## 



Project Name

EP ETL

Prepared By:

Changepond team

# Overview

This is an ETL Project having two .ktr file,two .kjb file, one .properties file and one schema.sql file for schema creation.

# Goals

1. To move the existing data to new archive DB.

# Requirements:

1. Connection to svn.
2. Pentaho version 6.0.1
   1. Links to download : <http://sourceforge.net/projects/pentaho/> or <http://www.pentaho.com/download>

# Setting Environment Variables:

Go to system advanced settings and or from user account → change my environment variables.

Click on new in user variables.

Give variable name as: “KETTLE\_HOME”

Give variable value as:” Path where you want to store the .kettle folder.”

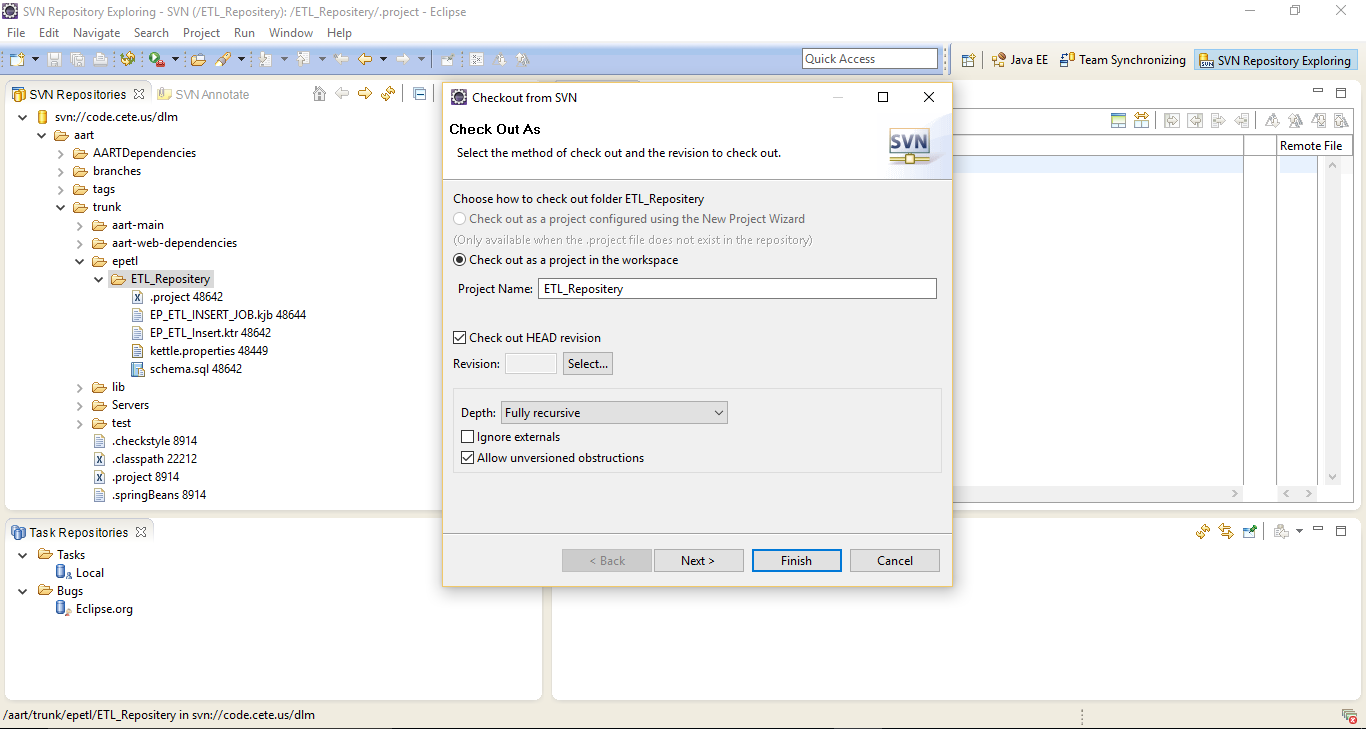
.kettle folder will have all related properties and XML file for environment variables.

Set one more variable as EP\_APP\_HOME and provide the path. In this path only the project will be built and all the required files will be copied here.

# 

# Steps to Checkout project:

* 1 Connect to svn via Eclipse to the below specified path:
  + svn://code.cete.us/dlm/aart/trunk/epetl/
* 2. Select ETL\_Repositery and checkout from ETL\_Repositery. A new dialog box will open which is given below:
* Select checkout as a project in the workspace.
* Mention the project name you want to mention.
* Click next and finish.



# Setting up properties file:

Move your kettle.properties file to .kettle folder and set all variables for database and schoolyear.

Edit kettle.properties file as per database:

#Source database Connection credentials:

ep.source.db.name = XXXXXX

ep.source.db.host = localhost

ep.source.db.port = NNNN

ep.source.db.userid = XXXX

ep.source.db.password = XXXXXX

#target Database Connection credentials:

ep.target.db.name = EP\_ETL

ep.target.db.host = localhost

ep.target.db.port = NNNN

ep.target.db.userid = XXXX

ep.target.db.password = XXXXXX

#Steps to set the log file path

ep\_etl\_log = “**Give your log file path location here upto file name without extension “(like etc\\srv\\etl\\log\\etl.log)**

# Steps to Run Project:

* Create Database with name EP\_ETL.
* Run the sql from schema.sql connecting this new database. (this is will create all the tables)
* Insert data in the “dataarchivestatus” table for the year to be processed.
* In the command from go the place where we checked out the project and enter “ant clean copy-mappings “

– This will delete the existing deployment and create the new version. (This command will copy and place the entire job file, KTR file, property files to EP\_APP\_HOME’s path)

Now, in command prompt and go to the path where your kitchen.bat file is located.

# Check your Pentaho version:

Kitchen .bat -version

# Run a Job from file

This example runs a transformation from file on a **Linux box**:

**For Archiving:**

**kitchen.sh -file="/<ETL\_Repository> /dataarchivejob.kjb" -level=Minimal**  
**\* here ="/<ETL\_Repositery> is the folder located in our EP\_APP\_HOME’s path**

**For Deleting:**

**kitchen.sh -file="/<ETL\_Repository> /deletejob.kjb" -level=Minimal**  
**\* here ="/<ETL\_Repositery> is the folder located in our EP\_APP\_HOME’s path**

**Redirecting output**

If you don't want the output of the file to appear on the screen but rather be put into a log file, you can use redirection.

**For Archiving:**

kitchen.sh -file**="/<ETL\_Repository> /dataarchivejob.kjb "**-level=Minimal >> /LOG/trans.log

This example writes the Kitchen output to a file that gets overwritten every time:

kitchen.bat /file: **="/<ETL\_Repository> /dataarchivejob.kjb"**/level:Basic > C:\LOG\trans.log

**For Deleting:**

kitchen.sh -file**="/<ETL\_Repository> /deletejob .kjb "**-level=Minimal >> /LOG/trans.log

This example writes the Kitchen output to a file that gets overwritten every time:

kitchen.bat /file: =**"/<ETL\_Repository> /deletejob .kjb"**/level:Basic > C:\LOG\trans.log

# Return codes

Kitchen returns an error code based on how the execution went:

* 0 : The transformation ran without a problem.
* 1 : Errors occurred during processing
* 2 : An unexpected error occurred during loading / running of the transformation
* 3 : Unable to prepare and initialize this transformation
* 7 : The transformation couldn't be loaded from XML or the Repository
* 8 : Error loading steps or plugins (error in loading one of the plugins mostly)
* 9 : Command line usage printing

# Note:

* In dataarchivestatus table, records with insertstatus value is false alone will be processed.