

Agile PLM Integration Pack for Oracle E-Business Suite

Design to Release 2.3 - Implementation Guide

v1.0



Part No. E14206-01

January 2009

Copyright and Trademarks

Copyright © 1995, 2009, Oracle and/or its affiliates. All rights reserved.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this software or related documentation is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS

Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

This software is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications which may create a risk of personal injury. If you use this software in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy and other measures to ensure the safe use of this software. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software in dangerous applications.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

This software and documentation may provide access to or information on content, products and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third party content, products and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third party content, products or services.

CONTENTS

Copyright and Trademarks.....	ii
Oracle AIA for Agile PLM.....	1
Oracle Process Integration for Agile PLM and Oracle E-Business Suite.....	1
Additional Resources	1
Oracle Application Integration Architecture Concepts and Technologies.....	1
Oracle Application Integration Architecture Core Components	2
Oracle Application Integration Architecture Developer's Guide	2
Oracle Application Integration Architecture Process Integration Packs.....	3
Agile PLM Integration for Oracle E-Business Suite.....	5
Architecture of Agile PLM Integration.....	6
Agile to Oracle EBS Processes.....	6
Oracle EBS to Agile Processes.....	7
Solution Design Assumptions and Constraints	8
Components of Agile PLM PIP.....	9
Agile PLM	9
Oracle Fusion Middleware.....	9
Oracle E-Business Suite.....	10
Environment	11
Process Integration for Change Order Release	13
Change Order Release Process	13
Release of Change Order in Agile.....	14
aXML generation by Agile Content Server	14
Parsing and transformation of aXML Data	14
Posting CO Data to Oracle EBS.....	17
Communicating CO Implementation status to Agile.....	18
CO Release Process Integration Solution Assumptions.....	18
Configuration Parameters.....	19
CO Release Integration Sequence.....	19
CO Release Process Flow	22
CO Release Integration Orchestration	23
AIA Services for CO Release	25

Core AIA Components for CO Release	25
Agile & Oracle EBS Components for CO Release	26
Integration Services for CO Release	27
CO Release Integration Customization Points	28
Queue Manager.....	29
Solution	30
Queue Manager Services.....	32
Process Integration for Change Order Update	35
Change Order Update Process in Oracle EBS	35
CO Update Process in Agile.....	36
Processing.....	37
CO Update Process Integration Solution Assumptions.....	38
CO Update Integration Sequence	39
CO Update Integration Flow.....	39
CO Update Integration Orchestration	40
AIA Services for CO Update.....	42
Core AIA Components for CO Update	42
Oracle EBS & Agile Components for CO Update.....	43
Integration Services for CO Update.....	43
CO Update Integration Customization Points	45
Process Integration for Item Attribute Update	47
Item Attribute Update Process	47
Processing.....	48
IA Update Process Integration Solution Assumptions.....	49
IA Update Integration Sequence	50
IA Update Integration Orchestration	51
AIA Services for ECO/PREL	52
Core AIA Components for IA Update	52
Oracle EBS & Agile Components for IA Update.....	52
Integration Services for IA Update.....	53
IA Update Integration Customization Points	54
Process Integration for Item Balance Update	57
Item Balance Update Process.....	57
IB Update Process Integration Solution Assumptions.....	58
IB Update Integration Sequence	59
IB Update Integration Orchestration	60
AIA Services for IB Update.....	61
Core AIA Components for IB Update	61

Oracle EBS & Agile Components for IB Update	61
Integration Services for IB Update.....	62
IB Update Integration Customization Points	63
Implementing the Process Integration Pack	65
Setting up the Participating Applications.....	65
Agile PLM	65
Oracle E-Business Suite.....	68
Loading Cross Reference Data	70
Creating Cross Reference Data	72
Configuring the PIP	74
Agile Configurations	75
Oracle EBS Configurations	76
Domain Value Maps	81
Out-of-the-Box DVMs	81
List of DVMs	84
Application Interfaces	88
Agile PLM Interfaces	88
Oracle EBS Interfaces	88
Handling Errors.....	90
Configuring AIA Error Notifications.....	90
Error Handling in PIP Queue Manager.....	95
Customizations.....	97
Customizing the Transformations.....	97
Transformation Rules	97
Customization in Agile	98
Customization in Oracle E-Business Suite	99
User Exits	101
Extensibility Points.....	101
Development Steps for User Exits.....	101
Appendices	103
Agile to Oracle E-Business Suite Entity Maps	103
Bill of Materials Mappings	103
Item EBO Mappings	108
Engineering Change Order EBO Mappings	114
Item Balance EBO Mappings	115

Concurrent Program Implementation Details	116
---	-----

Oracle AIA for Agile PLM

Oracle Process Integration for Agile PLM and Oracle E-Business Suite

This preface discusses:

- Additional resources
- Oracle Application Integration Architecture Concepts and Technologies
- Oracle Application Integration Architecture Core Components
- Oracle Application Integration Architecture Developer's Guide
- Oracle Application Integration Architecture Process Integration Packs

Additional Resources

The following resources are available:

Resource	Location
Installation Guide	Metalink https://metalink.oracle.com (https://metalink.oracle.com)
Documentation updates	Metalink https://metalink.oracle.com (https://metalink.oracle.com)
Release Notes	Oracle Technology Network http://www.oracle.com/technology http://www.oracle.com/technology/documentation/agile.html
Known issues, workarounds, and most current list of patches	Metalink https://metalink.oracle.com (https://metalink.oracle.com)

Oracle Application Integration Architecture Concepts and Technologies

Oracle Application Integration Architecture Concepts and Technologies is a companion volume to Oracle Application Integration Architecture Core Components and Oracle Application Integration Architecture Developer's Guide.

Oracle Application Integration Architecture Concepts and Technologies discusses:

- Enterprise business objects and enterprise business messages.
- Enterprise business services.
- Application business connector services.
- Interaction patterns.
- Extensibility.
- Versioning.
- Business processes.
- Batch processing.
- Infrastructure services.
- Security

Oracle Application Integration Architecture Concepts and Technologies contains a glossary of terms relevant to Oracle AIA.

Oracle Application Integration Architecture Core Components

Oracle Application Integration Architecture Core Components is a companion volume to Oracle Application Integration Architecture Concepts and Technologies and Oracle Application Integration Architecture Developer's Guide.

Oracle Application Integration Architecture Core Components discusses how to:

- Work with the Composite Application Validation System (CAVS).
- Work with the Business Service Repository (BSR).
- Set up and use error handling and logging.
- Work with the diagnostics framework.

Oracle Application Integration Architecture Developer's Guide

Oracle Application Integration Architecture Developer's Guide is a companion volume to Oracle Application Integration Architecture Concepts and Technologies and Oracle Application Integration Architecture Core Components.

The Oracle Application Integration Architecture Developer's Guide discusses how to:

- Create an integration scenario.
- Define business service patterns.

- Design and develop enterprise business flows.
- Design and construct application business connector services.
- Work with message transformation, enrichment, and configuration.
- Develop custom xpath functions.
- Design and construct JMS Adapter services.
- Work with enterprise message headers.
- Work with message routing.
- Work with transactions.
- Develop Oracle AIA services to work with the Composite Application Validation System (CAVS).
- Configure Oracle AIA processes to be eligible for error handling and logging.
- Extend enterprise business objects.

In addition, this book provides:

- Application Integration Architecture naming standards.
- Sample and template WSDLs for use with Oracle AIA

Oracle Application Integration Architecture Process Integration Packs

A process integration pack (PIP) is a pre-built set of integrated orchestration flows, application integration logic, and extensible enterprise business objects and services required to manage the state and execution of a defined set of activities or tasks between specific Oracle applications associated with a given process. A PIP provides everything you need to deploy a selected integrated business process area. The PIP product offering is suited to those customers seeking to rapidly implement a discreet business process.

Agile PLM Integration for Oracle E-Business Suite

This chapter includes the following:

▪ Architecture of Agile PLM Integration	6
▪ Agile to Oracle EBS Processes	6
▪ Oracle EBS to Agile Processes	7
▪ Solution Design Assumptions and Constraints.....	8
▪ Components of Agile PLM PIP	9
▪ Environment.....	11

The integration between Agile PLM and Oracle E-Business Suite, also referred as Oracle EBS, is designed to address the primary use cases around synchronization of product content information between Agile Product Collaboration and Oracle Manufacturing.

The Agile PLM Process Integration Pack (PIP) is aimed to enable and integrate Product Development process between Agile PLM and Oracle E-Business Suite. This allows for rapid implementation of Oracle's next Generation integrated enterprise PLM processes helping customer reduce cost and any risks associated with typical 3rd party and custom integrations.

This integration aims to meet the following objectives:

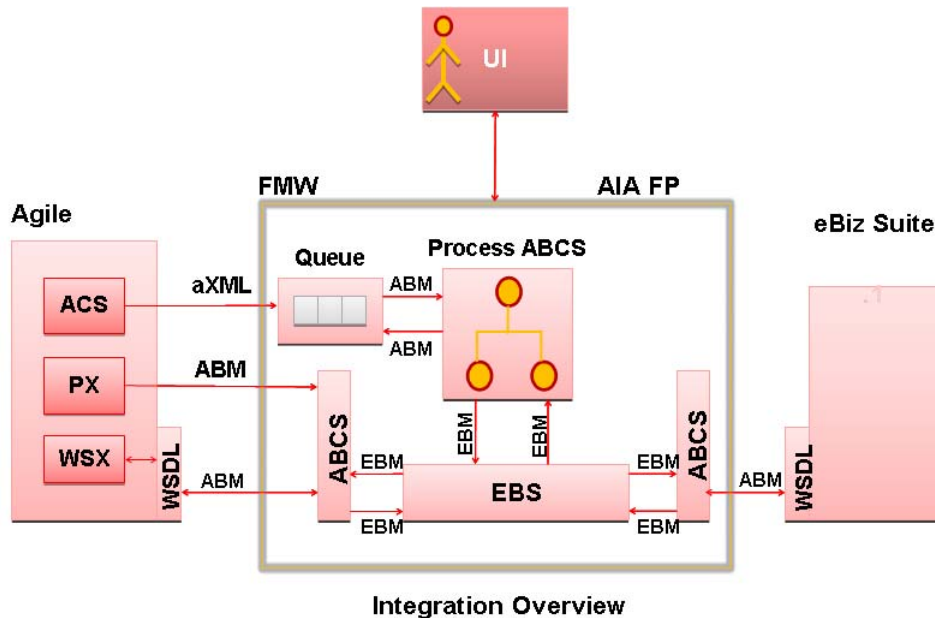
- The integration follows a Business Process flow
- Data Accuracy and Integrity are maintained at all times in both the systems
- The business process flow and data field mappings, as well as transformations, are easy to configure through tools such as Rules, Lookups and User Exits.
- Robust exception handling is done, including ease of understanding for end user, strong notification configurability and ease of troubleshooting.

Functionalities

- Manufacturing Release of new product definition and product launch
- Change Management of previously launched products
- Bi-directional synchronization of Engineering Change status and material attribute information, from Manufacturing to PLM
- Tracking and monitoring of the change processing queue

Architecture of Agile PLM Integration

Agile PLM to Oracle EBS Integration is Requester-Provider type, as depicted in the diagram below.



Agile to Oracle EBS Processes

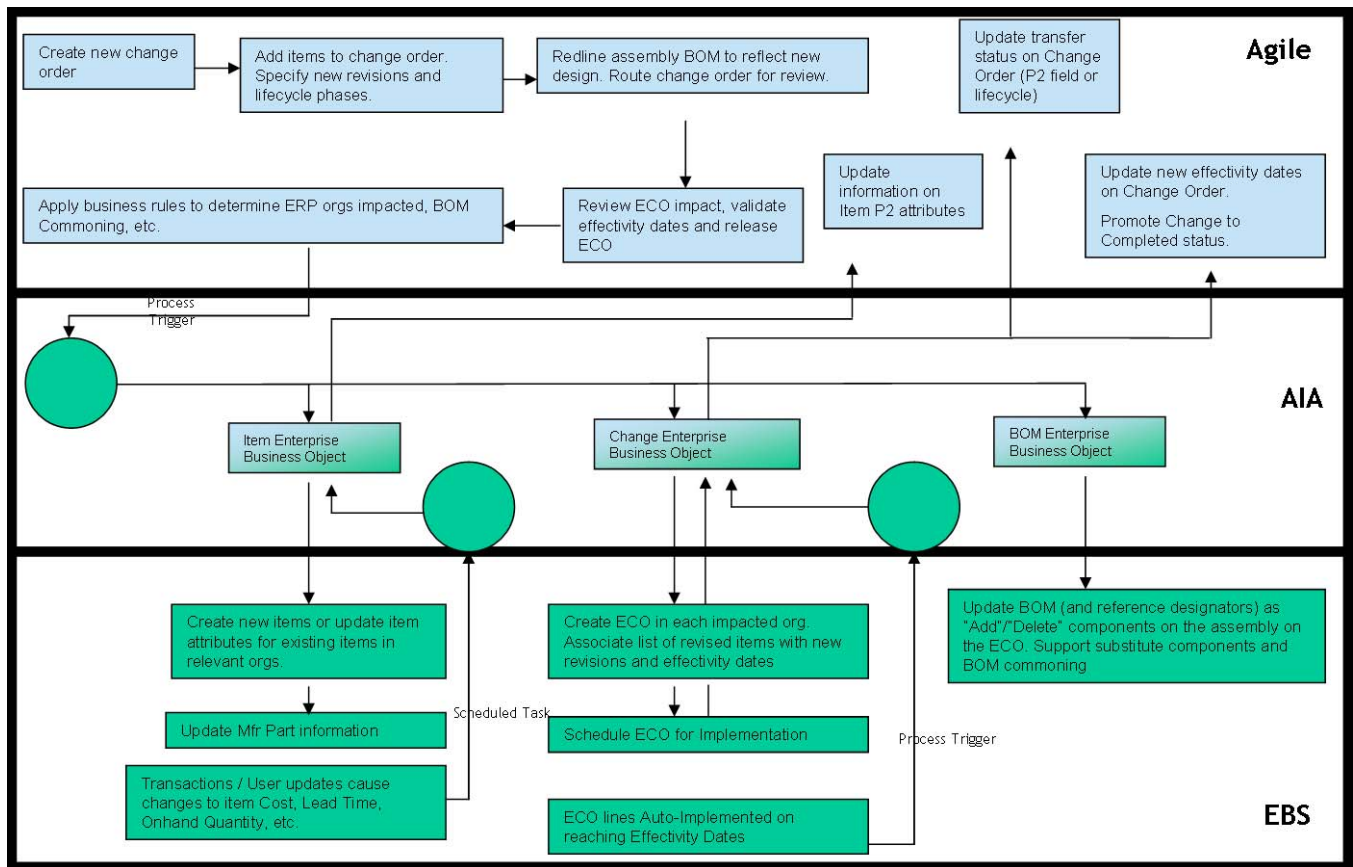
□ Change Order Release

During a product design phase, new products/parts are introduced and/or existing parts go through design changes. When the authoring of a part's attributes and design information is complete and is ready for publishing to the manufacturing system, it is released by means of Change Orders. The Change Order Release Process constitutes of New Part/Product Release (PREL) and Product Design Modification flows of Agile PLM

While Agile is the system of record for item description, design, specs and many other pieces of information above, the ERP system typically has many more attributes and placeholders for information than the PLM system. Hence, the changer order release needs to be updated in ERP system.

The release of a change order in Agile PLM system acts as a trigger for the synchronization of product design information with ERP system. Since Agile system is the system of record for product design data, the synchronization process typically involves transfer of the released revision of product design from Agile PLM to the manufacturing system.

Both these processes use the same integration sequence. [Chapter 2 - Process Integration for Change Order Release](#) "Process Integration for Change Order Release" on page 13 contains the integration details for both.



Oracle EBS to Agile Processes

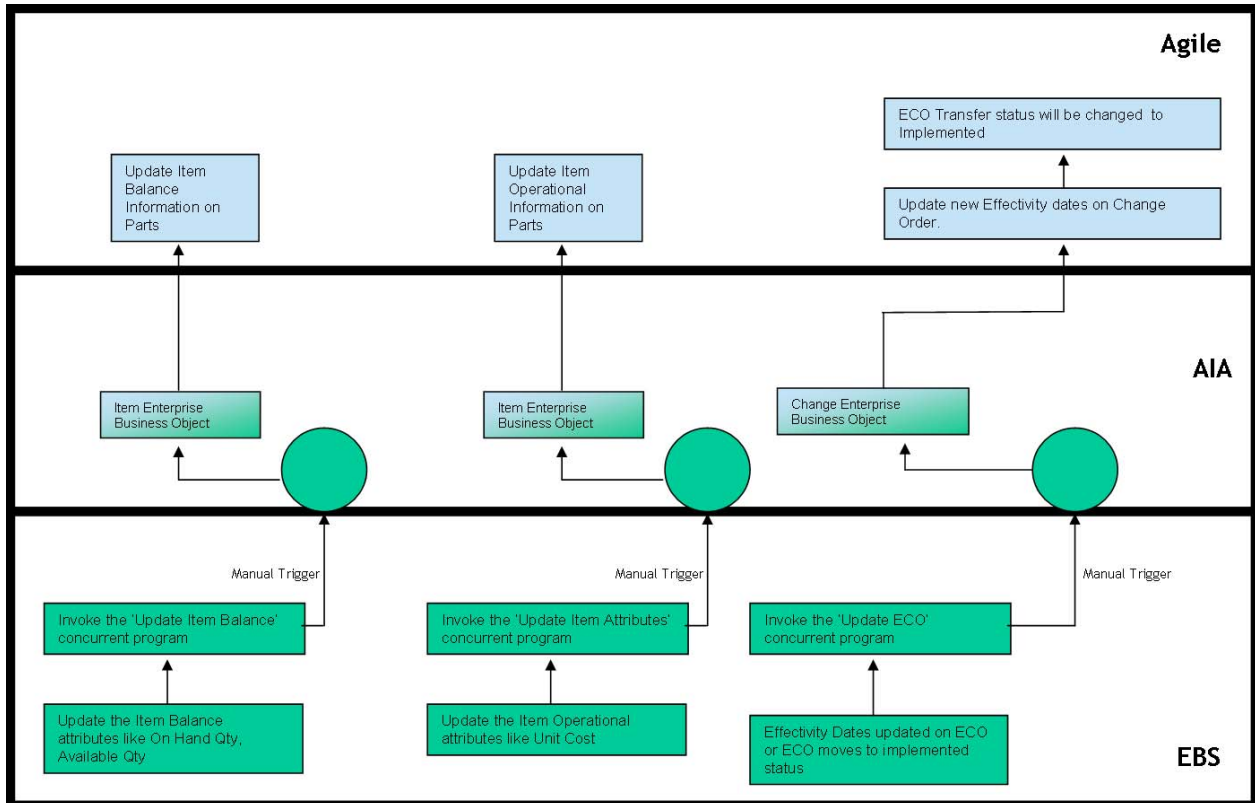
1. Oracle Change Order Implementation Information to Agile (Batch based, optionally Event based)
2. Oracle Item Operational Attributes and Unit Cost Information to Agile (Batch)
3. Oracle Item Balance Information to Agile (Batch)

The Change Order Update from Oracle EBS to Agile is a communication of change in status of the Change Order in the ERP system (also part of the Manufacturing Update business flow). It constitutes a key requirement for keeping users in Agile apprised of the lifecycle of a Change Order. This involves communicating the Change Order status in the ERP system to a configurable field on the Change Order in the Agile system, and/or changing the status of the Change Order in the PLM system.

As a necessary part of the Manufacturing Update process, the ability to update a Change Order Line in Agile with updates on Effectivity Date and Implementation Status from the ERP system is a key component of the bidirectional synchronization capability of the integration.

The Item Balance information in ERP system is stored in three heads - Reserved Quantity,

Available Quantity, On-hand Quantity. Further, an Item in ERP can exist in more than one org. Any change in any of the three types of quantities may take place in just one or in a few or in all the orgs. These changes are updated in Agile. Similarly, whenever any change is carried out in the attributes, for example cost, of an item in ERP system, it calls for a corresponding update in Agile system.



Solution Design Assumptions and Constraints

Design Assumptions

- Agile Content Server will be used for Events to trigger the payload from Agile to Integration.
- This design assumes that the following statements are true:
 - There will be pre-defined blank templates made available for Custom fields
 - Transformation logic for classification elements will be pre-coded in the OOTB XSL, but you may need to modify it to suit your PLM implementation
- This design leverages AIA error handling framework.

Design Constraints

- In few cases configuration driven XSLT may not reflect the changes immediately and may require a restart as the main XSL sheet would get cached after a successful compilation
- ACS limits the Events to be triggered from workflow only for the Change Status which can be used for this Integration.

- Error handling capabilities of this integration are constrained by the capabilities of AIA framework.

Components of Agile PLM PIP

Agile PLM

Agile Content Service (ACS)

Agile Content Service is an event-driven XML-based publishing service that makes the product record available to a wide variety of business applications and users, both internally and across the global manufacturing network. In addition to allowing employees and supply chain partners to publish the product record on demand, Agile Content Service can be configured to automatically publish the Item Master, BOM, and AML changes during any phase of the product lifecycle to multiple destinations, ensuring that everyone is working with up-to-the-minute information.

The output generated by an ACS module is an aXML file or a PDX package.

Web Service Extensions

Web service extensions (WSX) is a Web service engine enabling communication between Agile PLM and disparate systems both internal and external including Enterprise Resource Planning (ERP) systems. They can be used to provide content to exchanges, reports, and custom applications and import Product content data from ERP and other supply chain applications. WSX can simplify the process for aggregating raw product content and making critical product content available in real time to other core systems.

For more details on Agile components, refer *Agile Product Lifecycle Management Administrator Guide* and *SDK Developer Guide*.

Oracle Fusion Middleware

Oracle Fusion Middleware (OFM) is a portfolio of software products, produced by Oracle, which spans multiple services, including J2EE and developer tools, integration services, business intelligence, collaboration, and content management.

Oracle Fusion Middleware is designed to support development, deployment, and management of Service-Oriented Architecture. It includes what Oracle calls "Hot-Pluggable" architecture, which allows users to leverage existing investments in applications and systems from other software vendors such as IBM, Microsoft, and SAP.

Business Process Execution Language (BPEL)

BPEL is that technology that integrates and assembles the Web Services. BPEL is a XML-based workflow definition language that allows businesses to describe inter or intra enterprise business processes that are connected via Web services. BPEL opens a completely new way or at least enhanced way, for software development for mainstream business applications to allow a programmer to describe a business process that will take place across the Internet.

BPEL provides an XML-based grammar for describing the logic to control and coordinate Web services participating in a process flow.

Enterprise Service Bus (ESB)

An Enterprise Service Bus (ESB) is a software architecture for middleware that provides fundamental services for more complex architectures and can be thought of as a mechanism that manages access to applications and services.

Oracle Service Registry (OSR)

Oracle Services Repository is a Web Services Registry and Repository for building your Service-Oriented Architectures (SOA). It provides a UDDI v3-compliant platform for publishing, categorizing and discovering Web services and related resources across the enterprise. OSR enables service providers to expose and advertise service offerings, Allows service consumers to find, access and/or invoke services that meet defined criteria and provides critical features for SOA governance.

Application Integration Architecture (AIA)

Oracle Application Integration Architecture provides an open standards based framework for creating cross-application business processes that support the way you run your business today, while paving the way for your long term, strategic, business transformation plans. Its application independent framework enables you to utilize the applications of your choice to create composite business processes unique to your business, on a flexible service-oriented architecture.

Oracle E-Business Suite

Oracle Integration Repository

An integral part of Oracle E-Business Suite, Oracle Integration Repository is a compilation of information about numerous interface endpoints exposed by Oracle applications. It provides a complete catalog of Oracle E-Business Suite's business interfaces, and a comprehensive view of the interface mechanisms available. It can be used to easily discover and deploy the appropriate business interface from the catalog for integration with any system, application, or business partner.

Business Event System

Business Event System is an application service that uses the Oracle Advanced Queuing (AQ) infrastructure to communicate business events between systems. It consists of the Event Manager, which lets you register subscriptions to significant events, and event activities, which let you model business events within workflow processes. When a local event occurs, the subscribing code is executed in the same transaction as the code that raised the event. Subscription processing can include executing custom code on the event information, sending event information to a workflow process, and sending event information to other queues or systems.

Concurrent Programs/Manager

Concurrent Processing is an Oracle Applications feature that allows non-interactive and potentially long-running functions which may involve large number of data-intensive computations, to be executed efficiently alongside interactive operations. It uses operating system facilities to facilitate background scheduling of data- or resource-intensive jobs, via a set of programs and forms. To ensure that resource-intensive concurrent processing operations do not interfere with interactive operations, they are run on a specialized server, the Concurrent Processing server.

Environment

<i>Agile PLM</i>	9.2.2.4
<i>Oracle Manufacturing</i>	11.5.10 CU2/12.1 Oracle E-Business Suite environment must be installed and configured accordingly. The aru# 7563732 patch, which is specific to PIP, should be applied over the Oracle E-Business Suite 11.5.10 CU2 installation.
<i>Oracle SOA Suite</i>	10.1.3.4
<i>Oracle AIA Foundation Pack</i>	2.3
<i>Oracle Database</i>	10g Release 2 (10.2.0.1) Enterprise Edition DB upgrade patch 10.2.0.3
<i>Oracle Service Registry</i>	10.1.3.1
Java 2 Platform Standard Edition (J2SE) Development Kit (JDK) 5.0, Update 16	

See Capacity Planning Guide for technical details of infrastructure requirements.

Process Integration for Change Order Release

This chapter includes the following:

▪ Change Order Release Process	13
▪ CO Release Process Integration Solution Assumptions	18
▪ CO Release Integration Sequence	19
▪ AIA Services for CO Release	25
▪ CO Release Integration Customization Points	28
▪ Queue Manager	29

Integration of Change Order Release, namely the Engineering Change Order (ECO) and New Part/Product Release (PREL) from Agile to Oracle EBS is about introducing a new product, components & structure, and manufacturer information into Oracle EBS upon release of design from Agile, or updating existing design metadata, structure or manufacturer information in Oracle EBS.

Note The Integration flow is similar for both ECO and PREL use cases.

Change Order Release Process

The process of Change Order Release, which comprises of New Part Introduction and Manufacturing Update, is the flow of Item, Manufacturer Part and BOM Information from Agile to Oracle EBS. This information is pushed from Agile, triggered by an Event tied to the change in Status of a Change Order (CO) object. The information is then parsed in an Integration Object format and sent to Oracle EBS for implementation. A confirmation of its implementation status is sent back to Agile.

This process of Integration entails the following flow:

1. Release of Change Order in Agile
2. aXML generation by Agile Content Server
3. Parsing and transformation of aXML Data
4. Posting Changer Order Data to Oracle EBS
5. Communicating Order Processing Status to Agile

Release of Change Order in Agile

When a Change Analyst approves the Change Order in Agile, it is marked as Released. This makes all the changes specified in CO take effect in Agile.

Before the release of a CO, it is subjected to pre-validation at its approval stage. This involves validation of certain business rules to ascertain that the flow of CO from Agile to Oracle EBS would meet all the conditions set in the destination system.

aXML generation by Agile Content Server

The Agile Content Server generates an aXML (Agile XML) file from CO data. This file contains information of Items, BOM, Manufacturers and the CO itself.

For the purpose of Agile to Oracle EBS Integration Process, the ACS is configured to ensure the following:

1. The aXML file is configured to carry the following elements from a CO:
 - Change Order Data: Cover Page, Page Two, Page Three, Affected Items tab attributes
 - Revised Item Data: Title Block, Page Two, Page Three
 - BOM Data: BOM tab of Items (including reference designators) with delta BOMs only for the revision on CO.
 - AML Data: AML tab of Items with delta AMLs only.

Note	It is assumed that the Manufacturer already exists in Oracle EBS.
------	---

2. Upon release of a CO, the aXML file goes to a JMS Queue or a File Folder.

Parsing and transformation of aXML Data

The data contained in aXML file generated by Agile is not in the format that is understood by Enterprise Business Objects. Hence, this data has to be parsed and transformed.

The parsing and transformation of aXML data entails the following:

1. Segregation of Business Objects
2. Sequencing and Queuing of Change Orders
3. Translating Agile 'Site' specific objects into Oracle EBS's 'Organization' specific objects
4. Translating Agile CO Types into Oracle EBS's CO Type
5. Mapping Agile CO Attributes to corresponding attributes in Oracle EBS
6. Ascertaining existence of an Item in Oracle EBS
7. Defining User Exit Points for custom transformations

Segregation of Business Objects

The aXML file contains collective information about the business objects – CO, Item Attributes, Revised Item Lines, BOM Redlines, Reference designators, etc. This information is broken down into individual components and mapped, one to one, with corresponding EBOs, namely Item, Change and Structure.

In order to maintain referential integrity, the Change Number is associated with each business individual object.

Sequencing and Queuing of Change Orders

The CO release process begins with queuing of CO Number in Process Queue Controller, which sequences the COs for transfer of parsed data to Oracle EBS. Once the data is processed by Oracle EBS, and its implementation status received, the CO is removed from the Process Queue.

For complete details on sequencing and queuing of change orders, refer [Queue Manager](#) on page 29 section in this chapter.

Translating Site Objects

The data coming from Agile is split into individual Oracle EBS Organization specific business objects. This is because the data in Agile can be:

- Centralized – all design locations share the same product design information; or
- De-centralized – the site specific Item Attributes, change control, etc. are implemented to multiple sites.

The Data in Oracle EBS, however, is segregated by Organizations.

Refer [Appendices](#) on page 103 for information on Attribute Mapping.

Translating CO Types

In Agile, a Change Order is categorized into one of the following change type:

- Engineering Change Order (ECO)
- Manufacturing Change Order (MCO)
- Site Change Order (SCO)

These categories are called as Classes in Agile. A Class may have one or more sub-classes.

Oracle EBS does not have separate categories for each of these change types, and hence, cannot be differentiated. Hence, while parsing and processing of these change types in Agile, they are translated in Oracle EBS with the following characteristics:

Feature	ECO	MCO	SCO
Customer adoption	All installations	All installations	---
New Revisions for	Mandatory	Not supported	Not supported

Feature	ECO	MCO	SCO
revised items			
Tables redlined	BOMs and AML; Global as well as site-specific	AML only; Global as well as site-specific	Site-specific BOMs and AML only
Effectivity Date	At line level; Global when multi-site is not enabled; Separate for each site when multi-site is enabled	Not specified on Change Order	At line level; site-specific only
Other line-level attributes	Global when multi-site is not enabled; Separate for each site when multi-site is enabled	Global when multi-site is not enabled; Separate for each site when multi-site is enabled	Site-specific only
New Item Release	Supported	Supported	Only item updates are supported

Mapping CO Attributes

The Source system attributes are mapped to Destination system attributes. This mapping also defines the direction of data flow.

This is accomplished as follows:

1. Two different sets of mappings are defined. One for information flowing from Agile to EBOs. Another, for information flowing from EBOs to Agile. These are accessed in the UI as different 'mapping profiles'.
2. A parameter is specified with each mapping done from the UI. This parameter determines whether the mapping applies to one or both of the directions of data flow.

Note See [Appendices](#) on page 103 for complete list of Agile to Oracle EBS Mappings.

Ascertaining Item existence in Oracle EBS

An Item is created in the Master Org of Oracle EBS in two ways:

1. Agile releases New Part Introduction information to Oracle EBS through a CO, as a First Time release.
2. The Item information is loaded in Oracle EBS by an external system, other than Oracle EBS and Agile.

If the Item already exists in the Master Org, and Agile releases CO to create the same Item, the

system would error it out. Since, Agile does not explicitly pass information about first time or subsequent release of Item, a lookup table is employed to ascertain the existence of the Item in Oracle EBS.

This lookup table maintains unique identifiers for the Items received from Agile and corresponding Items created in Oracle EBS. It also maintains the unique identifiers for the Items created in Oracle EBS by an external application. These unique identifiers help in ascertaining the existence of an Item in Oracle EBS, thus eliminating any duplication errors.

Defining User Exit Points

User Exits have been provided in the integration to allow custom transformations or filtration routines that a customer may want to add in the process without affecting the main integration flow.

The User Exit points for each process are listed in their respective chapters. Also, see [Loading Cross Reference Data](#).

Posting CO Data to Oracle EBS

The processing of Change Order data into the Oracle EBS is the backbone of this integration. As part of this step, the following activities need to be carried out:

1. **Item Master synchronization:** For all items pushed to the Oracle EBS system, it is verified whether or not the items already exist, and have the same revision number as the old item from the Agile system. If the item didn't exist in Oracle EBS and is being released for the first time from PLM, it is created in Oracle EBS. If the item already exists in Oracle EBS and the two systems were in synch with regards to item revision, the existing item is updated with new attribute data from Agile.

If the two systems are not in synch with regards to the prior revision of the revised item (that is, the old revision of the item as per data from Agile does not match the current revision of the item in Oracle), an error is raised.

Alternately, the integration may also be configured to ignore prior revision matching, and to just create the item in the ERP system if the item doesn't already exist, or update the item if the item already exists and the incoming transaction type from Agile is updated, as specified in a subsection above.

2. **Item Organization assignment:** Items are assigned to organizations based on criteria specified in the section on supporting distributed manufacturing above.
3. **AML update:** New Approved Manufacturer List information from Agile replaces the existing item AML. AML information is supported at the master item level only by Oracle EBS.
4. **BOM update (including reference designator):** The XML data from Agile contains only the changes made to Bills of Material and not the complete Bills of Material. As a result, BOM data needs to be in sync between Agile and Oracle EBS for the older revision in order for the new revision of BOM data to be posted successfully.
5. **Create Change Order:** The actual Change Order is created as an object in the ERP system. At the end of the post, the Change Order is set to a status of *Scheduled* for implementation by the Oracle's *Auto-Implement Manager*. The Auto-Implement Manager implements the Change Order's lines as and when their scheduled Effectivity Date arrives. The complete Change Order is moved to the Implemented status when all the lines are successfully implemented.

A large number of business rule validations are carried out by the Oracle EBS APIs as part of this step, especially when creating Item and BOM data.

Some of the actions involved in the steps above (such as item creation, BOM update, etc) may involve different ways of handling exceptions from implementation to customer.

For example, if a component is being added to a BOM in a given organization in Oracle EBS, but it doesn't exist as an item in the Oracle EBS system, two kinds of actions can be carried out:

- BOM processing fails if an item doesn't exist in Oracle EBS. However, if the item exists in Master Org and the context org is the Child Org, then it is configurable. OR
- Create the component using information fed by Agile (or if it already exists in the master org in the Oracle EBS, extend it to the child org), and resume processing of the BOM. However, if a workflow is associated for the Change Order Type, then the incoming ECO is created in the initial stage itself.

Different customers may have different preferences on which action to carry out. The choice of such actions will therefore be exposed to the integration administrator using configurable parameters.

Communicating CO Implementation status to Agile

When the process of posting Change Order data into the Oracle EBS system completes (successfully or with errors), the following steps are taken:

1. If the parameter ERP Transfer and Implementation Status Field is set to null and the parameter Attach ERP Transfer Log is set to No, it indicates that no updates back into Agile are required. The Change Order Release process can then proceed to the final step of sending out success email notifications
2. If the parameter ERP Transfer and Implementation Status Field is Not Null, a PLM Change Order update process is initiated with the Change Order number, and a value of Errored or Transferred is passed for the field specified in the Oracle EBS Transfer and Implementation Status Field parameter.
3. If the parameter Attach ERP Transfer Log is set to Yes (checkbox checked) a text file is generated containing a log of the processing of the ECO. The log typically contains only summary (level 0) messages in case of success, but detailed messages in case of failure (especially for the step on which the error is encountered).

The file is named using the following convention:

<UniqueIntegrationName>_CO_Release_Processing_Log_<YYYYMMDD HH24:MM:SS>.txt

This text file is then sent to Agile PLM for attachment to the change order. It must be noted that the CO may be processed multiple times through the interface, and there may therefore be a number of such log files attached to the CO.

CO Release Process Integration Solution Assumptions

- If the part already exists in the ERP/PIM system because it was created via the ECO (Engineering Change Order) process, then update the part with information from Agile.
- If the part doesn't already exist in ERP system, then create the part using tools such as Oracle Templates
- If the part already exists in ERP system, update the part.

- Use the ECO to associate new revision, effectivity date, Bills of Material and Reference Designator information with the new item
- ECO should be used to transfer the new part/update part information from Agile to ERP system.
- There should be a way to monitor each ECO process.
- There should be a way to prioritize ECO processes.
- Another process can start its execution only after the execution of first process is completed.
- A mechanism should be provided for error handling/reporting to the end user.

Configuration Parameters

1. If the customer uses Agile multi-sites, the Administrator may assign organization names to sites to indicate the destination ERP organization set for an item.
2. A parameter called ERP Operations for Redline Update shall determine whether a modify operation on a BOM row in Agile is treated as a Delete and Add or a Modify operation on the ERP system.
3. A parameter called ERP Transfer and Implementation Status Field (type Text) shall be provided. The value of this parameter, when provided, would indicate the Change Order field to which the integration would post the status of the ERP post transaction.

In Agile, it is recommended that this field be:

- a. A Page Two field on the Change Orders, Manufacturer Orders and Site Change Orders classes. It must be kept in mind that the field should have the same display name on all three classes.
- b. Have a default value of Not Processed.
- c. Be editable only by the user id used by the integration to log in to Agile, and be editable for all statuses (just in case the Change is unreleased after its initial release)

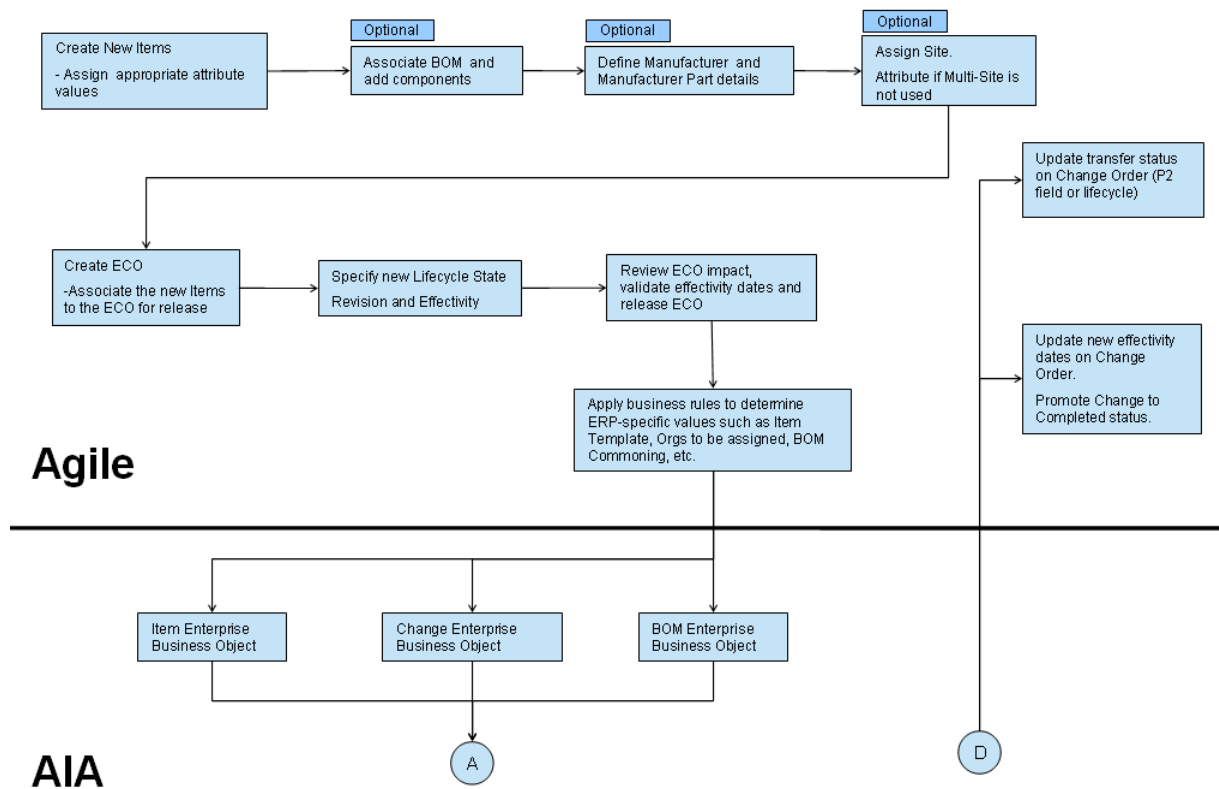
If a value is not specified for this parameter, that implies that a status update back to a CO flexfield in Agile is not required.

The field identified by this parameter will also be used by the integration to update the Change Implementation status back into Agile.

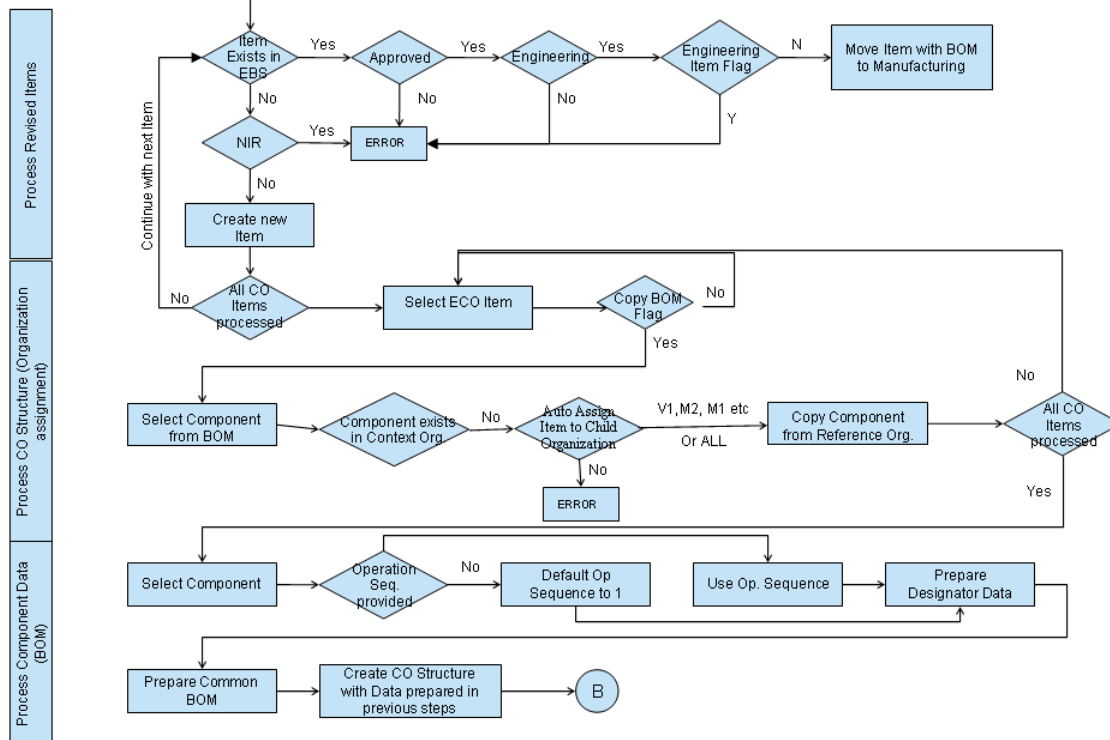
4. A parameter called Attach ERP Transfer Log (type Checkbox) shall be provided. If checked (set to Yes), this parameter would indicate that it is desired to attach a copy of the processing log of this ECO through the interface to the ECO in Agile.
5. A parameter called Integration Name (type Text) identifies the name that the customer would like to give to this integration. This integration name would be used in naming log files, among other things. The default value for this parameter should be Agile-EBS.

CO Release Integration Sequence

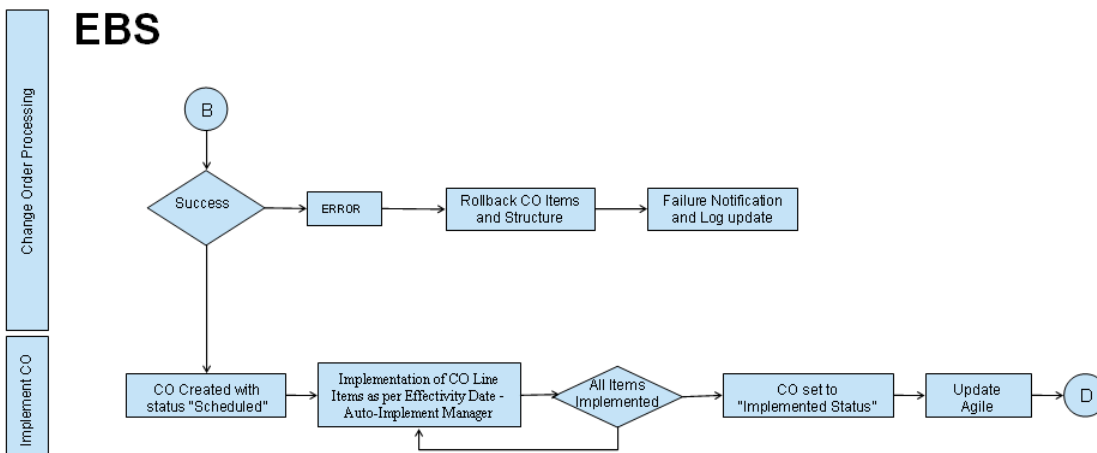
The Integration flow is same for both ECO and PREL use cases.



EBS

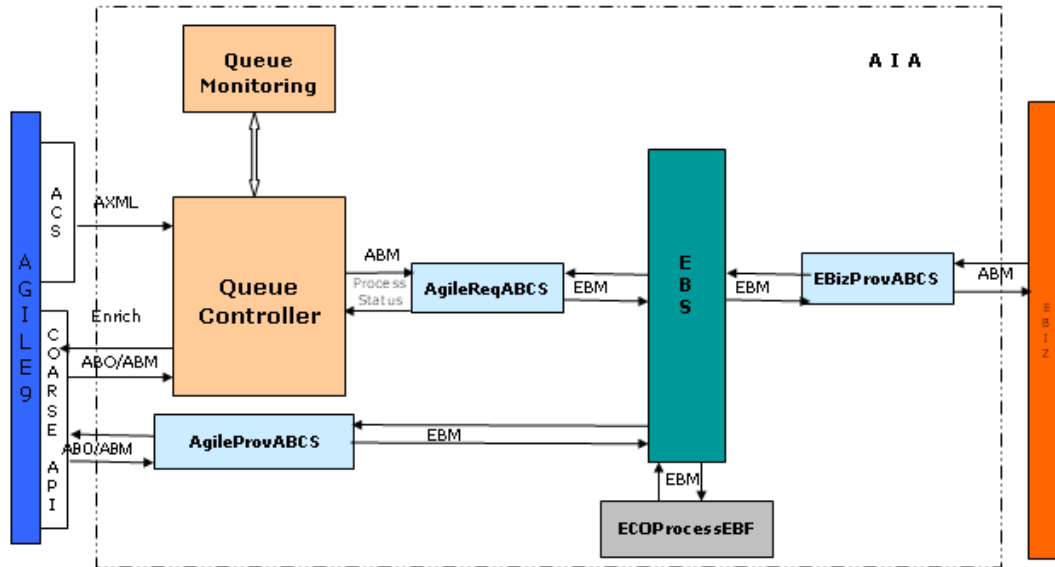


EBS



CO Release Process Flow

An Engineering Change Order is created with items in AI tab with new revisions and lifecycle phases specified.

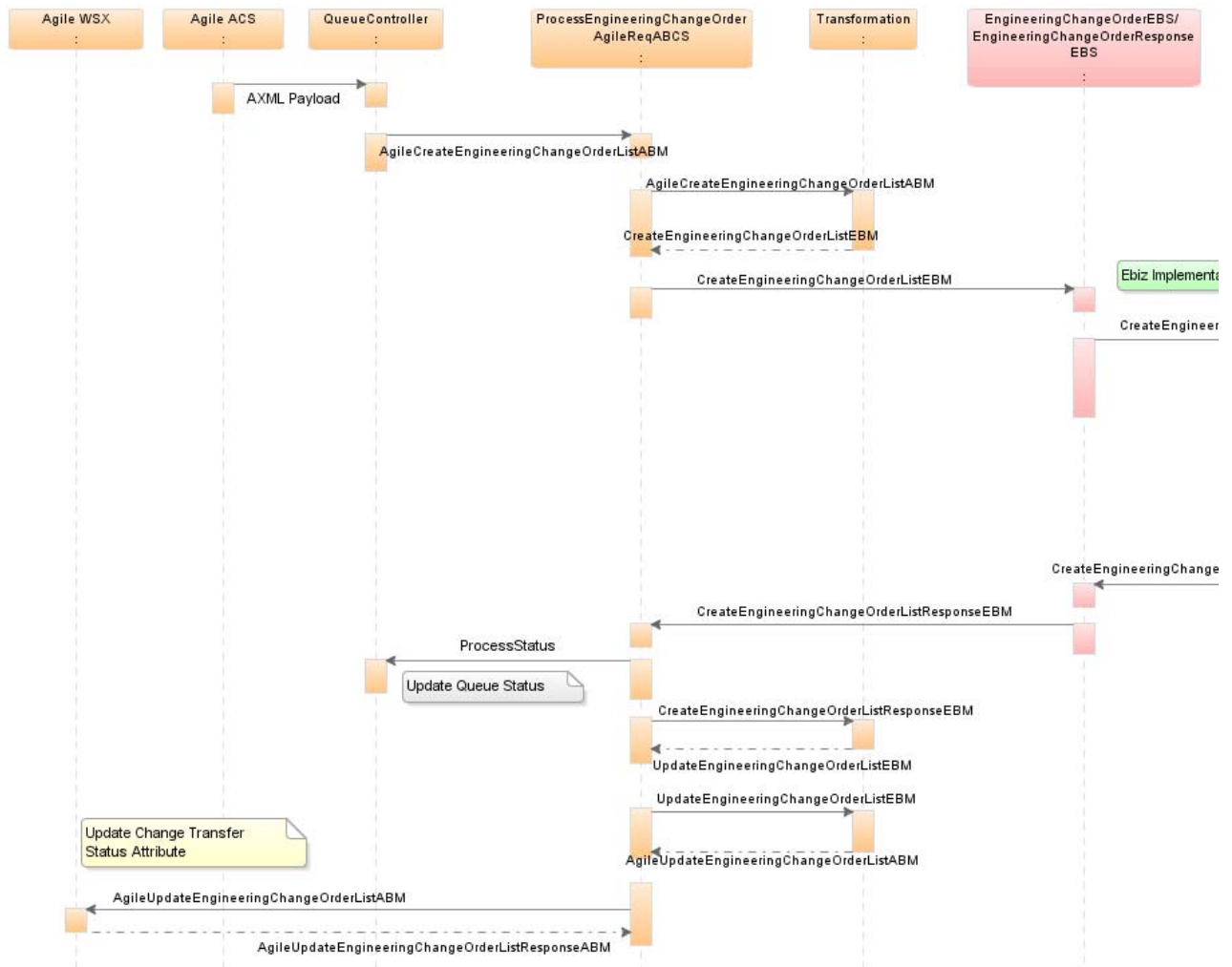


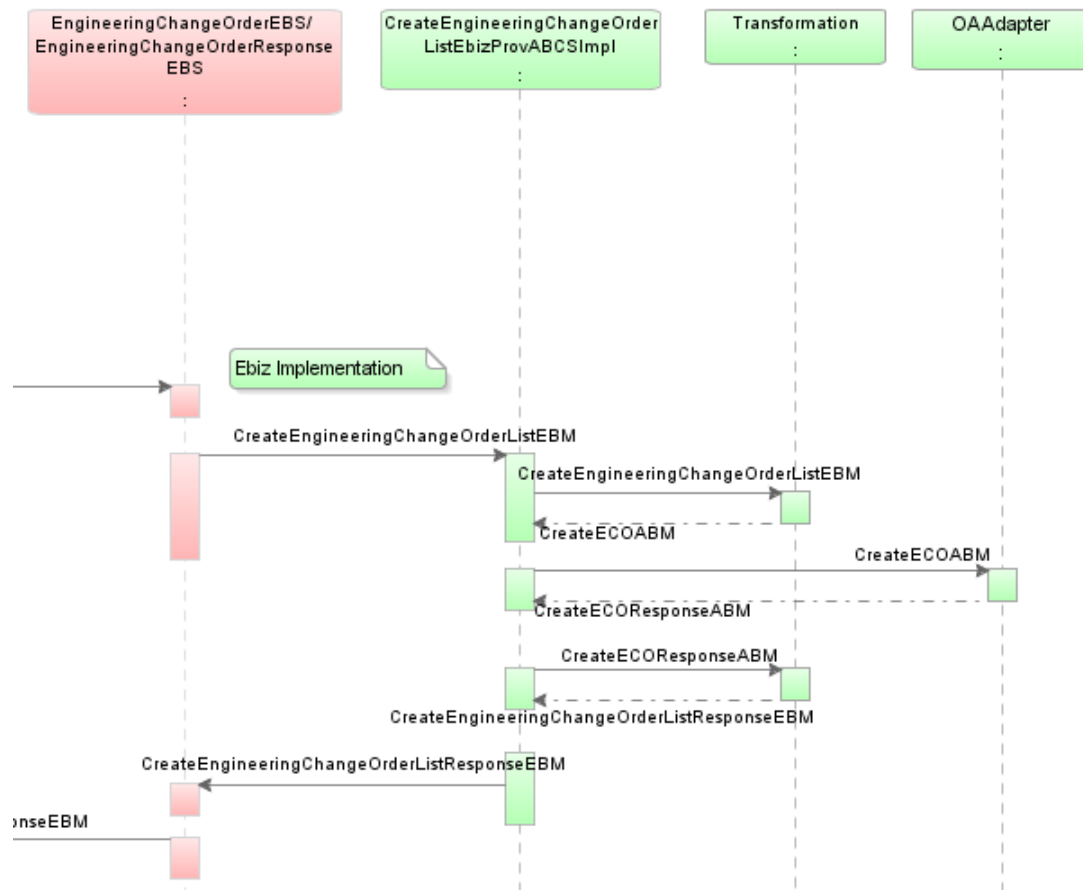
1. The ECO is routed for Approval (workflow step).
2. An ACS Workflow Event is generated on Approval of the ECO in Agile to trigger the ECO process flow.
3. The Queue framework captures the ACS payload (aXML) generated for the event and adds it to the integration Queue.
4. The Queue framework identifies the highest priority Queue Message, processes it to create an ECO ABM and triggers the Requestor ABCS.
5. The Requestor ABCS transforms the ECO ABM to ECO EBM and triggers an operation on the EBS which routes the EBM to the ECO business flow with ECO EBM as the input.
6. The ECO Business Flow -
 - Creates new items in PIM/ERP
 - Creates an Engineering Change Order in Oracle EBS.
 - Associates list of revised items with new revisions and effectivity dates, and Schedules the ECO for implementation.
 - Creates new BOM
 - Updates the Transfer Status in Agile.
7. The status of Queue Message is updated in the Integration Queue, for monitoring.
8. When the ECO lines are Implemented on reaching effectivity dates, a process trigger initiates the following flow:
 - Update the Effectivity date and Implementation status for each Affected Item on ECO in Agile.

- Change Status of ECO to Implemented if all the AI items are implemented.

CO Release Integration Orchestration

Agile Side



Oracle E-Business Side

#	Activity	Remarks
1	Agile ACS transmits Agile Engineering Change Order Data in payload in the form of predefined XML format known as aXML. This file will get queued up for the further processing.	Agile ACS acts as a trigger for ECO Use case.
2	The QueueController Framework reads the highest priority Queue Message and transforms the payload (aXML) to AgileCreateEngineeringChangeOrderListABM.	QueueController processes the payload
3	QueueController invokes the ProcessEngineeringChangeOrderAgileReqABCS with AgileCreateEngineeringChangeOrderListABM as input.	
4	AgileCreateEngineeringChangeOrderListABM is transformed into CreateEngineeringChangeOrderEBM.	ProcessEngineeringChangeOrderAgileReqABCS will make call backs to Agile Web services, if needed.
5	ProcessEngineeringChangeOrderAgileReqABCS invokes the CreateEngineeringChangeOrder operation on EngineeringChangeOrderEBS with	

#	Activity	Remarks
	CreateEngineeringChangeOrderEBM as input	
6	EngineeringChangeOrderEBS routes CreateEngineeringChangeOrderEBM to CreateEngineeringChangeOrderListEbizProvABCImpl	
7	CreateEngineeringChangeOrderListEbizProvABCImpl transforms CreateEngineeringChangeOrderEBM into the input of Ebiz Service and calls that service.	Creates items in PIM/ERP, creates an ECO, associate revised items to it and creates BOM.
8	CreateEngineeringChangeOrderListEbizProvABCImpl invokes CreateEngineeringChangeOrderResponse operation on EngineeringChangeOrderEBS with CreateEngineeringChangeOrderResponseEBM as input.	
9	The EngineeringChangeOrderEBS routes CreateEngineeringChangeOrderResponseEBM to ProcessEngineeringChangeOrderAgileReqABCS	Response message routing
10	ProcessEngineeringChangeOrderAgileReqABCS sends the status back to the Queue Controller to update the queue.	This status is updated against the Queue message in the database by the QueueController
11	ProcessEngineeringChangeOrderAgileReqABCS transforms CreateEngineeringChangeOrderResponseEBM into AgileUpdateEngineeringChangeOrderListABM AgileUpdateEngineeringChangeOrderListABM is sent as an input to the Agile Web Service. AgileUpdateEngineeringChangeOrderListResponseABM is sent back to ProcessEngineeringChangeOrderAgileReqABCS	The web services update the transfer status on the Change Order in Agile which will be predefined P2 or P3 attributes on ECO object in Agile.

AIA Services for CO Release

Core AIA Components for CO Release

The Process Integration for ECO/PREL uses the following industry components:

EBOs	▫ EngineeringChangeOrderEBO
EBMs	▫ CreateEngineeringChangeOrderListEBM ▫ CreateEngineeringChangeOrderListResponseEBM ▫ UpdateEngineeringChangeOrderListEBM ▫ UpdateEngineeringChangeOrderListResponseEBM
EBSs	▫ EngineeringChangeOrderListEBS (CreateEngineeringChangeOrderListResponseEBM)

Core Components Locations

EBO & EBM XSD files	http://[HOST:PORT]/AIAComponents/EnterpriseObjectLibrary/Core/EBO/
WSDL files	http://[HOST:PORT]/AIAComponents/EnterpriseBusinessServiceLibrary/Core/EBO/

- For detailed documentation of individual EBOs, click the EBO Name link on the *Integration Scenario Summary* page in the *Oracle AIA Console*. You can also use the *Integration Scenario Summary* page to search for and view integration scenarios that utilize a particular EBO or EBS.

For more information, see *Oracle Application Integration Architecture - Foundation Pack: Core Infrastructure Components Guide*, “Using the BSR,” Using the BSR UI to View Integration Scenarios.

- EBOs can be extended, for instance, to add new data elements. These extensions are protected, and will remain intact after a patch or an upgrade.

For more information, see *Oracle Application Integration Architecture – Foundation Pack: Integration Developer’s Guide*, “Extensibility for AIA Artifacts.”

Agile & Oracle EBS Components for CO Release

Services	Agile (Requester)	Oracle EBS (Provider)
ABMs	<ul style="list-style-type: none"> AgileCreateEngineeringChangeOrderListABM CreateEngineeringChangeOrderListResponseABM AgileUpdateEngineeringChangeOrderListABM AgileUpdateEngineeringChangeOrderListResponseABM 	<ul style="list-style-type: none"> CreateECOABM CreateECOResponseABM
ABCS	<ul style="list-style-type: none"> ProcessEngineeringChangeOrderAgileReqABCS 	<ul style="list-style-type: none"> CreateEngineeringChangeOrderListEbizProvABCImpl
EBS	<ul style="list-style-type: none"> EngineeringChangeOrderEBS (CreateEngineeringChangeOrderList Operation) 	<ul style="list-style-type: none"> EngineeringChangeOrderResponseEBS (CreateEngineeringChangeOrderListResponse Operation)
BPEL	<ul style="list-style-type: none"> CreateQueueService QueueProcessorServiceImpl 	---
ESB Service	<ul style="list-style-type: none"> ACSAXMLJMSConsumer ACSAXMLFileConsumer CreateQueueControlService QueueProcessorService EngineeringChangeOrderService 	---

Component Locations

ABO, ABM & Common XSD files	http://<servername>:<portname>/AIAComponents/ApplicationObjectLibrary/Agile/V1/sc hemas
WSDL files	http://<servername>:<portname>/AIAComponents/ApplicationObjectLibrary/Agile/V1/ws dls

Integration Services for CO Release

1. EngineeringChangeOrderEBS

EngineeringChangeOrderEBS is the Enterprise Business Service, which exposes the operations related to the Engineering Change Order Integration on the *EngineeringChangeOrder* EBO.

The following are the routing rules:

EngineeringChangeOrderEBS ESB service

1. ProcessEngineeringChangeOrderAgileReqABCS
 - Route *CreateEngineeringChangeOrderListEBM* to *CreateEngineeringChageOrderListEbizProvABCImpl*

EngineeringChangeOrderResponseEBS ESB service

1. CreateEngineeringChageOrderListEbizProvABCImpl
 - Route *CreateEngineeringChangeOrderListResponseEBM* to *ProcessEngineeringChangeOrderAgileReqABCS*

2. ProcessEngineeringChangeOrderAgileReqABCS

ProcessEngineeringChangeOrderAgileReqABCS is used for transforming *AgileCreateEngineeringChangeOrderListABM* into *CreateEngineeringChangeOrderListEBM*. This service invokes the *CreateEngineeringChangeOrder* operation on *EngineeringChangeOrderEBS* for creation of ECO in Oracle EBS. Based on the status of ECO creation in Oracle EBS, this service updates the Queue Status. Also the Transfer status attribute in Change Order is updated by this service.

This service is implemented as *Asynchronous* BPEL Process

1. The *QueueController* creates the *AgileCreateEngineeringChangeOrderListABM* and invokes the *ProcessEngineeringChangeOrderAgileReqABCS*.
2. Transforms the *AgileCreateEngineeringChangeOrderListABM* to *CreateEngineeringChangeOrderListEBM* and invokes the *CreateEngineeringChangeOrder* operation on *EngineeringChangeOrderEBS* with *CreateEngineeringChangeOrderListEBM* as input. This is routed for creation of ECO in Oracle EBS.

3. *CreateEngineeringChangeOrderListResponseEBM* is received from *EngineeringChangeOrderEBS* and based on the status of ECO creation in Oracle EBS, The *QueueController* is invoked to update the status of the Queue Message.
4. *ProcessEngineeringChangeOrderAgileReqABCS* transforms *CreateEngineeringChangeOrderListResponseEBM* into *AgileUpdateEngineeringChangeOrderListABM*, which is sent as an input to the Agile Web Service.
5. The web services update the transfer status of the Change Order in Agile which will be predefined P2 or P3 attributes on ECO object in Agile. *AgileUpdateEngineeringChangeOrderListResponseABM* is sent back to *ProcessEngineeringChangeOrderAgileReqABCS*.

Transformations

1. *AgileCreateEngineeringChangeOrderListABM_to_CreateEngineeringChangeOrderEBM.xsl*
Transforms *AgileCreateEngineeringChangeOrderListABM* to *CreateEngineeringChangeOrderListEBM*
2. *CreateEngineeringChangeOrderResponseEBM_to_AgileUpdateEngineeringChangeOrderListABM.xsl*
Transforms *CreateEngineeringChangeOrderResponseEBM* to *AgileUpdateEngineeringChangeOrderListABM*

3. CreateEngineeringChangeOrderListEbizProvABCSEmpl

This is a single operation service. This accepts an Engineering Change Order containing Item and BOMs information message as a request and returns a response.

In Agile to Oracle EBS flow, *CreateEngineeringChangeOrderListEbizProvABCSEmpl* is used for transforming *CreateEngineeringChangeOrderListEBM* into *CreateECOABM*, which invokes the *CreateEngineeringChangeOrder* operation in Oracle EBS.

In return flow, OA Adapter sends *CreateECOResponseABM*, which is transformed by *CreateEngineeringChangeOrderListEbizProvABCSEmpl* into *CreateEngineeringChangeOrderListResponseEBM*.

This service is implemented as *Asynchronous* BPEL Process.

CO Release Integration Customization Points

Agile

ProcessEngineeringChangeOrderAgileReqABCS (Agile Process ECO requestor flow)	AgileCreateEngineeringChangeOrderListABM_to_CreateEngineeringChangeOrderListEBM_Custom.xsl	ReqABM to ReqEBM (main)
	AgileCreateEngineeringChangeOrderListABM_to_CreateEngineeringChangeOrderListEBM_Impl.xsl	ReqABM to ReqEBM (custom elements)
	CreateEngineeringChangeOrderListEBM_EBMHeader_Custom.xsl	EBM to EBMHeader (custom elements)
	CreateEngineeringChangeOrderListEBM_EBMHeader_Impl.xsl	EBM to EBMHeader (main)

CreateEngineeringChangeOrderListResponseEBM_to_UpdateEngineeringChangeOrderListEBM_Impl.xsl	RespEBM to ReqEBM (main)
UpdateEngineeringChangeOrderListEBM_to_AgileUpdateEngineeringChangeOrderListABM_Impl.xsl	ReqEBM to ReqABM (main)

Oracle E-Business Suite

CreateEngineeringChangeOrderListEbizProvABCSImpl	ECOInputEBMtoABM_Custom.xsl	Custom transformations for Engineering Change Order Request EBM to Request ABM
	ECOResponseABMtoECOResponseEBM_Custom.xsl	Custom transformations for Engineering Change Order Response ABM to Response EBM

Queue Manager

The Queue Management feature in the PIP caters to the following requirements:

- An Event to produce filtered payload to a File Destination or JMS Destination.
- The Payload is defined using a standard XSD.
- The files or JMS Messages produced by Events are sequenced in the order in which the objects are released.

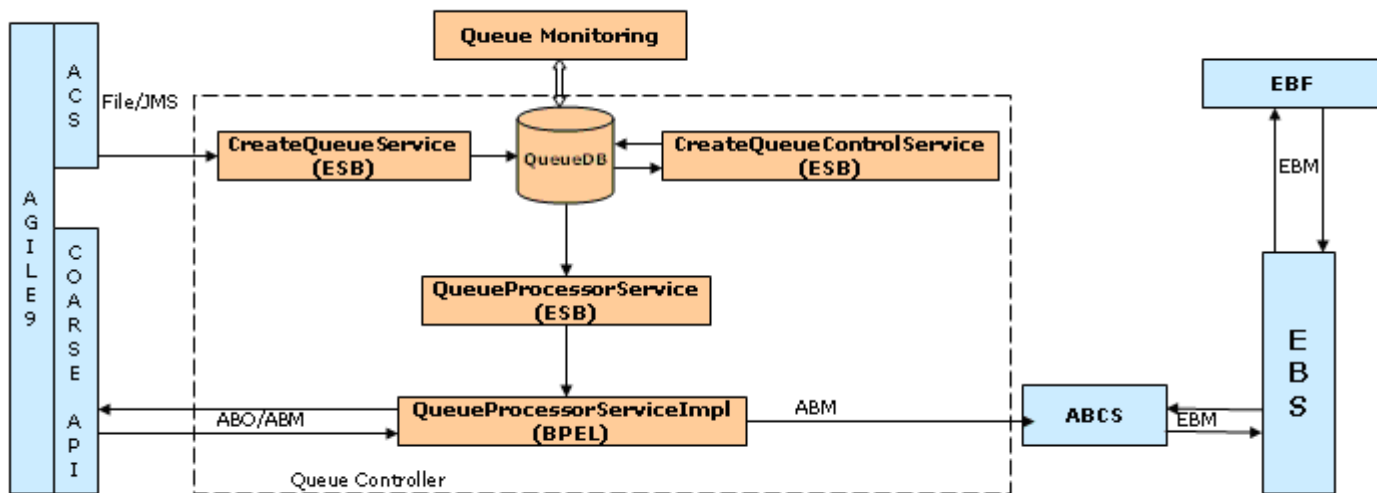
Note	These requirements are leveraged using the Agile Content Service (ACS). ACS has the ability to produce payload to a File or JMS destination. The payload is based on filtered configured for the ACS Event defined by Agile provided AXML schema definition. Also the ACS transmits the messages in the order in which the ATOs are released.
------	---

- A Queue to manage the order of Messages.
- A Queue Monitoring UI to enable reordering and resubmitting unprocessed messages.
- The Queue manages the payloads based on the Business Process for which the message is produced by the Event.
- A Queue controlling mechanism
 - Triggers the Business Flow based on the business process of message.
 - Processes the messages sequentially depending upon the order specified in the message (the highest order message is picked first for processing).
 - A message is not picked for processing unless; the processing of the previous message is complete.
 - The order of the messages, which have not been picked for processing, can be re-ordered.

Solution

The Queue Management Solution comprises of the following components:

- Queue DB: The database persist the data related to a Queue messages.
- Queue Controller: Polls for new Event payloads and add them to the Queue DB. The highest priority message for each Business Process is picked and processed sequentially to trigger its Business Flow.
- Queue Monitoring: UI which monitors the Queue message status supports Re ordering of priorities of the Queue Messages. Also provides the facility to re submit the un processed messages.



Queue Controller

A polling strategy on the Queue DB is used for addressing the Queue Management business requirements. The Queue Controller provides an ECO system to ensure that this polling strategy works in tandem to ensure the following:

- All Event transmitted File/JMS Messages are added to the Queue.
- At any given point of time there is only one pending message in the control table
- Once the processing of a message in control table is complete, insert the highest priority queue message from the queue table to the control table.
- In case the Integration flow errors out, the queue manager will wait until the message is resubmitted or removed.

Queue Schema

In order to support the above solution flow a polling strategy similar to "PollingControlTableStrategy"

is used. Two tables are used to manage the sequential processing and reordering of messages.

The first table QUEUE_TABLE will have all the queue messages that are being provided by the Event trigger. The QUEUE_CONTROL_TABLE table will store the relevant information of the message from the QUEUE_TABLE which has not been processed yet.

The Queue Manager needs to ensure that there is only one message in the control table which is not yet processed. When the processing of a message is complete a Pending message form the Queue table is inserted into this table. This would facilitate the Sequential processing of message. Also since all the pending messages are stored in the Queue table, they could be reordered.

Queue Monitor

When a Change Order is released by Agile Content Service (ACS), it is picked up by the Queue Controller. The Queue Monitor displays a list of all the Change Orders awaiting processing and facilitates you to reorder their sequence of processing.

For complete details on Queue Monitor, refer Agile to Oracle E-Business Suite Integration User Guide.

ORACLE® Application Integration Architecture
[Logout](#)

Filter

Filters:

Criteria:

Change Order Queue

	Reference	Change Number	Release Time	Processed Time	Process Status
<input type="checkbox"/>	ATO02555	ECO02499	26-Jun-2008 05:57:39 PDT	26-Jun-2008 05:58:13 PDT	Pending
<input checked="" type="checkbox"/>	ATO02552	ECO02498	26-Jun-2008 05:13:43 PDT	26-Jun-2008 05:14:22 PDT	Completed
<input checked="" type="checkbox"/>	ATO02551	ECO02496	26-Jun-2008 05:10:19 PDT	26-Jun-2008 05:11:00 PDT	Completed
<input checked="" type="checkbox"/>	ATO02550	ECO02495	26-Jun-2008 04:56:40 PDT	26-Jun-2008 05:08:02 PDT	Completed
<input checked="" type="checkbox"/>	ATO02547	ECO02492	26-Jun-2008 04:23:26 PDT	26-Jun-2008 04:24:07 PDT	Completed
<input checked="" type="checkbox"/>	ATO02542	ECO02487	26-Jun-2008 03:52:13 PDT	26-Jun-2008 03:57:18 PDT	Completed
<input checked="" type="checkbox"/>	ATO02531	EC2478	26-Jun-2008 01:59:30 PDT	26-Jun-2008 02:06:47 PDT	Completed
<input checked="" type="checkbox"/>	ATO02529	ECO02475	26-Jun-2008 01:40:12 PDT	26-Jun-2008 01:40:54 PDT	Completed
<input checked="" type="checkbox"/>	ATO02528	ECO02473	26-Jun-2008 01:21:00 PDT	26-Jun-2008 01:23:27 PDT	Completed
<input checked="" type="checkbox"/>	ATO02526	ECO02471	26-Jun-2008 01:06:11 PDT	26-Jun-2008 01:10:26 PDT	Completed
<input checked="" type="checkbox"/>	ATO02524	ECO02468	26-Jun-2008 00:53:40 PDT	26-Jun-2008 01:01:57 PDT	Completed
<input checked="" type="checkbox"/>	ATO02523	EC2446	25-Jun-2008 23:35:18 PDT	26-Jun-2008 01:36:35 PDT	Completed
<input checked="" type="checkbox"/>	ATO02521	EC2450	25-Jun-2008 23:12:54 PDT	26-Jun-2008 01:36:35 PDT	Completed

Queue Manager Services

The following services are deployed as part of the Queue Manager:

1. CreateQueueService
2. CreateQueueControlService
3. QueueProcessorService
4. QueueProcessorServiceImpl

CreateQueueService

The CreateQueueService is implemented as an ESB Routing Service. An Adapter Service (File/JMS Adapter) polls on the destinations for any Event payloads. The payload is in the form of aXML files. This service receives message as a binary element (aXML File). For each payload received the service inserts a new row into the QUEUE table. An Adapter Service (DB Adapter) is used for the same. The Toplink solution generates the required schema from the table for this DB Adapter.

- The service uses transformation services to populate any NOT NULL columns in the table.
- OBJECT_REFERENCE is inserted with the file name of the aXML file using the ESB header transformation extension functions.
- PROCESS_STATUS is Pending for the newly inserted row.
- PROCESS_PRIORITY is captured from the file name. (ACS can be configured to append an default order for the file name)

CreateQueueControlService

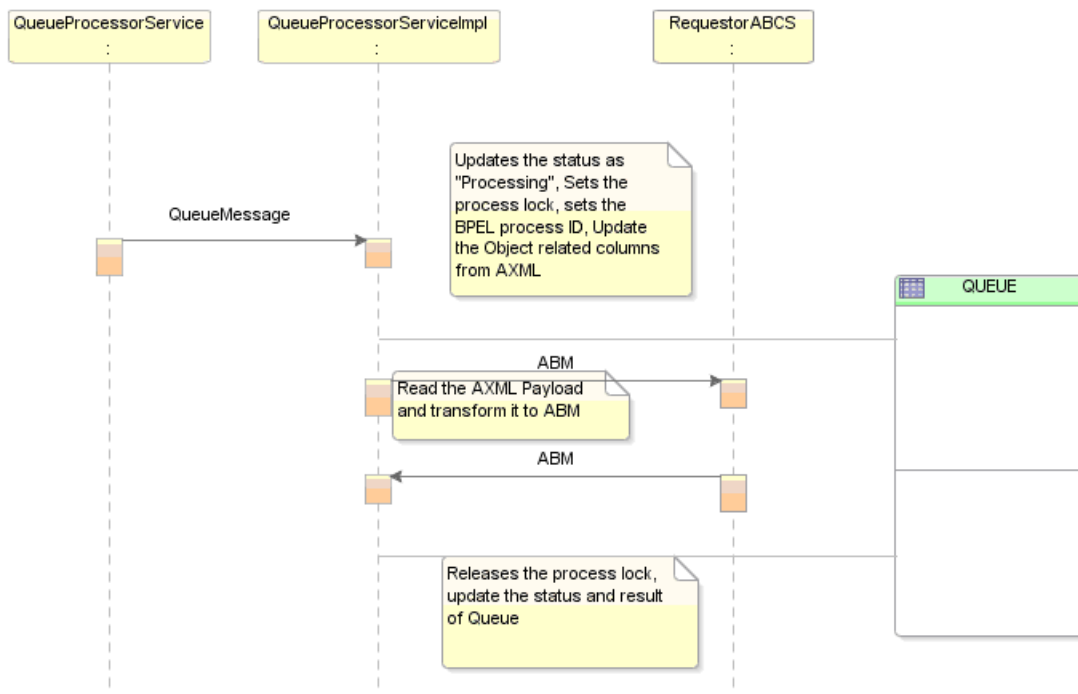
The CreateQueueControlService is implemented as an ESB Routing Service. A DB Adapter polls on the QUEUE_CONTROL_TABLE table. If there are no rows which are in Pending status, the CreateQueueControlService invokes a DB Adapter service which executes a custom SQL. This SQL identifies the highest priority pending Queue message from QUEUE_TABLE table and inserts the same in QUEUE_CONTROL_TABLE table.

This polling strategy ensures that at any point of time there is only one Pending message in the QUEUE_CONTROL_TABLE table. Once the Pending message is processed and status completed, a new Pending message is inserted from QUEUE_TABLE table to the QUEUE_CONTROL_TABLE table. When the status for a message is completed in the QUEUE_CONTROL_TABLE that row would be deleted from the table.

QueueProcessorService

The QueueProcessorService is implemented as an ESB service which acts like an Interface and provides a façade in front of the QueueProcessorServiceImpl service. A DB Adapter polls on the QUEUE_CONTROL table for any Pending messages. A Pending message in the table is routed to the QueueProcessorServiceImpl service which processes the message. Based on the result from the implementation service the status of the message is updated in the control table.

QueueProcessorServiceImpl



The primary task of this service is to invoke the RequestorABCS. The Response from RequestorABCS is processed and the Queue is updated with processing status.

Input: The QueueMessage generated by Toplink solution in the QueueProcessorService is used as the input for this Service.

Output: QueueStatusMessage which contains the status and result of processed Queue message.

#	Name	Step Description
1	QueueProcessorService Invokes QueueProcessorServiceImpl process	The <i>QueueProcessorService</i> invokes <i>QueueProcessorServiceImpl</i> with QueueMessage (generated by Toplink solution for QUEUE table) as input.
2	Invoke UpdateQueueStatus DB Adapter service	The input <i>QueueMessage</i> in this process is assigned with following values to update the Queue message in the Queue DB PROCESS_STATUS -> Processing PROCESS_ID -> BPEL Process Id PROCESS_LOCK -> 1
3	Transform AgileData (aXML) to ABM	The <i>QueueMessage</i> will have the AgileData payload which is transformed to ABM
4	Invoke RequestorABCS	<i>QueueProcessorServiceImpl</i> invokes the <i>RequestorABCS</i> with ABM as input.
5	Invoke Coarse Grained Web Service	<i>RequestorABCS</i> optionally invokes the coarse grained web services to get the ABM populated with any missing information required for the

#	Name	Step Description
		Integration flow.
6	RequestorABCS Transforms ABM to EBM	The response ABM from coarse grained WS is transformed to EBM and an operation on EBS is invoked with EBM as the input.
7	RequestorABCS orchestrates the business flow	The <i>RequestorABCS</i> routes the EBM to the EBS,
8	EBS routes the response to RequestorABCS	The response EBM from EBS is routed to the <i>RequestorABCS</i> which is transformed to ABM and returned to the <i>QueueProcessorServiceImpl</i>
9	QueueProcessorServiceImpl invokes UpdateQueueResult DB Adapter service	The result from the <i>RequestorABCS</i> is used to update the status of Queue in the Queue DB. Also the Process lock is released.

Transformations

The aXML payload is transformed to the ABM which is input for the RequestorABCS. Since the ABM schema is defined on the lines of aXML schema this transformation will be simpler to do in the Jdeveloper XSL Mapper.

Implementation Details

The QueueProcessorServiceImpl is implemented as a *Asynchronous BPEL* process. There are calls to the RequestorABCS, DB Adapters for updating Queue status and invoking the RequestorABCS. These involve some logic (parsing the aXML payload) which cannot be achieved using the ESB.

Note	The QueueID is used for correlation set between the QueueProcessorServiceImpl and the RequestorABCS.
------	--

Error Management

All errors in the Integration flow are handled in the RequestorABCS and EBF. Any such errors leading to failure of the Queue processing will be handled in this process. As a result of such error the Queue Status and Result with failure status is updated in the Queue DB.

Process Integration for Change Order Update

This chapter includes the following:

▪ Change Order Update Process in Oracle EBS.....	35
▪ CO Update Process in Agile	36
▪ CO Update Process Integration Solution Assumptions	38
▪ CO Update Integration Sequence.....	39
▪ AIA Services for CO Update	42
▪ CO Update Integration Customization Points	45

This process is triggered by Oracle EBS.

There are two cases when Change Order information needs to flow from Oracle EBS system to the PLM system. Two of these cases have restrictions in a multi-org environment, simply because of the centralized (Agile) to decentralized (Oracle EBS) nature of the interface.

1. Change Order gets completely implemented in the Oracle EBS system. This step may be configured to trigger a Change Order status change in the PLM system, and/or to trigger a Change Order Page Two or Page Three field update in the PLM system. Since a Change Order may be created in multiple organizations in e-Business Suite, and can have different statuses in each organization, this operation is carried out only when the Change Order is completely implemented in all the organizations that it has been created in the ERP system.
2. Change Order line status or effectivity date changes in the ERP system. In either case, the effectivity date of the Change Order line in ERP system is reflected on the corresponding line in Agile. This update can only be carried out if the Change Order is created in only one organization in the ERP system, or if there is a one-to-one correspondence between ERP organizations and Agile sites for posting Change Order data.

Change Order Update Process in Oracle EBS

There are two types of events which can cause a Change Order Update

1. A modification of the scheduled Effectivity Date of a Change line.
2. An update to a Change line's implementation status.

While the effectivity dates and implementation statuses in the Oracle EBS are tracked at a BOM line level, these are tracked at a Change Line level in Agile. When a business need requires update of any of the supported ECO attributes in Agile from Oracle EBS, it is done by providing a valid mapping to those attributes from values in the Enterprise Business Object.

CO Update Process in Agile

The term Change refers to all objects of the “Change” base class in Agile that are supported in the New Product Introduction and Manufacturing Update processes, viz. ECOs, MCOs, SCOs.

Inputs

The following parameters are needed as inputs for Change Order Update in Agile:

Mandatory

- Change Number, which will uniquely identify the Change object.
- Field to be updated with the transfer or implementation status of the Change in the ERP system.
- ERP Transfer and Implementation status value (such as Transferred, Implemented, Canceled etc).
 - When an ECO is first transferred to the ERP system, its status in the ERP may be “Open” or “Scheduled”. Upon successful transfer, the ECO Transfer Process updates the “ERP Transfer and Implementation Status” attribute with a value of “Transferred”.
 - As long as the ECO status in ERP system is "Open" or “Scheduled”, this attribute’s value remains "Transferred".
 - If the ECO status in ERP system is "Implemented", the field value will be "Implemented".
 - If the ECO status in ERP system is "Errored", the field value will be "Failed Implementation".
 - If the ECO status in ERP system is "Canceled", the field value will be "Canceled".

Optional

- Last transfer attempt log file.

The organization or site information is not an input to this operation. That’s because Agile PLM contains only a single instance of the Change Order, and not site-specific ones. It becomes relevant to the status update flow in cases where Agile data is posted to multiple organizations in the ERP system. ERP Implementation Status update in Agile shall therefore follow these rules pertaining to multiple organizations:

- The ERP Transfer and Implementation Status field in PLM reflects the status of “Transferred” until the change order is fully implemented in all the ERP organizations that it has been posted to
- When the Change is successfully implemented in all the organizations that it has been posted to, the ERP Transfer and Implementation Status field in PLM is modified to “Implemented”
- If the Change errors implementation in one or more orgs, but is awaiting implementation in other orgs or has been successfully implemented in other orgs, the status will remain “Transferred”.
- If the Change errors implementation in all orgs, the status will be changed to “Errored”. If the Change has been canceled in all orgs, the status will be changed to “Canceled”

Processing

The following steps are carried out as part of this process:

1. The Change number and the specific affected item record is queried for modification
2. New effectivity date for the Change line is updated on to the mapped column in Agile.

The following variations are also possible, and must be achieved by means of mappings and transformations done prior to this step:

1. Update a configured column with the incoming effectivity date only if the incoming status is "Implemented". In such a case, the action specified above remains the same, except that the mapped effectivity date column is updated only if the status is Implemented
2. Update a configured column with a configured value based on the incoming status.

An example of sample rules is as follows:

- a. If the ECO status in ERP is "Scheduled", update the "Affected Items.Status" (an Affected Items tab list-validated flexfield) with the value "Not Implemented".
 - b. If the ECO status in ERP is "Implemented", update the "Affected Items.Status" field with the value "Implemented", and update the "Affected Items.ERP Implementation Date" (an Affected Items tab date flexfield) with the "Effective Date" from the ERP system
3. Another variation of this process may require the ECO status to be changed to the next status (say, Complete or Implemented) if all the rows on the ECO have been implemented.

Change line update may be done in a batch mode using a scheduled process. This means that the effectivity information is read for a bunch of Change lines in the ERP system and sent through the interface. This process may also be triggered by the event of a Change line getting implemented in the ERP system.

The update step may process all the Change lines in the batch before committing, or it may commit each Change line individually before moving on to the next one. This decision should be taken purely from a technical efficiency perspective.

When processing a batch of Change lines, if the update for one record errors out, this step should continue to process other records in the batch. A list of all errors encountered when processing the batch should be accumulated and logged.

Change Status operation

The Change Status in Agile for the Update ECO flow from Oracle EBS to Agile considers status of ECOs in all orgs in Oracle EBS.

Update ECO Provider Service on Agile side takes care of the following requirements:

1. Whenever there is change in status change on any ECO in any org in Oracle EBS, Oracle EBS will send across all ECOs from all orgs in Oracle EBS to Agile.
2. UpdateEngineeringChangeOrderAgileProvImpl Service will check across the status on each ECO sent across by Oracle EBS. If the status is same on ECOs across all orgs in Oracle EBS, same will be updated as Transfer Status field on the ECO in Agile.
3. If all the ECOs in all the orgs in Ebiz have status as 'Implemented', the change in Agile for the

ECO needs to be pushed to implemented status besides updating Transfer Status field on ECO in Agile.

4. For this, the entry in the CHANGE_STATUS table will be used. If there is no entry corresponding to event 'Change Implementation' and Change type same as change type of ECO is Agile like ECO, MCO or SCO etc, only then the change is pushed to next status otherwise the change won't be pushed to next status at all.

To understand this better we will take following example:

1. Suppose there is an ECO C0001 released from Agile to Oracle EBS.
2. ECO C0001 is then created in three orgs in Oracle EBS, namely Org1, Org2, Org3.
3. Now the status is changed on ECO C0001 in Org1, say as "Implemented".
4. Oracle EBS triggers an event that sends across ECO list, which will have C0001 in Org1, C0001 in Org2 and C0001 in Org3. Only C0001 will have complete data including AI data, while C0001 in other Orgs have the header information.
5. When it is received on Agile side, ECO data is updated and workflow status of ECO is changed.
To change the workflow status, follow this rule:
 - If the status of ECO in all Orgs in Oracle EBS is implemented, then only push the ECO in Agile to "Implemented" status.
6. Besides this, the ERP transfer status field is updated on change based on following rule:
 - If the status of ECO is same across all the Orgs in Oracle EBS, then only is this updated in Agile.
7. Given this example, if the Status is implemented across all the C0001 ECOs in all Orgs, ERP Transfer status field on ECO in Agile is updated as 'Implemented'
8. If there is an entry in the CHANGE_STATUS table corresponding to the event 'Change Implemented' & SubClass 'ECO', and workflow mentioned on ECO C0001 in Agile, read the next status and push it to next status mentioned.
9. If there is no entry in table for the data mentioned above, no attempt will be made to push ECO C0001 to next status.

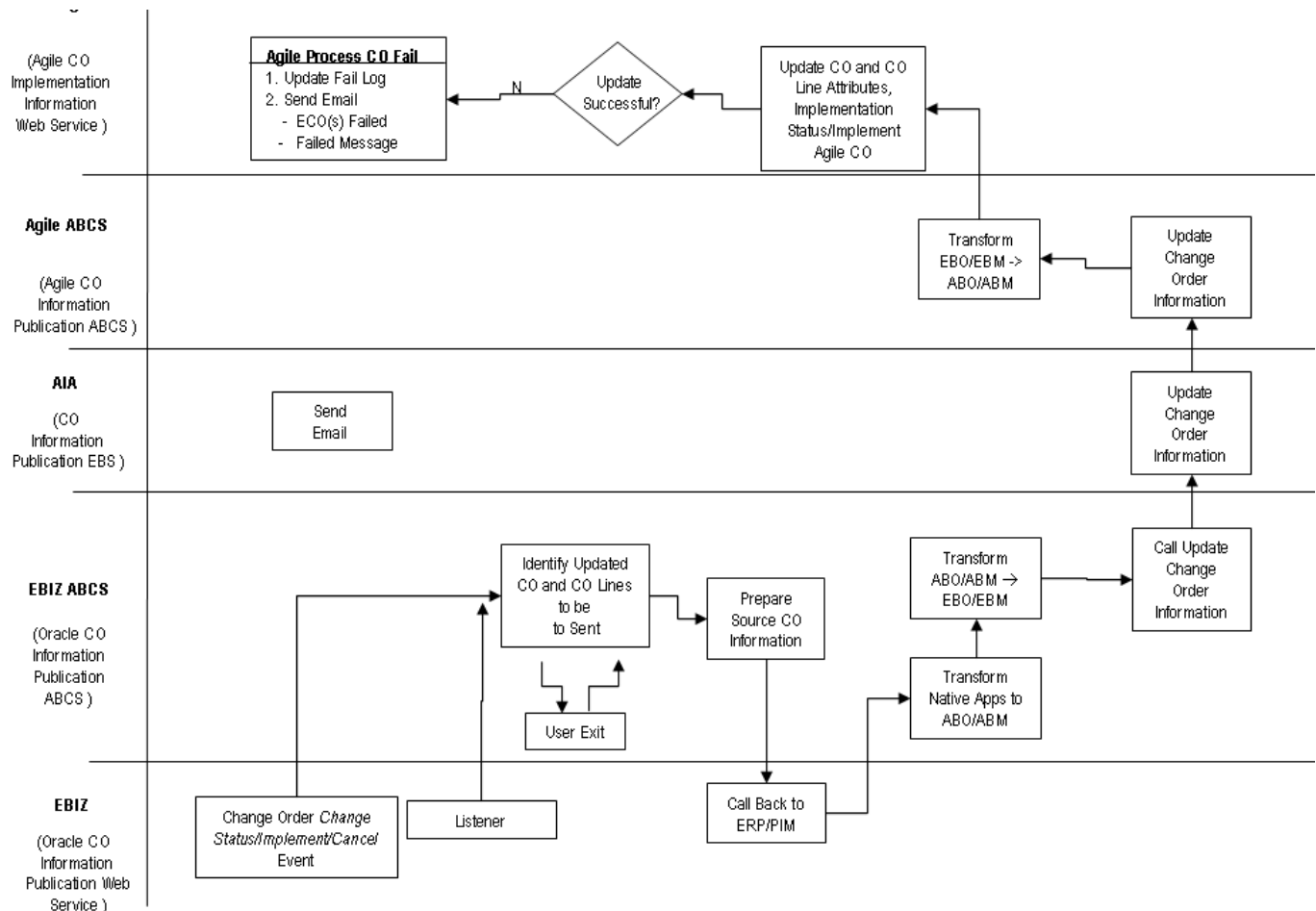
CO Update Process Integration Solution Assumptions

This release of the integration intends to solve a use case where any given Change is interfaced to only one destination system.

It is assumed that the log information pertaining to the complete end to end process of transferring a Change object from Agile to Oracle is available as a file to this operation.

CO Update Integration Sequence

The following is the Integration Sequence for Change Order Update from Oracle E-Business Suite to Agile.



CO Update Integration Flow

Change Order Update Process flows as follows:

1. The requester ABCS, defined as a "synchronous" process, receives a list of Change IDs from the Oracle EBS concurrent program. The list of IDs is those that have "last_update_date" greater than the "last run date" of the concurrent program.
2. The requester BPEL process filters the list of IDs to a list of IDs that are present in the XREF tables in the FMW layer. This provides a list of change orders that were actually from the Agile system alone. The original list of Change IDs may be those that are from non-Agile sources as well.
3. The BPEL process then makes an ESB call out for end point virtualization. This ESB layer has an OA Adapter which calls the PL/SQL API that provides the Change Order details that is

needed to be sent out. This is the Change Order ABM.

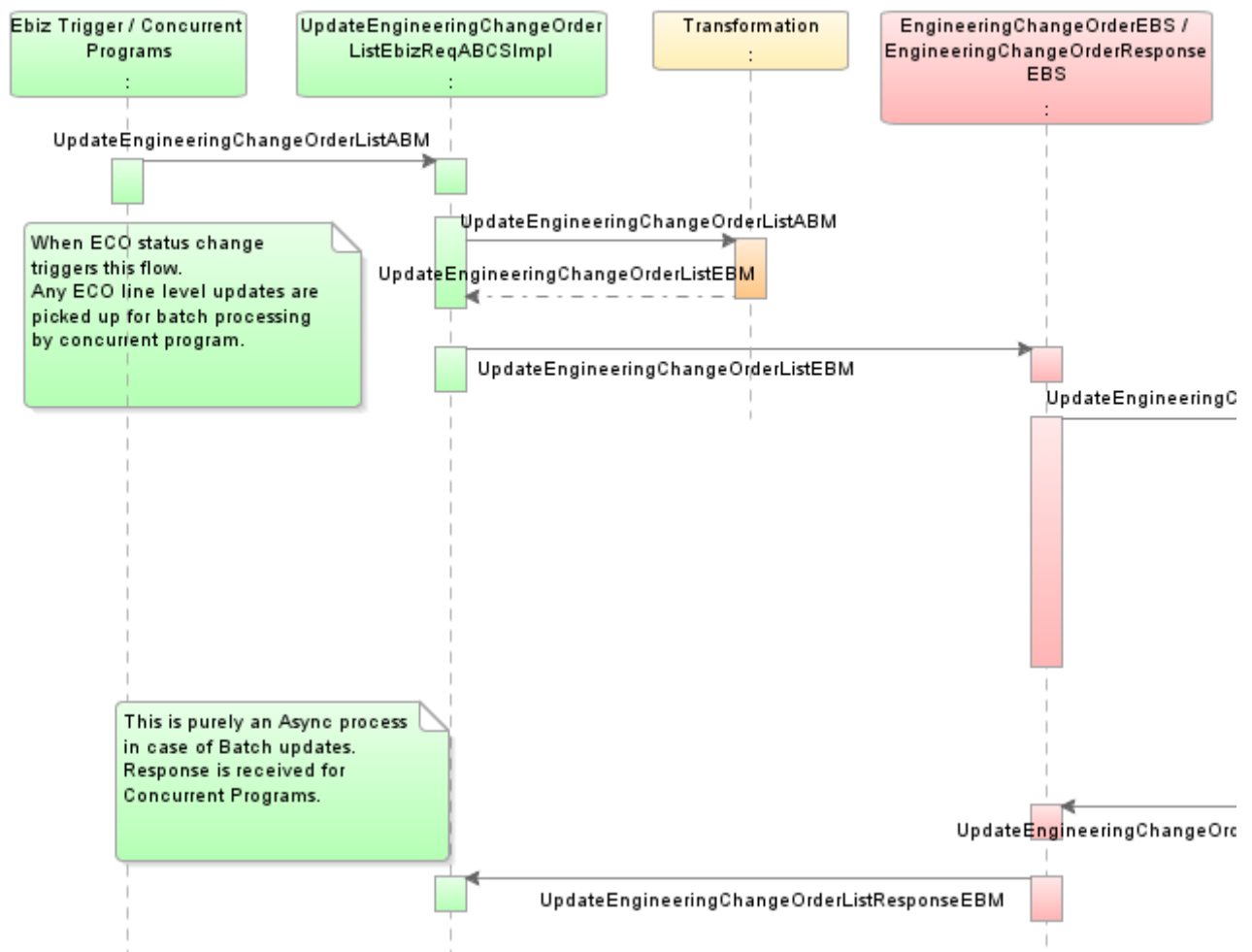
4. A transformation converts the ABM to an EBM.
5. An asynchronous request-delayed response call is made to the EngineeringChangeOrderEBS with the EngineeringChangeOrderListEBM. This will route to the appropriate provider.
6. The BPEL instance is invoked when the asynchronous call gets back from the provider and provides the status of the transaction back to the caller concurrent program (since that is a synchronous call).

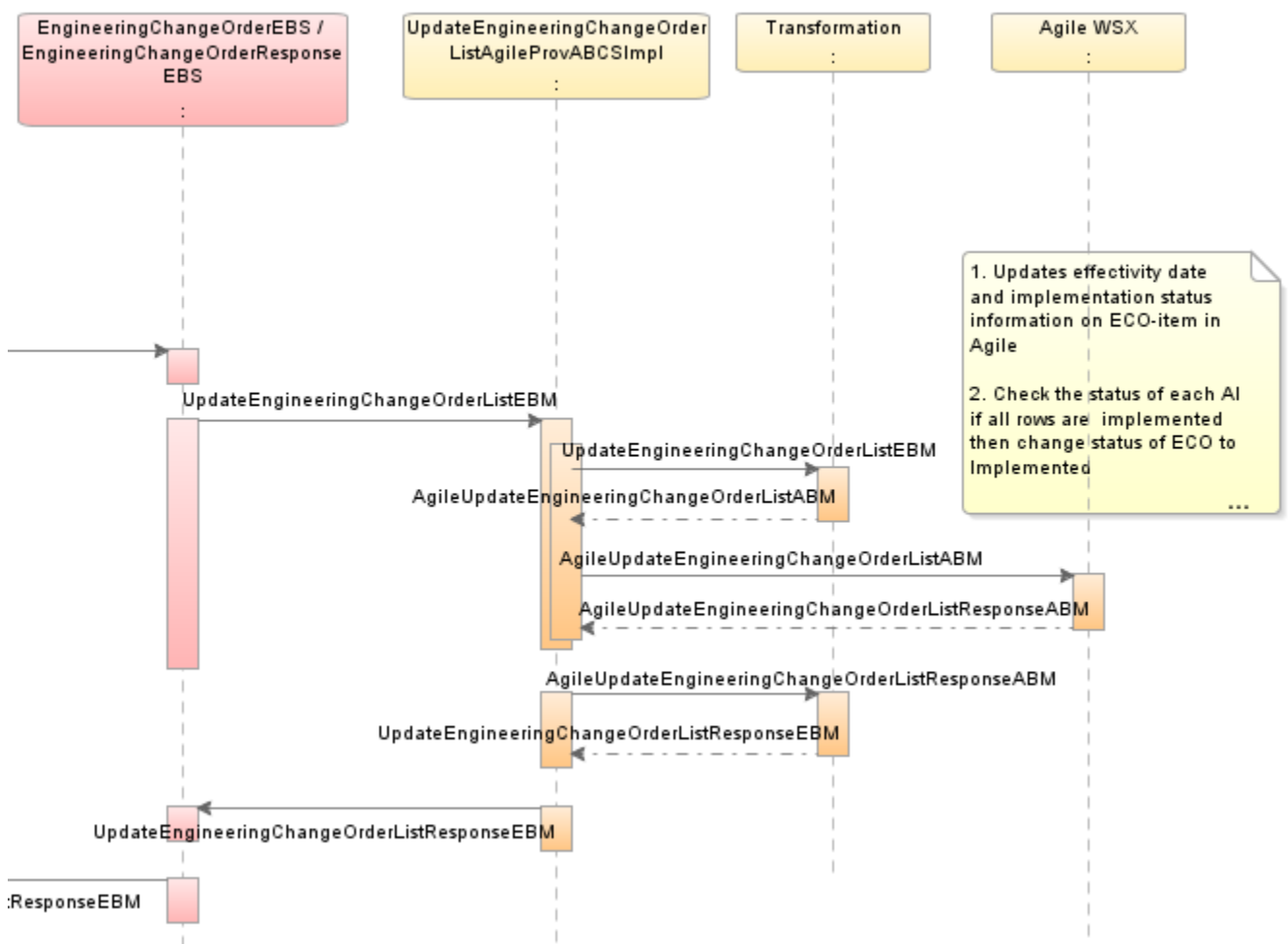
There is no impact on performance since it is a scheduled call and is invoked from backend and not a UI invoke.

7. The concurrent program logs the status of this call.

CO Update Integration Orchestration

Oracle E-Business Suite side



Agile side

#	Activity	Remarks
1	Invoke UpdateEngineeringChangeOrderListEbizReqABCImpl process	When ECO lines are Auto Implemented on reaching Effectivity Dates, <i>UpdateEngineeringChangeOrderListEbizReqABCImpl</i> is triggered. <i>UpdateEngineeringChangeOrderEBM</i> is created inside <i>UpdateEngineeringChangeOrderListEbizReqABCImpl</i> .
2	UpdateEngineeringChangeOrderListEbizReqABCImpl invokes the EngineeringChangeOrderEBS with UpdateEngineeringChangeOrderList operation	An invoke activity in <i>UpdateEngineeringChangeOrderListEbizReqABCImpl</i> invokes the <i>UpdateEngineeringChangeOrderList</i> operation on <i>EngineeringChangeOrderEBS</i> with <i>UpdateEngineeringChangeOrderListEBM</i> as the input.
3	EngineeringChangeOrderEBS routes UpdateEngineeringChangeOrderListEBM to UpdateEngineeringChangeOrderListAgileProvABCImpl	<i>EngineeringChangeOrderEBS</i> routes <i>UpdateEngineeringChangeOrderListEBM</i> to <i>UpdateEngineeringChangeOrderListAgileProvABCImpl</i>

#	Activity	Remarks
4	UpdateEngineeringChangeOrderListAgileProvABCS Impl does the transformation	<i>UpdateEngineeringChangeOrderListAgileProvABCSImpl</i> transforms <i>UpdateEngineeringChangeOrderListEBM</i> into <i>AgileUpdateEngineeringChangeOrderListABM</i>
5	UpdateEngineeringChangeOrderListAgileProvABCS Impl invokes Agile Web Services	<p>Agile Web Services are invoked with <i>AgileUpdateEngineeringChangeOrderListABM</i> as input.</p> <p>The Effectivity dates and implementation status of ECO is updated in agile.</p> <p>A check is made to see if all affected item rows are moved into implemented status.</p> <p>The ECO status will be moved to Implemented when all affected items are implemented.</p> <p><i>AgileUpdateEngineeringChangeOrderListResponseABM</i> is sent back to the <i>UpdateEngineeringCUpdateEngineeringChangeOrderListAgileProvABCSImpl</i></p>

AIA Services for CO Update

Core AIA Components for CO Update

The Process Integration for COU uses the following industry components:

EBOs	▫ EngineeringChangeOrderEBO
EBMs	▫ UpdateEngineeringChangeOrderListEBM
EBSs	▫ EngineeringChangeOrderEBS

Core Components Locations

EBO & EBM XSD files	http://[HOST:PORT]/AIAComponents/EnterpriseObjectLibrary/Core/EBO/
WSDL files	http://[HOST:PORT]/AIAComponents/EnterpriseBusinessServiceLibrary/Core/EBO/

- For detailed documentation of individual EBOs, click the EBO Name link on the *Integration Scenario Summary* page in the *Oracle AIA Console*. You can also use the *Integration Scenario Summary* page to search for and view integration scenarios that utilize a particular EBO or EBS.

For more information, see *Oracle Application Integration Architecture - Foundation Pack: Core Infrastructure Components Guide*, "Using the BSR," Using the BSR UI to View Integration Scenarios.

- EBOs can be extended, for instance, to add new data elements. These extensions are

protected, and will remain intact after a patch or an upgrade.

For more information, see *Oracle Application Integration Architecture – Foundation Pack: Integration Developer's Guide*, “Extensibility for AIA Artifacts.”

Oracle EBS & Agile Components for CO Update

Services	Oracle EBS (Requester)	Agile (Provider)
ABMs	<ul style="list-style-type: none"> EngineeringChangeOrderABM 	<ul style="list-style-type: none"> AgileUpdateEngineeringChangeOrderListABM AgileUpdateEngineeringChangeOrderListResponseABM
ABCS	<ul style="list-style-type: none"> UpdateEngineeringChangeOrderListEbizReqABCS 	<ul style="list-style-type: none"> UpdateEngineeringChangeOrderListAgileProvABCS
EBS	<ul style="list-style-type: none"> EngineeringChangeOrderEBS 	---

Component Locations

ABO, ABM & Common XSD files	http://<servername>:<portname>/AIAComponents/ApplicationObjectLibrary/Agile/V1/sc hemas
WSDL files	http://<servername>:<portname>/AIAComponents/ApplicationObjectLibrary/Agile/V1/ws dls

Integration Services for CO Update

1. EngineeringChangeOrderEBS

EngineeringChangeOrderEBS is the Enterprise Business Services which exposes the following operations related to the Engineering Change Order Integration on the EngineeringChangeOrder EBO.

The following are the routing rules:

EngineeringChangeOrderEBS ESB service

- UpdateEngineeringChangeOrderListEbizReqABCImpl
 - Route *UpdateEngineeringChangeOrderListEBM* to *UpdateEngineeringChangeOrderListAgileProvABCImpl*

EngineeringChangeOrderResponseEBS ESB service

- UpdateEngineeringChangeOrderListAgileProvABCImpl
 - Route *UpdateEngineeringChangeOrderListResponseEBM* to *UpdateEngineeringChangeOrderListEbizReqABCImpl*

2. UpdateEngineeringChangeOrderListEbizReqABCSEmpl

- The requester ABCS defined as a “synchronous” process, receives a list of Change IDs from the Oracle EBS concurrent program. The list of IDs are those which have last_update_date greater than the last run date of the concurrent program.
- The requester BPEL process filters the list of IDs to a list of IDs that are present in the XREF tables in the FMW layer. This provides a list of change orders that were actually from the Agile system alone. The original list of Change IDs may be those which are from non-Agile sources as well.
- The BPEL process then makes an ESB call out for end point virtualization. This ESB layer has an OA Adapter which calls the PL/SQL API that provides the Change Order details that is needed to be sent out. This is the Change Order ABM.
- A transformation converts the ABM to an EBM.
- An asynchronous request-delayed response call is made to the *EngineeringChangeOrderEBS* with the *EngineeringChangeOrderListEBM*. This routes to the appropriate provider.
- The BPEL instance is invoked when the asynchronous call gets back from the provider and provides the status of the transaction back to the caller concurrent program (since that is a synchronous call).
- The concurrent program logs the status of this call.

3. UpdateEngineeringChangeOrderListAgileProvABCSEmpl

The *UpdateEngineeringChangeOrderListAgileProvABCSEmpl* updates an Engineering Change Order in Agile. In the Integration the *UpdateEngineeringChangeOrderListAgileProvABCSEmpl* is used to update the Transfer Status related fields.

It is implemented as *Asynchronous* Process.

- *UpdateEngineeringChangeOrderListAgileProvABCSEmpl* is invoked by *EngineeringChangeOrderEBS* with *UpdateEngineeringChangeOrderListReqMsg* which contains *UpdateEngineeringChangeOrderListEBM* as input.
- Transform operation is called to convert the *UpdateEngineeringChangeOrderListEBM* into *AgileUpdateEngineeringChangeOrderListABM*.
- *AgileUpdateEngineeringChangeOrderListABM* is passed as input to the web service operation which carries out the following functionalities for this integration.
 - Update Transfer Status related attributes on ECO (For Agile to Oracle EBS flow)
 - Update Effectivity Dates and Implementation status. If all AI statuses are Implemented change Status of ECO to Implemented. (For Oracle EBS to Agile flow)
- *AgileUpdateEngineeringChangeOrderListResponseABM* is received on successful execution of Coarse Grained API UpdateChange.
- If the UpdateChange service operation fails on the Agile side, a new Fault is generated and sent across with appropriate error message.

CO Update Integration Customization Points

Oracle E-Business Suite

UpdateEngineeringChangeOrderListEbizReqABCSImpl	XformABMTtoECOEBM_Custom.xml	Custom transformations for Engineering Change Order Request ABM to Request EBM
---	------------------------------	--

Agile

UpdateEngineeringChangeOrderListAgileProvABCSImpl (Agile Update ECO provider flow)	AgileUpdateEngineeringChangeOrderListResponseABM_to_UpdateEngineeringChangeOrderListResponseEBM_Custom.xml	RespABM to RespEBM (custom element)
	AgileUpdateEngineeringChangeOrderListResponseABM_to_UpdateEngineeringChangeOrderListResponseEBM_Impl.xml	RespABM to RespEBM (main)
	UpdateEngineeringChangeOrderListEBM_to_AgileUpdateEngineeringChangeOrderListABM_Impl.xml	ReqEBM to ReqABM (main)

Process Integration for Item Attribute Update

This chapter includes the following:

▪ Item Attribute Update Process.....	47
▪ IA Update Process Integration Solution Assumptions	49
▪ IA Update Integration Sequence.....	50
▪ AIA Services for ECO/PREL.....	52
▪ IA Update Integration Customization Points	54

Required as part of the Manufacturing Update process for product information synchronization, the Item Attributes information from Oracle EBS is required to be updated in Agile. The term Item applies to both Parts and Documents in Agile.

Since item attribute information can exist in multiple organizations in the Oracle EBS, it is supported only for cases where the system can accurately determine which Org(s) in ERP to pick up item attribute information from.

Item Attribute Update Process

The update of Item Attribute information from Oracle EBS to PLM is done as a batch process.

The following steps are followed:

1. The batch process kicks off at a scheduled frequency, which is configurable.
2. Retrieve the last date and time as of which the item information update process completed successfully.
3. Retrieve the list of items whose attributes have changed since the last successful run. Also retrieve the list of mapped attribute values for these items. This data is retrieved from one or more organizations as per site-org mappings or from the configured single organization as described above.
4. Update the information into Agile. If the complete update is processed successfully, update the date and time of last successful run.

The input consists of a batch of items whose attribute values need to be updated in Agile. Unique identifiers for the items, along with values for all the attributes mapped for transfer from Oracle EBS to Agile are provided as input.

As of Agile 9.2.2.2, an item number is the only attribute that is required for uniquely identification of the item to be updated. A planned enhancement in the next major release of Agile PLM would require that the item revision be specified as part of the unique key as well, since all the item attributes could potentially be under revision control with that enhancement. Even in Agile 9.2.2.2,

Item Description and Mass can have different values for different revisions of the item.

A unique item revision in Agile is composed of a Rev Number as well as a Change Order number (since it's possible for certain types of Change Orders in Agile to revise an item without changing its revision number). In the context of integration, however, updates to item attributes applies to the latest released revision in most cases.

The following guidelines are followed when querying an item from Agile for update:

- In the first release of integration, developed on Agile 9.2.2.2, only the item number is used to query the item to be updated. The item number will pull up the latest released revision of the item by default, and all updates are made to the latest released revision of the item.
- A future release of the integration should plan on enhancing this capability to accept a revision number (and optionally, a Change number) along with the item number to pull up a unique item revision for update (if a Change number is not specified, the latest released Change corresponding to that revision is assumed).

Processing

Item attribute update is done in a batch mode using a scheduled process. This means that mapped attribute information is read for a bunch of items in the Oracle EBS and sent through the interface. Since this process does not use specific event triggers, it is not easy to identify from the source system which of the mapped item attributes really changed in the elapsed period. The update process, therefore, updates all the mapped attributes every time an item is updated with information from the Oracle EBS.

The update step may process all the items in the batch before committing, or it may commit each item individually before moving on to the next one.

When processing a batch of items, if the update for an item errors out, this step continues to process other items in the batch. All errors encountered when processing the batch are logged.

A status code indicating whether or not all the items in the batch were updated successfully is returned, along with an error message, if an exception was encountered. The error message contains a cumulative log of all the exceptions encountered when processing the items.

Exceptions

In all cases, the exception message displayed to the user must contain the item number that was being updated along with other supporting information as indicated.

- Insufficient privilege to discover, read or update the item
- Invalid field value
 - Invalid list value
 - Invalid data format for text, money or date fields
- Item does not exist

IA Update Process Integration Solution Assumptions

If item information needs to be retrieved from multiple organizations in Oracle EBS, the following constraints must be met in order to support this process:

1. Agile multi-sites is configured
2. There is a one-to-one mapping between Agile sites and Oracle EBS organizations
3. The attribute(s) to be updated with Oracle EBS data are on the Sites tab of the item

Alternately, customers may designate, as a part of configuration parameters, a single Oracle EBS organization from which all item attribute and onhand quantity information is always retrieved. In such a case, it is not necessary to have Agile multi-sites. Any Title Block or Page Two attribute may be updated with ERP data in such a case.

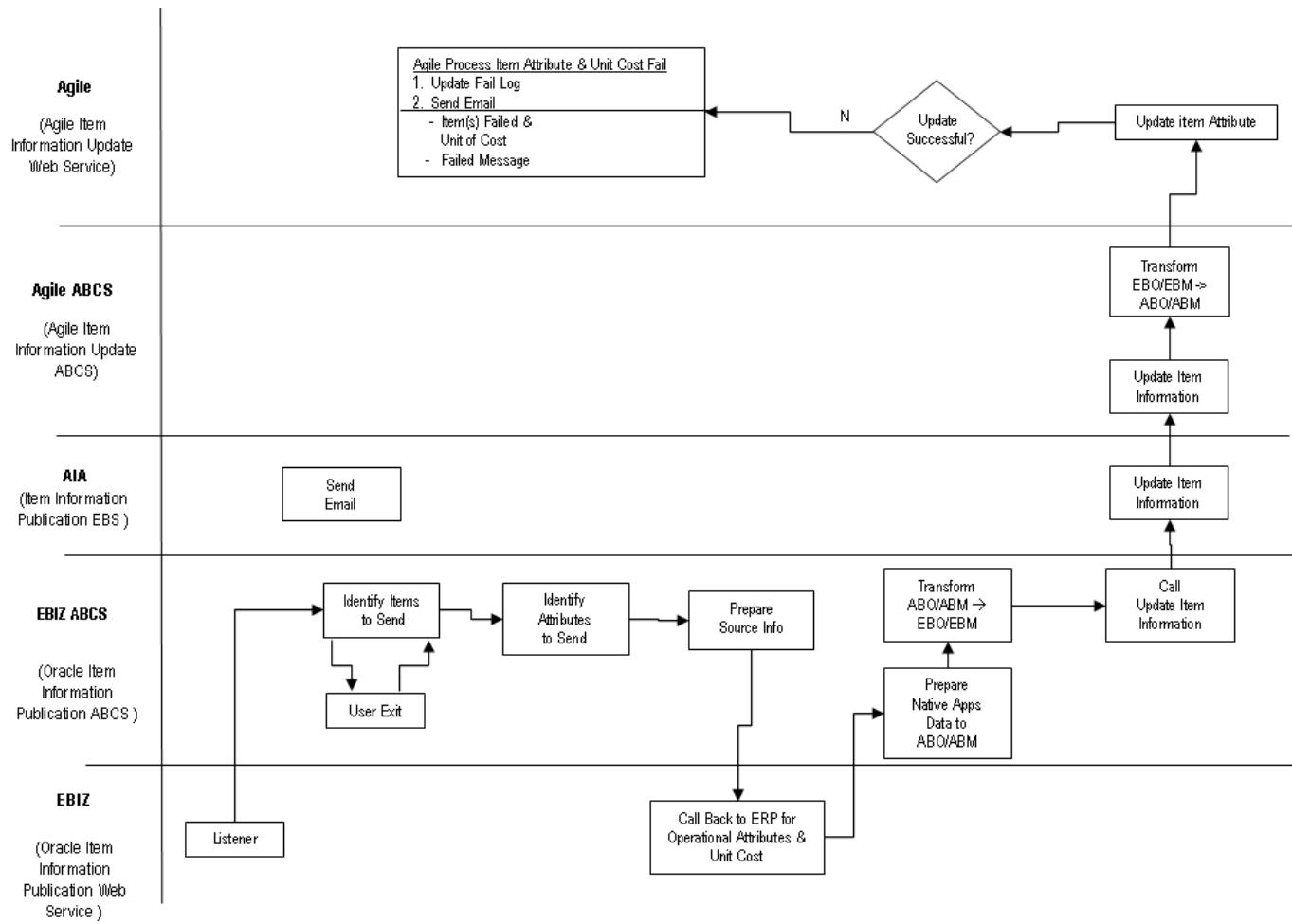
This release of the integration intends to solve a use case where any given Change is interfaced to only one destination system.

The following exception conditions are tested to make sure that they return user-friendly error messages. In all cases, the exception message displayed to the user must contain the Change number that was being updated and the step at which the error occurred, along with other supporting information as needed.

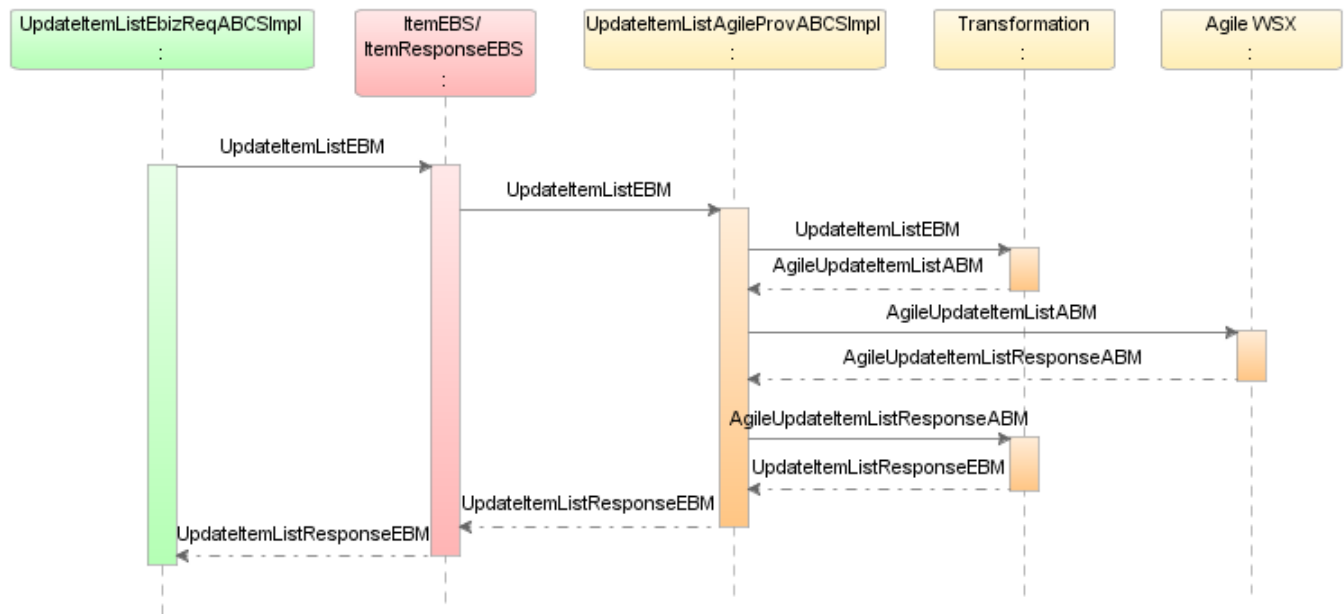
- Insufficient privilege to discover, read, add attachment to or update attributes on the Change object
- Invalid field value (indicate the field for which the value is incorrect, and the value that is being passed on to the field)
- Change object does not exist
- Error when adding attachment

Note	New Item Creation is not part of the scope for this activity. It is assumed that the item exists in both Agile and Oracle EBS, by means of a prior New Part Introduction or independent offline load processes.
------	---

IA Update Integration Sequence



IA Update Integration Orchestration



#	Activity	Remarks
1	Oracle Item Operational Attribute Update Action Trigger	Oracle Item Operational Attribute Update Action trigger Invokes the <i>UpdateItemListEbizReqABCSImpl</i>
2	UpdateItemListEbizReqABCSImpl invokes ItemEBS	An invoke activity in <i>UpdateItemListEbizReqABCSImpl</i> invokes the <i>UpdateItemList</i> operation on <i>ItemEBS</i> with <i>UpdateItemListEBM</i> as input
3	ItemEBS routes the SyncItemListEBM to UpdateItemListAgileProvABCSImpl	<i>ItemEBS</i> routes <i>UpdateItemListEBM</i> as input to <i>UpdateItemListAgileProvABCSImpl</i>
4	UpdateItemListAgileProvABCSImpl invokes Agile's Item Operational Attribute Update Web Service	<p><i>ItemEBS</i> routes <i>UpdateItemListEBM</i> as input to <i>UpdateItemListAgileProvABCSImpl</i>.</p> <p><i>UpdateItemListAgileProvABCSImpl</i> transforms the <i>UpdateItemListEBM</i> to <i>AgileUpdateItemListABM</i> and invokes <i>UpdateItem</i> service operation on Agile Web Service to update Item cost related attribute information for from Oracle to Agile.</p> <p>The <i>AgileUpdateItemListResponseABM</i> is returned back to <i>UpdateItemListAgileProvABCSImpl</i></p>
5	UpdateItemListAgileProvABCSImpl sends response back to the ItemEBS	<i>UpdateItemListAgileProvABCSImpl</i> transforms the <i>AgileUpdateItemListResponseABM</i> to <i>UpdateItemListListResponseEBM</i> sends back this to <i>ItemEBS</i>
6	ItemEBS sends the UpdateItemListListResponseEBM to	<i>ItemEBS</i> sends the <i>UpdateItemListListResponseEBM</i> to <i>UpdateItemListEbizReqABCSImpl</i>

#	Activity	Remarks
	UpdateItemLEbizReqABCImpl	

AIA Services for ECO/PREL

Core AIA Components for IA Update

The Process Integration for ECO/PREL uses the following industry components:

EBOs	▫ ItemEBO
EBMs	▫ UpdateItemLEBM ▫ UpdateItemLEBResponseEBM
EBSs	▫ ItemEBS ▫ ItemResponseEBS

Core Components Locations

EBO & EBM XSD files	http://[HOST:PORT]/AIAComponents/EnterpriseObjectLibrary/Core/EBO/
WSDL files	http://[HOST:PORT]/AIAComponents/EnterpriseBusinessServiceLibrary/Core/EBO/

- For detailed documentation of individual EBOs, click the EBO Name link on the *Integration Scenario Summary* page in the *Oracle AIA Console*. You can also use the *Integration Scenario Summary* page to search for and view integration scenarios that utilize a particular EBO or EBS.

For more information, see *Oracle Application Integration Architecture - Foundation Pack: Core Infrastructure Components Guide*, “Using the BSR,” Using the BSR UI to View Integration Scenarios.

- EBOs can be extended, for instance, to add new data elements. These extensions are protected, and will remain intact after a patch or an upgrade.

For more information, see *Oracle Application Integration Architecture – Foundation Pack: Integration Developer’s Guide*, “Extensibility for AIA Artifacts.”

Oracle EBS & Agile Components for IA Update

Services	Oracle EBS (Requester)	Agile (Provider)
ABMs	▫ UpdateItemLEBM	▫ AgileUpdateListABM ▫ AgileUpdateListResponseABM

ABCS	▫ UpdateItemListEbizReqABCS	▫ UpdateItemListAgileProvABCS
EBS	▫ ItemEBS	▫ ItemResponseEBS
BPEL	---	---
ESB Service	▫ GetItemAttributeService ▫ ItemServiceESB	---

Component Locations

ABO, ABM & Common XSD files	http://<servername>:<portname>/AIAComponents/ApplicationObjectLibrary/Agile/V1/sc hemas
WSDL files	http://<servername>:<portname>/AIAComponents/ApplicationObjectLibrary/Agile/V1/ws dls

Integration Services for IA Update

1. ItemEBS

ItemEBS is the Enterprise Business Services which exposes the following operations related to the Item Attribute Update Integration on the *ItemEBO*.

The following are the routing rules:

ItemEBS ESB service

1. UpdateItemListEbizReqABCSEImpl
 - Route *UpdateItemListEBM* to *UpdateItemListAgileProvABCSEImpl*

ItemResponseEBS ESB service

1. UpdateItemListAgileProvABCSEImpl
 - Route *UpdateItemListResponseEBM* to *UpdateItemListEbizReqABCSEImpl*

2. UpdateItemListEbizReqABCSEImpl

This is called when there is a need to update multiple items in Agile with all the operational attributes as well as attributes such as unit cost from Oracle EBS. This is a push from Oracle EBS to the Agile.

- The requester ABCS, defined as a “synchronous” process, receives a list of Item Ids from the Oracle EBS concurrent program. The list of Ids are those which have "last_update_date" greater than the "last run date" of the concurrent program.
- The requester BPEL process filters the list of Ids to a list of Ids that are present in the XREF tables in the FMW layer. This provides a list of Items that were actually from the Agile system alone. The original list of Item Ids may be those which are from non-Agile sources as well.

- The BPEL process then makes an ESB call out for end point virtualization. This ESB layer has an OA Adapter which calls the PL/SQL API that provides the Item details that is needed to be sent out. This is the Item ABM.
- A transformation converts the ABM to an EBM.
- An *Asynchronous* request-delayed response call is made to the *ItemEBS* with the *ItemListEBM*. This routes to the appropriate provider.
- The BPEL instance gets invoked when the asynchronous call gets back from the provider and provides the status of the transaction back to the caller concurrent program (since that is a synchronous call).
- The concurrent program logs the status of this call.

3. UpdateItemListAgileProvABCSImpl

UpdateItemListAgileProvABCSImpl is used to facilitate the communication between ItemEBS and Agile web service used for updating the Items' operational attributes, item cost information in batch mode in Agile.

- Receives *UpdateItemListReqMsg* that contains *UpdateItemListEBM*
- Transform operation is called to convert the *UpdateItemListEBM* into *AgileUpdateItemListABM*.
- *AgileUpdateItemListABM* is sent as input to the web service operation *UpdateItems* (Coarse Grained APIs on Agile side) to update Items in Agile.
- *AgileUpdateItemListResponseABM* is received on successful execution of Coarse Grained API.
- Transform operation is called to convert the *AgileUpdateItemListABM* to *UpdateItemListResponseEBM*, which is returned as output of this BEPL process.
- If the *UpdateItems* service operation fails on the Agile side, a new Fault is generated and is sent across with appropriate error message.

IA Update Integration Customization Points

Oracle E-Business Suite

UpdateItemListEbizReqABCSImpl	UpdateItemListABMToEBM_Custom.xml	Custom transformations for Item Request ABM to Request EBM
-------------------------------	-----------------------------------	--

Agile

UpdateItemListAgileProvABCSImpl (Agile update item attributes provider flow)	AgileUpdateItemListResponseABM_to_UpdateItemListResponseEBM_Custom.xml	RespABM to RespEBM (custom element)
	AgileUpdateItemListResponseABM_to_UpdateItemListResponseEBM_Impl.xml	RespABM to RespEBM (main)

UpdateItemListEBM_to_AgileUpdateItemListABM_Impl.x sl	ReqEBM to ReqABM (main)
--	-------------------------

Process Integration for Item Balance Update

This chapter includes the following:

▪ Item Balance Update Process	57
▪ IB Update Process Integration Solution Assumptions	58
▪ IB Update Integration Sequence.....	59
▪ AIA Services for IB Update	61
▪ IB Update Integration Customization Points	63

Required as part of the Manufacturing Update process for product information synchronization, the Item Availability, i.e. the Item Balance, information from Oracle EBS is required to be updated in Agile. The term Item applies to both Parts and Documents in Agile.

Since item balance information can exist in multiple organizations in the Oracle EBS, it is supported only for cases where the system can accurately determine which Org(s) in Oracle EBS to pick up item balance information from.

Item Balance Update Process

The Item Balance information in ERP system is stored in three heads -

- Reserved Quantity
- Available Quantity
- On-hand Quantity

These three heads/attributes are not available in Agile's Out-of-the-Box application. To update the item balance information from these three attributes, Agile's Flex-Fields are configured and mapped. This configuration information is given in AIAConfigProperties.xml file (*Multisite_Enabled* property).

Further, an Item in ERP can exist in more than one Org. Any change in any of the three types of quantities may take place in just one or in a few or in all the orgs. For the ERP system to determine, which org corresponds (maps) to which flex-field in Agile, the DVM *AGILE_TARGET_SITE_MAPPING* is used.

Configurations

- When *Multisite_Enabled* property in AIAConfigProperties.xml is set to FALSE, the Page2 or Page3 flexfields in Agile are updated.
- When *Multisite_Enabled* property in AIAConfigProperties.xml is set to TRUE, the flexfields on

Site Tab in Agile are updated.

The Update Process

The update of Item Balance information from Oracle EBS to PLM is carried out as a batch process.

The following steps are followed:

1. The batch process kicks off at a scheduled frequency, which is configurable.
2. Retrieve the last date and time as of which the item information update process completed successfully.
3. Retrieve the list of items whose attributes have changed since the last successful run. Also retrieve the list of mapped attribute values for these items. This data is retrieved from one or more organizations as per site-org mappings or from the configured single organization as described above.
4. Update the information into Agile. If the complete update is processed successfully, update the date and time of last successful run.

The input consists of a batch of items whose quantity values need to be updated in Agile. Unique identifiers for the items, along with values for all the quantities mapped for transfer from Oracle EBS to Agile are provided as input.

IB Update Process Integration Solution Assumptions

If item balance information needs to be retrieved from multiple organizations in Oracle EBS, the following constraints must be met to for an end-to-end process integration:

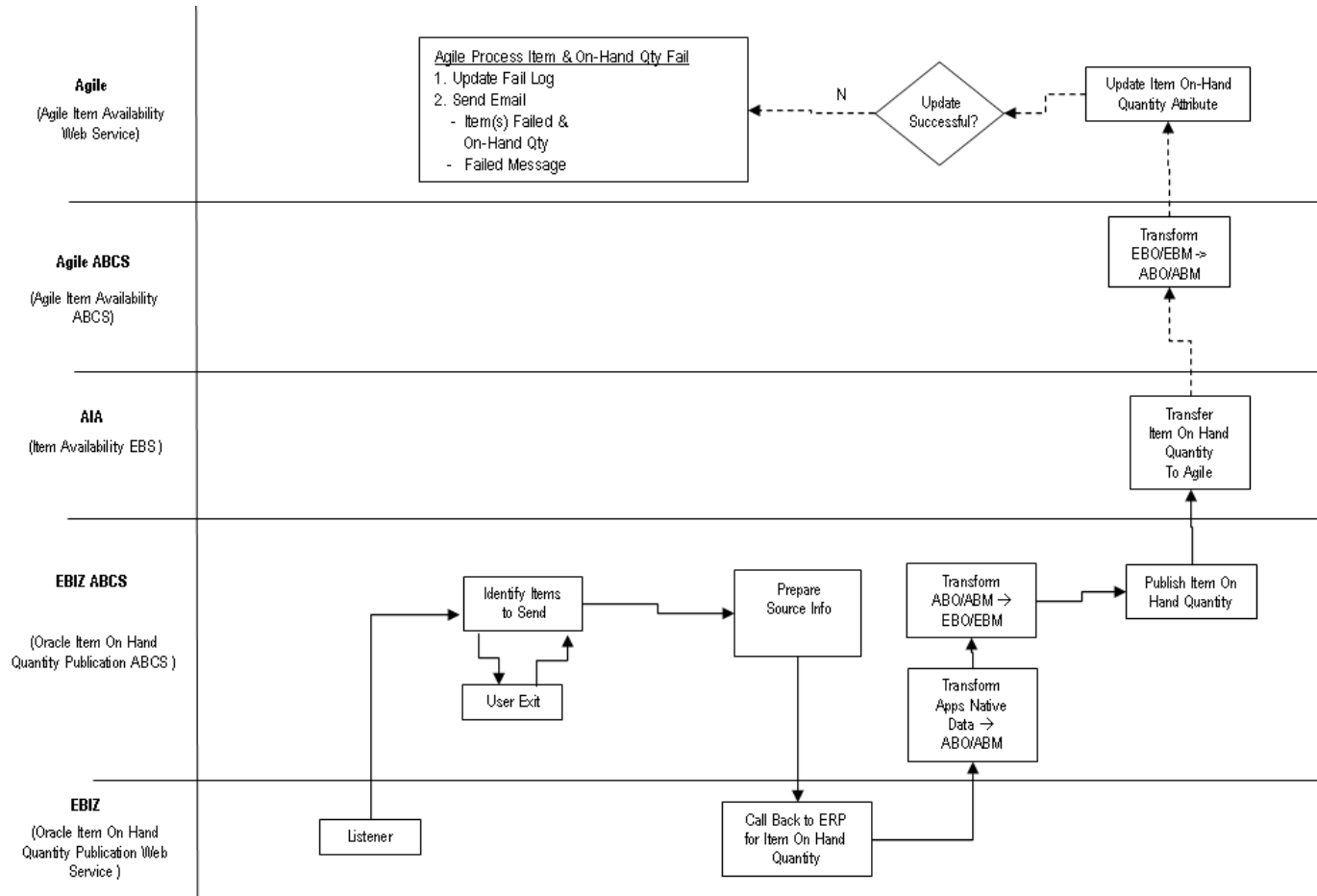
- Agile multi-sites is not configured, the attributes to be updated are on the Page2 or Page3 as configured in AIAConfigProperties.xml file.
- There is a one-to-one mapping between Agile sites and ERP organizations. This is derived from the AGILE_TARGET_SITE_MAPPING DVM.
- The attribute(s) to be updated with Oracle EBS data are on the Sites tab of the item. If the Agile MultiSites are configured.

Alternately, customers may designate, as a part of configuration parameters, a single ERP organization from which onhand quantity information is always retrieved. In such a case, it is not necessary to have Agile multi-sites. Any Title Block or Page Two attribute may be updated with ERP data in such a case.

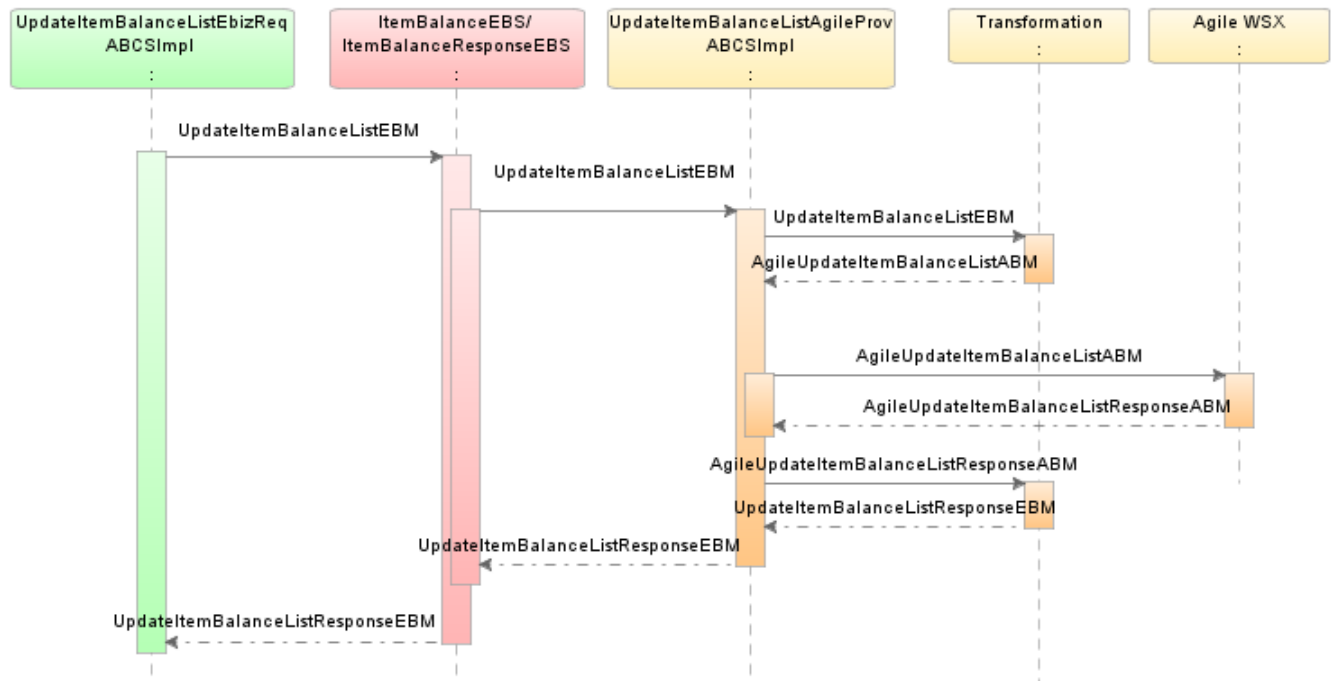
The following exception conditions are considered for return of user-friendly error messages. In all cases, the exception message displayed to the user contains the Change number that was being updated and the step at which the error occurred, along with other supporting information as needed.

- Insufficient privilege to discover, read, add attachment to or update attributes on the Change object
- Invalid field value (indicate the field for which the value is incorrect, and the value that is being passed on to the field)

IB Update Integration Sequence



IB Update Integration Orchestration



#	Activity	Remarks
1	Oracle Item Balance Action triggers	Oracle Item Balance Action Invokes the <i>UpdateItemBalanceListEbizReqABCS</i>
2	<i>UpdateItemBalanceListEbizReqABCSImpl</i> invokes <i>ItemBalanceEBS</i>	An invoke activity in <i>UpdateItemBalanceListEbizReqABCSImpl</i> invokes the <i>UpdateItemBalanceList</i> operation on <i>ItemBalanceEBS</i> with <i>UpdateItemBalanceList</i> as input
3	<i>ItemBalanceEBS</i> invokes the <i>UpdateItemBalanceListAgileProvABCSImpl</i>	<i>ItemBalanceEBS</i> sends <i>SyncItemBalanceListEBM</i> as input to <i>UpdateItemBalanceListAgileProvABCSImpl</i>
4	<i>UpdateItemBalanceListAgileProvABCSImpl</i> invokes Agile's Item Balance Web Service	<i>UpdateItemBalanceListAgileProvABCSImpl</i> transforms the <i>UpdateItemBalanceListEBM</i> to <i>AgileUpdateItemBalanceListABM</i> and invokes the updateItem service operation on Agile Web Service to update Item On-Hand Quantity information from Oracle to Agile. The <i>AgileUpdateItemBalanceListResponseABM</i> is returned back to <i>UpdateItemBalanceListAgileProvABCSImpl</i>
5	<i>UpdateItemBalanceListAgileProvABCSImpl</i> sends response back to the <i>ItemBalanceEBS</i>	<i>UpdateItemBalanceListAgileProvABCSImpl</i> transforms <i>AgileUpdateItemBalanceListResponseABM</i> to <i>UpdateItemBalanceListResponseEBM</i> and returns it to

#	Activity	Remarks
		<i>ItemBalanceEBS.</i>
6	ItemBalanceEBS sends the UpdateItemBalanceListResponseEBM to UpdateItemBalanceListEbizReqABCImpl	<i>ItemBalanceEBS sends the UpdateItemBalanceListResponseEBM to UpdateItemBalanceListEbizReqABCImpl</i>

AIA Services for IB Update

Core AIA Components for IB Update

The Process Integration for Item Balance Update uses the following industry components:

EBOs	▫ ItemBalanceEBO
EBMs	▫ UpdateItemBalanceListEBM ▫ UpdateItemBalanceListResponseEBM
EBSs	▫ ItemBalanceEBS

Core Components Locations

EBO & EBM XSD files	http://[HOST:PORT]/AIAComponents/EnterpriseObjectLibrary/Core/EBO/
WSDL files	http://[HOST:PORT]/AIAComponents/EnterpriseBusinessServiceLibrary/Core/EBO/

- For detailed documentation of individual EBOs, click the EBO Name link on the *Integration Scenario Summary* page in the *Oracle AIA Console*. You can also use the *Integration Scenario Summary* page to search for and view integration scenarios that utilize a particular EBO or EBS.

For more information, see *Oracle Application Integration Architecture - Foundation Pack: Core Infrastructure Components Guide*, “Using the BSR,” Using the BSR UI to View Integration Scenarios.

- EBOs can be extended, for instance, to add new data elements. These extensions are protected, and will remain intact after a patch or an upgrade.

For more information, see *Oracle Application Integration Architecture – Foundation Pack: Integration Developer’s Guide*, “Extensibility for AIA Artifacts.”

Oracle EBS & Agile Components for IB Update

Services	Oracle EBS (Requester)	Agile (Provider)
ABMs	---	▫ AgileUpdateItemBalanceListABM ▫ AgileUpdateItemBalanceListResponseABM
ABCS	▫ UpdateItemBalanceListEbizReqABCS	▫ UpdateItemBalanceListAgileProvABCS

EBS	---	▫ ItemBalanceEBS
BPEL	---	---
ESB Service	▫ ItemBalanceService	---

Component Locations

ABO, ABM & Common XSD files	http://<servername>:<portname>/AIAComponents/ApplicationObjectLibrary/Agile/V1/sc hemas
WSDL files	http://<servername>:<portname>/AIAComponents/ApplicationObjectLibrary/Agile/V1/ws dls

Integration Services for IB Update

1. ItemBalanceEBS

ItemBalanceEBS is the Enterprise Business Services which exposes the following operations related to the Item Availability Attributes Update Integration on the *ItemBalanceEBO*.

The following are the routing rules:

ItemBalanceEBS ESB service

1. UpdateItemBalanceListEbizReqABCImpl
 - Route *UpdateItemBalanceListEBM* to *UpdateItemBalanceListAgileProvABCImpl*

ItemBalanceResponseEBS ESB service

1. UpdateItemBalanceListAgileProvABCImpl
 - Route *UpdateItemBalanceListResponseEBM* to *UpdateItemBalanceListEbizReqABCImpl*

2. UpdateItemBalanceListEbizReqABCImpl

This is called when there is a need for a participating application to update multiple item balance attributes on their system such as on-hand quantity from Oracle EBS. This is a push from Oracle EBS to the other participating application.

- The requester ABCS defined as a “synchronous” process, receives a list of Item IDs from the Oracle EBS concurrent program. The list of IDs are those which have "last_update_date" greater the "last run date" of the concurrent program.
- The requester BPEL process filters the list of IDs to a list of IDs that are present in the xref tables in the FMW layer. This provides a list of Items that were actually from the Agile system alone. The original list of Item IDs may be those which are from non-Agile sources as well.
- The BPEL process then makes an ESB call out for end point virtualization. This ESB layer has an OA Adapter that calls the PL/SQL API that provides the Item Balance details that is needed

to be sent out. This is the ItemBalance ABM.

- A transformation converts the ABM to an EBM.
- An asynchronous request-delayed response call is made to the *ItemBalanceEBS* with the *ItemBalanceListEBM*. This routes to the appropriate provider.
- The BPEL instance gets invoked when the asynchronous call gets back from the provider and provides the status of the transaction back to the caller concurrent program (since that is a synchronous call). This does not impact the performance since it is a scheduled call and is invoked from backend and not a UI invoke.
- The concurrent program logs the status of this call.

3. UpdateItemBalanceListAgileProvABCSEImpl

UpdateItemBalanceListAgileProvABCSEImpl is used to facilitate the communication between *ItemBalanceEBS* and Agile web service used for updating the Items' On-Hand Quantity information in batch mode in Agile.

- Receives *UpdateItemBalanceListReqMsg*, which contains *UpdateItemBalanceListEBM*
- Transform operation is called to convert the *UpdateItemBalanceListEBM* into *AgileUpdateItemBalanceListABM*.
- *AgileUpdateItemBalanceListABM* is passed as input to the web service operation *UpdateItems* (Coarse Grained APIs on Agile side) to update Items in Agile.
- *AgileUpdateItemBalanceListResponseABM* is received on successful execution of Coarse Grained API.
- Transforms *AgileUpdateItemBalanceListResponseABM* to *UpdateItemBalanceListResponseEBM*, which is returned as output of this BEPL process.
- If the *UpdateItems* service operation fails on the Agile side, a new Fault is generated and will be sent across with appropriate error message.

IB Update Integration Customization Points

Oracle E-Business Suite

UpdateItemBalanceListEbizReqABCSEImpl	ItemBalanceABMtoEBM_Custom.xml	Custom transformations for Item Balance Request ABM to Request EBM
---------------------------------------	--------------------------------	--

Agile

UpdateItemBalanceListAgileProvABCSEImpl (Agile update item balance provider flow)	AgileUpdateItemBalanceListResponseABM_to_UpdateItemBalanceListResponseEBM_Custom.xml	RespABM to RespEBM (custom element)
	AgileUpdateItemBalanceListResponseABM_to_UpdateItemBalanceListResponseEBM_Impl.xml	RespABM to RespEBM (main)
	UpdateItemBalanceListEBM_to_AgileUpdateItemBalanceListABM_Impl.xml	ReqEBM to ReqABM (main)

Implementing the Process Integration Pack

This chapter includes the following:

▪ Setting up the Participating Applications	65
▪ Loading Cross Reference Data	70
▪ Configuring the PIP.....	74
▪ Domain Value Maps	81
▪ Application Interfaces	88
▪ Handling Errors.....	90

Note For complete information on PIP, refer Oracle AIA Installation Guide.

Deployment of Agile PLM PIP is carried out by AIA Installer. However, for conducive functioning, this requires certain settings and configurations in Partner Applications, viz., Agile PLM and Oracle E-Business Suite, and in AIA Configuration Properties files.

The following applications should to be existing before the PIP is installed:

- SOA 10.1.3.4 should be installed and relevant patches are to be applied.
- AIA Foundation Pack P2.3 should be installed with relevant patches applied, if any.
- Agile PLM must be installed and configured accordingly.
- Oracle E-Business Suite environment must be installed and configured accordingly. The aru# 7563732 patch, which is specific to PIP, should be applied over the Oracle E-Business Suite 11.5.10 CU2 installation.

The following sections describe post-installation configurations and settings that are necessary for the processes to function cohesively.

Setting up the Participating Applications

Agile PLM

After installation of Agile PIP, the Agile Administrator is required to setup Agile Content Services. The following are the settings/configurations:

1. Create New Destinations
2. Create New Events for ECO, MCO and SCO
3. Define Filters
4. Create New Subscribers for ECO, MCO, SCO

5. Set Privileges
6. Modify Flex Fields

These settings are done in Agile *Java Client*.

1. Create New JMS Destination

1. In Admin Tab, go to System Settings > Agile Content Service > Destinations
2. Select Protocol JMS
3. Enter/set the following

Field	Value/Setting
Name	Define your own
User Name	oc4jadmin (SOA server admin user name)
Password	welcome1
Connection Factory	jms/QueueConnectionFactory (default)
Default Provider URL	opmn:ormi://<SOAServer>:<OPMNPort>:<instance-name>
Destination Name	jms/ECOQueue

4. Click Test to validate

2. Create New Events for ECO, MCO and SCO

1. In Admin Tab, go to System Settings > Agile Content Service > Events
2. Enter/set the following

Field	Value/Setting		
	ECO	MCO	SCO
Name	Define your own		
Event Type	Workflow		
Workflow	Default Change Order	Default Manufacturer Orders	Default Site Change Orders
Workflow Status	Released		

3. Define Filters

1. In Admin Tab, go to System Settings > Agile Content Service > Filters
2. Modify Default Item Filter to set the following:

Field	Value/Setting
View Tabs	Add Sites
BOM Options	Tabs and Items

BOM Levels	Select All Levels checkbox
AML Options	Tabs and Manufacturer Parts
Attachment Options	Tab only

3. Modify the following filters to set the given fields:

Field	Value/Setting		
	Default Change Order Filter	Default Manufacturer Order Filter	Default Site Change Order Filter
Affected Items Options	Tab & Items	Tabs & Items	Tabs & Items
Redline Changes only	No	No	No

4. Create New Subscribers for ECO, MCO and SCO

- In Admin Tab, go to System Settings > Agile Content Service > Subscribers
- Create new Subscribers, one each for ECO, MCO and SCO and set the following:

Field	Value/Setting		
	For ECO	For MCO	For SCO
Name	Define your own		
Subclass	ATO		
Workflow	Default ATOs		
Criteria	All Change Orders	All Manufacturer Orders	All Site Change Orders
Event	Select the Name of the Event that you created #2 for ECO	Select the Name of the Event that you created #2 for MCO	Select the Name of the Event that you created #2 for SCO

3. Enter/set the Subscriber Details for each (ECO, MCO, SCO), by adding a new row, as follows

Field	Value/Setting		
	For ECO	For MCO	For SCO
Destinations	Select the JMS Destinations that you created for each		
Filter	Default Change Order Filter	Default Manufacturer Change Order Filter	Default Site Change Order Filter
	Default Item Filter	Default Item Filter	Default Item Filter
Roles	All		
Format	aXML		
Language	English		
Site	All		

4. Enable all the newly created Subscribers

5. Set Privileges

1. In Admin Tab, go to User Settings > Privileges > Modify.
2. Create new Modify Privileges for ECO, MCO and SCO.
3. Set Privilege to Modify.
4. Select the Criteria that correspond to each of the ECO, MCO and SCO.
5. Select all the Attributes, including the invisible/disabled attributes in the Applied to field and Save them.
6. In Where Used Tab, add Roles to all the created privileges. Default role is Admin user.
7. Create new Read privileges for MCO.
8. Select All Manufacturer Orders in the Criteria.
9. Choose both visible and invisible/disabled attributes in the Applied to field.
10. Modify the Read Changes and Read Items to get the Admin user in the Where Used Tab.

6. Modify Flex Fields

1. In Admin Tab, go to Data Settings > Classes.
2. For both Parts and Document classes, enable the flexfields on Page2, Page3 or Site tab according to the MultiSite_Enabled property value in AIAConfigProperties.xml for Agile module.
3. These field names in Agile reflect the following fields from the Oracle EBS:
 - Manufacturer Cost
 - Available Quantity
 - On Hand Quantity
 - Reserved Quantity
4. These fields should be the same as those entered as values for the following properties:
 - Item.UnitCostAttribute
 - Item.AvailableQuantityAttribute
 - Item.OnHandQuantityAttribute
 - Item.ReservedQuantityAttribute
5. Make the created fields as Visible

Note	Make sure that these attributes have Read and Modify privileges.
------	--

6. Similarly, enable a Page2 or Page3 flex field on the ECO, MCO, SCO to reflect Change.TransferStatusAttribute property value in AIAconfigProperties file.

Oracle E-Business Suite

The flow of data from Oracle E-Business Suite to Agile requires implementation of Concurrent Programs. These are configured as a Periodic/Scheduled publication or triggered ad-hoc for On Demand publication.

First Scheduled run:

When the concurrent program is scheduled, it sends back all those entities which have been updated since the last concurrent request has run. But for the very first run, this "Last run date" is not available. Hence it is defaulted to last 30 days. But this may result in huge number of items being picked up. Hence, the customer may choose to first perform an ad-hoc publish (this is after the Agile-Oracle EBS PIP is installed and a few items are transferred from Agile to Oracle EBS based on the implementation).

Also specific Orgs could be specified along with from and to date in the ad-hoc request parameters, from performance perspective to send the data in multiple requests. After this, the customer can set up a scheduled process and the first run of the scheduled process will pick up the adhoc request run time as the Last run date.

Another option is that the customer can just schedule the request to be run with the "Updated within the last x hours" parameter set. This will, even for the first run, pick up items which have been updated in the last x hours.

Subsequent Scheduled run:

It is recommended to set the 'Updated in the last X hrs' parameter with a reasonable value by default for the Concurrent Programs that are being setup to run at a schedule frequency. This parameter value should be specified apart from the schedule frequency setup in the Concurrent Program setup. We suggest Customers set the same 'X' hrs for the Concurrent Program Schedule setup. (OR)

Customers can leave all the parameters empty and schedule the concurrent program to run at a particular schedule frequency that he desires.

Profile Settings

The following settings have to be configured for EBS to support integration flow from Oracle EBS to Agile.

Set the following profile options in Oracle EBS:

EBS Integration Proxy Server Host	set it to the <SOA server /host name/>
EBS Integration Proxy Server Port	set it to the <SOA server http /port number/>
EBS Integration Server Domain	set it to </default/>
EBS Integration Server Host: Port	set it as </http://host:port/> for the SOA server

Note If the ECO update is event based, then setup for Subscribers and Events has to be done.

For navigating to profiles UI, you need to login as /sysadmin/sysadmin/, go to /System Administrator/responsibility, and then click on /System /link under /Profile /menu. Thereafter search for these profiles and set the appropriate values.

Loading Cross Reference Data

This PIP uses the xref_data table present in AIA schema to maintain a cross-reference between Agile and Oracle EBS. This cross-reference information helps map Agile Parts/ Documents/ Change Orders to Oracle EBS Items/ Change Orders.

There are two main virtual tables in the aia.xref_data table that maintain this cross-reference information.

- CHANGE_CHANGEID - maintains all the Change Order information
- ITEM_ITEMID - maintains Item information.

For example

- A Change Order ECO001 contains two revised items
 - P0001 in site V1
 - P0002 in site V2
- Both these revised items have component items.
 - P0001 has component items C0001 and C0002
 - P0002 has component items C0003 and C0004
- When this Change Order is released from Agile to the Oracle E-Business Suite, the following entries are made in the CHANGE_CHANGEID virtual table.

XREF_TABLE	XREF_COLUMN	ROW_NUMBER	VALUE
CHANGE_CHANGEID	AGILE9224_01	E45E015046AF11DD9F2E436FB39961A8	ECO001::V1
CHANGE_CHANGEID	COMMON	E45E015046AF11DD9F2E436FB39961A8	2d373833303237303132383837353631
CHANGE_CHANGEID	EBIZ_01	E45E015046AF11DD9F2E436FB39961A8	11075
CHANGE_CHANGEID	AGILE9224_01	E45E015046AF11DD9F2E436FB39961A9	ECO001::V2
CHANGE_CHANGEID	COMMON	E45E015046AF11DD9F2E436FB39961A9	2d373833303237303132383837353632
CHANGE_CHANGEID	EBIZ_01	E45E015046AF11DD9F2E436FB39961A9	11076

- The first row entry is made by the Agile BPEL flow for Change Order number ECO001 that is created on an item, which belongs to a site mapped to the V1 organization in Oracle E-Business Suite.
 - The second entry (Common) is also created by the Agile BPEL flow. It indicates the common "business component id" for this particular in integration entity and is used for linking change orders to Oracle EBS Change Orders.
 - The third row entry represents the Oracle EBS change ID corresponding to the Agile change order ECO001. Another set of entries will be made for the change order ECO0001 for site V2.
- For each revised item and component items in the Change Order, entries will be made into the ITEM_ITEMID virtual table.

For example

For ECO001 the following entries will be made in the ITEM_ITEMID table.

XREF_TABLE	XREF_COLUMN	ROW_NUMBER	VALUE
ITEM_ITEMID	AGILE9224_01	0078BE703EC711DDBF9CA7AA7FE3BDF B	P0001::V1
ITEM_ITEMID	COMMON	0078BE703EC711DDBF9CA7AA7FE3BDF B	35313835373739353732383638303435
ITEM_ITEMID	EBIZ_01	0078BE703EC711DDBF9CA7AA7FE3BDF B	66247::204::<operating unit ID>
ITEM_ITEMID	AGILE9224_01	0078BE703EC711DDBF9CA7AA7FE3BDF C	P0002::V2
ITEM_ITEMID	COMMON	0078BE703EC711DDBF9CA7AA7FE3BDF C	35313835373739353732383638303436
ITEM_ITEMID	EBIZ_01	0078BE703EC711DDBF9CA7AA7FE3BDF C	66248::207::<operating unit ID>
ITEM_ITEMID	AGILE9224_01	0078BE703EC711DDBF9CA7AA7FE3BDF D	C0001::V1
ITEM_ITEMID	COMMON	0078BE703EC711DDBF9CA7AA7FE3BDF D	35313835373739353732383638303437
ITEM_ITEMID	EBIZ_01	0078BE703EC711DDBF9CA7AA7FE3BDF D	66249::204::<operating unit ID>
ITEM_ITEMID	AGILE9224_01	0078BE703EC711DDBF9CA7AA7FE3BDF E	C0002::V1
ITEM_ITEMID	COMMON	0078BE703EC711DDBF9CA7AA7FE3BDF E	35313835373739353732383638303438
ITEM_ITEMID	EBIZ_01	0078BE703EC711DDBF9CA7AA7FE3BDF E	66250::204::<operating unit ID>
ITEM_ITEMID	AGILE9224_01	0078BE703EC711DDBF9CA7AA7FE3BDF F	C0003::V2
ITEM_ITEMID	COMMON	0078BE703EC711DDBF9CA7AA7FE3BDF F	35313835373739353732383638303439
ITEM_ITEMID	EBIZ_01	0078BE703EC711DDBF9CA7AA7FE3BDF F	66251::207::<operating unit ID>
ITEM_ITEMID	AGILE9224_01	0078BE703EC711DDBF9CA7AA7FE3BDF G	C0004::V2
ITEM_ITEMID	COMMON	0078BE703EC711DDBF9CA7AA7FE3BDF G	35313835373739353732383638303440
ITEM_ITEMID	EBIZ_01	0078BE703EC711DDBF9CA7AA7FE3BDF G	66252::207::<operating unit ID>

- The first row entry is made by the Agile BPEL flow for part P0001. This part belongs to an Agile site that is mapped to site V1.
 - The second entry (Common) is also created by the Agile BPEL flow. It indicates the common "business component ID" for this particular in integration entity and is used to link Agile Parts/ Documents/ Change orders to Oracle EBS Items/Change Orders.
 - The third row entry represents the Oracle EBS inventory item ID corresponding to the item P1B and the organization ID for the item. The rest of the entries represent the revised and component items for the ECO001.
- The ITEM_ITEMID virtual table contains the similar xref entries for each item/part created through the new part request process.

Creating Cross Reference Data

If data needs to be ported from other existing integrations to the Agile PLM PIP, first the relationship between Agile entities and Oracle EBS entities should be established and each Agile site should be mapped to corresponding Oracle EBS organization by inserting the relevant data in the xref_data table.

The ESB out of the box utility xrefimport.sh can be used to load the required integration data in the xref_data table to establish the link between Agile and Oracle EBS. This data will map the Agile entities to the Oracle EBS entities as described in the section above.

Even for existing Agile PLM PIP integrations, if an item or change order needs to be created in Oracle EBS and then is created in Agile, for the PIP to process the item XREF entry for that item needs to be made. All the update (reverse) flows for the PIP will update the item attributes in Agile only if there is an entry in the xref_data table for that particular item.

xrefimport.sh Usage

The xrefimport.sh is present under \$ORACLE_HOME/integration/esb/bin/. Before you execute xrefimport.sh, set the following environment variables

- OC4J_USERNAME
- OC4J_PASSWORD
- DB_URL
- DB_USER
- DB_PASSWORD.

The values should point to the schema where xref table is created.

Sample values are:

```
export DB_URL="jdbc:oracle:thin:@//<database server name>:< database port>/<sid>"
export DB_USER=aia
export DB_PASSWORD=aia
export OC4J_USERNAME=oc4jadmin
export OC4J_PASSWORD=welcome1
```

Then the data that has to be inserted in the xref_data table should be put in an XML file with the following format:

Sample Change Order Data File

```
<xref xmlns="http://xmlns.oracle.com/xref">
  <table name="CHANGE CHANGEID">
    <columns>
      <column name="EBIZ_01"/>
      <column name="COMMON"/>
      <column name="AGILE9224 01"/>
    </columns>
    <rows>
      <row>
        <cell colName="EBIZ 01">11075</cell>
```

```

        <cell colName="COMMON">2d373833303237303132383837353631</cell>
        <cell colName="AGILE9224 01">ECO001::V1</cell>
    </row>
    <row>
        <cell colName="EBIZ 01">11076</cell>
        <cell colName="COMMON">2d373833303237303132383837353632</cell>
        <cell colName="AGILE9224 01"> ECO001::V2</cell>
    </row>
</rows>
</table>
</xref>

```

- The change id number that is inserted in the EBIZ_01 column for a particular change order can be obtained using the following query:


```

select change_id from eng_engineering_changes
where
change_notice=<AgileChangeOrder>

```
- The business component id that is inserted in the COMMON column can be any unique number.
- The Agile change order number and the Oracle E-business Suite organization corresponding to the Agile site are inserted into the AGILE9224_01 column separated by "::".

Sample Item Data File

```

<xref xmlns="http://xmlns.oracle.com/xref">
  <table name="ITEM ITEMID">
    <columns>
      <column name="EBIZ 01"/>
      <column name="COMMON"/>
      <column name="AGILE9224 01"/>
    </columns>
    <rows>
      <row>
        <cell colName="EBIZ 01">39854::204::204</cell>
        <cell colName="COMMON">35313835373739353732383638303435</cell>
        <cell colName="AGILE9224 01">P0001::V1</cell>
      </row>
    </rows>
  </table>
</xref>

```

- The Item inventory number, the organization id and the operating unit ID are inserted in the EBIZ_01 column separated by "::"
- The inventory item id for a particular item can be obtained using the following query:


```

select inventory_item_id from mtl_system_items_b
where
segment1='<AgileItem>'

```
- The organization id for the given item can be obtained using the query -


```

select organization_id from mtl_parameters
where
organization_code=<EbizOrgMappedtotheItemAgileSite>

```

- The business component id that is inserted in the COMMON column can be any unique number.
- Agile item number and Oracle EBS organization corresponding to the Agile site are inserted into the AGILE9224_01 column separated by "::".
- The command to execute the xrefimport.sh to load the data in the xref_data table is.
\$ORACLE_HOME/integration/esb/bin/xrefimport.sh -file <xmldatafilename>.
The result of the import are logged in \$ORACLE_HOME/integration/esb/bin/xrefimport.log

Configuring the PIP

This PIP uses various configuration parameters that control the behavior of the flow. Standard AIA XML configuration file, AIAConfigurationProperties.xml, is used for capturing configuration parameters. AIA configuration file supports system level configuration parameters, service level parameters and module configuration parameters. System level parameters apply to all PIPs running on the SOA suite. Service level parameters can be configured at the individual service level such as ABCS.

Note	The configuration properties from Agile Module and Oracle E-Business Suite Module are listed separately in this section, only for the purpose of identification. The actual AIAConfigurationProperties.xml file on AIA Server is a merger of both.
------	--

Configuration Parameters

This PIP uses the following type of configuration parameters -

- PIP Level configuration parameters: PIP Configuration parameters are implemented using AIA module configuration entry. The module configuration entry has a name and can contain any number of configuration parameters. A naming convention of PIPS.PIPName is used for naming modules. The parameters inside the module are named using with cascaded naming convention where individual words are separated with dots.

For example, *agile.replicate.item*.
- Service Level configuration parameters: While most configuration requirements are satisfied by the PIP Level configuration parameters, sometimes the behavior of a flow needs to be controlled at the service level. These parameters can be captured using AIA service configuration parameters. Service configuration entry is identified by the service name such as *CreateItemAgileReqABCImpl*. The parameter names themselves are named using cascaded naming convention as explained above.

Note	Whenever the AIAConfigurationProperties.xml file is updated, the file must be reloaded for updates to be reflected in the applications or services that use the updated properties. You can perform this reload by clicking the Reload button on the Configuration page in the Oracle AIA Console. Alternatively, you can perform the reload by rebooting the server.
------	---

Note	See Oracle AIA Core Components Guide, "Working with the BSR," Loading Oracle AIA Configuration Properties File Updates.
------	---

Agile Configurations

Properties	(default) Value/Setting	Description
moduleName	Agile	
MULTISITE_ENABLED	TRUE	When set to True, the sites specified in Sites Tab of Items are used to determine the Orgs in eBiz to which they are mapped. When set to False, Page2 Multilist01 attribute is used to determine the Orgs in eBiz to which the Item will be extended to.
Item.UnitCostAttribute	Site.Numeric01	Determines the attribute to which the unit cost from eBiz would be updated in Agile.
Item.AvailableQuantityAttribute	Site.Numeric02	Determines the attribute to which the available quantity from eBiz would be updated in Agile.
Item.OnHandQuantityAttribute	Site.Numeric03	Determines the attribute to which the on-hand quantity from eBiz would be updated in Agile.
Item.ReservedQuantityAttribute	Site.Numeric04	Determines the attribute to which the reserved quantity from eBiz would be updated in Agile.
Change.TransferStatusAttribute	PageTwo.Text02	Determines the attribute to which the transfer status of a Change. When the Change flow is from Agile to eBiz, the possible values are "Transferred" or "Errored". When the Change flow is from eBiz to Agile, the value would be the same as that of the Status of the Change in all the Orgs of eBiz.
REPLICATE_BOM_ENABLED	FALSE	Used for sample replicate BOM customization. Refer Readme in Samples folder
COMMON_BOM_ENABLED	FALSE	Used for sample common BOM customization. Refer Readme in Samples folder

Notes:

Multisite_Enabled property is governed by [Distributed Processing](#) "Release of Change Order in Agile" on page 14 aspects covered in [Chapter 2: Process Integration for ECO/PREL](#) "Process Integration for Change Order Release" on page 13.

When it is set to TRUE (default), the Item.UnitCostAttribute, Item.AvailableQuantityAttribute, Item.OnHandQuantityAttribute and Item.ReservedQuantityAttribute are set to Site Tab Flex Attributes. You can use Numeric, Text or Money flex fields of Site tab for these settings, and is denoted by the first element, *Site*. For example, Site.Numeric01.

When it is set to FALSE, all these attributes will be set to Page2 or Page3 flex fields. Hence, the settings will need to be changed to PageTwo.Numeric01 or PageThree.Numeric01 accordingly.

The names of the attributes can be derived from the ItemABM Schema, which can be found in [Agile Application Interfaces](#) "Agile PLM Interfaces" on page 88.

Properties	(default) Value/Setting	Description
serviceName	CreateQueueService	
TRACE.LOG.ENABLED	false	Use tracelog for the flow
serviceName	QueueProcessorServiceImpl	
TRACE.LOG.ENABLED	false	Use tracelog for the flow
serviceName	UpdateEngineeringChangeOrderListAgileProvABCServiceImpl	
ABCSExtension.PreProcessABM	FALSE	User exit for the pre-process ABM should be called or not
ABCSExtension.PreProcessEBM	FALSE	User exit for the pre-process EBM should be called or not
ABCSExtension.PostProcessEBM	FALSE	User exit for the post-process EBM should be called or not
ABCSExtension.PostProcessABM	FALSE	User exit for the post-process ABM should be

Properties	(default) Value/Setting	Description
		called or not
TRACE.LOG.ENABLED	false	Use tracelog for the flow
ROUTE_TO_CAVS	false	Route to CAVS, if set as True
DEFAULT_TARGET_ENDPOINT_URI	http://<host>:<port>/event/CoreAgile/EngineeringChangeOrderResponseEBS	EngineeringChangeOrderResponseEBS runtime target endpoint URI
serviceName	UpdateItemBalanceListAgileProvABCSImpl	
ABCSExtension.PreProcessABM	FALSE	User exit for the pre-process ABM should be called or not
ABCSExtension.PreProcessEBM	FALSE	User exit for the pre-process EBM should be called or not
ABCSExtension.PostProcessEBM	FALSE	User exit for the post-process EBM should be called or not
ABCSExtension.PostProcessABM	FALSE	User exit for the post-process ABM should be called or not
TRACE.LOG.ENABLED	FALSE	Use tracelog for the flow
ROUTE_TO_CAVS	FALSE	Route to CAVS, if set as True
DEFAULT_TARGET_ENDPOINT_URI	http://<host>:<port>/event/CoreAgile/ItemBalanceResponseEBS	ItemBalanceResponseEBS runtime target endpoint URI
serviceName	UpdateItemListAgileProvABCSImpl	
ABCSExtension.PreProcessABM	FALSE	User exit for the pre-process ABM should be called or not
ABCSExtension.PreProcessEBM	FALSE	User exit for the pre-process EBM should be called or not
ABCSExtension.PostProcessEBM	FALSE	User exit for the post-process EBM should be called or not
ABCSExtension.PostProcessABM	FALSE	User exit for the post-process ABM should be called or not
TRACE.LOG.ENABLED	FALSE	Use tracelog for the flow
ROUTE_TO_CAVS	FALSE	Route to CAVS, if set as True
DEFAULT_TARGET_ENDPOINT_URI	http://localhost:8888/event/CoreAgile/ItemResponseEBS	ItemResponseEBS runtime target endpoint URI
serviceName	ProcessEngineeringChangeOrderAgileReqABCS	
ABCSExtension.PreProcessABM	FALSE	User exit for the pre-process ABM should be called or not
ABCSExtension.PreProcessEBM	FALSE	User exit for the pre-process EBM should be called or not
ABCSExtension.PostProcessEBM	FALSE	User exit for the post-process EBM should be called or not
ABCSExtension.PostProcessABM	FALSE	User exit for the post-process ABM should be called or not
TRACE.LOG.ENABLED	FALSE	Use tracelog for the flow
ROUTE_TO_CAVS	FALSE	Route to CAVS, if set as True
DEFAULT_TARGET_ENDPOINT_URI	http://<host>:<port>/event/CoreAgile/EngineeringChangeOrderEBS	EngineeringChangeOrderEBS runtime target endpoint URI

Oracle EBS Configurations

Properties	(default)	Description
------------	-----------	-------------

ModuleName	Value/Setting Ebiz	
FIXED_ORG_FOR_INBOUND_DATA_TRANSFER		If set, Ebiz to Agile flows will publish data only for this particular organization code
IS_PIM_INSTALLED	FALSE	True - PIM installed, false - non-PIM scenario. It should be kept to false for this release with ebiz 11i.
DEFAULT_MASTER_ORG		Default organization when the incoming one is empty for New Part Request
USER	sysadmin	User for setting FND Apps Context
RESPONSIBILITY	System Administrator	Responsibility for setting FND Apps Context

Properties	(default) Value/Setting	Description
ServiceName	CreateItemEbizProvABCSImpl	
ROUTING.ITEMSERVICE.ROUTE_TO_CAVS	FALSE	Use CAVS for ItemServiceESB
ROUTING.ITEMRESPONSEEBS.ROUTE_TO_CAVS	FALSE	Use CAVS for ItemResponseEBS
ROUTING.GIN.SDO.ROUTE_TO_CAVS	FALSE	Use CAVS for GenerateItemNumberService
GIN.EBIZ_01.SDO.DEFAULT.TARGET.ENDPOINT.URI		GenerateItemNumberService run time target endpoint URI
ROUTING.GIN.USEREXIT.ROUTE_TO_CAVS	FALSE	Use CAVS for GenerateItemNumberDescriptionUserExit
GIN.USEREXIT.DEFAULT.TARGET.ENDPOINT.URI		GenerateItemNumberDescriptionUserExit runtime target endpoint URI
CUSTOM.TRANSFORMATIONS.EBM_TO_ABM	FALSE	Use Custom transformation for EBM to ABM
CUSTOM.TRANSFORMATIONS.ABM_TO_EBM	FALSE	Use Custom transformation for ABM to EBM
ABCSEXTENSION.PREPROCESSABM	FALSE	User exit for the pre-process ABM should be called or not
ABCSEXTENSION.POSTPROCESSABM	FALSE	User exit for the post-process ABM should be called or not
ABCSEXTENSION.PREPROCESSEBM	FALSE	User exit for the pre-process EBM should be called or not
ABCSEXTENSION.POSTPROCESSEBM	FALSE	User exit for the post-process EBM should be called or not
ITEMSERVICE.EBIZ_01.DEFAULT.TARGET.ENDPOINT.URI		ItemServiceESB runtime target end point URI if multiple Ebiz Instances are involved
ITEMRESPONSEEBS.DEFAULT.TARGET.ENDPOINT.URI		ItemResponseEBS runtime target endpoint URI
TRACE.LOG.ENABLED	FALSE	Use tracelog for the flow
ServiceName	CreateEngineeringChangeOrderListEbizProvABCSImpl	
Default.SystemID	EBIZ_01	Used to get the default XREF Target column name when TargetId is empty in incoming EBM
ROUTING.ECOSERVICE.ROUTETOCAVS	FALSE	Use CAVS for ECOService
ROUTING.ENGINEERINGCHANGEORDERRESPONSEEBS.ROUTETOCAVS	FALSE	Use CAVS for EngineeringChangeOrderResponseEBS
ROUTING.ECOSERVICE.EBIZ_01.ENDPOINTURI		ECOService run time target endpoint URI
ROUTING.ECOSERVICE.CAVS.ENDPOINTURI		ECOService endpoint URI when set to CAVS
ROUTING.ENGINEERINGCHANGEORDERRESPONSEEBS.CAVS.ENDPOINTURI		EngineeringChangeOrderResponseEBS endpoint URI when set to CAVS
ABCSEXTENSION.PREPROCESSABM	FALSE	User exit for the pre-process ABM should be called or not
ABCSEXTENSION.POSTPROCESSABM	FALSE	User exit for the post-process ABM should be called or not
ABCSEXTENSION.PREPROCESSEBM	FALSE	User exit for the pre-process EBM should be called

Properties	(default) Value/Setting	Description
		or not
ABCSEXTENSION.POSTPROCESSEBM	FALSE	User exit for the post-process EBM should be called or not
CUSTOM.TRANSFORMATIONS.EBM_TO_ABM	FALSE	Use Custom transformation for EBM to ABM
CUSTOM.TRANSFORMATIONS.ABM_TO_EBM	FALSE	Use Custom transformation for ABM to EBM
ASSIGN_ITEM_TO_CHILD_ORG	T	Indicates if Components,SubstituteComponentst coming in does not exist in context org,should be assigned from master org or not
TEMPLATE_FOR_ITEM_UPDATE_ALLOWED	F	During Item Updation if this config value is 'T' then item template can be updated ,if is 'F' item template cannot be changed
VALIDATE_REVISD_ITEM_REVISION	F	If this Config value is set to T then in MCO/SCO case we will ensure that the incoming revision is either Current or Future implemented revision
DEFAULT_STRUCTURE_TYPE	EBOM	This parameter is only for R12 case.If incoming payload does not have any value for structure type then the value specified for this config parameter will be defaulted.
CREATE_ERP_CHANGE_ORDER	T	Indicates if an ERP change order should be created or a PLM change order. 'T' ---> ERP, 'F' ---> PLM
REPLICATE_BOM_VIEW_SCOPE	All	Indicates the BOM view scope for replicate BOM case. 'ALL' -- All the BOMs should be considered in Replicating BOM from source org to Destination org. 'CURRENT' -- BOM which is effective currently should be considered. 'CURRENT_AND_FUTURE' -- BOM which is effective currently and also those with future effective dates should be considered.
REPLICATE_BOM_IMPLEMENTATION_SCOPE	All	Indicates BOM implementation scope for replicate BOM case. 'IMPLEMENTED' -- Implemented BOMs in source org will be considered in Replicating BOM from source org to Destination org. 'UNIMPLEMENTED' -- Unimplemented BOMs will only be considered
TRACE.LOG.ENABLED	FALSE	Use tracelog for the flow
ServiceName	UpdateItemListEbizReq	ABCImpl
Default.SystemID	EBIZ_01	Used to get the default XREF Target column name when TargetId is empty in incoming EBM
Routing.GetItemAttrListService.GetItemService.RouteToCAVS	FALSE	Use CAVS for getItemAttrListService
Routing.ItemEBS.UpdateItemListEBM.RouteToCAVS	FALSE	Use CAVS for ItemEBS
Routing.GetItemAttrListService.EBIZ_01.EndpointURI		getItemAttrListService run time target endpoint URI
Routing.ItemEBS.EBIZ_01.EndpointURI		ItemEBS run time target endpoint URI
Routing.GetItemAttrListService.GetItemService.CAVS.EndpointURI		getItemAttrListService endpoint URI when set to CAVS
Routing.ItemEBS.UpdateItemListEBM.CAVS.EndpointURI		ItemEBS endpoint URI when set to CAVS
CUSTOM_TRANSFORMATIONS	FALSE	Used to determine whether custom transformations should be used or not
ASSET_ATTRS	T	When the property is set to true all Asset attributes will be published in output EBM
BOM_ATTRS	T	When the property is set to true all BOM attributes will be published in output EBM
COSTING_ATTRS	T	When the property is set to true all Costing attributes will be published in output EBM
GPLAN_ATTRS	T	When the property is set to true all Planning

Properties	(default) Value/Setting	Description
		attributes will be published in output EBM
INVENTORY_ATTRS	T	When the property is set to true all Inventory attributes will be published in output EBM
INVOICE_ATTRS	T	When the property is set to true all Invoice attributes will be published in output EBM
LEAD_TIME_ATTRS	T	When the property is set to true all Lead Times attributes will be published in output EBM
MPSMRP_ATTRS	T	When the property is set to true all MPS/MRP Planning attributes will be published in output EBM
ORDER_ATTRS	T	When the property is set to true all Order attributes will be published in output EBM
PHYSICAL_ATTRS	T	When the property is set to true all Physical attributes will be published in output EBM
PROCESS_ATTRS	T	When the property is set to true all Process attributes will be published in output EBM
PURCHASING_ATTRS	T	When the property is set to true all Purchasing attributes will be published in output EBM
RECEIVING_ATTRS	T	When the property is set to true all Receiving attributes will be published in output EBM
SERVICE_ATTRS	T	When the property is set to true all Service attributes will be published in output EBM
WEB_OPTION_ATTRS	T	When the property is set to true all Web Option attributes will be published in output EBM
WIP_ATTRS	T	When the property is set to true all Work In progress attributes will be published in output EBM
ITEM_ATTRS	T	When the property is set to true all Item attributes will be published in output EBM
DEFAULT_ITEM_COST_GROUP		If this value is given then cost will be derived based on this value. Note: Either DEFAULT_ITEM_COST_TYPE or DEFAULT_ITEM_COST_GROUP should be set but not both
DEFAULT_ITEM_COST_TYPE	1	If this value is given the cost will be derived based on this value. Note: Either DEFAULT_ITEM_COST_TYPE or DEFAULT_ITEM_COST_GROUP should be set but not both
TRACE.LOG.ENABLED	FALSE	Use tracelog for the flow
ABCSEXTENSION.PREPROCESSABM	FALSE	User exit for the pre-process ABM should be called or not
ABCSEXTENSION.POSTPROCESSABM	FALSE	User exit for the post-process ABM should be called or not
ABCSEXTENSION.PREPROCESSEBM	FALSE	User exit for the pre-process EBM should be called or not
ABCSEXTENSION.POSTPROCESSEBM	FALSE	User exit for the post-process EBM should be called or not
ServiceName	UpdateEngineeringChangeOrderEbizReqABCImpl	
Default.SystemID	EBIZ_01	Used to get the default XREF Target column name when TargetId is empty in incoming EBM
Routing.GetUpdateEngineeringChangeOrderListService.UpdateEngineeringChangeOrderListService.RouteToCAVS	FALSE	Use CAVS for getUpdateEngineeringChangeOrderListService
Routing.ECOEngineeringChangeOrderEBS.UpdateEngineeringChangeOrderList.RouteToCAVS	FALSE	Use CAVS for EngineeringChangeOrderEBS getUpdateEngineeringChangeOrderListService
Routing.GetUpdateEngineeringChangeOrderListService.EBIZ_01.EndpointURI		runtime target endpoint URI
Routing.ECOEngineeringChangeOrderEBS.EBIZ_01.Endp		EngineeringChangeOrderEBS runtime target

Properties	(default) Value/Setting	Description
ointURI		endpoint URI
Routing.GetUpdateEngineeringChangeOrderListService.UpdateEngineeringChangeOrderListService.CAVS.EndpointURI		getUpdateEngineeringChangeOrderListService endpoint URI when set to CAVS
Routing.ECOEngineeringChangeOrderEBS.UpdateEngineeringChangeOrderList.CAVS.EndpointURI		EngineeringChangeOrderEBS endpoint URI when set to CAVS
CUSTOM_TRANSFORMATIONS	FALSE	Used to determine whether custom transformations should be used or not
INCLUDE_REVISD_ITEMS	T	Has a single Boolean Char value (T/F) this property is passed to PI/Sql API based on the value of the property Revised Item details are populated in the output ABM
INCLUDE_COMPONENT_ITEMS	T	Has a single Boolean Char value (T/F) this property is passed to PI/Sql API based on the value of the property Component item details are populated in the output ABM
INCLUDE_SUBSTITUTE_COMPONENTS	T	Has a single Boolean Char value (T/F) this property is passed to PI/Sql API based on the value of the property Substitute item details are populated in the output ABM
INCLUDE_REFERENCE_DESIGNATORS	T	Has a single Boolean Char value (T/F) this property is passed to PI/Sql API based on the value of the property Reference Designators of the Component item details are populated in the output ABM
ABCSEXTENSION.PREPROCESSABM	FALSE	User exit for the pre-process ABM should be called or not
ABCSEXTENSION.POSTPROCESSABM	FALSE	User exit for the post-process ABM should be called or not
ABCSEXTENSION.PREPROCESSEBM	FALSE	User exit for the pre-process EBM should be called or not
ABCSEXTENSION.POSTPROCESSEBM	FALSE	User exit for the post-process EBM should be called or not
TRACE.LOG.ENABLED	FALSE	Use tracelog for the flow
ServiceName	UpdateItemBalanceListEbizReqABCImpl	
Default.SystemID	EBIZ_01	Used to get the default XREF Target column name when TargetId is empty in incoming EBM
Routing.ItemBalanceService.GetItemBalanceService.RouteToCAVS	FALSE	Use CAVS for ItemBalanceService
Routing.ItemBalanceEBS.UpdateItemBalanceList.RouteToCAVS	FALSE	Use CAVS for ItemBalanceEBS
ItemBalanceService.EBIZ_01.Default.Target.EndpointURI		ItemBalanceService runtime target endpoint URI
ItemBalanceEBS.EBIZ_01.Default.Target.EndpointURI		ItemBalanceEBS runtime target endpoint URI
Routing.ItemBalanceService.GetItemBalanceService.CAVS.EndpointURI		ItemBalanceService endpoint URI when set to CAVS
Routing.ItemBalanceEBS.UpdateItemBalanceList.CAVS.EndpointURI		ItemBalanceEBS endpoint URI when set to CAVS
CUSTOM_TRANSFORMATIONS	FALSE	Used to determine whether custom transformations should be used or not
TRACE.LOG.ENABLED	FALSE	Use tracelog for the flow
ABCSEXTENSION.PREPROCESSABM	FALSE	User exit for the pre-process ABM should be called or not
ABCSEXTENSION.POSTPROCESSABM	FALSE	User exit for the post-process ABM should be called or not
ABCSEXTENSION.PREPROCESSEBM	FALSE	User exit for the pre-process EBM should be called or not
ABCSEXTENSION.POSTPROCESSEBM	FALSE	User exit for the post-process EBM should be called or not

Properties	(default) Value/Setting	Description or not
------------	-------------------------	-----------------------

Domain Value Maps

Domain Value Maps templates are XML files that conform to the Oracle SOA Suite DVM schema. The DVMs are stored in the ESB database and are maintained using the ESB Console user interface. Domain values are to be used for static lookups. Using ESB administration, you may import these XML files and then modify them according to your needs.

Out-of-the-Box DVMs

During installation, the DVMs used for the PIP are imported with default data mappings. The values mapped by these DVMs have to be changed as needed. There are many DVMs that are seeded and need not be touched. Since most of the Agile attributes being mapped are list values, the Agile data is not seeded and should be changed accordingly.

Note ICC manufacturer should be in synch before the DVMs are called.

Note DVM about Orgs should already be configured.

Some of the most used DVMs, with their Out-of-the-Box values, are listed below. You can modify their values as per your requirements. For complete list of available DVMs, including the most used ones, see the List of DVMs.

Important You can only add more rows of value mappings but should not change the DVM name, the column names, or the number of columns.

To see or modify a DVM:-

7. Logon to ESB Control in Enterprise Manager

The path would be

<http://<environment name>.us.oracle.com:<port number>/esb>

1. Click Maps Icon to retrieve the DVMs



2. Click the requisite DVM to see/modify.

Sample DVMs

AGILE_SITE_TARGET_MAPPING

DEFAULT_MASTER_ORG in Oracle EBS is specified here. This is used when the Multisite_Enabled property is set to False and no Org is specified for the Item where it extends to.

When the Multisite_Enabled property is true, the Sites in Agile are mapped to various Orgs in Oracle EBS. A Site could be mapped to multiple Orgs in the eBiz column with "|" delimiter.

AGILE_TARGET_SITE_MAPPING

The eBiz Orgs to Agile Sites are mapped. This is used for eBiz to Agile flows. There is one to one mapping between the eBiz Org to Agile Site.

ECO_ENGINEERINGCHANGEORDERLINE_STATUS_CODE

Used for Ebiz attribute -

ECO_ATTR/ECO_REVISSED_ITEM_TYPE/ECO_REVISSED_ITEM_TYPE_ITEM/STATUS_NAME

EBIZ_01	COMMON
Release	RELEASE
Open1	OPEN_1
EBS_Release	EBS_RELEASE
EBS Open	EBS_OPEN
Implemented	IMPLEMENTED
Create	CREATE
In Progress	IN_PROGRESS
Scheduled1	SCHEDULED_1
Cancelled	CANCELLED
Approval	APPROVAL
Pending Response	PENDING_RESPONSE
EBS_Create	EBS_CREATE
EBS_Review	EBS_REVIEW
EBS_Archive	EBS_ARCHIVE
Completed	COMPLETED
Hold	HOLD
Draft	DRAFT
Review	REVIEW
Released	RELEASED
Archive	ARCHIVE

ECO_STATUS_CODE

Used for Ebiz attribute - ECO_CHANGE_ORDER_TYPE/STATUS_NAME,
ECO_REVISSED_ITEM_TYPE/STATUS_NAME.

Used for Agile's Status attribute of a Change.

EBIZ_01	COMMON	AGILE9224_01
Release	RELEASE	
EBS_Approval	EBS_APPROVAL	
Open1	OPEN_1	
EBS_Release	EBS_RELEASE	
EBS Open	EBS_OPEN	
Implemented	IMPLEMENTED	Implemented
Create	CREATE	
In Progress	IN_PROGRESS	
Scheduled	SCHEDULED	Released
Cancelled	CANCELLED	
Approval	APPROVAL	
Pending Response	PENDING_RESPONSE	
EBS_Create	EBS_CREATE	
EBS_Review	EBS_REVIEW	
EBS_Archive	EBS_ARCHIVE	
Completed	COMPLETED	
Hold	HOLD	
Draft	DRAFT	
Review	REVIEW	
Released	RELEASED	
Archive	ARCHIVE	

ITEM_STATUS_CODE

Used for Ebiz Item attribute - ITEM_OBJ/MAIN_OBJ_TYPE/INVENTORY_ITEM_STATUS_CODE.

The Agile Item Lifecycle phase attribute is mapped.

EBIZ_01	COMMON	AGILE9224_01
A	A	
Concept	CONCEPT	
Design	DESIGN	Preliminary
Engineer	ENGINEER	Pilot
Inactive	INACTIVE	Inactive
Lease	LEASE	
Non-Stock	NONSTOCK	
Nwe B	NWEB	
OPM	OPM	
Obsolete	OBSOLETE	Obsolete
Pending	PENDING	
Phase-Out	PHASEOUT	
Production	PRODUCTION	Production
Prototype	PROTOTYPE	Prototype
R	R	

List of DVMs

Note The most important DVMs are listed in Bold typeface.

Domain Value Map	Description
ECO_CLASSIFICATION_CODE	Used for Ebiz attribute - ECO_ATTR/ECO_CHANGE_ORDER_TYPE/CHANGE_MANAGEMENT_TYPE
ECO_ENGINEERINGCHANGEORDERLINE_AVAILABLETOMRPPINDICATOR	Used for Ebiz attribute - ECO_ATTR/ECO_REVISSED_ITEM_TYPE/ECO_REVISSED_ITEM_TYPE_ITEM/MRP_ACTIVE
ECO_ENGINEERINGCHANGEORDERLINE_DISPOSITION_TYPE_CODE	Used for Ebiz attribute - ECO_REVISSED_ITEM_TYPE/DISPOSITION_TYPE
ECO_ENGINEERINGCHANGEORDERLINE_REVISSEDBILLOFMATERIALS_BILLOFMATERIALSCOMPONENTITEM_ATPCHECKREQUIREDINDICATOR	Used for Ebiz attribute - ECO_REVISSED_ITEM_TYPE/COMPONENT_ITEM_TBL/CHECK_ATP

Domain Value Map	Description
ECO_ENGINEERINGCHANGEORDERLINE_REVISDBILLOFMATERIALS_BILLOFMATERIALSCOMPONENTITEM_BILLOFMATERIALSSUBSTITUTECOMPONENTITEM_CHANGETYPECODE	Used for Ebiz attribute - ECO_REVISDBILLOFMATERIALS_BILLOFMATERIALSCOMPONENTITEM_BILLOFMATERIALSSUBSTITUTECOMPONENTITEM_CHANGETYPECODE
ECO_ENGINEERINGCHANGEORDERLINE_REVISDBILLOFMATERIALS_BILLOFMATERIALSCOMPONENTITEM_CHANGETYPECODE	Used for Ebiz attribute - ECO_REVISDBILLOFMATERIALS_BILLOFMATERIALSCOMPONENTITEM_CHANGETYPECODE, ECO_REVISDBILLOFMATERIALS_BILLOFMATERIALSCOMPONENTITEM_CHANGETYPECODE, ECO_REVISDBILLOFMATERIALS_BILLOFMATERIALSCOMPONENTITEM_CHANGETYPECODE, ECO_REVISDBILLOFMATERIALS_BILLOFMATERIALSCOMPONENTITEM_CHANGETYPECODE
ECO_ENGINEERINGCHANGEORDERLINE_REVISDBILLOFMATERIALS_BILLOFMATERIALSCOMPONENTITEM_COSTROLLUPINCLUSIONINDICATOR	Used for Ebiz attribute - ECO_REVISDBILLOFMATERIALS_BILLOFMATERIALSCOMPONENTITEM_COSTROLLUPINCLUSIONINDICATOR
ECO_ENGINEERINGCHANGEORDERLINE_REVISDBILLOFMATERIALS_BILLOFMATERIALSCOMPONENTITEM_FRACTIONALLOWEDINDICATOR	Used for ebiz attribute - ECO_REVISDBILLOFMATERIALS_BILLOFMATERIALSCOMPONENTITEM_FRACTIONALLOWEDINDICATOR
ECO_ENGINEERINGCHANGEORDERLINE_REVISDBILLOFMATERIALS_BILLOFMATERIALSCOMPONENTITEM_MUTUALLYEXCLUSIVEOPTIONINDICATOR	Used for Ebiz attribute - ECO_REVISDBILLOFMATERIALS_BILLOFMATERIALSCOMPONENTITEM_MUTUALLYEXCLUSIVEOPTIONINDICATOR
ECO_ENGINEERINGCHANGEORDERLINE_REVISDBILLOFMATERIALS_BILLOFMATERIALSCOMPONENTITEM_OPTIONALINDICATOR	Used for Ebiz attribute - ECO_REVISDBILLOFMATERIALS_BILLOFMATERIALSCOMPONENTITEM_OPTIONALINDICATOR
ECO_ENGINEERINGCHANGEORDERLINE_REVISDBILLOFMATERIALS_BILLOFMATERIALSCOMPONENTITEM_REQUIREDFORREVENUEINDICATOR	Used for ebiz attribute - ECO_REVISDBILLOFMATERIALS_BILLOFMATERIALSCOMPONENTITEM_REQUIREDFORREVENUEINDICATOR
ECO_ENGINEERINGCHANGEORDERLINE_REVISDBILLOFMATERIALS_BILLOFMATERIALSCOMPONENTITEM_REQUIREDTOSHIPINDICATOR	Used for Ebiz attribute - ECO_REVISDBILLOFMATERIALS_BILLOFMATERIALSCOMPONENTITEM_REQUIREDTOSHIPINDICATOR
ECO_ENGINEERINGCHANGEORDERLINE_REVISDBILLOFMATERIALS_BILLOFMATERIALSCOMPONENTITEM_SHIPPINGALLOWEDINDICATOR	Used for Ebiz attribute - ECO_REVISDBILLOFMATERIALS_BILLOFMATERIALSCOMPONENTITEM_SHIPPINGALLOWEDINDICATOR
ECO_ENGINEERINGCHANGEORDERLINE_REVISDBILLOFMATERIALS_BILLOFMATERIALSCOMPONENTITEM_SHIPPINGDOCUMENTVISIBILITYINDICATOR	Used for Ebiz attribute - ECO_REVISDBILLOFMATERIALS_BILLOFMATERIALSCOMPONENTITEM_SHIPPINGDOCUMENTVISIBILITYINDICATOR
ECO_ENGINEERINGCHANGEORDERLINE_REVISDBILLOFMATERIALS_ENGINEERINGINDICATOR	Used for Ebiz attribute - ECO_REVISDBILLOFMATERIALS_ENGINEERINGINDICATOR
ECO_ENGINEERINGCHANGEORDERLINE_REVISDBILLOFMATERIALS_NAME	Used for Ebiz attribute - ECO_REVISDBILLOFMATERIALS_NAME
ECO_ENGINEERINGCHANGEORDERLINE_REVISDBILLOFMATERIALS_TYPECODE	Used for Ebiz attribute - ECO_REVISDBILLOFMATERIALS_TYPECODE, ECO_REVISDBILLOFMATERIALS_TYPECODE, ECO_REVISDBILLOFMATERIALS_TYPECODE
ECO_ENGINEERINGCHANGEORDERLINE_STATUS_CODE	Used for Ebiz attribute - ECO_ATTR/ECO_REVISDBILLOFMATERIALS_STATUS_CODE
ECO_ENGINEERINGCHANGEORDERLINE_UPDATEWIPINDICATOR	Used for Ebiz attribute - ECO_REVISDBILLOFMATERIALS_UPDATEWIPINDICATOR
ECO_PRIORITY_CODE	Used for Ebiz attribute - ECO_CHANGE_ORDER_TYPE/PRIORITY_CODE

Domain Value Map	Description
ECO_REASON_CODE	Used for Ebiz attribute - ECO_CHANGE_ORDER_TYPE/REASON_CODE. Used for Agile's Reason Code attribute of a Change.
ECO_STATUS_CODE	Used for Ebiz attribute - ECO_CHANGE_ORDER_TYPE/STATUS_NAME, ECO_REVISSED_ITEM_TYPE/STATUS_NAME. Use for Agile's Status attribute of a Change.
ECO_TYPECODE	Used for Ebiz attribute - ECO_CHANGE_ORDER_TYPE/CHANGE_TYPE_CODE. Used for Agile's Change Type attribute of a Change.
ITEM_BILLING_TYPE_CODE	Used for Ebiz Item attribute - ITEM_OBJ/SERVICE_OBJ_TYPE/MATERIAL_BILLABLE_FLAG
ITEM_BOM_ITEM_TYPE_CODE	Used for Ebiz Item attribute - ITEM_OBJ/BOM_OBJ_TYPE/BOM_ITEM_TYPE
ITEM_CONSIGNMENT_ITEM_INDICATOR	Used for Ebiz Item attribute - ITEM_OBJ/GPLANNING_OBJ_TYPE/CONSIGNMENT_FLAG
ITEM_CONTRACT_TYPE_CODE	Used for Ebiz Item attribute - ITEM_OBJ/ASSET_OBJ_TYPE/CONTRACT_ITEM_TYPE_CODE
ITEM_CREATE_FIXED_ASSET_INDICATOR	Used for Ebiz Item attribute - ITEM_OBJ/SERVICE_OBJ_TYPE/ASSET_CREATION_CODE
ITEM_DUAL_UOM_TRACKING_INDICATOR	Used for Ebiz Item attribute - ITEM_OBJ/MAIN_OBJ_TYPE/TRACKING_QUANTITY_IND
ITEM_EFFECTIVITY_CONTROL_CODE	Used for Ebiz Item attribute - ITEM_OBJ/BOM_OBJ_TYPE/EFFECTIVITY_CONTROL
ITEM_INDICATOR	YES/NO DVM used for multiple Item attributes - ITEM_OBJ/INVENTORY_OBJ_TYPE/SERIAL_STATUS_ENABLED, ITEM_OBJ/INVENTORY_OBJ_TYPE/LOT_STATUS_ENABLED, ITEM_OBJ/INVENTORY_OBJ_TYPE/CYCLE_COUNT_ENABLED_FLAG, ITEM_OBJ/INVENTORY_OBJ_TYPE/LOT_MERGE_ENABLED, ITEM_OBJ/INVENTORY_OBJ_TYPE/LOT_SPLIT_ENABLED, ITEM_OBJ/INVENTORY_OBJ_TYPE/STOCK_ENABLED_FLAG, ITEM_OBJ/PURCHASING_OBJ_TYPE/INSPECTION_REQUIRED_FLAG, ITEM_OBJ/PURCHASING_OBJ_TYPE/PURCHASING_ENABLED_FLAG, ITEM_OBJ/PURCHASING_OBJ_TYPE/RECEIPT_REQUIRED_FLAG, ITEM_OBJ/PURCHASING_OBJ_TYPE/RFQ_REQUIRED_FLAG, ITEM_OBJ/PURCHASING_OBJ_TYPE/TAXABLE_FLAG, ITEM_OBJ/PURCHASING_OBJ_TYPE/MUST_USE_APPROVED_VENDOR_FL, ITEM_OBJ/RECEIVING_OBJ_TYPE/ALLOW_SUBSTITUTE_RECEIPTS_F, ITEM_OBJ/RECEIVING_OBJ_TYPE/ALLOW_UNORDERED_RECEIPTS_FL, ITEM_OBJ/BOM_OBJ_TYPE/BOM_ENABLED_FLAG, ITEM_OBJ/BOM_OBJ_TYPE/ENG_ITEM_FLAG, ITEM_OBJ/COSTING_OBJ_TYPE/COSTING_ENABLED_FLAG, ITEM_OBJ/COSTING_OBJ_TYPE/INVENTORY_ASSET_FLAG, ITEM_OBJ/ORDER_OBJ_TYPE/CUSTOMER_ORDER_FLAG, ITEM_OBJ/ORDER_OBJ_TYPE/RETURNABLE_FLAG,
ITEM_INVENTORY_PLANNING_CODE	Used for Ebiz Item attribute - ITEM_OBJ/GPLANNING_OBJ_TYPE/INVENTORY_PLANNING_CODE

Domain Value Map	Description
ITEM_LOTEXPIRATION_ON_RECEIPT_INDICATOR	Used for Ebiz Item attribute - ITEM_OBJ/INVENTORY_OBJ_TYPE/LOT_CONTROL_CODE
ITEM_MAKEORBUY_CODE	Used for Ebiz Item attribute - ITEM_OBJ/GPLANNING_OBJ_TYPE/PLANNING_MAKE_BUY_CODE
ITEM_RECEIVING_ROUTING_CODE	Used for Ebiz Item attribute - ITEM_OBJ/RECEIVING_OBJ_TYPE/RECEIVING_ROUTING_ID
ITEM_REPLENISHMENT_SOURCE_CODE	Used for Ebiz Item attribute - ITEM_OBJ/GPLANNING_OBJ_TYPE/SOURCE_TYPE
ITEM_RESERVATION_ALLOWED_INDICATOR	Used for Ebiz Item attribute - ITEM_OBJ/INVENTORY_OBJ_TYPE/RESERVABLE_TYPE
ITEM_RETURN_INSPECTION_REQUIRED_INDICATOR	Used for Ebiz Item attribute - ITEM_OBJ/PURCHASING_OBJ_TYPE/INSPECTION_REQUIRED_FLAG
ITEM_SERIALIZATION_EVENT_CODE	Used for Ebiz Item attribute - ITEM_OBJ/INVENTORY_OBJ_TYPE/SERIAL_NUMBER_CONTROL_CODE
ITEM_SERVICE_REQUEST_ENABLED_INDICATOR	Used for Ebiz Item attribute - ITEM_OBJ/ASSET_OBJ_TYPE/SERV_REQ_ENABLED_CODE
ITEM_STATUS_CODE	Used for Ebiz Item attribute - ITEM_OBJ/MAIN_OBJ_TYPE/INVENTORY_ITEM_STATUS_CODE. The Agile Item Lifecycle phase attribute is mapped.
ITEM_TRACK_INSTANCE_INDICATOR	Used for Ebiz Item attribute - ITEM_OBJ/SERVICE_OBJ_TYPE/COMMS_NL_TRACKABLE_FLAG
ITEM_UOM_CODE	Unit of Measure DVM for Item attributes - ITEM_OBJ/PHYSICAL_OBJ_TYPE/VOLUME_UOM_CODE, ITEM_OBJ/PHYSICAL_OBJ_TYPE/WEIGHT_UOM_CODE
ITEM_UOM_CONVERSION_USAGE_CODE	Used for Ebiz Item attribute - ITEM_OBJ/MAIN_OBJ_TYPE/ALLOWED_UNITS_LOOKUP_CODE
ITEM_WIP_SUPPLY_CODE	Used for Ebiz Item attribute - ITEM_OBJ/WIP_OBJ_TYPE/WIP_SUPPLY_TYPE
ITEM_PRIMARYCLASSIFICATIONCODE	Used for Ebiz Item attribute - ITEM_OBJ/MAIN_OBJ_TYPE/ITEM_CATALOG_GROUP_CODE. The Agile PartType of an Item is mapped to the eBiz ICC.
AGILE_SITE_TARGET_MAPPING	DEFAULT_MASTER_ORG in eBiz is specified here. This is used when the Multisite_Enabled property is set to False and no Org is specified for the Item where it extends to. When the Multisite_Enabled property is true, the Sites in Agile are mapped to various Orgs in eBiz. A Site could be mapped to multiple Orgs in the eBiz column with " " delimiter.
AGILE_TARGET_SITE_MAPPING	The eBiz Orgs to Agile Sites are mapped. This is used for eBiz to Agile flows. There is one to one mapping between the eBiz Org to Agile Site.
AGILE_INTEGRATION_USERS	The Agile Change originator users are mapped to the RequesterPartyReference in the EBM.

Application Interfaces

Application Interfaces are the Web Services and APIs that communicate and transact between Application and Integration Layers.

Agile PLM Interfaces

WSDLs

CreateItemEbizProvABCImpl		
ItemServiceRouter.wsdl		The interface to the Ebiz Process Item API - INV_EBI_ITEM_PUB.process_item()
UpdateItemBalanceListEbizReqABCImpl		
ItemBalanceRouter.wsdl		Used for the OA adapter "INV_EBI_ITEM_PUB.GET_ITEM_BALANCE" pl/sql function call.
UpdateItemListEbizReqABCImpl		
GetItemAttributesRouter.wsdl		Used for the OA adapter "INV_EBI_ITEM_PUB.GET_ITEM_ATTRIBUTES" pl/sql function call.
UpdateEngineeringChangeOrderListEbizReqABCImpl		
GetEngineeringChangeOrderServiceRouter.wsdl		Used for the OA adapter "INV_EBI_CHANGE_ORDER_PUB.GET_ECO_LIST_ATTR" pl/sql function call.
CreateEngineeringChangeOrderListEbizProvABCImpl		
ECOServiceRouter.wsdl		Used for the OA adapter INV_EBI_CHANGE_ORDER_PUB/PROCESS_CHANGE_ORDER_LIST pl/sql function call.

XSDs

CreateItemEbizProvABCImpl		
APPS_INV_EBI_ITEM_PUB_PROCESS_ITEM.xsd		Contains the CreateItem Request and Response ABM
ItemAttributeABM.xsd		Contains the GenerateItemNumber User Exit Output Type

Oracle EBS Interfaces

WSDLs

CreateItemEbizProvABCImpl		
ItemServiceRouter.wsdl		The interface to the Ebiz Process Item API - INV_EBI_ITEM_PUB.process_item()
UpdateItemBalanceListEbizReqABCImpl		

ItemBalanceRouter.wsdl	Used for the OA adapter "INV_EBI_ITEM_PUB.GET_ITEM_BALANCE" pl/sql function call.
UpdateItemListEbizReqABCImpl	
GetItemAttributesRouter.wsdl	Used for the OA adapter "INV_EBI_ITEM_PUB.GET_ITEM_ATTRIBUTES" pl/sql function call.
UpdateEngineeringChangeOrderListEbizReqABCImpl	
GetEngineeringChangeOrderServiceRouter.wsdl	Used for the OA adapter "INV_EBI_CHANGE_ORDER_PUB.GET_ECO_LIST_ATTR" pl/sql function call.
CreateEngineeringChangeOrderListEbizProvABCImpl	
ECOServiceRouter.wsdl	Used for the OA adapter INV_EBI_CHANGE_ORDER_PUB/PROCESS_CHANGE_ORDER_LIST pl/sql function call.

XSDs

CreateItemEbizProvABCImpl	
APPS_INV_EBI_ITEM_PUB_PROCESS_ITEM.xsd	Contains the CreateItem Request and Response ABM
ItemAttributeABM.xsd	Contains the GenerateItemNumber User Exit Output Type
GenerateItemNumberService.xsd	Contains input/output elements for GenerateItemNumber service. It exists on Ebiz server - <a href="http://<host>:<port>/webservices/AppsWSProvider/oracle/apps/inv/ebi/item/GenerateItemNumberService?wsdl">http://<host>:<port>/webservices/AppsWSProvider/oracle/apps/inv/ebi/item/GenerateItemNumberService?wsdl
UpdateItemBalanceListEbizReqABCImpl	
ItemBalanceABM.xsd	ABM for ItemBalanceABO
ItemBalanceABO.xsd	Contains structure of UpdateItemBalanceListABM and UpdateItemBalanceListResponseABM details. Created based on the input from the CP and output to the CP for UpdateItemBalance flow
APPS_INV_EBI_ITEM_PUB_GET_ITEM_BALANCE.xsd	Defines the input and output ABM of the PL/SQL API call. The package used is INV_EBI_ITEM_PUB.GET_ITEM_BALANCE
CommonEbizComponents.xsd	Defines common ResponseType element
UpdateItemListEbizReqABCImpl	
ItemAttributeABM.xsd	ABM for ItemAttributeABO
ItemAttributeABO.xsd	Created based on the input from the Concurrent Program to BPEL process and output to the Concurrent Program from BPEL process
APPS_INV_EBI_ITEM_PUB_GET_ITEM_ATTRIBUTES.xsd	The input and output ABM for the PL/SQL API call are defined in this XSD. Package is INV_EBI_ITEM_PUB.GET_ITEM_ATTRIBUTES
UpdateEngineeringChangeOrderListEbizReqABCImpl	
EngineeringChangeOrderABM.xsd	ABM for EngineeringChangeOrderABO
EngineeringChangeOrderABO.xsd	Created based on the input from the Concurrent Program to BPEL process and output to the Concurrent Program from BPEL process
APPS_INV_EBI_CHANGE_ORDER_PUB_GET_ECO_LIST_ATTR.xsd	Used for input/output ABM of INV_EBI_CHANGE_ORDER_PUB.GET_ECO_LIST_ATTR API call

**CreateEngineeringChangeOrderListEbizProvAB
CSImpl**APPS_INV_EBI_CHANGE_ORDER_PUB_PROCESS_C
HANGE_ORDER_LIST.xsdUsed for input/output ABM of
INV_EBI_CHANGE_ORDER_PUB/PROCESS_CHANGE_ORDER_LIST API
call

Handling Errors

Based on the roles defined for the services, email notifications are sent if an error occurs. The roles below can be assigned at various levels in a hierarchy (service, process, domain) so that when a service errors out -- the Error Handling Framework will use the role value to derive the user(s) that need to be notified of the error. The Error Handling Framework then notifies the user(s) via their preferred notification method, puts the error in the user's Oracle Worklist, and puts the error in the error log.

- **Role:** Actor role associated with the error notification. Actor roles receive notifications for and are assigned to error scenarios occurring in Oracle AIA integration flows.

The task is editable in the Error Console and is meant to be worked on and resolved by the actor assigned to the task.
- **FYI Role:** FYI role associated with the error notification. This role receives for-your-information (FYI) notifications for error scenarios occurring in Oracle AIA integration flows. An example of an FYI role is a customer service representative. The task is displayed in read-only view in the Error Console.

The roles specified for the 'Role' and 'FYI Role' parameters should be defined and associated with users in the system-jazn-data.xml File. The details of mail ids of the users will be taken from the **user-properties.xml** file.

For more information about the errors thrown by the applications - *Agile SDK and Oracle E-Business Suite*, see their product documentation.

Configuring AIA Error Notifications

The steps involved in configuring the Error Notifications are:

1. Identify the Users to be configured and their Respective roles to be assigned.
2. Configure the user properties file with the user objects and the group objects.
3. Configure the ns-emails.xml file with the required details.
4. Finally configure the BSR with the notifications in the AIA Console Setup > Error Notification page.

To configure Users and Roles:

1. Go to \$SOA_ORACLE_HOME/j2ee/oc4j_soa/config/system-jazn-data.xml

In the system-jazn-data.xml file under the users add a new user by using the template below.

```
<user>
```



```

    <name>UserName</name>
    <credentials>!password</credentials>
  </user>

```

Example

AGILE_INTEGRATION_EBIZ is the user defined here.

```

<user>
  <name>AGILE_INTEGRATION_EBIZ</name>
  <credentials>!welcome1</credentials>
</user>

```

1. Add a Role to the user under the roles tab, using the following template:

```

<role>
  <name>RoleToDefine</name>
  <members>
    <member>
      typeuser</type>
      <name>UserName</name>
    </member>
  </members>
</role>

```

Example

EBIZ_TO_AGILE is the role assigned to user AGILE_INTEGRATION_EBIZ

```

<role>
  <name>EBIZ_To_AGILE</name>
  <members>
    <member>
      typeuser</type>
      <name>AGILE_INTEGRATION_EBIZ</name>
    </member>
  </members>
</role>

```

To configure User Properties:

1. Go to \$SOA_ORACLE_HOME/bpel/system/services/config/users-properties.xml

Under the userObject add a new userObject using the following template:

```

<userObject >
  <name>UserName</name>
  <description>Description</description>
  <email>UserMailId</email>
  Product CollaborationTitle</title>
  <firstName>UserFirstName</firstName>
  <lastName />
  <timeZone>America/Los_Angeles</timeZone>
  <languagePreference>en-US</languagePreference>
  <notificationPreferences>Mail</notificationPreferences>
</userObject>

```

Similarly, add the group object using the following template:

```

<groupObject >
  <name>UserName</name>
  <email>MailId</email>
  <owners>OwnerName</owners>
</groupObject>

```

Example

```
<userObject >
  <name>AGILE_INTEGRATION_EBIZ</name>
  <description>AGILE_INTEGRATION_EBIZ User</description>
  <email>testuser@oracle.com</email>
  Product CollaborationAGILE_INTEGRATION_AGILE</title>
  <firstName>EBIZ_To_AGILE</firstName>
  <lastName />
  <timeZone>America/Los_Angeles</timeZone>
  <languagePreference>en-US</languagePreference>
  <notificationPreferences>Mail</notificationPreferences>
</userObject>
<groupObject >
  <name>AGILE_INTEGRATION_EBIZ</name>
  <email>testuser@oracle.com</email>
  <owners>AGILE_INTEGRATION_EBIZ</owners>
</groupObject>
```

Note See that the User Object and Group Object are declared before closing of the </principalObjects> tag in the User-properties.xml file.

To configure ns-emails.xml:

1. Go to \$SOA_ORACLE_HOME/bpel/system/services/config/ns_emails.xml

Edit the file ns_emails.xml file and set the NotificationMode="EMAIL". Then change *from name* and *from Email* to the address you want to configure. Set the SMTP Host name and port details for the outgoing mail settings.

To configure BSR:

The BSR provides a UI for managing mappings between actor and FYI roles and their participating applications for use during AIA error notifications. The error notifications you define on the Error Notifications page are stored in the BSR_ERROR_NOTIFICATIONS table.

Login to the AIA Console and navigate to Setup.Error Notifications Tab to see the page for configuring the Error Notifications.

The fields should be filled with the following details.

<i>Error Code</i>	Enter the error code associated with the error notification you are searching for. For all practical purposes, this can be set to any logical name in this release.
<i>System Code</i>	Select the system code associated with the error notification you are searching for. This is the system code of the participating application which should be configured using the System tab under Setup in BSR.
<i>Process Name</i>	Enter the BPEL process name associated with the error notification you are searching for.
<i>Service Name</i>	Enter the service name associated with the error notification you are searching for. This is the business process in which the service is participating. For all practical purposes, this can be the same as the BPEL process name.
<i>Role</i>	Select the actor role associated with the error notification you are searching for. Specify an actor role that you want to receive notification regarding this error. This is the role that will be responsible for taking action to correct the error that generated the notification.

<i>FYI Role</i>	Select the FYI role associated with the error notification you are searching for. This is the role that will be notified of the error, but will not be responsible for taking any actions to correct the error that generated the notification.
<i>Search</i>	Click to execute a search for error notifications based on your search criteria. Note For a given process, if no entry is found in the BSR_ERROR_NOTIFICATIONS table, the default roles specified in AIAConfigurationProperties.xml are used. Therefore, you are not required to populate the BSR_ERROR_NOTIFICATIONS table unless there is an explicit need.
<i>Delete</i>	Select radio button for the error notification row you want to delete and click the Delete button to execute the deletion.
<i>Create</i>	Click to add a row to the Search Result grid, where you can enter details for a new error notification.
<i>Save</i>	Click to save all entries and updates to the page.
<i>Undo</i>	Click to undo all updates made to the page after the last save.

Logic used to determine Notification Roles for an Error

The Error Handling Framework uses runtime values and the data you enter on this page to execute the following hierarchical logic to determine the appropriate notification roles for an error:

- If all four runtime values (SYSTEM_CODE, ERROR_CODE, SERVICE_NAME, and PROCESS_NAME) are available and they map to an error notification entry in this table, use the specified notification roles.
- If the ERROR_CODE, SERVICE_NAME, and PROCESS_NAME are available and map to an error notification entry in this table, use the specified notification roles.
- If the SERVICE_NAME and PROCESS_NAME are available and map to an error notification entry in this table, use the specified notification roles.
- If the SERVICE_NAME is available and maps to an error notification entry in this table, use the specified notification roles.

- If none of these values are available, the default values are fetched from the AIAConfigurationProperties.xml file.

ORACLE Application Integration Architecture Home Logout

Home Service Repository Validation System **Setup**

Error Notification | System | Flexfield | Configuration

Setup > Error Notification

Error Notification

Error Code System Code
 Process Name Role
 Service Name FYI Role

Search Result

|

Select	Error Code	System Code	Process Name	Service Name	Role	FYI Role
<input type="radio"/>	DIG_ERROR_CODE	DIAGNOSTICS_SYSTEM v	DIAGNOSTICSEHPROCESS	DIAGNOSTICSEHSERVICE	DIAGNOSTICS_ACTOR	DIAGNOSTICS_CSR
<input type="radio"/>	CREATE_ECO_Req	Agile9224_01 v	ProcessEngineeringChangeOrd	ProcessEngineeringChangeOrd	AGILE_To_EBIZ	AGILE_To_EBIZ
<input type="radio"/>	Create_ECO_Prov	M00MQ102 v	CreateEngineeringChangeOrd	CreateEngineeringChangeOrd	AGILE_To_EBIZ	AGILE_To_EBIZ
<input type="radio"/>	UPDATE_ITEM	M00MQ102 v	UpdateItemLEbizReqABCSIr	UpdateItemLEbizReqABCSIr	EBIZ_To_AGILE	EBIZ_To_AGILE
<input type="radio"/>	Update_ECO	M00MQ102 v	UpdateEngineeringChangeOrd	UpdateEngineeringChangeOrd	EBIZ_To_AGILE	EBIZ_To_AGILE
<input type="radio"/>	UPDATE_bal	M00MQ102 v	UpdateItemBalanceListAgilePr	UpdateItemBalanceListAgilePr	EBIZ_To_AGILE	EBIZ_To_AGILE

Worklist Application

The Oracle Worklist application is used to provide an Error Console for the Oracle Application Integration Architecture (AIA). The Error Console application is a user interface (UI) that Actor roles, such as integration administrators, and FYI roles can use to access details about Oracle AIA ecosystem service errors that have been assigned to them. Based on their roles, users will be able to interact with the following types of tasks in the Error Console:

Login into the worklist using the UserName and Password you configured in the system-jazn-data.xml to see the worklists in the below URL.

http://<hostname>:<port>/integration/worklistapp/Login

Example: Login with AGILE_INTEGRATION_EBIZ/welcome1 to worklist app to see the error notifications assigned to this user.

ORACLE BPM Worklist
Welcome, AGILE_INTEGRATION_EBIZ [jazzn.com]

Home | Reports | Preferences | Logout

My Tasks | Initiated Tasks

My Tasks (Inbox)

Work Queues

- Inbox
 - My Work Queues
 - Standard Views
 - High Priority Tasks
 - Tasks Due Soon
 - New Tasks
 - My Views
 - None
 - Proxy Work Queues
 - Delegated Views
 - None

Search: My & Group Any Assigned Go

Keyword Category Priority Status Advanced Search

Task Number	Title	Priority	Assigned Users	Assigned Groups	State	Created Date	Expiration Date	Actions
10107	Error in AIA UpdateItemListEbizReqABCSImpl Process FYI	3		EBIZ_To_AGILE	Assigned	Jul 11, 2008 2:26 AM		-- Select an Action -- Go
10109	Error in AIA UpdateItemListEbizReqABCSImpl Process	3		EBIZ_To_AGILE	Assigned	Jul 11, 2008 2:26 AM		-- Select an Action -- Go

Error Handling in PIP Queue Manager

Any failure in the processing of a Change Order is captured by the Queue Manager. This "Errored" process can be identified in the Queue Monitor, symbolized by ⚠. See the sample screen below.

Change Order Queue

Resubmit

Remove

⏸ Suspend

⏪ Resume

⏩ Refresh

⏶ Select All

⏷ Select None

	Reference	Change Number	Release Time	Processed Time	Process Status
<input type="checkbox"/> ⚠	ATO03096	EC3014	13-Jul-2008 23:23:17 PDT	13-Jul-2008 23:24:01 PDT	Errored
<input checked="" type="checkbox"/> ✓	ATO03095	MO0208	13-Jul-2008 23:17:56 PDT	13-Jul-2008 23:18:53 PDT	Completed
<input checked="" type="checkbox"/> ✓	ATO03094	EC3012	13-Jul-2008 23:17:50 PDT	13-Jul-2008 23:18:42 PDT	Completed

To see the reason of error, click the link 'Errored' in the Process Status column. It pops-up an Error Message window, similar to the sample message given below.

Org Code: V1 ECO Name: EC3014 Err Msg: ORA-00001: unique constraint (INV.MTL_MFG_PART_NUMBERS_U1) violated INV_EBI_ITEM_HELPER.process_part_num_association

Details

Name: CreateEngineeringChangeOrderListEbizProvABCSImpl

Id: 220087

Stack Trace

This Error Message comprises of two parts:

1. Error Text - This is the text of error source, which can be from any participating ABCS that may have faulted.
2. Details - The Error Details consist of:
 - Service Name - The name of the service where an Instance failed to process.
 - Instance ID - The identification number of the Instance that failed.

Multiple faults generated by the service are captured and displayed in this UI. In addition to this, any failures in the flows would be captured in the AIA's Error Logs. These can be seen from Enterprise Manager Console logs section.

In an event when a certain service is down and the error is not related to the payload, users may resubmit the change in the Queue Manager UI.

Customizations

This chapter includes the following:

▪ Customizing the Transformations	97
▪ User Exits	101

The Integration provides two approaches of customizations:

- Customizing the Transformations - This approach allows to modify, add the mappings between the attributes of the participating applications.
- User Exits - These Pre-defined extension points provided in the OOTB BPEL flow. You can plug-in your own logic at these exit points to validate, enrich and transform data.

Customizing the Transformations

There may be a need for Out-of-the-Box and User Defined attributes mapping between applications, which are not covered as part of standard transformations. Considering this, the transformation files, i.e., the XSLs, have been externalized to facilitate the implementers to carry out the following:

- Modify the Out-of-the-Box transformations.
- Add new mappings for the Agile attributes to EBM attributes. These Agile attributes could either be the ones that have not been mapped out-of-the-box including any of the flex-fields.

The transformations support the following requirements:

- In the integration flow, there are multiple transformations involving multiple ABO/Ms and EBO/Ms.
- The transformations support the flex field mappings
- In this mechanism of transformations, where in customer can provide the XSLs for the complex transformations that are not part of the standard transformations. It is required to support the transformations for user-defined EBO extensions (ex. Custom tags) and in places where customer want to override the standard transformation logic.

Transformation Rules

- For a transformation from ABM into EBM, all the flex-fields (interchangeably user defined attribute / UDA/ Flex attribute) go under SpecificationGroup element under the main EBM element with a matching type like ValueText (for text values), ValueNumeric (for numeric values) etc carrying the values.

- For the transformations from EBM into ABM depending upon the identification element of Specification and SpecificationGroup pull processing is done to populate the UDAs.
- For Classification elements, like ItemClassification etc, the field values are translated using a configuration, into EBM and vice-versa. The configuration states are <<Name in Agile>> <<Name on EBO>> <<Name on Oracle Manufacturing>>. For rest of the classification elements like part type, product family etc fields need to be packed/ unpacked on/from the XXCatalog element on the EBM.

Customization in Agile

The XSL transformations in Agile PLM Integration are externalized, i.e., these are hosted on implementation server under

oracle_home/Apache/Apache/htdocs/AIAComponents/Transformations/Agile/Release1/<ABCName>

Note Refer Customization Points of each Process (in corresponding chapters).

The implementers modify the XSLs based on the customization points against each process to modify, include new mappings for the Agile attributes to the EBM elements. The entire behavior of the OOTB mappings can enhanced using this approach.

Important Server has to be restarted to bring the changes into effect.

A few sample customizations have been provide at <AIA_HOME>\PIPS\Core\Agile\Samples. A sample customization of ECO attributes is given below.

Sample Customizations

User Defined ECO - in Sites Tab under Items

Mapping

Agile	EBM	Oracle EBS
Item:Site:List01	corecomEBO:RevisedItem/corecomEBO:BaseUOMCode	Primary UOM Code
Item:Site:List06	corecomEBO:RevisedItem/corecomEBO:ItemManufacturingCharacteristics/corecomEBO:StructureAllowedIndicator	BOM Allowed Flag
Item:Site:List05	corecomEBO:RevisedItem/corecomEBO:ItemManufacturingCharacteristics/corecomEBO:EngineeringItemIndicator	Engineering Item Flag

Custom Code

Location:

Templates:

```
createEngineeringChangeOrderLines_With_SiteData
createEngineeringChangeOrderLines_With_OrgData
createEngineeringChangeOrderLines_With_DefaultMasterOrgData
```

Landmark:

Code Snippet: Code snippet is given only for "BOM Allowed" , "Engineering Item" flags

```
<xsl:if test="./changeABO:AffectedItem/changeABO:Site[itemABO:SiteName =
$varSiteID]/itemABO:List04">
<corecomEBO:StructureAllowedIndicator><xsl:value-of
select="./changeABO:AffectedItem/changeABO:Site[itemABO:SiteName =
$varSiteID]/itemABO:List04"/></corecomEBO:StructureAllowedIndicator>
</xsl:if>

<xsl:if test="./changeABO:AffectedItem/changeABO:Site[itemABO:SiteName =
$varSiteID]/itemABO:List05">
<corecomEBO:EngineeringItemIndicator><xsl:value-of
select="./changeABO:AffectedItem/changeABO:Site[itemABO:SiteName =
$varSiteID]/itemABO:List05"/></corecomEBO:EngineeringItemIndicator>
</xsl:if>
```

Customization in Oracle E-Business Suite

To customize a Mapping File:

1. Pickup the <flow>_Custom XSL file from
 oracle_home/Apache/Apache/htdocs/AIAComponents/Transformations/<flow_name>/
 where <flow> is the process name, such as ECO_CHANGE_ORDER_TYPE
 - a. If only flex-field transformation has to be modified then the custom targets can be modified to include the new mappings.
 For example, ECO_CHANGE_ORDER_TYPE_Custom in Create Eco flow for the change order flex attributes
 - b. The user need not set the custom transformations property in this case as the targets are already included in the base transformation file.
1. If the entire mapping has to be modified then the user can copy the base mappings into the custom target (e.g. Custom in Create ECO flow) in the custom file and modify the mappings as desired.
 - a. After modifying the file the user has to set the Custom Transformations property in the AIA Configurations file and reload the configurations from the AIA console.
2. After doing either 2) or 3) the server has to be re-started so that the new transformations get loaded into the JVM.
3. The changes transformations will now take effect.

Templates In The Custom Files:

1. CreateEngineeringChangeOrderListEbmToAbm: The following templates are used in the custom transformation files that are used to map the flex field attributes.

- ECO_CHANGE_ORDER_TYPE_Custom
- ECO_REVISED_ITEM_TYPE_ITEM_Custom
- REFERENCE_DESIGNATOR_TBL_Custom
- COMPONENT_ITEM_TBL_Custom
- SUBSTITUTE_COMPONENT_TBL_ITEM_Custom
- STRUCTURE_HEADER_Custom

The Custom template is used for modifying the entire mapping.

2. ItemEbmToAbm: The template CUSTOM_OBJ_TYPE_Custom is used to map the flex field attributes.

The template "Custom" is used to replace the entire mapping.

3. UpdateItemListABMToEBM: The template ItemSpecificationGroup_Custom is used to map the flex field to the specification group in the EBM. The "UpdateItemListABMToEBMCustom" template is used to replace the entire file.

4. ItemBalanceAbmToEbm: Since no flex-field is involved here only one template "CustomABMToEBM" for the entire mapping customization is provided.

Mapping of Flex Attribute1 in CreateEngineeringChangeOrderListEbmToAbm_Custom.xsl

Note	Since the templates are directly included inside the objects only the specific attributes being mapped should be put in the custom xsl and not the entire hierarchy.
------	--

Replacing The entire Mapping:

Here the template *UpdateItemListABMToEBMCustom* is actually used to do the entire ABM to EBM transformation.

1. Change the Property in AIA Configurations file:

Note	1.File Path: <AIA_HOME>/config/AIAConfigurationProperties.xml
------	---

2. Not needed when only the flex field mapping is done.

3.The use custom transformation property is per integration flow and has to be set accordingly.

2. Reload the AIA Configurations file:

3. Restart the server:

Note	The URL of the page: http://<host:port>/em
------	--

User Exits

Extensibility Points

Agile to Oracle E-Business Suite Flow

- Just prior to the execution of transformation of ABM to EBM
- Just prior to the invocation of Enterprise Business Service
- Just prior to the execution of transformation of EBM to ABM
- Just prior to the invocation of callback service or response return

Oracle E-Business Suite to Agile Flow

- Just prior to the execution of transformation of EBM to ABM
- Just prior to the invocation of Application Service
- Just prior to the execution of transformation of ABM to EBM
- Just prior to the invocation of callback EBS or return of response message

Development Steps for User Exits

1. Identify which out-of-box flow is to be extended.
2. Identify the suitable exit point in the Flow.
3. Develop the Flow
4. Configure the Out-of-box flow to include the newly developed flow
5. Test the developed flow

Appendix A

Appendices

This Appendix includes the following:

- Agile to Oracle E-Business Suite Entity Maps 103
- Concurrent Program Implementation Details..... 116

Agile to Oracle E-Business Suite Entity Maps

Bill of Materials Mappings

Agile Entity: Attribute	Bill of Materials EBO	Oracle EBS Entity: Attribute	Comments
Part/Document. Title Block.Number	ItemReference/Identification/ID	Structure: Primary: Item Number	
Site	ItemReference/Identification/ ContextID (schemeID: OrganizationCode)	Structure: Context Organization Code	
	ItemReference/Identification/ Revision/Code	Structure: Primary: Item Revision Code	
	TypeCode	Structure: Primary: Engineering Flag	AIA EBO Team bug 6709708
	Name	Structure: Primary: Structure Name	
	AttributeOverrideAllowedIndicator	Structure: Primary: Enable for Attribute Update	For Common Bill
	ImplementationDate	Structure: Primary: Implementation date	
	EffectivityControlCode	Structure: Primary: Effectivity Control	
	PrimaryIndicator	Structure: Primary: Structure Name - Primary	Indicates if it is Primary BOM/Structure.
BOM Notes	Note/Content	Structure: Primary: Description	
	Note/LanguageCode		
Change Order	EngineeringChangeOrderReference/ EngineeringChangeOrderIdentification/ ID	Structure: Primary: Change Order Number	Change Notice
Site	EngineeringChangeOrderReference/ EngineeringChangeOrderIdentification/ ContextID (schemeID: OrganizationCode)	Structure: Primary: Change Order Organization Code	
Part/Document. Title Block.Number	CommonBillOfMaterialsReference/ ItemIdentification/ID	Structure: Primary: Common Bill Item Number	For Common BOM in Agile
Site	CommonBillOfMaterialsReference/ ItemIdentification/ContextID (schemeID: OrganizationCode)	Structure: Primary: Common Bill Item Organization Code	For Common BOM in Agile
	OriginalBillOfMaterialsReference/ ItemIdentification/ID	Structure: Primary:	

Agile Entity: Attribute	Bill of Materials EBO	Oracle EBS Entity: Attribute	Comments
		Copy Bill Item Number	
	OriginalBillOfMaterialsReference/ ItemIdentification/ContextID (schemeID: OrganizationCode)	Structure: Primary: Copy Bill Item Organization Code	
Part/Document.BOM. Find Number	ComponentItem/Identification/ID (SchemeID: ComponentItemSequence)	Component: Primary: Component Item Sequence	
	ComponentItem/Identification/ContextID (SchemeID: ComponentOperationSequence)	Component: Primary: Component Operation Sequence	
	ComponentItem/Identification/ContextID (SchemeID: ComponentNewOperationSequence)	Component: Primary: Component New Operation Sequence	For Change Order Processing
Part/Document. Title Block.Number	ComponentItem/ItemReference Identification/ID	Component: Primary: Item Number	
Site	ComponentItem/ItemReference Identification/ ContextID (schemeID: OrganizationCode)	Component: Context Organization Code	
	ComponentItem/StorageUnitCode	Component: Primary: Basis	
Qty	ComponentItem/Quantity	Component: Primary: Quantity	
	ComponentItem/BasisQuantity	Component: Primary: Sales Order Basis	
	ComponentItem/MinimumQuantity	Component: Primary: Minimum Quantity	
	ComponentItem/MaximumQuantity	Component: Primary: Maximum Quantity	
	ComponentItem/YieldFactor	Component: Primary: Yield	
Item Notes	ComponentItem/Comment (languageCode)	Component: Primary: Comment	
	ComponentItem/ PlanningQuantityMultiplier	Component: Primary: Planning	
	ComponentItem/OptionalIndicator	Component: Primary: Optional	
	ComponentItem/ModelPlanLevelCode	Component: Primary: Plan Level Code	
	ComponentItem/ OptionalOnModelIndicator	Component: Primary: Optional On Model Indicator	
	ComponentItem/ MutuallyExclusiveOptionIndicator	Component: Primary: Mutually Exclusive	
	ComponentItem/CostRollupInclusionIndicator	Component: Primary: Include in Cost Rollup	
	ComponentItem/ATPCheckRequiredIndicator	Component: Primary: Check ATP	
	ComponentItem/ShippingAllowedIndicator	Component: Primary: Shippable	
	ComponentItem/ ShippingDocumentVisibilityIndicator	Component: Primary: Include on ship Docs	
	ComponentItem/ChangeTypeCode	Component: Primary: Add/Modify/ Disable	
	ComponentItem/FractionAllowedIndicator	Component: Primary: Enforce Integer Quantity	
	ComponentItem/ AutoRequestIndicator	Component: Primary:	

Agile Entity: Attribute	Bill of Materials EBO	Oracle EBS Entity: Attribute	Comments
		Auto Request Material	
	ComponentItem/ OverrideAttributesComponentItemIdentifier	Component: Primary: Override Attributes	For Common Bill Components.
	ComponentItem/ RequiredToShipIndicator	Component: Primary: Required To Ship	
	ComponentItem/ RequiredForRevenueIndicator	Component: Primary: Required For Revenue	
Effective Date	ComponentItem/ EffectiveTimePeriod/ StartDateTime	Component: Primary: Date Effective From	
	ComponentItem/ EffectiveTimePeriod/ EndDateTime	Component: Primary: Date Effective To	
	ComponentItem/ EngineeringChangeOrderReference/ EngineeringChangeOrderIdentification/ ID	Component: Primary: Change Order Number	Change Notice
	ComponentItem/ EngineeringChangeOrderReference/ EngineeringChangeOrderIdentification/ ContextID (schemeID: OrganizationCode)	Component: Primary: Change Order Organization Code	
	ComponentItem/ InventoryLocationReference/ LocationIdentification/ID	Component: Primary: Supply Locator	
	ComponentItem/ InventoryLocationReference/ LocationIdentification/ ContextID	Component: Primary: Supply Locator Organization Code	
	ComponentItem/ ModelBillOfMaterialsComponentItemReference/ BillOfMaterialsComponentItemIdentification/ID	Component: Primary: Model Item Number	
	ComponentItem/ ModelBillOfMaterialsComponentItemReference/ BillOfMaterialsComponentItemIdentification/ ContextID (schemeID: OrganizationCode)	Component: Primary: Model Item Organization Code	
	ComponentItem/ ComponentItemSpecificationGroup/ SpecificationGroup/Name	BOM/Structure Component Flex/User Defined Attribute Group Name	
	ComponentItem/ ComponentItemSpecificationGroup/ SpecificationGroup/ Specification/Name	BOM/Structure Component Flex/User Defined Attribute Name	
	ComponentItem/ ComponentItemSpecificationGroup/ SpecificationGroup/Specification/ ValueCode BillOfMaterialsSpecificationGroup / SpecificationGroup/Specification/ ValueText (languageCode)	BOM/Structure Component Flex/User Defined Attribute Value	
	ComponentItem/ ComponentItemSpecificationGroup/ SpecificationGroup/Specification/ ValueNumeric		
	ComponentItem/ ComponentItemSpecificationGroup/ SpecificationGroup/Specification/ ValueQuantity (unitCode)		

Agile Entity: Attribute	Bill of Materials EBO	Oracle EBS Entity: Attribute	Comments
	ComponentItem/ ComponentItemSpecificationGroup/ SpecificationGroup/Specification/ ValueAmount (currencyCode)		
	ComponentItem/ ComponentItemSpecificationGroup/ SpecificationGroup/Specification/ ValueDate		
	ComponentItem/ ComponentItemSpecificationGroup/ SpecificationGroup/Specification/ ValueDateTime		
	ComponentItem/SubstituteComponentItem/ ItemReference/Identification/ID	Substitute Component: Item Number	
	ComponentItem/SubstituteComponentItem/ ItemReference/Identification/ContextID (schemeID: OrganizationCode)	Substitute Component: Context Organization Code	
	ComponentItem/SubstituteComponentItem/ Quantity	Substitute Item: Quantity	
	ComponentItem/SubstituteComponentItem/ ChangeTypeCode	Substitute Item: Add/Disable	
	ComponentItem/SubstituteComponentItem/ EngineeringChangeOrderReference/ EngineeringChangeOrderIdentification/ID	Substitute Item: Change Order Number	
	ComponentItem/SubstituteComponentItem/ EngineeringChangeOrderReference/ EngineeringChangeOrderIdentification/ ContextID (schemeID: OrganizationCode)	Substitute Item: Change Order Organization Code	
	ComponentItem/ SubstituteComponentItemSpecificationGroup Name	Substitute Component Flex Attribute Group	
	ComponentItem/ SubstituteComponentItemSpecificationGroup/ Specification/Name	Substitute Component Flex Attributes	
	ComponentItem/ SubstituteComponentItemSpecificationGroup/ SpecificationGroup/ Specification/ ValueCode	Substitute Component Flex Attribute Value	
	ComponentItem/ SubstituteComponentItemSpecificationGroup/ SpecificationGroup/Specification/ ValueText (languageCode)		
	ComponentItem/ SubstituteComponentItemSpecificationGroup/ SpecificationGroup/Specification/ ValueNumeric		
	ComponentItem/ SubstituteComponentItemSpecificationGroup/ SpecificationGroup/Specification/ ValueQuantity (unitCode)		
	ComponentItem/ SubstituteComponentItemSpecificationGroup/ SpecificationGroup/Specification/ ValueAmount (currencyCode)		
	ComponentItem/ SubstituteComponentItemSpecificationGroup/ SpecificationGroup/Specification/ ValueDate		

Agile Entity: Attribute	Bill of Materials EBO	Oracle EBS Entity: Attribute	Comments
	ComponentItem/ SubstituteComponentItemSpecificationGroup/ SpecificationGroup/Specification/ ValueDateTime		
Ref Des	ComponentItem/ProcessingInstruction/ Identification/ID	Reference Designator	
Ref Des	ComponentItem/ ProcessingInstruction/Note/Content	Reference Designator: Description	
	ComponentItem/ ProcessingInstruction/Note/LanguageCode		
	ComponentItem/ ProcessingInstruction/ChangeTypeCode	Reference Designator: Add/Disable	
	ComponentItem/ ProcessingInstruction/ EngineeringChangeOrderReference/ EngineeringChangeOrderIdentification/ID	Reference Designator: Change Order Number	
	ComponentItem/ ProcessingInstruction/ EngineeringChangeOrderReference/ EngineeringChangeOrderIdentification/ContextID (schemeID: OrganizationCode)	Reference Designator: Change Order Organization Code	
	BillOfMaterialsSpecificationGroup/ SpecificationGroup/Name	BOM/Structure Header Flex/User Defined Attribute Group Name	
	BillOfMaterialsSpecificationGroup/ SpecificationGroup/ Specification/Name	BOM/Structure Header Flex/User Defined Attribute Name	
	BillOfMaterialsSpecificationGroup / SpecificationGroup/Specification/ ValueCode	BOM/Structure Header Flex/User Defined Attribute Value	
	BillOfMaterialsSpecificationGroup / SpecificationGroup/Specification/ ValueText (languageCode)		
	BillOfMaterialsSpecificationGroup / SpecificationGroup/Specification/ ValueNumeric		
	BillOfMaterialsSpecificationGroup / SpecificationGroup/Specification/ ValueQuantity (unitCode)		
	BillOfMaterialsSpecificationGroup / SpecificationGroup/Specification/ ValueAmount (currencyCode)		
	BillOfMaterialsSpecificationGroup / SpecificationGroup/Specification/ ValueDate		
	BillOfMaterialsSpecificationGroup / SpecificationGroup/Specification/ ValueDateTime		

Item EBO Mappings

Agile Entity: Attribute	Item EBO	Oracle EBS Entity: Attribute Group: Attribute	Comments
Part/Document. Title Block.Number	ItemIdentification/ID	Item: Primary: Item Number	Item Concatenated Segments. Required for Item Creation
Site	ItemIdentification/ContextID (schemeID: OrganizationCode)	Item: Context Organization Code	Context Organization Code. Required for Item Creation
Part/Document. Title Block. Description	ItemIdentification/Description	Item: Primary: Description	Required for Item Creation
	ItemIdentification/ApplicationObjectKey/ID	Item: Inventory Item Id	Inventory item identifier
	ItemIdentification/ApplicationObjectKey/ContextID (schemeID: OrganizationId)	Item: Organization Id	Organization identifier
	ItemIdentification/AlternateObjectKey/ID (schemeID: "SourceSystemReference") (SchemeAgencyID: <source system >)	Item: Source System Cross-Reference: Value of source system reference	
	ItemIdentification/Revision/Number (schemeID: RevisionId)	Item Revision: Revision Id	Item revision identifier
Change Orders Class.Affected Items.New Revision	ItemIdentification/Revision/Code	Item Revision: Revision Code	3 Character Revision Code, Alphanumeric, Revision sort sequence based on ASCII value of Revision Code and Revision Effective Date. Required for Revision Creation
	ItemIdentification/Revision/Label	Item Revision: Revision Label	Required for Revision Creation
	ItemIdentification/Revision/Description	Item Revision: Description	
	ItemIdentification/Revision/Reason	Item Revision: Reason	
ECO/MCO/SCO. Affected Item. Effectivity Date	ItemIdentification/Revision/EffectiveDate	Item Revision: Effective Date	Required for Revision Creation
	TypeCode	Item: Primary: User Item Type	
	SerialControlIndicator	Item: Inventory: Serial Status Enabled	
	LotControlIndicator	Item: Inventory: Lot Status Enabled	
	ServiceIndicator	Item: Service: Contract Item Type (Service)	Service Item Flag is maintained internally at the table level when Contract Item Type Value is 'Service'.
	DualUOMTrackingIndicator	Item: Primary: Tracking	
	UOMConversionUsageCode	Item: Primary: Conversions	
	BaseUOMCode	Item: Primary: Primary Unit of Measure	Required for Item Creation. Can be defaulted using Profile Option.
	SecondaryUOMCode	Item: Primary: Secondary	

Agile Entity: Attribute	Item EBO	Oracle EBS Entity: Attribute Group: Attribute	Comments
		Unit of Measure	
	StorageUOMCode	N/A	
	ShippingUOMCode	N/A	
Item Type	PrimaryClassificationCode	Item: Primary: Item Catalog Category	
Change Orders Class.Affected Items.New Lifecycle	Status/Code	Item: Primary: Item Status	Required for Item Creation. Can be defaulted using Profile Option.
	TemplateItemReference/Name (languageCode)	Item Template: Template Name Item Template: Language	Template can be used to setup values for multiple attributes for an Item. Certain operational attributes like BOM allowed flag etc required for integration can be either provided via attribute mapping or provided using the Item template.
	ItemSpecificationGroup / SpecificationGroup/Name	Item Flex/User Defined Attribute Group Name	Item Flex/User Defined Attribute Group Name
	ItemSpecificationGroup/SpecificationGroup/ Specification/Name	Item Flex/User Defined Attribute Name	Item Flex/User Defined Attribute Name
	ItemSpecificationGroup/ SpecificationGroup/Specification/ ValueCode	Item Flex/User Defined Attribute Value	Item Flex/User Defined Attribute Value
	ItemSpecificationGroup/ SpecificationGroup/Specification/ ValueText (languageCode)		Language Code for Translatable Text attributes
	ItemSpecificationGroup/ SpecificationGroup/Specification/ ValueNumeric		
	ItemSpecificationGroup/ SpecificationGroup/Specification/ ValueQuantity (unitCode)		
	ItemSpecificationGroup/ SpecificationGroup/Specification/ ValueAmount (currencyCode)		Unit of Measure for Number attributes
	ItemSpecificationGroup/ SpecificationGroup/Specification/ ValueDate		
	ItemSpecificationGroup/ SpecificationGroup/Specification/ ValueDateTime		
	LifecycleCharacteristics/LifecycleCode	Item: Primary: Lifecycle	
	LifecycleCharacteristics/LifecyclePhaseCode	Item: Primary: Lifecycle Phase	Item Lifecycle is required to define Lifecycle Phase.
	LifecycleCharacteristics/RevisionLifecyclePhaseCode	Item Revision: Lifecycle Phase	Item Lifecycle and Lifecycle phase should have been defined for the Item.

Agile Entity: Attribute	Item EBO	Oracle EBS Entity: Attribute Group: Attribute	Comments
	PhysicalCharacteristics/VolumeMeasure (unitCode)	Item: Physical Attributes: Volume	
		Item: Physical Attributes: Volume Unit Of Measure	
	PhysicalCharacteristics/WeightMeasure (unitCode)	Item: Physical Attributes: Weight	
		Item: Physical Attributes: Weight Unit Of Measure	
	PhysicalCharacteristics/HeightMeasure (unitCode)	Item: Physical Attributes: Height	
		Item: Physical Attributes: Dimension Unit of Measure	
	PhysicalCharacteristics/LenghtMeasure (unitCode)	Item: Physical Attributes: Length	
		Item: Physical Attributes: Dimension Unit of Measure	
	PhysicalCharacteristics/WidthMeasure (unitCode)	Item: Physical Attributes: Width	
		Item: Physical Attributes: Dimension Unit of Measure	
	InventoryCharacteristics/CycleCountEnabledIndicator	Item: Inventory: Cycle Count Enabled	
	InventoryCharacteristics/LotExpirationOnReceiptIndicator	Item: Inventory: Control	Item: Inventory: Lot Expiration Control
	InventoryCharacteristics/LotMergeEnabledIndicator	Item: Inventory: Lot Merge Enabled	
	InventoryCharacteristics/LotSplitEnabledIndicator	Item: Inventory: Lot Split Enabled	
	InventoryCharacteristics/ReservationAllowedIndicator	Item: Inventory: Reservable	
	InventoryCharacteristics/SerializationEventCode	Item: Inventory: Serial Number Generation	
	InventoryCharacteristics/ShelfLifeDuration	Item: Inventory: Shelf Life Days	
	InventoryCharacteristics/StockingAllowedIndicator	Item: Inventory: Stockable	
	InventoryCharacteristics/InitialLotNumberPrefix	Item: Inventory: Lot Starting Prefix	
	InventoryCharacteristics/InitialLotNumberSuffix	N/A	
	InventoryCharacteristics/InitialSerialNumberPrefix	Item: Inventory: Serial Starting Prefix	
	InventoryCharacteristics/InitialSerialNumberSuffix	N/A	
	InventoryCharacteristics/UnitCost/Amount (currencyCode)	Item: Cost	
	InventoryCharacteristics/UnitCost/PerQuantity (unitCode)		
	ItemPurchasingCharacteristics/AssetClassificationCode	Item: Purchasing: Asset Category	
	ItemPurchasingCharacteristics/DebitGLAccountCode	Item: Purchasing: Expense Account	
	ItemPurchasingCharacteristics/HazardClassificationCode	Item: Purchasing: Hazard Class	
	ItemPurchasingCharacteristics/ ReceiptInspectionRequiredIndicator	Item: Purchasing: Inspection Required	
	ItemPurchasingCharacteristics/UnitListPrice/Amount	Item: Purchasing: List	

Agile Entity: Attribute	Item EBO	Oracle EBS Entity: Attribute Group: Attribute	Comments
		Price	
	ItemPurchasingCharacteristics/PurchasingAllowedIndicator	Item: Purchasing: Purchasable	
	ItemPurchasingCharacteristics/ReceiptRequiredIndicator	Item: Purchasing: Receipt Required	
	ItemPurchasingCharacteristics/RFQRequiredIndicator	Item: Purchasing: RFQ Required	
	ItemPurchasingCharacteristics/TaxableIndicator	Item: Purchasing: Taxable	
	ItemPurchasingCharacteristics/TaxCode	Item: Purchasing: Tax Code	
	ItemPurchasingCharacteristics/IssueUOMCode	Item: Purchasing: Unit of Issue	
	ItemPurchasingCharacteristics/ UseApprovedSupplierIndicator	Item: Purchasing: Use Approved Supplier	
	ItemPurchasingCharacteristics/ ReceiptSubstitutionAllowedIndicator	Item: Receiving: Allow Substitute Receipts	
	ItemPurchasingCharacteristics/ UnorderedReceiptAllowedIndicator	Item: Receiving: Allow Unordered Receipts	
	ItemPurchasingCharacteristics/ OverReceiptQuantityPercent	Item: Receiving: Tolerance Percentage	
	ItemPurchasingCharacteristics/ReceiptRoutingCode	Item: Receiving: Receipt Routing	
	ItemPurchasingCharacteristics/ReceivingDurationTolerance/ UnderDuration	Item: Receiving: Days Early	
	ItemPurchasingCharacteristics/ReceivingDurationTolerance/ OverDuration	Item: Receiving: Days Late	
	ItemPlanningCharacteristics/ConsignmentItemIndicator	Item: General Planning: Consigned	
	ItemPlanningCharacteristics/LotSizeMultiplier	Item: General Planning: Fixed Lot Multiplier	
	ItemPlanningCharacteristics/InventoryPlanningCode	Item: General Planning: Inventory Planning Method	
Make/Buy	ItemPlanningCharacteristics/MakeOrBuyCode	Item: General Planning: Make or Buy	
	ItemPlanningCharacteristics/ MaximumProductionOrderQuantity	Item: General Planning: Maximum	Item: General Planning: Maximum Order Quantity
	ItemPlanningCharacteristics/ MinimumProductionOrderQuantity	Item: General Planning: Minimum	Item: General Planning: Minimum Order Quantity
	ItemPlanningCharacteristics/ReplenishmentSourceCode	Item: General Planning: Type	Item: General Planning: Source Type
	ItemPlanningCharacteristics/ShrinkageRate	Item: MPS/MRP Planning: Shrinkage Rate	
	ItemPlanningCharacteristics/ReorderCharacteristics/ ReorderQuantity	Item: General Planning: Fixed Quantity	
	ItemPlanningCharacteristics/ReorderCharacteristics/ MaximumReorderQuantity	Item: General Planning: Maximum Order	
	ItemPlanningCharacteristics/ReorderCharacteristics/ MinimumInventoryQuantity	Item: General Planning: Minimum Order	

Agile Entity: Attribute	Item EBO	Oracle EBS Entity: Attribute Group: Attribute	Comments
	ItemPlanningCharacteristics/ReorderCharacteristics/ MaximumSupplyDuration	Item: General Planning: Maximum Days of Supply	
	ItemPlanningCharacteristics/ReorderCharacteristics/ MinimumSupplyDuration	Item: General Planning: Minimum Days of Supply	
	ItemPlanningCharacteristics/ ProcessingLeadTimeCharacteristics/ CumulativeManufacturingDuration	Item: Lead Times: Cumulative Manufacturing	
	ItemPlanningCharacteristics/ ProcessingLeadTimeCharacteristics/ CumulativeTotalDuration	Item: Lead Times: Cumulative Total	
	ItemPlanningCharacteristics/ ProcessingLeadTimeCharacteristics/ FixedDuration	Item: Lead Times: Fixed	
	ItemPlanningCharacteristics/ ProcessingLeadTimeCharacteristics/ DurationUOMCode	N/A	
	ItemPlanningCharacteristics/ ProcessingLeadTimeCharacteristics/ ReferenceLotSizeQuantity	Item: Lead Times: Lead Time Lot Size	
	ItemPlanningCharacteristics/ ProcessingLeadTimeCharacteristics/ PostprocessingDuration	Item: Lead Times: Postprocessing	
	ItemPlanningCharacteristics/ ProcessingLeadTimeCharacteristics/ PreprocessingDuration	Item: Lead Times: Preprocessing	
	ItemPlanningCharacteristics/ ProcessingLeadTimeCharacteristics/ ProcessingDuration	Item: Lead Times: Processing	
	ItemPlanningCharacteristics/ ProcessingLeadTimeCharacteristics/ UnitProductuionDuration	Item: Lead Times: Variable	
	ItemPlanningCharacteristics/MinMaxCharacteristics/ MaximumQuantity	Item: General Planning: Maximum	Item: General Planning: Maximum Min-Max Quantity
	ItemPlanningCharacteristics/MinMaxCharacteristics/ MinimumQuantity	Item: General Planning: Minimum	Item: General Planning: Minimum Min-Max Quantity
	ItemManufacturingCharacteristics/StructureAllowedIndicator	Item: Bill of Materials: BOM Allowed	
	ItemManufacturingCharacteristics/BOMItemTypeCode	Item: Bill of Materials: BOM Item Type	
	ItemManufacturingCharacteristics/ ConfiguratorModelTypeCode	Item: Bill of Materials: Configurator Model Type	
	ItemManufacturingCharacteristics/EffectivityControlCode	Item: Bill of Materials: Effectivity Control	
	ItemManufacturingCharacteristics/EngineeringItemIndicator	Item: Bill of Materials: Engineering Item	
Cost	ItemManufacturingCharacteristics/CostingEnabledIndicator	Item: Costing: Costing Enabled	
	ItemManufacturingCharacteristics/InventoryAssetIndicator	Item: Costing: Inventory Asset Value	
	ItemManufacturingCharacteristics/StandardLotSizeQuantity	Item: Costing: Standard Lot Size	
	ItemManufacturingCharacteristics/WIPSupplyTypeCode	Item: Work In Progress: Type	
	ItemManufacturingCharacteristics/ReturnPercentTolerance/ UnderPercent	Item: Order Management: Under Return	Item: Order Management: Under Return Tolerance
	ItemManufacturingCharacteristics/ReturnPercentTolerance/ OverPercent	Item: Order Management: Over Return	Item: Order Management: Over Return Tolerance
	ItemManufacturingCharacteristics/ShipmentPercentTolerance/	Item: Order Management:	Item: Order

Agile Entity: Attribute	Item EBO	Oracle EBS Entity: Attribute Group: Attribute	Comments
	UnderPercent	Under Shipment	Management: Under Shipment Tolerance
	ItemManufacturingCharacteristics/ShipmentPercentTolerance/OverPercent	Item: Order Management: Over Shipment	Item: Order Management: Over Shipment Tolerance
	ItemManufacturingCharacteristics/BaseModelItemReference/Identification/ID	Item: Bill of Materials: Base Model	
	ItemOrderManagementCharacteristics/AssembleToOrderIndicator	Item: Order Management: Assemble to Order	
	ItemOrderManagementCharacteristics/BackOrderEnabledIndicator	Item: Web Store: Back Orderable	
	ItemOrderManagementCharacteristics/SaleableIndicator	Item: Order Management: Customer Ordered	
	ItemOrderManagementCharacteristics/OrderableIndicator	Item: Order Management: Customer Orders Enabled	
	ItemOrderManagementCharacteristics/ReturnAllowedIndicator	Item: Order Management: Returnable	
	ItemOrderManagementCharacteristics/ReturnInspectionRequiredIndicator	Item: Order Management: RMA Inspection Required	
	ItemOrderManagementCharacteristics/InvoicingEnabledIndicator	Item: Invoicing: Invoice Enabled	
Shippable Item	ItemOrderManagementCharacteristics/ShippableIndicator	Item: Order Management: Shippable	
	ItemServiceCharacteristics/BillingTypeCode	Item: Service: Billing Type	
	ItemServiceCharacteristics/ContractTypeCode	Item: Service: Contract Item Type	
	ItemServiceCharacteristics/CreateFixedAssetIndicator	Item: Service: Create Fixed Asset	
	ItemServiceCharacteristics/DefaultServiceContractDuration	Item: Service: Duration	Item: Service: Service Contracts Duration
	ItemServiceCharacteristics/DefaultServiceContractPeriodCode	Item: Service: Duration Period	Item: Service: Service Contracts Duration Period
	ItemServiceCharacteristics/ContractCoverageEnabledIndicator	Item: Service: Enable Contract Coverage	
	ItemServiceCharacteristics/DefectTrackingEnabledIndicator	Item: Service: Enable Defect Tracking	
	ItemServiceCharacteristics/ProvisioningEnabledIndicator	Item: Service: Enable Provisioning	
	ItemServiceCharacteristics/ServiceBillingEnabledIndicator	Item: Service: Enable Service Billing	
	ItemServiceCharacteristics/ServiceRequestEnabledIndicator	Item: Service: Service Request	
	ItemServiceCharacteristics/WarrantyDelayDuration	Item: Service: Starting Delay (Days)	
	ItemServiceCharacteristics/TrackInstanceIndicator	Item: Service: Track in Installed Base	

Agile Entity: Attribute	Item EBO	Oracle EBS Entity: Attribute Group: Attribute	Comments
Commodity, Item Category, Product Line(s), Part Family	ItemClassification/ClassificationCode	Category: Name	Alternate Catalog Category Concatenated Segments
	ItemClassification/CatalogReference/CatalogIdentification/ID	Catalog: Name	Alternate Catalog
Part/Document. Manufacturers. Manufacturer	ItemManufacturer/ ManufacturerPartyReference/OrganizationName	Manufacturer: Manufacturer Name	
Part/Document. Manufacturers. Mfr Part Number	ItemIdentification/ManufacturerItemID (SchemeAgencyID: <Manufacturer Name>)	MPN: Manufacturer Part Number	
	ItemManufacturer/Status/Code	MPN: Approval Status	
	ItemManufacturer/EffectiveTimePeriod/StartDateTime	MPN: Effective From	
	ItemManufacturer/EffectiveTimePeriod/EndDateTime	MPN: Effective To	

Engineering Change Order EBO Mappings

Agile Entity: Attribute	Engineering Change Order EBO	Oracle EBS Entity: Attribute	Comments
Number	Identification/ID	Change Order: Primary: Change Order Number	Change Notice
Site	Identification/ContextID (schemeID: OrganizationCode)	Change Order: Context Organization Code	
Description of Change	Description	Change Order: Primary: Description	
Date Originated	InitiationDate	Change Order: Primary: Initiation Date	
Final Complete Date	ImplementationDate	Change Order: Primary: Implementation Date	
	PriorityCode	Change Order: Primary: Priority	
Reason Code	ReasonCode	Change Order: Primary: Reason	
Change Type	TypeCode	Change Order: Primary: Change Order Type	
Change Category	ClassificationCode	Change Order: Primary: Classification Code	
Status	Status/Code	Change Order: Primary: Status	
Originator	RequesterPartyReference/ PersonName	Change Order: Primary: Requestor	
Change Analyst, Component Engineer	OwnerPartyReference/ PersonName	Change Order: Primary: Assigned To	
	EngineeringChangeOrderLine/ Description	N/A	
Disposition01	EngineeringChangeOrderLine/ DispositionTypeCode	Revised Item: Primary: Disposition	

Agile Entity: Attribute	Engineering Change Order EBO	Oracle EBS Entity: Attribute	Comments
	EngineeringChangeOrderLine/ EarliestEffectiveDate	Revised Item: Primary: Earliest Schedule Date and Time	
Effective Date	EngineeringChangeOrderLine/ EffectiveDate	Revised Item: Primary: Schedule Date and Time	
	EngineeringChangeOrderLine/ AvailableToMRPIndicator	Revised Item: Primary: MRP Active	
	EngineeringChangeOrderLine/ UpdateWIPIndicator	Revised Item: Primary: Update Jobs/Schedules	
	EngineeringChangeOrderLine/ EffectivityCode	N/A	
	EngineeringChangeOrderLine/ UseUpMRPPlanName	Revised Item: Primary: Plan Name	
	EngineeringChangeOrderLine/ Status/Code	Revised Item: Primary: Status	
	EngineeringChangeOrderLine/ EffectivityControllItemReference/ Identification/ID	Revised Item: Primary: Use Up Item Number	
	EngineeringChangeOrderLine/ EffectivityControllItemReference/ Identification/ContextID	Revised Item: Primary: Use Up Item Organization Code	
Flex Attributes	EngineeringChangeOrderLine/ EngineeringChangeOrderLineSpecificationGroup/	Revised Item Flex Attributes	
	EngineeringChangeOrderLine/ CurrentItem		
Revised Item	EngineeringChangeOrderLine/ RevisedItem	Revised Item	
	EngineeringChangeOrderLine/ CurrentBillOfMaterial		
Revised BOM/Structure	EngineeringChangeOrderLine/ RevisedBillOfMaterials	Revised BOM/Structure	
	EngineeringChangeOrderStatusHistory	Change Order: Primary: History	Change Order Status History.
	EngineeringChangeOrderSpecificationGroup	Flex/User Defined Attribute Group	

Item Balance EBO Mappings

Agile Entity: Attribute	Item Balance EBO	Oracle EBS Entity: Attribute	Comments
Part/Document. Title Block.Number	InventoryBalance/ItemReference/ Identification/ID	Item: Primary: Item Number	
Site	InventoryBalance/ItemReference/ Identification/ContextID	Item: Context Organization	
	InventoryBalance/OnHandQuantity	Item: On Hand Quantity	

Concurrent Program Implementation Details

The three flows from Oracle E-Business Suite to Agile are to be scheduled for data to be sent out at regular intervals. This is carried out by using Oracle E-Business Suite Concurrent Programs, which can be run at various intervals and scheduled with UI options. The programs are:

1. Publish Item Attributes Updates
2. Publish Item Balance Updates
3. Publish Engineering Change Order Updates

Features:

After the PIP Integration setup is complete, carry out the following:

1. Initial Publication of Items/ECOs from Ebiz

Run an ad-hoc request specifying from and to date (If this is not done the CP will pick all the Items/ECOs that got updated in the last 30 days). Also specific Orgs could be specified along with from and to date in the CP parameters, from performance perspective to send the data in multiple CP requests as part of Implementation setup

2. Periodic/Scheduled Publication of Items/ECOs from Ebiz

We recommend customers setting the 'Updated in the last X hrs' parameter with a reasonable value by default for the Concurrent Programs that are being setup to run at a schedule frequency. This parameter value should be specified apart from the schedule frequency setup in the Concurrent Program setup. We suggest Customers set the same 'X' hrs for the Concurrent Program Schedule setup

(OR)

Customers can leave all the parameters empty and schedule the CP to run at a particular schedule frequency that he desires.

3. On Demand/Ad-hoc Publication of Items/ECOs from Ebiz

The Items/ECOs that failed during the scheduled execution must be sent as separate ad-hoc request by specifying appropriate value to the parameters. The Items/ECOs of the Organizations that failed during the scheduled publication can be obtained from the log information to provide input parameters for the ad-hoc Concurrent Request to sync up the data between Ebiz and Agile after resolving the reported publication error.

Parameters:

1. Item/ECO Names

- The Items/ECOs that are to be published should be entered separated by *double semi-colon*.
- This is a text parameter of maximum length 240.

Examples:

ItemName1;;ItemName2;;ItemName3

ChangeOrderName1;;ChangeOrderName2;; ChangeOrderName3

1. Organization Codes

- Organization codes must be specified separated by double semi-colon.
- This is a text parameter of maximum length 240.

Example1:

Item Names - Item1;;Item2;;Item3

Organization Codes - Org1;;Org2

If Item1 exists in Org1, Item2 in Org2 and Item3 in both Org1 & Org2 then,

Items Published: Item1:Org1, Item2:Org2, Item3:Org1, Item3:Org2

Example2:

Organization Codes - Org1;;Org2

Updated in the last X Hrs – 10

Items/ECOs Published: Items/ECOs that got updated in the last 10 hours from Org 1 & Org2.

2. From Date

This is a Standard Date Time parameter.

3. To Date

This is a Standard Date Time parameter. This parameter should be entered only if From Date parameter is given a value.

4. Updated in the last X hrs

This is a number parameter of maximum length 15. If this parameter is provided a value then the From Date and To Date parameter values will not be considered

Supported Functionalities:

1. Scheduled Request with no value provided to all parameters

Items/ECOs that got updated from the last completed-scheduled request will be picked for publication. The Errored Items/ECOs from the previous run will not be automatically picked for the publication. All the errored Items/ECOs need to be published by using ad-hoc CP request functionality and providing appropriate CP request parameters for such ad-hoc requests.

2. Ad-hoc Request with no value provided to all parameters

It is recommended for the customers to provide appropriate parameter values for the CP ad-hoc requests. In case if no value is provided then Items/ECOs that got updated from the last completed-scheduled request will be published.

If there are no scheduled requests prior to the current request then it will pick all the Items/ECOs that got updated from the last completed request. If there are no completed requests prior to the current one (First request of CP) then it will choose the data that got updated in the last 30 days.

Expected Behaviors

The following table shows the expected behavior of the requests based on the input values provided. A tick-mark (x) implies that the values are specified, while adash implies they are not.

Item/ECO Names	Organization Codes	From Date	To Date	Updated in last X hrs	Expected Items/ECOs to be published
x	-	-	-	-	Specified Items/ECOs from all assigned Organization
-	x	-	-	-	Items/ECOs updated from the last Completed request from the specified organization (If no last completed request then last 30 days). This is done in order to maintain the performance of the system.
-	-	x	-	-	Items/ECOs updated between the specified From Date and System Date from all Organizations
-	-	-	x	-	Error (From Date cannot be empty when to Date is specified)
-	-	-	-	x	Items/ECOs updated in the last X hours from all Organizations
x	x	-	-	-	Specified Items/ECOs from Specified Organizations if exist
-	x	x	-	-	Items/ECOs updated between the specified From Date and System Date from specified Organizations
-	x	-	-	x	Items/ECOs updated in the last X hours from specified Organizations
x	-	x	-	-	Specified Items/ECOs updated between the specified From Date and System Date from all assigned Organizations
x	-	-	x	-	Error (From Date cannot be empty when to Date is specified)
x	-	-	-	x	Specified Items/ECOs updated in the last 'X' hrs specified from all assigned Organizations
x	-	x	x	-	Specified Items/ECOs updated between the specified From Date and To Date from all assigned Organizations
x	-	x	x	x	Specified Items/ECOs updated in the last 'X' hrs specified, from all assigned Organizations. The specified From Date and To Date will be ignored
x	x	-	x	x	Specified Items/ECOs from Specified Organizations if exist and updated in the last 'X' hrs specified. The specified To Date will be ignored
x	x	x	x	x	Specified Items/ECOs from Specified Organizations if exist and updated in the last 'X' hrs specified. The specified From Date and To Date will be ignored