Automate the Import Bulk Data Process

Flowchart

START

Iterate for each entity:

- Pre-validate the source data against and Get Clean Data
- Transform Data by applying all rules, filter clauses to build the SQL Query
- Execute the SQL Query dynamically and Export Data in CLOB
- Write CLOB to a .csv File
- Zip all .csv(s) for an object as BLOB
- Convert BLOB to Base64 encoding
- Prepare the parameter list for Import Ess Job
- Prepare the SOAP Payloa for importBulkData of ERP Int. Svc.
- Invoke importBulkData opertaion & Receive SOAP Response
- Parse response to get request Id of Load Interface File for Import Job

RUNNING:

- Get status of Load Interface File for Import Job
- Check Status for N times at X secs interval — **SUCCEEDED** / **ERROR/ WARNING/ CANCELLED**
- Invoke ReportService to derive the request if of Import Job

RUNNING:

- Get status of Import Job
- Check Status for N times at X secs interval — **SUCCEEDED** / **ERROR/ WARNING/ CANCELLED**
- Invoke ReportService to download the reconciliation report output
- Place the file DBA Directory/ DB Table/ Object Storage

END

Logic Implemented:

1. This automation process starts with pre-validating data or assessing the data quality to get the clean data by discarding the bad or error data.

2. Apply all the transformation rules e.g. default rules, derivation rules, filter conditions on the clean data. Basically a SQL Statement is prepared.

3. Execute the SQL Statement being created at Step #2 dynamically to get the data set and export as CLOB.

4. Write the data in CLOB to a .csv file and place in a DBA Directory.

5. Iterate Step #1 to Step #4 for each entity (FBDI sheet) for the object concerned.

6. Zip all the .csv(s) for the object as BLOB.

7. Convert the BLOB to Base64 encoded text.

8. Prepare the parameter list for Import ESS Job.

9. Prepare the SOAP Payload for importBulkData operation of ERPIntegrationService.

10. Invoke importBulkData operation of ERPIntegrationService and receive SOAP response.

11. Parse response to get request Id of Load Interface File for Import Job.

12. Get status of Load Interface File for Import Job in a loop where it will **iterate for N times at interval of X secs** and **exit when program ends with either of the statuses SUCCEEDED or ERROR or WARNING or CANCELLED**.

13. If the status from Step #12 is SUCCEEDED, then Invoke ReportService to derive the request id of the Import Job. Else, abort the process

14. Get status of Import Job in a loop where it will **iterate for N times at interval of X secs** and **exit when program ends with either of the statuses SUCCEEDED or ERROR or WARNING or CANCELLED**.

15. If the status from Step #14 is SUCCEEDED, then Invoke ReportService to download the Reconciliation Report output. Else, abort the process

16. Place the downloaded report output to a DBA Directory or Object Storage or DB table.