List\_850\_reprocess\_files (1.2.0)

Author: <Radhika>

Creation Date: May 31, 2021

Last Updated: May 31, 2021

Document Ref: <D>

Version: 1

1. **Title, Subject, Last Updated Date, Reference Number**, **and** **Version** are marked by a Word Bookmark so that they can be easily reproduced in the header and footer of documents. When you change any of these values, be careful not to accidentally delete the bookmark. **You can make bookmarks visible by selecting the Office Button>Word Options>Advanced> and checking the Show bookmarks option in the Show document content region.**

**Approvals:**

|  |  |
| --- | --- |
| <Approver 1> |  |
| <Approver 2> |  |

1. To add additional approval lines, press [Tab] from the last cell in the table above.

oracle.gif

1. You can delete any elements of this cover page that you do not need for your document.

# Document Control

## Change Record

5

| Date | Author | Version | Change Reference |
| --- | --- | --- | --- |
| 31-May-21 | <Radhika Thipparthi> |  | No Previous Document |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

## Reviewers

| Name | Position |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |

Contents

1 Document Control ii

1.1 Change Record ii

1.2 Reviewers ii

2 Technical Overview 1

2.1 Building Blocks 1

2.2 Block Relationship Diagram 1

3 Connections………………………………………………………………………..2

4. Interface Design 3

5 Open and Closed Issues 5

5.1 Open Issues 5

5.2 Closed Issues 5

1. To update the table of contents, put the cursor anywhere in the table and press [F9]. To change the number of levels displayed, select the menu option Insert‑>Index and Tables, make sure the Table of Contents tab is active, and change the Number of Levels to a new value.

# Technical Overview

1. The purpose of this task is to assemble all the information that is required to describe the design of a software component into a complete Design Specification. This task is not a substitute for executing the individual design tasks. This specification work product can serve as a structure for completing the design for each component by providing pointers back into the Design Tasks:   
   - DS.040 Develop Design Architecture Description  
   - DS.080 Design Software Components  
   - DS.090 Design Data  
   - DS.100 Design Behavior  
   - DS.130 Design User Interface

This Design Specification documents detailed design for Oracle fusion Cloud(OFC) to Oracle Commerce Cloud(OCC) List\_850\_reprocess\_files (1.2.0) interface.

* Oracle Fusion Cloud application is used to getting Credit files from ERP and this data is interfaced to Oracle Commerce Cloud to upload credit limit for the customer. To achieve this, Integration needs to be established between ERP and Cloud Commerce for update the credit limit.
* This interface is used to updates the Credit limit for the Customer’s in OCC.

## Building Blocks

1. The intent of this section is to list the building blocks that are required to design the designated component. This includes classes, objects, modules, etc. Reference the Module View of the Architecture Description (RD.130) and appropriate Software Component Design (DS.080) to derive the list of classes and their relationships.

Building Blocks

* + OCC – Oracle Commerce Cloud
  + OIC – Oracle Integration Cloud
  + OFC – Oracle Fusion Cloud

## Block Relationship Diagram

1. The intent of this section is to graphically depict how the component under consideration interfaces to related components, external systems, and other actors that interact with the use-case package. Reference the Conceptual View and Module View of the Architecture Description (RD.130) and the class diagram prepared in the Software Component Design (DS.080) and Component Behavior Design (DS.100).

The diagram below represents the base tables of each block or zone of the form (vertical) and tables referenced for validation or lookups (horizontal).

Oracle Cloud

Commerce

Oracle

Integration

Cloud

Oracle

Fusion

Cloud

1. The intent of this section is to identify the table, columns, and source values that are required to support the above data elements. Refer to the Physical Database Design (IM.040), to identify the existing tables where the above attributes are located.
2. The intent of this section is to define the design considerations necessary to achieve the data retrieval and storage requirements for performance. Include performance requirements as specified in the Supplemental Requirements (RD.055) for service level requirements (i.e., 1-minute response time, etc.)
3. The intent of this section is to define the implementation strategy for each business rule within this component. Refer to the DS.110 Business Rules Design to capture the Business Rules for this component.

# CONNECTIONS

# Query

# Interface Design

1. The intent of this section is to design the services between the components and the interfaces with external systems for each Use Case. Refer to DS.080 Software Component Design and focus on calling arguments (i.e., service signature) and logic definition.

Overview description: Interface is Scheduled based interface.

1. Assign global variables testEmail, prodEmail
2. Write headers by using FTP adapter.
3. The XML SPR unprocessed files are expected to be placed at the ‘**/home/spr/in/unprocessed**’ directory. List files by using FTP adapter and configure as below.

* Enter the name 🡪 ListSPRUnprocessed.
* Basic Info 🡪 Enter endpoint name.

1. Check 2 conditions.
2. Check if ItemCount>0.0, then for each SPR file download SPR file by using FTP adapter.
3. Read SPR file by using Stage file.
4. Check 2 conditions.

* Check if substring(PO\_ID, 1.0, 1.O) =”0”, then write SPR file PO Text by using stage file.



* Otherwise, write SPR file by using stage file. For loop ends.

1. Otherwise, go to next step.
2. List LCI unprocessed files in the ‘**/home/lci/in/unprocessed**’ directory. List files by using FTP adapter and configure as below.

* Enter the name 🡪 ListLCIUnprocessed.
* Basic Info 🡪 Enter endpoint name.

1. Check 2 conditions.
2. Check if ItemCount>0.0, then for each LCI file download LCI file by using FTP adapter.
3. Read LCI file by using Stage file.
4. Check 2 conditions.

* Check if substring(PO\_ID, 1.0, 1.O) =”0”, then write LCI file PO Text by using stage file.



* Otherwise, write LCI file by using stage file. For loop ends.

1. Otherwise, go to next step.
2. List HDSupply unprocessed files in the ‘**/home/hdsupply/in/unprocessed**’ directory. List files by using FTP adapter and configure as below.

* Enter the name 🡪 ListHDSupplyUnprocessed.
* Basic Info 🡪 Enter endpoint name.

1. Check 2 conditions.
2. Check if ItemCount>0.0, then for each HDSupply file download HDSupply file by using FTP adapter.
3. Read HDSupply file by using Stage file.
4. Check 2 conditions.

* Check if substring(PO\_ID, 1.0, 1.O) =”0”, then write HDSupply file PO Text by using stage file.



* Otherwise, write HDSupply file by using stage file. For loop ends.

1. Otherwise, go to next step.
2. List IFB unprocessed files in the ‘**/home/ifb/in/unprocessed**’ directory. List files by using FTP adapter and configure as below.

* Enter the name 🡪 ListIFBUnprocessed.
* Basic Info 🡪 Enter endpoint name.

1. Check 2 conditions.
2. Check if ItemCount>0.0, then for each IFB file download IFB file by using FTP adapter.
3. Read HDSupply file by using Stage file.
4. Check 2 conditions.

* Check if substring(PO\_ID, 1.0, 1.O) =”0”, then write IFB file PO Text by using stage file.



* Otherwise, write IFB file by using stage file. For loop ends.

1. Otherwise, go to next step.
2. List SPS unprocessed files in the ‘**/out/Unprocessed\_850**’ directory. List files by using FTP adapter and configure as below.

* Enter the name 🡪 ListSPSFiles.
* Basic Info 🡪 Enter endpoint name.

1. Check 2 conditions.
   1. Check if ItemCount>0.0, then for each SPS file download SPS file by using FTP adapter.

For each SPS file,

* Check 3 conditions.

1. Check if



Then read MSC file by using stage file and again check 2 conditions.

* Check if



Then write MSC file PO text by using stage file.

* Otherwise, then write MSC file by using stage file.

1. Check if



Then read Office Depot file by using stage file and again check 2 conditions.

* Check if



Then write OFD file PO text by using stage file.

* Otherwise, then write OFD file by using stage file.

1. Otherwise, read Grainger file by using stage file and again check 2 conditions.

* Check if



Then write Grainger file PO text by using stage file.

* Otherwise, then write Grainger file by using stage file and for loop ends here. Got to next step.
  1. Otherwise, go to next step.

1. List NIB unprocessed files in the ‘**/home/abilityone/in/unprocessed**’ directory. List files by using FTP adapter and configure as below.

* Enter the name 🡪 ListNIBUnprocessed.
* Basic Info 🡪 Enter endpoint name.
  1. Check 2 conditions.

1. Check if ItemCount>0.0, then for each NIB file download NIB file by using FTP adapter.

* Read NIB file by using stage file.

Check 2 conditions.

1. Check if substring(PO\_ID, 1.0, 1.O) =”0”, then write NIB file PO Text by using stage file.



1. Otherwise, write NIB file by using stage file. For loop ends.
2. Otherwise, go to next step.
3. List NIB unprocessed files in the ‘**/home/blainewarren/in/unprocessed**’ directory. List files by using FTP adapter and configure as below.

* Enter the name 🡪 ListBWFiles.
* Basic Info 🡪 Enter endpoint name.
  1. Check 2 conditions.

1. Check if ItemCount>0.0, then for each NIB file download NIB file by using FTP adapter.

* Read NIB file by using stage file.
* Write NIB file PO text by using stage file. For loop ends.

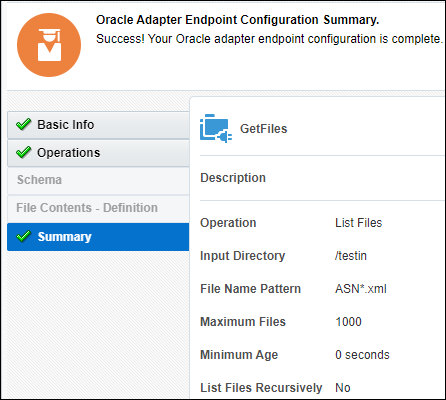
1. Otherwise, go to next step.
2. List Essendant unprocessed files in the ‘**/home/essendant3/in/unprocessed**’ directory. List files by using FTP adapter and configure as below.

* Enter the name 🡪 ListEssFiles.
* Basic Info 🡪 Enter endpoint name.
  1. Check 2 conditions.

1. Check if ItemCount>0.0, then for each Essendant file download Essendant file by using FTP adapter.

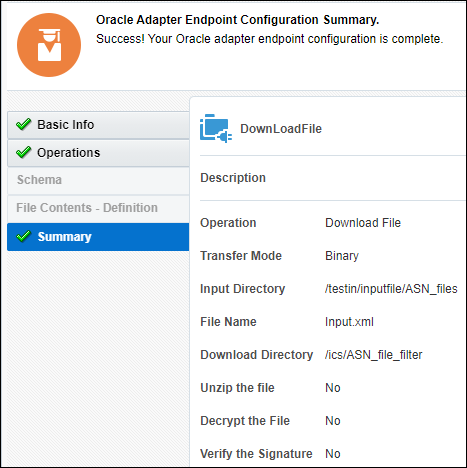
* Read Essendant file by using stage file.
* Write Essendant file PO text by using stage file. For loop ends.

1. Otherwise, go to next step.
2. Send notification reprocess 850 files to the team.
3. Terminate the job.

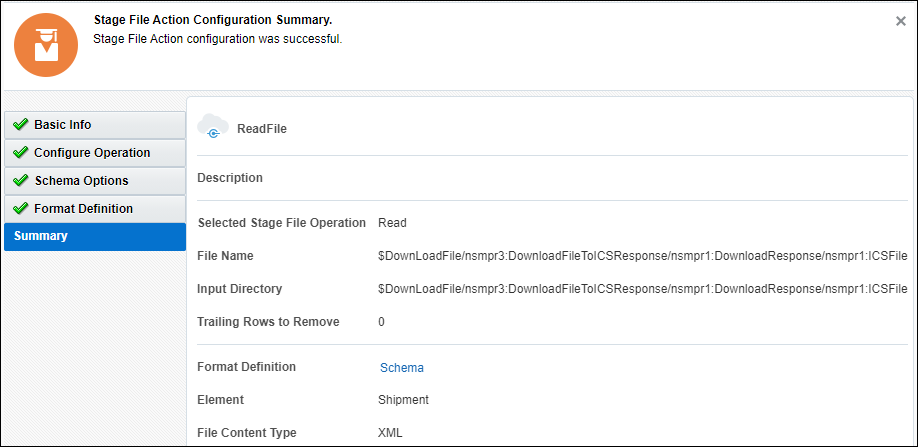


Step2: For each file, Scope of the items(File\_scope)

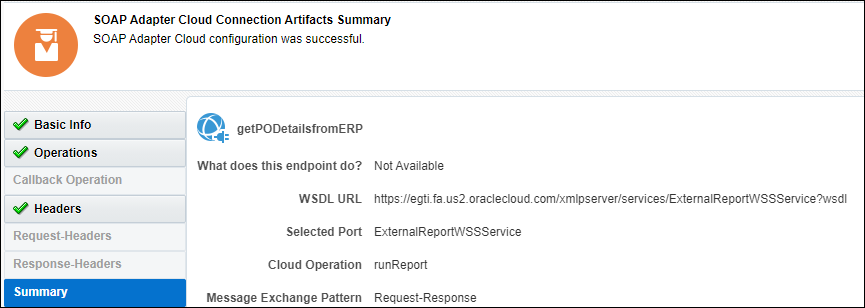
1. Download files by using FTP adapter.



1. Read files by using Stage file.

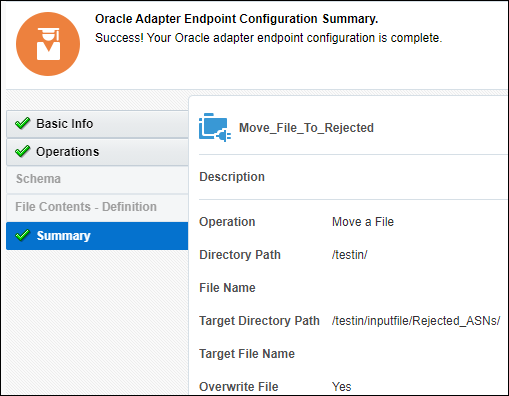


1. Assign variables Query\_Part\_1, Query\_Part\_2.
2. Get PO Details from ERP by using SOAP adapter.

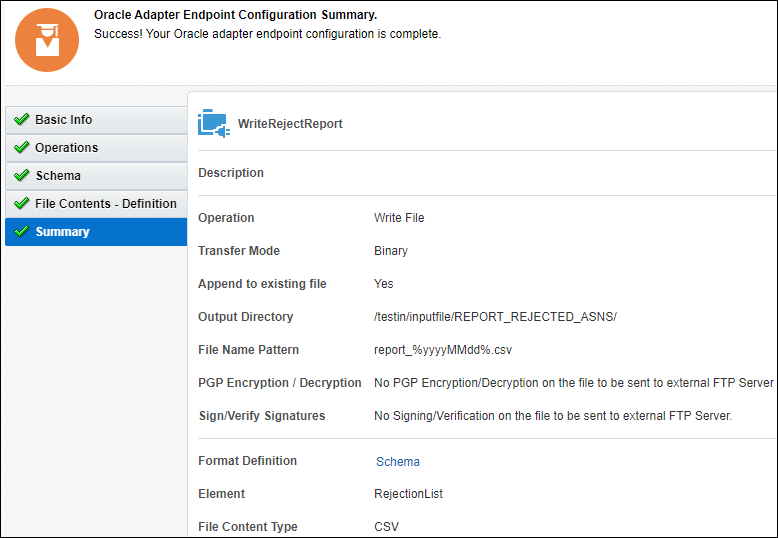


1. Check 2 conditions.
   1. Check if,

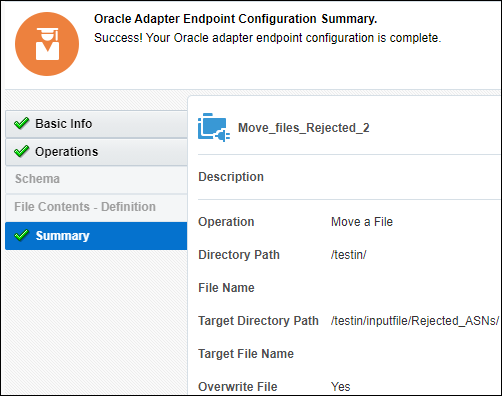
* (No data found) [reportBytes = "UE9OVU1CRVIsRE9DVU1FTlRfU1RBVFVTLExJTkVfTlVNLFZFTkRPUl9QUk9EVUNUX05VTSxJVEVN X05VTUJFUixJVEVNX0RFU0NSSVBUSU9OLFVPTV9DT0RFLE5FVF9RVUFOVElUWSxDSEFOR0VEX1BB](https://ibvi-oic-ibvioic.integration.ocp.oraclecloud.com/ic/integration/home/faces/link?page=integration&code=REPROC_ASN_INBOUN_FILE_FILTER&version=03.01.0001&mode=VIEW_MODE&consumer_url=https://ibvi-oic-ibvioic.integration.ocp.oraclecloud.com/ic/home?root=integrations&oj_Router=1N4IgDghg5gpiBcoBuMBOBnAlgewHYJAAYBmAOkIEZzDKQAaEAY2wBM54QAlAUQAVOA8gGEA+gEEAygDkRASSkAhAQFUZAMVkAZbiI2aAKt070QAW1bsQANVncA6iICyAgCLcT5xgGtHFhLgBXABsghjAggKhMXCs0LDx-YKCAX2SgA), then move file rejected directory ‘**/out/856/ARCHIVE/REPROCESSED\_ARCHIVE/testArchive/ReRejected**’ by using FTP adapter.



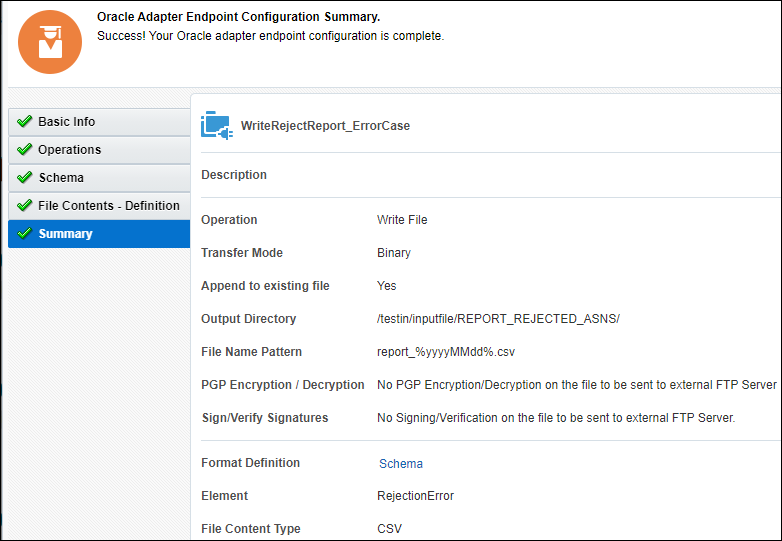
* Write rejected files by using FTP adapter and go to next step.



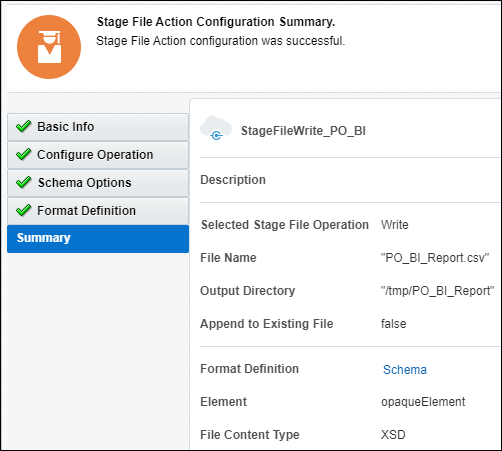
* 1. Check if,
* (Error) 1.0!=1.0, then move the files to directory ‘**/out/856/ARCHIVE/REPROCESSED\_ARCHIVE/testArchive/ReRejected**’.



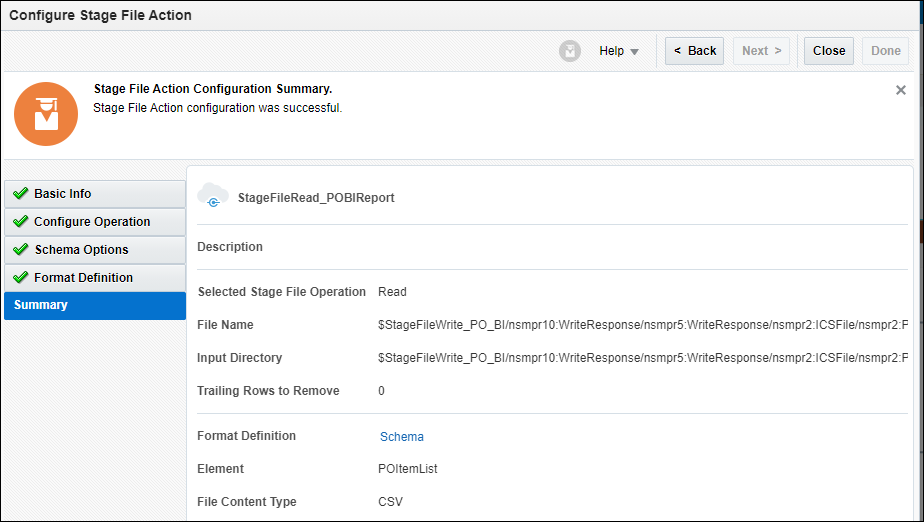
* Write rejected files by using FTP adapter and go to next step.



* 1. Otherwise, write PO BI report by using Stage file.



* Read PO BI report by using Stage file. Assign variables VENDOR\_NAME, BUYER and go to next step.



1. Scope of the items(Verify\_TrackingID\_Scope)

Assign variables wayBillNumber, wayBillExistsNumInERP.

Scope of the items(waybill\_query\_Scope)

Check 3 conditions.

* Check if String-length(ShippingSerialID>0.0), then assign wayBillNumber and go to next step.
* Check if String-length(CarrierPackageID>0.0), then assign wayBillNumber and go to next step.
* Otherwise, go to next step.

1. Get Waybill details by using SOAP adapter.
2. Write waybill by using Stage file.
3. Check 3 conditions.

* Check if no data found, then assign wayBillNumExistsInERP.
* Check if success, then read Waybill result by using Stage file and assign wayBillNumExistsInERP.
* Otherwise, go to next step.

1. Check 3 conditions.
2. Check if String-length(ShippingSerialID>0.0 and $wayBillNumExistsInERP=’YES’), then move rejected by using FTP adapter.

* Write to report by using FTP adapter.

1. Check if String-length(CarriagePackageID>0.0 and $wayBillNumExistsInERP=’YES’), then move rejected by using FTP adapter.

* Write to report by using FTP adapter.

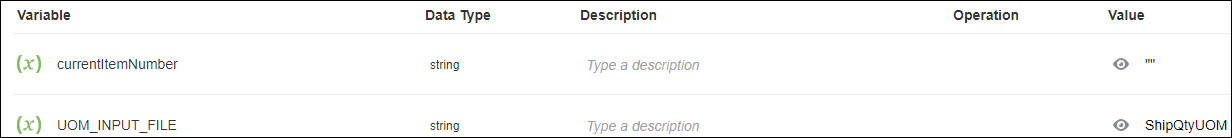
1. Otherwise, go to next step.
2. Scope of the items(While\_Each\_Item\_Scope).

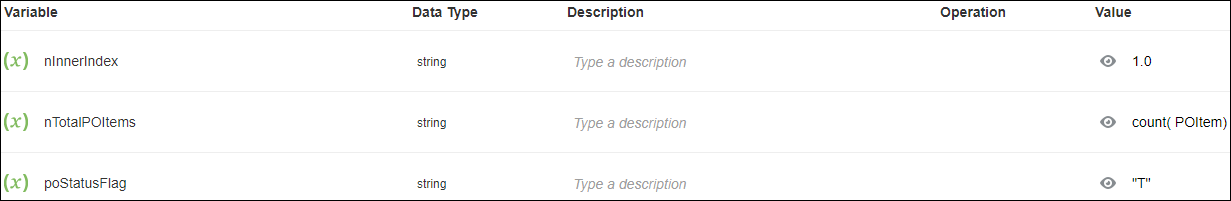
While each item from input file.



Check 2 conditions.

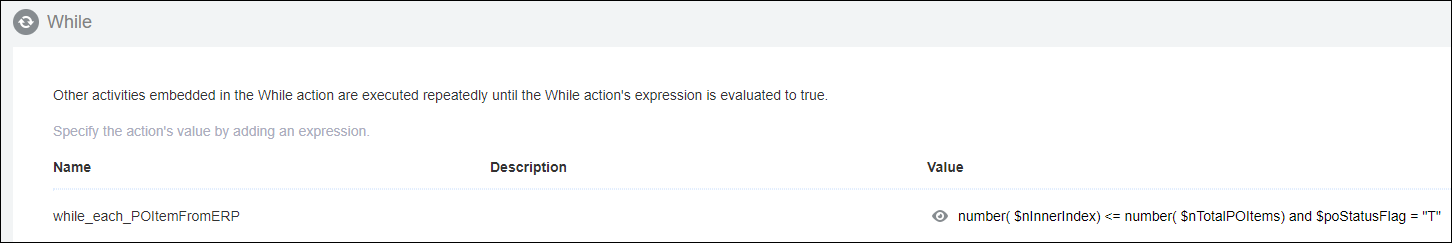
1. Check if 1.0 = 1.0(do not check anything), then Assign variables.





Scope of items(While\_each\_POItems\_ERP)

While



1. Check if LineSequencenumber=LINE\_NUM,

Check 2 conditions.

1. Compare upper-case(DOCUMENT\_STATUS=’OPEN’), then check 2 conditions.
2. Check if ShipQtyUOM= UOM\_CODE,

Check 2 conditions.

* Check if NET\_QTY>= ShipQty, then go to next step.
* Otherwise, write to report net quantity by using FTP adapter and assign variable poStatusFlag.

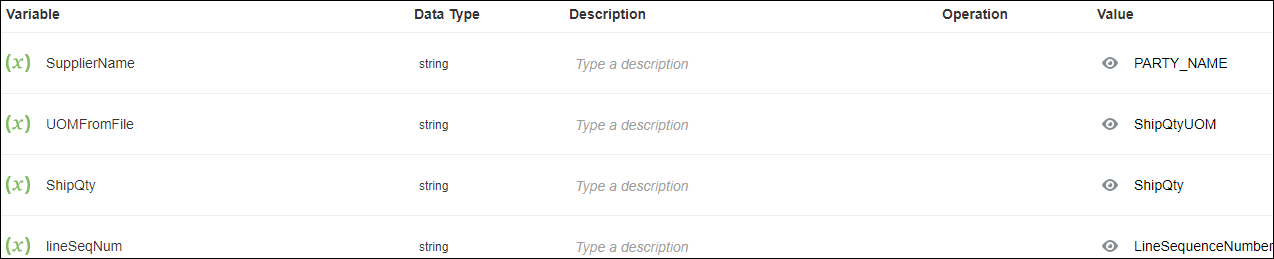
1. Otherwise, write reject report by using FTP adapter and assign variable poStatusFlag.
2. Otherwise, assign variable PO\_Line\_Number and write file action document status is not open by using FTP adapter and assign variable poStatusFlag.
3. Otherwise, write report buyer part number missing by using FTP adapter and go to next step.

Assign variable nInnerIndex. While loop ends here.

Scope ends here.

1. Otherwise, assign variables and write report as buyer part number is missing by using FTP adapter.

Go to next step.



1. Check 2 conditions.

* Check if number(totalItems)= nInnerIndex, then move valid files to /out/856/ARCHIVE/REPROCESSED\_ARCHIVE/testArchive/ReAproved’ directory by using FTP adapter.
* Otherwise, go to next step.

Assign variable nCounter(Add 1 number to it). While loop ends here. Scope ends here.

Terminate the job.

# Open and Closed Issues

1. Add open issues that you identify while writing or reviewing this document to the open issues section. As you resolve issues, move them to the closed issues section and keep the issue ID the same. Include an explanation of the resolution.  
     
   When this work product is complete, any open issues should be transferred to the project- or process-level Issue Log (Manage focus area) and managed using a project level Issue Form (Manage focus area). In addition, the open items should remain in the open issues section of this work product, but flagged in the resolution column as being transferred.

## Open Issues

| ID | Issue | Resolution | Responsibility | Target Date | Impact Date |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

## Closed Issues

| ID | Issue | Resolution | Responsibility | Target Date | Impact Date |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

1. The intent of this section is to document the design changes necessary to provide archiving required to support this component. Refer to the Logical Database Design (DS.150).