Oracle Product Data Synchronization Integration Pack for Oracle Utilities Customer Care and Billing and Siebel Energy 11.1 -

Implementation Guide

Release 11.1

E26762-02

February 2012

ORACLE

Oracle Product Data Synchronization Integration Pack for Oracle Utilities Customer Care and Billing and Siebel Energy 11.1 - Implementation Guide, Release 11.1

E26762-02

Copyright © 2001, 2012, Oracle and/or its affiliates. All rights reserved.

Primary Author: Oracle Corporation

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

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

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

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

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

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

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

Contents

Preface 5
Oracle Application Integration Architecture Guides 5
Additional Resources 5
Part 1: Understanding the Delivered Integrations 7
Chapter 1: Product Data Synchronization Overview 8
Overview 8
Participating Applications 8
References 9
Chapter 2: Product Data Synchronization 10
Product Data Synchronization Overview 10
Conceptual Design 12
Supported Functionality 21
Assumptions and Constraints 21
Chapter 3: Understanding Integration Interfaces and Components 22
Product Data Integration Flow 23
Oracle Utilities Customer Care and Billing Interfaces 25
Siebel Energy Interfaces 26
Industry Oracle Application Integration Architecture (AIA) Components 26
Integration Services 27
Part 2: Implementing the Delivered Integrations 29
Chapter 4: Configuration Guidelines 30
Setting up Oracle Utilities Customer Care and Billing 30
Setting up Siebel Energy 44
Setting up the Product Data Synchronization Integration Pack 46
Chapter 5: Monitoring, Error Handling, and Troubleshooting 51
Monitoring from Oracle Utilities Customer Care and Billing 52
Monitoring from Siebel Energy 53
Monitoring from the Integration 53
Troubleshooting 54
Chapter 6: Extensibility Options 58
Extending EBOs 59

Copyright © 2012, Oracle. All rights reserved.

Extending ABC Services 62
Passing Data without Extending the EBO 63
Chapter 7: Security 63
Applied Security Policies 63
Appendix A: Data Mapping 65
Appendix B: Cross-References 70
Appendix C: Using Session Pool Manager 72
Configuring Session Pool Manager 73
Appendix D: CAVS Testing 74

Copyright © 2012, Oracle. All rights reserved.

Preface

Welcome to the Oracle Product Data Synchronization Integration Pack for Oracle Utilities Customer Care and Billing and Siebel Energy.

Oracle Application Integration Architecture (AIA) provides the guides and resources indicated in this section.

Oracle Application Integration Architecture Guides

- Oracle Fusion Middleware Infrastructure Components and Utilities User's Guide for Oracle Application Integration Architecture Foundation Pack 11g Release 1 (11.1.1.4.0)
- Oracle Fusion Middleware Installation and Upgrade Guide for Oracle Application Integration Architecture Foundation Pack 11g Release 1 (11.1.1.4.0)
- Oracle Fusion Middleware Concepts and Technologies Guide for Oracle Application Integration Architecture Foundation Pack 11g Release 1 (11.1.1.4.0)
- Oracle Fusion Middleware Reference Process Models User's Guide for Oracle Application Integration Architecture Foundation Pack 11g Release 1 (11.1.1.4.0)
- Oracle Fusion Middleware Migration Guide for Oracle Application Integration Architecture 11g Release 1 (11.1.1.4.0)
- Oracle Fusion Middleware Developer's Guide for Oracle Application Integration Architecture Foundation Pack 11g Release 1 (11.1.1.4.0)

Additional Resources

Additional documentation provided that is specific to this release.

- Oracle Application Integration Architecture: Product-to-Guide Index
- Known Issues and Workarounds
- Release Notes
- Documentation Updates Guide
- Functional Diagrams

Note: Please refer to the Oracle Technology Network (OTN) site for the most recent version of this documentation.

Copyright © 2012, Oracle. All rights reserved.

Part 1: Understanding the Delivered Integrations

Chapter 1: Overview

Chapter 2: Product Data Synchronization

Chapter 3: Understanding Integration Interfaces and Components

Copyright © 2012, Oracle. All rights reserved.

Chapter 1: Product Data Synchronization Overview

This chapter provides information regarding:

- Overview
- Participating Applications
- References

Overview

The Product Data Synchronization Integration Pack for Oracle Utilities Customer Care and Billing and Siebel Energy enables the synchronization of Product information between Oracle Utilities Customer Care and Billing (CC&B) and Siebel Energy to support the Order-to-Bill process.

- Oracle Utilities Customer Care and Billing is the product and pricing master. Siebel Energy needs this information to support selling activities.
- The basic product and pricing information synchronized from Oracle Utilities Customer Care and Billing is enriched in Siebel Energy to support product configuration models, product eligibility and compatibility rules, product recommendations (for example, up-sell and cross-sell), promotions (for example, product bundling and pricing discounts), and catalog.
- The integration flow handles both Create and Updates of Product information from Oracle Utilities Customer Care and Billing to Siebel.

Participating Applications

This chapter provides a general description of each of the edge applications that are participating in the integration process.

The Product Data Synchronization Integration Pack for Oracle Utilities Customer Care and Billing and Siebel Energy is an integration involving the following products:

- Oracle Utilities Customer Care and Billing (CC&B)
- Siebel Energy (Siebel)

Copyright © 2012, Oracle. All rights reserved.

Oracle Utilities Customer Care and Billing

Oracle Utilities Customer Care and Billing (CC&B) is a customer and billing system that manages all aspects of customer service needed by most utilities to operate their business. Basic objects form the core of the system: person, account, premise, service agreement, and service point. These objects hold demographic, geographic, and financial information about a company's customers and properties. Related to these objects are the processes that they manage: bills, payments, meter readings, field activities, and so forth.

Siebel Energy

Siebel Energy is designed to meet the sales, marketing, call center, and customer field service requirements of regulated electric, gas, and water Distribution Companies, as well as unregulated Energy Service Providers (ESPs). Siebel Energy serves residential, commercial, and industrial customers. Siebel Energy can support large or small call center operations and integrates with legacy and packaged Customer Information Systems (CIS) and Billing solutions. It supports account hierarchy and premises records with their service points. It enables efficient customer order and contract management. It also allows the tracking, sharing, and reporting of usage, billing, and other service details.

References

Following references are provided regarding terms and concepts described in this document.

For more information on the terms and entities related to the integration, see Appendix B: Cross-References.

For more information on AIA terms and concepts please refer to the *Oracle Fusion Middleware Concepts and Technologies Guide for Oracle Application Integration Architecture Foundation Pack 11g Release 1 guide*.

Copyright © 2012, Oracle. All rights reserved.

Chapter 2: Product Data Synchronization

This chapter provides an overview of Product Data Synchronization and discusses:

- Overview
- Conceptual Design
- Supported Functionality
- Assumptions and Constraints

Product Data Synchronization Overview

The Synchronize Product and Pricing integration flow focuses on the initial setup and ongoing maintenance of products in Oracle Utilities Customer Care and Billing (CC&B) and Siebel Energy to be used for ordering and billing. The approach outlined in this integration flow is to setup and maintain product information in Oracle Utilities Customer Care and Billing and pass it to Siebel Energy where it is enriched to support the sales process.

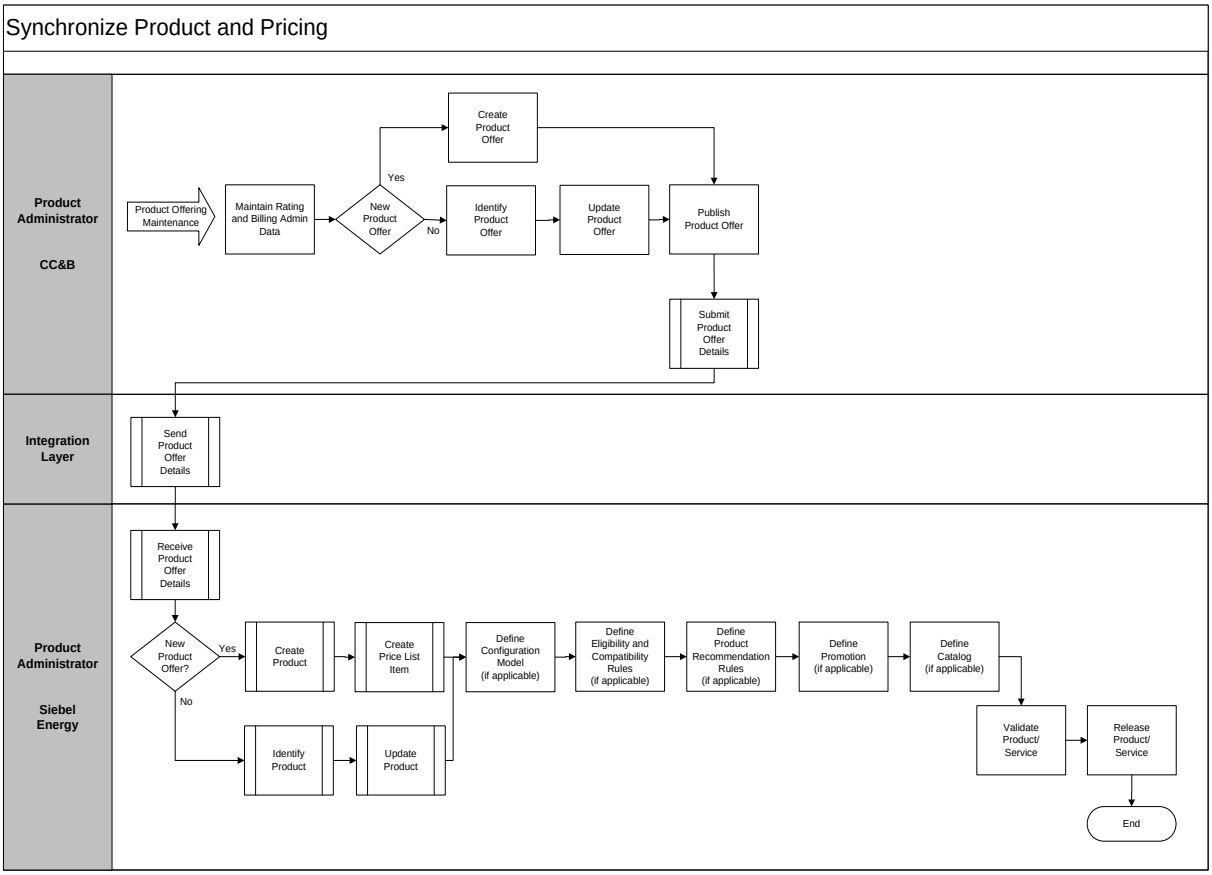
The Product Administrator examines the product marketing, ordering, rating, and billing requirements to determine how the Utility Service Product should be modeled in Oracle Utilities Customer Care and Billing and Siebel Energy. This includes:

- Whether the product is a part of a larger product offering, a product bundle, or a promotional offering
- What product attributes or characteristics may influence prices and product eligibility and compatibility
- What non-product attributes impact prices or trigger-specific charges (for example, contract duration with penalty, duration of promotional pricing, bundle pricing with unbundling penalties, eBilling option, automatic payments, and so forth.)
- How these product definitions should be setup in Oracle Utilities Customer Care and Billing for rating and invoicing

Synchronize (Create/Update) Product Business Flow

The following process flow diagram depicts the activities related to Synchronize Product and Pricing process integration flow:

Copyright © 2012, Oracle. All rights reserved.



Synchronize Product and Pricing Integration Flow Diagram

If the Product Administrator needs to create/update a product, the following activities are performed:

1. Product Administrator maintains the Rating and Billing admin data used to determine billing and pricing rules, as needed. This rating and billing admin data includes Service Agreement Type, Rate (Rate Schedule, Rate Version, Rate Component, Bill Factors), and Service Agreement Type Start Option (Start Option, Recurring Charge Amount, Contract Rider, Contract Value, Contract Quantity, Characteristics).
2. Product Administrator creates and updates a product in Oracle Utilities Customer Care and Billing and publishes it.
3. The Product information is transformed in the integration layer and is created in Siebel Energy.
4. Product Administrator enriches the product in Siebel Energy by setting up the configuration models, eligibility/compatibility rules, product recommendation rules, promotions, and catalog to support the quoting or ordering process.
5. Product Administrator validates and releases the product used for selling.

Copyright © 2012, Oracle. All rights reserved.

Conceptual Design

The following sections outline the conceptual design between Oracle Utilities Customer Care and Billing (CC&B) and Siebel Energy for various key entities related to product information:

- Products in Oracle Utilities Customer Care and Billing
- Products in Siebel Energy
- Product Data Synchronization Process

Products in Oracle Utilities Customer Care and Billing

Product Offer is new functionality available in Oracle Utilities Customer Care and Billing. It provides a simplified view of products for marketing and customer service, to respond to customer queries, while also abstracting the complexity of the billing and pricing rules. Additionally, it will simplify the integration of product information with external systems like Siebel Energy or Oracle Utilities Quotation Management.

Note: Product Catalog can be used as a standalone solution or integrated solution.

The Product Administrator creates an instance of a Product Offer in the Product Catalog for each product. The Product Offer includes the following information:

- Descriptions of pricing and eligibility information
- Types of service agreements to create for a product
- Product component information (or customer/contract specific info for a product) e.g. price for on peak and off peak kWh

Note: The rating and billing admin data used to define the service agreement in Oracle Utilities Customer Care and Billing still needs to be setup by the Product Administrator. The key rating and billing admin data includes Service Agreement Type, Rate (Rate Schedule, Rate Version, Rate Component, Bill Factors), Service Agreement Type Start Option (Start Option, Recurring Charge Amount, Contract Rider, Contract Value, Contract Quantity, Characteristics), and Column References.

The existing Service Task Type Maintenance Object (MO) is used to define product offer information in Oracle Utilities Customer Care and Billing.

New portals/zones have been developed for product offer setup and query.

The existing Framework provided sync request process is used to integrate products to Siebel Energy. Several new business objects have been added to support sync/publishing of products.

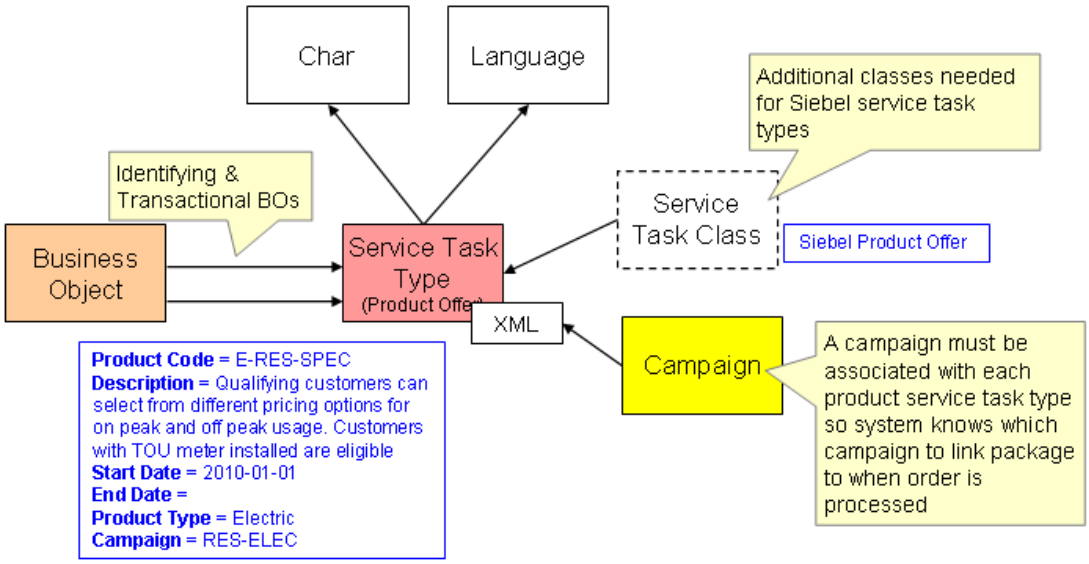
The relationship between product offers in Oracle Utilities Customer Care and Billing and products in Siebel Energy will be maintained in a cross-reference table in integration.

Copyright © 2012, Oracle. All rights reserved.

For more information about the sync request process, the business objects, maintenance objects, and other components used for this process, see the Oracle Utilities Framework User Guide section titled “Data Synchronization”.

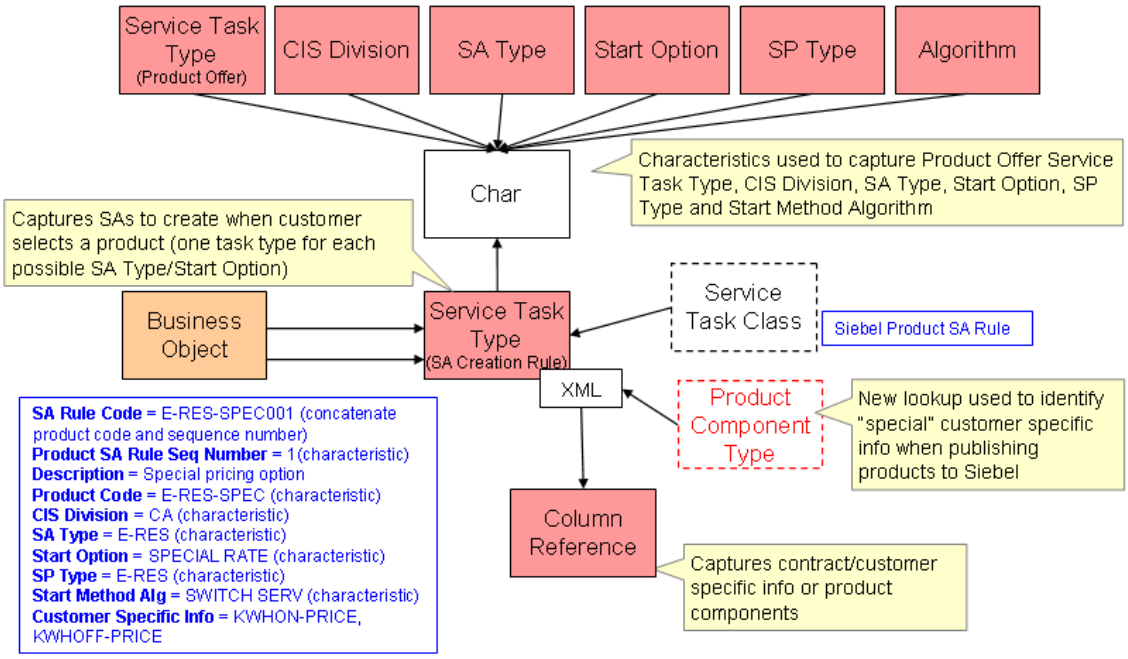
Entity Relationship Diagrams of a Product in Oracle Utilities Customer Care and Billing

Below are the Entity Relationship Diagrams showing the relationship between different components involved in the creation of Product Offer in Oracle Utilities Customer Care and Billing.

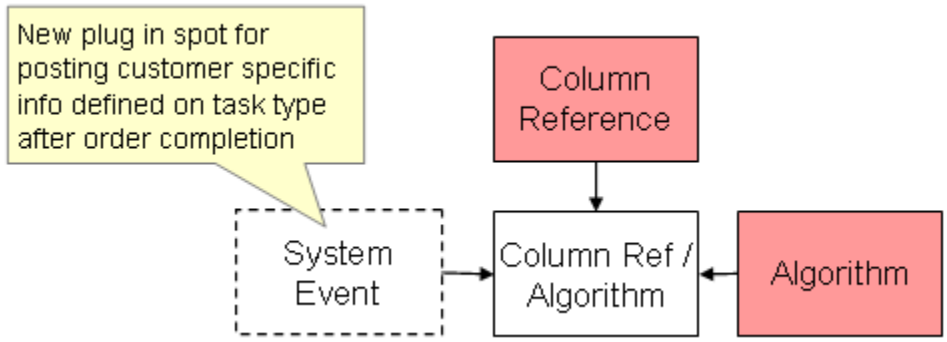


Product Offer Diagram

Copyright © 2012, Oracle. All rights reserved.



SA Creation Rule Diagram



Column Reference Diagram

A service task type is created for each product offer. In addition, a service task type is created for each SA creation rule under the product. The SA creation rule service task type captures the collection of customer/contract specific info (called product components) to be posted on the SA when order is processed.

Business Objects

The following business objects have been created in Oracle Utilities Customer Care and Billing for this integration:

- Product Offer (MO is Service Task Type)

Copyright © 2012, Oracle. All rights reserved.

- SA Creation Rule (MO is Service Task Type)
- Product Offer Sync Request
- Product Offer Outbound Message
- Siebel Integration Master Configuration (this BO was added for the customer maintenance flow and will be extended to include a default campaign code, product offer sync BO)

Examples of Product Offers in Oracle Utilities Customer Care and Billing

Following are some examples of different products that can be created in Oracle Utilities Customer Care and Billing.

Example 1: Simple Product - Single SA

The following shows a simple product offer with no customer/contract specific info (i.e. no product component information). The contract/pricing info is setup on start options and on the rate. Selection of this product will result in the creation of an electric SA in Oracle Utilities Customer Care and Billing.

Product	Description	Start Date	End Date
E-RES-EMP	Standard electric service with default rate of $0.14/kwh. Employee discount of 15% applies. Only utility employees are eligible for this product.	2010-01-01	

Product definition captures product code, description, effective period

Product	CIS Division	SA Type	Start Option
E-RES-EMP	CA	E-RES	EMP DISCOUNT

List of SAs to create at order fulfillment

Simple Product (Single SA, no customer specific info)

Example 2: Simple Product - Multi SA

The following shows a simple product offer with no customer/contract specific info (i.e. no product component information). The contract/pricing info is setup on start options and on the rate. Selection of this product will result in the creation of 2 SAs in Oracle Utilities Customer Care and Billing (electric and gas service).

Copyright © 2012, Oracle. All rights reserved.

Product	Description	Start Date	End Date
BASIC-RES	Standard electric service with default rate of $0.14/kwh. Standard gas service with default rate of $1.15/therm. All customers are eligible for this product.	2010-01-01	

Product	CIS Division	SA Type	Start Option
E-RES-STD	CA	E-RES	E-BASIC
G-RES-STD	CA	G-RES	G-BASIC

Simple Product (Multi SA, no customer specific info)

Example 3: Product Requiring Customer Specific Info

The following shows a product offer with customer/contract specific info (the price that the customer will pay for on peak and off peak kWh is only known at order fulfillment time). Selection of this product will result in the creation of an electric SA in Oracle Utilities Customer Care and Billing.

Product	Description	Start Date	End Date
E-RES-SPEC	Qualifying customers can select from different pricing options for on peak and off peak usage Customers with TOU meter installed are eligible	2010-01-01	

Product	CIS Division	SA Type	Start Option	Customer Specific Info
E-RES-SPEC	CA	E-RES	SPECIAL PRICE	KWHON-PRICE
				KWHOFF-PRICE

Customer specific info published to Siebel and required on order fulfillment

Product Requiring Customer Specific Info

Products in Siebel Energy

The following descriptions provide a high-level definition of the key product-related features in Siebel Energy:

1. **Product**: Product is a physical good or service that can be sold to a customer. Products can be implemented in a number of different ways:
    - Simple Product without Attributes: This is a product that only comes in one form, such as a book. The customer does not make any decisions about features of the product.

Copyright © 2012, Oracle. All rights reserved.

- Product Bundles: A product bundle is a group of products sold together. It cannot be customized. For example, you might offer customers vacation packages that include airfare, hotel accommodations for a specific number of days, and specific special events, all for one price.
- Simple Product with Attributes: A product with attributes has features that the customer can choose but does not have components the customer can choose. For example, a customer buying a t-shirt might be able to choose the shirt's color and its size. The shirt has two attributes, color and size. Each of these attributes has several possible values, such as white, gray, black, and S, M, L, XL.
- Product with Components: This is a product with components that a customer can select. For example, a customer buying a computer might have to select a mouse, a floppy disk drive, a monitor, and other components.

  The Simple Product with Attributes and Product with Components are known as Customizable Products.

2. **Promotion**: Product promotion is a marketing definition of a product. Product promotion is time-sensitive, and they state discounted prices and contractual terms. They may be classified as one of two types:
   - Coupons: This promotion includes a single product or service. This sort of promotion is used primarily for retail sales and is also called a retail promotion.
   - Bundled Promotions*:* This promotion includes a group of products or services.
3. **Product Configurator**: Product Configurator is a Siebel module used to configure a customizable product based on customer selection. It provides a user interface and rules engine that drives the customization of a particular product instance. Configuration rules can be setup to exclude, require, recommend, display messages, and set preference for components of a customizable product.
4. **Product Eligibility Rules**: Product Eligibility Rules specify which customers are eligible to buy products and promotions. Eligibility and compatibility checks are performed by the same workflow and the same engine at the same time. Product Compatibility (discussed next) is one step in the overall determination of eligibility.
5. **Product Compatibility Rules**: Product Compatibility Rules specify which products and promotions are compatible with each other. Eligibility and compatibility checks are performed by the same workflow and the same engine at the same time. Compatibility is one step in the overall determination of eligibility. Compatibility rules are just one of a number of different criteria that the eligibility procedure checks to determine whether a product can be sold to a customer given the customer's profile.
6. **Product Recommendation Rules**: Product Recommendations allow a Marketing Administrator to define up-sell and cross-sell recommendations for a product. The Marketing Administrator enters the text of the recommendation message and a score indicating how important this recommendation is compared with other recommendations for the product. When a sales representative uses the product catalog to add a product to a quote or order, the Messages list interactively displays the recommendations for this product in order of their score.

Note: The Product Recommendation rules outlines is standard Siebel Order Management functionality. It is not the functionality delivered by Oracle Real-Time Decisions (RTD) product.

Copyright © 2012, Oracle. All rights reserved.

7. **Siebel Product Catalog**: Siebel Product Catalog allows employees to browse for products that are available for selling. A catalog is a collection of items arranged in a logical hierarchy of categories. Categories group content items; therefore, categories can contain content items and other categories (called subcategories). Each category can belong to only one catalog and one parent category. Products can be associated with multiple categories.
8. **Price List**: Price list is a set of standard prices for products or services. A product must exist in a price list so that it appears in a catalog. So, the price list is one of the requirements for a product's visibility. A product can appear in many different price lists. A product price defined in a price list is commonly used as the starting price. Price list must be assigned to customers so that it controls the prices available to customers. Products are not visible to customers unless they are associated with price lists that are assigned to the customers.

The basic product and pricing information synchronized from Oracle Utilities Customer Care and Billing will be enriched in Siebel Energy to support product configuration models, product eligibility and compatibility rules, product recommendation (up-sell and cross-sell), promotions (product bundling and pricing discount), and catalog.

Product Data Synchronization Process

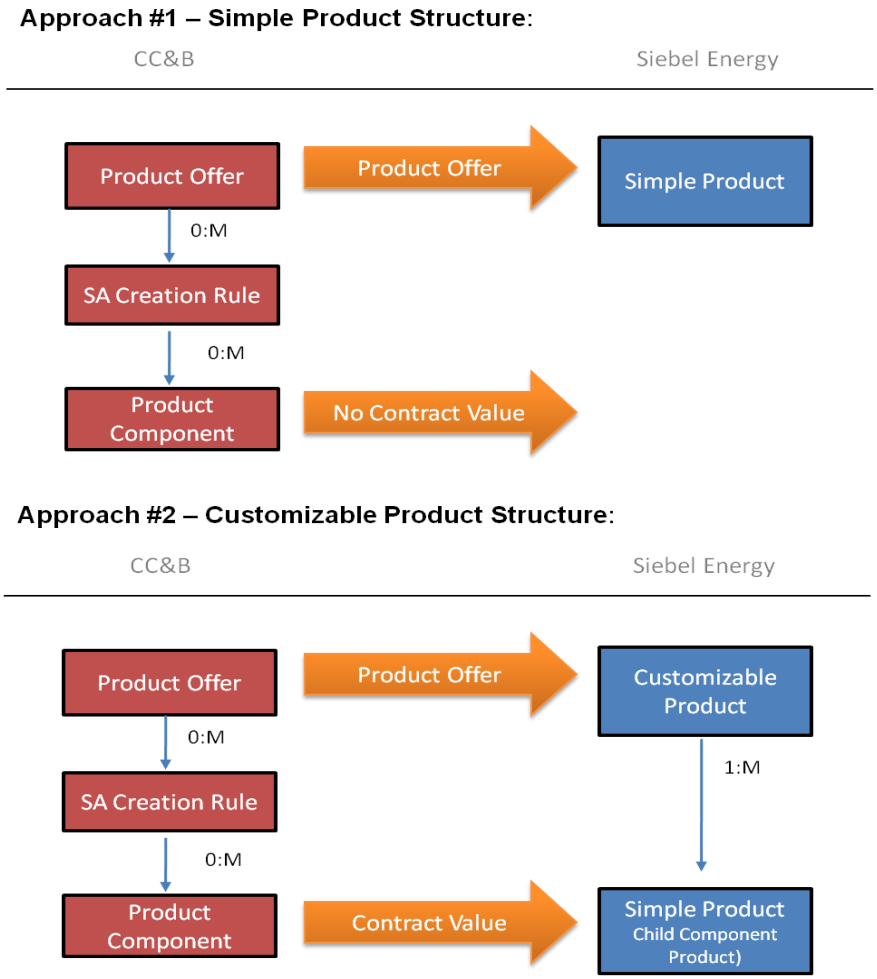
The Product Administrator will maintain the product offer as needed in Oracle Utilities Customer Care and Billing and publish the product offer to trigger the product data synchronization process to Siebel Energy.

Products in Siebel Energy are created in one of the following ways as part of this Integration:

- Simple Product without Attributes
- Product with Components - In Siebel Energy, Product with Components are known as Customizable Products.

Copyright © 2012, Oracle. All rights reserved.

The following diagram illustrates the high-level object mapping for these two different approaches:



High-level object mapping for Simple Product and Customizable Product Structure

Process Summary

The product structure created in Siebel Energy is based on the product component definition in Oracle Utilities Customer Care and Billing. The product component is used to identify what dynamic values are expected by Oracle Utilities Customer Care and Billing from an external system to support rating and billing. The product components can populate five different contract term types on the service agreement including recurring charge amount, contract value, contract rider, contract quantity, and characteristics.

Product Components

If the product offer in Oracle Utilities Customer Care and Billing has no product components defined to capture a pricing-related contract value, a simple product is created in Siebel Energy. If the product offer in Oracle Utilities Customer Care and Billing has one or more product components defined to capture a pricing-related contract value, a customizable product is created in Siebel Energy. This is based on the product offer and the pricing-related product component.

Copyright © 2012, Oracle. All rights reserved.

Pricing Information

The contract value contract term type is used to capture customer-specific price (rate) information in Oracle Utilities Customer Care and Billing. Since Siebel Energy only allows for a single price to be specific per product (quote or order line item), the product component used to capture pricing-related contract values are setup as component products for a customizable product in Siebel Energy. Customer-specific pricing can be captured on the appropriate component product and passed as part of an order processing integration flow to Oracle Utilities Customer Care and Billing. The pricing information provided from Siebel Energy for this product component is captured in the appropriate contract value on the service agreement based on the product offer setup in Oracle Utilities Customer Care and Billing.

Any product or pricing-related changes in Siebel Energy is not synchronized back into Oracle Utilities Customer Care and Billing as part of the product or pricing master.

Contract Terms

The product components used for populating recurring charge amount, contract rider, contract quantity, and characteristic contract term types on the service agreement can be included in the extract from Oracle Utilities Customer Care and Billing; however, they are not mapped to Siebel Energy based on the product integration as there is no standard way to handle these types. These product components are handled as product attributes and are very implementation specific. However, it is important that this information is captured as part of the order capture process and sent back to Oracle Utilities Customer Care and Billing as part of the order processing integration flow from Siebel Energy.

Product Offer and Product

The relationship between product offer in Oracle Utilities Customer Care and Billing and product in Siebel Energy is maintained in a cross-reference table.

The simple product and customizable product created in Siebel Energy is assigned to a price list so that it can be visible in the Siebel product catalog; therefore, the product offer synchronized from Oracle Utilities Customer Care and Billing will be defaulted to a single price list in Siebel Energy. The component products are not associated with a price list since they do not need to be visible independent of the customizable product. Instead, the original list price on the price list item is left blank. The effective start date for the price list item is set to the creation date/time and the effective end date will not be set.

The product offer description includes user-readable information about the pricing details since specific pricing is not synchronized from Oracle Utilities Customer Care and Billing to Siebel Energy.

Deleted Product Offer

If a product offer is deleted in Oracle Utilities Customer Care and Billing, no update is sent to Siebel Energy and the corresponding product in Siebel Energy is not deleted or inactivated automatically. The product must be inactivated manually in Siebel Energy. If you delete a product offer in Oracle Utilities Customer Care and Billing that has already been synchronized to Siebel Energy, the cross-reference data for the synchronized product is not deleted. It is recommended that product offers not be deleted in Oracle Utilities Customer Care and Billing but instead inactivate the product in Oracle Utilities Customer Care and Billing by setting the end date and publishing the change to Siebel Energy.

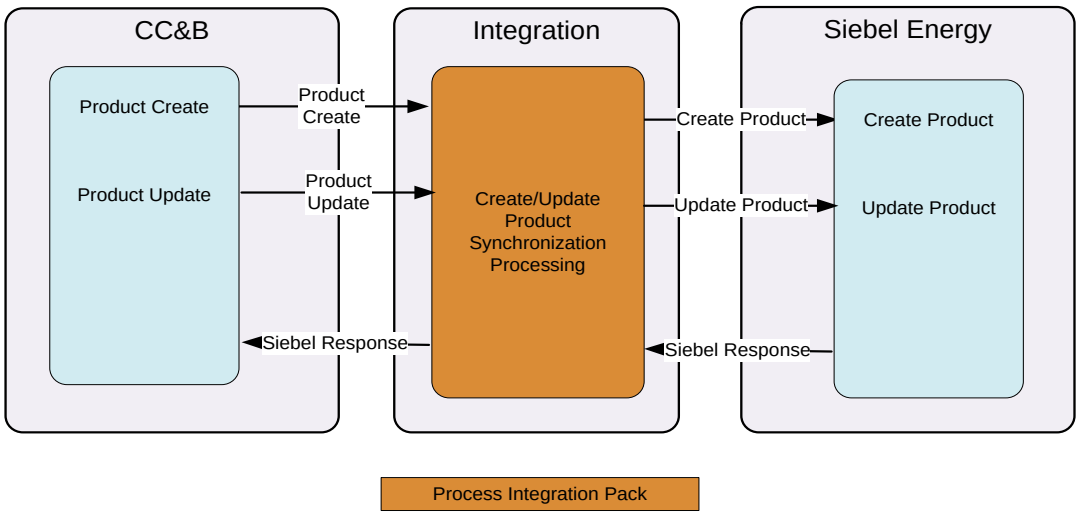
Copyright © 2012, Oracle. All rights reserved.

Supported Functionality

Products are created in Oracle Utilities Customer Care and Billing (CC&B) and sent to Siebel Energy. Oracle Utilities Customer Care and Billing is the Product master. The products created are used by the Order Capture module in Siebel Energy. The product synchronization integration flow also enables updates to existing products in Oracle Utilities Customer Care and Billing. The updates are then synchronized to Siebel Energy.

The following integration flows are implemented as part of Product Synchronization between Siebel and Oracle Utilities Customer Care and Billing:

- The Synchronization (Create/Update) Product/Pricing Integration Flow, which interfaces Oracle Utilities Customer Care and Billing Products to create and update product data in Siebel Energy



Create/Update Product and Pricing Integration Flow diagram

Assumptions and Constraints

The following are the assumptions and constraints related to this integration between edge applications, Oracle Utilities Customer Care and Billing and Siebel Energy:

- Oracle Utilities Customer Care and Billing is the product and pricing master.
- The product and pricing information is synchronized one way from Oracle Utilities Customer Care and Billing to Siebel Energy. Any changes in Siebel Energy, however, are not synchronized back into Oracle Utilities Customer Care and Billing as part of the product or pricing master.
- The initial system loading (data conversion) of product and pricing information from Oracle Utilities Customer Care and Billing to Siebel Energy is supported by the standard integration, dependent upon both systems with the integration in place and configured accordingly when the product information is set up in Oracle Utilities Customer Care and Billing.
- The Utility Service products synchronized from Oracle Utilities Customer Care and Billing is associated with a single price list in Siebel Energy and the original list price on the price list

Copyright © 2012, Oracle. All rights reserved.

item is left blank.

- Pricing information is captured as a description in Oracle Utilities Customer Care and Billing and passed to Siebel Energy.
- Product Configurator Model, Product Eligibility and Compatibility Rules, Product Recommendations (Up-Sell and Cross-Sell), Promotion (Promotion Bundles and Pricing Discounts), and Catalog are defined in Siebel Energy.
- No changes to application-level security (that is, read-only) are implemented in Siebel Energy or Oracle Utilities Customer Care and Billing as part of this integration product to restrict data updates.
- This integration supports the use of only one language for any given implementation. Therefore, any descriptions will be synchronized as is between Oracle Utilities Customer Care and Billing and Siebel Energy.

Chapter 3: Understanding Integration Interfaces and Components

This chapter describes the Create/Update product integration flows and discusses Siebel Energy and Oracle Utilities Customer Care and Billing interfaces, industry Oracle Application Integration Architecture (AIA) components, and Integration services.

This chapter includes the following sections:

- Product Data Integration Flow
- Oracle Utilities Customer Care and Billing Interfaces
- Siebel Energy Interfaces
- Industry Oracle Application Integration Architecture (AIA) Components
- Integration Services

These entities do not require configuration; however, they can be helpful in understanding the integration as a whole.

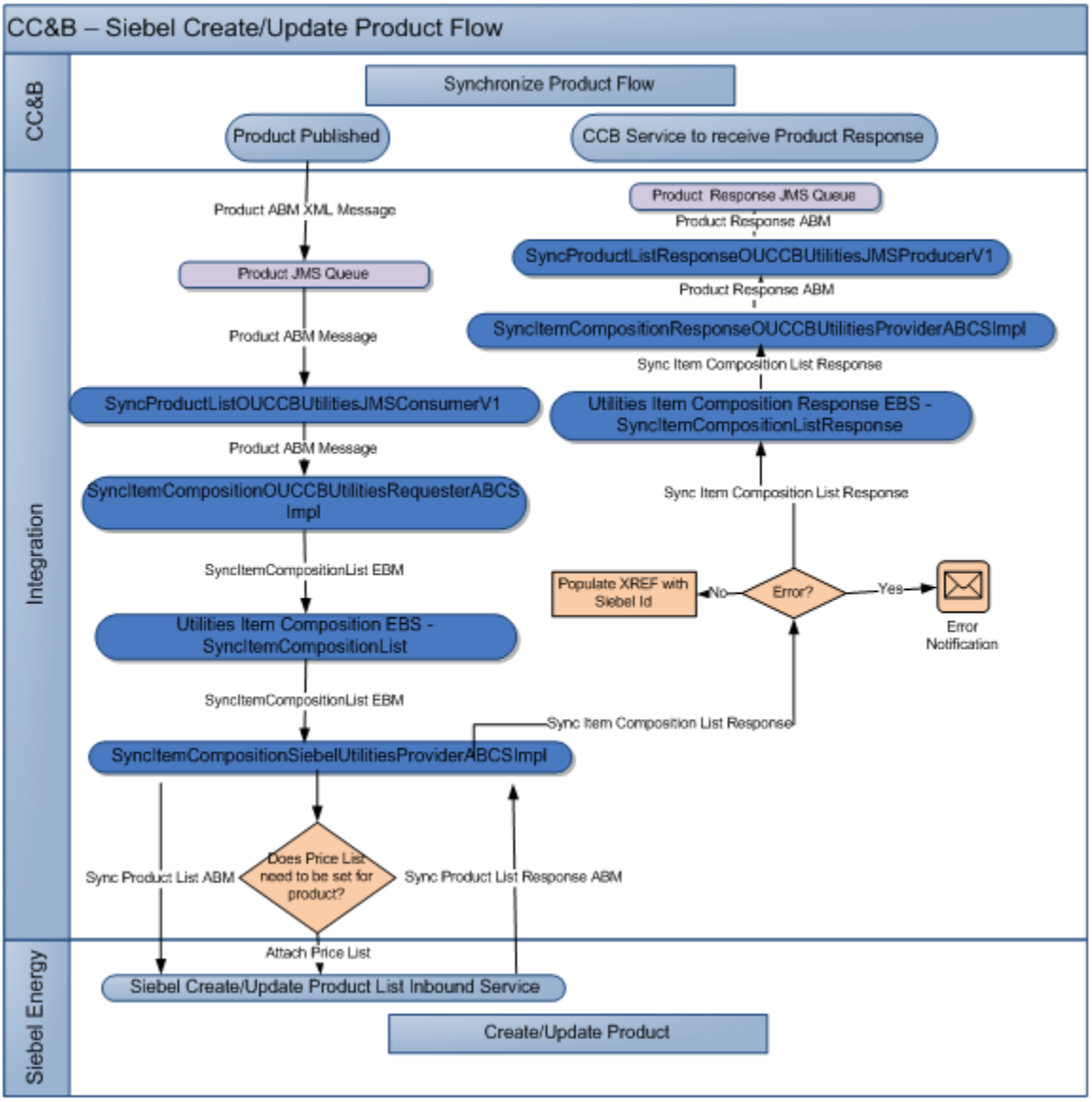
Note: The Utilities EBO and EBM XSD files can be located by EBO within the $AIA_HOME/AIAMetaData/AIAComponents/EnterpriseObjectLibrary/industry/Utilities/EBO/ parent folder. The core EBS WSDL files can be located by EBO within the $AIA_HOME/AIAMetaData/AIAComponents/EnterpriseBusinessServiceLibrary/industry/Utilities/EBO/ parent folder.

For more information about EBOs, see Appendix B: Cross-References.

Copyright © 2012, Oracle. All rights reserved.

Product Data Integration Flow

The following diagram illustrates the Create/Update product flow for Oracle Utilities Customer Care and Billing (CC&B) and Siebel Energy.



Oracle Utilities Customer Care and Billing - Siebel Product create/update flow diagram

Processing Details

The following are the Processing details for this integration process:

1. **Publish Products:** In Oracle Utilities Customer Care and Billing, a Product administrator configures a product and publishes it to Siebel. The product is published to an Oracle Utilities Customer Care and Billing Product JMS Queue.
2. **Product JMS Consumer:** SyncProductListOUCCBUtilitiesJMSConsumerV1 consumes the Oracle Utilities Customer Care and Billing Product queue message and invokes the SyncItemCompositionOUCCBUtilitiesRequesterABCSImpl with the operation SyncProduct.

Copyright © 2012, Oracle. All rights reserved.

3. **Oracle Utilities Customer Care and Billing Item Composition Requester ABCS:** The SyncItemCompositionOUCCBUtilitiesRequesterABCSImpl, transforms the Oracle Utilities Customer Care and Billing Product message into **ItemCompositionEBM** and calls the UtilitiesItemCompositionEBSV1 with the operation SyncItemCompositionList. It also makes entries in the XREF table for the CCB Parent Product Id and Component Ids.
4. **Utilities Item Composition EBS:** UtilitiesItemCompositionEBS is a routing service that routes the message to SyncItemCompositionSiebelUtilitiesProviderABCSImpl.
5. **Utilities Siebel Item Composition Provider ABCS:** The SyncItemCompositionSiebelUtilitiesProviderABCSImpl tranforms the ItemCompositionEBM into Siebel product message and calls the Siebel product webservice 'SWIProductImportUpsert'.
6. The Siebel web service processes the request and either creates or updates the product in Siebel workspace and returns a response message.
7. **Utilities Siebel Item Composition Provider ABCS:** If the parent Product does not exist already, then another call is made to webservice 'SWIProductImportUpsert' to associate the parent product with a default PriceList. The PriceList is read from AIAConfigurationProperties file.
8. The SyncItemCompositionSiebelUtilitiesProviderABCSImpl then transforms the Siebel Response to ItemCompositionResponseEBM and calls the UtilitiesItemCompositionResponseEBS with the operation SyncItemCompositionListResponse. It also makes entries in the XREF table for the Siebel Product Ids.
9. **Item Composition Response EBS:** UtilitiesItemCompositionResponseEBSV1 is a routing service that routes the message to SyncItemCompositionResponseOUCCBUtilitiesProviderABCSImpl.
10. **Oracle Utilities Customer Care and Billing Product Response Provider:** The Product Response Provider ABCS, SyncItemCompositionResponseOUCCBUtilitiesProviderABCSImpl transforms the response message to CCB Response ABM format and invokes the JMS Producer – SyncProductListResponseOUCCBUtilitiesJMSProducerV1.
11. **Product Response JMS Producer:** The Product Response JMS Producer, SyncProductListResponseOUCCBUtilitiesJMSProducerV1 puts the message in a queue to be picked up by Oracle Utilities Customer Care and Billing, which reads the response from the queue and updates the Sync Request Status.

Components used in this Integration flow:

- SyncProductListOUCCBUtilitiesJMSConsumerV1
- SyncItemCompositionOUCCBUtilitiesReqABCSImpl
- UtilitiesItemCompositionEBS
- SyncItemCompositionSiebelUtilitiesProvABCSImpl
- UtilitiesItemCompositionResponseEBS
- SyncItemCompositionResponseOUCCBUtilitiesProvABCSImpl
- SyncProductListResponseOUCCBUtilitiesJMSProducerV1

Copyright © 2012, Oracle. All rights reserved.

Technical Assumptions

The following Technical Assumptions are described in this section:

- Oracle Utilities Customer Care and Billing (CC&B) will write Product Data message to a Product JMS queue on the SOA Server.
- Siebel interfaces to create/update Product Data message will be through an inbound web service invoked from Integration. The web service call to Siebel is synchronous.
- Siebel is invoked with an Upsert operation for Product, i.e Siebel checks if the product exists, if yes it will update the Product else it will insert the Product.
- Description in Oracle Utilities Customer Care and Billing (CC&B) is mapped to Product Name in Siebel and hence Description in CC&B should be unique across all products when synchronized with Siebel.
- The Price List creation in Siebel is a pre-requisite to Product Creation. The task of creating a price list is an independent task in Siebel and should be done prior to running the integration. The Siebel Price List Id and Price List Name are pre-configured in AIA Configuration Properties file and will be used during creation of Product in Siebel.
- The response from Siebel will be put into a queue for Oracle Utilities Customer Care and Billing (CC&B) to consume.
- Cross Reference between Oracle Utilities Customer Care and Billing (CC&B) Product Id and Siebel Product Id will be maintained through an XREF table.
- No Integration Ids are stored in Siebel. Oracle Utilities Customer Care and Billing (CC&B) Product Id will be stored as Part# on Siebel Product.
- If the SOA server is down or unavailable, Oracle Utilities Customer Care and Billing (CC&B) message will be in Error Status.
- If Siebel is down messages will be rolled back to Oracle Utilities Customer Care and Billing (CC&B) Product Error queue.

Oracle Utilities Customer Care and Billing Interfaces

This section describes in detail the Edge Application Interfaces for Product Data Synchronization which consists of the following:

Outbound CC&B Message

Message Name	Invokes	Message Description
C1-ProductOfferSyncReqOutMsg	Oracle Utilities Customer Care and Billing Work Order Requester ABCS	Product Offer information

Copyright © 2012, Oracle. All rights reserved.

Inbound CC&B Message

Message Name	Populated By	Message Description
F1-UpdateAndTransitionSyncRequest	Oracle Utilities Customer Care and Billing Product Response Provider ABCS	Product Response updated in the Sync Request.

Siebel Energy Interfaces

Inbound Siebel Web Services

Web Service Name	Invoked By	Web service Description
SWIProductImport	Siebel Item Composition Provider ABCS	Invoked for Creating and Updating a product and associating the product with a Price List in Siebel.

Industry Oracle Application Integration Architecture (AIA) Components

Note: The core EBO and EBM XSD files can be located by EBO within the $AIA_HOME/AIAMetaData/AIAComponents/EnterpriseObjectLibrary/Core/EBO/ parent folder. The core EBS WSDL files can be located by EBO within the $AIA_HOME/AIAMetaData/AIAComponents/EnterpriseBusinessServiceLibrary/Core/EBO/ parent folder.

The integration flow uses the following components:

EBO	EBM	File Locations
ItemCompositionEBO	SyncItemCompositionEBM - Used for Product messages SyncItemCompositionResponseEBM - Used for Product Response/Acknowledgement message	The core EBO and EBM XSD files are in: $AIA_HOME/AIAMetaData/AIAComponents/EnterpriseObjectLibrary/Core/EBO/ parent folder.

For more information about EBOs, see Appendix B: Cross-References.

Copyright © 2012, Oracle. All rights reserved.

Integration Services

ABCS

These values are cross-referenced in the configuration guidelines section for the integration.

ABCS Name	Description
SyncItemCompositionOUCCBUtilitiesReqABCSImpl	Oracle Utilities Customer Care and Billing Item Composition Requester ABCS
SyncItemCompositionSiebelUtilitiesProvABCSImpl	Siebel Item Composition Provider ABCS
SyncItemCompositionResponseOUCCBUtilitiesProvABCSImpl	Oracle Utilities Customer Care and Billing Item Composition Requester ABCS

SyncItemCompositionSiebelUtilitiesProvABCSImpl uses AIASessionPoolMAnager

For more information on AIA Session Pool Manager, see Appendix C: Using Session Pool Manager.

Enterprise Business Services

EBS Name	Description
UtilitiesItemCompositionEBS	Receives the SyncItemCompositionEBM and routes it to the Siebel Item Composition Provider ABCS.
UtilitiesItemCompositionResponseEBS	Receives the SyncItemCompositionResponseEBM and routes it to the CCB Product Response JMS Producer.
The core EBS WSDL files are located in: $AIA_HOME/AIAMetaData/AIAComponents/EnterpriseBusinessServiceLibrary/Core/EBO/ parent folder	

Copyright © 2012, Oracle. All rights reserved.

Adapter Services

Adapter Service Name	Description
SyncProductListOUCCBUtilitiesJMSConsumerV1	This is the JMS consumer service responsible for listening to the Product JMS Queue for Oracle Utilities Customer Care and Billing and sending the messages to Oracle Utilities Customer Care and Billing Item Composition Requester ABCS.
SyncProductListResponseOUCCBUtilitiesJMSProducerV1	This is the JMS producer service that is invoked when the Product Response EBM is routed to Oracle Utilities Customer Care and Billing. This service is responsible for posting the message to the Product Response JMS Queue for Oracle Utilities Customer Care and Billing.

Queues

Adapter Service Name	Description
AIA_OUCCBProductJMSQueue	Product messages published by CC&B into Integration Queue
AIA_OUCCBProductJMSQueue_ErrorQ	Product Error Queue
AIA_OUCCBProductResponseJMSQueue	Product Response Queue
AIA_OUCCBProductResponseJMSQueue_ErrorQ	Product Response Error Queue

Note: PIP creates an error queue for each main queue in integration with the name of the main queue followed by _Error

Copyright © 2012, Oracle. All rights reserved.

Part 2: Implementing the Delivered Integrations

Chapter 4: Configuration Guidelines

Chapter 5: Monitoring, Error Handling and Troubleshooting

Chapter 6: Extensibility Options

Chapter 7: Security

Copyright © 2012, Oracle. All rights reserved.

Chapter 4: Configuration Guidelines

This chapter provides details on the required configuration settings for the integration process and provides post-install configuration instructions. The following configuration settings are described on how to:

- Set up Oracle Utilities Customer Care and Billing
- Set up Siebel Energy
- Set up the Product Data Synchronization Integration Pack

For more information about installation and deployment of your PIP, see *Oracle Fusion Middleware Installation and Upgrade Guide* for Oracle Application Integration Architecture Foundation Pack.

Setting up Oracle Utilities Customer Care and Billing

To set up Oracle Utilities Customer Care and Billing for the integration, configure the following:

- Admin Data Setup
- System Data Setup
- JMS Configuration
- XAI Configuration

For more information on configuring and working with Oracle Utilities Customer Care and Billing, refer to the Oracle Utilities Customer Care and Billing user documentation.

Note: Some configurations described may be required for general functionality and do not necessarily relate directly to the integration; however, these are called out as particularly significant configuration items. The inclusion of such items does not mean that other general items that are not mentioned do not need to be configured.

Configure Admin Data Tables

This section details how to configure the Admin Data Tables and the unique setup issues specifically related to configuring your system for the integration.

Copyright © 2012, Oracle. All rights reserved.

For more information about configuring Oracle Utilities Customer Care and Billing, refer to the Oracle Utilities Customer Care and Billing User Guide.

Master Configuration

Your implementation should configure an instance of the Siebel integration master configuration in Oracle Utilities Customer Care and Billing. The master configuration contains pertinent information needed for a Siebel integration, including details used when publishing product offers to Siebel Energy.

Navigation	Guideline
Admin Menu → Master Configuration	Create a master configuration to be used Siebel integration.

See the demo environment for a sample Siebel Integration master configuration. Embedded help is available on the master configuration to guide you through the setup process.

Campaign

Setup a default campaign to be used on product offers. Ensure that this campaign is referenced on the Siebel Integration master configuration, as well as on the product offers that your implementation configures. A few things to note when setting up this campaign:

- It must not include any questions and miscellaneous fields since these are not populated during order processing.
- A campaign eligibility system event is required on all campaigns. The base product includes algorithm C1-ELIGIBLE that can be specified on the product offer campaign.

Navigation	Guideline
Main Menu → Sales & Marketing → Campaign	Setup a campaign to use on product offers

Column Reference

Setup the column references your implementation may require for order processing. A column reference is used to identify each product component or miscellaneous element related to the product offer. The column reference controls how the information is processed on order completion.

Navigation	Guideline
Admin Menu → Column Reference	Setup product components required by product offers

Copyright © 2012, Oracle. All rights reserved.

Product Offers

Configure your product offers and the corresponding SA creation rules. Each SA creation rule controls the type of service agreement(s) that are created if this product offer is selected.

Navigation	Guideline
Admin Menu → Product Offer	Configure your product offers

See the demo environment for sample product offers. Embedded help is available to guide you through the setup process

Configure System Data Tables

The System Data Tables describes the Look Up Field values, Business Object items, and Data Area elements. Each of these System Data Tables is described in detail in the succeeding sections.

Look Up Field

There are a few Lookup Field values that your implementation must decide on before setting up your product offers in Oracle Utilities Customer Care and Billing.

Look Up Field	Description
C1_PROD_TYPE_FLG	This lookup field captures the values for Product Type. Product type is used to classify the product offers. For example: Electric, Gas, Water, Dual Fuel, and so forth.
C1_PROD_COMP_TYPE_FLG	This lookup field captures the values for Product Component Type. Product component type is used to categorize the product components (or column references) defined on product offers. For example, Pricing Component, Other Component, and so forth.

Copyright © 2012, Oracle. All rights reserved.

Business Object

Business Object items are described in the following table:

Business Object	Description
C1-ProductOfferTaskType	This business object is used to define the behavior of a product offer. To configure your system for product offer using this business object: - Ensure that a master configuration is set up for Siebel integration - A product type must be assigned to each product offer. Product types are used to classify product offers, for example electric, gas, dual fuel and so forth. Your implementation must define valid product types. Navigate to the lookup page and define valid values for the lookup field C1_PROD_TYPE_FLG. - Create a campaign that will be used for your product offers. In addition to the standard BO options, the following BO options are relevant to product offer business objects: - SA Creation Rule BO: Defines the business object for the product offer's SA creation rules. Product offers are configured in Oracle Utilities Customer Care and Billing and published to Siebel Energy. These product definitions are then further enhanced in Siebel and offered to end customers. When a customer accepts a product offer a Siebel order is created for processing. If the Siebel order requires processing in Oracle Utilities Customer Care and Billing, the assumption is that a service task will be created in Oracle Utilities Customer Care and Billing and any necessary actions executed within the service task's lifecycle. Anticipated actions are as follows: - Given the product offer on the order line, determine which SA Creation Rules are eligible. - Create a package using the campaign defined on the product offer. This package captures the eligible SA Creation Rules. - Add an order for the new package and complete the order. The package can then be deleted.
C1-SACreationRuleTaskType	This business object defines the behavior of the SA creation rules for a product offer. Each SA creation rule indicates the type of service agreement to create for the product offer. It also captures the product components or column references used when setting up the service agreement. To configure your system for SA creation rules using this business object: - Set up SA types, SP types, start options, and so forth, required for the product offer - Set up column references for additional attributes that may be needed on the resultant SA(s) - A product component type must be assigned to each product component. This is used to categorize product components used as pricing components for example, Navigate to the lookup page and define valid values for the lookup field C1_PROD_COMP_TYPE_FLG.

Copyright © 2012, Oracle. All rights reserved.

Business Object	Description
C1-ProductOfferSyncRequest	This business object defines the behavior of an outbound sync request used when publishing product offers to Siebel Energy. In addition to the standard options, the following BO options are relevant to product offer outbound sync request business objects: - Outbound Message Type: This contains a reference to the outbound message BO to use. The base package includes BO C1-ProductOfferSyncReqOutMsg for the Siebel Product Offer Sync - External System: This contains the reference to the outbound message type and its corresponding configuration for communicating with the external system. The base package includes the message XSL C1-CCBJMSQAddNamespace.xsl - Product Offer Read BO and SA Creation Rule Read BO: Contains the business objects used to read the product offer information. The base product includesC1-ProductOffer and C1-SACreationRule to be used for this purpose. - Sync Request DA: Contains the data area that has all the elements needed in the product offer sync request. The base product includes C1-ProductOfferSnapshot to be used for this purpose.
C1-ProductOfferSyncReqOutMsg	This business object is used to publish product offers in Siebel integration. Define this BO on the outbound message type that your implementation configures for the Siebel Product Offer sync.
C1-ProductOffer	This business object contains the product offer elements that are relevant when publishing product offers to Siebel Energy. It is used to read product offer information when building the initial and final snapshots for the sync request. If your implementation wishes to use this BO, define it as the Product Offer Read BO option on the sync request BO. The base product includes BO C1-ProductOfferSyncRequest for this purpose.
C1-SACreationRule	This business object contains the SA creation rule elements that are relevant when publishing product offers to Siebel Energy. It is used to read SA creation rule information when building the initial and final snapshots for the sync request. If your implementation wishes to use this BO, define it as the SA Creation Rule Read BO option on the sync request BO. The base product includes BO C1-ProductOfferSyncRequest for this purpose.

Data Area

Data Area description is as follows:

Data Area	Description
C1-ProductOfferSnapshot	This data area describes the elements that are included in the Product Offer Sync Request Outbound Message.

Copyright © 2012, Oracle. All rights reserved.

JMS Configuration

This section describes the JMS configuration required in the Oracle Utilities Customer Care and Billing Weblogic server and in the Oracle Utilities Customer Care and Billing deployment XML files. Configure the JMS to receive JMS messages from the integration layer.

WebLogic Server JMS Configuration

To configure JMS in the Oracle Utilities Customer Care and Billing WebLogic server, log on to the console using the URL http://<server_name>:<port_number>/console.

For example: http://ccbserver:7001/console

Creating a JMS Module

You can create a new JMS module in the WebLogic console.

To create a JMS module used for remote queue configuration:

1. Open the WebLogic console and create a new JMS module.
2. Enter a meaningful name for the JMS module. This JMS module is used to create configurations which consume messages from remote WebLogic queues.

   For example: SiebelIntegrationModule

Copyright © 2012, Oracle. All rights reserved.

Creating a Foreign Server

Create a new Foreign server under the JMS module in the WebLogic console.

To create the Foreign Server used for remote queue configuration:

1. Enter the WebLogic console and select the JMS module created for the integration.
2. Create a Foreign server under the JMS module.
3. Enter the following for the Foreign server:
    - **Name** – Name for the Foreign server. For example: CCBSiebelForeignServer
    - **JNDI Initial Context Factory** – ***weblogic.jndi.WLInitialContextFactory***
    - **JNDI Connection URL** – Add the URL for the Integration SOA server. For example: **t3://soaserver.com:soa-port**
    - **JNDI Properties Credential** – Password for the SOA server user
    - **JNDI Properties** - ***java.naming.security.principal=<SOA Server user>*** For example: weblogic
4. Under the Foreign server, create a foreign destination for each remote queue.
    - **Name** – Name of the foreign destination.
    - **Local JNDI Name** – Add a local JNDI name for the Integration Queue. Local JNDI name is later added manually as part of configuration in weblogic-ejb-jar.xml → <weblogic-enterprise-bean> → <message-driven-descriptor> → <destination-jndi-name>.
    - **Remote JNDI Name** – JNDI name of the queue on the Integration SOA server.

Destination Name	Local JNDI Name	Remote JNDI Name
AIA_OUCCBProductResponseJMSQueue	jms/aia/LocalAIA_OUCCBProductResponseJMSQueue	jms/aia/AIA_OUCCBProductResponseJMSQueue

5. Under the Foreign server, create a Remote Connection Factory.
    - **Name** – Name of the remote connection factory.
    - **Local JNDI Name** – Add a local JNDI name to the Integration Connection Factory. This JNDI name is added manually later as part of configuration in WebLogic-ejb-jar.xml → <weblogic-enterprise-bean> → <message-driven-descriptor> → <connection-factory-jndi-name>.

Copyright © 2012, Oracle. All rights reserved.

- **Remote JNDI Name** – JNDI name of the JMS Connection factory on the Integration SOA server. For example:

Connection Factory Name	Local JNDI Name	Remote JNDI Name
OUCCBSiebelConnectionFactory	jms/aia/LocalAIAOUCCBCF	jms/aia/AIAOUCCBCF

Configuration File Changes

To make changes to the Configuration files, use the following:

Configuring Message Driven Beans (MDB)

It is recommended to use the Oracle Utilities Customer Care and Billing template and Customer Modification (CM) feature to make changes to these configuration files. This ensures that modifications cannot be overwritten by future application patches.

Modifying filesejb-jar.xml and weblogic-ejb-jar.xml

These files are located at Oracle Utilities Customer Care and Billing Enterprise Archive (EAR) file

- The Oracle Utilities Customer Care and Billing configuration files, ejb-jar.xml and ejb-weblogic-jar.xml, must be modified to configure Message Driven Beans (MDB). MDBs receive messages from the integration queues. These files are part of the Oracle Utilities Customer Care and Billing Enterprise Archive (EAR) file.
- The Oracle Utilities Customer Care and Billing application needs to be redeployed after these changes are made.

Managing configuration files

Configuration files such as config.xml, ejb-jar.xml, and ejb-weblogic-jar.xml are managed through template configuration files which reside in the environment directory. When the initialSetup.sh script is executed, environment specific information is combined with the template to create the target file which is then deployed to the correct location. When the environment is started (spl.sh start), changes are automatically deployed to WebLogic.

Overriding the existing template

It is possible to override the existing template by providing another template file in the same location, with the same name but prefixed with ".cm". For example, if etc/cm.ejb-jar.xml.template is found when initialSetup is run, the cm.ejb-jar.xml.template is used to generate the target ejb-jar.xml file.

Copyright © 2012, Oracle. All rights reserved.

Enabling changes for integration

To enable your changes for the integration, it is recommended that you first make a "CM" copy of the existing template and then make changes to the CM version. If there are any problems with starting the application, delete the CM versions of the files and rerun initialSetup to regenerate and redeploy the original versions.

- If CM versions of the template files are made and later a patch is installed which updates the base template, then the CM version is not updated.

Creating a new MDB

You can create a new MDB to receive messages from each integration inbound queue. For simplicity, the names of the target configuration files are referred to in the following examples; however, it is recommended to make changes in the etc/cm.<target file>.template version of the file, and then execute initalSetup.sh (Unix) or initalSetup.cmd (Windows) to deploy the generated file.

Note: Working examples of the configuration files are available for download from My Oracle Support in patch 13545553 - CCB - SBL INTEGRATION CONFIGURATION EXAMPLES . Before installing the examples, read the Product Fix design document included in the patch for more information.

To create a new MDB receive messages from integration inbound queue:

1. Create an MDB for each Oracle Utilities Customer Care and Billing inbound queue to receive messages and invoke CCB service.
2. Modify the ejb-jar.xml and weblogic-ejb-jar.xml files to configure the MDBs.
3. Add the <message-driven> and <container-transaction> tag for each inbound queue in ejb-jar.xml. Add a security role with role “cisusers” in ejb-jar.xml. Following is an example:

```
<?xml version="1.0" encoding="UTF-8"?>
<ejb-jar
    version="2.1"
    xmlns="http://java.sun.com/xml/ns/j2ee"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://java.sun.com/xml/ns/j2ee
http://java.sun.com/xml/ns/j2ee/ejb-jar_2_1.xsd">
    <display-name>ServiceBean</display-name>
    <enterprise-beans>
        <session>
            <description>Deployment descriptor for the Service
Bean</description>
            <display-name>Service Bean</display-name>
            <ejb-name>SPLServiceBean</ejb-name>
            <home>com.splwg.ejb.service.ServiceHome</home>
            <remote>com.splwg.ejb.service.Service</remote>
            <ejb-class>com.splwg.ejb.service.impl.ServiceBean</ejb-
class>
            <session-type>Stateless</session-type>
```

Copyright © 2012, Oracle. All rights reserved.

```
            <transaction-type>Bean</transaction-type>
            <env-entry>
               <description>XAI HTTP Caller URL</description>
               <env-entry-name>XAIServerURL</env-entry-name>
               <env-entry-type>java.lang.String</env-entry-type>
               <env-entry-value>http://localhost: CCB-
port/spl/XAIApp/xaiserver</env-entry-value>
            </env-entry>
            <env-entry>
               <description>XAI HTTP Caller User</description>
               <env-entry-name>HTTPBasicAuthUser</env-entry-name>
               <env-entry-type>java.lang.String</env-entry-type>
               <env-entry-value>SPLXAI</env-entry-value>
            </env-entry>
            <env-entry>
               <description>XAI HTTP Caller User
password</description>
               <env-entry-name>HTTPBasicAuthPasswordEnc</env-entry-
name>
               <env-entry-type>java.lang.String</env-entry-type>
               <env-entry-value>splxai00</env-entry-value>
            </env-entry>
         </session>

         <!—Product Integration Point -->
          <message-driven>
            <description>MDB for Product Response</description>
            <display-name>AIA_OUCCBProductResponseJMSQueue</display-
name>
            <ejb-name>AIA_OUCCBProductResponseJMSQueue</ejb-name>
            <ejb-class>com.splwg.ejb.mdb.MessageProcessor</ejb-
class>
            <messaging-type>javax.jms.MessageListener</messaging-
type>
            <transaction-type>Bean</transaction-type>
            <message-destination-type>javax.jms.Queue</message-
destination-type>
         </message-driven>

    </enterprise-beans>

    <assembly-descriptor>

        <security-role>
            <role-name>cisusers</role-name>
        </security-role>

        <!-- Product Integration Point -->
        <container-transaction>
            <method>
                <ejb-name>AIA_OUCCBProductResponseJMSQueue</ejb-
name>
                <method-name>onMessage</method-name>
            </method>
            <trans-attribute>NotSupported</trans-attribute>
        </container-transaction>
    </assembly-descriptor>
```

Copyright © 2012, Oracle. All rights reserved.

```
</ejb-jar>
```

4. Modify weblogic-ejb-jar.xml. Add the <weblogic-enterprise-bean> tag for each inbound queue. Add a security role with the role “cisusers”.
5. References in the <weblogic-enterprise-bean> tag are as follows:
    - <ejb-name> - MDB name given in the ejb-jar.xml file.
    - <destination-jndi-name> - JNDI name provided in JMS module→ Foreign server → Foreign destination → Local JNDI name.
    - <connection-factory-jndi-name> - JNDI name provided in JMS module → Foreign server → Remote Connection Factory → Local JNDI name. Following is an example:

```
<?xml version="1.0" encoding="UTF-8"?>
<!--
  ** This file was automatically generated by
  ** EJBGen WebLogic Server 10.0 SP0  Wed May 9 18:10:27 EDT 2007
933139
-->
<weblogic-ejb-jar
  xmlns="http://www.bea.com/ns/weblogic/90"
xmlns:j2ee="http://java.sun.com/xml/ns/j2ee"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.bea.com/ns/weblogic/90
http://www.bea.com/ns/weblogic/90/weblogic-ejb-jar.xsd">
  <weblogic-enterprise-bean>
    <ejb-name>SPLServiceBean</ejb-name>
    <jndi-name>spl/servicebean</jndi-name>
  </weblogic-enterprise-bean>

     <!--Product Integration Point -->
     <weblogic-enterprise-bean>
        <ejb-name>AIA_OUCCBProductResponseJMSQueue</ejb-name>
        <message-driven-descriptor>
            <pool>
                <max-beans-in-free-pool>5</max-beans-in-free-pool>
                <initial-beans-in-free-pool>1</initial-beans-in-
free-pool>
            </pool>
            <destination-jndi-
name>jms/aia/LocalAIA_OUCCBProductResponseJMSQueue</destination-
jndi-name>
            <connection-factory-jndi-
name>jms/aia/LocalAIAOUCCBCF</connection-factory-jndi-name>
        </message-driven-descriptor>
    </weblogic-enterprise-bean>

    <security-role-assignment>
        <role-name>cisusers</role-name>
        <principal-name>cisusers</principal-name>
    </security-role-assignment>
```

Copyright © 2012, Oracle. All rights reserved.

```
</weblogic-ejb-jar>
```

XAI Configuration

The XAI Configuration changes can be done by the following:

Creating XAI JNDI Server

Create a new XAI JNDI server pointing to the Integration SOA server.

To create XAI JNDI server configured to communicate with integration layer:

1. In the Admin menu, navigate to XAI JNDI server.
2. Enter the XAI JNDI server name. For example: ***SIEBEL_JNDI***
3. Enter the XAI JNDI server description For example: ***Siebel Integration server***
4. Enter the Provider URL in the format **t3//<SOA Server>: <SOA Port>.** For example: t3://soaserver.us.oracle.com:8002

Creating XAI JMS Queue

Create a new XAI JMS queue to publish products where Oracle Utilities Customer Care and Billing sends messages.

To create XAI JMS queue:

1. In the Admin menu, navigate to XAI JMS queue.
2. Enter the following:
    - **XAI JMS Queue –** Queue name in Oracle Utilities Customer Care and Billing.
    - **Description –** Queue description
    - **Queue Name –** JNDI name of the queue on the Integration server. For example: ***jms/aia/AIA_OUCCBProductJMSQueue***.
    - **Target Client Flag – *JMS***
    - **XAI JNDI Server –** Select the XAI JNDI server created for integration.

Copyright © 2012, Oracle. All rights reserved.

XAI JMS Queue	Description	Queue Name	Target Client Flag	XAI JNDI Server
ProdSyncReq	Product Sync Request	jms/aia/AIA_OUCCBProductJMS Queue	JMS	SIEBEL_JNDI

Creating XAI JMS Connection

Create a new XAI JMS connection used to connect to the integration queues.

To create XAI JMS Connection:

1. In the Admin menu, navigate to XAI JMS connection.
2. Enter the following:
    - **XAI JMS Connection –** Connection name in Oracle Utilities Customer Care and Billing.
    - **Description –** Connection description
    - **XAI JNDI Server –** Select the XAI JNDI server created for this integration (as described in the XAI JNDI Server section).
    - **JNDI ConnectionFactory –** JNDI name of the connection factory on the Integration server.

XAI JMS Connection	Description	XAI JNDI Server	JNDI Connection Factory
SIEBEL_CF	Oracle Utilities Customer Care and Billing Siebel Integration Connection Factory	SIEBEL_JNDI	jms/aia/AIAOUCCBCF

Creating XAI Sender

Create a new XAI Sender for the Product queue.

To create a realtime XAI sender configured to communicate with integration layer:

1. In the Admin menu, navigate to XAI Sender.
2. Enter a unique XAI sender and its description.
3. Populate the following values:
    - **XAI Sender –** Sender name in Oracle Utilities Customer Care and Billing.

Copyright © 2012, Oracle. All rights reserved.

- **Description –** Sender description
- **Invocation Type – *Real-time***
- **XAI Class – *RTJMSQSNDR*** (Realtime JMS Queue Sender)
- **Active -** Select the checkbox.
- **MSG Encoding – *UTF-8 message encoding***
- **XAI JMS Connection –** XAI JMS connection created for the integration.
- **XAI JMS Queue –** XAI JMS queue created for the Oracle Utilities Customer Care and Billing outbound queue.

4. Select the **Context** tab and set values for the following context types:

- **JMS Message Type (Bytes(Y)/Text(N))** – ***N***
- **JMS User Name** – user for the SOA server to be accessed.
- **JMS User Password –** Password for the SOA server to be accessed.

Creating an Outbound Message Type

Create a new Outbound Message Type

To create Outbound Message Type:

1. In the Admin menu, navigate to Outbound Message Type.
2. Enter an outbound message type, description, and detailed description.
3. Select the outbound message business object created for a specific outbound queue.

Outbound Message Type	Description	Business Object
SIEBEL_PRODOFFER	Siebel Product Offer Sync Request Outbound Message	C1-ProductOfferSyncReqOutMsg

Creating an External System

You can create a new external system for Oracle Utilities Customer Care and Billing integration with Siebel Energy.

To create a new external system for Oracle Utilities Customer Care and Billing integration to Siebel Energy:

1. In the Admin menu, navigate to External System.
2. Enter a unique external system and description. For example: Name = CI_SEBL1, Description = Siebel Energy
3. Set the Our Name in Their System field to Customer Care and Billing.

Copyright © 2012, Oracle. All rights reserved.

4. Associate the outbound message types created to the external system. For each outbound message type, set the following:
    - **Outbound Message Type** – Set the outbound message type created above.
    - **Processing Method** – ***Real-time***
    - **XAI Sender** – Set the XAI sender created for the queue.
    - **Message XSL** - ***C1-CCBJMSQAddNamespace.xs***

Setting up Siebel Energy

The following sections provide details regarding Siebel Energy configurations required to facilitate the integration:

- Configuring a Business Unit
- Configuring a Workspace
- Configuring a Price List
- Configuring a Service Type

For more information on configuring and working with Siebel Energy, see the *Siebel Energy User Guide and Configuration Guide*.

Configuring a Business Unit

A default business unit needs to be defined in Siebel Energy to be used for all products synchronized through this integration flow. The default business unit will be specified in the AIA Configuration Properties file (discussed in the Setting AIA Configuration Properties section).

To configure the Business Unit:

1. Navigate to Administration-Group->Organization.
2. Create a new Organization giving all the details.
3. Click on Menu->About Record to know the Id for the Business Unit

Configuring a Workspace

A default workspace needs to be defined in Siebel Energy to be used for all products synchronized through this integration flow. The default workspace will be specified in the AIA Configuration Properties file (discussed in the Setting AIA Configuration Properties section).

Copyright © 2012, Oracle. All rights reserved.

To configure the Workspace:

1. Navigate to Administration-Product ->Workspace Projects
2. Create a new Workspace with all the details.

Configuring a Price List

A default price list needs to be defined in Siebel Energy to be used for all products synchronized through this integration flow. The default price list will be specified in the AIA Configuration Properties file (discussed in the Setting AIA Configuration Properties section).

To configure the Price List:

1. Navigate to Administration-Pricing ->Price Lists
2. Create a new Price List giving all the details
3. Save the record
4. Click on Menu > About Record to know the ID for the price list

Configuring a Price Type

Products in Siebel support multiple price types. Configure a Price Type by performing the following to facilitate the integration process.

To configure Price Type

1. Launch Siebel Tools
2. Go to Screens-> List of Values from Menu
3. Query for “UNIT_OF_MEASURE” in Type Column and “Price Type” in Parent LIC Column
4. Add a new record using “UNIT_OF_MEASURE” as Type , <Price Type value> in Display Name Column and set Parent LIC to “Price Type”

Configuring a Service Type

Configure a Service Type to facilitate integration.

To configure Service Type:

1. Launch Siebel Tools
2. Go to Screens-> List of Values from Menu.
3. Query for “UT_SERVICE_TYPE” in Type Column

Copyright © 2012, Oracle. All rights reserved.

4. Add a new record using “UT_SERVICE_TYPE” as Type and <Service Type value> in Display Name Column

Configuring a Billing Type

Configure a Billing Type to facilitate integration.

To configure Billing Type

1. Launch Siebel Tools
2. Go to Screens-> List of Values from Menu.
3. Query for “PROD_BILLING_TYPE” in Type Column
4. Add a new record using “PROD_BILLING_TYPE” as Type and <Billing Type value> in Display Name Column

Setting up the Product Data Synchronization Integration Pack

The following needs to be setup to configure Product Data Synchronization Integration Pack for Oracle Utilities Customer Care and Billing and Siebel Energy Integration:

- AIAConfigurationProperties.xml
- Domain Value Maps
- Error Handling

Setting AIA Configuration Properties

This section lists the configurations in this file that are used by the Product Data Synchronization Integration Pack. Various configurations that apply to the entire Oracle Application Integration Architecture (AIA) system, Core Infrastructure Components, and specific process integration services are stored in the AIAConfigurationProperties.xml file located in <AIA.HOME>/aia_instance/<INSTANCE_NAME>/AIAMetadata/config/ and the AIAConfigurationProperties.xml is stored in Metadata Service (MDS).

These configurations hold several configurable values that are picked up by the integration at runtime to:

- Default some business entity value sent to the target edge applications when the data is moved from one system to another. These configuration values may need to be updated to your implementation specific values for the integration to work correctly. These properties are described in this section in detail.
- Activate custom implemented extension points available inside the ABCS. By default these properties are set not to invoke any of the extension points as the extension points need to be implemented using the AIA ABCS Extension guidelines before being activated. These properties need to be changed only if you decide to extend the core functionality provided by

Copyright © 2012, Oracle. All rights reserved.

this Product Data Synchronization Integration Pack, using the predefined ABCS extension points. These properties are not listed in this section but can be identified in the AIAConfigurationProperties.xml file as the name of these Service Configuration properties start with ABCSExtension and are available in the file under each ABCS Name.

For more information on extending the ABCS using extension points, see Extensibility Options.

- Get the endpoint URLs for Provider ABCS. These properties are set to appropriate values during the installation of Product Data Synchronization Integration Pack, based on the information specified during the installation. You should not have to modify the values.

For more information about requirements for working with AIAConfigurationProperties.xml, see *Oracle Fusion Middleware Developer's Guide for Oracle Application Integration Architecture Foundation Pack 11g Release 1*, "Building AIA Integration Flows," How to Set Up AIA Workstation."

Settings for System Properties

Two sets of configuration properties are described in this section:

- Module Configurations are the properties that are shared by multiple integration flows within this Product Data Synchronization Integration Pack for Oracle Utilities Customer Care and Billing and Siebel Energy.

Note: The only module configuration properties needed for this integration are those required for AIA Error Handling, CAVS and Shared Services Session Pool Manager. Refer to Appendix C for further details about Session Pool Manager.

- Service Configurations are the properties that are used by a specific ABCS.

Service Configurations

ABCS Name	Property Name	Default / Shipped Value	Description
SyncItemCompositionOUCCBUtilitiesReqABCSImpl	BillingTypeForProductComponent	Event	Default Billing Type used for Product Components
SyncItemCompositionSiebelUtilitiesProvABCSImpl	BusinessUnitId		Siebel Business Unit ID. This value should be determined based on the Siebel Business Unit defined in the prior section.
	Parent.Price.Type	Usage	Price Type for the parent product in Siebel

Copyright © 2012, Oracle. All rights reserved.

ABCS Name	Property Name	Default / Shipped Value	Description
	Component.Price.Type	Usage	Price Type for the product components in Siebel
	Workspace.Name		Work Space in Siebel. This is the Siebel Workspace defined in the prior section.
	Workspace.ReleaseFlag	N	
	Workspace.ReuseFlag	Y	
	Price.List.Name		Price List associated with a parent product. This is the Siebel Price List defined in the prior section.
	Price.List.Id		Price List ID in Siebel. This value should be determined based on the Siebel Price List defined in the prior section.
	Long.Description.Length	250	Maximum length of the Long description value passed from Oracle Utilities Customer Care and Billing to Siebel
SyncItemCompositionResponseOUCCBUtilitiesProvABCSImpl	Generic.MessageCategory	11114	CCB Message Category populated in the Product Response in case of error
	GenericBusinessException.MessageNumber	11001	CCB Message Number populated in the Product Response in case of error

Domain Value Maps

Domain value maps (DVMs) are a standard feature of the Oracle SOA Suite which maps codes and other static values across applications. The DVMs are stored in Metadata Service (MDS). For example, **FOOT** and **FT** or **US** and **USA**.

DVMs are static in nature, though administrators can add additional maps as needed. Transactional business processes never update DVMs—they only read from them. They are stored in XML files and cached in memory at runtime.

To maintain the information within the domain value maps:

1. Open a browser and access the SOA Composer (http://soa-server:soa-port/soa/composer) for your installation
2. On the SOA Composer application open DVM.
3. Search and select the relevant DVM you wish to maintain.

Copyright © 2012, Oracle. All rights reserved.

4. Set up the required values for each integrated application.

The Product Data Synchronization Integration Pack for Oracle Utilities Customer Care and Billing and Siebel Energy includes the following DVMs:

DVM	Description
ITEMCOMPOSITION_ITEM_TYPE_BILLINGTYPECODEPARENT	DVM mapping for billing type code for parent product
ITEM_BILLINGTYPECODE	DVM mapping for billing type code
ITEMCOMPOSITION_ITEM_TYPECODE	DVM mapping for product type
ITEMCOMPOSITION_ITEM_COMPONENTTYPECODE	DVM mapping for component code

For more information about DVMs, see *Oracle Fusion Middleware Developer's Guide for Oracle Application Integration Architecture Foundation Pack*, "Understanding Message Transformation, Enrichment, and Configuration," or "Working with Message Transformations" or *Oracle Fusion Middleware Developer's Guide for Oracle SOA Suite*, "Working with Domain Value Maps. For information on matching values for each DVM, refer to the applicable section in Setting up Oracle Utilities Customer Care and Billing for OU_CCB_01 and Siebel Energy for SEBL_01.

ITEMCOMPOSITION_ITEM_TYPE_BILLINGTYPECODEPARENT

Billing Type Code is required for products in Siebel Energy.

This DVM is used to map Billing Type to the Oracle Utilities Customer Care and Billing (CC&B) Product Type. This value will be populated by the Oracle Utilities Customer Care and Billing Item Composition Requester ABCS in the EBO.

OUCCB_ProductType	OUCCB_BillingType
Must match values for CCB Product Type. Must be unique for each DVM row.	Value of **CCB Billing Type**.

ITEM_BILLINGTYPECODE

This DVM is used to map Billing Type from Oracle Utilities Customer Care and Billing to Siebel.

COMMON	OUCCB_01	SEBL_01
AIA common value: Describes the billing type. Must be unique for each DVM row.	Values of **CCB Billing Type** that were derived using ITEMCOMPOSITION_ITEM_TYPE_BILLINGTYPECODEPARENT DVM above.	Must match values for **Billing Type** in Siebel.

ITEMCOMPOSITION_ITEM_TYPECODE

This DVM is used to map Product Type from Oracle Utilities Customer Care and Billing to Service Type in Siebel.

Copyright © 2012, Oracle. All rights reserved.

COMMON	OUCCB_01	SEBL_01
AIA common value: Describes the Product Type. Must be unique for each DVM row.	Must match values for **Product Offer Task Type**.	Must match values for **Service Type**.

ITEMCOMPOSITION_ITEM_COMPONENTTYPECODE

This DVM is used to determine the Product Component Types that need to be moved from Oracle Utilities Customer Care and Billing to Siebel.

COMMON	OUCCB_01	SEBL_01
AIA common value: Describes the product component type. Must be unique for each DVM row.	Must match values for **Component Type**.	Same values as CC&B Component Types.

Cross-References

Cross-references map and connect the records within the application network, and enable these applications to communicate in the same language. The integration server stores the relationship in a persistent way so that others can refer to it.

These are the cross-references for Product Data Synchronization Integration Pack for Oracle Utilities Customer Care and Billing and Siebel Energy:

Note: During implementation you do not need to do anything with this table. This table is populated by the integration processes as products are created.

XREFTABLENAME	COLUMN NAME	DESCR	USAGE
XREF_UTL_ITEMCOMPOSITION_PRDID	OU_CCB_01	CC&B Product ID	Populated by CC&B Item Composition Requester ABCS
	COMMON		Populated by CC&B Item Composition Requester ABCS
	SEBL_01	Siebel Product ID	Populated by Siebel Item Composition Provider ABCS

For more information about cross-references, see *Oracle Fusion Middleware Developer's Guide for Oracle Application Integration Architecture Foundation Pack,* "Understanding Message Transformation, Enrichment, and Configuration," Creating Cross-References."

Copyright © 2012, Oracle. All rights reserved.

Updating MDS with AIA Meta Data

If new artifacts are created, if extensions or customizations are made on the artifacts, or if DVM changes are made in AIAMetaData, you must upload the artifacts to Oracle Metadata Services (MDS).

Oracle Metadata Services (MDS) repository contains all metadata. The contents under <AIA_HOME>/AIAMetaData are uploaded to SOA-MDS > apps/AIAMetaData. This includes specific schemas, WSDLS, DVMs, Cross Reference metadata, default faultPolicies, AIAConfigurationProperties.xml and AIAEHNotification.xml

For more information about updating MDS, see *Oracle Application Integration Architecture – Foundation Pack: Integration Developers Guide,* "Updating MDS.\

Error Notification Setup

Based on the roles defined for the services, email notifications are sent if a service errors out.

For more information about AIA error handling and error notifications; see the Oracle Application Integration Architecture – *Foundation Pack: Core Infrastructure Components Guide,* "Setting Up Error Handling and Logging, Using Error Notifications."

SOA Infrastructure Common Properties

Payload Validation property defined in SOA Infrastructure should be turned off from the Enterprise Manager.

This property is defined on SOA level under Common Properties and can be overridden for each process.

Ensure that your implementation always has Payload Validation disabled for all integration processes.

Chapter 5: Monitoring, Error Handling, and Troubleshooting

This chapter provides details on monitoring, error handling, and troubleshooting, and describes how to perform:

- Monitoring from Oracle Utilities Customer Care and Billing
- Monitoring from Siebel Energy
- Monitoring from the Integration
- Troubleshooting

Copyright © 2012, Oracle. All rights reserved.

Monitoring from Oracle Utilities Customer Care and Billing

This section describes the following:

- Notifications
- Oracle Utilities Customer Care and Billing Error Logs

Notifications in Oracle Utilities Customer Care and Billing (CC&B)

When the products are published from Oracle Utilities Customer Care and Billing, the Sync Requests are transitioned immediately. If the integration layer is down and the message cannot be published to the queue, then the Sync Request is moved to a 'Pending' state. The 'Sync Request Monitor Process' batch program 'F1-SYNRQ' is configured to run periodically to transition the pending Sync Requests for Product Offer.

Once the message is published to the queue, the Sync Requests is in 'Awaiting Acknowledgment' state.

If a positive response is received from Siebel, then the Sync Request is moved to 'Synchronized' state.

If a negative response is received from Siebel or if a time out occurs, then the Sync Request is moved to 'Error' state.

Implementation can plug an algorithm on 'Error' state to create a To Do.

Note: The XAI inbound service is invoked to process the response message. Regardless of whether To Do was set up or not, the errors are logged in spl-service.log file.

Oracle Utilities Customer Care and Billing Error Logs

The Oracle Utilities Customer Care and Billing (CC&B) error logs consist of:

- Errors related to the online sync request transition are stored in the CCB_ENVIRONMENT_NAME/logs/system folder.

  For example: V231_CCB_PERF_BLD10_LIN_ORA_WLS/logs/system

- Errors related to the batch sync request transition are stored in the $SPLOUTPUT/CCB_ENVIRONMENT_NAME folder.

  For example: /spl/sploutput/V231_CCB_PERF_BLD10_LIN_ORA_WLS

For more information about errors and notifications, see the Oracle Utilities Customer Care and Billing documentation.

Copyright © 2012, Oracle. All rights reserved.

Monitoring from Siebel Energy

You can verify whether a product was received by Siebel Energy.

To Verify Receipt of a Product

1. Navigate to the Administration-Product->Products Screen
2. Query for the Product created, if the query is successful, check if product is created with correct details
3. Query for the Price List to which the product is to be attached
4. Navigate to Price List name
5. Query for the Product in Price List Line Item Applet, if the query is successful with correct details of product, then it means that the product is created successfully.

Monitoring from the Integration

To monitor the integration flow using the Integration, use any of the following:

- Monitoring the composite instances using SOA Enterprise Manager
- Monitoring the WebLogic logs
- Monitoring the queues using WebLogic Console
- Data Purge

Monitoring using SOA Enterprise Manager

To monitor through the SOA Enterprise Manager Console, following actions are performed:

1. The Administrator user receives a notification email for any failure in integration flows. The notification email includes an error text and description and a link to the failed instance of the composite on Oracle Enterprise Manager (EM) console.
2. After logging into the Enterprise Manager Console, the user can see additional details about the instance.
3. The status of the ABCS instance, payload and child processes can be monitored from the Oracle Enterprise Manager Console.

Copyright © 2012, Oracle. All rights reserved.

For more information about error notifications, see *Oracle Fusion Middleware Developer's Guide for Oracle Application Integration Architecture Foundation Pack 11g Release 1,* "Working with Error Handling and Logging," Using Error Notifications.

Monitoring using WebLogic logs

Log in to the machine where SOA server is installed. The SOA logs are stored in: <WebLogic installation folder>/user_projects/domains/<SOA Domain name>/servers/<SOA Server name>/logs

Monitoring the Queues using WebLogic Console

Monitoring the Queues using the WebLogic Console involves the following:

To monitor the queues through the WebLogic Console:

1. Log in to the WebLogic Console, and then go to the Services → Messaging → JMS Modules.
2. All queues used for the integration are available in the JMS Module **AIAJMSModule**.
3. Select the appropriate queue on the list and go to the Monitoring tab. In the Monitor tab, you can check if the message still exists in the queue. This can happen if there are no consumers listening to the queue in which case, the Consumers Current column will be 0.
4. To check the message rolled back to the error queue, select the appropriate error queue on the list and go to the Monitoring tab. In the Monitor tab, you can view the message.

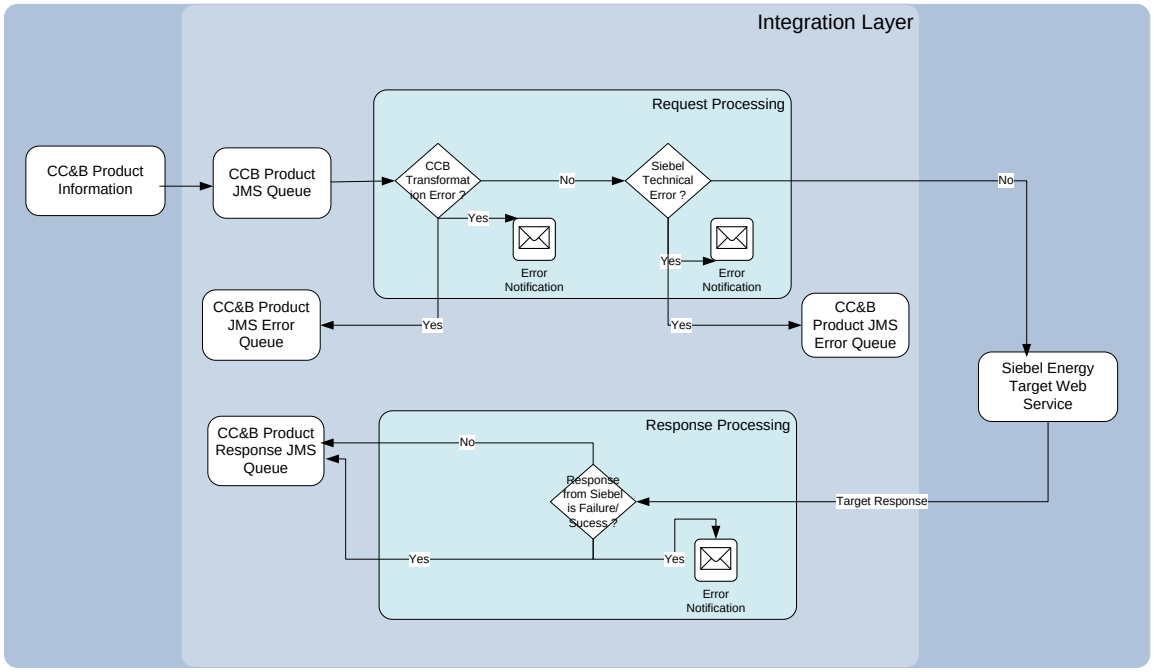
Data Purge

To maintain maximum system integrity, the Oracle Fusion Middleware database should be purged periodically. For information about how to complete this task, refer to the note 815896.1 on https://support.oracle.com.

Troubleshooting

This section explains the actions that can be taken to fix transactions failing in the edge applications. This section explains how the failed messages can be resubmitted, in case if there are any error messages.

Copyright © 2012, Oracle. All rights reserved.



Oracle Utilities Customer Care and Billing –Siebel Energy Integration Layer Flow Diagram

The error scenarios are divided into two categories

Business Errors

These errors occur in the Edge applications or in the Integration layer because of incorrect data setup. For these errors, the message needs to be resubmitted after the data has been corrected.

For example, the default price list in the AIAConfigurationProperties is incorrect and a business error is returned from Siebel.

Technical Errors

These errors are usually connection failures and occur when the Siebel application is unreachable. These messages are moved to the Oracle Utilities Customer Care and Billing (CC&B) Product Error Queue and can be resubmitted from the error queue once the Siebel application is reachable.

Business Errors

Outgoing Messages Resend

Oracle Utilities Customer Care and Billing Outgoing Messages can be resend in case of Sync Request record is in Error status.

- If the error is caused because of missing configuration information, the configuration

Copyright © 2012, Oracle. All rights reserved.

information can be added or revised and the original message can be resend by changing the status of the Sync Request from ‘Error’ to ‘Send Request’. This sends another message.

- If the error requires that something be changed on the Product Offer, the user can make updates to the data on the Product Offer in Oracle Utilities Customer Care and Billing and publish it again.

Examples of case scenarios showing Product Offer Sync Request in the event of an error when attempting to send an outbound message are:

Case 1. Product Offer Sync Request created with no non-final related sync request

1. Publish button is clicked.
2. PO Sync is created in Pending state.
3. Monitor process in Pending state will check if there are other non-final sync requests with the same MO and PK Values as the current one.
4. None found, PO Sync will transition to Capture Final Snapshot, then to Send Request. Any error encountered during this transition will cause the PO Sync to be rolled back. That is, the PO Sync will not persist. The error encountered will be displayed on CC&B's User Interface. If implementation decides to stop all PO Syncs in Pending state, they can plug-in a Monitor Process in this state.

Case 2. Product Offer Sync Request created, non-final related sync exists

1. Publish button is clicked.
2. PO Sync is created ("B") in Pending state.
3. Monitor process in Pending state will check if there are other non-final sync with the same MO and PK Values as "B".
4. Non-final PO Sync ("A") found, new PO Sync "B" will stay in Pending state.
5. Non-final PO sync "A" transitioned to a final state.
6. Run deferred batch to transition PO Sync "B".
7. PO Sync "B" will transition to Capture Final Snapshot, then to Send Request. Any error encountered during this transition will cause PO Sync "B" to rollback to the Pending state.

Technical Errors

This section describes the technical errors and how to handle these error types.

Resubmit from JMS Error Queue

If there is a connection failure while connecting to Siebel, messages are moved to the Oracle Utilities Customer Care and Billing (CC&B) Product Error queue. Log on to the Weblogic Console to monitor the JMS Queues and move messages from the Error Queue to the Main Queue once the reason for failure is resolved.

Copyright © 2012, Oracle. All rights reserved.

Queue		Error Queue	JMS Module
Customer Care and Billing			
Product	AIA_OUCCBProductJMSQueue	AIA_OUCCBProductJMSQueue_ErrorQ	AIAJMSModule
Product Response	AIA_OUCCBProductResponseJMSQueue	AIA_OUCCBProductResponseJMSQueue_ErrorQ	AIAJMSModule

Resubmitting Transactions using Message Resubmission Utility

The Message Resubmission Utility API enables external programs to utilize the functionality of enabling a message that is in error state to resend for a transaction. Typically this utility should be run after the problem is fixed.

For more information about running the Message Resubmission Utility, see *Oracle Fusion Middleware Infrastructure Components and Utilities User's Guide for Oracle Application Integration Architecture Foundation Pack 11g Release 1,* "Using the Message Resubmission Utility."

Error Handling Summary

S.No	Integration Flow	Type of error	Action	Notification Type	Retry
A1	Product published from Oracle Utilities Customer Care and Billing	Integration layer is down and Oracle Utilities Customer Care and Billing cannot publish the message to the queue	Sync Request will be in Pending State.		'F1-SYNRQ' batch program should be scheduled to run periodically to transition the Sync Request.
A2		Internal failure in Requester ABCS	Message goes from the queue to the corresponding error queue. + AIA error handling.	Email notification and worklist entry	After fixing the data, move the messages from the error queue to the main queue.
A3		Requester ABCS cannot reach Item Composition EBS	Message goes from the queue to the corresponding error queue. + AIA error handling.	Email notification and worklist entry	After fixing the data, move the messages from the error queue to the main queue.

Copyright © 2012, Oracle. All rights reserved.

S.No	Integration Flow	Type of error	Action	Notification Type	Retry
A4		Provider ABCS cannot be reached	Message goes from the queue to the error queue. + AIA error handling.	Email notification and worklist entry	After fixing the data, move the messages from the error queue to the main queue.
A5		Technical error in Provider ABCS. Siebel is down	Message goes from the queue to the error queue. + AIA error handling.	Email notification and worklist entry	After Siebel is available, move the messages from the error queue to the main queue.
A6		Business Error in Provider ABCS from Siebel	Negative acknowledgement back to Oracle Utilities Customer Care and Billing. Sync Request will be in 'Error' state.	Email notification and worklist entry. To Do can be configured in Oracle Utilities Customer Care and Billing.	If the Product needs to be updated, publish the product again. Otherwise, update the configuration causing the error and move the Sync Request to 'Send Request' state.
A7		The Item Composition Response EBS cannot be reached.	Message goes from the queue to the error queue. + AIA error handling.	Email notification and worklist entry	After fixing the data, move the messages from the error queue to the main queue.
A8		The Item Composition Response JMS Producer cannot be reached.	Original message moved to the Product error queue. + AIA error handling.	Email notification and worklist entry	After fixing the data, move the messages from the error queue to the main queue.
A9		Messages stuck in Product Response queue	Sync Request will be in 'Awaiting Acknowledgement' state and if timeout occurs, will be moved to 'Error' state.		Verify if MDBs are running in CCB and fix them.

Chapter 6: Extensibility Options

One of the key principles for the design of Application Integration Architecture (AIA) is its extensibility model. AIA offers extensibility in different shapes.

EBOs - The design of the Enterprise Business Objects (EBOs) includes mechanisms to extend generic objects in an upgrade-safe manner by providing hooks to plug in additional industry-specific or customer specific information.

Copyright © 2012, Oracle. All rights reserved.

XSLs/ABCS - The transformations (XSLs) provided with ABCS have custom transformation templates available to map to/from customer specific elements. The ABCS also provide extension points to extend their own functionality.

This chapter provides guidelines on extending the Product Data Synchronization Integration Pack, and discusses how to:

- Extend an EBO
- Extend ABC Services
- Pass Data without customizing the EBO

Extending EBOs

The EBOs provided with the integration are carefully defined to represent a typical business process; however, these definitions do not contain any attributes that are customer specific. To meet the needs of customers with more tailored processing, the integration pack provides the means to extend EBOs in a nonintrusive way.

For example, if your business process requires more information when creating a work order, you can extend EBOs to include the additional information.

A custom schema definition file (XSD) is provided for every EBO that is shipped with the integration. The EBO can be modified by changing these custom files. The XSDs are incorporated into the EBOs so that any custom extensions that are added to the EBOs are protected when patches or upgrades are applied.

EBO Customization

The following sections describe a sample scenario of when and how to customize EBOs. To extend EBOs, add custom elements to the EBO then complete custom mapping for the requester and for the provider.

To add custom elements to the EBO

1. From the Item Composition EBO, identify which section or EBO Type needs to be extended.
2. Add the custom elements to the correct custom EBO Type section in the custom Item Composition xsd file

```
<?xml version='1.0' encoding='UTF-8'?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns="http://xmlns.oracle.com/EnterpriseObjects/Core/Custom/EBO/ItemComposition/V1" xmlns:corecom="http://xmlns.oracle.com/EnterpriseOb
<xsd:import namespace="http://xmlns.oracle.com/EnterpriseObjects/Core/Common/V2" schemaLocation="../../../../Common/V2/CommonComponents.xsd"/>
<xsd:import namespace="http://xmlns.oracle.com/EnterpriseObjects/Core/Common/V2" schemaLocation="../../../../Common/V2/CodeLists.xsd"/>
<xsd:import namespace="http://xmlns.oracle.com/EnterpriseObjects/Core/Common/V2" schemaLocation="../../../../Common/V2/DataTypes.xsd"/>
<xsd:import namespace="http://xmlns.oracle.com/EnterpriseObjects/Core/Common/V2" schemaLocation="../../../../Common/V2/Meta.xsd"/>
<xsd:import namespace="http://xmlns.oracle.com/EnterpriseObjects/Core/Custom/Common/V2" schemaLocation="../../../Common/V2/CustomCommonComponents.xsd"/>
<xsd:complexType name="CustomItemCompositionEBOType">
<xsd:sequence>
<xsd:element name="itemcompEboCustomId1" type="xsd:string"/>
</xsd:sequence>
</xsd:complexType>
</xsd:schema>
```

Copyright © 2012, Oracle. All rights reserved.

For more information about Extending EBOs, see *Oracle Fusion Middleware Developer's Guide for Oracle Application Integration Architecture Foundation Pack,* "Extensibility for Oracle AIA Artifacts – Extending EBOs".

To map the EBO to the requester

Continuing the example from the previous section:

1. Navigate to the Oracle Utilities Customer Care and Billing Item Composition Requester process BPEL composite SyncItemCompositionOUCCBUtilitiesReqABCSImpl and open the custom ABM to EBM transformation. (For example, Xform_ItemCompositionABMReqMsg_to_ItemCompositionEBMReqMsg_Custom.xsl)
2. Map the elements from Oracle Utilities Customer Care and Billing to the custom EBO elements
3. Ensure that the elements are added to the correct section or extension type that needs to be extended

To map the EBO to the provider

1. Navigate to the Siebel Utilities Item Composition Provider BPEL composite **v** and open the custom EBM to ABM transformation under xsl folder. (For example, Xform_ItemCompositionEBMRequestMsg_to_ItemCompositionABMRequestMsg_Custom.xsl)
2. Map the elements from the custom EBO elements to the ABM

Copyright © 2012, Oracle. All rights reserved.

3. Make sure the elements are added to the correct section or extension type that needs to be extended.
   Sample code:

```
<xsl:stylesheet version="1.0"
                xmlns:svcdoc="http://xmlns.oracle.com/Services/Documentation/V1"
                xmlns:xp20="http://www.oracle.com/XSL/Transform/java/oracle.tip.pc.services.functions.Xpath20"
                xmlns:bpws="http://schemas.xmlsoap.org/ws/2003/03/business-process/"
                xmlns:bpel="http://docs.oasis-open.org/wsbpel/2.0/process/executable"
                xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
                xmlns:corecomEBO="http://xmlns.oracle.com/EnterpriseObjects/Core/CommonEBO/V1"
                xmlns:bpm="http://xmlns.oracle.com/bpm20/extensions"
                xmlns:tns="http://siebel.com/asi/V0"
                xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
                xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
                xmlns:coreitemcompositioncust="http://xmlns.oracle.com/EnterpriseObjects/Core/Custom/EBO/ItemComposition/V1"
                xmlns:dvm="http://www.oracle.com/XSL/Transform/java/oracle.tip.dvm.LookupValue"
                xmlns:hwf="http://xmlns.oracle.com/bpel/workflow/xpath"
                xmlns:med="http://schemas.oracle.com/mediator/xpath"
                xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
                xmlns:ids="http://xmlns.oracle.com/bpel/services/IdentityService/xpath"
                xmlns:xacml="urn:oasis:names:tc:xacml:2.0:policy:schema:cd:04"
                xmlns:xdk="http://schemas.oracle.com/bpel/extension/xpath/function/xdk"
                xmlns:ns0="http://xmlns.oracle.com/EnterpriseObjects/Core/EBO/ItemComposition/V1"
                xmlns:xref="http://www.oracle.com/XSL/Transform/java/oracle.tip.xref.xpath.XRefXPathFunctions"
                xmlns:xsd="http://www.w3.org/2001/XMLSchema"
                xmlns:corecom="http://xmlns.oracle.com/EnterpriseObjects/Core/Common/V2"
                xmlns:xacml-context="urn:oasis:names:tc:xacml:2.0:context:schema:cd:04"
                xmlns:nsl="http://xmlns.oracle.com/EnterpriseObjects/Core/Custom/CommonEBO/V1"
                xmlns:ldap="http://schemas.oracle.com/xpath/extension/ldap"
                exclude-result-prefixes="xsi xsl svcdoc corecomEBO corecomcust wsa coreitemcompositioncust xacml ns0 xsd corecom xacml-context nsl tns soapenc soap
   <xsl:template name="ParentProductDefinition_ext">
   <!-- Customers add tranformations here -->
   <tns:FieldReplaceable>
     <xsl:value-of select="ns0:custom/coreitemcompositioncust:itemcompEboCustomIdl"/>
   </tns:FieldReplaceable>
 </xsl:template>
 <xsl:template name="PriceListItemVbc_ext">
   <!-- Customers add tranformations here -->
 </xsl:template>
 <xsl:template name="ComponentDefinition_ext">
   <!-- Customers add tranformations here -->
```

4. Copy the modified EBO xsd file(s) to the correct location on the application server. Ex:<AIA_HOME>/AIAMetaData/AIAComponents/EnterpriseObjectLibrary/<Path_to_xsd>
5. Update the modified xsd in the above step to the MDS Repository for the changes to take effect.
6. Follow the steps below to deploy the modified Requester and Provider BPEL processes.
7. Run the <AIA_HOME>/aia_instances/<instance_name>/bin aiaenv.bat or aiaenv.sh depending on the operating system. This sets the environment variables required by the build process.
8. Back up AIA_HOME>/aia_instances/<INSTANCE_NAME>/AIAMetaData/config/AIAConfigurationProperties.xml. The properties for the Requesters and Providers to be deployed get overwritten and any user defined properties need to be set again after the processes are deployed. Write your own customDP files for updating the metadata and deploying the processes, which are changed as part of customization. Sample code:

```
<DeploymentPlan component="Product" version="1.0"> <Configurations>
   <UpdateMetadata wlserver="pips.OUCCBToSiebelEnergyProduct">
```

Copyright © 2012, Oracle. All rights reserved.

```
<fileset dir="${AIA_HOME}/AIAMetaData">
                    <include
name="<Path_of_file>/<file_to_getUpdated>" />
</fileset>
      </UpdateMetadata>
 </Configurations>
  </DeploymentPlan>
```

The custom DP script for deployment is as follows. The one, which is changed, should be deployed and the sample script for the same will be as follows.

```
<DeploymentPlan component="Product" version="1.0">
<Deployments>
      <Composite compositeName="<processname>"
      compositedir="${AIA_HOME}/services/industry/Utilities/OUCCB/Reque
      storABCS/<processname>" revision="1.0"
      wlserver="pips.OUCCBToSiebelEnergyProduct" action="deploy" />
  </Deployments>
</DeploymentPlan>
```

10. Once the custom scripts are ready the we can execute the custom scripts using the following command

    ant -f $AIA_HOME/Infrastructure/Install/AID/AIAInstallDriver.xml -DDeploymentPlan=$AIA_HOME/pips/OUCCBToSiebelEnergyProduct/DeploymentPlans/CustomDP.xml -DPropertiesFile=$AIA_HOME/aia_instances/<instance_name>/config/AIAInstallProperties.xml -l $AIA_HOME/pips/OUCCBToSiebelEnergyProduct/DeploymentPlans/CustomDeploy.log

11. Test to make sure the mapping is correct and the information is passed from one application to the other.

Extending ABC Services

All Requester and Provider ABC Services provided with this integration pack have extension points available to extend the functionality. For example extension points can be used for additional data enrichment, if required.

There exist two or more extension points in each ABCS depending on the number of services it invokes and the message exchange pattern for those services.

Customers can develop add-ins and have them hooked to these extensibility points. These customer-developed services behave as an extension to the delivered ABC service.

For more information about Extending ABCS, see *Oracle Fusion Middleware Developer's Guide for Oracle Application Integration Architecture Foundation Pack,* "Extensibility for Oracle AIA Artifacts – Extending ABC Services".

Copyright © 2012, Oracle. All rights reserved.

Passing Data without Extending the EBO

In some cases it is necessary to pass a number of fields from one application to another without using the integration layer transformation. If the information does not need to be translated using DVMs, it can simply pass through the integration layer.

In this case, Oracle Utilities Customer Care and Billing uses a custom BO to pass additional information for Product Offer.

For more information on Custom Extensions, refer *to Oracle Utilities Customer Care and Billing Implementation guide*.

To Pass Data without Customizing EBO

1. Configure the source application to pass the new custom element in escaped xml format. This can be done using XSL on the outbound message in the source application.Configure a custom xsl in the Requester ABCS to map the source data to a custom element in EBM.
2. Configure a custom xsl in the Provider ABCS to map the custom element in EBM to the Siebel service ABM.
3. The target application is now able to receive the escaped xml in a custom element and process it.

Chapter 7: Security

This chapter describes the different security policies used for different services.

The Oracle Utilities Product Data Synchronization Integration Pack is enhanced with Oracle Web Services Manager (OWSM) to provide the following security enhancements:

- Attach security policies to services
- Pass username/password information through csf-keys
- Define and store declarative policies from a central location
- Enforce security and management policies through configurable agents

Applied Security Policies

This section identifies the security policies used for various services.

Copyright © 2012, Oracle. All rights reserved.

Global Service Policy

By default all AIA Services - Application Business Connector Services (ABCS), Enterprise Business Services (EBS) and Transport Adapter Services are secured with the following Global Service Policy:

oracle/aia_wss_saml_or_username_token_service_policy_OPT_ON

Applications invoking secured AIA Web Services need to send credentials.

Global Client Policy

Inter-AIA communication is handled by Global Client Policy. By default all Client services are secured with the Global Client Policy:

oracle/aia_wss_saml_or_username_token_client_policy_OPT_ON

Local Client No Authentication Policy

The Siebel Item Composition Provider ABCS that invokes Siebel web service has overridden the above mentioned Global Client policy with a local security policy:

oracle/no_authentication_client_policy

SOA Composite	Attached To
SyncItemCompositionSiebelUtilitiesProvABCSImpl	SWIProductImport

By default, the following Extension web services have been overridden with a local security policy:

oracle/no_authentication_client_policy

If authentication is required by implementation then it will have to be overridden with specific authentication policies required.

SOA Composite	Attached To
SyncItemCompositionOUCCBUtilitiesReqABCSImpl	SyncItemCompositionOUCCBUtilitiesReqABCSImplV1ExtensionService
SyncItemCompositionResponseOUCCBUtilitiesProvABCSImpl	SyncItemCompositionResponseOUCCBUtilitiesProvABCSImplV1ExtensionService
SyncItemCompositionSiebelUtilitiesProvABCSImpl	SyncItemCompositionSiebelUtilitiesProvABCSImplV1ExtensionService

Copyright © 2012, Oracle. All rights reserved.

For more information about security validation, see *Oracle Fusion Middleware Developer's Guide for Oracle Application Integration Architecture Foundation Pack 11g Release 1 (11.1.1.5.0)*, "Working with Security" and *Oracle Fusion Middleware Installation Guide for Oracle Application Integration Architecture Foundation Pack 11*g *Release 1 (11.1.1.5.0).*

Appendix A: Data Mapping

Data Mapping components have been explained in this table:

CC&B Field	EBM Field	Siebel Field	Comments	DVM
syncRequestId	SyncItemCompositionListEBM/DataArea/SyncItemCompositionList/Item/ItemIdentification/AlternateObjectKey/ID		Used to identify the request ID from Oracle Utilities Customer Care and Billing. A response will be sent back to CCB for this Sync Request Id.	
serviceTaskType	SyncItemCompositionListEBM/DataArea/SyncItemCompositionList/Item/ItemIdentification/BusinessComponentID	SWIProductImportUpsert_Input/ListOfProductImport/ProductIntegration/ListOfProductDefinition/ProductDefinition/CommonId	UTILITIES_ITEMCOMPOSITION_PRODUCTID.xref	
	SyncItemCompositionListEBM/DataArea/SyncItemCompositionList/Item/ItemIdentification/ApplicationObjectKey/Id	SWIProductImportUpsert_Input/ListOfProductImport/ProductIntegration/ListOfProductDefinition/ProductDefinition/PartNumber		
description	SyncItemCompositionListEBM/DataArea/SyncItemCompositionList/Item/Name	SWIProductImportUpsert_Input/ListOfProductImport/ProductIntegration/ListOfProductDefinition/ProductDefinition/ProductName		
longDescription	SyncItemCompositionListEBM/DataArea/SyncItemCompositionList/Item/Description	SWIProductImportUpsert_Input/ListOfProductImport/ProductIntegration/ListOfProductDefinition/ProductDefinition//Description	255 chars in Siebel. Truncated depending on the length in AIA Configuration Properties.	

Copyright © 2012, Oracle. All rights reserved.

CC&B Field	EBM Field	Siebel Field	Comments	DVM
productDetails/startDate	SyncItemCompositionListEBM/DataArea/SyncItemCompositionList/Item/ItemManufacturer/EffectiveTimePeriod/StartDateTime	SWIProductImportUpsert_Input/ListOfProductImport/ProductIntegration/ListOfProductDefinition/ProductDefinition/EffectiveStartTime	This is just Date not DateTime in Siebel	
endDate	SyncItemCompositionListEBM/DataArea/SyncItemCompositionList/Item/ItemManufacturer/EffectiveTimePeriod/EndDateTime	SWIProductImportUpsert_Input/ListOfProductImport/ProductIntegration/ListOfProductDefinition/ProductDefinition/EffectiveEndTime		
productType	SyncItemCompositionListEBM/DataArea/SyncItemCompositionList/Item/ItemClassification/ClassificationCode[@listID = 'ServiceType']	SWIProductImportUpsert_Input/ListOfProductImport/ProductIntegration/ListOfProductDefinition/ProductDefinition/ServiceType	A DVM ItemComposition_Item_TypeCode is used to map the Product Offer Type from Oracle Utilities Customer Care and Billing to Siebel. Since Oracle Utilities Customer Care and Billing does not send the Billing Type information, the Billing type is derived from the product offer type using the ItemComposition_Item_Type_BillingTypeCode DVM .	ITEMCOMPOSITION_ITEM_TYPECODE.dvm , ITEMCOMPOSITION_ITEM_TYPE_BILLINGTYPECODEPARENT.dvm mapping product type of CC&B to billing type of CC&B
		SWIProductImportUpsert_Input/ListOfProductImport/ProductIntegration/ListOfProductDefinition/ProductDefinition/Orderable Flag	'Y' for Parent product	
	SyncItemCompositionListEBM/DataArea/SyncItemCompositionList/Item/ItemServiceCharacteristics/ServiceBillingEnabledIndicator	SWIProductImportUpsert_Input/ListOfProductImport/ProductIntegration/ListOfProductDefinition/ProductDefinition/BillableFlag	Used internally in Siebel Energy. Default="Y"	

Copyright © 2012, Oracle. All rights reserved.

CC&B Field	EBM Field	Siebel Field	Comments	DVM
	SyncItemCompositionListEBM/DataArea/SyncItemCompositionList/Item/ItemServiceCharacteristics/TrackInstanceIndicator	SWIProductImportUpsert_Input/ListOfProductImport/ProductIntegration/ListOfProductDefinition/ProductDefinition/TrackAsAssetFlag	Used internally in Siebel Energy. Default="Y"	
	SyncItemCompositionListEBM/DataArea/SyncItemCompositionList/Item/ItemClassification/ClassificationCode[@listID = 'Type']	SWIProductImportUpsert_Input/ListOfProductImport/ProductIntegration/ListOfProductDefinition/ProductDefinition/BillingType	DVM used to map to Billing Type Code. Note: The Billing Type of CC&B Product is derived from the product type using the DVM ItemComposition_Item_Type_BillingTypeCode	ITEM_BILLINGTYPECODE.dvm
		SWIProductImportUpsert_Input/ListOfProductImport/ProductIntegration/ListOfProductDefinition/ProductDefinition/BusinessUnitId	Default on Siebel Provider from AIA Configuration Properties to One Business Unit	
		SWIProductImportUpsert_Input/ListOfProductImport/ProductIntegration/ListOfProductDefinition/ProductDefinition/PriceType	Used internally in Siebel Energy. Default="Usage" Use AIA Configuration Properties to default to 'Usage'. Default on Provider side	
productComponentType	SyncItemCompositionListEBM/DataArea/SyncItemCompositionList/Item/ItemClassification/ClassificationCode[@listID = 'ComponentType']		Only map components to Siebel where ComponentType passed from CC&B has been defined in the DVM. Do filtering on the Provider side and not on the Requestor side.	ITEMCOMPOSITION_ITEM_COMPONENTTYPECODE.dvm

Copyright © 2012, Oracle. All rights reserved.

CC&B Field	EBM Field	Siebel Field	Comments	DVM
productComponent	SyncItemCompositionListEBM/DataArea/SyncItemCompositionList/Item/ItemIdentification/BusinessComponentID	SWIProductImportUpsert_Input/ListOfProductImport/ProductIntegration/ListOfProductDefinition/ProductDefinition/CommonId	Common ID not persisted in Siebel.	
productComponent	SyncItemCompositionListEBM/DataArea/SyncItemCompositionList/Item/Name	SWIProductImportUpsert_Input/ListOfProductImport/ProductIntegration/ListOfProductDefinition/ProductDefinition/ProductName		
	If SyncItemCompositionListEBM/DataArea/SyncItemCompositionList/Item/ItemClassification/ClassificationCode[@listID = 'Type'] = 'EVENT'"> OrderableFlag = 'N' else OrderableFlag = 'Y'	SWIProductImportUpsert_Input/ListOfProductImport/ProductIntegration/ListOfProductDefinition/ProductDefinition/Orderable Flag	'N' for Child Component.	
	SyncItemCompositionListEBM/DataArea/SyncItemCompositionList/Item/ItemServiceCharacteristics/ServiceBillingEnabledIndicator	SWIProductImportUpsert_Input/ListOfProductImport/ProductIntegration/ListOfProductDefinition/ProductDefinition/BillableFlag	Used internally in Siebel Energy. Default="Y"	
	SyncItemCompositionListEBM/DataArea/SyncItemCompositionList/Item/ItemServiceCharacteristics/TrackInstanceIndicator	SWIProductImportUpsert_Input/ListOfProductImport/ProductIntegration/ListOfProductDefinition/ProductDefinition/TrackAsAssetFlag	Used internally in Siebel Energy. Default="Y"	
	SyncItemCompositionListEBM/DataArea/SyncItemCompositionList/Item/ItemClassification/ClassificationCode[@listID = 'Type']	SWIProductImportUpsert_Input/ListOfProductImport/ProductIntegration/ListOfProductDefinition/ProductDefinition/BillingType	AIAConfigProperty	
		SWIProductImportUpsert_Input/ListOfProductImport/ProductIntegration/ListOfProductDefinition/ProductDefinition/BusinessUnitId	Defaulted from AIAConfigProperty on Provider side	
		SWIProductImportUpsert_Input/ListOfProductImport/ProductIntegration/ListOfProductDefinition/ProductDefinition/PriceType	Used internally in Siebel Energy. Default="Usage" Use AIA Configuration Properties to default to 'Usage'. Default on Provider side	

Copyright © 2012, Oracle. All rights reserved.

CC&B Field	EBM Field	Siebel Field	Comments	DVM
		SWIProductImportUpsert_Input/ListOfProductImport/ProductIntegration/ListOfProductDefinition/ProductDefinition/ListOfProductStructure/ProductStructure/ComponentProductName		
		SWIProductImportUpsert_Input/ListOfProductImport/ProductIntegration/ListOfProductDefinition/ProductDefinition/ListOfProductStructure/ProductStructure/ProductName		
		SWIProductImportUpsert_Input/ListOfProductImport/ProductIntegration/ListOfProductStructure/ProductStructure/ProductId	IDs not available hence cannot be mapped to Siebel on Create. On update IDs are available and can be used.	
		SWIProductImportUpsert_Input/ListOfProductImport/ProductIntegration/ListOfProductStructure/ProductStructure/ComponentProductId	IDs not available hence cannot be mapped to Siebel on Create. On update IDs are available and can be used.	
		SWIProductImportUpsert_Input/ListOfProductImport/ProductIntegration/ListOfProductDefinition/ProductDefinition/ListOfSwiPriceListItemVbc/PriceListItemVbc/PriceListId	Only create Price List for the Parent. Defaulted from AIA Configuration Properties	
	Defaulted form AIAConfigurationProperties.	SWIProductImportUpsert_Input/ListOfProductImport/ProductIntegration/ListOfProductDefinition/ProductDefinition/ListOfSwiPriceListItemVbc/PriceListItemVbc/PriceListName	Defaulted from AIA Configuration Properties	

Copyright © 2012, Oracle. All rights reserved.

CC&B Field	EBM Field	Siebel Field	Comments	DVM
	SyncItemCompositionListEBM/DataArea/SyncItemCompositionList/Item/Name	SWIProductImportUpsert_Input/ListOfProductImport/ProductIntegration/ListOfProductDefinition/ProductDefinition/ListOfSwiPriceListItemVbc/PriceListItemVbc/ProductName		
	Defaulted to Current Date	SWIProductImportUpsert_Input/ListOfProductImport/ProductIntegration/ListOfProductDefinition/ProductDefinition/ListOfSwiPriceListItemVbc/PriceListItemVbc/PriceStartDate		
		SWIProductImportUpsert_Input/ListOfProductImport/ProductIntegration/WorkspaceName	AIAConfiguration Property	
		SWIProductImportUpsert_Input/ListOfProductImport/ProductIntegration/WorkspaceReleaseFlag	AIAConfiguration Property	
		SWIProductImportUpsert_Input/ListOfProductImport/ProductIntegration/WorkspaceReuseFlag	AIAConfiguration Property	

Appendix B: Cross-References

The following sections provide references for where you can find more information on some of the terms and entities related to this integration.

ABCs

For more information, see *Oracle Fusion Middleware Developer's Guide for Oracle Application Integration Architecture Foundation Pack 11g Release 1*, "Designing and Constructing ABC Services" and *Oracle Fusion Middleware Concepts and Technologies Guide for Oracle Application Integration Architecture Foundation Pack 11g Release 1*, "Understanding ABC Services".

AggregatorAdapters

For more information, see Oracle Fusion Middleware Developer's Guide for Oracle Application Integration Architecture Foundation Pack 11g Release 1, "Describing the Event Aggregation Programming Model".

Copyright © 2012, Oracle. All rights reserved.

Creating Cross-References

For more information, see Oracle Fusion Middleware Concepts and Technologies Guide for Oracle Application Integration Architecture Foundation Pack 11g Release 1, "Understanding Message Transformation, Enrichment, and Configuration," Cross-References.

DVMs

For more information on domain value maps shipped with this product, see Working with Domain Value Maps.

For more information, see *Oracle Fusion Middleware Developer's Guide for Oracle Application Integration Architecture Foundation Pack*, "Understanding Message Transformation, Enrichment, and Configuration," Domain Value Maps.

EBFs

For more information, see *Oracle Fusion Middleware Developer's Guide for Oracle Application Integration Architecture Foundation Pack*, "Designing and Constructing EBFs" and *Oracle Fusion Middleware Concepts and Technologies Guide for Oracle Application Integration Architecture Foundation Pack*, "Understanding EBSs," Enterprise Business Flow Processes.

EBOs

For detailed documentation of individual EBOs and EBMs, click the AIA Reference Doc link on EBO and EBM detail pages in Oracle Enterprise Repository.

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 about using the Oracle Enterprise Repository and configuring it to provide the AIA Reference Doc link, see *Oracle Fusion Middleware Concepts and Technologies Guide for Oracle Application Integration Architecture Foundation Pack*, "Configuring and Using Oracle Enterprise Repository as the Oracle AIA SOA Repository.

EBSs

For more information, see *Oracle Fusion Middleware Developer's Guide for Oracle Application Integration Architecture Foundation Pack*, "Designing and Developing EBSs" and Oracle Application Integration Architecture - Foundation Pack: Concepts and Technologies Guide, "Understanding EBSs".

Copyright © 2012, Oracle. All rights reserved.

Error Handling

For more information about the errors thrown by Siebel CRM or Oracle EBS, see the documentation for that product. For more information about AIA error handling, see the *Oracle Fusion Middleware Infrastructure Components and Utilities User's Guide for Oracle Application Integration Architecture Foundation Pack*, "Setting Up and Using Error Handling and Logging."

For more information on the Oracle Worklist Application, see Oracle Application Integration Architecture: Foundation Pack, "Infrastructure Components and Utilities User's Guide for Oracle Application Integration Architecture Foundation Pack, "Using the Error Console."

Error Roles

For more information about setting up error notifications using these values, see Oracle Fusion Middleware Infrastructure Components and Utilities User's Guide for Oracle Application Integration Architecture Foundation Pack, "Setting up Error Notifications and Trace Logging."

JMS Adapters (Producers and Consumers)

For more information, see Oracle Fusion Middleware Developer's Guide for Oracle Application Integration Architecture Foundation Pack, "Designing and Constructing JMS Adapter Services".

Setting Config Properties

For more information, see *Oracle Fusion Middleware Infrastructure Components and Utilities User's Guide for Oracle Application Integration Architecture Foundation Pack*, "Using the BSR," Loading Oracle AIA Configuration File Updates.

Appendix C: Using Session Pool Manager

Session Pool Manager (SPM) 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.

Reusing session tokens significantly enhances the overall performance of BPEL flows that call web services. This is because the session token creation process is a time-consuming operation in the application web server.

Some features of SPM include:

- SPM is automatically initialized upon the request of a session token.
- The session token pool is stored in memory.

Copyright © 2012, Oracle. All rights reserved.

- SPM supports concurrent (multithreaded) BPEL flows
- SPM supports multiple application web server instances (hosts), with one SPM instantiated per host.
- Includes the Sentinel, which periodically checks session tokens, removes expired sessions, and replaces them with new ones.

For more information about Session Pool Manager, see the Oracle Application Integration Architecture - Pre-Built Integrations 11.1: Utilities Guide.

Configuring Session Pool Manager

SPM configuration properties are set in the AIAConfigurationProperties.xml file as a Module Configuration. The module name is SessionPoolManager.

SPM can work with multiple hosts (application web servers). Therefore, each property can be set as a default for all hosts, and overridden for a specific host. The only exception is the TRACE.LOG.ENABLED property, which cannot be set to be server-specific.

Each property has a prefix that indicates the application web server. For example:

```
<Property name="all_hosts.PoolSize_Max">40</Property>
<Property name="SEBL_01.PoolSize_Max">20</Property>
```

The first line defines **40** as the default maximum pool size for all hosts.

The second line overrides the default pool size to **20** for the application web server SEBL_01.

The concept of system ID and Host ID are synonymous.

For example, if you use SEBL_01as the system ID for the Siebel application web server, you can see SEBL_01 in the AIAConfigurationProperties.xml file as the “Default.SystemID" property for the services connecting to the Siebel application web server. This SEBL_01 value should also be used as the HostId value in SPM to refer to the Siebel application web server.

The values of "SEBL_01.EndpointURI" properties for the process, should match with the value of "SEBL_01.EndpointURI" property under the SessionPoolManager module configuration section.

For example:

```
   <Property
name="SEBL_01.EndpointURI">http://[siebel.http.host]:[si
ebel.http.port]/eai_enu/start.swe?SWEExtSource=SecureWebService&amp;
SWEExtCmd=Execute&amp;WSSOAP=1</Property>
```

Note: In cases, where a Siebel instance is restarted, you must terminate the Session Pool Manager for that Siebel instance as its Siebel sessions are no longer valid.

Copyright © 2012, Oracle. All rights reserved.

Appendix D: CAVS Testing

The Composite Application Validation System (CAVS) is a framework that provides an approach to test integration of Oracle Application Integration Architecture services. CAVS includes test initiators that simulate web service invocations and simulators that simulate service endpoints.

The CAVs properties in AIAConfigurationProperties.xml are by default set to not route the messages to CAVS. CAVS related properties need to be changed only if you decide to use CAVS for testing. These properties can be identified in the AIAConfigurationProperties.xml file as the name of these Service Configuration properties ends with either RouteToCAVS or CAVS.EndpointURI, and are available in the file under each ABCS Name.

For more information, about CAVS; see the *Oracle Fusion Middleware Infrastructure Components and Utilities User's Guide for Oracle Application Integration Architecture Foundation Pack 11g Release 1,* “Introduction to the Composite Application Validation System”.

Copyright © 2012, Oracle. All rights reserved.