PO Acknowledgment File Validation Process (3.0.13)

Creation Date: May 31,2021

Last Updated: May 31, 2021

Document Ref: <D>

Version: DRAFT 1A

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 |
| --- | --- | --- | --- |
| 09-Jun-20 | <Radhika Thipparthi> | Draft 1a | 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 ERP BI - Report Design 2

3.1 BTST Query 2

3.2 ITEM Query 2

4 Interface Design 5

4.1 Service Design 7

4.2 External Interface Design 8

5 Misc 9

5.1 SFTP folder structure 9

5.2 Sample Input file 9

6 Open and Closed Issues 10

6.1 Open Issues 10

6.2 Closed Issues 10

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 the detailed design for Compare\_Unconverted\_Inbound\_850 (1.0.1). This specification, the design specifications for the other components that are part of this use-case package (package), along with the Analysis Specification for the package constitute the complete detailed design for this use case package.

## 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



## 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 Fusion

Oracle Integration Cloud

SFTP

# ERP BI - Report Design

1. The intent of this section is to document the report format designs for the reports provided by the component. Reference or include the report specification from the User Interface Design (DS.130) for the component.

## BTST Query

SELECT distinct hcsu.SITE\_USE\_CODE,hp.PARTY\_ID,hca.CUST\_ACCOUNT\_ID,hcsu.SITE\_USE\_ID,hps.PARTY\_SITE\_ID,hca.account\_number,hl.ADDRESS1, hl.ADDRESS2,hl.ADDRESS3,hl.ADDRESS4,hl.city,hl.state,hl.postal\_code FROM hz\_cust\_accounts hca, hz\_parties hp, hz\_party\_sites hps, hz\_cust\_acct\_sites\_all hcas, hz\_cust\_site\_uses\_ALL hcsu, hz\_locations hl WHERE 1=1 AND hp.party\_id =hca.party\_id and hp.status = 'A' and hca.status = 'A' AND hp.party\_id =hps.party\_id and hps.status = 'A' AND hcas.party\_site\_id =hps.party\_site\_id and hcas.cust\_account\_id=hca.cust\_account\_id and hcas.status = 'A' AND HPS.PARTY\_SITE\_ID=hcas.PARTY\_SITE\_ID AND hcsu.CUST\_ACCT\_SITE\_ID=hcas.CUST\_ACCT\_SITE\_ID AND hcsu.status = 'A' And hcas.cust\_account\_id=hca.cust\_account\_id and hl.status\_flag = 'A' AND hl.location\_id =hps.location\_id and hcsu.SITE\_USE\_CODE='BILL\_TO' and hca.account\_number=''and upper(hl.ADDRESS1) ='' and upper(hl.city)='' and upper(hl.postal\_code) ='' and TRUNC(sysdate) <= nvl(TRUNC(hca.ACCOUNT\_TERMINATION\_DATE) , TRUNC(sysdate) ) and sysdate <= nvl(hps.END\_DATE\_ACTIVE , sysdate) and sysdate <= nvl(hcas.end\_date, sysdate) UNION SELECT distinct hcsu.SITE\_USE\_CODE,hp.PARTY\_ID,hca.CUST\_ACCOUNT\_ID,hcsu.SITE\_USE\_ID,hps.PARTY\_SITE\_ID,hca.account\_number,hl.ADDRESS1, hl.ADDRESS2,hl.ADDRESS3,hl.ADDRESS4,hl.city,hl.state,hl.postal\_code FROM hz\_cust\_accounts hca, hz\_parties hp, hz\_party\_sites hps, hz\_cust\_acct\_sites\_all hcas, hz\_cust\_site\_uses\_ALL hcsu, hz\_locations hl WHERE 1=1 AND hp.party\_id =hca.party\_id and hp.status = 'A' and hca.status = 'A' AND hp.party\_id =hps.party\_id and hps.status = 'A' AND hcas.party\_site\_id =hps.party\_site\_id and hcas.cust\_account\_id=hca.cust\_account\_id and hcas.status = 'A' AND HPS.PARTY\_SITE\_ID=hcas.PARTY\_SITE\_ID AND hcsu.CUST\_ACCT\_SITE\_ID=hcas.CUST\_ACCT\_SITE\_ID AND hcsu.status = 'A' And hcas.cust\_account\_id=hca.cust\_account\_id and hl.status\_flag = 'A' AND hl.location\_id =hps.location\_id and hcsu.SITE\_USE\_CODE='SHIP\_TO' and hca.account\_number=''

## ITEM Query

select distinct esi.item\_number, esi.primary\_uom\_code, substr(ESI.ATTRIBUTE15,1,4)||'-'||substr(ESI.ATTRIBUTE15,5,2)||'-'||substr(ESI.ATTRIBUTE15,7,3)||'-'||substr(ESI.ATTRIBUTE15,10,4) NSN, tp.tp\_item\_number, tp.attribute1 tp\_uom from egp\_system\_items\_b esi, hz\_parties hp, inv\_org\_parameters iop, (select etp.tp\_item\_number, etp.attribute1, eir.inventory\_item\_id, etp.trading\_partner\_id from egp\_trading\_partner\_items etp, egp\_item\_relationships\_b eir where etp.tp\_item\_id = eir.tp\_item\_id) tp where esi.inventory\_item\_id = tp.inventory\_item\_id(+) and tp.trading\_partner\_id(+) = hp.party\_id and iop.organization\_id = esi.organization\_id and iop.organization\_code = 'MADE\_WA' and hp.party\_name in ('HD SUPPLY FACILITIES') and (tp.tp\_item\_number in ('')) or esi.attribute15 in ())

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.

# 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

1. The PO Acknowledgement File Validation Process files from unconverted in list files are expected to be placed in </out>folder on SFTP
2. The OIC Interface is scheduled to run at predefined intervals.
3. Given input files are in x12 format, so covert the inputs files into xml format.
4. The OIC interface when started will pick up all the .xml files from the above SFTP location
5. Verifies the foreach file and then go to Scope(Scope Each\_file).otherwise terminate the job.
6. Download the input files into the local OIC by using FTP adapter.
7. By using stage file read the files. Reset the P\_flag and again search the eachfile line file.
8. If Check LineSequenceNumber not equal to 0 then go to assign the global variable P\_FLAG is “yes” otherwise P\_FLAG1 is “no”.
9. If check P\_FLAG!=”no”

then goto GetVendorInfo by using SOAP Adapter then go to step:10.

Otherwise,

* Send mail by using notification to team action and move file to IBVI by using FTP Adapter.
* Move to reject the file MoveToRejected by using FTP Adapter and terminate the job.

10. If check NoDataFound then go to write file by using FTP Adapter.

10.1. check if Error then go to Write file by using FTP Adapter.

10.2. otherwise

* + 1. WriteStageFile file by using StageAction.
    2. ReadStageFile by using StageAction.
    3. Get VendorBiList files by using StageAction.
    4. Get VendorStageBiReadFile by using StageAction and get the Line numbers.

1. If check String\_length(CustomerOrderNumber)> 0.0

assign the CustomerOrderNumber and terminate the job.

Otherwise,

* Assign the CustomerPO is null value.
* Write the Stage\_Write\_Order\_Details by using StageFile Actiion and assign global variables in table format and call foreach.

1. If ItemStatusCode is rejected then assign the global variable update\_table and go to Step :12 .

* If ItemStatusCode is BackOrder assign the global variable BackOrderUpdateTable.
* If ItemStatusCode is ItemChange assign the global variable AssignItemChangeTable.
* If ItemStatusCode is DataChange assign the global variable AsignDataChangeTable.
* If ItemStatusCode is QuantityChange assign the global variable AsignQuantityChangeTable.
* If ItemStatusCode is PriceChange assign the global variable AsignPriceChangeTable.
* If ItemStatusCode is PartialBackOrder assign the global variable AsignPartialBackOrderTable.
* Otherwise terminate the job .

1. If check IrCount>0.0

assign the Status rejected variable and notification to team and go to step :13.

* Otherwise terminate the job.

1. if check BackOrder number( $P\_ibCount) > 0.0

then send notification to the team by using notification Action and go to step no 14.

* Otherwise terminate the job.

1. If check ItemCount number( $P\_icCount) > 0.0

then send sendICNotification to team by using Notification Action. Goto step no 15.

* Otherwise terminate the job.

1. If check DataChange number( $P\_drCount) > 0.0

then send sendDRNotification to team by using Notification Action. Goto step no 16.

* Otherwise terminate the job.

1. If check QuantityChange number( $P\_drCount) > 0.0

then send sendIQNotification to team by using Notification Action. Goto step no 17.

* Otherwise terminate the job.

1. If check PriceChange number( $P\_ipCount) > 0.0

then send sendIPNotification to team by using Notification Action. Goto step no 18.

* Otherwise terminate the job.

1. If check PartialBackOrder number( $P\_bpCount) > 0.0

then send sendBPNotification to team by using Notification Action. Goto step no 19.

* Otherwise terminate the job.

1. If check Accepted number( $P\_iaCount) > 0.0

Assign the global variable P\_Subject variable.

* Send an email AcceptedEmailnotification to team by using Notification Action and go to Step:20.
* Otherwise terminate the job.

1. Write the file by using File Transfer protocol Adapter.

21. Delete a file by using the File Transfer Protocol Adapter.and go to EachFile.

## Service Design

1. The intent of this section is to document the services that are provided by this component, for each Use Case. Refer to the Services Design (DS.120).

POACK(3.0.13)

<Service Name> is published for this component with the following arguments:

| Argument | Prompt | Value Set | Default Value |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

ERPQueryServiceProcss ( 1.0 )

ERPQueryServiceProcss is published for this component with the following arguments:

| Argument | Prompt | Value Set | Default Value |
| --- | --- | --- | --- |
| ORG\_ID | Organization id | Generic | :$PROFILE$.mfg\_organization\_id |
| USER\_ID | Application User ID | Generic | :$PROFILE$.user\_id |
| DATE | Selection Date | FND\_Date4\_required | :$$DATE$$ |

## External Interface Design

1. The intent of this section is to document the External interface messages (i.e., name, arguments etc.) that are sent or received by this component for all external systems for each Use Case.

CreateServiceOrder

<Message Name> is sent or received by this component with the following arguments:

| Argument | Prompt | Value Set | Default Value |
| --- | --- | --- | --- |
| ORG\_ID | Organization id | Generic | :$PROFILE$.mfg\_organization\_id |
| USER\_ID | Application User ID | Generic | :$PROFILE$.user\_id |
| DATE | Selection Date | FND\_Date4\_required | :$$DATE$$ |

LocationService

<Message Name> is sent or received by this component with the following arguments:

| Argument | Prompt | Value Set | Default Value |
| --- | --- | --- | --- |
| ORG\_ID | Organization id | Generic | :$PROFILE$.mfg\_organization\_id |
| USER\_ID | Application User ID | Generic | :$PROFILE$.user\_id |
| DATE | Selection Date | FND\_Date4\_required | :$$DATE$$ |

1. The intent of this section is to define the design considerations necessary to achieve the performance requirements for each interface. Include performance supplemental requirements as specified in the Supplemental Requirements (RD.055) work product for service level requirements include the Software Component Design (DS.080) with focus on the interface performance Design.

OrganizationService

<Message Name> is sent or received by this component with the following arguments:

| Argument | Prompt | Value Set | Default Value |
| --- | --- | --- | --- |
| ORG\_ID | Organization id | Generic | :$PROFILE$.mfg\_organization\_id |
| USER\_ID | Application User ID | Generic | :$PROFILE$.user\_id |
| DATE | Selection Date | FND\_Date4\_required | :$$DATE$$ |

CustomerAccountService

<Message Name> is sent or received by this component with the following arguments:

| Argument | Prompt | Value Set | Default Value |
| --- | --- | --- | --- |
| ORG\_ID | Organization id | Generic | :$PROFILE$.mfg\_organization\_id |
| USER\_ID | Application User ID | Generic | :$PROFILE$.user\_id |
| DATE | Selection Date | FND\_Date4\_required | :$$DATE$$ |

1. The intent of this section is to define the design considerations necessary to achieve the performance requirements for each interface. Include performance supplemental requirements as specified in the Supplemental Requirements (RD.055) work product for service level requirements include the Software Component Design (DS.080) with focus on the interface performance Design.
2. 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).

# Misc

## SFTP folder structure

Incoming files : /out

Archived files: //home/EDI/out/POAck/Archive

Unprocessed files : "/home/EDI/out/POAck/Rejected\_POACK"

## Sample Input file



## OIC LookUp DVM details

PO Acknowlwdgement File Validation Process

|  |  |  |
| --- | --- | --- |
| DVM EDI\_850\_LOOKUP | | |
| Input\_Type | Input\_File\_Directory | File\_Extn |
| Incoming files | /out | \*.xml |
| POACK\_Archive | /home/EDI/out/POAck/Archive |  |
| RejectedPO\_Dir | "/home/EDI/out/POAck/Rejected\_POACK" |  |

# 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 |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |