Oracle® Application Integration Architecture

Agile Product Lifecycle Management Integration Pack for SAP: Design to Release Implementation Guide

Release 11.2

E36364-03

June 2013

The Agile Product Lifecycle Management Integration Pack for SAP: Design to Release implementation guide is a valuable resource for administrators and developers involved in the implementation, administration and deployment of Oracle's next-generation integrated enterprise PLM processes provided by Oracle Application Integration Architecture (Oracle AIA) Pre-Built Integrations Release 11.4.

The first part of this guide focuses on understanding the pre-built integration between Agile PLM and SAP. It also lists the various assumptions and constrains, process flows,interfaces and integration services used by Agile PLM Pre-Built integrations.

The second part of this guide discusses the prerequisites, post installation configuration steps, cross-references and National language support (NLS) required for integrating Agile PLM with SAP.

Oracle Application Integration Architecture Agile Product Lifecycle Management Integration Pack for SAP: Design to Release Implementation Guide, Release 11.2

E36364-03

Primary Author: Oracle Corporation

Contents

Preface

Welcome to the Agile Product Lifecycle Management Integration Pack for SAP: Design to Release Implementation guide.

What’s New in this Guide

- The Oracle Application Integration Architecture Installation and Upgrade Guide for Pre-Built Integrations is restructured into a general installation chapter with an individual configuration and deployment chapter for each pre-built integration.
- The term process integration pack is replaced with the term pre-built ntegrations.
- The implementation guides are restructured into two parts: design and set up.
  - Part I - Design: This part provides functional overviews, activity diagrams, assumptions and constraints, and technical sequence diagrams and steps.
  - Part II - Set up: This part provides prerequisites, data requirements, and configuration steps.
- Starting with this release, these integrations are no longer available:
  - Oracle CRM On Demand Integration Pack for JD Edwards EnterpriseOne: Lead to Order
  - Oracle Workforce Administration Integration Pack for PeopleSoft Human Resources

Common Oracle AIA Pre-Built Integration Guides

- Oracle Application Integration Architecture Pre-Built Integrations 11.4 includes the following guides shared by all products delivered with this release:
- Oracle Application Integration Architecture Installation and Upgrade Guide for Pre-Built Integrations Release 11.4.

  This guide provides an overview of the installation process, including how to install, configure, and deploy your pre-built integrations. The steps required to upgrade your pre-built integrations to the latest release are also provided.

- Oracle Application Integration Architecture Pre-Built Integrations 11.1: Utilities Guide

  This guide describes:

  - How to work with and configure Session Pool Manager (SPM), which is a service in the Oracle SOA Suite web server whose primary function is to manage a pool of web server session tokens that can be reused by BPEL flows.

  - How to deploy and configure the AIACompositeScheduler. This is a utility component that is used by pre-built integrations to schedule a service-oriented architecture (SOA) composite to be invoked at the specified time interval.

- Oracle Application Integration Architecture Pre-Built Integrations 11.4: Product-to-Guide Index.The product-to-Guide index lists the guides that provide information for each product delivered in this release.

Master Notes

Master notes provide a one-stop reference point for the following:

- Latest manuals
- Alerts
- Troubleshooting details
- FAQs
- Patching information
- AIA community link and more.

For master notes for the Agile Product Lifecycle Management Integration Pack for SAP: Design to Release, visit Master Note ID 1419205.2 on My Oracle Support (MOS): https://support.oracle.com/

Additional Resources

The following resources are also available:

- Oracle Application Integration Architecture Foundation Pack:

  Oracle AIA Pre-Built integrations require Foundation Pack 11.1.1.6, RUP patch 14253266 or later be installed. Refer to the Foundation Pack documentation library on OTN to download the Foundation Pack guides at http://docs.oracle.com/cd/E23943_01/aia.htm.

- Oracle Application Integration Architecture: Product-to-Guide Index:

  Oracle Technology Network: http://www.oracle.com/technetwork/index.html

- Known Issues and Workarounds:

  My Oracle Support: https://support.oracle.com/

- Release Notes:

  Oracle Technology Network: http://www.oracle.com/technetwork/index.html

- Documentation updates:

  Oracle Technology Network: http://www.oracle.com/technetwork/index.html

Chapter 1: Agile PLM Integration Pack for SAP: Design to Release

The Agile Product Lifecycle Management Integration Pack for SAP: Design to Release pre-built integration is designed to address the primary requirements for product information synchronization between Agile Product Collaboration and System Application and Products (SAP).

This chapter includes the following sections:

- Overview
- Agile Product Lifecycle Management (PLM) to SAP Integration Architecture
- Solution Design Assumptions and Constraints
- Components of Agile SAP Design To Release Pre-Built Integration

Overview

The Agile Product PLM Integration Pack for SAP: Design to Release pre-built integration is aimed at enabling product development process between Agile PLM and SAP. This integration allows rapid implementation of Agile's next generation integrated enterprise PLM processes and helps customers reduce cost and mitigate risks associated with third party and custom integrations.

This integration aims to meet the following objectives:

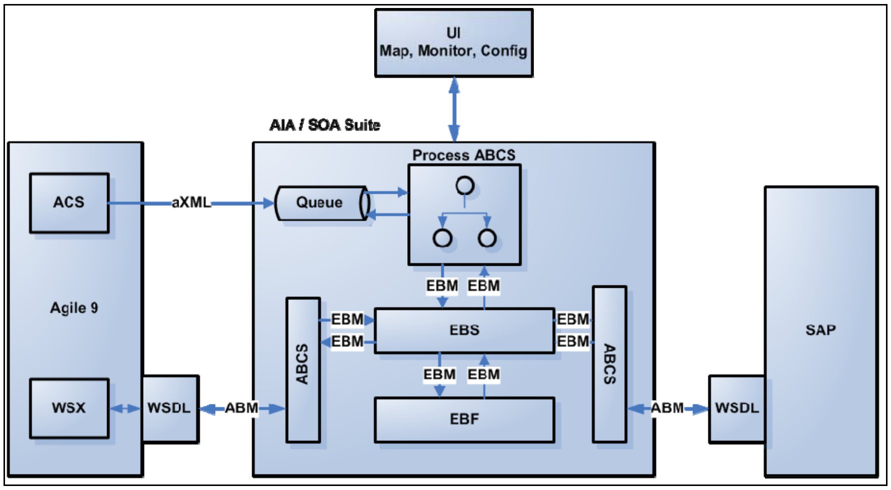
- Follows a business process flow
- Maintains data accuracy and integrity at all times in both the systems
- Allows for easy configuration of business process flow, data field mappings, and transformations tools, such as Rules, Lookups, and User Exits
- Provides for robust exception handling including ease of understanding for end user, strong notification configurability, and easy troubleshooting

  These are the functionalities of the Agile PLM to SAP integration:

- Manufacturing release of new product definition and product launch
- Change Management of previously launched products
- Bi-directional synchronization of Engineering Change status and Item Cost information, from SAP to Agile PLM
- Tracking and monitoring of the change processing queue

Agile PLM to SAP Integration Architecture

The Agile PLM to SAP integration is requester-provider type integration, as shown in this diagram:



Agile to SAP Processes

These are the Agile to SAP Processes:

- Change Order Release
- New Part Request

Change Order Release

During the product design phase, new products and parts are introduced and existing parts go through design changes. Change Orders (CO) are used to release information about the attributes as well as the design information of a part to the manufacturing system.

The Change Order Release Process (CORP) 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, specifications, and so on, the Enterprise Resource Planning (ERP) system typically has many more attributes and placeholders for information than the PLM system. Therefore, the change order release must be updated in the ERP system.

The release of a change order in Agile PLM system acts as a trigger for the synchronization of product design information with the ERP system. Because the Agile PLM system is the system of records for product design data, the synchronization process typically involves the transfer of the released revision of the product design from Agile PLM to the manufacturing system. The Agile PLM system should be configured to trigger the synchronization process on any Change Order status as required.

The *Change Order Release* and *New Product Release* processes use the same integration sequence.

For more information about the integration for both processes, see Chapter 2 - Process Integration for Change Order Release.

New Part Request

You can trigger a New Part Request (NPR) process on-demand. This is a synchronous process, which reserves and fetches the part number from the SAP system.

For more information about NPR processes, see Chapter 3 - Process Integration for New Part Request.

SAP to Agile Processes

Although Agile PLM is considered the system of records, there may be several part attributes that are maintained within ERP. To provide information on these attributes within the PLM item data, the SAP to Agile process enables synchronous or batch updates of the configured field values from ERP into the designated item fields of Agile PLM.

For example, the update of the cost of an item in ERP gets updated in Agile PLM in batch mode, based on the background schedule of the job in SAP.

Solution Design Assumptions and Constraints

These are the design assumptions and constraints:

Design Assumptions

1. Agile Content Server (ACS) is used for Events to trigger the payload from Agile PLM to the Integration.
2. This design assumes that the following statements are true:
   - Pre-defined blank templates are available for Custom fields
   - Transformation logic for classification elements are pre-coded in the out-of-the-box (OOTB) XSL. However, modifications may be required to suit your PLM implementation
3. This design leverages Oracle Application Integration Architecture (Oracle AIA) Error Handling Framework.

Design Constraints

4. In some cases, configuration driven XSLT may not reflect the changes immediately and may require a restart because the main XSL sheet gets cached after a successful compilation
5. ACS limits the Events to be triggered from workflow only for the Change Status, which may be used for this Integration.
6. Error handling capabilities of this integration are constrained by the capabilities of the Oracle AIA Error Handling Framework.

Components of Agile PLM Integration Pack for SAP : Design to release

The Agile PLM Integration Pack for SAP: Design to Release pre-built integration includes the following components:

- Agile Content Service (ACS)
- Software Development Kit (SDK)
- Agile Process Extensions (PX)
- Web Service Extensions (WSX)
- Oracle AIA Foundation Pack
- SAP

Agile Content Service

ACS 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, ACS can be configured to automatically to publish the Item Master and Bill of material (BoM) changes during any phase of the product lifecycle to multiple destinations, thereby ensuring that everyone is working with up-to-the-minute information.

The output generated by an ACS module is an Agile XML (aXML) file or a PDX package.

Software Development Kit

SDK contains a collection of tools; application programming interfaces (APIs), sample applications, and documentation. You use SDK to build custom applications that access the Agile application server functionality. By using Agile SDK, it’s possible to create programs that perform tasks automatically in Agile PLM.

Agile SDK enables the following operations:

- Integrate Agile PLM with SAP or other custom applications.
- Develop applications to process product data.
- Perform batch operations against the Agile Application Server.

Agile SDK has the following modules:

Agile API: A Java API with interfaces that expose Agile PLM business objects. Use Agile API to create additional Agile PLM clients. You can also use it as part of an extension developed using web service extensions (WSX) or process extensions (PX).

Process Extension (PX): A framework that allows you to extend the functionality of Agile PLM clients by adding external reports, user-driven and workflow-driven customized actions, customized tools, and customized automatic number sources. PX helps in binding the data in Agile PLM with other applications.

Note: The new part request is the only process integration that uses PX to extract the Agile PLM data and send it to the Application Business Connector Service (ABCS).

Web service Extension (WSX): A framework that allows you to extend the functionality of the Agile PLM server and expose customer-specific solutions using a web service. WSX is a Web service engine enabling communication between Agile PLM and disparate systems both internal and external, including ERP systems. WSX 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 simplifies the process for aggregating raw product content and making critical product content available in realtime to other core systems.

For more information about Agile PLM components, see the *Agile Product Lifecycle Management Administrator Guide* and the *SDK Developer Guide*.

Oracle AIA Foundation Pack

The Oracle AIA Foundation Pack allows you to simplify cross-application business process integrations using a standards-based, pre-built integration solution. Designed to promote rapid deployment, re-usability, and configurability of business processes, the Oracle AIA Foundation Pack saves significant time, effort and cost, compared to building integrations from the ground up. It also helps you realize the value of a Service Oriented Architecture (SOA) at an accelerated pace.

The Oracle AIA pre-built integrations are specific to business processes and applications. Built using the Oracle AIA Foundation Pack, these specific business process integrations provide an open, standards-based approach for organizations to integrate end-to-end business processes across a broad range of custom, Oracle or third-party applications.

For more information about the Oracle AIA Foundation Pack development methodology, see Oracle Fusion Middleware Developer's Guide for Oracle Application Integration Architecture Foundation**.**

SAP

SAP is an acronym for System Application and Products, which creates a common centralized database for all the applications running in an organization. SAP products are generally focused on ERP.

SAP applications are built to manage product operations, cost accounting, assets, materials and personnel. SAP ERP runs on a majority of platforms including Microsoft Windows and it uses client and server model. SAP provides a majority of new dimensional enterprise products namely Strategic Enterprise Management (SEM), Supplier Relationship Management (SRM), Customer Relationship Management (CRM), Advanced Planning and Optimizer (APO), and so on.

Chapter 2: Process Integration for Change Order Release

Integration of Change Order Release, namely the Engineering Change Order (ECO) and New Part/Product Release (PREL) from Agile to System Application and Products (SAP) is about introducing a new product, components, and structure into SAP upon the release of design from Agile or updating existing design metadata or structure. This process is triggered from Agile.

This chapter includes the following sections:

- Change Order Release Process
- Posting CO Data to SAP
- CO Release Process Integration Solution Assumptions
- CO Release Integration Sequence
- Oracle Application Integration Architecture (Oracle AIA) Services for CO Release
- Integration Services
- CO Release Integration Customization Points
- Properties and Domain Value Maps (DVMs)

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 and Bill of Material (BoM) information from Agile to SAP. 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 SAP for implementation. A confirmation of the implementation status is sent back to Agile.

This integration process supports the following flow:

1. Release of CO in Agile
2. Agile XML (aXML) generation by Agile Content Server (ACS)
3. Parsing and transformation of aXML data
4. Posting Changer Order data to SAP
5. Communicating order processing status to Agile

Release of CO in Agile

When a change analyst approves the CO in Agile, it is marked as *Released*. This makes all the changes specified in the CO take effect in Agile.

Before the CO is released, it is pre-validated at the approval stage. This involves validation of certain business rules to ascertain that the flow of the CO from Agile to SAP meets all the conditions set in the destination system.

aXML Generation by ACS

The ACS generates an aXML file from the CO data. This file contains information of items, BoM, manufacturers, as well as the CO itself.

For the purpose of the Agile to SAP integration process, the ACS is configured to ensure the following:

- 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.
- Upon release of a CO, the aXML file goes to a JMS Queue.

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. Therefore, this data must be parsed and transformed.

The parsing and transformation of aXML data entails the following:

1. Segregation of business objects
2. Sequencing and queuing of COs
3. Translating Agile *Site* specific objects into SAP *Plant* specific objects
4. Translating Agile CO Types into SAP CO Type.

   All the change types from Agile are mapped to the Engineering Change Master record in SAP.

5. Ascertaining the business objects existence from SAP to prepare the data in either *Create* or *Update* mode.
6. Mapping Agile CO attributes to corresponding attributes in SAP
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, and so on. This information is broken down into individual components and mapped one to one with corresponding Enterprise Business Objects (EBOs), namely *Item, Change* and *Structure*.

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

Sequencing and Queuing of COs

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

For more information about the sequencing and queuing of COs, see Appendix D, Queue Management Solution.

Translating Site Objects

The data coming from Agile is split into individual SAP organization specific business objects. This is because the data in Agile can be either:

- **Centralized** – all design locations share the same product design information
- **De-centralized** – the Item Attributes, change control, and so on are implemented to multiple sites.

The data in SAP is segregated by Plants.

Translating CO Types

In Agile, a CO is categorized into the following change types:

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

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

SAP does not have separate categories for each of these change types and cannot be differentiated. Therefore, during the parsing and processing of these change types in Agile, they are translated in SAP with the following characteristics:

Feature	ECO	MCO	SCO
Customer adoption	All installations	All installations	---
New Revisions for revised	Mandatory	Not supported	Not supported
Tables redlined	BOMs;	Global BOM as well as site-spec	Site-specific BOMs.

Feature	ECO	MCO	SCO
	Global as well as site-specific		
Effectivity Date	At line level; Global when multi-site is not en Separate for each site when m enabled	Not specified on Change Order	At line level; site-specific on
Other line-level attributes	Global when multi-site is not en Separate for each site when m enabled	Global when multi-site is not ena Separate for each site when mu enabled	Site-specific only
New Item Release	Supported	Supported	Only item updates are supp

Ascertaining Item Existence in SAP

An Item is created in the SAP *Plant* in two ways:

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

If the Item already exists in the SAP *Plant*, and Agile releases a CO to create the same Item, the system updates the item. Because Agile does not explicitly pass information about first time or subsequent release of Item; a cross-reference table is employed to ascertain the existence of the Item in SAP.

This cross-reference table maintains unique identifiers for the Items received from Agile as well as the corresponding Items created in SAP. It also maintains the unique identifiers for the Items created in SAP by an external application. These unique identifiers help in ascertaining the existence of an Item in SAP, thereby eliminating any duplicate errors.

Mapping CO Attributes

The source system attributes are mapped to destination system attributes. This mapping also defines the direction of the data flow. This is accomplished as follows:

1. Different sets of mappings are defined for:
   - Information flowing from Agile to Enterprise Business Objects (EBOs).
   - Iinformation flowing from EBOs to Agile.
2. These are accessed in the user interface (UI) as different ‘mapping profiles’.
3. 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.

For more information about Agile to SAP mappings, see Appendix C: Agile to SAP Entity Maps.

Defining User Exit Points

User Exits are provided in the integration to allow custom transformations or filtration routines that you 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.

Posting CO Data to SAP

Agile PLM records the design specifications of a part. This information is updated either on an ECO or a MCO. A new part is created in the ERP using reference material in SAP. Revisions to a part, effective dateBoM, item, and reference designator information are controlled by the ECO, which transfers the information from Agile to the ERP system.

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

The Process Queue Controller drives the integration of the change order release process between Agile PLM and the SAP system. To maintain the integrity of data in the ERP system, it is important that COs are transferred to that system in the same sequence they were released by the source system.

The Agile PLM to SAP process flow consists of two phases: *Pre-Process ECO* and *Post-Process ECO*:

Pre-Process ECO

Application Business Message (ABM) to Enterprise Business Message (EBM) transformations:

1. Invoke Provider
2. Receive Response
3. Send Response to the Queue

Post-Process ECO

- Update the transfer status in Agile PLM

CO Processing in SAP

The processing of CO data into the ERP system is the backbone of this integration. As part of this step, the following activities must be performed:

- **Item Master synchronization**: For all items pushed to the ERP system, verify whether or not the items already exist. If the item does not exist in ERP then create it. If the Item already exists in ERP, then update the Item. It is assumed that the source and target systems are in sync before the integration begins so that it is not necessary for the integration to check if both the systems are in sync before posting the data.
- **Create** Change Order: The actual CO is created as *Change Master Object* in the ERP system. When the Change Master record is posted to SAP, the status is set to *Inactive*. When all the line items on this change are posted successfully to SAP then the integration changes the

Change Master status in SAP to *Active*. This means this Change Master is complete and all the changes effective on this change are in-force from scheduled effectivity and valid from date. This *Active* CO status from SAP is transferred to the corresponding mapped Page Two field in Agile as *Transferred*.

- **Item Plant Assignment**: Items are assigned to plants based on criteria specified in the section on supporting distributed manufacturing above.
- **BoM Update** (including reference designator): The aXML data from Agile contains only the changes made to BoMl and not the complete BoM. As a result, BOM data must be in sync between Agile and ERP for the older revision in order for the new revision of BOM data to be posted successfully.

Communicating CO Implementation status to Agile

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

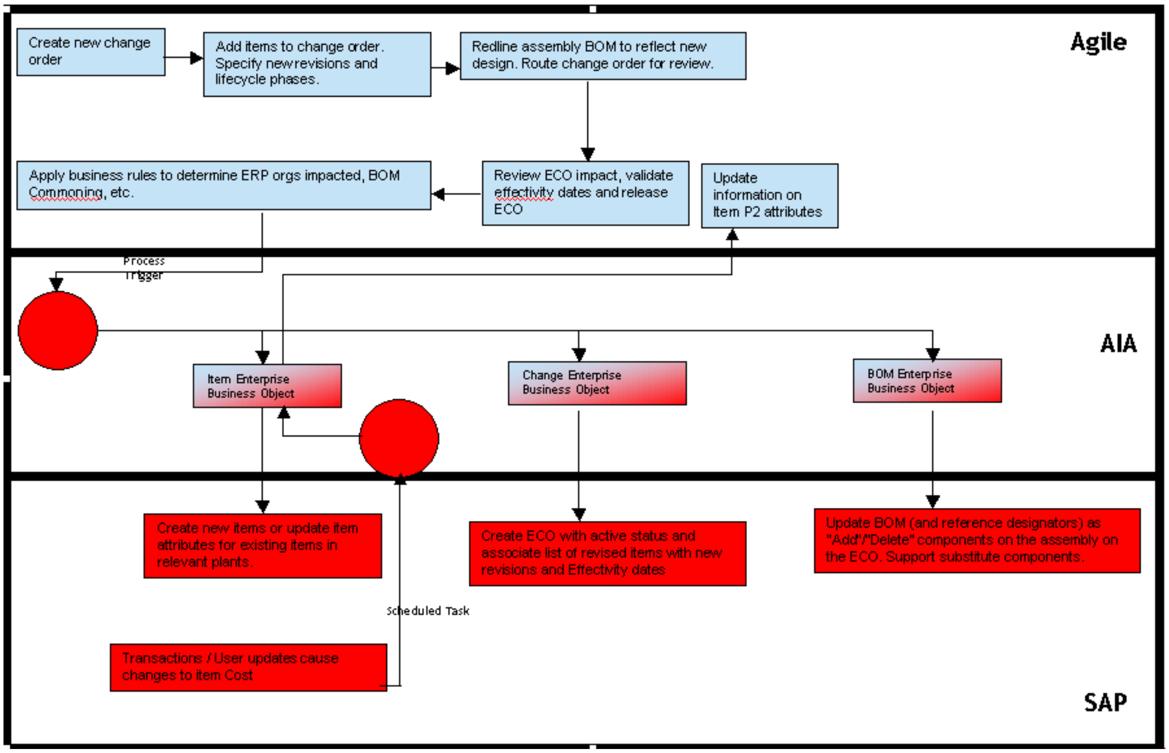
If the parameter **Change.TransferStatusAttribute** is assigned to Page Two field in Agile, then upon successful creation of the CO in SAP with status *Active* it is propagated back to Agile as *Transferred* in the specified Page Two field, else *Errored* status is propagated.

CO Release Process Integration Solution Assumptions

1. COs must be transferred to ERP in the order in which they were released by the source system.
2. If a part does not exist in ERP, then create the part.
3. If a part exists in ERP system, then update the part.
4. Design changes, Effectivity Date, BoM, Item, and Reference Designators are all controlled by the ECO.
5. Part information and design specifications are updated from Agile to ERP system using the ECO.
6. There is a business process that prioritizes and monitors the Change Order Process.
7. Only one CO (first one in the queue) undergoes ERP Processing at a time. The next process can start its execution only after the execution of the first process is completed.
8. The integration administrator has the provision of error handling and reporting the same to the end user.

CO Release Integration Sequence

The integration flow is same for both ECO and PREL:


Agile
Create new change order
Add items to change order. Specify new revisions and lifecycle phases.
Redline assembly BOM to reflect new design. Route change order for review.
Apply business rules to determine ERP orgs impacted, BOM Commoning, etc.
Review ECO impact, validate effectivity dates and release ECO
Update information on Item P2 attributes
Process Trigger
AIA
Item Enterprise Business Object
Change Enterprise Business Object
BOM Enterprise Business Object
Create new items or update item attributes for existing items in relevant plants.
Create ECO with active status and associate list of revised items with new revisions and Effectivity dates
Update BOM (and reference designators) as "Add"/"Delete" components on the assembly on the ECO. Support substitute components
Scheduled Task
Transactions / User updates cause changes to item Cost
SAP

CO Release Integration Flow

An ECO is created with items in Affected Items (AI) tab with new revisions and lifecycle phases specified. The ECO is routed for Approval (workflow step).

1. An ACS Workflow Event is generated on Approval of the ECO in Agile to trigger the ECO process flow.

2. The Queue framework captures the ACS payload (aXML) generated for the event and adds it to the integration Queue.

3. The Queue framework identifies the highest priority Queue Message, processes it to create an ECO ABM and triggers the Requestor ABCS.

4. The Requestor ABCS transforms the ECO ABM to ECO EBM and triggers an operation on the Enterprise Business Service (EBS), which routes the EBM to the ECO business flow with ECO EBM as the input.

5. The ECO Business Flow:

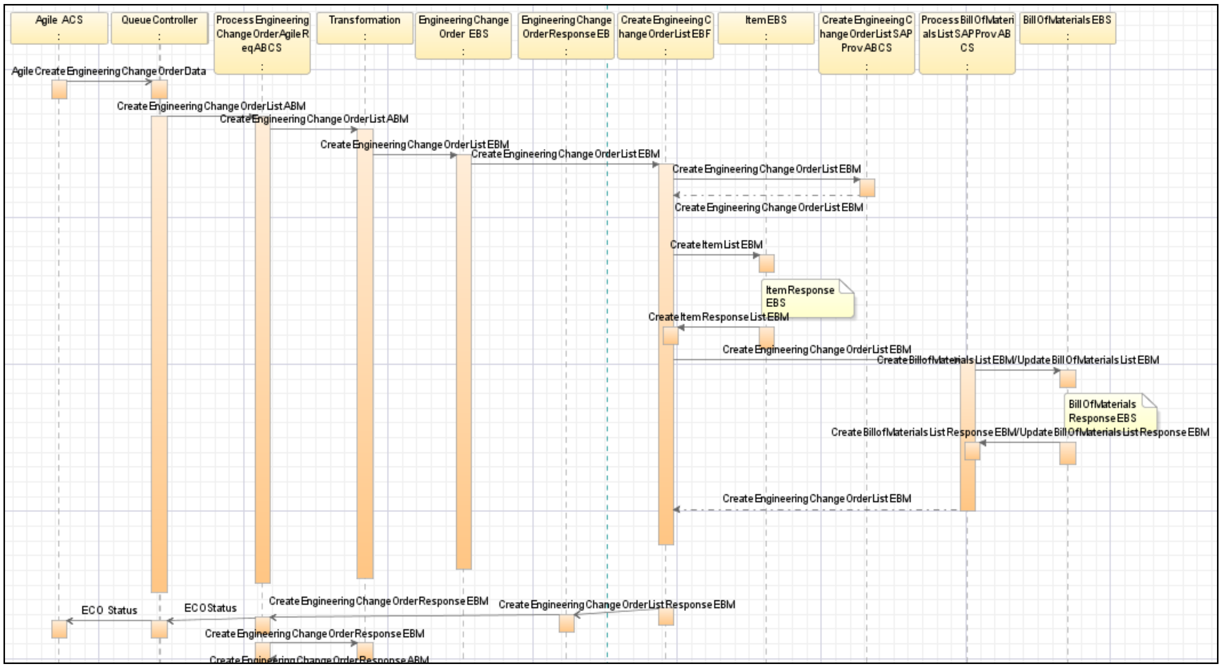
   - Creates new items in ERP
   - Creates an ECO in the SAP.
   - Associates the list of revised items with new revisions and effectivity dates, and schedules the ECO for effectivity date.
   - Creates new BoM
   - Updates existing Item in SAP
   - Updates existing BoM in SAP

- Updates the Transfer Status in Agile.

6. The status of Queue Message is updated in the Integration Queue, for monitoring.

CO Release Services Orchestration

The below diagram illustrates the Change Order Services orchestration from Agile Application to SAP ERP.


Agile ACS
Queue Controller
Process Engineering Change Order Agile Req ABCS
Transformation
Engineering Change Order EBS
Engineering Change Order Response EB
Create Engineeing Change Order List EBF
Item EBS
Create Engineeing Change Order List SAP Prov ABCS
Process Bill Of Materials List SAP Prov ABCS
Bill Of Materials EBS
Agile Create Engineering Change Order Data
Create Engineering Change Order List ABM
Create Engineering Change Order List ABM
Create Engineering Change Order List EBM
Create Engineering Change Order List EBM
Create Engineering Change Order List EBM
Create Engineering Change Order List EBM
Create Item List EBM
Item Response EBS
Create Item Response List EBM
Create Engineering Change Order List EBM
Create Billof Materials List EBM/Update Bill Of Materials List EBM
Bill Of Materials Response EBS
Create Billof Materials List Response EBM/Update Bill Of Materials List Response EBM
Create Engineering Change Order List EBM
ECO Status
ECO Status
Create Engineering Change Order Response EBM
Create Engineering Change Order List Response EBM
Create Engineering Change Order Response EBM
Create Engineering Change Order Response ABM

This table list the activities involved in the Change Order Release Services orchestration:

#	Activity	Remarks
1	Agile ACS acts as a trigger for ECO Use case.	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.
2	QueueController processes the payload	The Queue Controller Framework reads the highest priority Queue Message and transforms the payload (aXML) to AgileCreateEngineeringChangeOrderListABM.
3	Invoke ProcessEngineeringChangeOrderAgileReqABCS	QueueController invokes the ProcessEngineeringChangeOrderAgileReqABCS with AgileCreateEngineeringChangeOrderListABM as input.

#	Activity	Remarks
4	ProcessEngineeringChangeOrderAgileReqABCS makes call back to Agile Web services, if needed.	ProcessEngineeringChangeOrderAgileReqABCS calls the web services exposed in agile side to enrich AgileCreateEngineeringChangeOrderListABM. AgileCreateEngineeringChangeOrderListABM is transformed into CreateEngineeringChangeOrderEBM.
5	ProcessEngineeringChangeOrderAgileReqABCS invokes EngineeringChangeOrderEBS	ProcessEngineeringChangeOrderAgileReqABCS invokes CreateEngineeringChangeOrder operation on EngineeringChangeOrderEBS with CreateEngineeringChangeOrderEBM as input
6	EngineeringChangeOrderEBS routes CreateEngineeringChangeOrderListEBM to CreateEngineeringChangeOrderListEBF	CreateEngineeringChangeOrderListEBM is passed to CreateEngineeringChangeOrderListEBF as input.
7	CreateEngineeringChangeOrderEBF invokes ItemEBS	CreateItemListEBM is passed to ItemEBS as input
8	ItemEBS routes the CreateItemListEBM to SyncItemListSAPProvABCS	CreateItemListEBM is passed to SyncItemListSAPProvABCS as input
9	SyncItemListSAPProvABCS transforms the input and calls the SAP service	SyncItemListSAPProvABCS transforms CreateItemListEBM to the input of SAP service and calls that service.Creates items in ERP
10	SyncItemListSAPProvABCS calls ItemResponseEBS (Response Port) with CreateItemListResponseEBM	SyncItemListSAPProvABCS invokes the CreateItemListResponse operation on ItemResponseEBS with CreateItemListResponseEBM as input
11	CreateEngineeringChangeOrderListEBF calls ProcessBillOfMaterialsListSAPProvABCS	CreateEngineeringChangeOrderListEBM will be passed to ProcessBillOfMaterialsListSAPProvABCS as the input
12	ProcessBillOfMaterialsListSAPProvABCS invokes BillOfMaterialsEBS	
13	Based on the existence of the BOM BillOfMaterialsEBS routes to CreateBillOfmaterialsListSAPProvABCSImpl (or) UpdateBillOfMaterialsListSAPProvABCSImpl	
13 a)	BillOfMaterialsEBS calls CreateBillOfMaterialsListSAPProvABCSImpl	CreateBillOfMaterialsListEBM is passed to CreateBillOfmaterialsListSAPProvABCSImpl
13 b)	ProcessBillOfMaterialsListSAPProvABCSImpl is invoked before BillOfMaterialsEBS calls UpdateBillOfMaterialsListSAPProvABCSImpl	CreateEngineeringChangeOrderListEBM is passed to ProcessBillOfMaterialsListSAPProvABCSImpl & after this UpdateBillOfMaterialsListEBM is passed to UpdateBillOfMaterialsListSAPProvABCSImpl
14 a)	CreateBillOfMaterialsListSAPProvABCSImpl transforms the input and calls the SAP service	CreateBillOfMaterialsListSAPProvABCSImpl will transform CreateBillOfMaterialsListEBM to the input of SAP service and calls that service.

#	Activity	Remarks
		Creates the BillOfMaterialsList in ERP
14 b-i)	ProcessBillOfMaterialsListSAPProvABCSImpl transforms the input and calls the SAP service	ProcessBillOfMaterialsListSAPProvABCSImpl transforms CreateEngineeringChangeOrderListEBM to the input of SAP service and calls that service
14 b.ii)	UpdateBillOfMaterialsListSAPProvABCSImpl transforms the input and calls the SAP service	UpdateBillOfMaterialsListSAPProvABCSImpl transforms UpdateBillOfMaterialsListEBM to the input of SAP service and calls that service. Updates the BillOfMaterialsList in ERP
15 a)	CreateBillOfMaterialsListSAPProvABCSImpl calls BillOfMaterialsResponseEBS (Response Port) with CreateBillOfMaterialsListResponseEBM	CreateBillOfMaterialsListSAPProvABCSImpl invokes CreateBillOfMaterialsListResponse operation on BillOfMaterialsResponseEBS with CreateBillOfMaterialsrListResponseEBM as input
15 b)	UpdateBillOfMaterialsListSAPProvABCSImpl calls BillOfMaterialsResponseEBS (Response Port) with UpdateBillOfMaterialsListResponseEBM	UpdateBillOfMaterialsListSAPProvABCSImpl invokes UpdateBillOfMaterialsListResponse operation on BillOfMaterialsEBS with UpdateBillOfMaterialsListResponseEBM as input
16	CreateEngineeringChangeOrderEBF calls CreateEngineeringChangeOrderSAPProvABCS	CreateEngineeringChangeOrdeListEBM is passed to CreateEngineeringChangeOrderListSAPProvABCS as input
17	CreateEngineeringChangeOrderListSAPProvABCS transforms the input and calls the SAP service	CreateEngineeringChangeOrderListSAPProvABCS transforms CreateEngineeringChangeOrderListEBM into the input of SAP Service and calls that service. Creates items in ERP, creates an ECO, associates revised items to it and creates BOM
18	CreateEngineeringChangeOrderListEBF calls EngineeringChangeOrderResponseEBS (Response Port) with the CreateEngineeringChangeOrderResponseEBM	CreateEngineeringChangeOrderListEBF invokes CreateEngineeringChangeOrderListResponse operation on EngineeringChangeOrderResponseEBS with CreateEngineeringChangeOrderListResponseEBM as input.
19	EngineeringChangeOrderResponseEBS routes the response message to ProcessEngineeringChangeOrderAgileReqABCS.	EngineeringChangeOrderResponseEBS *routes* CreateEngineeringChangeOrderListResponseEBM *to* ProcessEngineeringChangeOrderAgileReqABCS*.*
20	ProcessEngineeringChangeOrderAgileReqABCS sends the status back to the QueueController to update the queue.	ProcessEngineeringChangeOrderAgileReqABCS replies back to the QueueController with the status of the ECO business process. This status is updated against the Queue message in the database by the QueueController
21	ProcessEngineeringChangeOrderAgileReqABCS invokes the Agile Web Services.	ProcessEngineeringChangeOrderAgileReqABCS transforms CreateEngineeringChangeOrderListResponseEBM into AgileUpdateEngineeringChangeOrderListABM.

#	Activity	Remarks
		AgileUpdateEngineeringChangeOrderListABM is sent as an input to the Agile Web Service. The web services would update transfer status on the Change Order in Agile which will be predefined P2 or P3 attributes on ECO object in Agile. AgileUpdateEngineeringChangeOrderListResponseABM is sent back to ProcessEngineeringChangeOrderAgileReqABCS.

Oracle AIA Services for CO Release

The process integration for ECO and PREL uses the following industry components:

EBOs	1. EngineeringChangeOrderEBO 2. BillOfMaterialsEBO 3. ItemEBO
EBMs	4. CreateEngineeringChangeOrderListEBM 5. CreateEngineeringChangeOrderListResponseEBM 6. CreateBillOfMaterialsListEBM 7. CreateBillOfMaterialsListResponseEBM 8. UpdateBillOfMaterialsListEBM 9. UpdateBillOfMaterialsListResponseEBM 10. CreateItemListEBM 11. CreateItemListResponseEBM
EBSs	12. EngineeringChangeOrderListEBS (CreateEngineeringChangeOrderListResponseEBM) 13. BillOfMaterialsEBS (CreateBillOfMaterialsListEBM/ CreateBillOfMaterialsListResponseEBM) 14. ItemEBS (CreateItemListEBM/ CreateItemListResponseEBM)

Core Components Locations

EBO & EBM XSD files	$AIA_HOME/AIAMetaData/AIAComponents/EnterpriseObjectLibrary/Core/EBO/EngineeringChangeOrder/ $AIA_HOME/AIAMetaData/AIAComponents/EnterpriseObjectLibrary/Core/EBO/BillOfMaterials/ $AIA_HOME/AIAMetaData/AIAComponents/EnterpriseObjectLibrary/Core/EBO/Item/
WSDL files	$AIA_HOME/AIAMetaData/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.

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 & SAP Components for CO Release

Services	Agile (Requester)	SAP (Provider)
ABMs	AgileCreateEngineeringChangeOrderListABM AgileCreateEngineeringChangeOrderListResponseABM AgileUpdateEngineeringChangeOrderListABM AgileUpdateEngineeringChangeOrderListResponseABM	BAPI_MATERIAL_GET_ALL BAPI_MATERIAL_GET_ALLResponse BAPI_MATERIAL_GET_DETAIL BAPI_MATERIAL_GET_DETAILResponse BAPI_MATERIAL_SAVEDATA BAPI_MATERIAL_SAVEDATAResponse CSAP_MAT_BOM_READ CSAP_MAT_BOM_READResponse CCAP_REV_LEVEL_MAINTAIN CCAP_REV_LEVEL_MAINTAINResponse BAPI_MAT_BOM_EXISTENCE_CHECK BAPI_MAT_BOM_EXISTENCE_CHECKResponse CSAP_MAT_BOM_ALLOC_CREATE CSAP_MAT_BOM_ALLOC_CREATEResponse CSAP_MAT_BOM_CREATE CSAP_MAT_BOM_CREATEResponse CSAP_MAT_BOM_MAINTAIN CSAP_MAT_BOM_MAINTAINResponse CCAP_ECN_MAINTAIN CCAP_ECN_MAINTAINResponse
ABCS	ProcessEngineeringChangeOrderAgileReqABCS	CreateEngineeringChangeOrderListSAPProvABCSImpl CreateEngineeringChangeOrderListEBF ProcessBillOfMaterialsListSAPProvABCS ProcessBillOfMaterialsListSAPProvABCSImpl CreateBillOfMaterialsListSAPProvABCSImpl UpdateBillOfMaterialsListSAPProvABCSImpl SyncItemListSAPProvABCS
EBS	EngineeringChangeOrderEBS (CreateEngineeringChangeOrderList Operation)	EngineeringChangeOrderResponseEBS (CreateEngineeringChangeOrderListResponse Operation) BillOfMaterialsEBS BillOfMaterialsResponseEBS ItemEBS ItemResponseEBS
BPEL	CreateQueueService QueueProcessorServiceImpl	---

Services	Agile (Requester)	SAP (Provider)
Mediator Service	ACSAXMLJMSConsumer ACSAXMLFileConsumer CreateQueueControlService QueueProcessorService EngineeringChangeOrderService	ItemService BOMServiceESB ReserveItemservice ECORouterService

Component Locations

ABO XSD files	$AIA_HOME/AIAMetaData/AIAComponents/ApplicationObjectLibrary/Agile/V1/schemas
ABM XSD files	$AIA_HOME/AIAMetaData/AIAComponents/ApplicationObjectLibrary/Agile/ V1/schemas $AIA_HOME/AIAMetaData/AIAComponents/ApplicationObjectLibrary/SAP/Release1/Core/ABM/ EngineeringChangeOrder

Integration Services

These are the integration services for Change Order release:

EngineeringChangeOrderEBS

EngineeringChangeOrderEBS is the EBS, which exposes the operations related to the Engineering Change Order Integration on the EngineeringChangeOrder EBO. The following are the routing rules:

EngineeringChangeOrderEBS Mediator service

- ProcessEngineeringChangeOrderAgileReqABCS
  - Route CreateEngineeringChangeOrderListEBM to CreateEngineeringChangeOrderListSAPProvABCSImpl

EngineeringChangeOrderResponseEBS Mediator Service

- CreateEngineeringChangeOrderListSAPProvABCSImpl
  - Route CreateEngineeringChangeOrderListResponseEBM to ProcessEngineeringChangeOrderAgileReqABCS

ItemEBS

ItemEBS Mediator Service

- CreateEngineeringChangeOrderListEBF

  - Route CreateItemListEBM to SyncItemListSAPProvABCS

ItemResponseEBS Mediator service

- SyncItemListSAPProvABCS
  - Route CreateItemListResponseEBM to CreateEngineeringChangeOrderListEBF

BillOfMaterialsEBS

BillOfMaterialsEBS Mediator service

- ProcessBillOfMaterialsListSAPProvABCS
  - Route CreateBillOfMaterialsListEBM to CreateBillOfMaterialsListSAPProvABCSImpl
  - Route UpdateBillOfMaterialsListEBM to UpdateBillOfMaterialsListSAPProvABCSImpl

BillOfMaterialsResponseEBS Mediator service

- CreateBillOfMaterialsListSAPProvABCSImpl
  - Route CreateBillOfMaterialsListResponseEBM to ProcessBillOfMaterialsListSAPProvABCS
- UpdateBillOfMaterialsListSAPProvABCSImpl
  - Route UpdateBillOfMaterialsListResponseEBM to ProcessBillOfMaterialsListSAPProvABCS

ProcessEngineeringChangeOrderAgileReqABCS

ProcessEngineeringChangeOrderAgileReqABCS is used for transforming AgileCreateEngineeringChangeOrderListABM into CreateEngineeringChangeOrderListEBM. This service invokes the CreateEngineeringChangeOrder operation on EngineeringChangeOrderEBS for creation of the ECO in SAP.

Based on the status of ECO creation in SAP this service updates the queue status. Also the Transfer status attribute in CO is updated by this service. This service is implemented as *Asynchronous* Business Process Execution Language (BPEL) composite.

The QueueController creates the AgileCreateEngineeringChangeOrderListABM and invokes the ProcessEngineeringChangeOrderAgileReqABCS.

1. Transforms the AgileCreateEngineeringChangeOrderListABM to CreateEngineeringChangeOrderListEBM and invokes the CreateEngineeringChangeOrder operation on EngineeringChangeOrderEBS with CreateEngineeringChangeOrderListEBM as input. This is routed for creation of the ECO in SAP.
2. CreateEngineeringChangeOrderListResponseEBM is received from EngineeringChangeOrderEBS and based on the status of ECO creation in SAP; the QueueController is invoked to update the status of the Queue Message.

3. ProcessEngineeringChangeOrderAgileReqABCS transforms CreateEngineeringChangeOrderListResponseEBM into AgileUpdateEngineeringChangeOrderListABM and is sent as an input to the Agile Web Service.
4. The web services updates the transfer status of the CO in Agile, which is predefined P2 or P3 attributes on the ECO object in Agile. AgileUpdateEngineeringChangeOrderListResponseABM is sent back to ProcessEngineeringChangeOrderAgileReqABCS.

Transformations

AgileCreateEngineeringChangeOrderListABM_to_CreateEngineeringChangeOrderEBM.xsl

1. Transforms AgileCreateEngineeringChangeOrderListABM to CreateEngineeringChangeOrderListEBM
2. CreateEngineeringChangeOrderResponseEBM_to_AgileUpdateEngineeringChangeOrderListABM.xsl
3. Transforms CreateEngineeringChangeOrderResponseEBM to AgileUpdateEngineeringChangeOrderListABM

CreateEngineeringChangeOrderListSAPProvABCSImpl

This is a single operation service. This accepts an ECO containing Item and BoMs information message as a request and returns a response.

1. In Agile to SAP flow, CreateEngineeringChangeOrderListSAPProvABCSImpl is used for transforming CreateEngineeringChangeOrderListEBM into CCAP_ECN_MAINTAINABM, which invokes the CreateEngineeringChangeOrder operation in SAP.
2. In return flow, CCAP_ECN_MAINTAIN BAPI sends CCAP_ECN_MAINTAINResponse which is transformed by CreateEngineeringChangeOrderListSAPProvABCSImpl into CreateEngineeringChangeOrderListResponseEBM.

This service is implemented as an asynchronous BPEL Composite

CO Release Integration Customization Points

This table lists the CO integration customization information:

Agile

Name	Property	Description
ProcessEngineeringChangeOrderAgileReqABCS (Agile Process ECO requestor flow)	AgileCreateEngineeringChangeOrderListABM_to_CreateEngineeringChangeOrderListEBM_ Impl .xsl	ReqABM to ReqEBM (main)
	AgileCreateEngineeringChangeOrderListABM_to_CreateEngineeringChangeOrderListEBM_ Custom.xsl	ReqABM to ReqEBM (custom elements)

Name	Property	Description
	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)

SAP

Name	Property	Description
CreateEngineeringChangeOrderListSAPProvABCSImpl CreateEngineeringChangeOrderListEBF	CreateEngineeringChangeOrderListEBMtoABM_Custom.xsl	Custom transformations for Engineering Change Order Request EBM to Request ABM
	XformReqEBMToResEBM_Custom.xsl	Custom transformations for Engineering Change Order Request EBM to Engineering Change Order Response EBM
	CreateECOEBMtoItemListEBM_Custom.xsl	Custom transformations for Engineering Change Order Request EBM to Item List EBM
	UpdateXformEngineeringChangeOrderListEBMtoEBM_Custom.xsl	Custom transformations for update Engineering Change Order list EBM to Engineering Change Order list EBM
	XformEBMtoBillOfMaterialProcess_Custom.xsl	Custom transformations for Engineering Change Order list EBM to Engineering Change Order list EBM for BOM
	XformECOListEBMtoECOListResponseEBM_Custom.xsl	Custom transformations for Engineering Change Order list request EBM to Engineering Change Order list Response EBM
	XformEngineeringChangeOrderListEBMtoEBM_Custom.xsl	Custom transformations for create Engineering Change Order list EBM to Engineering Change Order list EBM

Name	Property	Description
	ECOEBMToExistenceCheck_Custom.xsl	Custom transformations for Engineering Change Order list EBM to BOM_Existence_Check ABM
ProcessBillOfMaterialsListSAPProvABCS	ECOEBMToGlobalBOMExistenceCheck_Custom.xsl	Custom transformations for Engineering Change Order list EBM to global BOM_Existence_Check ABM
	ECOEBMToGlobalUpdateBOMEBM_Custom.xsl	Custom transformations for Engineering Change Order list EBM to global Update BillOfMaterials List EBM
	ECOListEBMToBOMListEBM_Custom.xsl	Custom transformations for Engineering Change Order list EBM to Create BillOfmaterials List EBM
	ECOListEBMToUpdateBOMListEBM_Custom.xsl	Custom transformations for Engineering Change Order list EBM to Update BillOfMaterials List EBM
	XformEBMHeadertoECOListResponseEBM_Custom.xsl	Custom transformations for EBM Header to Create EngineeringChangeOrder List EBM
	XformECOListEBMtoEBMHeader_Custom.xsl	Custom transformations for Create EngineeringChangeOrder List EBM to EBM Header
CreateBillOfMaterialsListSAPProvABCSImpl	BOMEBMToCreateGlobalBOMABM_Custom.xsl	Custom transformations for Create BillOfMaterials List EBM to Create Global BOM ABM
	BOMEBMToExistenceCheck_Custom.xsl	Custom transformations for Create BillOfMaterials List EBM to global BOM Existence Check ABM
	BOMEBMToPlantExistenceCheck_Custom.xsl	Custom transformations for Create BillOfMaterials List EBM to BOM Existence Check ABM
	BOMEBMToPlantExtensionABM_Custom.xsl	Custom transformations for Create BillOfMaterials List EBM to BOM Plant Extension ABM
	BillOfMaterialsListEBMToCreateBOM_Custom.xsl	Custom transformations for Create BillOfMaterials List EBM to Create BOM ABM
	XformBOMReqEBMToResEBM_Custom.xsl	Custom transformations for Create BillOfMaterials List EBM to Create BillOfMaterials List Response EBM

Name	Property	Description
UpdateBillOfMaterialsListSAPProvABCSImpl	BOMEBMToGlobalBOMABM_Custom.xsl	Custom transformations for Update BillOfMaterials List EBM to Update global BOM ABM
	BOMEBMToPlantExistenceCheck_Custom.xsl	Custom transformations for Update BillOfMaterials List EBM to BOM Existence Check ABM
	BOMEBMToPlantExtensionABM_Custom.xsl	Custom transformations for Update BillOfMaterials List EBM to BOM Plant Extension ABM
	BOMEBMToUpdateBOMABM_Custom.xsl	Custom transformations for Update BillOfMaterials List EBM to Update BOM ABM
	XformBOMReqEBMToResEBM_Custom.xsl	Custom transformations for Update BillOfMaterials List EBM to Update BillOfMaterials List Response EBM
ProcessBillOfMaterialsListSAPProvABCSImpl	XformECOEBMtoABM.xsl	Custom transformations for CreateEngineeringChangeOrderListEBM to Read BOM ABM
	XformABMtoECOEBM.xsl	Custom transformations for Read BOM ResponseABM to CreateEngineeringChangeOrderList EBM
SyncItemListSAPProvABCS	XFormItemEBMToGetDetailRefMaterial_Custom.xsl	Custom transformations for CreateItem List EBM to Get detail Reference Material ABM
	XFormItemEBMToReferenceMaterial_Custom.xsl	Custom transformations for CreateItem List EBM to Get Reference Material ABM
	XFormItemEBMToRevisionABM_Custom.xsl	Custom transformations for CreateItem List EBM to Create Revision ABM
	XFormItemListEBMToSavedData_Custom.xsl	Custom transformations for CreateItem List EBM to Create Item List ABM
	XFormItemListEBMToUpdateItem_Custom.xsl	Custom transformations for CreateItem List EBM to Update Item list ABM(for Get All ReferenceMaterial-4.6)

Name	Property	Description
	XFormItemListEBMToUpdate_Item_Custom.xsl	Custom transformations for CreateItem List EBM to Update Item list ABM(for Get Detail ReferenceMaterial)
	XformItemListEBMtoItemListResponseEBM_Custom.xsl	Custom transformations for CreateItem List EBM to Create ItemList Response EBM

Properties and DVMs

The following mandatory DVMs and Properties (in AIAConfigProperties.xml) must be set for the CO Release process:

- AGILE_SITE_TARGET_MAPPING
- AGILE_TARGET_SITE_MAPPING
- LANGUAGE_CODE
- ECO_ENGINEERINGCHANGEORDERLINE_REVISEDBILLOFMATERIALS_BILLOFMATERIALSCOMPONENTITEM_CHANGETYPECODE
- ITEM_PRIMARYCLASSIFICATIONCODE
- ITEM_STATUS_CODE
- ITEM_UOM_CODE
- ECO_STATUS_CODE
- REFERENCE MATERIALS

Chapter 3: Process Integration for New Part Request

This chapter provides an overview about new part numbers and describes the new part request process in Agile and SAP, discusses integration solution assumptions, integration sequence of events, Oracle Application Integration Architecture (Oracle AIA) services, and integration customization points.

This chapter includes the following sections:

- Overview
- NPR Process in Agile
- NPR Process in SAP
- NPR Process Integration Solution Assumptions
- NPR Integration Sequence
- Oracle AIA Services for NPR
- NPR Integration Customization Points

Overview

New part numbers may originate in a system outside of Agile, which is easily understood when looking at the interactions of various roles that can be involved in the process of generating new part numbers.

An engineer designing an electronic assembly, a circuit board for example, may require a part, such as a new resistor, which cannot be found in the system. The engineer raises a new part request (NPR) to document control to route the request to the materials manager for review. A buyer looks into catalogs offered by approved manufacturers and finds a part that meets the criteria. The buyer then contacts the manufacturer, confirms availability, procures specifications, and then approves the NPR with the new manufacturer part number attached to it. Document control compiles all the information and assigns an internal part number corresponding to the manufacturer part and informs the engineer.

The process integration allows you to trigger a NPR process on demand. This is a synchronous process, which reserves and fetches the part number from the SAP system.

NPR Process in Agile

A New Part Request process in Agile is executed using an object of the Engineering Change Requests (ECR) class. Since the part number(s) being requested do not exist yet in Agile, there are no affected items on this ECR. A number of variants of this process are possible:

- A single item number is reserved in SAP per ECR

- Multiple item numbers are reserved in SAP per ECR

On Demand New Part Request

The synchronous on-demand NPR process allows you to achieve the following:

A new part number is required, which is to be used on a design. In Agile, the user clicks on the **New Object** icon to create a new item for a given subclass such as *Part* or *Document*. The on-demand NPR process compliments or replaces the new item creation process in Agile by providing the user a Process Extension (PX) that a user can use to trigger the creation of a new item.

The overall process flow is as follows:

1. User calls a PX from Agile's auto number menu on the **Create New Item** popup. This option is available whenever the part number generation process is generic (mainly using a sequence).
2. Since a SAP material type is required to get the next available sequence number from SAP, the item type (subclass) is passed as a parameter as part of this process. In DVM (Domain Value Mapping) the Agile's subclass is mapped to the corresponding SAP material type.
3. The PX requests a new part number from SAP. This process reserves the next available part number in SAP.
4. New reserved part number is sent back to Agile.
5. The new part number is returned to the calling PX and the Agile New Part Creation process is executed as usual from there onwards. If an error is encountered during PX execution, then the error message is displayed as a popup window in Agile.

For detail error information, log on to Business Process Execution Language (BPEL) console.

NPR Process in SAP

It is assumed that the internal number range assignments are configured in SAP for those *Material* types that are mapped to the corresponding *Part* or *Document* subclass of Agile, which requires auto number generation.

When the synchronous auto number PX is executed from Agile, it calls a standard SAP application programming language (API) with input as *Material* type.

SAP checks if there is a valid internal number range defined for the provided *Material* type. If yes, the program returns the next available number to the calling program. If no valid internal number range is defined, it returns an error to the calling program.

This complete process is synchronous; therefore, the user does not experience the SAP processes at all as they run in the background to return the SAP assigned number. For the user, the process appears similar to the auto number assignment within Agile.

NPR Process Integration Solution Assumptions

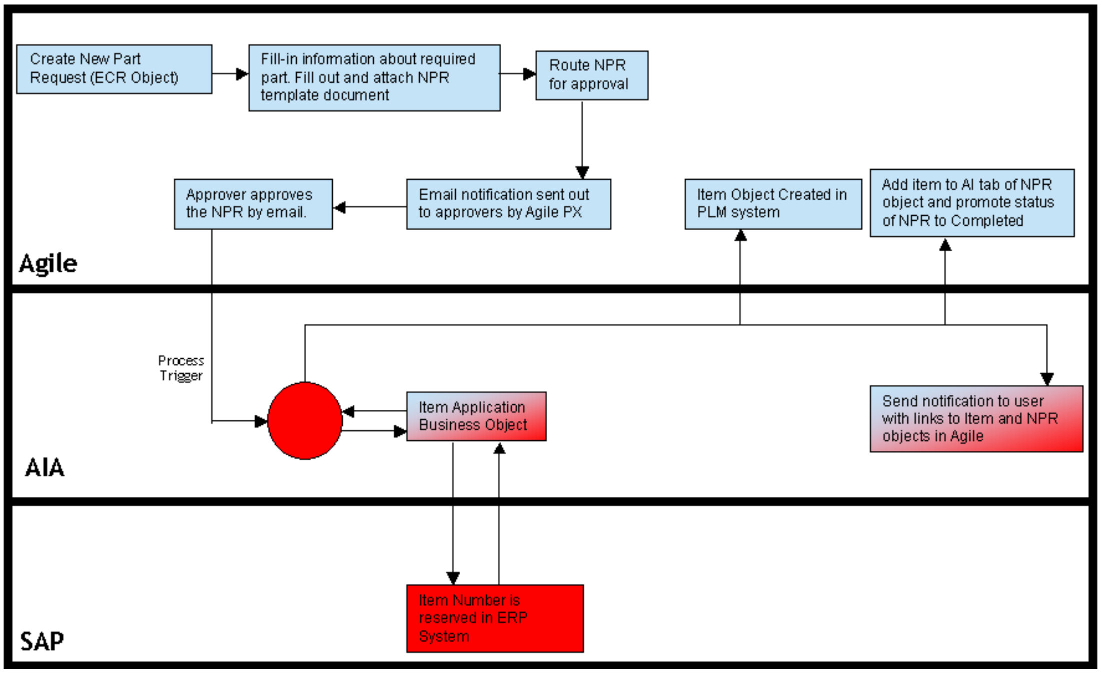
The following are the assumptions for the NPR process:

1. The *Item* type or *Agile* subclass type is mapped to *Material* type (MARA-MTART) in SAP.

2. For a given valid *Material* type, internal number range assignment is already configured in SAP.

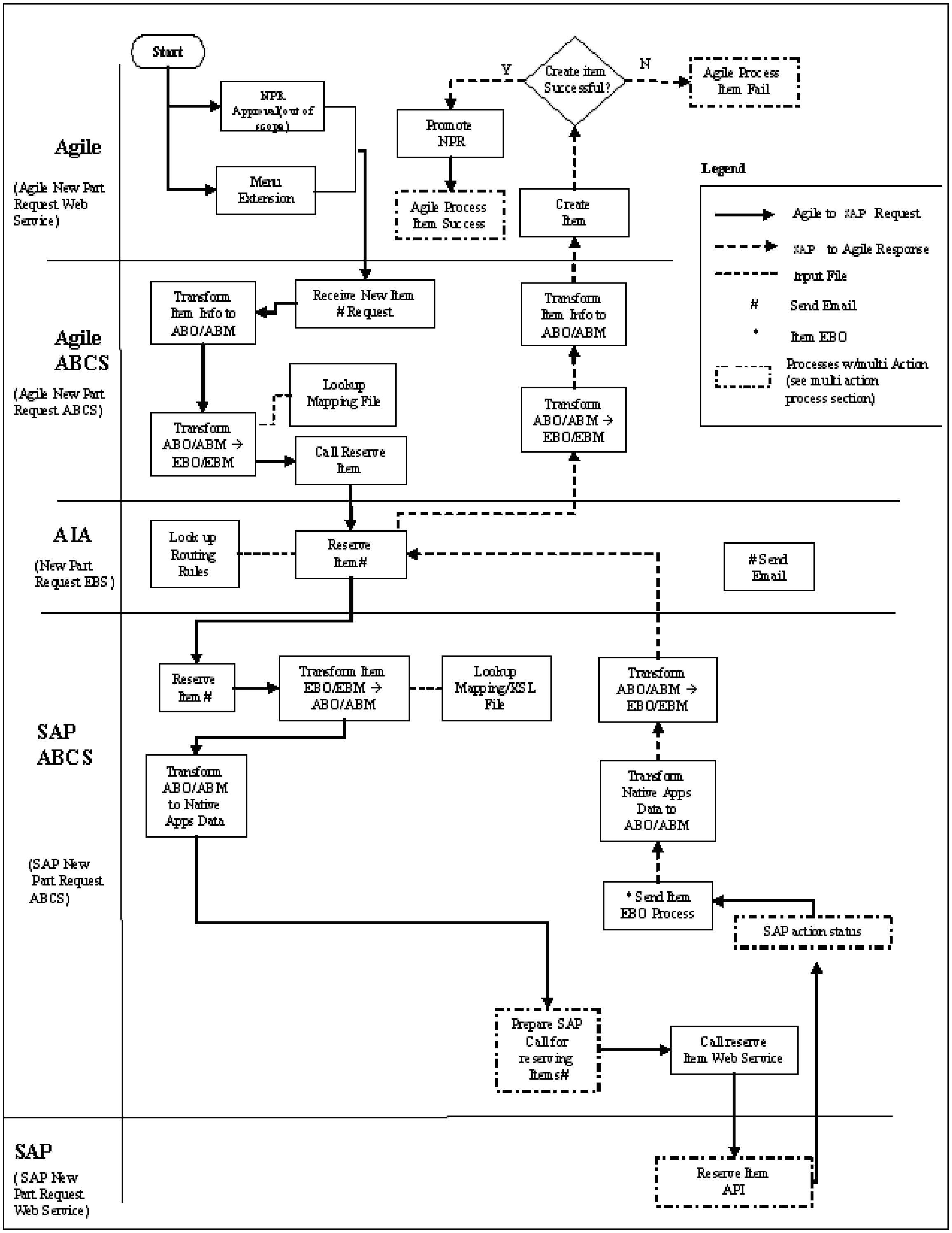
NPR Integration Sequence

This figure shows the process flow from Agile to SAP for the NPR process.


Create New Part Request (ECR Object)
Fill-in information about required part. Fill out and attach NPR template document
Route NPR for approval
Approver approves the NPR by email.
Email notification sent out to approvers by Agile PX
Item Object Created in PLM system
Add item to AI tab of NPR object and promote status of NPR to Completed
Agile
Process Trigger
Item Application Business Object
Send notification to user with links to Item and NPR objects in Agile
AIA
Item Number is reserved in ERP System
SAP

NPR process flow

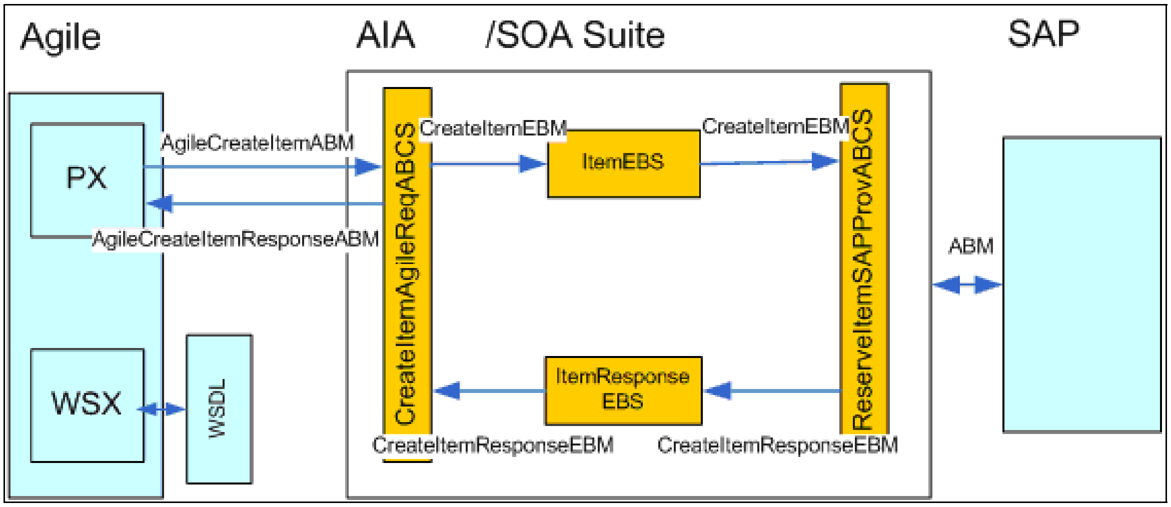
This figure shows the sequence of events that occur when creating a NPR:


Start
NPR Approval(out of scope)
Menu Extension
Agile
(Agile New Part Request Web Service)
Y
Create item Successful?
N
Agile Process Item Fail
Promote NPR
Agile Process Item Success
Create Item
Legend
Agile to SAP Request
SAP to Agile Response
Input File
# Send Email
* Item EBO
Processes w/multi Action (see multi action process section)
Agile ABCS
(Agile New Part Request ABCS)
Transform Item Info to ABO/ABM
Receive New Item # Request
Transform Item Info to ABO/ABM
Lookup Mapping File
Transform ABO/ABM → EBO/EBM
Call Reserve Item
Transform ABO/ABM → EBO/EBM
AIA
(New Part Request EBS)
Look up Routing Rules
Reserve Item#
#Send Email
SAP ABCS
(SAP New Part Request ABCS)
Reserve Item #
Transform Item EBO/EBM → ABO/ABM
Lookup Mapping/XSL File
Transform ABO/ABM → EBO/EBM
Transform ABO/ABM to Native Apps Data
Transform Native Apps Data to ABO/ABM
* Send Item EBO Process
SAP action status
Prepare SAP Call for reserving Items#
Call reserve Item Web Service
SAP
(SAP New Part Request Web Service)
Reserve Item API

Sequence for creating a new part number

NPR Integration Flow

The NPR process is triggered from a PX. It is a synchronous process and therefore the process is designed based on the Requestor - Provider pattern as shown in this figure:



Process flow from Agile to SAP for new part number

The PX creates the Agile request Application Business Message (ABM) (*AgileCreateItemABM*) for *the* CreateItemAgileReqABCS and invokes it:

The request AgileCreateItemABM is transformed to the Enterprise Business Message *CreateItemEBM* for NPR process.

1. Using the Asynchronous Message Pattern, the NPR process is invoked on the ItemEBS by the CreateItemAgileReqABCSImpl with the CreateItemEBM as input.
2. The CreateItemEBM is routed to the ReserveItemSAPProvABCS with the CreateItemEBM as input.
3. The ReserveItemSAPProvABCS implements the business logic for generating the new part number in SAP.
4. A response CreateItemResponseEBM with New Part Number is returned to the ItemResponseEBS and routed back to the CreateItemAgileReqABCSImpl.
5. The CreateItemAgileReqABCSImpl transforms the response CreateItemResponseEBM to Agile response AgileCreateItemResponseABM.
6. CreateItemAgileReqABCS receives AgileCreateItemResponseABM and returns back to the PX.
7. The Part Number from the AgileCreateItemResponseABM is returned to the web client and displayed in the **Number** field in the Create Item screen.

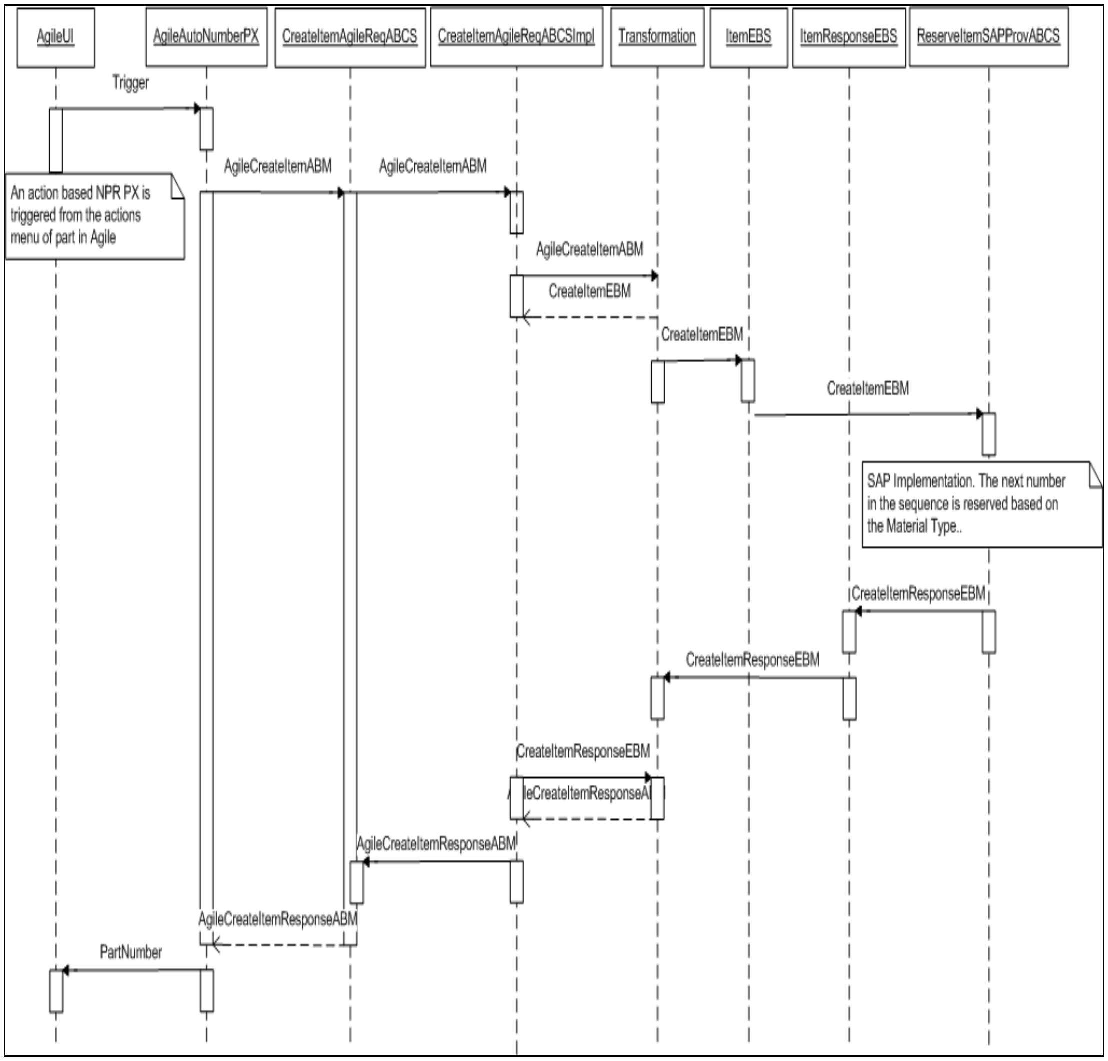
NPR Services Orchestration

The NPR Process itself is an asynchronous process, whereas PX expects the response synchronously. Therefore, there may be a delayed response from the provider as the CreateItemAgileReqABCSImpl is implemented as an asynchronous BPEL Composite. To facilitate this, the synchronous BPEL Composite CreateItemAgileReqABCS is used to invoke the CreateItemAgileReqABCSImpl and receive the response back. CreateItemAgileReqABCS is invoked by the PX synchronously.

The asynchronous message pattern Request-Delayed response pattern with one-way calls in Enterprise Business Services (EBS) is used here with EBMID as the correlation ID.

Between the CreateItemAgileReqABCS and CreateItemAgileReqABCSImpl, ABMHeader Id is used as the correlation ID.

This figure shows the sequence of events during NPR services orchestration:



AutoNumber PX

This table describes the activities involved when creating a NPR:

#	Activity	Remarks
1	Agile NPRAutonumberPX is triggered	The Agile NPRAutonumberPX is triggered by an Agile user from the Agile Web client as part of New Part creation in Agile.
2	Invoke CreateItemAgileReqABCS with AgileCreateItemABM as input	NPRAutonumberPX process invokes the CreateItemAgileReqABCS with AgileCreateItemABM ABM as input.
3	Invoke CreateItemAgileReqABCSImpl with AgileCreateItemABM as input	NPRAutonumberPX process invokes the CreateItemAgileReqABCSImpl with AgileCreateItemABM ABM as input.
4	CreateItemAgileReqABCSImpl invokes the ItemEBS with CreateItem operation	An invoke activity in CreateItemAgileReqABCSImpl transforms the AgileCreateItemABM ABM to CreateItemReqMsg EBM and invokes the CreateItem operation on ItemEBS with CreateItemReqMsg as the input. CreateItemReqMsg will be routed to SAP ReserveItemSAPProvABCS.
5	ReserveItemSAPProvABCS on the SAP first transforms & then calls the SAP service	ReserveItemSAPProvABCS will first transform CreateItemReqMsg into the input of SAP Service to reserve the item in SAP and then calls that service. This would create an Item in using a generic part number generation process (mainly using a sequence). ReserveItemSAPProvABCS invokes ItemResponseEBS with CreateItemResponseMsg as input which is routed back to the CreateItemAgileReqABCSImpl.
6	CreateItemAgileReqABCSImpl first transforms & then returns the response to CreateItemAgileReqABCS	CreateItemAgileReqABCSImpl will first transform CreateItemResponseMsg to AgileCreateItemResponseABM and returns the same to CreateItemAgileReqABCS.
7	CreateItemAgileReqABCS returns AgileCreateItemResponseABM to NPRAutonumberPX.	
8	NPRAutonumberPX returns the partNumber to New Part Creation UI in Agile	NPRAutonumberPX gets the partNumber from the *AgileCreateItemResponseABM* and returns it to the New Part Creation UI in Agile web client which gets displayed in the Number field.

Oracle AIA Services for NPR

The process integration for NPR uses the following industry components:

Core components for NPR

EBOs	ItemEBO
EBMs	CreateItemEBM CreateItemResponseEBM
EBSs	ItemEBS

Core Components Locations

EBO and EBM XSD files	$AIA_HOME/AIAMetaData/AIAComponents/EnterpriseObjectLibrary/Core/EBO/Item
WSDL files	$AIA_HOME/AIAMetaData/AIAComponents/EnterpriseBusinessServiceLibrary/Core/EBO/Item

For detailed documentation of individual Enterprise Business Objects (EBOs), click **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.

EBOs can be extended, for instance, to add new data elements. These extensions are protected, and 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 and SAP Components for NPR

This table contains the Agile and SAP components for NPR:

Services	Agile (Requester)	SAP (Provider)
ABMs	AgileCreateItemABM AgileCreateItemResponseABM	BAPI_STDMATERIAL_GETINTNUMBER BAPI_STDMATERIAL_GETINTNUMBERResponse
ABCS	CreateItemAgileReqABCS CreateItemAgileReqABCSImpl	ReserveItemSAPProvABCS
EBS	ItemEBS	ItemResponseEBS

Component Locations

ABO XSD files	$AIA_HOME/AIAMetaData/AIAComponents/ApplicationObjectLibrary/Agile/V1/schemas
ABM XSD files	$AIA_HOME/AIAMetaData/AIAComponents/ApplicationObjectLibrary/Agile/V1/schemas $AIA_HOME/AIAMetaData/AIAComponents/ApplicationObjectLibrary/SAP/Release1/Core/ABM/Ite

Integration Services

These are the integration services for NPR:

ItemEBS

ItemEBS is the EBS that exposes the operations related to the Item Integration on the Item EBO. The following are the routing rules:

ItemEBS Mediator service

- CreateItemAgileReqABCSImpl
  - Route CreateItemEBM to ReserveItemSAPProvABCS

ItemResponseEBS Mediator service

- ReserveItemSAPProvABCS
  - Route CreateItemResponseEBM to CreateItemAgileReqABCSImpl

CreateItemAgileReqABCSImpl

The CreateItemAgileReqABCSImpl transforms the Agile message (AgileCreateItemABM) into CreateItemEBM and calls the routing service to Create Item and get the CreateItemResponseEBM response from SAP. It then transforms the CreateItemResponseEBM response from the routing EBS back to an Agile message (AgileCreateItemResponseABM) and sends it to CreateItemAgileReqABCS, which returns it to the calling Agile PX.

Flow

Receives AgileCreateItemABM from CreateItemAgileReqABCS, which gets the same from NPRAutonumberPX process extensions.

1. Transforms this message into the CreateItemEBM -
   a. Populates the EBM header
   b. Determines Target System ID(s) and adds them into the EBM header to control routing
   c. Validates required fields

   d. Maps to EBM

2. Call Target ItemEBS with operation CreateItem
3. Transform CreateItemResMsg Response EBM (CreateItemResponseEBM) to AgileCreateItemResponseABM Agile message.
4. Send AgileCreateItemResponseABM back to CreateItemAgileReqABCS, which in turn sends it back to the respective PX.
5. CreateItemAgileReqABCSImpl has the following transformation:
   - AgileCreateItemABM_to_CreateItemEBM2.
     CreateItemResponseEBM_to_AgileCreateItemResponseABM

ReserveItemSAPProvABCS

ReserveItemSAPProvABCS is a single operation service. it accepts a *Material* type of an Item information message as a request and returns a response.

In the Agile to SAP flow, ReserveItemSAPProvABCS is used for transforming CreateItemEBM into BAPI_STDMATERIAL_GETINTNUMBER ABM, which invokes the Reserve Item operation in SAP.

In the return flow, BAPI_STDMATERIAL_GETINTNUMBER sends BAPI_STDMATERIAL_GETINTNUMBERResponseABM, which is transformed by ReserveItemSAPProvABCS into CreateItemResponseEBM.

This service is implemented as an asynchronous BPEL Composite.

NPR Integration Customization Points

These are the NPR integration customization points:

Agile

Property	Name	Description
CreateItemAgileReqABCSImpl (Agile Process Item requestor flow)	AgileCreateItemABM_to_CreateItemEBM_Custom.xsl	Request ABM to Request EBM
	CreateItemEBM_EBMHeader_Custom.xsl	Request EBM to EBM Header

SAP

Property	Name	Description
ReserveItemSAPProvABCS	GenerateItemEBMtoABM_Custom.xsl	Request EBM to Request ABM
	GenerateItemABMtoEBM_Custom.xsl	ResponseABM to ResponseEBM

Properties and DVMs

The ITEM_PRIMARYCLASSIFICATIONCODE property must be set for the NPR in the AIAConfigProperties.xml to work.

Chapter 4: Process Integration for Item Attribute Update

This chapter provides an overview of the process integration for Item Attribute (IA) Update and discusses solutiontion assumptions, the IA Update integration sequence, Oracle Application Integration Architecture (Oracle AIA) services for IA Update, and IA Update integration customization points.

This chapter includes the following sections:

- Overview:
- Item Attribute Update Process
- IA Update Process Integration Solution Assumptions
- IA Update Integration Sequence
- Oracle AIA Services for IA Update
- Integration Services
- IA Update Integration Customization Points
- Essential Properties and Domain Value Maps (DVMs)

Overview

Required as part of the Manufacturing Update process for product information synchronization, the Item Attributes information from the Enterprise Resouce Planning (ERP) system is required to be updated in Agile. The term Item applies to both Parts and Documents in Agile considering the documents are integrated in to SAP as a part object only and not as a Document Info Record (DIR). DIR is not in scope for this version of integration.

Since item attribute information can exist in multiple plants in the ERP system, it is supported only for cases where the system can accurately determine which plant in ERP to pick up item attribute information from.

Item Attribute Update Process

The update of Item Attribute information from ERP to Agile Process Lifecycle Management (PLM) is performed as event-based and in near real time.

The following steps are executed:

1. The process kicks off when configured fields are modified and saved on an item in System Application and Products (SAP).
2. Retrieve the attributes that have changed on the item in SAP.

This data is retrieved from one or more organizations (plants) as per site-org mappings or from the configured single organization.

3. Update the information into Agile.

   The input consists of the item attribute values, which must be updated in Agile. Unique identifiers for the item, along with values for all the attributes mapped for transfer from ERP to Agile are provided as input.

   In Agile 9.3.1.2 an item number is the only attribute that is required for uniquely identifying the item to be updated.

   A unique item revision in Agile is composed of a revision number as well as a Change Order (CO) number (because it is possible for certain typesof COs in Agile to revise an item without changing its revision number). In the context of integration, in most cases, updates to item attributes applies only to the latest released revision.

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

In this release of integration, developed on Agile 9.3.1.2, 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

IA Update is performed either near real time or batch. In order to achieve this, implementation requires a one-time configuration of change pointers and *Application Link Enabling* (ALE) in SAP.

For more information about this configuration setup, see Setting up SAP.

As part of this configuration, all the fields that must be updated back in to Agile are configured such that any changes saved to these fields on an item triggers the update process immediately. For the batch process, it is required to schedule the **IDOC** as a background job.

The moment you save the changes on the item in SAP, the change pointers configuration triggers an event. The event pushes the data to the configured partner (ALE configurations) and in this case it is the Agile system. All the changes made on the item are captured and updated on the specified item in Agile. This process triggers once for each item change save event.

Exceptions

In all cases, the exception message displayed 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 are:

- 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

These are the solution assumptions for the process integration for IA Update:

1. If item information must be retrieved from multiple plants in the SAP (ERP) system, the following constraints must be met in order to support this process:

    a. Agile multi-sites is configured

    b. There is a one-to-one mapping between Agile sites and ERP plants

2. The attributes to be updated with ERP data are on the **Sites** tab of the item

    In this release of integration “Changes are interfaced to only one destination system”.

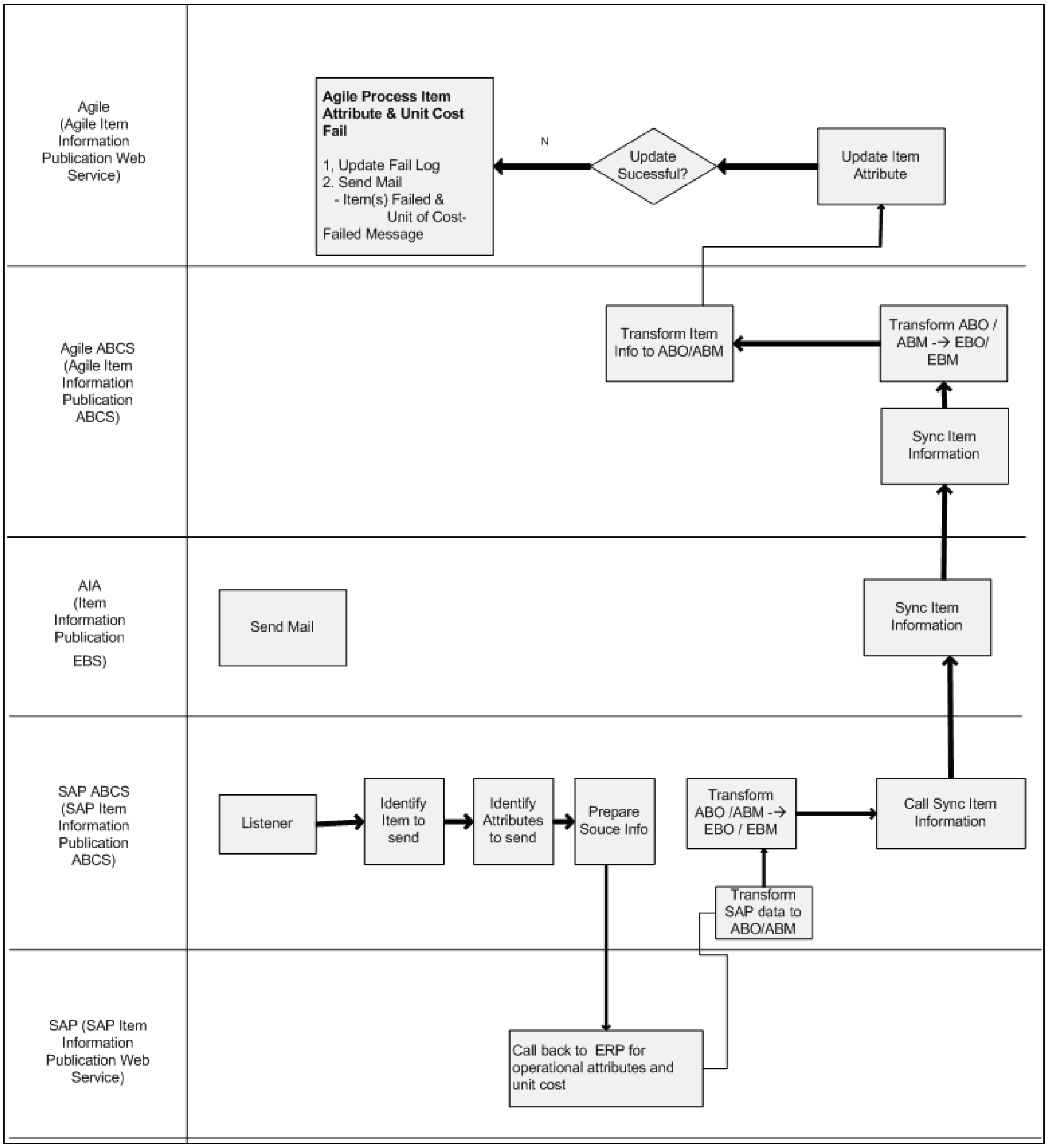
3. The following exception conditions are tested to make sure that they return user-friendly error messages. In all cases, the exception message displayed must contain the Item number that was being updated and the step at which the error occurred, along with other supporting information as needed.

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

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

IA Update Integration Sequence

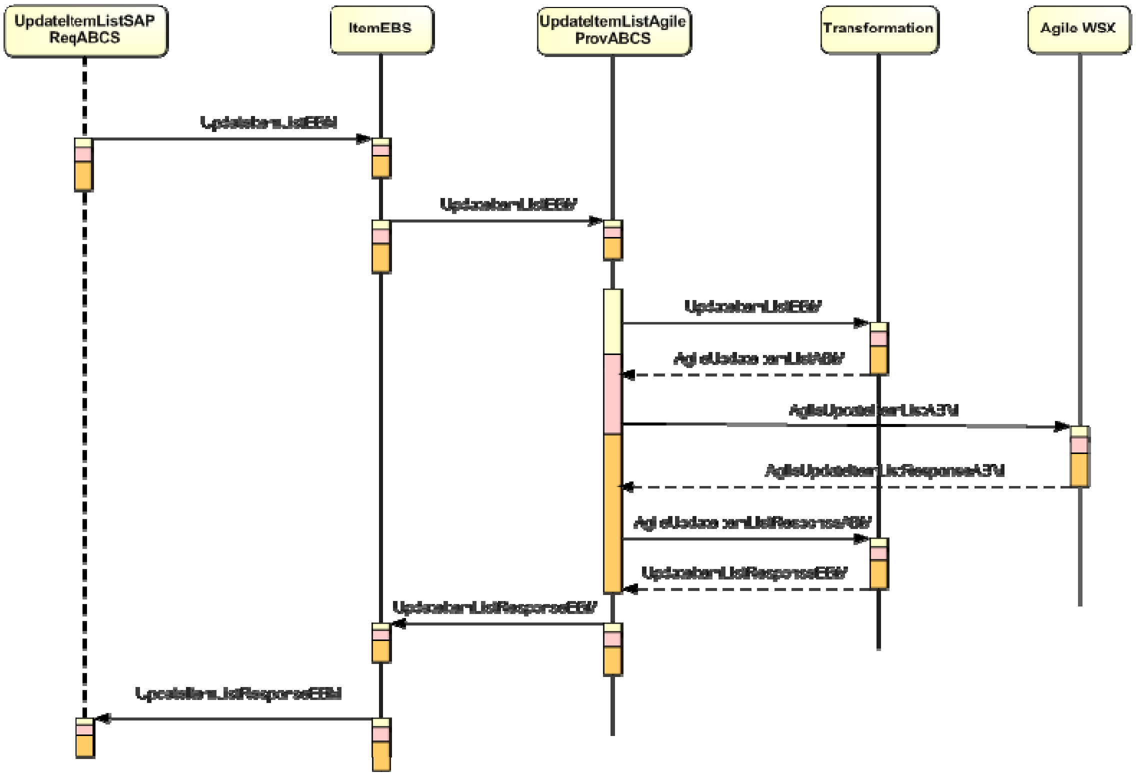
This figure shows the sequence of events that occur during IA update:



IA update sequence

IA Update Services Orchestration

This sequence diagram shows the process flow for IA update process flow:


UpdateItemListSAP ReqABCS
ItemEBS
UpdateItemListAgile ProvABCS
Transformation
Agile WSX
UpdateItemListEBM
UpdateItemListEBM
UpdateItemListEBM
AgileUpdateItemListABM
AgileUpdateItemListABM
AgileUpdateItemListResponseABM
AgileUpdateItemListResponseABM
UpdateItemListResponseEBM
UpdateItemListResponseEBM
UpdateItemListResponseEBM

Sequence diagram for IA update process flow

This table describes the activities involved in the IA update process:

#	Activity	Description
1	SAP Item Operational Attribute Update Action Trigger	SAP Item Operational Attribute Update Action trigger Invokes the UpdateItemListSAPReqABCS
2	UpdateItemListSAPReqABCS calls UpdateItemListSAPReqABCSImpl	UpdateItemListABM will be passed to UpdateItemListSAPReqABCSImpl as the input
3	UpdateItemListSAPReqABCSImpl invokes ItemEBS	An invoke activity in UpdateItemListSAPReqABCSImpl invokes the UpdateItemList operation on ItemEBS with UpdateItemListEBM as input
4	ItemEBS routes UpdateItemListEBM to UpdateItemListAgileProvABCSImpl	ItemEBS routes UpdateItemListEBM as input to UpdateItemListAgileProvABCSImpl

#	Activity	Description
5	UpdateItemListAgileProvABCSImpl invokes Agile's Item Operational Attribute Update Web Service	UpdateItemListAgileProvABCSImpl transforms the UpdateItemListEBM to AgileUpdateItemListABM and invokes UpdateItem service operation on Agile Web Service to update Item cost related attribute information from SAP to Agile. The AgileUpdateItemListResponseABM is returned back to UpdateItemListAgileProvABCSImpl

Oracle AIA Services for IA Update

The process integration for IA Update uses the following industry components:

Core Components for AIA Update

EBOs	ItemEBO
EBMs	UpdateItemListEBM UpdateItemListResponseEBM
EBSs	ItemEBS ItemResponseEBS

Core Components Locations

EBO & EBM XSD files	$AIA_HOME/AIAMetaData/AIAComponents/EnterpriseObjectLibrary/Core/EBO/Item
WSDL files	$AIA_HOME/AIAMetaData/AIAComponents/EnterpriseBusinessServiceLibrary/Core/EBO/Item

For detailed documentation of individual Enterprise Business Objects (EBOs), click **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 Enterprise Business Service (EBS).

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 about EBOs and Enterprise Business Messages (EBMs), see *Oracle Application Integration Architecture – Foundation Pack: Integration Developer's Guide*, "Extensibility for AIA Artifacts."

SAP and Agile Components for IA Update

This table lists the SAP and Agile components for IA updates:

Services	SAP (Requester)	Agile (Provider)
ABMs	UpdateItemListABM	AgileUpdateListABM AgileUpdateListResponseABM
ABCS	UpdateItemListSAPReqABCSImpl UpdateItemListSAPReqABCS	UpdateItemListAgileProvABCSImpl
EBS	ItemEBS	ItemResponseEBS

Component Locations

ABO XSD files	$AIA_HOME/AIAMetaData/AIAComponents/ApplicationObjectLibrary/Agile/V1/schemas

Integration Services

These are the integration services for IA Update:

ItemEBS

ItemEBS is the Enterprise Business Service, which exposes the operations related to the Item Attribute Update Integration on the ItemEBO. The following are the routing rules:

ItemEBS Mediator service

- UpdateItemListSAPReqABCSImpl
  - Route UpdateItemListEBM to UpdateItemListAgileProvABCSImpl

UpdateItemListSAPReqABCSImpl

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

The requester Application Business Connector Service (ABCS) is a synchronous process, which receives a list of Item Ids from the SAP Change Pointer program. The lists of Ids are those which have last_update_date greater than the last run date of the Change Pointer.

- The requester Business Process Execution Language (BPEL) Composite filters the list of Ids that are present in the cross-reference tables in the Fusion Middleware (FMW) layer. This provides a list of Items that were actually from the Agile system. The original list of Item Ids may be those from non-Agile sources as well.

- The BPEL Composite then makes a Mediator call for end point virtualization. This Mediator layer has a SAP Adapter, which calls the Business Application Programming Interface (BAPI) that provides the Item details required to be sent out. This is the Item Application Business Message (ABM).
- A transformation converts the ABM to an Enterprise Business Message (EBM).
- An asynchronous request-delayed response call is made to the ItemEBS with the ItemListEBM. This is routed 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.

UpdateItemListAgileProvABCSImpl

UpdateItemListAgileProvABCSImpl is used to facilitate the communication between ItemEBS and the 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 application programming interfaces (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 BPEL Composite.
- If the UpdateItems service operation fails on the Agile side, a new Fault is generated and is sent across with the appropriate error message.

IA Update Integration Customization Points

These are the IA Update integration customization points:

SAP

Property	Name	Description
UpdateItemListSAPReqABCSImpl	XFormABMToEBM_Custom.xsl	Custom transformations for Item Request ABM to Request EBM

Agile

Property	Name	Description
UpdateItemListAgileProvABCSImpl (Agile update item attributes provider flow)	AgileUpdateItemListResponseABM_to_UpdateItemListResponseEBM_Custom.xsl	RespABM to RespEBM (custom element)
	AgileUpdateItemListResponseABM_to_UpdateItemListResponseEBM_Impl.xsl	RespABM to RespEBM (main)
	UpdateItemListEBM_to_AgileUpdateItemListABM_Impl.xsl	ReqEBM to ReqABM (main)

Essential Properties and DVMs

The following mandatory properties and DVMs (in AIAConfigProperties.xml) must be set for the Item Attribute Update process to flow to work:

- AGILE_SITE_TARGET_MAPPING
- AGILE_TARGET_SITE_MAPPING
- ITEM_PRIMARYCLASSIFICATIONCODE

Chapter 5: Implementing the Process Integration Pack

This chapter describes how to implement the Agile PLM Integration Pack for SAP: Design to Release, including prerequisites, setting up the participating applications, loading cross-reference data, configuring the pre-built integration, working with domain value maps (DVMs), application interfaces, and how to handle errors.

This chapter includes the following sections:

- Prerequisites
- Setting up Agile PLM
- Setting Up SAP
- Loading Cross Reference Data
- Configuring the integration
- Working with DVMs
- Application Interfaces
- Handling Errors

Prerequisites

The Agile Product Lifecycle Management (PLM) integration is deployed through the Oracle Application Integration Architecture (Oracle AIA) Installer. For proper functioning, the integration requires certain settings and configurations in partner applications such as Agile PLM and System Application and Products Enterprise Resource Planning (SAP ERP), as well as the Oracle AIA configuration property files.

- Service-Oriented Architecture (SOA) 11.1.1.6 must be installed and relevant patches applied.
- Oracle AIA Foundation Pack Foundation Pack 11.1.1.6, RUP patch 14253266 or later must be installed.
- Agile PLM must be installed and configured accordingly.
- Oracle Application adapter for SAP must be installed.
- All the required webservices for the corresponding Business Application Programming Interfaces (BAPIs) and IDOCs of SAP must be generated from the Oracle Application adapter.
- SAP environment must be installed and configured accordingly.
- Before enabling the Agile PLM Integration Pack for SAP: Design to Release, you must set up the following participating applications:

- Agile PLM
- SAP
- Cross-reference data

Setting up Agile PLM

After installating the Agile PLM Integration Pack for SAP: Design to Release, the Agile Administrator must set up the Agile Content Services (ACS) as well as the following post-installation configurations using Java client:

- Create a new Java Message Service (JMS) Destination.
- Create new events for Engineering Change Order (ECO), Manufacturing Change Order (MCO), and Site Change Order (SCO).
- Create new subscribers for ECO, MCO, and SCO.
- Create Auto Numbers
- Update Change Order (CO) Page Two Fields
- Configure Sites
- Configure Site Attributes
- Configure and Deploy Process Extension (PX)

Creating a New JMS Destination

To create new JMS Destination:

1. Login to Agile Java Client using the login credentials provided.
2. Navigate to Admin, System Settings, Agile Content Service, Destinations.
3. Click New to create a new JMS Destination with the following information.

Parameters	Variables
Name	SOA Application Server
Description	SOA Application Server
Protocol	JMS
Username	SOA Application Server Username
Password	SOA Application Server Password
Provider Context Factory	weblogic.jndi.WLInitialContextFactory
Connection Factory	jms/aia/AIAAgilePLMECOCF

Parameters	Variables
Default Provider URL	t3://<soa_hostname>:<port>
Destination Name	jms/aia/AIAAgilePLMECOJMSQueue

Note: If you are pointing Agile to a clustered Fusion Middleware (FMW) environment, the **Default Provider URL** must contain all possible URLs in the cluster separated by a comma. For example, *t3://<soa_hostname1>:<soa_port1>,<soa_hostname2>:<soa_port2>*.

4. Test the JMS Destination.

   The test must be successful.

Create New Events for ECO, MCO and SCO

To create new events for ECO, MCO and SCO

1. Navigate to Admin, System Settings, Agile Content Service, Events.
2. Click New to create new Event(s).
   - a. Enter values for **Name** and **Description**.
   - b. Choose relevant workflows.
   - c. For SCO choose *Default Site Change Orders*.
   - d. For ECO choose *Default Change Orders*.
   - e. For MCO choose *Default Manufacturer Orders*.
   - f. Choose the workflow status as *Released*.

Creating New Subscribers for ECO, MCO and SCO

To create new subscribers for ECO, MCO and SCO:

1. Navigate to Admin, System Settings, Agile Content Service, Subscribers
2. Click New to create new Subscribers for ECO, MCO and SCO
   - a. Enter a **Name** and a **Description** for the subscribers
   - b. Associate the workflow as *Automated Transfer Orders* (ATOs)
   - c. Choose *All Change Orders, All Manufacturer Orders*, and *All Site Change Orders* as criteria for ECO, MCO and SCO respectively
   - d. Choose the appropriate **Event** that you created for ECO, MCO and SCO
   - e. Click **OK**

3. Select the **Subscriber Details** tab to add the subscriber detail information.

4. Click the **Add Row** icon to create a new row.

   This opens Subscriber Detail window

   a. Click **Destinations** and select the JMS Destination that was created

   b. Click **Filters** and choose the following filters for **Filters** field using **> button** in the pop up that appears

   c. ECO: *Default Change Order Filter*

   d. MCO: *Default Manufacture Order Filter*

   e. SCO: *Default Site Changes Order Filter*

   f. Add the *Default Item Filter* as filter for the ECO, MCO and SCO

   g. Click **Role(s)** and select all the roles using **> button** in the pop up that appears

   h. Select *aXML* as **Format**

   i. Select *English* as **Language**

   j. Select *All* as **Site**

5. Select each subscriber at a time and click **Enable** to enable all the created subscriptions

6. Navigate to **Admin, System Settings, Agile Content Service, Filters**

7. Select **Default Item Filter** and double-click

8. Navigate to the **Filters** tab and append **Sites** to the **View Tabs** field value

   a. Choose *Tab and Items* as the value for **BOM Options**

   b. Select *All Levels* as the value for **BOM Level**

   c. Choose *Tab and Manufacturer Parts* as the value for **AML Options**

9. Select Default Change Order Filter and double-click

10. Navigate to the **Filters** tab

    a. Choose **Redline Changes Only** option as *No*

    b. Choose **Affected Items Options** as *Tab and Items*

11. Select Default Site Change Order Filter and double-click

12. Navigate to the **Filters** tab

    a. Choose **Redline Changes Only** option as *No*

    b. Choose **Affected Items** options as *Tab* and *Items*

13. Navigate to Admin, User Settings, Privileges

14. Click **Edit** to modify privileges for *Changes, MCO* and *SCO*

a. Enter a **Name** and **Description**
b. Select *Yes* for **Enabled**
c. Select *Modify* for **Privilege**
d. Click **Criteria** and choose the relevant criteria for *Changes, MCO* and *SCO*
e. Click **Applied to** and select all attributes, including the invisible and disabled attributes
f. Click **OK**

This opens the privilege screen of the entity for which you set privileges. You must now assign roles to the privileges that you have created for *Changes, MCO*, and *SCO*.

15. Navigate to the **Where Used** tab.
16. Click **Add Roles** and select the roles in the **Select Roles** pop up
17. Click **OK** to assign these roles to the privilege that you created.
18. Navigate to **Admin**, **User Settings**, **Privileges** and double click **Read**.
19. Click **New** in the **Privileges to Read** screen to create **Read** privileges for *MCO*
20. In the Create Privilege pop up that opens

a. Enter **Name** and **Description**
b. Choose *Yes* for **Enabled**
c. Choose *Read* for **Privilege**
d. Choose *All Manufacturer Orders* for **Criteria**
e. Choose both visible as well as invisible and disabled attributes for **Applied**
f. Click **OK**

This opens privilege information screen for *MCO*.

21. Navigate to **Where Used** tab
22. Click **Add Roles** to select roles and assign to the created privilege
23. Modify the **Read Changes** and **Read Items** to include **Admin** user
24. Click **OK**
25. Navigate to **Admin**, **Data Settings** and double click **Classes**
26. Double click **Parts** in Classes window
27. Navigate to the **User Interface Tabs** tab
28. Double click **Page Two**
29. Navigate to the **Attributes Page Two** tab

30. Check for the attribute **Numeric01** and double click it and change the **Name** to *ERP Manufacturer Cost*

Note: If you do not have the *Sites* version on the Agile application, continue performing steps 31 through 45. If you have the *Sites* version, continue to the next section.

31. Navigate to **Admin**, **Data Settings** and double click **Classes**
32. Double click **Parts** in Classes window

Note: Repeat the process also for *Document*.

33. Navigate to the **User Interface Tabs** tab
34. Double click **Page Two**
35. Navigate to the **Attributes Page Two** tab.
36. Check for the **MultiList01** and double click it and change the **Name** to *OrganizationCodes*.

    You must also assign the values that are specific to Sites or Organization codes on the Agile side.

37. Click **New List** to create the List of Values (LOV)
38. Click **OK**

    This opens the Create a New List Value screen.

39. Enter the values for **Name** and **Description** specifying the list of Organizations used
40. Click **Add**
41. Navigate to **List** tab
42. Click **Add** to add the batch list
43. Add batch list values in the pop up window
44. Click **OK**
45. Ensure that **Page Two** on these classes in enabled

Creating Auto Numbers

To create auto numbers:

1. Navigate to Admin, Data Settings, AutoNumbers
2. Click **New** to create auto numbers for New Part Request (NPR) Document Number
3. Enter the following values in the Define the AutoNumber screen

   a. Enter the name as *NPR Document Number*
   b. Choose *Yes* for **Enabled**
   c. Choose *Custom* for **Type**
   d. Choose *Document* for **Where Used**
   e. Choose *com.oracle.aia.npr.NPRAutoNumber* for **Custom AutoNumber**
   f. Click **OK**

4. Click **New** to create auto numbers for NPR Part Number
5. Enter the following values in the Define the AutoNumber screen
   a. Enter the name as *NPR Part Number*
   b. Choose *Yes* for **Enabled**
   c. Choose *Custom* for **Type**
   d. Choose *Part* for **Where Used**
   e. Choose ***com****.oracle.aia.npr.NPRAutoNumber* for **Custom AutoNumber**
   f. Click **OK**

Updating Change Order Page Two Fields

To update Change Order Page Two fields:

1. Navigate to Admin, DataSettings, Classes
2. Double click **Classes** and navigate to **Changes**, **Change Orders**
3. Double click **Change Orders** and navigate to the **User Interface** tabs in the Change Order window
4. Double click **Page Two** and Navigate to the **Attribute Page Two** tab
5. Select **Text02** and double click
6. Set **Visible** to *Yes* in the Attributes Text02 window
7. Click Save
8. Navigate to **Admin**, **User Settings**, **Privileges** and double click **Read**
9. Click **New** to create a new *Read* privilege to read **all** the ECO fields.
10. Select required fields for the **Applied To** field in the Read Changes Page Two window.
11. Click OK
12. Navigate to **Where Used** tab.
13. Click **Select Roles** to select roles and select appropriate users in the **Select Roles** window.

Use the same procedure that you used for setting Read privileges and set Modify privilege for ECO fields.

Configuring Sites

You are only required to configure sites if the Agile application is enabled with the *Sites* version.

To configure sites:

1. Log in to the web client using http://<AGILE_HOSTNAME>/Agile/PLMServlet
2. Navigate to **Search** criteria and select Sites option
3. Select **Sites** from the options
4. Select *Site* for **Type** and enter **Site Name**
5. Click **Finish**

Configuring Sites Attributes

To configure Site attributes:

1. Navigate to Admin, Data Settings, Classes
2. Double click **Classes** and navigate to **Items**, **Parts** in Classes window.
3. Double click **Parts** and navigate to **User Interface Tabs**, **Sites**.
4. Double click **Sites** and navigate to **Attributes: Sites** tab.
5. Select **Numeric01** and double click in the window that opens
   a. Change **Name** to *UnitCostAttribute*
   b. Set **Visibility** to *Yes*

Setting up SAP

The SAP Application Link Enabling (ALE) interface is part of the integration layer within SAP's Business Framework Architecture (BFA). It is a component-based architecture that enables business process integration and asynchronous data communication either between two or more SAP systems or between SAP and external systems.

Application systems are loosely coupled in an ALE integrated system and the data is exchanged asynchronously. Consistency is maintained as data is exchanged between application systems in a controlled manner.

The following steps must be performed and maintained for ALE configuration:

- Define logical systems

- Assign logical systems
- Model the distribution scenario
- Define the Remote Function Call (RFC) Destination
- Define ports
- Define partner profiles
- Globally activate the change pointers
- Set up reference material

Defining Logical Systems

To define logical systems:

1. Enter transaction code SALE in the transaction space.

   Go to **Sending and Receiving Systems, Logical Systems** and select the *Define Logical System* option.

2. In the Change View "Logical System" Overview screen, select the New Entries option.
3. In New Entries: Overview of Added Entries screen, type in the logical system names, which must be assigned to the systems that are in communication.
4. Click Save.

Assigning Logical Systems

To assign logical systems:

1. Enter the transaction code SALE in the transaction space

   Go to **Sending and Receiving Systems > Logical Systems** and select *Assign Client to Logical Systems* option.

2. In the Change View "Clients": Overview screen, select the *New Entries* option.

   By selecting *New Entries*, you are automatically directed to the New Entries: Details of Added Entries screen.

3. Enter the Client & Logical System defined earlier. Also enter the other appropriate values and save the data.

Modeling the Distribution Scenario

To model the distribution scenario:

1. Enter transaction code **BD64**, switch to *edit* mode and select **Create Model View**.
2. In the Create Model View screen, enter values for Short Text, Technical Name, Start Date and End Date.
3. Select the Model view and Add Message Type *MATMAS*.
4. Save the entries.
5. Select the Model view and then choose the EDIT option in the menu bar. Choose Model view and then Distribute.

   The Model view is now *Distributed*.

Defining the RFC Destination

To define the RFC destination:

1. Enter transaction **SM59**, choose **TCP/IP connections** and then go to the menu and click **Create**
2. In the RFC Destination screen, enter data in RFC destination, Connection type as T and Description, respectively.
3. Click **Enter** to enable TCP/IP connection related fields.
4. Choose **Registration** and provide **Program ID** and save the entries.

Defining Ports

To define ports:

1. Enter transaction code WE21 in the transaction space.
2. Select Transactional RFC and choose the Create option.
3. In the Ports in IDoc processing pop-up, select the Generate port name option.
4. Provide the port name, if the Own port name option is selected.
5. In the Creating a RFC port screen, enter Description, RFC Destination and select an appropriate Version and save the entries.

Defining Partner Profiles

To define partner profiles:

1. Enter transaction code **WE20** in the transaction space.
2. From the drop down list, choose **Partner Type LS** and select **Create** option.
3. In the Partner profiles screen, add message type **SYNCH** in **out bound parameters** and MATMAS in **inbound parameters**.

   Provide any other relevant information, like ports, idoc type and **save** the inputs.

Note: The Outbound parameters screen displays the parameters and the setup of ALE Configuration is completed.

Activating Change Pointers Globally

You must globally activate Change Pointers so that they trigger the MATMAS Idoc to send Material Attributes from SAP to Agile.

To activate Change Pointers:

1. Enter transaction code **BD61** in the transaction area.
2. In the Activate Change Pointers Generally screen, select **Change Pointers** and save the transaction.

To activate Change Pointers for MATMAS Message Type:

1. Enter transaction code **BD50**
2. In the Change View “Activate Change Pointers for Messagee Type”: Overview screen, select *MATMAS* message type
3. Click **Save**

To activate Change Pointers for Individual fields in the Material Master:

1. Enter transaction code **BD52**
2. In the work area enter the **Message Type** as *MATMAS* and click **Continue**
3. Delete those fields for which Change Pointer activation are not required
4. Click Save

Setting Up Reference Material

To create an Item in SAP, reference material must be configured based on the Material type. If the Item is created in more than one plant then the reference material must be maintained in all the plants .The purpose of this reference material is to default all the mandatory values, which are required for the creation of material in SAP. This reference material varies from customer to customer based on their implementation setup.

Loading Cross-Reference Data

A SQL script is used to load cross-reference records.

The following is an example of a SQL script that loads cross-reference records for the XRef table, ITEM_ITEMID with columns AGILE_01 and SAP_01. This sample script allows you to build XREF records for a single item.You can modify this SQL script tosuit your specific requirements.

```
DECLARE
XrefTableName_vc2 VARCHAR2 (2000);
XrefColumnName1_vc2 VARCHAR2 (2000);
XrefColumnName2_vc2 VARCHAR2 (2000);
XrefColumnName3_vc2 VARCHAR2 (2000);
Value_vc2 VARCHAR2 (2000);
RowNumber_vc2 VARCHAR2 (48);
IsDeleted_vc2 VARCHAR2 (1);
LastModified_ts TIMESTAMP (6);
BEGIN
/* The following values need to be set just once per load*/
XrefTableName_vc2 := 'oramds:/apps/AIAMetaData/xref/ITEM_ITEMID.xref';
IsDeleted_vc2 := 'N';
XrefColumnName1_vc2 := 'COMMON';
XrefColumnName2_vc2 := 'AGILE_01';
XrefColumnName3_vc2 := 'SAP_01';
/* You'll just need to change the Value_vc2 variable for your specific
edge app values
*/
/* Row Number is unique to a group of three inserts and ties them
together. It’s a system generated guid in all cases */
/* Copy this section as many times as needed for each set of three
records you need inserted */
/********************************************** Begin set of 3 inserts
*******************************************/
RowNumber_vc2 := SYS_GUID;
/* Common row should have a guid for its value */
Value_vc2 := SYS_GUID;
LastModified_ts := SYSTIMESTAMP;
Insert into XREF_DATA (XREF_TABLE_NAME, XREF_COLUMN_NAME, ROW_
NUMBER, VALUE, IS_DELETED, LAST_MODIFIED) values
(XrefTableName_vc2, XrefColumnName1_vc2, RowNumber_vc2, Value_vc2,
IsDeleted_vc2, LastModified_ts);
/* Value specific to each application, for Agile value generation
please check XREF_
Instructions.doc */
Value_vc2 := 'AgileValue1';
```

```
LastModified_ts := SYSTIMESTAMP;
Insert into XREF_DATA (XREF_TABLE_NAME, XREF_COLUMN_NAME, ROW_
NUMBER, VALUE, IS_DELETED, LAST_MODIFIED) values
(XrefTableName_vc2, XrefColumnName2_vc2, RowNumber_vc2, Value_vc2,
IsDeleted_vc2, LastModified_ts);
/* Value specific to each application, for SAP value generation please
check XREF_
Instructions.doc */
Value_vc2 := 'SAPValue1';
LastModified_ts := SYSTIMESTAMP;

Insert into XREF_DATA (XREF_TABLE_NAME, XREF_COLUMN_NAME, ROW_
NUMBER, VALUE, IS_DELETED, LAST_MODIFIED) values
(XrefTableName_vc2, XrefColumnName3_vc2, RowNumber_vc2, Value_vc2,
IsDeleted_vc2, LastModified_ts);
/************************************************** End of set of 3
inserts
*****************************************/
END;
```

The following is an example of a SQL script that loads cross-reference records for the XRef table, CHANGE_CHANGEID with columns AGILE_01 and SAP_01. You can modify this SQL script to suit your specific requirements.

```
DECLARE
XrefTableName_vc2 VARCHAR2 (2000);
XrefColumnName1_vc2 VARCHAR2 (2000);
XrefColumnName2_vc2 VARCHAR2 (2000);
XrefColumnName3_vc2 VARCHAR2 (2000);
Value_vc2 VARCHAR2 (2000);
RowNumber_vc2 VARCHAR2 (48);
IsDeleted_vc2 VARCHAR2 (1);
LastModified_ts TIMESTAMP (6);

BEGIN
/* The following values need to be set just once per load*/
XrefTableName_vc2 := 'oramds:/apps/AIAMetaData/xref/CHANGE_
CHANGEID.xref';
IsDeleted_vc2 := 'N';
XrefColumnName1_vc2 := 'COMMON';
XrefColumnName2_vc2 := 'AGILE_01';
XrefColumnName3_vc2 := 'SAP_01';
/* You'll just need to change the Value_vc2 variable for your specific
edge app values
*/
/* Row Number is unique to a group of three inserts and ties them
together. Its a
system generated guid in all cases */
/* Copy this section as many times as needed for each set of three
records you need
inserted */
/********************************************** Begin set of 3 inserts
*****************************************/
RowNumber_vc2 := SYS_GUID;
```

```
/* Common row should have a guid for its value */
Value_vc2 := SYS_GUID;
LastModified_ts := SYSTIMESTAMP;
Insert into XREF_DATA (XREF_TABLE_NAME, XREF_COLUMN_NAME, ROW_
NUMBER, VALUE, IS_DELETED, LAST_MODIFIED) values
(XrefTableName_vc2, XrefColumnName1_vc2, RowNumber_vc2, Value_vc2,
IsDeleted_vc2, LastModified_ts);
/* Value specific to each application, for Agile value generation
please check XREF_
Instructions.doc */
Value_vc2 := 'AgileValue1';
LastModified_ts := SYSTIMESTAMP;
Insert into XREF_DATA (XREF_TABLE_NAME, XREF_COLUMN_NAME, ROW_
NUMBER, VALUE, IS_DELETED, LAST_MODIFIED) values
(XrefTableName_vc2, XrefColumnName2_vc2, RowNumber_vc2, Value_vc2,
IsDeleted_vc2, LastModified_ts);
/* Value specific to each application, for SAP value generation please
check XREF_
Instructions.doc */
Value_vc2 := 'SAPValue1';
LastModified_ts := SYSTIMESTAMP;
Insert into XREF_DATA (XREF_TABLE_NAME, XREF_COLUMN_NAME, ROW_
NUMBER, VALUE, IS_DELETED, LAST_MODIFIED) values

(XrefTableName_vc2, XrefColumnName3_vc2, RowNumber_vc2, Value_vc2,
IsDeleted_vc2, LastModified_ts);

/************************************************** End of set of 3
inserts
******************************************/
END;
```

Identifying Cross-Reference Data

This pre-built integration uses the XREF_DATA table present in Metadata Database (MDS) to maintain cross-reference information between Agile and SAP. This cross-reference information helps map Agile Parts, Documents, and Change Orders to SAP Items and Change Orders.

There are two main virtual tables in the AIA_XREF schemas XREF_DATA table that maintain this cross-reference information.

CHANGE_CHANGEID - maintains all the Change Order information

ITEM_ITEMID - maintains Item information.

Example:

A Change Order *ECO001* contains two revised items:

- P0001 in site Detroit
- P0002 in site Michigan

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 SAP, the following entries are made in the CHANGE_CHANGEID virtual table.

XREF_TABLE	XREF_COLUMN	ROW_NUMBER	VALUE
CHANGE_CHANGEID	AGILE_01	E45E015046AF11DD9F2E436FB39961A8	ECO001::DETROIT
CHANGE_CHANGEID	COMMON	E45E015046AF11DD9F2E436FB39961A8	2d373833303237303132383837353631
CHANGE_CHANGEID	SAP_01	E45E015046AF11DD9F2E436FB39961A8	11075
CHANGE_CHANGEID	AGILE_01	E45E015046AF11DD9F2E436FB39961A9	ECO001::MICHIGAN
CHANGE_CHANGEID	COMMON	E45E015046AF11DD9F2E436FB39961A9	2d373833303237303132383837353632
CHANGE_CHANGEID	SAP_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 0001 Plant in SAP.

The second entry (Common) is also created by the Agile BPEL flow. It indicates the common *business component id* for this particular integration entity and is used for linking change orders to SAP change orders.

The third row entry represents the SAP change id corresponding to the Agile change order *ECO001*. Another set of entries are made for the change order *ECO0001* for site PAV0.

For each revised item and component items in the Change Order, entries are made into the ITEM_ITEMID virtual table. For *ECO001*, the following entries are made in the ITEM_ITEMID table:

XREF_TABLE	XREF_COLUMN	ROW_NUMBER	VALUE
ITEM_ITEMID	AGILE_01	0078BE703EC711DDBF9CA7AA7FE3BDFB	P0001::DETROIT
ITEM_ITEMID	COMMON	0078BE703EC711DDBF9CA7AA7FE3BDFB	35313835373739353732383638303435
ITEM_ITEMID	SAP_01	0078BE703EC711DDBF9CA7AA7FE3BDFB	66247::204
ITEM_ITEMID	AGILE_01	0078BE703EC711DDBF9CA7AA7FE3BDFC	P0002::MICHIGAN
ITEM_ITEMID	COMMON	0078BE703EC711DDBF9CA7AA7FE3BDFC	35313835373739353732383638303436

XREF_TABLE	XREF_COLUMN	ROW_NUMBER	VALUE
ITEM_ITEMID	SAP_01	0078BE703EC711DDBF9CA7AA7FE3BDFC	66248::207
ITEM_ITEMID	AGILE_01	0078BE703EC711DDBF9CA7AA7FE3BDFD	C0001::DETROIT
ITEM_ITEMID	COMMON	0078BE703EC711DDBF9CA7AA7FE3BDFD	35313835373739353732383638303437
ITEM_ITEMID	SAP_01	0078BE703EC711DDBF9CA7AA7FE3BDFD	66249::204
ITEM_ITEMID	AGILE_01	0078BE703EC711DDBF9CA7AA7FE3BDFE	C0002::DETROIT
ITEM_ITEMID	COMMON	0078BE703EC711DDBF9CA7AA7FE3BDFE	35313835373739353732383638303438
ITEM_ITEMID	SAP_01	0078BE703EC711DDBF9CA7AA7FE3BDFE	66250::204
ITEM_ITEMID	AGILE_01	0078BE703EC711DDBF9CA7AA7FE3BDFF	C0003::MICHIGAN
ITEM_ITEMID	COMMON	0078BE703EC711DDBF9CA7AA7FE3BDFF	35313835373739353732383638303439
ITEM_ITEMID	SAP_01	0078BE703EC711DDBF9CA7AA7FE3BDFF	66251::207
ITEM_ITEMID	AGILE_01	0078BE703EC711DDBF9CA7AA7FE3BDFG	C0004::MICHIGAN
ITEM_ITEMID	COMMON	0078BE703EC711DDBF9CA7AA7FE3BDFG	35313835373739353732383638303440
ITEM_ITEMID	SAP_01	0078BE703EC711DDBF9CA7AA7FE3BDFG	66252::207

The first row entry is made by the Agile PLM BPEL flow for part *P0001*. This part belongs to an Agile site that is mapped to site *DETROIT*.

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, and Change Orders to SAP Items and Change Orders.

The third row entry represents the SAP inventory item Iid corresponding to the item P1B and the organization Id for the item. The rest of the entries represent the revised and component items for *ECO001*.

The ITEM_ITEMID virtual table contains the similar cross-reference entries for each item and part created through the new part request process.

Creating Cross-Reference Data

If data is ported from other existing integrations to the Agile PLM pre-built integration, first the relationship between the Agile entities and the SAP entities must be established and each Agile site must be mapped to the corresponding SAP plant by inserting the relevant data in the XREF_DATA table.

You must load the required integration data manually in to the XREF_DATA table to establish the link between Agile PLM and SAP. This data maps the Agile PLM entities to the SAP entities as described in the previous section.

Even for existing Agile PLM pre-built integrations, if an item or change order is created in SAP then it must be created in Agile. For the pre-built integration to process the item, a cross-reference entry for that item must be made. All the update (reverse) flows for the pre-built integration updates the item attributes in Agile only if there is an entry in the XREF_DATA table for that particular item.

The cross-reference file is stored in the MDS repository in the below path

oramds:/apps/AIAMetaData/xref/<xref-name>.xref where xref–name is ITEM_ITEMID or CHANGE_CHANGE_ID

The data that has to be inserted in the XREF_DATA table must be put in an XML file with the following format.

This is a sample change order data file:

```
<xref xmlns="http://xmlns.oracle.com/xref">
   <table name="CHANGE_CHANGEID">
      <columns>
         <column name="SAP_01"/>
         <column name="COMMON"/>
         <column name="AGILE_01"/>
      </columns>
   <rows>
      <row>
          <cell colName="SAP_01">11075</cell>
          <cell
colName="COMMON">2d37383330323730313238383735363l</cell>
          <cell colName="AGILE_01">ECO001::DETROIT</cell>
      </row>
      <row>
          <cell colName="SAP_01">11076</cell>
          <cell
colName="COMMON">2d373833303237303132383837353632</cell>
          <cell colName="AGILE_01"> ECO001::MICHIGAN</cell>
      </row>
   </rows>
</table>
</xref>
```

The Change Id number that is inserted in the SAP_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 SAP organization corresponding to the Agile site are inserted into the AGILE_01 column separated by "**:”**

This is a sample Item Data file:

```
<xref xmlns="http://xmlns.oracle.com/xref">
   <table name="ITEM_ITEMID">
      <columns>
         <column name="SAP_01"/>
         <column name="COMMON"/>
         <column name="AGILE_01"/>
      </columns>
   <rows>
      <row>
          <cell colName="SAP_01">66247::0001</cell>
          <cell
colName="COMMON">35313835373739353732383638303435</cell>
          <cell colName="AGILE_01"> P0001::DETROIT</cell>
      </row>
      <row>
          <cell colName="SAP_01">66248::PAV0</cell>
          <cell
colName="COMMON">35313835373739353732383638303436</cell>
          <cell colName="AGILE_01"> P0002::MICHIGAN</cell>
      </row>
            <row>
          <cell colName="SAP_01">66249::0001</cell>
          <cell
colName="COMMON">35313835373739353732383638303437</cell>
          <cell colName="AGILE_01"> C0001::DETROIT</cell>
      </row>
      <row>
          <cell colName="SAP_01">66250::0001</cell>
          <cell
colName="COMMON">35313835373739353732383638303438</cell>
          <cell colName="AGILE_01"> C0002::DETROIT</cell>
      </row>
<row>
          <cell colName="SAP_01">66251::PAV0</cell>
          <cell
colName="COMMON">35313835373739353732383638303439</cell>
          <cell colName="AGILE_01"> C0003::MICHIGAN</cell>
      </row>
      <row>
          <cell colName="SAP_01">66252::PAV0</cell>
          <cell
colName="COMMON">35313835373739353732383638303440</cell>
          <cell colName="AGILE_01"> C0004::MICHIGAN</cell>
      </row>
   </rows>
```

```
</table>
</xref>
```

The Item inventory number and the organization id is inserted in the SAP_01 column separated by ":".

The business component id that is inserted in the COMMON column can be any unique number.

The Agile item number and the SAP plant corresponding to the Agile site are inserted into the AGILE_01 column separated by "::".

Populating Cross-References

If you want to perform an initial data load, after the data load is performed in edge applications, then you must manually populate the ITEM_ITEMID and CHANGE_CHANGEID cross-reference tables after installing and configuring the re-built integration. You must manually add the cross-reference data into the ITEM_ITEMID and CHANGE_CHANGEID XRef tables by using SQL insert statements.

For each SAP item processed through initial loads, the following rows must be created in the XREF_DATA table:

- One row for Agile: AGILE_01
- One row for SAP: SAP_01
- One row for Common: COMMON

The physical XREF_DATA table is located in the database configured for FMW and SOA and contains the following columns:

XREF_DATA Table Format

Column Name	Description
XREF_TABLE_NAME	This column stores the XREF data types. For example: oramds:/apps/AIAMetaData/xref/ITEM_ITEMID.xref oramds:/apps/AIAMetaData/xref/CHANGE_CHANGEID.xref
XREF_COLUMN_NAME	This column stores the values indicating the edge applications and COMMON value: AGILE_01 SAP_01 COMMON
ROW_NUMBER	This column stores the GUID.
IS_DELETED	This column indicates whether the column is deleted or not. For example, 'N'.
LAST_MODIFIED	This column stores last modified time.

ITEM_ITEMID XRef Table Value Format

XRef_Column_Name	Value
COMMON	GUID
AGILE_01	Agile ItemID::Site For example, Part1::DETROIT
SAP_01	SAP ItemID::Plant ID For example, 60031::0001

Sample SQL Insert Statements for ITEM_ITEMID Table:

For SAP_01:

```
INSERT INTO XREF_DATA (XREF_TABLE_NAME, XREF_COLUMN_NAME, ROW_NUMBER,
VALUE, IS_DELETED, LAST_MODIFIED) VALUES ('
oramds:/apps/AIAMetaData/xref/ITEM_ITEMID.xref ', 'SAP_01',
'BD618D40B30C11DEBFA5D9490F57512E ', '532922::0001', 'N',
SYSTIMESTAMP)
```

For AGILE_01:

```
INSERT INTO XREF_DATA (XREF_TABLE_NAME, XREF_COLUMN_NAME, ROW_NUMBER,
VALUE, IS_DELETED, LAST_MODIFIED) VALUES ('
oramds:/apps/AIAMetaData/xref/ITEM_ITEMID.xref ', ' AGILE_01',
'BD618D40B30C11DEBFA5D9490F57512E', ' TM0101P::DETROIT', 'N',
SYSTIMESTAMP)
```

For COMMON:

```
INSERT INTO XREF_DATA (XREF_TABLE_NAME, XREF_COLUMN_NAME, ROW_NUMBER,
VALUE, IS_DELETED, LAST_MODIFIED) VALUES ('
oramds:/apps/AIAMetaData/xref/ITEM_ITEMID.xref ', ' COMMON ',
'BD618D40B30C11DEBFA5D9490F57512E',
'2d373037303636323539363632323735', 'N', SYSTIMESTAMP)
```

Sample Data for XRef ITEM_ITEMID Table

XRef_Table	XRef_Column	Row_Number	Value	IS_Deleted	Last_Modified
oramds:/apps/AIAMetaData/xref/ITEM_ITEMID.xref	SAP_01	BD618D40B3 0C11DEBFA5 D9490F57512E	532922 ::0001	N	2012-4-11.11.29. 54. 210000000
oramds:/apps/AIAMetaData/xref/ITEM_ITEMID.xref	AGILE_01	BD618D40B30C11D EBFA5D9490F575 12E	TM010 1P::DETROIT	N	2012-4-11.11.29. 54. 210000000

XRef_Table	XRef_Column	Row_Number	Value	IS_Deleted	Last_Modified
oramds:/apps/AIAMetaData/xref/ITEM_ITEMID.xref	COMMON	BD618D40B30C11 DEBFA 5D9490F57512E	2d37303730 3636323539 3636323237 35	N	2012-4-11.11.29. 54. 210000000

CHANGE_CHANGEID XRef Table Value Format

XRef_Column_Name	Value
COMMON	GUID
AGILE_01	Agile ItemID::Site For example, Part1::DETROIT
SAP_01	SAP ChangeID For example, 60031

Sample SQL Insert Statements for CHANGE_CHANGEID Table

For SAP_01:

```
INSERT INTO XREF_DATA (XREF_TABLE_NAME, XREF_COLUMN_NAME, ROW_
NUMBER, VALUE, IS_DELETED, LAST_MODIFIED) VALUES ('
oramds:/apps/AIAMetaData/xref/CHANGE_CHANGEID.xref ', ' SAP_01',
'C7D0C3D0740D11E1BFF50FA2D024BE6E', '92219', 'N', SYSTIMESTAMP)
```

For AGILE_01:

```
INSERT INTO XREF_DATA (XREF_TABLE_NAME, XREF_COLUMN_NAME, ROW_
NUMBER, VALUE, IS_DELETED, LAST_MODIFIED) VALUES ('
oramds:/apps/AIAMetaData/xref/CHANGE_CHANGEID.xref ', ' AGILE_01',
'C7D0C3D0740D11E1BFF50FA2D024BE6E ', ' C00374::V1', 'N', SYSTIMESTAMP)
```

For COMMON:

```
INSERT INTO XREF_DATA (XREF_TABLE_NAME, XREF_COLUMN_NAME, ROW_
NUMBER, VALUE, IS_DELETED, LAST_MODIFIED) VALUES ('
oramds:/apps/AIAMetaData/xref/CHANGE_CHANGEID.xref ', ' COMMON', '
C7D0C3D0740D11E1BFF50FA2D024BE6E ',
'2d38393233363334373536313436303 5', 'N',
SYSTIMESTAMP)
```

Sample Data for CHANGE_CHANGEID Table

XRef_Table	XRef_Column	Row_Number	Value	IS_Deleted	Last_Modified
oramds:/apps/AIAMetaData/xref/CHANGE_CHANGEID.xref	SAP_01	BD618D40B30C11D EBFA5D9490F57512E	92219	N	2012-4-11.11.29.54. 210000000
oramds:/apps/AIAMetaData/xref/CHANGE_CHANGEID.xref	AGILE_01	C7D0C3D0740D11E1BFF50FA2D024BE6E	C0037 4::DETROIT	N	2012-4-11.11.29.54. 210000000
oramds:/apps/AIAMetaData/xref/CHANGE_CHANGEID.xref	COMMON	C7D0C3D0740D11E1BFF50FA2D024BE6E	2d3839323336333437353631 34363035	N	2012-4-11.11.29.54. 210000000

Configuring the Integration

This pre-built integration uses various configuration parameters that control the behavior of the flow. The standard AIA XML configuration file, *AIAConfigurationProperties.xml*, is used for capturing configuration parameters. The Oracle AIA configuration file supports system-level configuration parameters, service-level parameters, and module configuration parameters. System-level parameters apply to all pre-built integrations running on the SOA suite. Service-level parameters are configured at the individual service levels, such as Application Business Connector Service (ABCS).

The configuration properties from the Agile Module and the SAP Module are listed separately in this section for the purpose of identification. The actual AIAConfigurationProperties.xml file on the Oracle AIA Server is a merger of both.

Configuration Parameters

This pre-built integration uses the following type of configuration parameters -

PIP-Level configuration parameters: PIP configuration parameters are implemented using the Oracle 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 a 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 must be controlled at the service level. These parameters can be captured using Oracle AIA service configuration parameters. Service configuration entry is identified by the service name, such as *CreateItemAgileReqABCSImpl*. The parameter names themselves are named using a cascaded naming convention as explained previously.

To upload to MDS:

1. Update the required property in the AIAConfigurationProperties.xml file
2. Go to <AIA_HOME>\aia_instances\AIA1115\config' path and open the UpdateMetaDataDP.xml file. Add <include name="config/AIAConfigurationProperties.xml"/>.
3. Go to <AIA_HOME>\Infrastructure\Install\config and run the following command:

```
ant –f UpdateMetaData.xml
```

Note: Whenever the AIAConfigurationProperties.xml file is updated, the file must be updated in MDS.

Agile Configurations

Properties	(default) Value/Setting	Description
Module Name	Agile	
MULTISITE_ENABLED	TRUE	When set to True, the sites specified in Sites Tab of Items are used to determine the Plants in SAP to which they are mapped. When set to False, Page2 Multilist01attribute is used to determine the Plants in SAP to which the Item will be extended to.
Item.UnitCostAttribute	Site.Numeric01	Determines the attribute to which the unit cost from SAP would be updated in Agile.
Change.TransferStatusAttribute	PageTwo.Text02	Determines the attribute to which the transfer status of a Change should be written. When the Change flow is from Agile to SAP, the possible values are "Transferred" or "Errored". When the Change flow is from SAP to Agile, the value would be the same as that of the Status of the Change in all the Plants of SAP.

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	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
Routing.ItemABSService.AGILE_01.EndpointURI	http://${agile.host}:${agile.port}/Agile/integration/services/ItemABS	Route to ItemABS
ItemResponseEBS.UpdateItemListResponse.RouteToCAVS	FALSE	Route to CAVS is set to TRUE
Routing.ItemResponseEBS.UpdateItemListResponse.CAVS.EndpointURI	http://${fp.server.soaserverhostname}:${fp.server.soaserverport}/ AIAValidationSystemServlet/syncresponsesimulator	If Route to CAVS set as true this endpoint URI will trigger
ServiceName	ProcessEngineeringChangeOrderAgileReqABCSImpl	
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.SystemID	AGILE_01	

Properties	(default) Value/Setting	Description
Routing.ChangeABSService.RouteToCAVS	False	
Routing.ChangeABSService.AGILE_01.EndpointURI	http://${agile.host}:${agile.port}/Agile/integration/services/ChangeABS	
Routing.EngineeringChangeOrderEBS.CreateEngineeringChangeOrderList.RouteToCAVS	False	CAVS property set to false
Routing.EngineeringChangeOrderEBS.CreateEngineeringChangeOrderList.CAVS.EndpointURI	http://${fp.server.soaserverhostname}:${fp.server.soaserverport}/AIAValidationSystemServlet/syncresponsesimulator	CAVS end point URI

Note: Multisite_Enabled property is governed by Distributed Processing aspects covered in Chapter 2: Process Integration for ECO/PREL. When it is set to TRUE (default), the Item.UnitCostAttribute is set to Site Tab Flex Attributes. You can use Numeric, Text or Money flex fields in the Site tab for these settings and is denoted by the first element, Site. For example, if Site.Numeric01 is set to FALSE, all these attributes are set to Page2 or Page3 flex fields. Hence, the settings 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.

SAP Configurations

Properties	(Default) Value/Setting	Description
ModuleName	SAP	
LANG_LOCALE	en-US	Used to specify the language code and locale of the SAP system
GLOBAL_LEVEL_BOM	FALSE	When this property is set to FALSE, in SAP the BOM are created at plant level and not at the global level. If set to TRUE, the BOM will be created at the global level.
DEFAULT_SYSTEM_ID	SAP_01	Used to get the default XREF Target column name when TargetId is empty in incoming EBM. It should be kept SAP_01 as default for this release
MATERIAL_TYPE	FERT	Used to specify the material type in SAP to be “Finished Product”
SAP46_N_ABOVE	TRUE	Specifies that the version of SAP used is greater than 4.6.

Properties	(Default) Value/Setting	Description
BOM_USAGE	1	BOM_USAGE denotes the usage of the BOM. 1indicates production level BOM. This value is dependent on the business process implemented at the client site. Some other values are: 2 indicates engineering/design BOM 3 indicates universal BOM 4 indicates plant maintenance BOM
ITEM_CATEGORY	L	ITEM_CATEGORY denotes the category of the item that is added to the BOM. L indicates stock item. This value is dependent on the business process implemented at the client site. Some other values are: N indicates non-stock item T denotes text item D denotes document item
ServiceName	ReserveItemSAPProvABCS	
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.POSTPROCESSITEMEBM	False	User exit for the post-process Item EBM should be called or not
ABCSEXTENSION.PREPROCESSITEMEBM	False	User exit for the pre-process Item EBM should be called or not
Default.SystemID	SAP_01	Used to get the default XREF Target column name when TargetId is empty in incoming EBM. It should be kept SAP_01 as default for this release
Routing.ItemResponseEBS.RouteToCAVS	False	Route to CAVS, if set as True
Routing.ItemResponseEBS.CAVS.EndpointURI	http://${fp.server.soaserverhostname}:${fp.server.soaserverport}/AIAValidationSystemServlet/asyncresponserecipient	CAVS Endpoint URI
Routing.GenerateItemNumberService_Reserve.RouteToCAVS	False	Route to CAVS, if set as True
Routing.GenerateItemNumberService_Reserve.SAP_01.EndpointURI	http://${fp.server.soaserverhostname}:${fp.server.soaserverport}/soa-infra/services/default/ReserveItemSer	ReserveItemSAP runtime target endpoint URI

Properties	(Default) Value/Setting	Description
	vice/ReserveItemService_ep	
Routing.GenerateItemNumberService_Reserve.CAVS.EndpointURI	http://${fp.server.soaserverhostname}:${fp.server.soaserverport}/AIAValidationSystemServlet/syncresponsesimulator	CAVS Enpoint URI
TRACE.LOG.ENABLED	False	Use tracelog for the flow
ServiceName	CreateEngineeringChangeOrderListEBF	
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.POSTPROCESSITEMEBM	False	User exit for the post-process Item EBM should be called or not
ABCSEXTENSION.PREPROCESSITEMEBM	False	User exit for the pre-process Item EBM should be called or not
Default.SystemID	SAP_01	Used to get the default XREF Target column name when TargetId is empty in incoming EBM. It should be kept SAP_01 as default for this release
Routing.ItemEBSV2.RouteToCAVS	False	Route to CAVS, if set as True
Routing.ItemEBSV2.SAP_01.EndpointURI	http://${fp.server.soaserverhostname}:${fp.server.soaserverport}/soa-infra/services/default/ItemEBSV2/ItemEBSV2_ep	ItemEBS runtime target endpoint URI
Routing.ItemEBSV2.CAVS.EndpointURI	http://${fp.server.soaserverhostname}:${fp.server.soaserverport}/AIAValidationSystemServlet/syncresponsesimulator	CAVS Endpoint URI
Routing.EngineeringChangeOrderResponseEBS.RouteToCAVS	False	Route to CAVS, if set as True
Routing.EngineeringChangeOrderResponseEBS.SAP_01.EndpointURI	http://${fp.server.soaserverhostname}:${fp.server.soaserverport}/soa-infra/services/default/EngineeringChangeOrderResponseEBS!1.0/EngineeringChangeOrderResponseEBS_ep	EngineeringChangeOrderEBS runtime target endpoint URI
Routing.EngineeringChangeOrderRes	http://${fp.server.soaserverhostname}:	CAVS Endpoint URI

Properties	(Default) Value/Setting	Description
ponseEBS.CAVS.EndpointURI	${fp.server.soaserverport}/AIAValidationSystemServlet/ asyncresponserecipient A	
TRACE.LOG.ENABLED	False	Use tracelog for the flow
ServiceName	CreateEngineeringChangeOrderListSAPProvABCS	
ABCSEXTENSION.PREPROCESSABM	False	User exit for the pre-process ABM should be called or not
ABCSEXTENSION.POSTPROCESSABM	False	User exit for the post-process ABM should be called or not
ABCSEXTENSION.PREPROCESSEBM	False	User exit for the pre-process EBM should be called or not
ABCSEXTENSION.POSTPROCESSEBM	False	User exit for the post-process EBM should be called or not
Default.SystemID	SAP_01	Used to get the default XREF Target column name when TargetId is empty in incoming EBM. It should be kept SAP_01 as default for this release
Routing.CreateEngineeringChangeOrderService.RouteToCAVS	false	Route to CAVS, if set as True
Routing.CreateEngineeringChangeOrderService.SAP_01.EndpointURI	http://${fp.server.soaserverhostname}:${fp.server.soaserverport}/soa-Infra/services/default/ECOService/ECORouterService_ep	CreateEngineeringChangeOrderService runtime target endpoint URI
Routing.CreateEngineeringChangeOrderService.CAVS.EndpointURI	http://${fp.server.soaserverhostname}:${fp.server.soaserverport}/AIAValidationSystemServlet/syncresponsesimulator	CAVS Endpoint URI
TRACE.LOG.ENABLED	False	Use tracelog for the flow
ServiceName	SyncItemListSAPProvABCS	
ABCSEXTENSION.PREPROCESSREVLEVELABM	False	User exit for the pre-process ABM should be called or not
ABCSEXTENSION.POSTPROCESSREVLEVELABM	False	User exit for the post-process ABM should be called or not
ABCSEXTENSION.PREPROCESSSAVEDDATAABM	False	User exit for the pre-process SAVED DATA ABM should be called or not
ABCSEXTENSION.POSTPROCESSSAVEDDATAABM	False	User exit for the post-process SAVED DATA ABM should be called or not
ABCSEXTENSION.PREPROCESSREFMATERIALABM	False	User exit for the pre-process REF MATERIAL ABM should be called or

Properties	(Default) Value/Setting	Description
		not
ABCSEXTENSION.POSTPROCESSREFMATERIALABM	False	User exit for the post-process REF MATERIAL 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
Default.SystemID	SAP_01	Used to get the default XREF Target column name when TargetId is empty in incoming EBM. It should be kept SAP_01 as default for this release
Routing.Call_ItemListResponseEBS.RouteToCAVS	False	Route to CAVS, if set as True
Routing.Call_ItemListResponseEBS.CAVS.EndpointURI	http://${fp.server.soaserverhostname}:${fp.server.soaserverport}/AIAValidationSystemServlet/asyncresponserecipient	CAVS Endpoint URI
Routing.CreateReferencematerial.RouteToCAVS	False	Route to CAVS, if set as True
Routing.CreateReferencematerial.SAP_01.EndpointURI	http://${fp.server.soaserverhostname}:${fp.server.soaserverport}/soa-infra/services/default/ItemServiceESB/ItemServiceESB_ep	CreateReferenceMaterial runtime target endpoint URI
Routing.CreateReferencematerial.CAVS.EndpointURI	http://${fp.server.soaserverhostname}:${fp.server.soaserverport}/AIAValidationSystemServlet/syncresponsesimulator	CAVS Endpoint URI
Routing.CreateDetailReferencematerial.RouteToCAVS	False	Route to CAVS, if set as True
Routing.CreateDetailReferencematerial.SAP_01.EndpointURI	http://${fp.server.soaserverhostname}:${fp.server.soaserverport}/soa-infra/services/default/ItemServiceESB/ItemServiceESB_ep	CreateDetailReferenceMaterial runtime target endpoint URI
Routing.CreateDetailReferencematerial.CAVS.EndpointURI	http://${fp.server.soaserverhostname}:${fp.server.soaserverport}/AIAValidationSystemServlet/syncresponsesimulator	CAVS Endpoint URI
Routing.CreateItemService.RouteToCAVS	False	Route to CAVS, if set as True

Properties	(Default) Value/Setting	Description
Routing.CreateItemService.SAP_01.EndpointURI	http:// ${fp.server.soaserverhostname}:${fp.server.soaserverport}/soa-Infra/services/default/ItemServiceESB/ItemServiceESB_ep	CreateItemService runtime target endpoint URI
Routing.CreateItemService.CAVS.EndpointURI	http:// ${fp.server.soaserverhostname}:${fp.server.soaserverport}/AIAValidationSystemServlet/syncresponsesimulator	CAVS Endpoint URI
Routing.CreateItemRevisionService.RouteToCAVS	False	Route to CAVS, if set as True
Routing.CreateItemRevisionService.SAP_01.EndpointURI	http:// ${fp.server.soaserverhostname}:${fp.server.soaserverport}//soa-Infra/services/default/ItemServiceESB/ItemServiceESB_ep	CreateItemRevisiopnService runtime target endpoint URI
Routing.CreateItemRevisionService.CAVS.EndpointURI	http:// ${fp.server.soaserverhostname}:${fp.server.soaserverport}/AIAValidationSystemServlet/syncresponsesimulator	CAVS Endpoint URI
TRACE.LOG.ENABLED	False	Use tracelog for the flow
ServiceName	ProcessBillOfMaterialsListSAPProvABCS	
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
Default.SystemID	SAP_01	Used to get the default XREF Target column name when TargetId is empty in incoming EBM. It should be kept SAP_01 as default for this release
Routing.GlobalBillOfMaterialExistenceCheckService.RouteToCAVS	False	Route to CAVS, if set as True
Routing.GlobalBillOfMaterialExistenceCheckService.SAP_01.EndpointURI	http://${fp.server.soaserverhostname}: ${fp.server.soaserverport}/soa-infra/services/default/BOMServiceESB/BOMServiceESB_ep	Global BillOfMaterialExistenceCheckService runtime target endpoint URI

Properties	(Default) Value/Setting	Description
Routing.GlobalBillOfMaterialExistenceCheckService.CAVS.EndpointURI	http://${fp.server.soaserverhostname}:${fp.server.soaserverport}/AIAValidationSystemServlet/syncresponsesimulatorr	CAVS Endpoint URI
Routing.BillOfMaterialExistenceCheckService.RouteToCAVS	False	Route to CAVS, if set as True
Routing.BillOfMaterialExistenceCheckService.SAP_01.EndpointURI	http://${fp.server.soaserverhostname}:${fp.server.soaserverport}/soa-infra/services/default/BOMServiceESB/BOMServiceESB_ep	BillOfMaterialExistenceCheckService runtime target endpoint URI
Routing.BillOfMaterialExistenceCheckService.CAVS.EndpointURI	http://${fp.server.soaserverhostname}:${fp.server.soaserverport}/AIAValidationSystemServlet/syncresponsesimulator	CAVS Endpoint URI
TRACE.LOG.ENABLED	False	Use tracelog for the flow
ServiceName	CreateBillOfMaterialsListSAPProvABCSImpl	
ABCSEXTENSION.PREPROCESSBOMEXISTENCECHECKABM	False	User exit for the pre-process BOM Existence Check ABM should be called or not
ABCSEXTENSION.POSTPROCESSBOMEXISTENCECHECKABM	False	User exit for the post-process BOM Existence Check ABM should be called or not
ABCSEXTENSION.PREPROCESSBOMPLANTEXTENSIONABM	False	User exit for the pre-process BOM Plant Existence Check ABM should be called or not
ABCSEXTENSION.POSTPROCESSBOMPLANTEXTENSIONABM	False	User exit for the post-process BOM Plant Existence Check ABM should be called or not
ABCSEXTENSION.PREPROCESSBOMCREATEABM	False	User exit for the pre-process BOM Create ABM should be called or not
ABCSEXTENSION.POSTPROCESSBOMCREATEABM	False	User exit for the post-process BOM Create 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
Default.SystemID	SAP_01	Used to get the default XREF Target column name when TargetId is empty in incoming EBM. It should be kept SAP_01 as default for this release

Properties	(Default) Value/Setting	Description
Routing.BillOfMaterialsResponseEBS.RouteToCAVS	False	Route to CAVS, if set as True
Routing.BillOfMaterialsResponseEBS.CAVS.EndpointURI	http://${fp.server.soaserverhostname}:${fp.server.soaserverport}/AIAValidationSystemServlet/asyncresponserecipient	CAVS Endpoint URI
Routing.CreateGlobalBillOfMaterialService.RouteToCAVS	False	Route to CAVS, if set as True
Routing.CreateGlobalBillOfMaterialService.SAP_01.EndpointURI	http://${fp.server.soaserverhostname}:${fp.server.soaserverport}/soa-infra/services/default/BOMServiceESB/BOMServiceESB_ep	CreateBillOfMaterialService runtime target endpoint URI
Routing.CreateGlobalBillOfMaterialService.CAVS.EndpointURI	http://${fp.server.soaserverhostname}:${fp.server.soaserverport}/AIAValidationSystemServlet/syncresponsesimulator	CAVS Endpoint URI
Routing.CreateBillOfMaterialService.RouteToCAVS	False	Route to CAVS, if set as True
Routing.CreateBillOfMaterialService.SAP_01.EndpointURI	http://${fp.server.soaserverhostname}:${fp.server.soaserverport}/soa-infra/services/default/BOMServiceESB/BOMServiceESB_ep	CreateBillOfMaterialService runtime target endpoint URI
Routing.CreateBillOfMaterialService.CAVS.EndpointURI	http://${fp.server.soaserverhostname}:${fp.server.soaserverport}/AIAValidationSystemServlet/syncresponsesimulator	CAVS Endpoint URI
Routing.BOMPlantExtensionService.RouteToCAVS	False	Route to CAVS, if set as True
Routing.BOMPlantExtensionService.SAP_01.EndpointURI	http://${fp.server.soaserverhostname}:${fp.server.soaserverport}/soa-infra/services/default/BOMServiceESB/BOMServiceESB_ep	BOMPlantExtensionService runtime target endpoint URI
Routing.BOMPlantExtensionService.CAVS.EndpointURI	http://${fp.server.soaserverhostname}:${fp.server.soaserverport}/AIAValidationSystemServlet/syncresponsesimulator	CAVS Endpoint URI
Routing.BillOfMaterialExistenceCheckService.RouteToCAVS	False	Route to CAVS, if set as True
Routing.BillOfMaterialExistenceCheckService.SAP_01.EndpointURI	http://${fp.server.soaserverhostname}:${fp.server.soaserverport}/soa-infra/services/default/BOMServiceESB/BOMServiceESB_ep	BillOfMaterialExistenceCheckService runtime target endpoint URI

Properties	(Default) Value/Setting	Description
Routing.BillOfMaterialExistenceCheckService.CAVS.EndpointURI	http://${fp.server.soaserverhostname}:${fp.server.soaserverport}/AIAValidationSystemServlet/syncresponsesimulator	CAVS Endpoint URI
TRACE.LOG.ENABLED	False	Use tracelog for the flow
ServiceName	UpdateBillOfMaterialsListSAPProvABCSImpl	
ABCSEXTENSION.PREPROCESSBOMEXISTENCECHECKABM	False	User exit for the pre-process BOM Existence Check ABM should be called or not
ABCSEXTENSION.POSTPROCESSBOMEXISTENCECHECKABM	False	User exit for the post-process BOM Existence Check ABM should be called or not
ABCSEXTENSION.PREPROCESSBOMPLANTEXTENSIONABM	False	User exit for the pre-process BOM Plant Existence Check ABM should be called or not
ABCSEXTENSION.POSTPROCESSBOMPLANTEXTENSIONABM	False	User exit for the post-process BOM Plant Existence Check ABM should be called or not
ABCSEXTENSION.PREPROCESSBOMUPDATEABM	False	User exit for the pre-process BOM Update ABM should be called or not
ABCSEXTENSION.POSTPROCESSBOMUPDATEABM	False	User exit for the post-process BOM Update 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
Default.SystemID	SAP_01	Used to get the default XREF Target column name when TargetId is empty in incoming EBM. It should be kept SAP_01 as default for this release
Routing.BillOfMaterialsResponseEBS.RouteToCAVS	False	Route to CAVS, if set as True
Routing.BillOfMaterialsResponseEBS.CAVS.EndpointURI	http://${fp.server.soaserverhostname}:${fp.server.soaserverport}/AIAValidationSystemServlet/asyncresponserecipient	CAVS Endpoint URI
Routing.UpdateGlobalBillOfMaterialService.RouteToCAVS	False	Route to CAVS, if set as True
Routing.UpdateGlobalBillOfMaterialService.SAP_01.EndpointURI	http://${fp.server.soaserverhostname}:${fp.s	UpdateBillOfMaterialService runtime target endpoint URI

Properties	(Default) Value/Setting	Description
	erver.soaserverport}/soa-infra/services/default/BOMServiceESB/BOMServiceESB_ep	
Routing.UpdateGlobalBillOfMaterialService.CAVS.EndpointURI	http:// ${fp.server.soaserverhostname}:${fp.server.soaserverport}/AIAValidationSystemServlet/syncresponsesimulator	CAVS Endpoint URI
Routing.BillOfMaterialExistenceCheckService.RouteToCAVS	False	Route to CAVS, if set as True
Routing.BillOfMaterialExistenceCheckService.SAP_01.EndpointURI	http:// ${fp.server.soaserverhostname}:${fp.server.soaserverport}/soa-infra/services/default/BOMServiceESB/BOMServiceESB_ep	BillOfMaterialExistenceCheckService runtime target endpoint URI
Routing.BillOfMaterialExistenceCheckService.CAVS.EndpointURI	http:// ${fp.server.soaserverhostname}:${fp.server.soaserverport}/AIAValidationSystemServlet/syncresponsesimulator	CAVS Endpoint URI
Routing.BOMPlantExtensionService.RouteToCAVS	False	Route to CAVS, if set as True
Routing.BOMPlantExtensionService.SAP_01.EndpointURI	http:// ${fp.server.soaserverhostname}:${fp.server.soaserverport}/soa-infra/services/default/BOMServiceESB/BOMServiceESB_ep	BOMPlantExtensionService runtime target endpoint URI
Routing.BOMPlantExtensionService.CAVS.EndpointURI	http:// ${fp.server.soaserverhostname}:${fp.server.soaserverport}/AIAValidationSystemServlet/syncresponsesimulator	CAVS Endpoint URI
Routing.UpdateBillOfMaterialService.RouteToCAVS	False	Route to CAVS, if set as True
Routing.UpdateBillOfMaterialService.SAP_01.EndpointURI	http:// ${fp.server.soaserverhostname}:${fp.server.soaserverport}/soa-infra/services/default/BOMServiceESB/BOMServiceESB_ep	UpdateBillOfMaterialService runtime target endpoint URI
Routing.UpdateBillOfMaterialService.CAVS.EndpointURI	http:// ${fp.server.soaserverhostname}:${fp.server.soaserverport}/AIAValidationSystemServlet/syncresponsesimulator	CAVS Endpoint URI
TRACE.LOG.ENABLED	False	Use trace log for the flow
ServiceName	UpdateItemListSAPReqABCSImpl	

Properties	(Default) Value/Setting	Description
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.POSTPROCESSABM	False	User exit for the post-process ABM should be called or not
ABCSEXTENSION.POSTPROCESSEBM	False	User exit for the post-process EBM should be called or not
Default.SystemID	SAP_01	Used to get the default XREF Target column name when TargetId is empty in incoming EBM. It should be kept SAP_01 as default for this release
Routing.ItemEBSV2.RouteToCAVS	False	Route to CAVS, if set as True
Routing.ItemEBSV2.SAP_01.EndpointURI	http:// ${fp.server.soaserverhostname}:${fp.server.soaserverport}/soa-infra/services/default/ItemEBSV2/ItemEBSV2_ep	ItemEBS runtime target endpoint URI
Routing.ItemEBSV2.CAVS.EndpointURI	http://${fp.server.soaserverhostname}:${fp.server.soaserverport}/AIAValidationSystemServlet/syncresponsesimulator	CAVS Endpoint URI
ServiceName	UpdateItemListSAPReqABCS	
ABCSEXTENSION.POSTPROCESSABM	False	User exit for the post-process ABM should be called or not
ABCSEXTENSION.PREPROCESSABM	False	User exit for the pre-process ABM should be called or not
Default.SystemID	SAP_01	
TRACE.LOG.ENABLED	False	
ServiceName	ProcessBillOfMaterialsListSAPProvABCSImpl	
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
Default.SystemID	SAP_01	Used to get the default XREF Target column name when TargetId is empty in incoming EBM. It should be kept SAP_01 as default for this release
Routing.Retreive_BOMData.RouteToCAVS	False	Route to CAVS, if set as True
Routing.Retreive_BOMData.SAP_01.EndpointURI	http://${fp.server.soaserverhostname}:${fp.server.soaserverport}/soa-infra/services/default/BOMServiceESB/BOMServiceESB_ep	ProcessBillOfMaterialsListImpl runtime target endpoint URI
Routing.Retreive_BOMData.CAVS.EndpointURI	http://${fp.server.soaserverhostname}:${fp.server.soaserverport}/AIAValidationSystemServlet/syncresponsesimulator	CAVS Endpoint URI
TRACE.LOG.ENABLED	False	Use tracelog for the flow

Setting up National Language Support

You must perform the following steps to set up National Language Support (NSL):

- Set up NLS in Agile
- Set up NLS in FMW for Agile
- Edit AIAConfigurations
- Set Up DVMs
- Set up the DVMs for NLS

Setting up NLS in Agile

To set up NLS in Agile:

1. In Agile Java Client, for each subscriber of MCO, SCO, ECO, go to the subscriber details page and set the language.
2. Change the language preference of all the users creating CO and Items, including integration user.

Setting up NLS in FMW for Agile

To set up NLS in FMW for Agile:

1. Edit the following XSL file and replace the hard-coded string 'Preliminary' to the preferred language equivalent coming from Agile List values

   \AIA_HOME\services\core\Agile\UtilityServices\QueueProcessorServiceImpl\AgileData_to_AgileCreateEngineeringChangeOrderListABM_Impl.xsl

2. Edit the following XSL file and replace the following values:

   \AIA_HOME\services\core\Agile\RequesterABCS\ProcessEngineeringChangeOrderAgileReqABCSImpl\
   AgileCreateEngineeringChangeOrderListABM_to_CreateEngineeringChangeOrderListEBM_Impl.xsl

   a. Replace the hard-coded string *Preliminary* to the preferred language equivalent coming from Agile list values.

   b. Replace the hard-coded string *SCO* to the preferred language equivalent coming from Agile list values.

3. Edit the following XSL file and replace hard coded strings *Errored* with the preferred language equivalent

   \AIA_HOME\services\core\Agile\RequesterABCS\ProcessEngineeringChangeOrderAgileReqABCSImpl\
   AgileCreateEngineeringChangeOrderListABM_to_AgileUpdateEngineeringChangeOrderListABM_Impl.xsl

4. Edit the following XSL file and replace the following values:

   \AIA_HOME\services\core\Agile\RequesterABCS\ProcessEngineeringChangeOrderAgileReqABCSImpl\
   UpdateEngineeringChangeOrderListEBM_to_AgileUpdateEngineeringChangeOrderListABM_Impl.xsl

   a. Replace the hard coded string Transferred with its preferred language equivalent.

   b. Replace the hard coded string ‘Errored’ with its preferred language equivalent.

Note: Use an UTF-8 based editor such as JEdit to perform these tasks. Choose UTF-8 as the character set while loading the file to edit. Set File transfer protocol (FTP) transfer mode to *Binary* while uploading these files to Oracle AIA components.

Editing AIAConfigurations

Edit the property LANG_LOCALE under module Agile to the preferred language. The language value is available under column COMMON in the DVM - LANGUAGE_CODE.

Setting up DVMs

To set up DVMs, enter the preferred language values under Agile column, AGILE_01 in the following DVMs:

ITEM_PRIMARYCLASSIFICATIONCODE

ITEM_STATUS_CODE

ITEM_UOM_CODE

ECO_STATUS_CODE

After all the xsls are modified, the corresponding process flows must be redeployed for the xsl changes to take effect.

Setting up the DVMs for NLS

To set up DVMs for NLS, configure SAP_01 with the appropriate language code of SAP in the *LANGUAGE_CODE* DVM column.

Working with Domain Value Maps

DVMs are a standard feature of the Oracle SOA Suite. They enable you to equate lookup codes and other static values across applications, for example, FOOT and FT or US and USA.DVMs are static in nature, though administrators can add additional maps as required. Transactional business processes never update DVMs; they only read from them. DVMs are stored in XML files and cached in memory at run time.

DVM types are seeded for the Agile PLM Integration Pack for SAP: Design to Release pre-built integration flows and administrators can extend the list of mapped values by adding more maps. The DVM data should be synchronized with what the participating applications use. This synchronization should occur before any initial loads are run or any incremental transactional flows are initiated.

During installation, the DVMs used for the pre-built integration are imported with default data mappings. The values mapped by these DVMs must be changed as required. Many DVMs are seeded and do not require any changes. Because most of the Agile PLM attributes being mapped are list values, the Agile PLM data is not seeded and should be changed accordingly and then updated to MDS.

Out-of-the-Box DVMs

During installation, the DVMs used for the pre-built integration 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. Hence, it should be changed accordingly and then updated in the MDS.

Sample DVMs

Some of the most used DVMs, with their out-of-the-box (OOTB) values, are listed here. 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.

Note: You can only add more rows of value mappings and must not change the DVM name, the column names, or the number of columns.

ECO_ENGINEERINGCHANGEORDERLINE_STATUS_CODE

Used for SAP attribute - ECO_ATTR/ECO_REVISED_ITEM_TYPE/ECO_REVISED_ITEM_TYPE_ITEM/STATUS_NAME

SAP_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 SAP attribute - ECO_CHANGE_ORDER_TYPE/STATUS_NAME, ECO_REVISED_ITEM_TYPE/STATUS_NAME. Use for Agile's Status attribute of a Change.

SAP_01	COMMON	AGILE_01
Release	RELEASE	
EBS_Approval	EBS_APPROVAL	

SAP_01	COMMON	AGILE_01
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

This value is used for SAP Item attribute ITEM_OBJ/MAIN_OBJ_TYPE/INVENTORY_ITEM_STATUS_CODE. The Agile Item Lifecycle phase attribute is mapped.

SAP_01	COMMON	AGILE_01
A	A	
Concept	CONCEPT	
Design	DESIGN	Preliminary
Engineer	ENGINEER	Pilot
Inactive	INACTIVE	Inactive, 02
Lease	LEASE	
Non-Stock	NONSTOCK	
Nwe B	NWEB	

SAP_01	COMMON	AGILE_01
OPM	OPM	
Obsolete	OBSOLETE	Obsolete, 02
Pending	PENDING	
Phase-Out	PHASEOUT	
Production	PRODUCTION	Production, 03
Prototype	PROTOTYPE	Prototype
R&D	R&D	

AGILE_SITE_TARGET_MAPPING

DEFAULT_MASTER_ORG in SAP is specified here. This is used when the **Multisite_Enabled** property is set to *False* and no Plant is specified for the Item where it extends to.

When the **Multisite_Enabled** property is set to *True*, the Sites in Agile are mapped to various Plants in SAP. A Site may be mapped to multiple Plants in the SAP column with | delimiter.

AGILE_TARGET_SITE_MAPPING

The SAP Plants to Agile Sites are mapped. This is used for SAP to Agile flows. There is a one-to-one mapping between the SAP Plant to Agile Site.

List of DVMs

This table describes the domain value mappings and their description:

Domain Value Map	Description
ITEM_UOM_CODE:	This DVM is used to maintain the Unit of Measure values of Agile corresponding to SAP
REFERENCEMATERIALS:	We are maintaining this DVM to populate the mandatory parameters of SAP BasicData1, Basic data2, MRP1, MRP2, MRP3 views which are not being sent from Agile
LANGUAGE_CODE:	This DVM is used to populate the Multi languages, as of now we are dealing with only English language
ITEM_STATUS_CODE:	This DVM is used to maintain the Life Cycle Phase codes and Special Plant Material Status of SAP
ITEM_PRIMARYCLASSIFICATIONCODE:	This DVM is used to populate the Material type of SAP
ECO_STATUS_CODE:	The DVM is used to handle the Status Codes of Change Order in SAP.
AGILE_SITE_TARGET_MAPPING	This DVM is used to map SAP Plants to Agile Sites. This is used for Agile to SAP flows

Domain Value Map	Description
AGILE_TARGET_SITE_MAPPING	This DVM is used to map SAP Plants to Agile Sites. This is used for SAP to Agile flows
ECO_ENGINEERINGCHANGEORDERLINE_REVISEDBILLOFMATERIALS_BILLOFMATERIALSCOMPONENTITEM_CHANGETYPECODE:	We are using this DVM to recognize the Creation or updation of Bill of Materials in Agile

Application Interfaces

Application Interfaces are the web services and APIs that communicate and transact between Application and Integration Layers.

Agile PLM Interfaces

These are the Agile PLM interfaces used in this integration:

WSDLs

ItemABS.wsdl	Used to create an item in SAP
ItemABS.wsdl	Used to update an Item in SAP
ChangeABS.wsdl	Used to create an EngineeringChangeOrder in SAP

XSDs

ItemABM.xsd	Contains the Item Request and Response ABM
ItemABM.xsd	Contains the Update Item List Request and Response ABM
EngineeringChangeOrderABM.xsd	Contains CreateEngineeringChangeOrderListABM and CreateEngineeringChangeOrderListResponseABM

SAP Interfaces

These are the SAP interfaces used in this integration:

WSDLs

ReserveItemservice.wsdl	The interface to the SAP Process Item API
GenerateItemNumberService_Reserve.wsdl	BAPI_STDMATERIAL_GETINTNUMBER
ItemServiceESB.wsdl	The interface to the SAP Process Item API -

CreateReferencematerial.wsdl	BAPI_MATERIAL_GET_ALL
CreateItemService.wsdl	BAPI_MATERIAL_SAVEDATA
CreateDetailReferencematerial.wsdl	BAPI_MATERIAL_GET_DETAIL
CreateItemRevisionService.wsdl	CCAP_REV_LEVEL_MAINTAIN
BOMServiceESB.wsdl	The interface to the SAP Process BOM API
BOMPlantExtensionService.wsdl	CSAP_MAT_BOM_ALLOC_CREATE
CreateBillOfMaterialService.wsdl	CSAP_MAT_BOM_CREATE
BillOfMaterialExistenceCheckService.wsdl	BAPI_MAT_BOM_EXISTENCE_CHECK
BOMServiceESB.wsdl	The interface to the SAP Process BOM API -
BOMPlantExtensionService.wsdl	CSAP_MAT_BOM_ALLOC_CREATE
UpdateBillOfMaterialService.wsdl	CSAP_MAT_BOM_MAINTAIN
BillOfMaterialExistenceCheckService.wsdl	BAPI_MAT_BOM_EXISTENCE_CHECK
BOMServiceESB.wsdl	The interface to the SAP Process BOM API -
BillOfMaterialExistenceCheckService.wsdl	BAPI_MAT_BOM_EXISTENCE_CHECK
ECORouterService.wsdl	The interface to the SAP Process ECO API -
CreateEngineeringChangeOrderService.wsdl	CCAP_ECN_MAINTAIN
MATMAS01_receive.wsdl	The interface to the SAP Process Item API - MATMAS01
ProcessBOMService.wsdl	The interface to the SAP Process BOM API
ProcessBillOfMaterialsListImpl.wsdl	CSAP_MAT_BOM_READ

XSDs

BAPI_MAT_BOM_EXISTENCE_CHECK_request.xsd	Contains the BillOfMaterialExistenceCheck Request ABM
BAPI_MAT_BOM_EXISTENCE_CHECK_response.xsd	Contains the BillOfMaterialExistenceCheck Response ABM
CCAP_ECN_MAINTAIN_request.xsd	Contains the CreateEngineeringChangeOrderList Request ABM
CCAP_ECN_MAINTAIN_response.xsd	Contains the CreateEngineeringChangeOrderList Response ABM
MATMAS01_event.xsd	Contains the UpdateItemList ABM
BAPI_STDMATERIAL_GETINTNUMBER_request.xsd	Contains the ReserveItem Request ABM
BAPI_STDMATERIAL_GETINTNUMBER_response.xsd	Contains the ReserveItem Response ABM

BAPI_MATERIAL_GET_ALL_request.xsd	Contains the CreateReferencematerial Request ABM
BAPI_MATERIAL_GET_ALL_response.xsd	Contains the CreateReferencematerial Response ABM
BAPI_MATERIAL_GET_DETAIL_request.xsd	Contains the CreateDetailReferencematerial Request ABM
BAPI_MATERIAL_GET_DETAIL_response.xsd	Contains the CreateDetailReferencematerial Response ABM
BAPI_MATERIAL_SAVEDATA_request.xsd	Contains the CreateItem Request ABM
BAPI_MATERIAL_SAVEDATA_response.xsd	Contains the CreateItem Response ABM
CCAP_REV_LEVEL_MAINTAIN_request.xsd	Contains the CreateItemRevision Request ABM
CCAP_REV_LEVEL_MAINTAIN_response.xsd	Contains the CreateItemRevision Response ABM
BAPI_MAT_BOM_EXISTENCE_CHECK_request.xsd	Contains the BillOfMaterialExistenceCheck Request ABM
BAPI_MAT_BOM_EXISTENCE_CHECK_response.xsd	Contains the BillOfMaterialExistenceCheck Response ABM
CSAP_MAT_BOM_ALLOC_CREATE_request.xsd	Contains the BOMPlantExtension Request ABM
CSAP_MAT_BOM_ALLOC_CREATE_response.xsd	Contains the BOMPlantExtension Response ABM
CSAP_MAT_BOM_CREATE_request.xsd	Contains the CreateBillOfMaterial Request ABM
CSAP_MAT_BOM_CREATE_response.xsd	Contains the CreateBillOfMaterial Response ABM
BAPI_MAT_BOM_EXISTENCE_CHECK_request.xsd	Contains the BillOfMaterialExistenceCheck Request ABM
BAPI_MAT_BOM_EXISTENCE_CHECK_response.xsd	Contains the BillOfMaterialExistenceCheck Response ABM
CSAP_MAT_BOM_ALLOC_CREATE_request.xsd	Contains the BOMPlantExtension Request ABM
CSAP_MAT_BOM_ALLOC_CREATE_response.xsd	Contains the BOMPlantExtension Response ABM
CSAP_MAT_BOM_MAINTAIN_request.xsd	Contains the UpdateBillOfMaterial Request ABM
CSAP_MAT_BOM_MAINTAIN_response.xsd	Contains the UpdateBillOfMaterial Response ABM
CSAP_MAT_BOM_READ_request.xsd	Contains the Read BOM Request ABM

CSAP_MAT_BOM_READ_response.xsd	Contains the Read BOM Response ABM

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 uses the role value to derive the user(s) that must be notified of the error. The Error Handling Framework then notifies the user(s) through their preferred notification method, puts the error in the user's Oracle Worklist as well as 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. An example of an Actor role is a task editable in the Error Console and is meant to be worked on, and resolved by the actor assigned to the task.

FYI Role: For Your Information (FYI) role associated with the error notification. This role receives FYI notifications for error scenarios occurring in Oracle AIA integration flows. An example of an FYI role is a customer service representative (CSR). The task is displayed in read-only view in the Error Console.

For more information about Oracle AIA error handling, see *Oracle Fusion Middleware Infrastructure Components and Utilities User's Guide for Oracle Application Integration Architecture Foundation Pack 11g Release 1*,"Setting Up Error Handling."

Chapter 6: Customizing the Transformations{ XE "transformations:customizing"}

This chapter discusses how to modify mappings between the attributes of the participating applications and how to plug in your own logic at pre-defined extension points (user exits) provided in the out-of-the-box (OOTB) Business Process Execution Language (BPEL) flow to validate, enrich, and transform data.

This chapter includes the following sections:

- Overview
- Customizations in Agile
- Customizations in SAP
- User Exits

Overview

You may require OOTB and user-defined attribute (UDA) mapping between applications, which are not covered as part of standard transformations. Considering this, the transformation files (XSLs) are externalized to help perform the following:

- Modify the OOTB transformations:
- Add new mappings for the Agile attributes to Enterprise Business Message (EBM) attributes.

  These Agile attributes are those that have not been mapped OOTB, including any of the flex-fields.

The transformations support the following:

- Multiple transformations involving multiple Application Business Objects (ABOs) and Application Business Messages (ABMs) as well as multiple Enterprise Business Objects EBOs) and EBMs.
- Flex-field mappings
- Customer provided XSLs for complex transformations that are not part of the standard transformations.

  It is required to support the transformations for user-defined EBO extensions (such as Custom tags) and in places where you have overriden the standard transformation logic.

Transformation Rules

These are the transformation rules:

- For a transformation from ABM into EBM, all the flex-fields (interchangeably user-defined attribute or flex-field attribute) go under the Specification Group element under the main EBM element with a matching type such as *ValueText* (for text values), *ValueNumeric* (for numeric values), and so on for carrying the values.

- For the transformations from EBM into ABM, depending upon the identification element of Specification and Specification Group, pull processing is performed to populate the UDAs.
- For Classification elements, such as *Item Classification*, the field values are translated using a configuration, into EBM and vice-versa. The configuration states are <<Name in Agile>> <<Name on EBO>> <<Name SAP ERP>>. For the remainder of the classification elements such as Part Type, Product Family, and so on fields must be packed and unpacked on or from the *XXCatalog* element on the EBM.

Customization in Agile

The XSL transformations in Agile Product Lifecycle Management (PLM) iIntegration are externalized, which means they are hosted on the implementation server under:

<AIA_Home>/AIAComponents/Transformations/Agile/

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

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

Note: The server must be restarted to bring the changes into effect.

Sample Customization

A few sample customizations have been provided at *<AIA_HOME>\PIPS\Core\Agile\Samples*. The following is a sample customization of Engineering Change Order (ECO) attributes:

```
User-Defined ECO - in Sites Tab Under Items
```

Mapping:

Agile	EBM	SAP
Item.Site:List01	corecomEBO:RevisedItem/corecomEBO:BaseUOMCode	Primary UOM Code

Template:

createEngineeringChangeOrderLines_With_SiteData

- createEngineeringChangeOrderLines_With_OrgData
- createEngineeringChangeOrderLines_With_DefaultMasterOrgData

Customization in SAP

To customize a Mapping File:

1. Pickup the <flow>_Custom XSL file from the Requester, Provider, or EBS service where the <flow> is the process name, such as ECO_CHANGE_ORDER_TYPE.

   a. If only flex-field transformation is 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. It is not necessary to set the custom transformations property in this case as the targets are already included in the base transformation file.

2. If the entire mapping is to be modified then you can copy the base mappings into the custom target (such as Custom in Create ECO flow) in the custom file and modify the mappings as desired.

3. After modifying the file you must set the Custom Transformations property in the Oracle Application Integration Architecture (Oracle AIA) configurations file and update the configuration file from the Oracle Metadata Services (Oracle MDS) respository

4. Restart the server after peforming the above three steps. The server must be restarted for the new transformations to load into the Java Virtual Machine (JVM).

Templates in the Custom Files

- **CreateEngineeringChangeOrderListEbmToAbm**: The following templates are used in the custom transformation files 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.

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

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

Replacing the Entire Mapping

To replace the entire mapping:

Use this template, **UpdateItemListABMToEBMCustom** for complete ABM to EBM transformation.

1. Change the *Use Custom Transformation* property in Oracle AIA Configurations file.

   File Path*: $AIA_HOME/config/AIAConfigurationProperties.xml*

Note: This step is not required if the flex field mapping is done.

2. Reload the Oracle AIA Configurations file.
3. Restart the Server.

Note: The Use Custom Transformation property is per integration flow and must be set accordingly.

User Exits

The Agile PLM Integration Pack for SAP: Design to Release integration provides user exits to allow custom transformations or filtration routines that you 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.

These are the user exit points for:

- Requester flows
- Provider flows

Requester Flows

User exits points for the Requester flow:

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

Provider Flows

User exits points for the Provider flow:

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

Implementing User Exits

To implement user exits:

1. Identify which OOTB flow is to be extended.
2. Identify the suitable exit point in the flow.
3. Develop the flow.
4. Configure the OOTB flow to include the newly developed flow.
5. Test the developed flow.

Appendix A: National Language Support for the Agile PLM Integration Pack for SAP: Design to Release

National Language Support (NLS) is implemented in the Agile PLM Integration Pack for SAP: Design to release integration.

This appendix includes the following sections:

- Setting Up Requestor Flows
- Setting Up Provider Flows

Setting Up Requester Flows

Before sending the EBM to Agile, the Language code is converted to the common value, which is configured from the **LANGUAGE_CODE** Domain Value Map (DVM).

Note: You must verify that the values are configured for the LANGUAGE_CODE DVM before the flows are triggered.

The **Language_Code** DVM is used to input parameters in the transformations, which has a default value as **en**-US.

Setting Up Provider Flows

The default language code in Agile is set to en-US. This code value is sent over to the provider Application Business Connector Service (ABCS) in SAP without any conversion.

However, from the Oracle AIA Release 2.4 pre-built integration onwards, this language code from Agile PLM is converted to a SAP system value using the **LANGUAGE_CODE** DVM.

Appendix B: Functionalities Available

The following functionalities are available across different versions of Agile PLM and SAP combination deployments:

i.

Features/Agile and SAP Versions	Agile 9.2.2.6 and SAP 4.7	Agile 9.2.2.6 and SAP ECC 6.0	Agile 9.2.2.7 and SAP 4.7	Agile 9.2.2.7 and SAP ECC 6.0	Agile 9.3,9.3.x, 9.3.1.1,9.3.1.2 and SAP 4.7	Agile 9.3, 9.3.x, 9.3.1.1,9.3.1.2 and ECC 6.0
NPR (Action PX)	N	N	N	N	N	N
NPR (Auto Number PX)	Y	Y	Y	Y	Y	Y
PREL(ECO Forward Flow (From Agile to SAP)	Y	Y	Y	Y	Y	Y
ECO Update Flow (From Agile to SAP)	Y	Y	Y	Y	Y	Y
Item Balance Update Flow (From SAP to Agile)	N	N	N	N	N	N
Item Operational Attribute Update Flow (Cost Attribute Updates only) (From SAP to Agile)	Y	Y	Y	Y	Y	Y
Queue Functionality (AQ (Database Persistent))	Y	Y	Y	Y	Y	Y
NPR (Action PX)	N	N	N	N	N	N
NPR (Auto Number PX)	Y	Y	Y	Y	Y	Y
PREL(ECO Forward Flow (From Agile to SAP)	Y	Y	Y	Y	Y	Y
ECO Update Flow (From Agile to SAP)	Y	Y	Y	Y	Y	Y

Features/Agile and SAP Versions	Agile 9.2.2.6 and SAP 4.7	Agile 9.2.2.6 and SAP ECC 6.0	Agile 9.2.2.7 and SAP 4.7	Agile 9.2.2.7 and SAP ECC 6.0	Agile 9.3,9.3.x, 9.3.1.1,9.3.1.2 and SAP 4.7	Agile 9.3, 9.3.x, 9.3.1.1,9.3.1.2 and ECC 6.0
Item Balance Update Flow (From SAP to Agile)	N	N	N	N	N	N
Item Operational Attribute Update Flow (Cost Attribute Updates only) (From SAP to Agile)	Y	Y	Y	Y	Y	Y
Queue Functionality (AQ (Database Persistent))	Y	Y	Y	Y	Y	Y

Appendix C: Agile PLM to SAP Entity Maps

This appendix contains information about Agile Product Lifecycle Management (PLM) to SAP maps.

This appendix includes the following sections:

- Bill of Materials Mappings
- Item Enterprise Business Object (EBO) Mappings
- Item Attribute Update Mappings
- Engineering Change Order (ECO) Enterprise Business Object (EBO) Mappings

Bill of Materials Mappings

Agile Entity Attribute	Bill of Materials EBO	SAP Entity: Attribute Group: Attribute
Part/Document. Title Block. Number	\\BillOfMaterialsEBO\ItemReference\Identification\ID	MATNR:Material Number
Changes.changeNumber	\\BillOfMaterialsEBO\Identification\ContextID	AENNR:Change Number
BOM.ITEM_NUMBER	\\BillOfMaterialsEBO\BillOfMaterialsComponentItem\ItemReference\ID	IDNRK:BOM Component
BOM.FIND_NUMBER	\\BillOfMaterialsEBO\BillOfMaterialsComponentItem\Identification\ContextID	POSNR:Position Number
REFDESIG.LABEL	\\BillOfMaterialsEBO\BillOfMaterialsComponentItem\ProcessingInstruction/Identification/ID	EBORT:Installation point for sub-item
BOM.QUANTITY	\\BillOfMaterialsEBO\BillOfMaterialsComponentItem\Quantity	Menge:Quantity
BOM.DESCRIPTION	\\BillOfMaterialsEBO\Identification\ItemReference\Description	POTX1:BOM item text (line 1)
	\\BillOfMaterialsEBO\BillOfMaterialsComponentItem\StorageUnitCode	MEINS:Comp Unit Of Measure
	\\BillOfMaterialsEBO\BillOfMaterialsComponentItem\ItemReference\ClassificationCode	POSTP:Item Category
	\\BillOfMaterialsEBO\BillOfMaterialsComponentItem\ItemReference\TypeCode	STLAN:BOM Usage

Agile Entity Attribute	Bill of Materials EBO	SAP Entity: Attribute Group: Attribute
BOM. Sites. Site Name	\\BillOfMaterialsEBO\Identification\ContextID	werks:Plant

Item EBO Mappings

Agile Entity Attribute	Item EBO	BAPI_MATERIAL_SAVEDATA	Comments
Part/Document. Title Block. Number	\\ItemEBO\Name	BAPIMATHEAD: MATNR: Material number:	Associate type Field Description
Part/Document. Title Block. Description	\\ItemEBO\Description	BAPI_MAKT: MATL_DESC Material Description:	
	\\ItemEBO\TypeCode	BAPIMATHEAD: MATl_TYPE Material Type	
Mass	\\ItemEBO\BaseUOMCode	BAPI_MARA: BASE_UOM: Base Unit of Measure	
	\\ItemEBO\PrimaryClassificationCode	BAPI_MARA: MATL_GROU:P Material group	
	\\ItemEBO\TemplateItemReference\ ClassificationCode	BAPI_MARA: MAT_GRP_SM: Material group	
	\\ItemEBO\ItemCatalog\Identification\ID	BAPI_MARA: ITEM_CAT: General Item Category Group:	
Part/Document. Title Block. Lifecycle Phase	\\ItemEBO\ItemLifecycleCharacteristics\Lifecycle Code	BAPI_MARA: MSTAE: X-Plant matl status	
Part/Document. Title Block. Description	\\ItemIdentification\BaseUOMCode\Revison/Label	BAPI_MLTX: TEXT_ID / TEXT_NAME: Basic Data text	
	\\itemEBO\ItemPhysicalCharacteristics\ VolumeMeasure	BAPI_MARM: VOLUME:	

Agile Entity Attribute	Item EBO	BAPI_MATERIAL_SAVEDATA	Comments
		VOLUMEUNIT: Volume Unit of Measure	
	\\ItemEBO\ItemPurchasingCharacteristics\UnitListPrice\Amount	BAPI_MBEW: STD_PRICE: Standard Price	
	\\ItemPhysicalCharacterstics\heightmeasure \\ItemPhysicalCharacterstics\lengthmeasure \\ItemPhysicalCharacterstics\widthmeasure	BAPI_MARM: HEIGHT,LENGTH,WIDTH	
		UNIT_DIM Unit of Dimension for height,length,width	
	\\ItemOrderManagementCharacteristics\custom/Delivery Plant	BAPI_MVKE: DELYG_PLNT: Delivery Plant	
Part/Document. Sites. Site Name	\\ItemIdentification\ContextID	BAPI_MARC: PLANT: PLANT	
	\\ItemEBO\InventoryLocation\SerialNumberSpecificationCode	BAPI_MARC: SERNO_PROF: Serial Number Profile	
	\\ItemEBO\relatedItem\relationshipCode	BAPI_MARA: BASIC_MATL WRKST_NEW	
	\\ItemEBO\itemClassification\classificationCode	BAPI_MARA: DIVISION: DIVISION	
	\\itemEBO\itemManufacturingCharacteristics\custom\laborOffice	BAPI_MARA: DSN_OFFICE LABOR	
	\\ItemIdentification\Gtin	BAPI_MARM: EAN_UPC: International Article Number	
	\\ItemIdentification\EANCode	BAPI_MARM: EAN_CAT: Category of International Article Number	
	\\ItemEBO\itemRevisionHistory\identification\id	BAPI_MARA: OLD_MAT_NO: Old material number	

Agile Entity Attribute	Item EBO	BAPI_MATERIAL_SAVEDATA	Comments
	\\ItemEBO\itemPurchasingCharacteristics\custom\purchasingGroup	BAPI_MARC: PUR_GROUP: Purchasing Group	
	\\ItemPurchasingCharacteristics/ AssetClassificationCode/Codetype	BAPI_MARC: COMM_CODE: Harmonized code	
	\\ItemPlanningCharacteristics\inventoryPlanning Code	BAPI_MARC: AVAILCHECK: Availability check	
	\\ItemAttachment\attachment\note\status	BAPI_MARA: DOC_CHG_NO: Document change number	
Part/Document. Changes. Lifecycle Phase	\\ItemEBO\ItemLifecycleCharacteristics\LifecyclePhaseCode	BAPI_MARA: SAL_STATUS: Cross distribution chain status	
	\\Item Order Management Characteristics\Custom\Transportation Group	BAPI_MARA: TRANS_GRP: Transportation group	
	\\ItemEBO\InventoryLocation\Identification\ID	BAPI_MARD: STGE_LOC: Storage Location	
Part/Document. Sites. Make/Buy	\\ItemPlanningCharacteristics\MakeOrBuycode		
	\\ItemPlanningCharacteristics\MakeOrBuycode	BAPI_MARC: SPPROCTYPE: Special procurement type	
	\\itemPlanningCharacteristics\ProcessingleadTimeCharacteristics\ fixedDuration	BAPI_MARC: GR_PR_TIME: Goods receipt processing time in days	
	\\ItemPlanningCharacteristics\ProcessingleadTimeCharacteristics\ postProcessingDuration	BAPI_MARC: PLND_DELRY: Planned Delivery Time	
	\\ItemEBO\Custom/Product Hierarchy	BAPI_MARA: PROD_HIER: Product Hierarchy	

Agile Entity Attribute	Item EBO	BAPI_MATERIAL_SAVEDATA	Comments
	\\ItemPhysicalCharacteristics\custom/Netweight	BAPI_MARA: NET_WEIGHT: Net weight	
		UNIT:OF_WT Unit Of weight: Weight	
	\\ItemEBO\ItemPhysicalCharacteristics\WeightMeasure	BAPI_MARAM: GROSS_WT: Gross Weight	
		UNIT_OF_WT UNIT OF WEIGHT: Unit of weight	
	\\itemPlanningCharacteristics\Custom\MRPController	BAPI_MARC: MRP_CTRLER: MRP Controller	
	\\ItemOrderManagementCharacteristics\Custom\LoadingGroup	BAPI_MARC: LOADINGGRP: Loading Group	
	\\ItemOrderManagementCharacteristics\custom\Sales Organization	BAPI_MVKE: SALES_ORG: Sales Organization	
	\\ItemEBO\Custom/Product Allocation	BAPI_MARA: PROD_ALLOC: Product Allocation Determination Procedure	
	\\ItemEBO\Custom\Authorisation Group	BAPI_MARA: AUTHORITYGROUP Authorization Group	
	\\ItemManufacturingCharacteristics\Custom\CountryOf Origin	BAPI_MARC: COUNTRYORI: Country of Origin	

Item Attribute Update Mappings

Agile Entity Attribute	Item EBO	MATMAS01

Agile Entity Attribute	Item EBO	MATMAS01
Part/Document. Title Block. Number	\\ItemEBO\Name	E1MARAM: MATNR Material number:
Part/Document. Title Block. Unit Cost Attribute	\\ItemEBO\ItemPurchasingCharacteristics\UnitListPrice\Amount	E1MBEWM: STPRS Standard Price E1MBEWM: VERPR Moving Average Price
Part/Document. Sites. Site Name	\\ItemIdentification\ContextID	E1MARCM: WERKS Plant

Engineering Change Order EBO Mappings

Agile Entity Attribute	Engineering Change Order EBO	CCAP_ECN_MAINTAIN SAP Entity: Attribute:Description	Comments
CHANGE.CHANGE_NUMBER	\\Identification\ ID	AENR_API01:CHANGE_NO:Change Number	Change Notice
CHANGE.DESCRIPTION	\\Identification\ Description	AENR_API01: DESCRIPT: Change number description	
CHANGE.RELEASE_DATE	InitiationDate	AENR_API01:VALID_FROM:Valid from date	
ECO/MCO/SCO. Affected Item. Effectivity Date	EngineeringChangeOrderLine/Effective Date	AEDT_API01:ALT_DATE:Alternative date external key	
CHANGE.REASON	\\Identification\Revision\Reason	AENR_API01:REASON_CHG:Reason for change	
	\\Identification\Status	AENR_API01:STATUS:Status of change number	
Changes .Affected Items. New Revision	\\EngineeringChangeOrderLine\EffectivityControlItem Reference\Revision\Number	REVLV:Revison level	
Agile Entity Attribute	Engineering Change Order EBO	CCAP_REL_LEVEL_MAINTAIN	
CHANGE.CHANGE_NUMBER	\\Identification\ ID	AENNR Change Number	
ECO/MCO/SCO. Affected Item. Item Number	\\EngineeringChangeOrderLine\RevisedItem	MATNR Material Number	
Change Orders Class.Affected Items. New Revision	\\EngineeringChangeOrderLine\EffectivityControlItemReference\Revision\Number	REVLV Revision level	

Appendix D: Queue Management

This appendix discusses the requirements and solution for Queue Management, the Queue Manager Services, as well as transformations, and describes the Queue User Interface (UI) application.

This appendix includes the following sections:

- Queue Management Requirements
- Queue Management Solution
- Queue Manager Services
- Transformations
- Queue User Interface (UI) Application

Queue Management Requirements

The Queue Management feature is based on the following requirements, which are leveraged using the Agile Content Service (ACS).

- Event to produce filtered payload to a file destination or JMS destination.

  ACS can produce the payload to either a File or JMS destination.

- Payload is defined using a standard XSD.

  The payload is based on filters configured for the ACS Event, which is defined by the Agile provided aXML schema definition.

- Files or JMS Messages produced by events are sequenced in the order in which the objects are released.

  ACS transmits the messages in the order in which the ATOs are released.

- Queue to manage messages.
- 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.

- 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.

For more information on Queue Management, see the *Agile PLM Integration Pack for SAP User Guide* at http://www.oracle.com/technology/documentation/

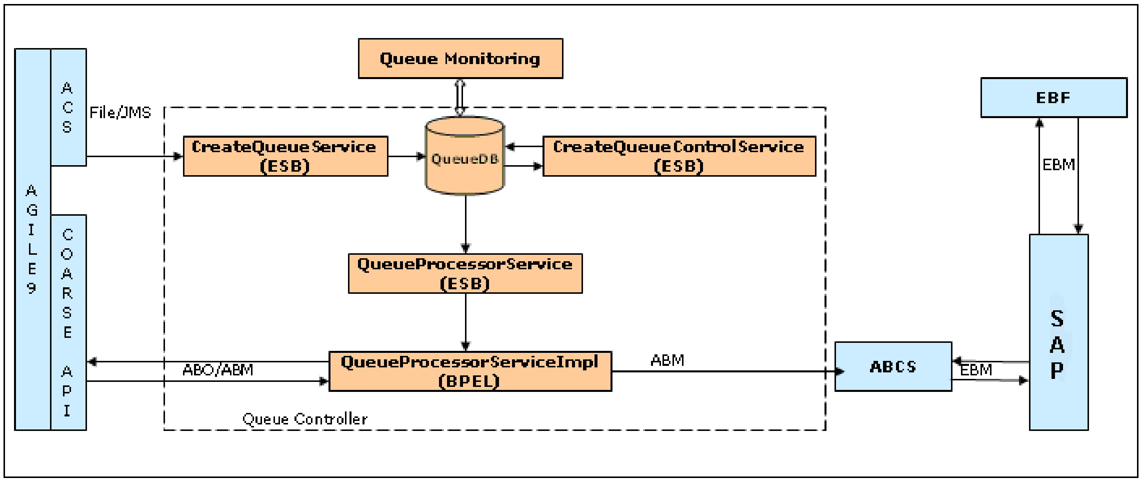
Queue Management Solution

The Queue Management Solution comprises of the following components:

Queue DB: The database persist the data related to 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 that monitors the Queue message status and supports re-ordering of priorities of the Queue Messages. Also provides the facility to resubmit the unprocessed messages.



Queue Monitoring

Queue Schema

To support the queue monitoring solution flow, a polling strategy similar to “PollingControlTableStrategy” is used. Two tables are used to manage the sequential processing and reordering of messages.

The QUEUE_TABLE contains all the queue messages that are being provided by the Event trigger. The QUEUE_CONTROL_TABLE stores the relevant information of the message from the QUEUE_TABLE, which has not been processed yet.

The Queue Manager must 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 from the Queue table is inserted into this table. This facilitates the sequential processing of messages. Also since all the pending messages are stored in the Queue table, they can be reordered.

Queue DB Details

The Queue Schema has the following tables:

Table Name	Description
ECO_QUEUE	This table holds the data of Process Engineering Change Order (ECO). The PROCESS_TYPE column is used as identifier for Process ECO.
ECO_QUEUE_CONTROL	This table stores the details about the rows that are currently under processing state.
ECO_QUEUE_TABLE_CRITERIA	This table contains the data required for a criterion. One set of criteria forms a filter.
ECO_QUEUE_TABLE_FILTER	This table contains the data required to form a filter, isuch as the criterion to be used to form a specific filter.
ECO_QUEUE_STATUS	This table holds the data to control the simultaneous processing and suspending the Queue. By changing the values in the ECO_QUEUE_STATUS column, the number of simultaneously processed ECOs can be changed.

The structure of ECO_QUEUE_STATUS table is:

ECO_QUEUE_STATUS_ID	ECO_QUEUE_STATUS	Description
1	1 or 0	The status of the Queue - in suspended or resume mode.
2	1	The count of rows that can be processed simultaneously for Process ECO. A value of 1 means sequential processing.
3	5	The count of rows that can be processed simultaneously for Validate ECO.

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:

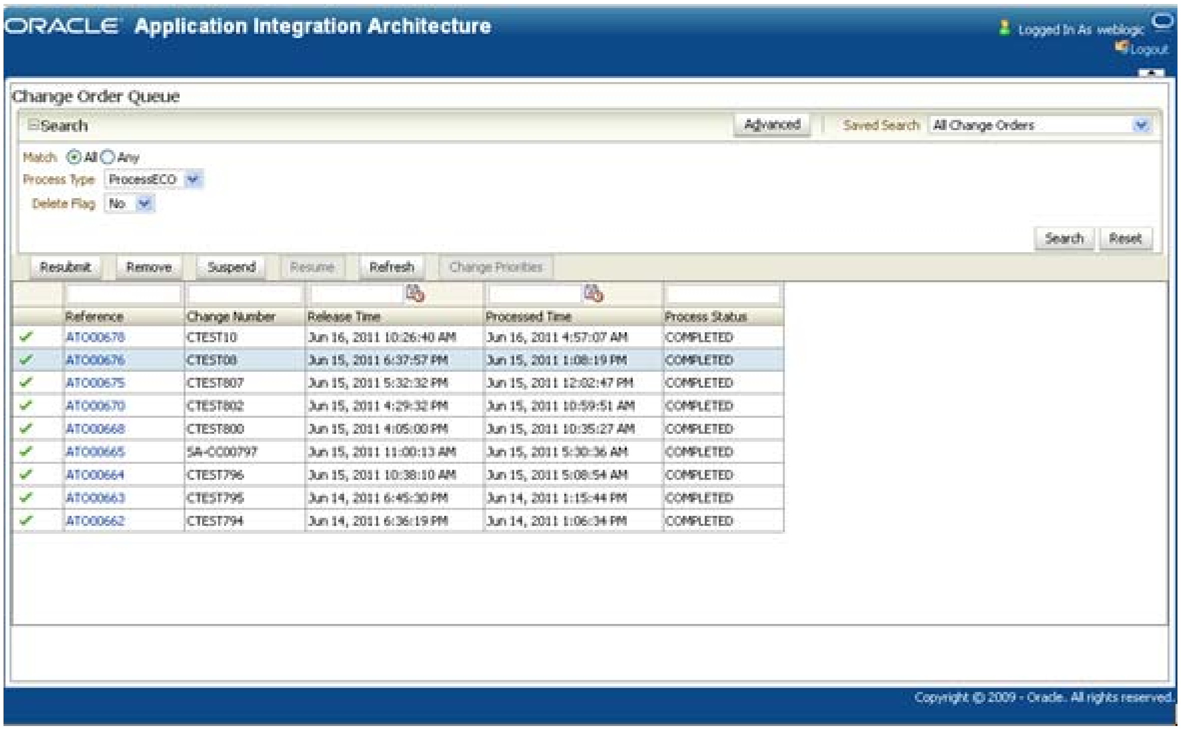
- All Event transmitted File or JMS messages are added to the Queue.

- 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 waits until the message is resubmitted or removed.

Queue Monitor

When a Change Order is released by ACS, it is picked up by the Queue Controller. The Queue Monitor displays a list of all the Change Orders that are waiting to be processed and facilitates you to reorder their sequence of processing.

For the first time after deployment, the Queue is in *Suspended* state. You are required to resume it.



Saved Search: All Change Orders

Match: ● All ○ Any

Process Type: ProcessECO

Delete Flag: No

	Reference	Change Number	Release Time	Processed Time	Process Status
✓	ATO00678	CTEST10	Jun 16, 2011 10:26:40 AM	Jun 16, 2011 4:57:07 AM	COMPLETED
✓	ATO00676	CTEST08	Jun 15, 2011 6:37:57 PM	Jun 15, 2011 1:08:19 PM	COMPLETED
✓	ATO00675	CTEST807	Jun 15, 2011 5:32:32 PM	Jun 15, 2011 12:02:47 PM	COMPLETED
✓	ATO00670	CTEST802	Jun 15, 2011 4:29:32 PM	Jun 15, 2011 10:59:51 AM	COMPLETED
✓	ATO00668	CTEST800	Jun 15, 2011 4:05:00 PM	Jun 15, 2011 10:35:27 AM	COMPLETED
✓	ATO00665	SA-CO00797	Jun 15, 2011 11:00:13 AM	Jun 15, 2011 5:30:36 AM	COMPLETED
✓	ATO00664	CTEST796	Jun 15, 2011 10:38:10 AM	Jun 15, 2011 5:08:54 AM	COMPLETED
✓	ATO00663	CTEST795	Jun 14, 2011 6:45:30 PM	Jun 14, 2011 1:15:44 PM	COMPLETED
✓	ATO00662	CTEST794	Jun 14, 2011 6:36:19 PM	Jun 14, 2011 1:06:34 PM	COMPLETED

Queue Monitor – Change Order Queue

For more information about the Queue Monitor, see the *Agile PLM Integration Pack for SAP User Guide* at http://www.oracle.com/technology/documentation/.

Queue Manager Services

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

- CreateQueueService

- CreateQueueControlService
- QueueProcessorService
- QueueProcessorServiceImpl

CreateQueueService

CreateQueueService is implemented as a Mediator Routing Service. An Adapter Service (File or JMS Adapter) polls on the destinations for any Event payloads. The payload is in the form of aXML files. This service receives a 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.

This 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 Mediator header transformation extension functions.

PROCESS_STATUS is *Pending* status for the newly inserted row.

PROCESS_PRIORITY is captured from the file name. (ACS can be configured to append a default order for the file name)

CreateQueueControlService

CreateQueueControlService is implemented as a Mediator Routing Service. A DB Adapter polls on the QUEUE_CONTROL_TABLE table. If there are no rows that 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.

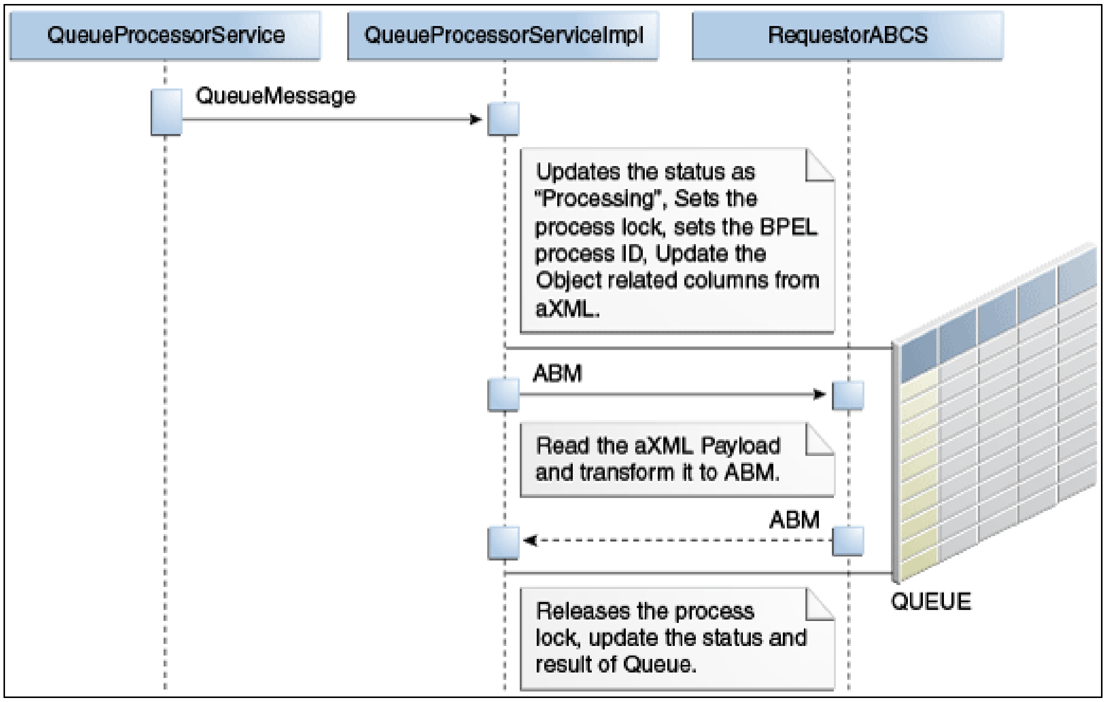
This polling strategy ensures that at any point of time there is only one pending message in the QUEUE_CONTROL_TABLE table. Once the pending message is processed and status completed, a new pending message is inserted from QUEUE_TABLE table to the QUEUE_CONTROL_TABLE table. When the status for a message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

QueueProcessorService

QueueProcessorService is implemented as a Mediator service, which acts similar to 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 following figure illustrates how QueueProcessorServiceImpl invokes the RequestorABCS:


QueueProcessorService
QueueProcessorServiceImpl
RequestorABCS
QueueMessage
Updates the status as "Processing", Sets the process lock, sets the BPEL process ID, Update the Object related columns from aXML.
ABM
Read the aXML Payload and transform it to ABM.
ABM
QUEUE
Releases the process lock, update the status and result of Queue.

QueueProcessorServiceImpl

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

The following table lists the steps required to invoke the RequestorABCS:

#	Name	Step Description
1.	QueueProcessorService Invokes QueueProcessorServiceImpl process	The QueueProcessorService invokes QueueProcessorServiceImpl with QueueMessage (generated by Toplink solution for QUEUE table) as input.

2	Invoke UpdateQueueStatus DB Adapter service	The input QueueMessage 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 Agile Data(AXML) to ABM	The QueueMessage has the Agile Data payload, which is transformed to an Application Business Message (ABM)
4.	Invoke RequestorABCS	QueueProcessorServiceImpl invokes the RequestorABCS with ABM as input.
5.	Invoke Coarse Grained Web Service	RequestorABCS optionally invokes the coarse grained

#	Name	Step Description
		web services to get the ABM populated with any missing information required for the Integration flow.
6.	RequestorABCS Transforms ABM to EBM	The response ABM from coarse grained WS is transformed to Enterprise Business Message (EBM) and an operation on the Enterprise Business Service (EBS) is invoked with EBM as the input.
7.	RequestorABCS orchestrates the business flow	The RequestorABCS routes the EBM to the EBS.
8.	EBS routes the response to RequestorABCS	The response EBM from EBS is routed to the RequestorABCS, which is transformed to ABM and returned to the QueueProcessorServiceImpl
9.	QueueProcessorServiceImpl invokes UpdateQueueResult DB Adapter service	The result from the RequestorABCS 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 is simpler to do in the JDeveloper Extensible Stylesheet Language (XSL) mapper.

Implementation Details

The QueueProcessorServiceImpl is implemented as an asynchronous Business Process Execution Language (BPEL) composite. 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 Mediator.

Note: 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 the Enterprise Business Flow (EBF). Any such errors leading to failure of the Queue processing is handled in this process. As a result of such an error, the Queue Status and Result with failure status is updated in the Queue DB.

Queue User Interface Application

The Queue User Interface (UI) application provides the following features:

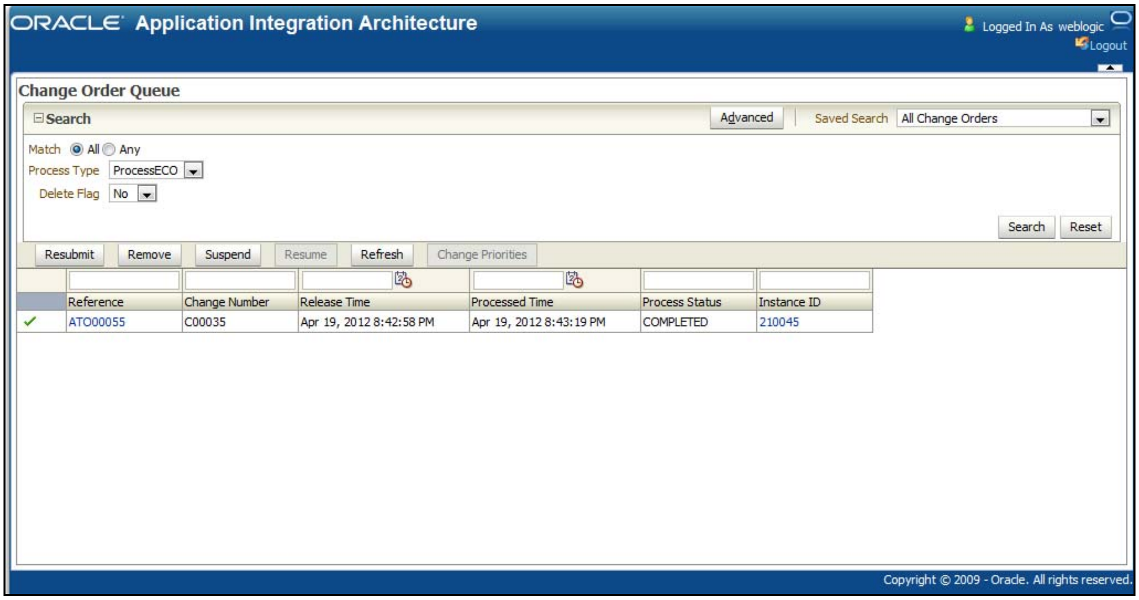
- Easy access Enterprise Manager (EM) console link to Service-Oriented Architecture (SOA) composite instances for Change Orders
- User role-based access
- Add new users into user roles

Easy Access EM Console Link to SOA Composite Instances for Change Orders

This feature allows you to easily access the flow trace of the SOA composite instances in EM console for a given change order. The column **Instance ID** in the Queue UI application shows the instance Id value of the first SOA composite in the flow with a hyperlink to view the full flow trace.

The hyperlinked **Instance ID** takes you to the Login screen of the EM console. After successful authentication, you can access the flow trace of execution for a given change order.

The following screenshots show how to access the EM console for successful and errored instances of SOA composites through the hyperlinked **Instance ID** field in the Queue UI application.



Saved Search: All Change Orders

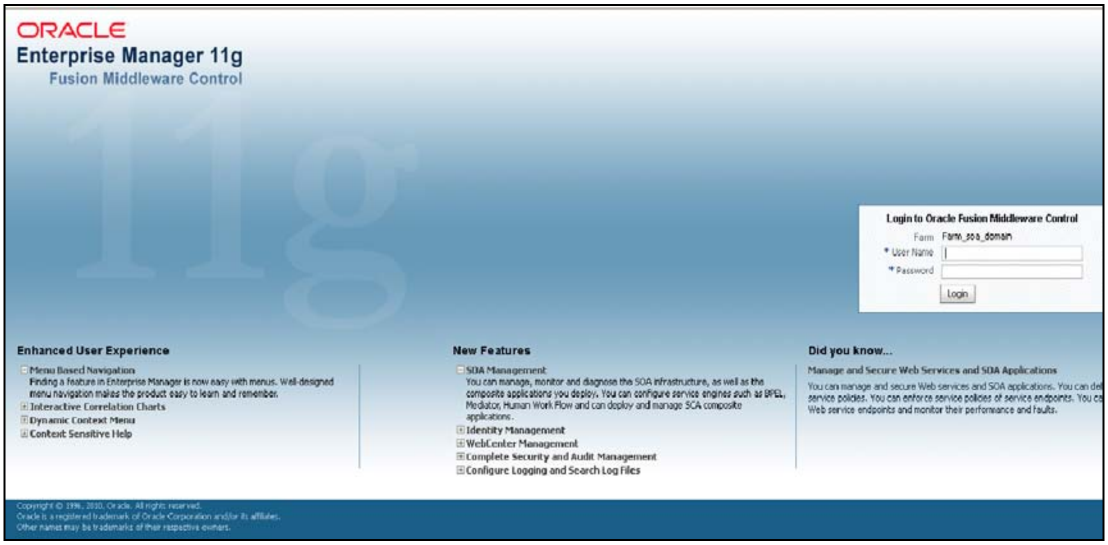
Match ● All ○ Any

Process Type: ProcessECO

Delete Flag: No

	Reference	Change Number	Release Time	Processed Time	Process Status	Instance ID
✓	ATO00055	C00035	Apr 19, 2012 8:42:58 PM	Apr 19, 2012 8:43:19 PM	COMPLETED	210045

Change Order Queue



ORACLE

Enterprise Manager 11g

Fusion Middleware Control

Login to Oracle Fusion Middleware Control

Farm Farm_soa_domain

* User Name

* Password

Login

Enhanced User Experience

- Menu Based Navigation
  Finding a feature in Enterprise Manager is now easy with menus. Well-designed menu navigation makes the product easy to learn and remember.
- Interactive Correlation Charts
- Dynamic Context Menu
- Context Sensitive Help

New Features

- SOA Management
  You can manage, monitor and diagnose the SOA infrastructure, as well as the composite applications you deploy. You can configure service engines such as BPEL, Mediator, Human Work Flow and can deploy and manage SCA composite applications.
- Identity Management
- WebCenter Management
- Complete Security and Audit Management
- Configure Logging and Search Log Files

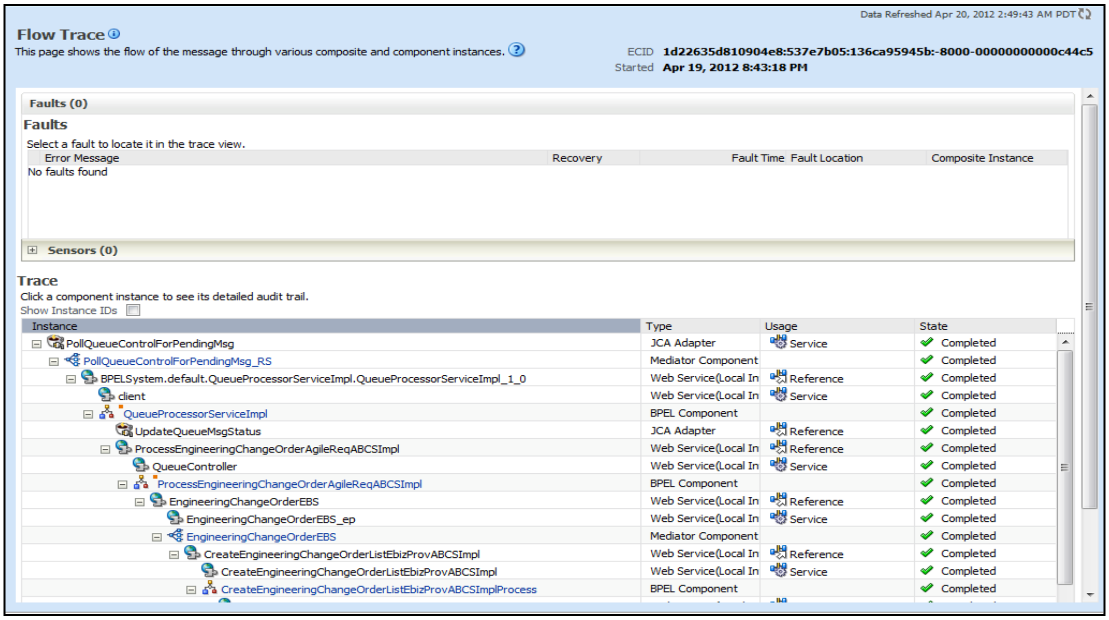
Did you know...

Manage and Secure Web Services and SOA Applications

You can manage and secure Web services and SOA applications. You can del service policies. You can enforce service policies of service endpoints. You ca Web service endpoints and monitor their performance and faults.

Copyright © 1996, 2010, Oracle. All rights reserved.
Oracle is a registered trademark of Oracle Corporation and/or its affiliates.
Other names may be trademarks of their respective owners.

Enterprise Manager - Login



Data Refreshed Apr 20, 2012 2:49:43 AM PDT

Flow Trace

This page shows the flow of the message through various composite and component instances.

ECID 1d22635d810904e8:537e7b05:136ca95945b:-8000-0000000000c44c5

Started Apr 19, 2012 8:43:18 PM

Faults (0)

Faults

Select a fault to locate it in the trace view.

Error Message	Recovery	Fault Time	Fault Location	Composite Instance
No faults found				

⊞ Sensors (0)

Trace

Click a component instance to see its detailed audit trail.

Show Instance IDs □

Instance	Type	Usage	State
⊟ PollQueueControlForPendingMsg	JCA Adapter	Service	✓ Completed
⊟ PollQueueControlForPendingMsg_RS	Mediator Component		✓ Completed
⊟ BPELSystem.default.QueueProcessorServiceImpl.QueueProcessorServiceImpl_1_0	Web Service(Local In	Reference	✓ Completed
client	Web Service(Local In	Service	✓ Completed
⊟ QueueProcessorServiceImpl	BPEL Component		✓ Completed
UpdateQueueMsgStatus	JCA Adapter	Reference	✓ Completed
⊟ ProcessEngineeringChangeOrderAgileReqABCSImpl	Web Service(Local In	Reference	✓ Completed
QueueController	Web Service(Local In	Service	✓ Completed
⊟ ProcessEngineeringChangeOrderAgileReqABCSImpl	BPEL Component		✓ Completed
⊟ EngineeringChangeOrderEBS	Web Service(Local In	Reference	✓ Completed
EngineeringChangeOrderEBS_ep	Web Service(Local In	Service	✓ Completed
⊟ EngineeringChangeOrderEBS	Mediator Component		✓ Completed
⊟ CreateEngineeringChangeOrderListEbizProvABCSImpl	Web Service(Local In	Reference	✓ Completed
CreateEngineeringChangeOrderListEbizProvABCSImpl	Web Service(Local In	Service	✓ Completed
⊟ CreateEngineeringChangeOrderListEbizProvABCSImplProcess	BPEL Component		✓ Completed

Enterprise Manager – Flow Trace

User Role-Based Access in Queue UI Application

The Queue UI application ships with role-based controlled access to the application. Following are the two roles:

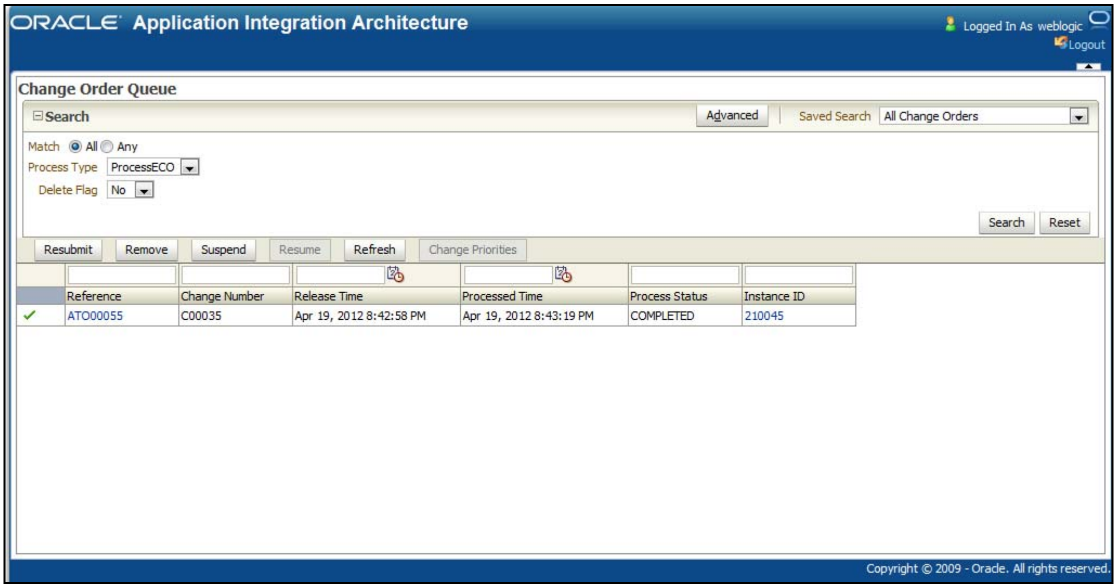
Role	Description
For Change UI administrators: Role Name: ChangeOrderUIAdminRole	Have full access to the Queue UI application. Have exclusive access to: • Suspend or resume the queue. • Resubmit an errored message. • Reorder the priorities of messages. • Remove a message from the queue. • Access the EM console for successful and errored instances of SOA composites through hyperlinked Instance ID field in the Queue UI application.
For business users: Role Name: ChangeOrderUIBusinessRole	Have view only access to the Queue UI application.

As part of the Queue UI installation, the Queue UI application ships with the following two default users:

- ChangeOrderUIAdmin: This role belong to ChangeOrderUIAdminRole
- ChangeOrderUIBusiness: This role belongs to ChangeOrderUIBusinessRole

The default password for these users is same as that of the application server password provided at the time of the pre-built integration installation. The WebLogic administrator is (by default) granted ChangeOrderUIAdmin role.

The following are the screenshots for different user role scenarios:



Change Order Queue

Search

Saved Search: All Change Orders

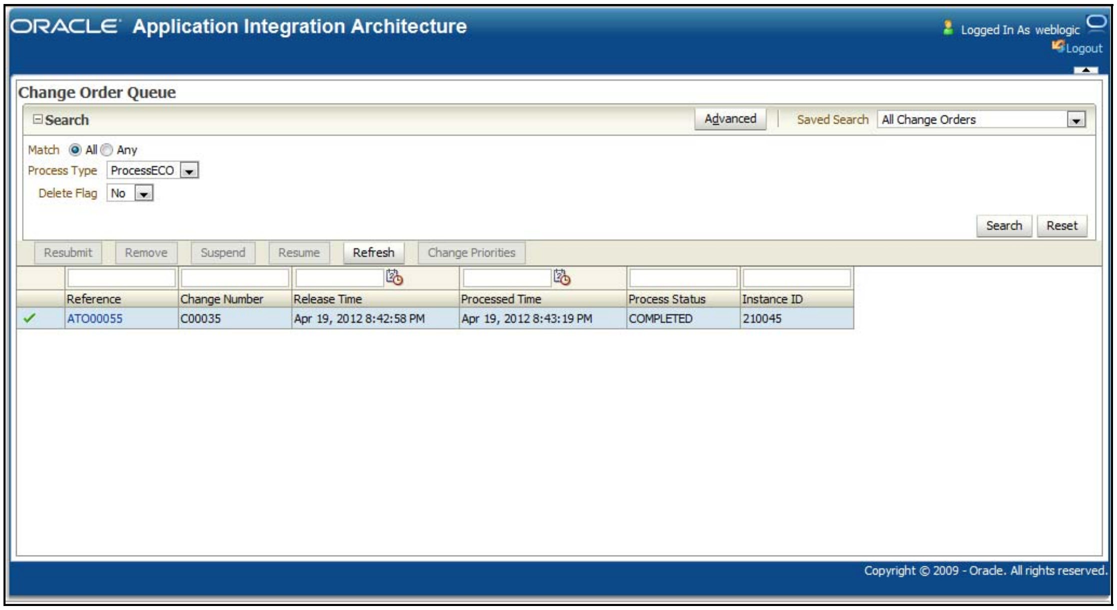
Match ● All ○ Any

Process Type: ProcessECO

Delete Flag: No

	Reference	Change Number	Release Time	Processed Time	Process Status	Instance ID
✓	ATO00055	C00035	Apr 19, 2012 8:42:58 PM	Apr 19, 2012 8:43:19 PM	COMPLETED	210045

Administrator role view of the Queue UI application



Change Order Queue

Search

Saved Search: All Change Orders

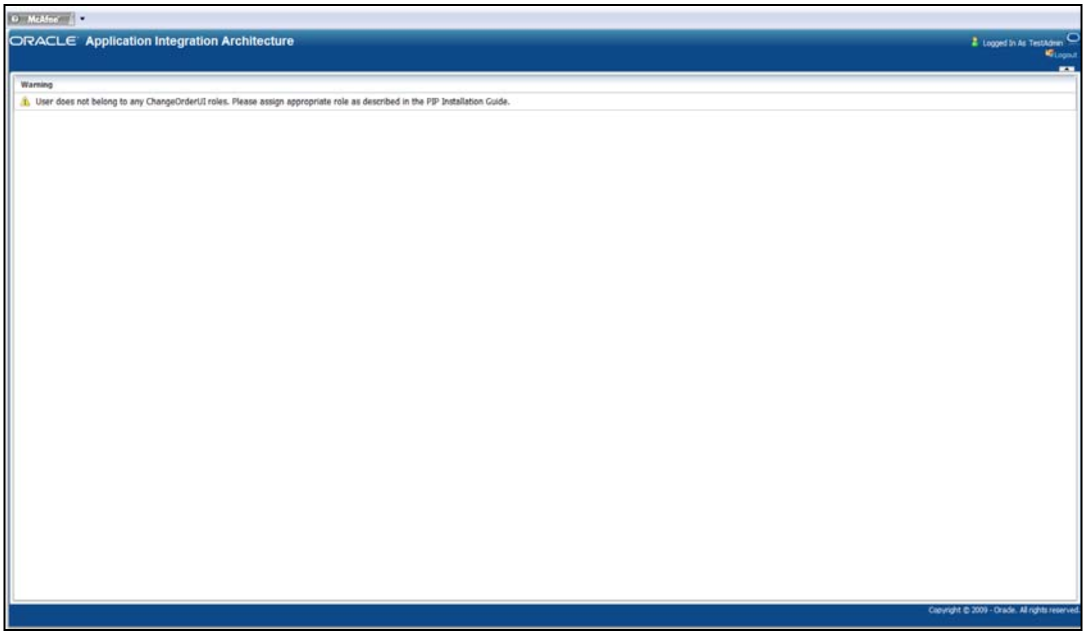
Match ● All ○ Any

Process Type: ProcessECO

Delete Flag: No

	Reference	Change Number	Release Time	Processed Time	Process Status	Instance ID
✓	ATO00055	C00035	Apr 19, 2012 8:42:58 PM	Apr 19, 2012 8:43:19 PM	COMPLETED	210045

Business user role view of the Queue UI application


McAfee
ORACLE Application Integration Architecture
Logged In As TestAdmin
Logout
Copyright © 2009 - Oracle. All rights reserved.

Warning

⚠ User does not belong to any ChangeOrderUI roles. Please assign appropriate role as described in the PIP Installation Guide.

Queue UI application when a user is not assigned any of the Change Order UI roles

Add New Users into User Roles in Queue UI Application

To add new users to user roles in the Queue UI application:

1. Log on to the Weblogic Server by using the appropriate port, http://<hostname>:<portnumber>/console.
2. Click **Security Realms** on the left pane of the Weblogic Administration Console.
3. Click **myrealm** (or the realm with true in the Default Realm column) in the Realms table.
4. Click **Users and Groups** tab.
5. Add a new user and provide all the details.
6. Click the newly added user and go to the **Groups** tab.
7. Select *ChangeOrderUIAdminRole* or *ChangeOrderUIBusinessRole* from the list of available groups and move them to the list of chosen groups for the user.
8. Click **Save** to save the changes.

Appendix E: Troubleshooting

This appendix describes common troubleshooting issues and solutions pertaining to Engineering Change Orders (ECOs), installation, Queue Management, Agile, and SAP.

This appendix includes the following sections:

- Engineering Changer Order Issues
- Installation Issues
- Queue Management Issues
- SAP Issues
- Agile Issues

Engineering Change Order Issues

These are the common ECO issues and solutions:

Issue: In ECO forward flow, after the ECO is processed successfully, the transfer status attribute (flex) in the ECO in Agile is not getting updated.

Solution: Check which flexfield attribute has been enabled corresponding to the change. Ensure that the same attribute has been configured in the AIAConfigurationProperties.xml for that property.

Issue: For the Item Cost Update update flows, the attributes in Agile are not getting updated.

Solution: Check to see if the Multisite Enabled property is set to *True* or *False*. Based on this given value, verify that the Cost attributes in AIAConfigurationProperties.xml are correctly set.

For more information about item update, see, Chapter 4: Process Integration for Item Attribute Update.

Installation Issues

These are the common installation issues and solutions:

Issue: While installing the Oracle Application Integration Architecture Foundation Pack (Oracle AIA FP) in the Linux box, the deployment of configurations and deploying the pre-built integration did not succeed due to the Service-Oriented Architecture (SOA) server being unable to restart itself automatically using the deployment script given in the Oracle AIA FP.

Solution: To resolve this installation issue:

1. Stop SOA server.
2. Restart the SOA server from this location:

   $Middleware_Home\user_projects\domains\soa_domain\bin>.\startManagedWeblogic.sh soa_server1

3. Click **Retry** in the configuration wizard.

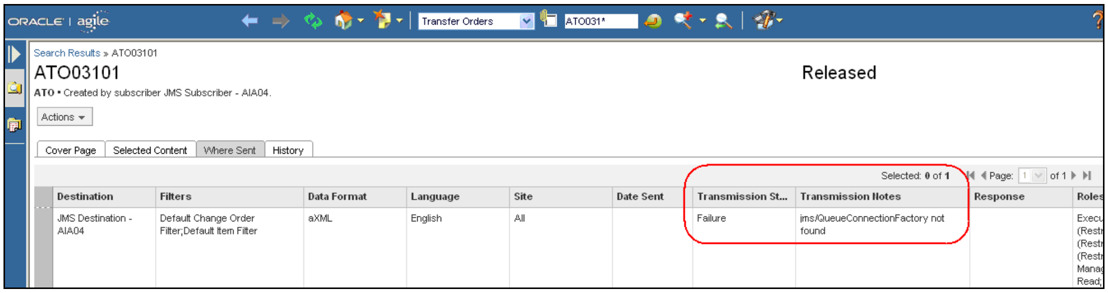
Queue Management Issues

These are the common queue issue and solutions:

Issue: Once an ECO, Manufacturing Change Order (MCO), or Site Change Order (SCO) has been released in Agile, the Queue does not display any corresponding entry for the change order.

Solution: To resolve this queue issue:

1. Check Agile for assemble-to-order (ATO), which was created on the release of the particular change order. Check the status on the **Where Sent** tab of the ATO.
2. If it shows a *Failure* message, this implies that an error occurred while Agile Content Service (ACS) was processing the publishing of the data to the Java Message Service (JMS) destination. The error message is specified in the **Transmission Notes** column as shown in this screenshot.



Search Results » ATO03101

ATO03101

Released

ATO • Created by subscriber JMS Subscriber - AIA04.

Selected: 0 of 1 Page: 1 of 1

Destination	Filters	Data Format	Language	Site	Date Sent	Transmission St...	Transmission Notes	Response	Roles
JMS Destination - AIA04	Default Change Order Filter;Default Item Filter	aXML	English	All		Failure	jms/QueueConnectionFactory not found		Execu (Restr (Restr (Restr Manag Read;

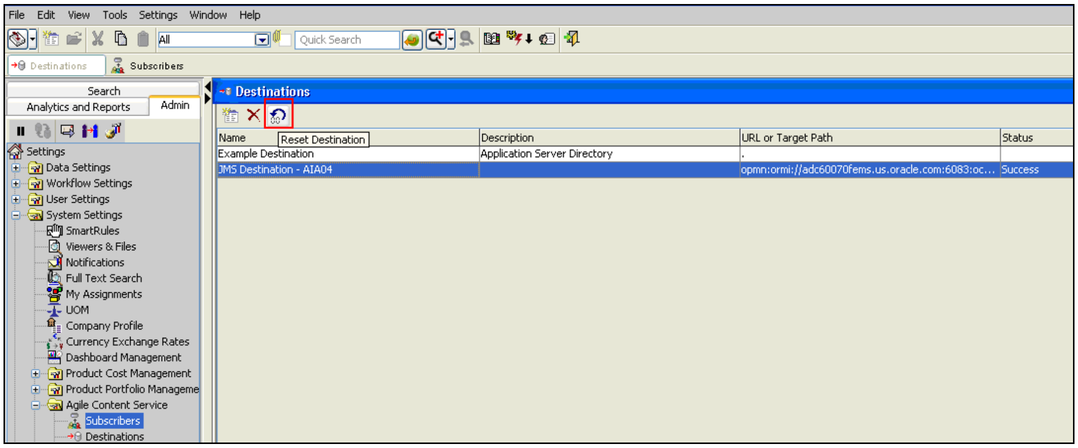
Agile ATO

3. After you make any necessary changes to the transfer order or the destination to correct the problem, reset the destination to attempt delivery again.

   Once a destination has failed, no other transfer orders can be sent to that destination until it has been reset.

4. To reset the destination, in Agile Java client, go to **Admin**, **System Settings**, **Agile Content Service**, **Destinations**.
5. Select the required destination and click **Reset**.

   The **Reset** icon is next to the **Delete** icon.



Name	Description	URL or Target Path	Status
Example Destination	Application Server Directory	.	
JMS Destination - AIA04		opmn:ormi://adc60070fems.us.oracle.com:6083:oc...	Success

ACS - Destinations

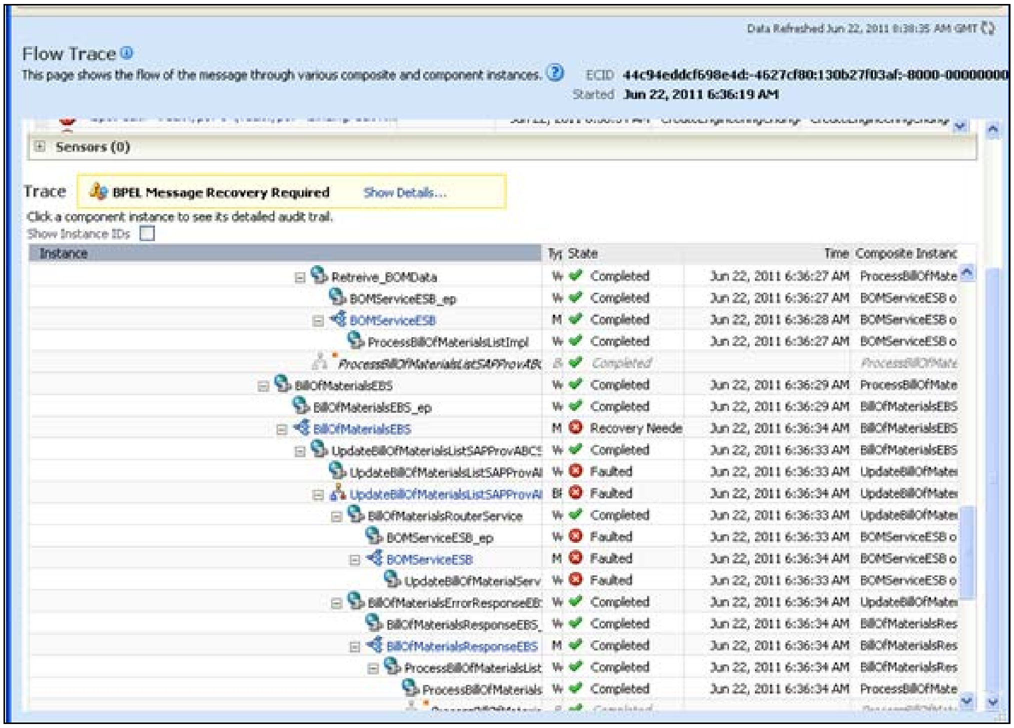
6. After resetting the destination, test the destination to ensure the test is successful for the destination.

   In case it fails, this has to be resolved, mainly by ensuring all the ECO Queue settings are correct and SOA Port specified in the URL is correct.

   If the status of the ATO transfer is **Success**, it implies that the ACS is publishing the data to JMS queue successfully. Therefore you must troubleshoot in the BPEL console.

7. Navigate to the Weblogic EM Console: `http://<host name>:<port number>/em`

8. Click the **Instances** tab. Check for the instance of the *UpdateBillOfMaterialsListSAPProvABCSImpl* for which the error occurred.



Data Refreshed Jun 22, 2011 8:38:35 AM GMT

Flow Trace

This page shows the flow of the message through various composite and component instances.

ECID 44c94eddcf698e4d:-4627cf80:130b27f03af:-8000-00000000

Started Jun 22, 2011 6:36:19 AM

Sensors (0)

Trace BPEL Message Recovery Required Show Details...

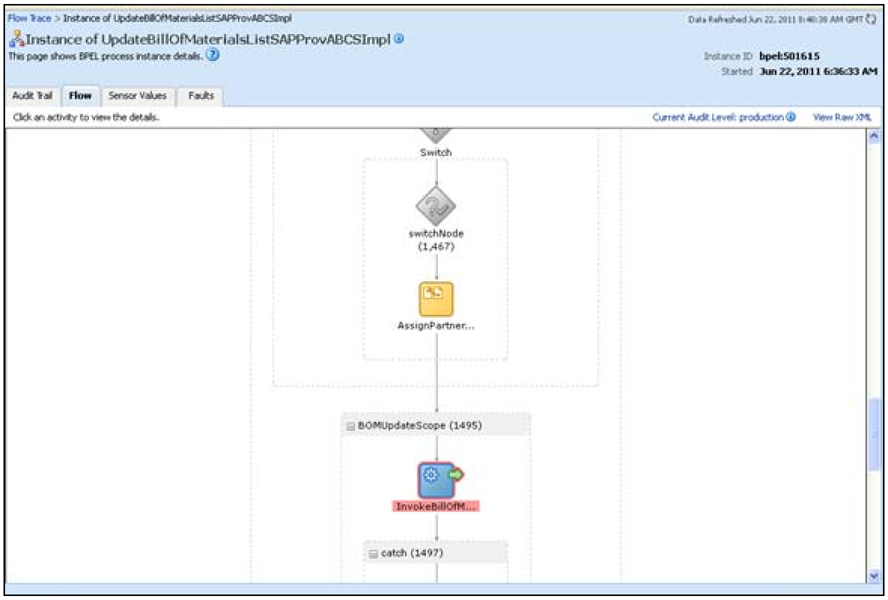
Click a component instance to see its detailed audit trail.

Show Instance IDs □

Instance	Ty	State	Time	Composite Instanc
Retreive_BOMData	W	✓ Completed	Jun 22, 2011 6:36:27 AM	ProcessBillOfMate
BOMServiceESB_ep	W	✓ Completed	Jun 22, 2011 6:36:27 AM	BOMServiceESB o
BOMServiceESB	M	✓ Completed	Jun 22, 2011 6:36:28 AM	BOMServiceESB o
ProcessBillOfMaterialsListImpl	W	✓ Completed	Jun 22, 2011 6:36:27 AM	BOMServiceESB o
ProcessBillOfMaterialsListSAPProvABC	B	✓ Completed		ProcessBillOfMate
BillOfMaterialsEBS	W	✓ Completed	Jun 22, 2011 6:36:29 AM	ProcessBillOfMate
BillOfMaterialsEBS_ep	W	✓ Completed	Jun 22, 2011 6:36:29 AM	BillOfMaterialsEBS
BillOfMaterialsEBS	M	✗ Recovery Neede	Jun 22, 2011 6:36:34 AM	BillOfMaterialsEBS
UpdateBillOfMaterialsListSAPProvABCS	W	✓ Completed	Jun 22, 2011 6:36:33 AM	BillOfMaterialsEBS
UpdateBillOfMaterialsListSAPProvAI	W	✗ Faulted	Jun 22, 2011 6:36:33 AM	UpdateBillOfMater
UpdateBillOfMaterialsListSAPProvAI	BI	✗ Faulted	Jun 22, 2011 6:36:34 AM	UpdateBillOfMater
BillOfMaterialsRouterService	W	✓ Completed	Jun 22, 2011 6:36:33 AM	UpdateBillOfMater
BOMServiceESB_ep	W	✗ Faulted	Jun 22, 2011 6:36:33 AM	BOMServiceESB o
BOMServiceESB	M	✗ Faulted	Jun 22, 2011 6:36:34 AM	BOMServiceESB o
UpdateBillOfMaterialServ	W	✗ Faulted	Jun 22, 2011 6:36:33 AM	BOMServiceESB o
BillOfMaterialsErrorResponseEB	W	✓ Completed	Jun 22, 2011 6:36:34 AM	UpdateBillOfMater
BillOfMaterialsResponseEBS_	W	✓ Completed	Jun 22, 2011 6:36:34 AM	BillOfMaterialsRes
BillOfMaterialsResponseEBS	M	✓ Completed	Jun 22, 2011 6:36:34 AM	BillOfMaterialsRes
ProcessBillOfMaterialsList	W	✓ Completed	Jun 22, 2011 6:36:34 AM	BillOfMaterialsRes
ProcessBillOfMaterials	W	✓ Completed	Jun 22, 2011 6:36:34 AM	ProcessBillOfMate

BPEL Console - Instances

9. Click on the **Instance Name** and go to the *Flow* link. In the BPEL flow shown, find the element at which the error has occurred and then click on the element to view the Activity Audit trail for more details.



Flow Trace > Instance of UpdateBillOfMaterialsListSAPProvABCSImpl

Data Refreshed Jun 22, 2011 8:40:38 AM GMT

Instance of UpdateBillOfMaterialsListSAPProvABCSImpl

This page shows BPEL process instance details.

Instance ID bpel:501615

Started Jun 22, 2011 6:36:33 AM

Click an activity to view the details.

BPEL Flow

SAP Issues

When implementing the Agile PLM Integration Pack for SAP: design to Release pre-built integration, you might encounter the following issues.

Issue: When updating SAP Item attributes to Agile, if the Agile application is down or an error occurs in processing the data, then the data cannot be resubmitted to Agile.

Solution: If there is any error in the SAP item attributes update to Agile, an email is sent to the customer or administrator specifying the failure.

To resolve the issue, identify the failed IDOC and resubmit from SAP:

1. Open the error mail that was sent to the customer, user, or administrator.
2. Click on the hyperlink provided on the error mail.
3. Open the BPEL Console and navigate to the instance that was given in the link.
4. Scroll down to the **Assign Fault** and check the **IDO**C number that has failed.
5. Navigate to the SAP application and resubmit the IDOC with the item attribute details provided in the email.

Note: Verify that the Agile application server is up and running before initiating the process.

Issue: If there is an error in **CreateBillOfMaterialsListSAPProvABCSImpl** with the error description as *"javax.xml.rpc.soap.SOAPFaultException: com.sap.mw.jco.JCO$AbapException: (126) ERROR: Terminate processing"*.

Solution: Perform these steps:

1. Verify the components in the Bill Of Materials (BoM) were created in SAP. If the components are not created, add the components to the affected item and then release the Engineering Change Order (ECO).
2. Verify the components have a valid *Item Category, BoM Usage* and *Effectivity Date*.

Issue: If there is an error in **UpdateBillOfMaterialsListSAPProvABCSImpl** with the error description as *"javax.xml.rpc.soap.SOAPFaultException: com.sap.mw.jco.JCO$AbapException: (126) ERROR: Terminate processing"*

Solution: Perform these steps:

1. Verify the components in the BoM were created in SAP. If the components are not created, add the components to the affected item and then release the ECO.
2. Verify the components have a valid *Item Category, BoM Usage* and *Effectivity Date*.
3. Verify that the components defined while creating the BoM have a unique **Find Number** in Agile PLM.

Issue: If there is an error in **SyncItemListSAPProvABCS** with the error description as "*Exception during SOAP invoke: com.sap.mw.jco.JCO$AbapException: (126) REVISION_LEVEL_EXISTS: REVISION_LEVEL_EXISTS*" even though the process is showing as success.

Solution: This error can be ignored as it is only a check for revision existence for the particular item or material.

Issue: If there is an error in **SyncItemListSAPProvABCS** or ReserveItemSAPProvABCS with the error description as "*Exception during SOAP invoke: java.lang.Exception: BapiError: Enter a material type; nested exception is: javax.xml.rpc.soap.SOAPFaultException: java.lang.Exception: BapiError: Enter a material type*"

Solution: Verify the ITEM_PRIMARYCLASSIFICATIONCODE Domain Value Map (DVM) is mapped correctly. If not modify the same and resubmit from the Queue Manager console.

Also verify the REFERENCEMATERIALS is mapped with the corresponding material in SAP.

Issue: If there is an error in **SyncItemListSAPProvABCS** with the error description as *"Exception during SOAP invoke: com.ibi.sap.SapAdapterException: Field: REV_LEVEL_NEW exceeds defined in structure INPUT allowed length 2: ABC; nested exception is: javax.xml.rpc.soap.SOAPFaultException: com.ibi.sap.SapAdapterException: Field: REV_LEVEL_NEW exceeds defined in structure INPUT allowed length 2: ABC"*

Solution: Verify the revision level defined for the affected item is 2 characters.

Issue: **MediatorMessageProcessingFailed** due to timed out exception

Solution: Remove the ECO from the queue with processing state and re-release the ECO from Agile.

Agile Issues

Issue: Issue in fetching the Part number from SAP using New Part Request (NPR).

Solution: Encrypt the SOA_LOGIN_PASSWORD by using the following command in Agile:

<AgileHome>/agileDomain/bin/encryptpwd.sh and place the encrypted password in the config.properties in NPRpx.jar of Agile.