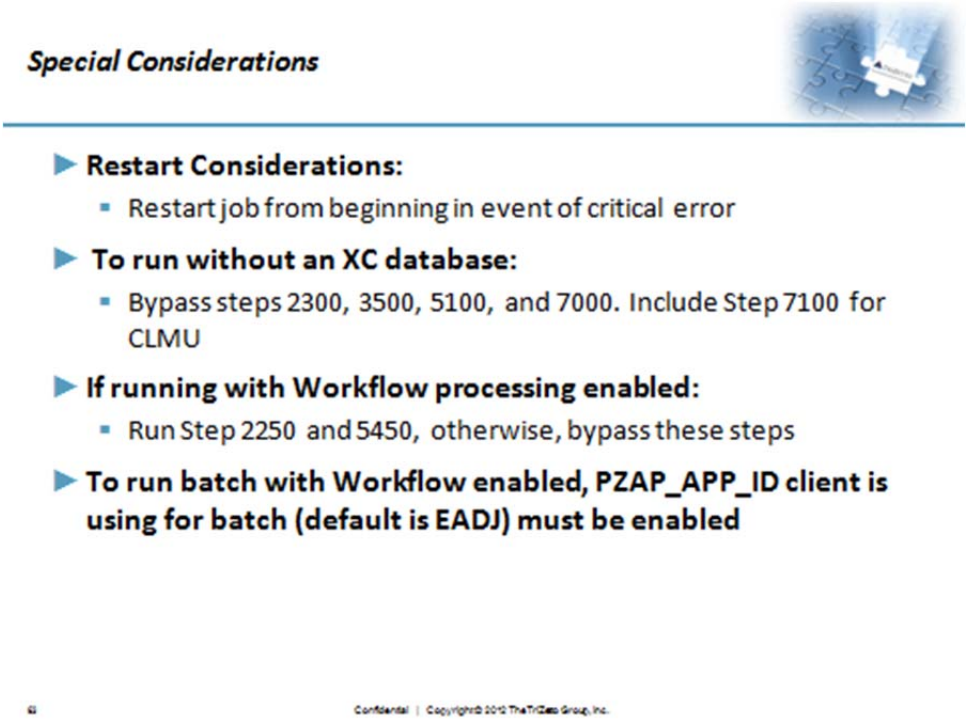


The Facets claims adjudication routine incorporates a series of routines, including Eligibility, Provider/PCP/Network, and Pricing, that together apply rules and parameters to the claim.

Adjudication Workflow - continued

8. **Line Item Prefixes**
9. **Pricing (SEPC)**
10. **Service Rules, Deductibles, Limits, Penalties**
11. **COB**
12. **Accumulator Update**
13. **Payment Drag**
14. **Status update**

Special Considerations



Restart Considerations:

- The user should restart this job from the beginning in the event of a critical error in any step.

To run without an XC database:

- Bypass steps 2300, 3500, 5100, and 7000. Include Step 7100 for CLMU.

If the user runs this step with Workflow processing enabled:

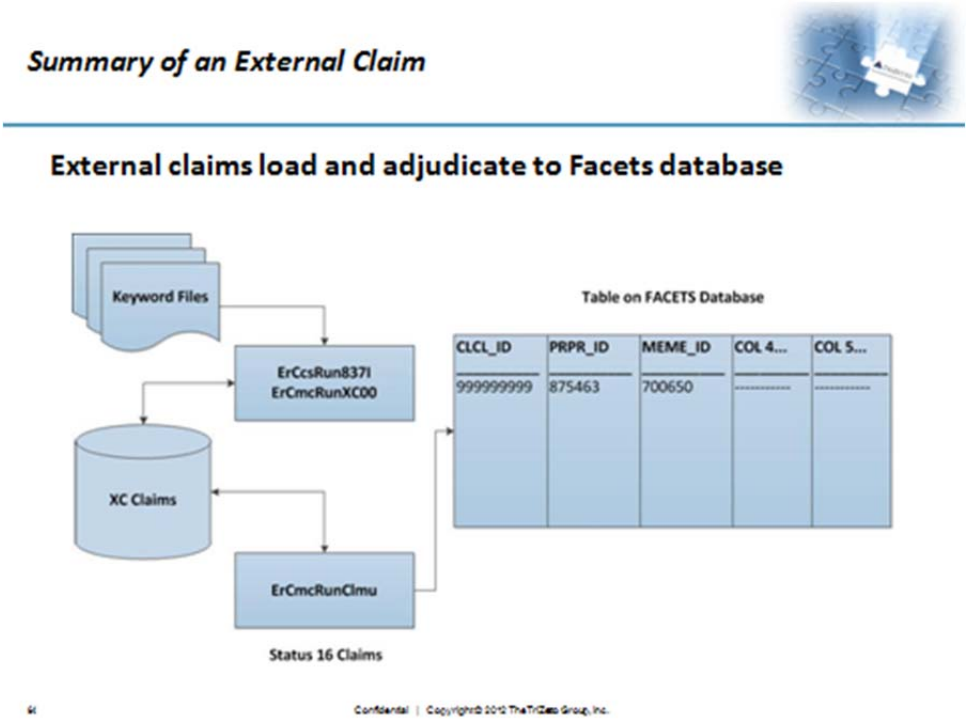
- Run Step 2250 and 5450. Otherwise, bypass these steps.

To run the batch with Workflow enabled, the user must enable the PZAP_APP_ID used for the batch (default = EADJ) for Workflow processing on table, **CER_PZAP_APP**.

The user must enable the PZAP_APP_ID for WKFL (Workflow) in order to achieve steps 2250/5450, which apply to claims sent into the batch process via a Workflow application such as Mass Claims Adjustments and Itinerary processing. The user must uncomment the tag WorkflowReprocess in the runbook and set to it Y.

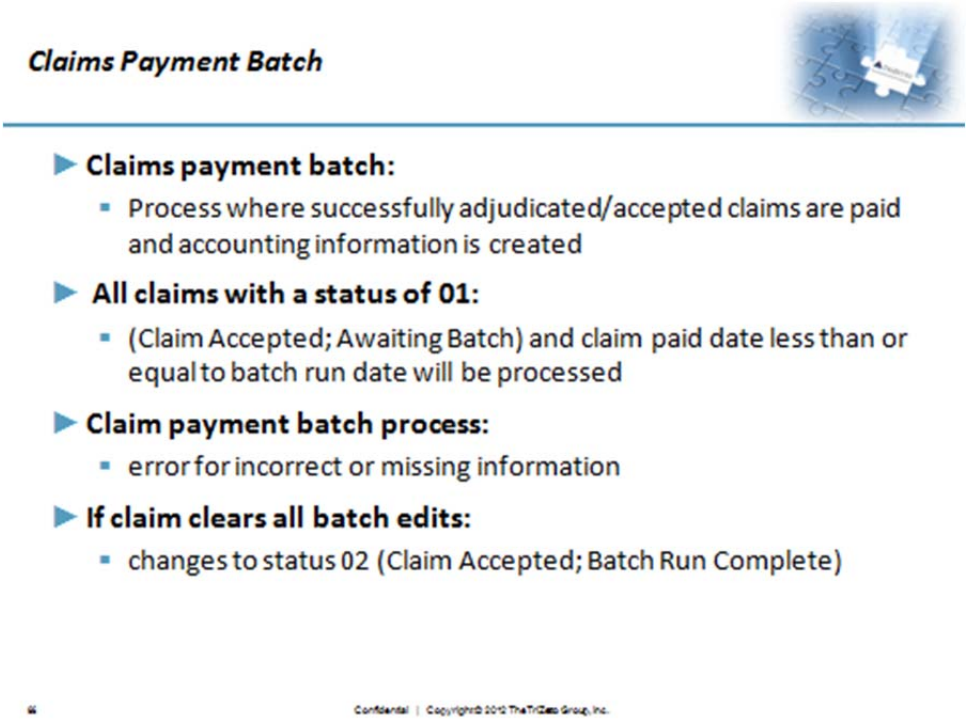
Note: If the user only enables the PZAP_APP_ID for the adjudication batch for Workflow, batch claims process through Workflow as usual, but won't pick up Mass Adjustment and itinerary claims reprocessed through the steps 2250 and 5450.

Summary of an External Claim



ErCmcRunClmu must run to move the XC claims into Facets.

Claims Payment Process / Claims Payment Batch



In Facets, the claims payment batch successfully adjudicates/accepts claims (regardless of process). The claims pay and Facets creates accounting information.

All claims (hospital/medical) with a status of 01 (Claim Accepted; Awaiting Batch) and a claim paid date (CLCL_PAID_DT on the **CMC_CLCL_CLAIM** table) less than or equal to the batch run date entered in the claim payment batch run book (ERCMCRUNCKMM) process by the batch.

The claim payment batch process errors for incorrect or missing information, such as no provider remit address and no provider Tax ID.

If the claim clears all batch edits, it changes to a status 02 (Claim Accepted; Batch Run Complete), and Facets writes all related accounting record rows to the database.

ErCmcRunCkmm Runbook Parameters

ErCmcRunCkmm Runbook Parameters

- User Id
- NumberOfEnginesClmm
- NumberOfEnginesCkmm
- NumberOfQuesClmm
- NumberOfQuesCkmm
- DeadlockRetryAttempts
- DeadlockRetryWaitSeconds
- SecurityLimit
- JobRetCdThreshold

Users enter the system User ID.

Parameter	Meaning
NumberOfEnginesClmm	Enter the desired number of application engines for step 3500. Defaults to 1 application instance.
NumberOfEnginesCkmm	Enter the desired number of application engines for step 5500. Defaults to 1 application instance.
NumberOfQuesClmm	Enter the desired number of work for 3500. **Note:** This equals the total number of queues, not queues per engine.
NumberOfQuesCkmm	Enter the desired number of work for 5500. **Note:** This equals the total number of queues, not queues per engine.
DeadlockRetryAttempts	Enter the number of times a deadlocked transaction attempts database update before bypassing.
DeadlockRetryWaitSeconds	Enter the number of seconds between a deadlocked transaction's database update retry attempts.
SecurityLimit	This identifies the default security limit that stops any claims with a total payable amount equal to or over this amount. This only applies if the user security limit is not set.
JobRetCdThreshold	This indicates the highest return code allowed for subsequent steps.

ErCmcRunCkmm Runbook Parameters

- **InputDataSelectProc**
  - User manually selects a procedure to run
- **Run file includes:**
  - Commented examples for other supplied medical claim selection stored procedures
- **GovernorBreak**
  - Used to prevent large combined claims from causing memory failures in batch

Parameter	Meaning
InputDataSelectProc	The user enters the selection procedure name. The override appears in the run file as: <!--<Item name=”InputDataSelectProc”>CMCSP_CLMM_LOAD_PAY_CL_ALL_MED</ Item>-->

The run file also contains commented examples for the other supplied medical claim selection stored procedures. They include:

- CMCSP_CLMM_LOAD_PAY_CL_MED
- CMCSP_CLMM_LOAD_PAY_CL_HOSP
- CMCSP_CLMM_LOAD_PAY_CL_PR
- CMCSP_CLMM_LOAD_PAY_CL_SB
- CMCSP_CLMM_LOAD_PAY_CL_ITS
- CMCSP_CLMM_LOAD_PAY_CL_NO_ITS
- CMCSP_CLMM_LOAD_PAY_CL_HRA
- CMCSP_CLMM_LOAD_PAY_CL_NO_HRA
- CMCSP_CLMM_LOAD_PAY_CL_BY_GRGR

Parameter	Meaning
GovernorBreak	The Governor Break variable prevents large combined claims from causing memory failures in batch by limiting the number of claims that can combine into one check payment.

Batch failure results when not enough available system resources support a large combined claim's use of memory.

ErCmcRunCkmm Runbook Parameters

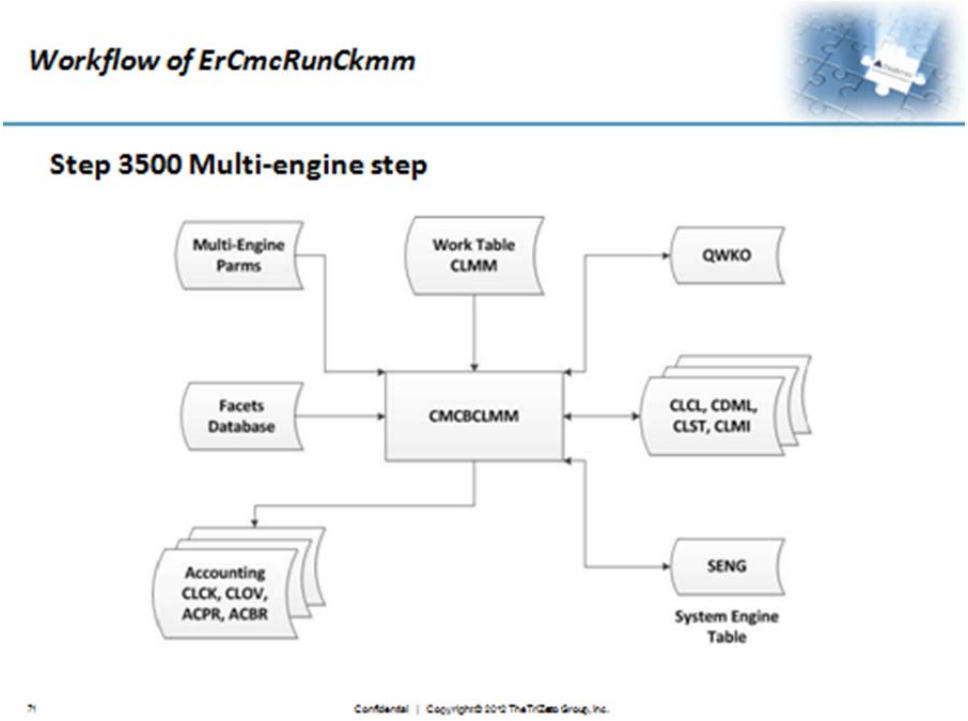
- SecurityLimit
- EngWaitTimeClmm/EngWaitTimeCkmm
- JobRetCdThreshold
- WarningRetCd
- ErrorRetCd

Parameter	Meaning
SecurityLimit	This identifies the default security limit that stops claims with a total payable amount equal to or over this amount. This only applies if the user security limit is not set.
EngWaitTimeClmm/EngWaitTimeCkmm	This identifies the delay time (in seconds) between engine startup for steps 3500 and 5500.
JobRetCdThreshold	This identifies the highest return code allowed for subsequent steps.
WarningRetCd	This identifies the return code used for warning conditions indicated by the CheckQues step that follows the multi-engine step.
ErrorRetCd	This identifies the return code used for error conditions indicated by the CheckQues step that follows the multi-engine step.

Workflow of ErCmcRunCkmm

Workflow of ErCmcRunCkmm

- Step 1500 – If processing claims by group, check if group exists
- Step 1600 – Check date of last batch run
- Step 1700 - ErApp0BatDisplaySyinRefId.vbs
- Step 2100 – Truncate old SENG rows
- Step 2200 – Truncate old QWK0 rows
- Step 2300 – Truncate CLMM table
- Step 3100 – Select claims according to type specified in runbook, in status 01, write to CLMM table
- Step 3200 – Sort data by MEME_CK
- Step 3300 - Create QWK0 rows



For each claim:

- Determine the payee.
- Recoup money owed to the MCO by checking the **CLOV** table and updating ACPR and ACRH (ACBR if alternate funded claims).
- Calculate discounts, fees, interest and apply them to the claim.
- Create CLCK rows for payments per claim.

Workflow of ErCmcRunCkmm

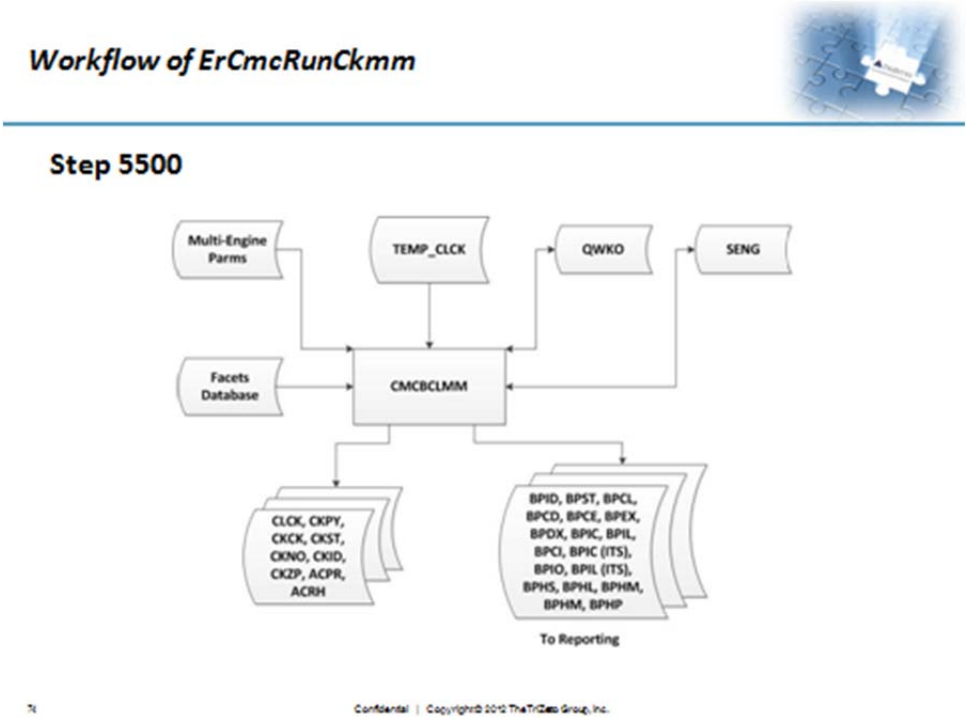
- **Step 3600 – Check QWK0 rows for status 99**
- **Step 3700 – Truncate SENG, QWK0, and CLMM of old rows**
- **Step 3800 – Select rows that deadlocked in step 3500 and create one QWK0 row for cleanup process**
- **Step 3900 – Repeat multi-engine step 3500 with one engine for cleanup process**
- **Step 4000 – Summarize run controls for each engine**
- **Step 4100 – Truncate SENG of old rows**
- **Step 4200 – Truncate QWK0 rows of old rows**

...continued

Workflow of ErCmcRunCkmm

continued…

- **Step 5100 – Sort CLCK rows by LOBD, PAYEE_IND, and COMB_IND**
- **Step 5200 – Write rows to CKMM table**
- **Step 5300 – Create QWK0 rows for processing**
- **Step 5400 - ErApp0BatGetPriorEngines.vbs**



For each CLCK row:

- Check for a combination indicator.
- Create a CKPY row with a CKPY_REF_ID.
- Create a CKCK (check data), CKST (check status), CKID, CKZP, and CKNO (check numbering) rows.
- Update accounting tables, **ACPR** and **ACRH**.
- Create batch print data to support the creation of EOB and reporting data.

Workflow of ErCmcRunCkmm

- **Step 5600 – Check status of QWK0 rows**
- **Step 5700 – Truncate CLCK table and repopulate with any rows are left unprocessed because of deadlocking**
- **Step 5800 – Truncate QWK0 for old rows**
- **Step 5900 – Create one QWK0 row for cleanup**
- **Step 5950 - ErApp0BatGetPriorEngines.vbs**
- **Step 6000 – Repeat of Step 5500 (multi-engine) for cleanup process**

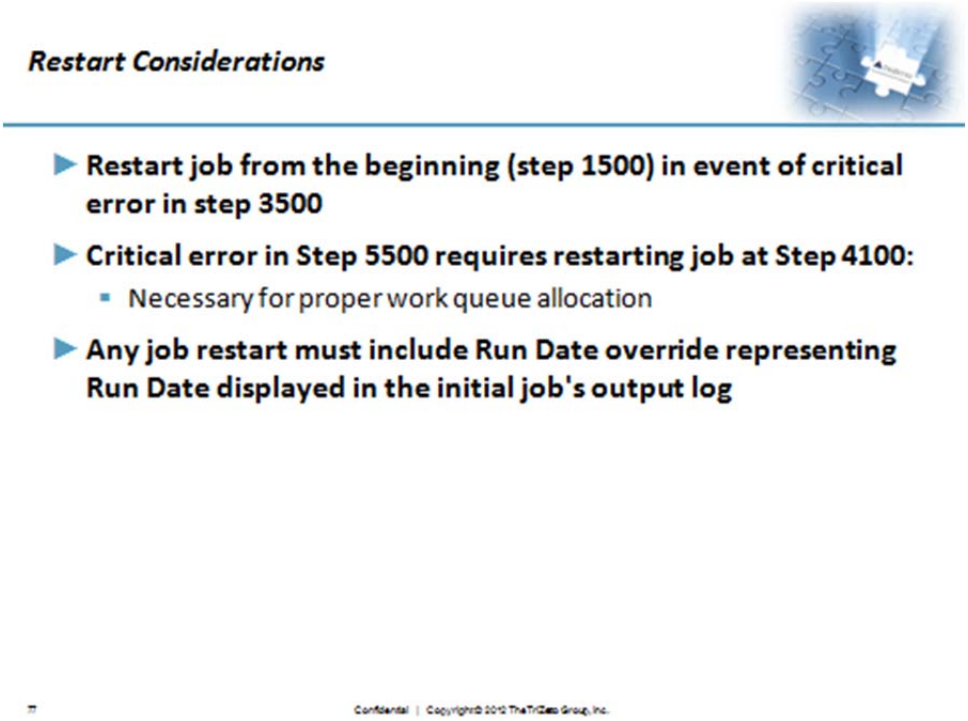
...continued

Workflow of ErCmcRunCkmm

continued…

- **Step 6100 – Update CKCK table with data processed in Step 6000**
- **Step 6200 – Summarize run controls from each engine**
- **Step 9500 – Create the SyinRefID for reporting (optional)**

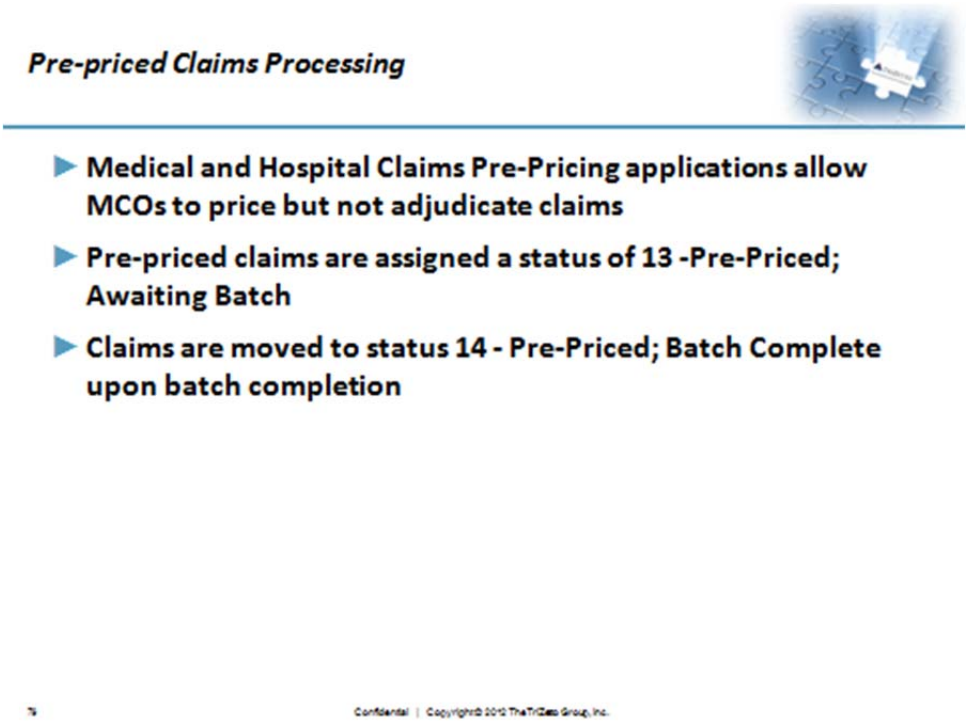
Restart Considerations



Payment batches use control break processing to combine many payment segments into one payment.

Control break CLCK rows from step 3500 break on Ckpy_Pay_Dt, LOBD, Payee_Ind, and when Clck_Comb_Ind = Y-Combine by Payable Entity.

Pre-priced Claims Processing

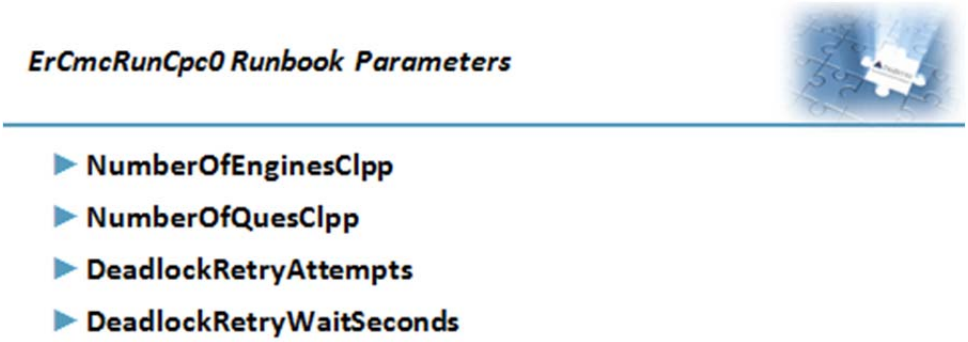


The Medical and Hospital Claims Pre-Pricing applications allow MCOs to price but not adjudicate claims.

Facets assigns pre-priced claims a status of 13 -Pre-Priced; Awaiting Batch. A batch job (ERCMCRUNCPC0) selects this claim status to complete the pre-priced process.

Upon completion of this batch, these claims move to a status 14 - Pre-Priced; Batch Complete.

ErCmcRunCpc0 Runbook Parameters



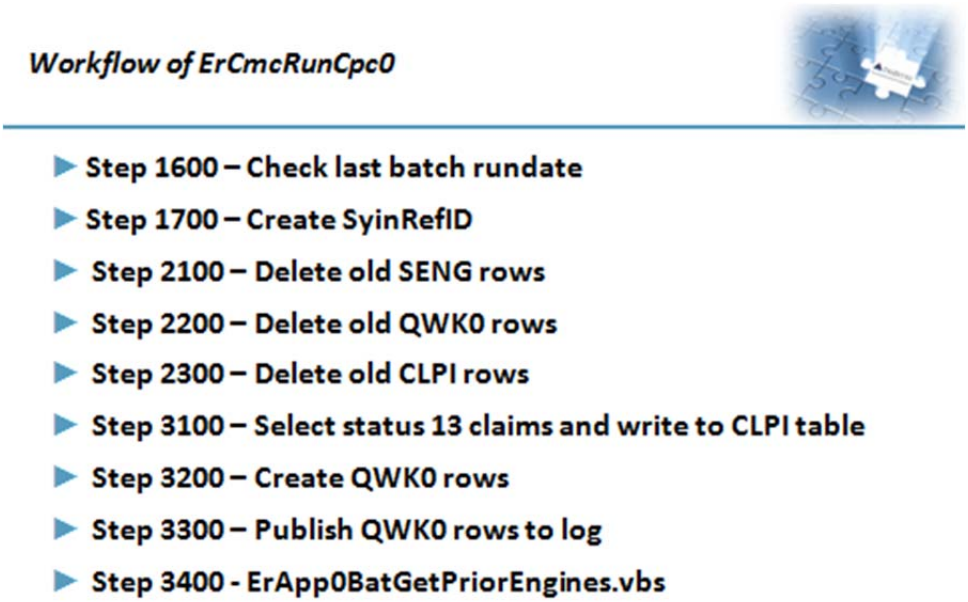
Parameter	Meaning
NumberOfEnginesClpp	The user enters the desired number of application engines for step 3500. The job defaults to one application instance.
NumberOfQuesClpp	The user enters the desired number of work for 3500. **Note:** This is the total number of queues, not queues per engine.
DeadlockRetryAttempts	The user enters the number of times a deadlocked transaction attempts database update before bypassing.
DeadlockRetryWaitSeconds	The user enters the number of seconds between a deadlocked transaction's database update retry attempts.

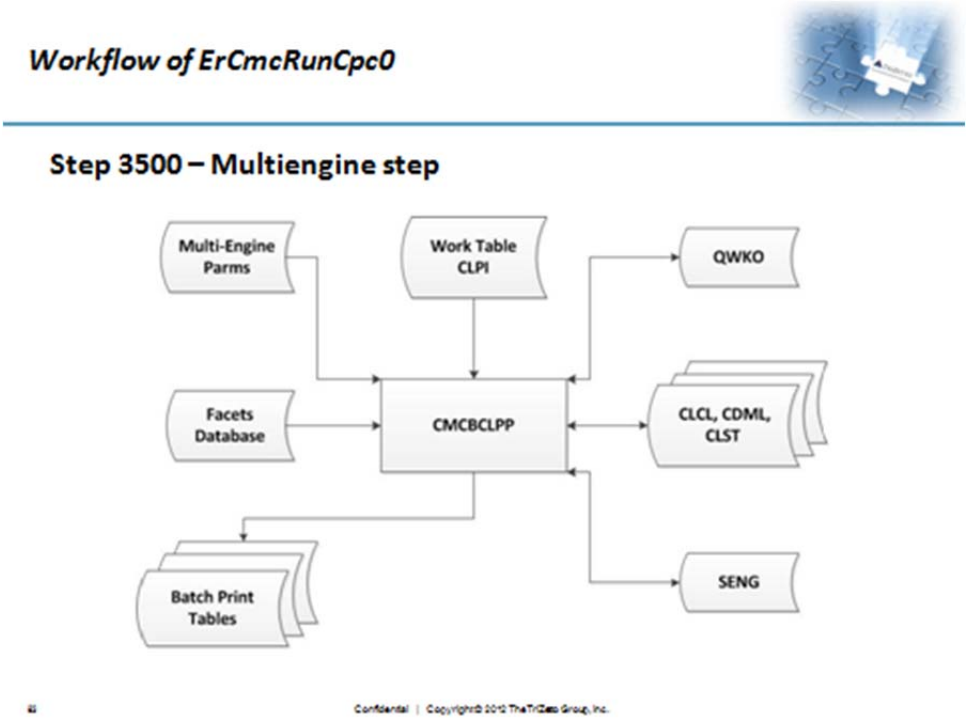
ErCmcRunCpc0

ErCmcRunCpc0

```
->
<!--<Item name="BypassStep">1000</Item>-->
<!--<Item name="RestartStep">1000</Item>-->
<!--<Item name="StopStep">1000</Item>-->
<!--<Item name="RunDate">mm/dd/yyyy  00:00:00.000</Item>-->
<!--<Item name="NumberOfEnginesClpp">1</Item>-->
<!--<Item name="NumberOfQuesClpp">1</Item>-->
<!--<Item name="DeadlockRetryAttempts">1</Item>-->
<!--<Item name="DeadlockRetryWaitSeconds">3</Item>-->
<!--#############################################################
# Step 9500 Display SyinRefId is being run in step 1700 now.
# If you want to still display it in 9500, comment out the BypassStep
#############################################################-->
<Item name="BypassStep">9500</Item>
```

Workflow of ErCmcRunCpc0





For each claim:

1. Calculate discounts or fees.
2. Apply interest.
3. Determine payee.
4. Assign claim status of 14 (paid), and update pay date.
5. Create batch print data for reporting and accounting.

Workflow of ErCmcRunCpc0

- Step 3600 – Check status of QWK0 rows
- Step 3700 – Truncate CLPI table for cleanup process
- Step 3800 – Delete old QWK0 rows
- Step 3900 – Populate CLPI table with deadlocked rows and create one QWK0 row
- Step 3950 - ErApp0BatGetPriorEngines.vbs
- Step 4000 – Repeat multi-engine step (3500) with one engine
- Step 4100 – Summarize run controls from each engine
- Step 9500 – Create SyinRefID for reporting (Optional)

Special Considerations

Special Considerations

Restart:

- **Restart job from beginning (step 1600) in the event of critical error in step 3500**
- **Any restart must include a Run Date override representing Run Date displayed in the initial job's output log**
- **Return code value 8 or existence of transaction errors in output log for given job step indicates critical transaction error:**
  - Use the error log to identify and correct transaction errors

Objective Summary

Upon successful completion of this chapter, you will be able to:

- Explain the purpose of the SYIN_INST ID
- Explain the purpose and structure of the XC database
- Explain the difference between electronic adjudication (CLMU) and claim payment batch processes (CKMM)
- Explain the workflow and relevant batch parameters for each run file/batch process:
  - Erccsrun837i
  - Ercmcrunxc00
  - Ercmcrunclmu
  - Ercmcrunckmm
  - Ercmcruncpc0

Coming Up