

Version 1.0.0

June28, 2022

fedqic Codeshuttle  
Deploy Guide

# Summary of Changes

|  |  |  |
| --- | --- | --- |
| Change | Version | Owner |
| Initial Version | 1.0 | Lavanya Gopal |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

Contents

[Summary of Changes 2](#_Toc107426540)

[1. Purpose 4](#_Toc107426541)

[2. Required Software 4](#_Toc107426542)

[3. Software Installation Steps 4](#_Toc107426543)

[3.1 PENTAHO KETTLE 4](#_Toc107426544)

[4. ETL Code Deployment 7](#_Toc107426545)

[5. Cron Jobs 9](#_Toc107426546)

[6. Verify Deploy 9](#_Toc107426547)

[7. Data Flow Diagram 13](#_Toc107426548)

# Purpose

The purpose of this document is to provide detailed description of the steps involved to move and FedQIC ETL code to Code Shuttle process.

# Required Software

1. Pentaho Kettle 8.3
2. Java
3. Oracle Client

# Software Installation Steps

## 3.1 PENTAHO KETTLE

1. Download Java Version 8 Update 192(build 1.8.0\_192)
2. Download pdi-ce-8.3.0.0-371.zip from <https://sourceforge.net/projects/pentaho/>
3. Download ojdbc8 from  
   <https://www.oracle.com/technetwork/database/application-development/jdbc/downloads/jdbc-ucp-19c-5460552.html>
4. Create folder /u01/app/appadmin/product/pentaho\_8.3
5. Unzip pdi-ce-8.3.0.0-371.zip in the folder /u01/app/appadmin/product/pentaho\_8.3
6. Copy ojdbc8.jar to /u01/app/appadmin/product/pentaho\_8.3/data-integration/lib
7. Download the file svn://svn-staging.maximus.com/dev1d/maxdat/trunk/CAHCO/deploy/jtds-1.3.1.zip.
8. (For reference, this file was originally downloaded from <https://sourceforge.net/p/jtds/bugs/725/>.)
9. Extract the file jtds-1.3.1.jar from jtds-1.3.1.zip
10. Copy the file jtds-1.3.1.jar to /u01/app/appadmin/product/pentaho\_8.3/data-integration/lib (replace if it exists).
11. Remove the file /u01/app/appadmin/product/pentaho\_8.3/data-integration/lib/jtds-1.2.5.jar, if it exists
12. Update /u01/app/appadmin/product/pentaho\_8.3/data-integration/spoon.sh

  After the line below:

setPentahoEnv

Add the line below:

\_PENTAHO\_JAVA='/u01/app/appadmin/product/java/jdk1.8.0\_192/bin/java'

Change this at the bottom of the file:

if [ -z ""$PENTAHO\_DI\_JAVA\_OPTIONS"" ]; then

PENTAHO\_DI\_JAVA\_OPTIONS=""-Xms1024m -Xmx2048m -XX:MaxPermSize=256m""

TO:

if [ -z ""$PENTAHO\_DI\_JAVA\_OPTIONS"" ]; then

PENTAHO\_DI\_JAVA\_OPTIONS=""-Xms1024m -Xmx4096m -XX:MaxPermSize=256m""

1. Rename the following files and directories in /u01/app/appadmin/product/pentaho\_8.3/data-integration if exists using the following format:

<Name>.backup.yyyymmdd for example: kettle-lifecycle-listeners.xml.backup.20190129

/u01/app/appadmin/product/pentaho\_8.3/data-integration/classes/kettle-lifecycle-listeners.xml

/u01/app/appadmin/product/pentaho\_8.3/data-integration/classes/kettle-registry-extensions.xml

/u01/app/appadmin/product/pentaho\_8.3/data-integration/classes/log4j.xml

/u01/app/appadmin/product/pentaho\_8.3/data-integration/lib  
pdi-engine-api-8.3.0.0-371.jar  
pdi-osgi-bridge-core-8.3.0.0-371.jar  
pentaho-connections-8.3.0.0-371.jar  
pentaho-osgi-utils-api-8.3.0.0-371.jar  
pentaho-cwm-1.5.4.jar  
org.apache.karaf.jaas.boot-3.0.3.jar  
org.apache.karaf.main-3.0.3.jar  
org.apache.karaf.util-3.0.3.jar  
mondrian-8.3.0.0-371.jar

/u01/app/appadmin/product/pentaho\_8.3/data-integration/plugins/pentaho-big-data-plugin

/u01/app/appadmin/product/pentaho\_8.3/data-integration/plugins/kettle5-log4j-plugin/log4j.xml

/u01/app/appadmin/product/pentaho\_8.3/data-integration/system/karaf

/u01/app/appadmin/product/pentaho\_8.3/data-integration/system/mondrian

/u01/app/appadmin/product/pentaho\_8.3/data-integration/system/osgi

1. Copy below file under /u01/app/appadmin/product/pentaho\_8.3/data-integration/plugins/kettle5-log4j-plugin/ dir and rename file to log4j.xml

svn://svn-staging.maximus.com/dev1d/maxdat/trunk/Kettle8/Config/kettle5-log4j-plugin\_log4j.xml

1. Copy below file under /u01/app/appadmin/product/pentaho\_8.3/data-integration/classes/ directory and rename to log4j.xml

svn://svn-staging.maximus.com/dev1d/maxdat/trunk/Kettle8/Config/classes\_log4j.xml

1. Open the /u01/app/appadmin/product/java/jdk1.8.0\_192/jre/lib/security/java.security file in a text editor.

Change the line and save the file.

securerandom.source=[file:/dev/random](file:///\\dev\random)

to read:

securerandom.source=[file:/dev/urandom](file:///\\dev\urandom)

# ETL Code Deployment

1. Create the following directories under : /u01/maximus/maxdat-prd

/u01/maximus/maxdat-prd/FEDQIC8

/u01/maximus/maxdat-prd/FEDQIC8/bin

/u01/maximus/maxdat-prd/FEDQIC8/deploy

/u01/maximus/maxdat-prd/FEDQIC8/input

/u01/maximus/maxdat-prd/FEDQIC8/input/archive

/u01/maximus/maxdat-prd/FEDQIC8/config

/u01/maximus/maxdat-prd/FEDQIC8/config/.kettle

/u01/maximus/maxdat-prd/FEDQIC8/logs

1. Download kettle.properties file from:

svn://svn-staging.maximus.com/dev1d/maxdat/trunk/Kettle8/FEDQIC/config/PRD/.kettle/kettle.properties

And deploy to:

/u01/maximus/maxdat-prd/FEDQIC8/config/.kettle

1. Replace clear text passwords with encrypted passwords in kettle.properties
2. Execute: /u01/app/appadmin/product/pentaho\_8.3/data-integration/encr.sh -kettle <clear text password> for each of the passwords below using the kettle.properties file from the /u01/maximus/maxdat-prd/FEDQIC8/config/.kettle directory
3. Note if the password contains a special character &,,$, etc. Place single quotes around it.
4. Take the output and replace the clear text passwords with the encrypted password produced by the encr.sh script.
5. Note: Ensure the complete encrypted password is copied. The password starts with the word 'Encrypted '
6. Download shared.xml file from:

svn://svn-staging.maximus.com/dev1d/maxdat/trunk/Kettle8/FEDQIC/config/PRD/.kettle/shared.xml

And deploy to:

/u01/maximus/maxdat-prd/FEDQIC8/config/.kettle

Download the following zip file

svn://svn-staging.maximus.com/dev1d/maxdat/trunk/Kettle8/FEDQIC/scripts\_PRD.zip

Unzip scripts.zip into the /u01/maximus/maxdat-prd/FEDQIC8/ directory

1. Set permissions on all FEDQIC8 folder/sub-folders similar to FEDQIC folder/sub-folders

Convert all shell script files to unix format (dos2unix \*.sh) in these directories:

/u01/maximus/maxdat-prd/FEDQIC8/scripts

1. Copy the following file **if it exists** from /u01/maximus/maxdat-prd/FEDQIC/input/supervisors.csv to /u01/maximus/maxdat-prd/FEDQIC8/input/supervisors.csv
2. Copy all the files from /u01/maximus/maxdat-prd/FEDQIC/input/archive to /u01/maximus/maxdat-prd/FEDQIC8/input/archive
3. Assign for validation

# Cron Jobs

Coordinate with developer to make sure no jobs are running and replace the current CRON with the new file from:   
svn://svn-staging.maximus.com/dev1d/maxdat/trunk/Kettle8/FEDQIC/CronJob\_Kettle8.3\_PRD.txt

# Verify Deploy

1. Alter Session to MAXDAT.

alter session set current\_schema = MAXDAT;

1. Check Job Statistics table

  SELECT js.JOB\_ID, js.JOB\_NAME, js.JOB\_STATUS\_CD, js.PROCESSED\_COUNT, js.RECORD\_UPDATED\_COUNT, to\_char(js.JOB\_START\_DATE,'mm/dd/yyyy hh:mi:ss am'),

to\_char(js.JOB\_END\_DATE,'mm/dd/yyyy hh:mi:ss am'),

Round(((js.JOB\_END\_DATE-js.JOB\_START\_DATE)\*24\*60),0) elapseMinutes

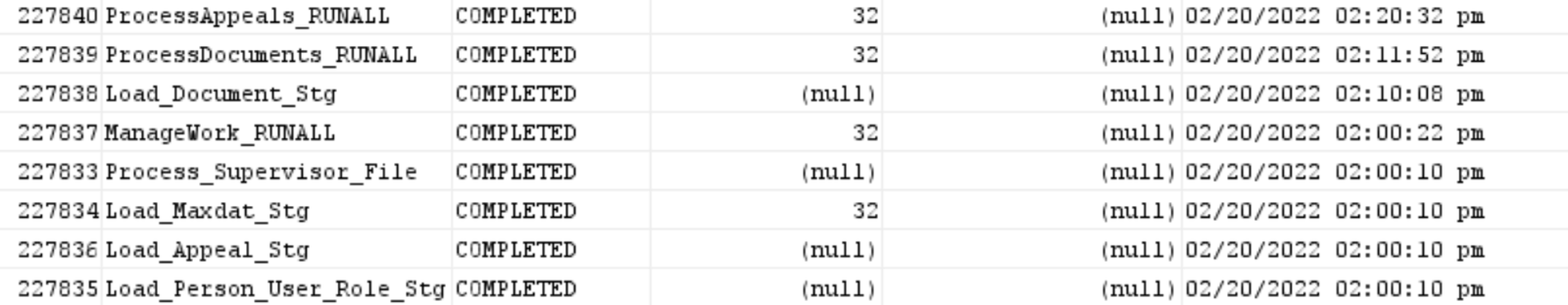
FROM CORP\_ETL\_JOB\_STATISTICS js

WHERE

js.JOB\_START\_DATE > TRUNC(SYSDATE-1)

ORDER by  js.JOB\_START\_DATE DESC;

1. Make sure the following job entries are COMPLETED in the job\_statistics table



1. Check BPM Queue

  Select

  --bsl.BSL\_ID,

  bsl.NAME "Staging Table Name",

--  count(bueq.BSL\_ID) "Total",

  sum(case when bueq.WROTE\_BPM\_SEMANTIC\_DATE is null then 1 else 0 end) "Semantic Pending",

  --sum( case when bueq.WROTE\_BPM\_SEMANTIC\_DATE is not null then 1 else 0 end) as Not\_Archived,

  nvl(round(((sysdate - min(EVENT\_DATE)) \* 24),2),0) "Hours Delay",

  nvl(to\_char(min(EVENT\_DATE),'YYYY-MM-DD HH24:MI'),'Current') "Oldest Unprocessed"

from BPM\_SOURCE\_LKUP bsl

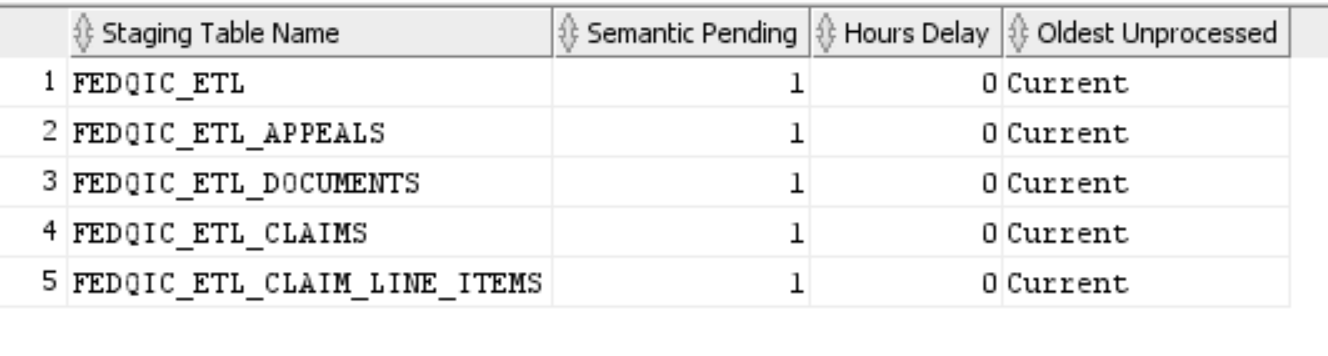
left outer join BPM\_UPDATE\_EVENT\_QUEUE bueq on bsl.BSL\_ID = bueq.BSL\_ID

--and bsl.BSL\_ID = 15

group by bsl.BSL\_ID,NAME

order by bsl.BSL\_ID asc;

1. Make sure all the rows are processed in the Queue:



1. a)  If there are rows that are stuck in BPM Queue then try resetting the BPM Queue:

execute maxdat.MAXDAT\_ADMIN.RESET\_BPM\_QUEUE\_ROWS; ( Reset the stuck rows which are unprocessed in the queue )

b) If the rows are still not cleared then try shutting down the BPM Queue and restart it.

execute maxdat.MAXDAT\_ADMIN.SHUTDOWN\_JOBS

execute maxdat.MAXDAT\_ADMIN.STARTUP\_JOBS***;***

c) If the rows are still not clear then try finding out the error by running the following:

select to\_char(log\_date,'mm/dd/yyyy hh24:mi:ss AM'),bpml.\* from maxdat.bpm\_logging bpml order by log\_date desc;

This gives the old and new values that are stuck in the queue.

select q.new\_data.getClobVal(), q.\* from maxdat.bpm\_update\_event\_queue q --where identifier = 11779500;

\*\* bueq\_id is from bpm\_update\_event\_queue , get identifier from bpm\_logging table for the error record

1. Log files can also be checked to make sure there are no errors:

Log File Directory - /u01/maximus/maxdat-prd/FEDQIC8/logs

1. Verify if all jobs are running as expected after clearing the PIDs and resetting the queue process.

# Data Flow Diagram

