Oracle® Application Server Adapter

for Siebel User's Guide

10*g* Release 2 (10.1.2)

Part No. B14062-01

November 2004

Oracle Application Server Adapter for Siebel User's Guide, 10*g* Release 2 (10.1.2)

Part No. B14062-01

Primary Author: Matthew Barbarello

Contributing Author: Meera Srinivasan

Contributor: Arvind Jain

Contents

3 Deployment and Integration

4 Examples

5 Troubleshooting and Error Messages

6 Advanced Topics

A Using Siebel Workflows

Send Us Your Comments

Oracle Application Server Adapter for Siebel User's Guide, 10*g* Release 2 (10.1.2)

Part No. B14062-01

Oracle welcomes your comments and suggestions on the quality and usefulness of this publication. Your input is an important part of the information used for revision.

- Did you find any errors?
- Is the information clearly presented?
- Do you need more information? If so, where?
- Are the examples correct? Do you need more examples?
- What features did you like most about this manual?

If you find any errors or have any other suggestions for improvement, please indicate the title and part number of the documentation and the chapter, section, and page number (if available). You can send comments to us in the following ways:

- Electronic mail: appserverdocs_us@oracle.com
- FAX: (650) 506-7375. Attn: Oracle Application Server Documentation Manager
- Postal service:

  Oracle Corporation
  Oracle Application Server Documentation
  500 Oracle Parkway, Mailstop 1op6
  Redwood Shores, CA 94065
  USA

If you would like a reply, please give your name, address, telephone number, and electronic mail address (optional).

If you have problems with the software, please contact your local Oracle Support Services.

Preface

This guide explains how to use the Oracle Application Server Adapter for Siebel (OracleAS Adapter for Siebel) to access Siebel Business Components and Business Services. In this guide you will learn how to define a delivery channel for Siebel and add an interaction to generate native events, which are XML instances defined by XSD (XML payload defined by an XML Schema Definition instance). In this guide you will also find a chapter describing the datatype mapping between Siebel and XSD.

This preface contains these topics:

- Documentation Accessibility
- Intended Audience
- Organization
- Related Documentation
- Conventions

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible, with good usability, to the disabled community. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Standards will continue to evolve over time, and Oracle is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For additional information, visit the Oracle Accessibility Program Web site at

http://www.oracle.com/accessibility/

Accessibility of Code Examples in Documentation JAWS, a Windows screen reader, may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, JAWS may not always read a line of text that consists solely of a bracket or brace.

Accessibility of Links to External Web Sites in Documentation This documentation may contain links to Web sites of other companies or organizations that Oracle does not own or control. Oracle neither evaluates nor makes any representations regarding the accessibility of these Web sites.

Intended Audience

Oracle Application Server Adapter for Siebel User's Guide is intended for those who perform the following tasks:

- Create delivery channels and interactions with a Siebel system
- Maintain applications

To use this document, you need some knowledge of Siebel Business Components and Business Services.

Organization

This document contains:

Chapter 1, "Introduction to Oracle Application Server Adapter for Siebel"

This chapter describes the OracleAS Adapter for Siebel and the hardware and software requirements.

Chapter 2, "Adapter Configuration Using OracleAS Adapter Application Explorer"

This chapter provides instructions for starting Application Explorer, for creating projects, establishing a connection to Siebel, and creating schemas and Web services. It also explains how to configure the Event Adapter.

Chapter 3, "Deployment and Integration"

This chapter describes Oracle Containers for J2EE (OC4J) deployment and integration with OracleAS Integration Interconnect.

Chapter 4, "Examples"

This chapter contains examples.

Chapter 5, "Troubleshooting and Error Messages"

This chapter describes how to troubleshoot and interpret error messages.

Chapter 6, "Advanced Topics"

This chapter includes advanced topics for expert users.

Appendix A, "Using Siebel Workflows"

This appendix includes information on Siebel workflows.

Related Documentation

For more information, see these Oracle resources:

- *Oracle Application Server Adapter Concepts*
- *Oracle Application Server Adapters Installation Guide*
- *Oracle Application Server Administrator's Guide*
- *Oracle Application Server Concepts*
- *Oracle Application Server Containers for J2EE User's Guide*

- *Oracle Application Server Integration InterConnect User's Guide*

Printed documentation is available for sale in the Oracle Store at

```
http://oraclestore.oracle.com
```

To download free release notes, installation documentation, white papers, or other collateral, please visit the Oracle Technology Network (OTN). You must register online before using OTN; registration is free and can be done at

```
http://www.oracle.com/technology/membership
```

If you already have a username and password for OTN, then you can go directly to the documentation section of the OTN Web site at

```
http://www.oracle.com/technology/documentation
```

Conventions

This section describes the conventions used in the text and code examples of this documentation set. It describes:

- Conventions in Text
- Conventions in Code Examples
- Conventions for Windows Operating Systems

Conventions in Text

We use various conventions in text to help you more quickly identify special terms. The following table describes those conventions and provides examples of their use.

Convention	Meaning	Example
Bold	Bold typeface indicates terms that are defined in the text or terms that appear in a glossary, or both.	When you specify this clause, you create an **index-organized table**.
Italics	Italic typeface indicates book titles or emphasis.	*Oracle Database Concepts* Ensure that the recovery catalog and target database do *not* reside on the same disk.
`UPPERCASE monospace (fixed-width) font`	Uppercase monospace typeface indicates elements supplied by the system. Such elements include parameters, privileges, datatypes, RMAN keywords, SQL keywords, SQL*Plus or utility commands, packages and methods, as well as system-supplied column names, database objects and structures, usernames, and roles.	You can specify this clause only for a `NUMBER` column. You can back up the database by using the `BACKUP` command. Query the `TABLE_NAME` column in the `USER_TABLES` data dictionary view. Use the `DBMS_STATS.GENERATE_STATS` procedure.

Convention	Meaning	Example
`lowercase monospace (fixed-width) font`	Lowercase monospace typeface indicates executables, filenames, directory names, and sample user-supplied elements. Such elements include computer and database names, net service names, and connect identifiers, as well as user-supplied database objects and structures, column names, packages and classes, usernames and roles, program units, and parameter values. **Note:** Some programmatic elements use a mixture of UPPERCASE and lowercase. Enter these elements as shown.	Enter `sqlplus` to open SQL*Plus. The password is specified in the `orapwd` file. Back up the datafiles and control files in the `/disk1/oracle/dbs` directory. The `department_id`, `department_name`, and `location_id` columns are in the `hr.departments` table. Set the `QUERY_REWRITE_ENABLED` initialization parameter to `true`. Connect as `oe` user. The `JRepUtil` class implements these methods.
`lowercase italic monospace (fixed-width) font`	Lowercase italic monospace font represents placeholders or variables.	You can specify the *`parallel_clause`*. Run U*`old_release`*`.SQL` where *`old_release`* refers to the release you installed prior to upgrading.

Conventions in Code Examples

Code examples illustrate SQL, PL/SQL, SQL*Plus, or other command-line statements. They are displayed in a monospace (fixed-width) font and separated from normal text as shown in this example:

```
SELECT username FROM dba_users WHERE username = 'MIGRATE';
```

The following table describes typographic conventions used in code examples and provides examples of their use.

Convention	Meaning	Example
[ ]	Brackets enclose one or more optional items. Do not enter the brackets.	`DECIMAL (`*`digits`*` [ , `*`precision`*` ])`
{ }	Braces enclose two or more items, one of which is required. Do not enter the braces.	`{ENABLE | DISABLE}`

|	A vertical bar represents a choice of two or more options within brackets or braces. Enter one of the options. Do not enter the vertical bar.	`{ENABLE | DISABLE}`
		`[COMPRESS | NOCOMPRESS]`

...	Horizontal ellipsis points indicate either:
	■ That we have omitted parts of the code that are not directly related to the example
	■ That you can repeat a portion of the code
		`CREATE TABLE ... AS `*`subquery`*`;`
		`SELECT `*`col1`*`, `*`col2`*`, ... , `*`coln`*` FROM employees;`

Convention	Meaning	Example

.
.
.
	Vertical ellipsis points indicate that we have omitted several lines of code not directly related to the example.	`SQL> SELECT NAME FROM V$DATAFILE;`
		`NAME`
		`------------------------------------`
		`/fsl/dbs/tbs_01.dbf`
		`/fs1/dbs/tbs_02.dbf`
		`.`
		`.`
		`.`
		`/fsl/dbs/tbs_09.dbf`
		`9 rows selected.`

Other notation	You must enter symbols other than brackets, braces, vertical bars, and ellipsis points as shown.	`acctbal NUMBER(11,2);`
		`acct    CONSTANT NUMBER(4) := 3;`

Italics	Italicized text indicates placeholders or variables for which you must supply particular values.	CONNECT SYSTEM/*system_password* DB_NAME = *database_name*

UPPERCASE	Uppercase typeface indicates elements supplied by the system. We show these terms in uppercase in order to distinguish them from terms you define. Unless terms appear in brackets, enter them in the order and with the spelling shown. However, because these terms are not case sensitive, you can enter them in lowercase.	`SELECT last_name, employee_id FROM employees;`
		`SELECT * FROM USER_TABLES;`
		`DROP TABLE hr.employees;`

lowercase	Lowercase typeface indicates programmatic elements that you supply. For example, lowercase indicates names of tables, columns, or files.
	Note: Some programmatic elements use a mixture of UPPERCASE and lowercase. Enter these elements as shown.
		`SELECT last_name, employee_id FROM employees;`
		`sqlplus hr/hr`
		`CREATE USER mjones IDENTIFIED BY ty3MU9;`

Conventions for Windows Operating Systems

The following table describes conventions for Windows operating systems and provides examples of their use.

Convention	Meaning	Example
Choose Start >	How to start a program.	To start the Database Configuration Assistant, choose Start > Programs > Oracle - *HOME_NAME* > Configuration and Migration Tools > Database Configuration Assistant.

File and directory names	File and directory names are not case sensitive. The following special characters are not allowed: left angle bracket (<), right angle bracket (>), colon (:), double quotation marks ("), slash (/), pipe (|), and dash (-). The special character backslash (\) is treated as an element separator, even when it appears in quotes. If the file name begins with \\, then Windows assumes it uses the Universal Naming Convention.	`c:\winnt"\"system32 is the same as`
		`C:\WINNT\SYSTEM32`

Convention	Meaning	Example
`C:\>`	Represents the Windows command prompt of the current hard disk drive. The escape character in a command prompt is the caret (^). Your prompt reflects the subdirectory in which you are working. Referred to as the *command prompt* in this manual.	`C:\oracle\oradata>`

Special characters	The backslash (\) special character is sometimes required as an escape character for the double quotation mark (") special character at the Windows command prompt. Parentheses and the single quotation mark (’) do not require an escape character. Refer to your Windows operating system documentation for more information on escape and special characters.	`C:\>exp scott/tiger TABLES=emp`
		`QUERY=\"WHERE job='SALESMAN' and`
		`sal<1600\"`
		`C:\>imp SYSTEM/`*password* `FROMUSER=scott`
		`TABLES=(emp, dept)`

HOME_NAME	Represents the Oracle home name. The home name can be up to 16 alphanumeric characters. The only special character allowed in the home name is the underscore.	`C:\> net start Oracle`*HOME_NAME*`TNSListener`
ORACLE_HOME and *ORACLE_BASE*	In releases prior to Oracle8i release 8.1.3, when you installed Oracle components, all subdirectories were located under a top level *ORACLE_HOME* directory. For Windows, the default location was `C:\orant`. This release complies with Optimal Flexible Architecture (OFA) guidelines. All subdirectories are not under a top level *ORACLE_HOME* directory. There is a top level directory called *ORACLE_BASE* that by default is `C:\oracle`. If you install the latest Oracle release on a computer with no other Oracle software installed, then the default setting for the first Oracle home directory is *C:\oracle\orann*, where *nn* is the latest release number. The Oracle home directory is located directly under *ORACLE_BASE*. All directory path examples in this guide follow OFA conventions. Refer to *Oracle Database Platform Guide for Windows* for additional information about OFA compliances and for information about installing Oracle products in non-OFA compliant directories.	Go to the `ORACLE_BASE\ORACLE_HOME\rdbms\admin` directory.

1

Introduction to Oracle Application Server Adapter for Siebel

Oracle Application Server connects to a Siebel system through the Oracle Application Server Adapter for Siebel (OracleAS Adapter for Siebel). The OracleAS Adapter for Siebel provides connectivity and executes interactions on a Siebel system.

This chapter discusses the following topics:

- Adapter Features
- The Siebel Application Model
- Integration with Siebel
- Using Application Explorer with the OracleAS Adapter for Siebel

Adapter Features

The **adapter** provides a means to exchange real-time business data between Siebel systems and other application, database, or external business partner systems. The adapter enables external applications for inbound and outbound processing with Siebel.

The OracleAS Adapter for Siebel can be deployed as a JCA 1.0 resource adapter. This deployment is referred to as the OracleAS Adapter JCA. It can also be deployed as a Web services servlet and as such is referred to as the Oracle Application Server Adapter Business Services Engine (BSE).

The adapter uses XML messages to enable non-Siebel applications to communicate and exchange transactions with Siebel using services and events.

- Services: Applications use this capability to initiate a Siebel business event.
- Events: Applications use this capability to access Siebel data only when a Siebel business event occurs.

To support event functionality, two features are implemented:

- Port

  A **port** associates a particular business object exposed by an adapter with a particular disposition. A disposition defines the protocol and location of the event data. The port defines the end point of the event consumption.

  The port is the Oracle adapter component that pushes the event received from the EIS to the adapter client. The only port supported in this release is Remote Method Invocation (RMI). It is used for integration with Oracle Application Server Integration InterConnect (OracleAS Integration InterConnect).

- Channel

  A **channel** represents configured connections to particular instances of back-end or other types of systems. A channel binds one or more event ports to a particular **listener** managed by an adapter.

  The channel is the adapter component that receives events in real time from the Enterprise Information System (EIS) application. The channel component can be a File reader, an HTTP listener, a TCP/IP listener, or an FTP listener. A channel is always EIS specific. The adapter supports multiple channels for a particular EIS. This enables the user to choose the optimal channel component based on deployment requirements.

The OracleAS Adapter for Siebel:

- Supports synchronous and asynchronous, bidirectional message interactions for Siebel Business Services, Business Components, and Integration Objects.
- Includes Oracle Application Server Adapter Application Explorer (Application Explorer), a GUI tool that uses the Siebel Object Manager to explore Siebel metadata and build XML schemas or Web services.
- Supports Siebel transports—MQSeries, File, and HTTP. It also supports MSMQ messaging.
- XML schemas for the OracleAS Adapter JCA.
- Web services for the Business Services Engine.

> **See Also:** *Oracle Application Server Adapter Concepts*

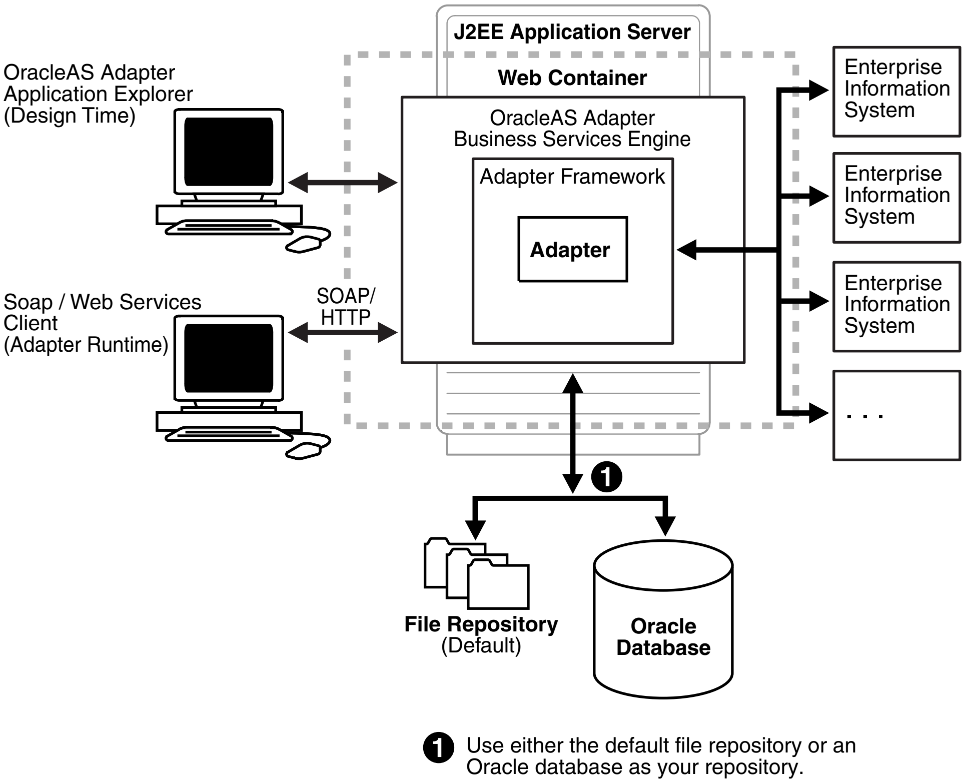
Oracle Application Server Adapter Business Services Engine Architecture

Figure 1–1 shows the generic architecture for the Oracle Web service adapter for packaged applications. The adapter works in conjunction with the Oracle Application Server Adapter Business Services Engine (BSE), as deployed to a Web container in a J2EE application server. BSE serves as host to the adapters, enabling Web service requests to the adapters.

Oracle Application Explorer, a design-time tool deployed along with BSE, is used to configure adapter connections, browse EIS objects, configure services, and configure listeners to listen for EIS events. Metadata created while you perform these operations are stored in the repository by BSE.

BSE uses SOAP as a protocol for consuming requests from clients, interacting with the EIS, and sending responses from the EIS back to clients.

Figure 1–1 Oracle Application Server Adapter Business Services Engine Architecture

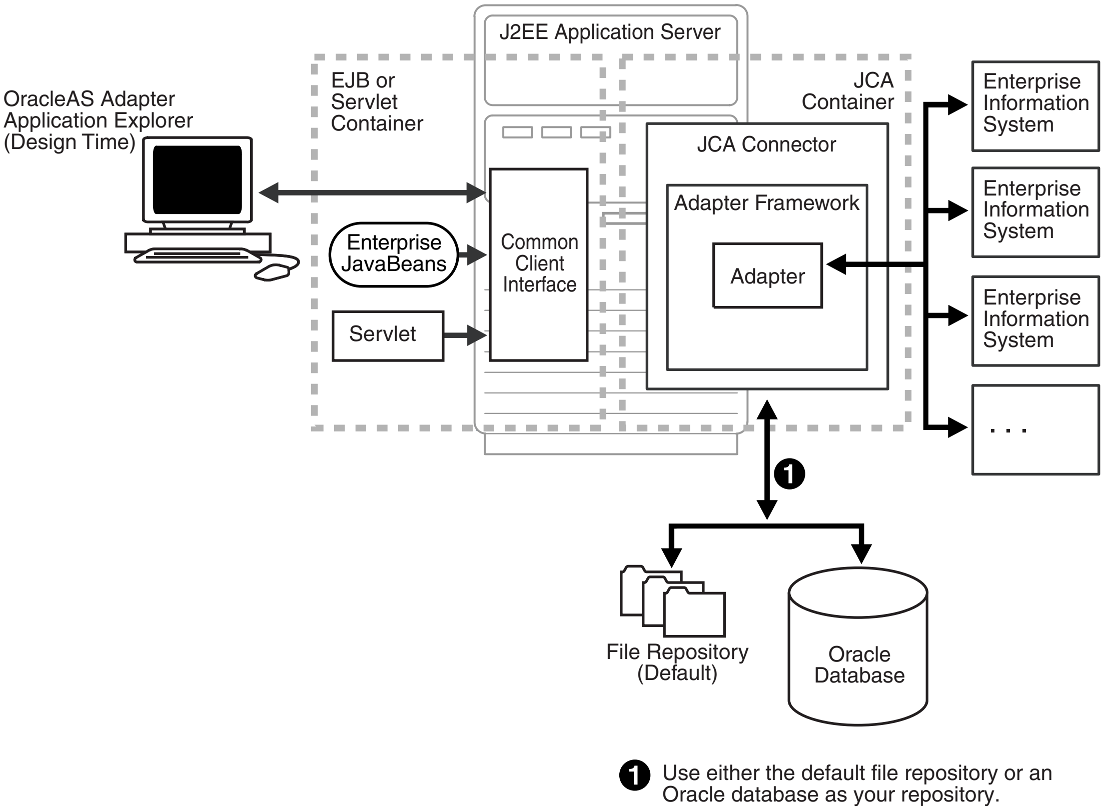

J2EE Application Server
Web Container
OracleAS Adapter Application Explorer (Design Time)
OracleAS Adapter Business Services Engine
Adapter Framework
Adapter
Soap / Web Services Client (Adapter Runtime)
SOAP/ HTTP
Enterprise Information System
Enterprise Information System
Enterprise Information System
. . .
1
File Repository (Default)
Oracle Database
1 Use either the default file repository or an Oracle database as your repository.

Oracle Application Server Adapter Generic JCA Architecture

Figure 1–2 shows the generic architecture for the OracleAS Adapter JCA for packaged applications. The OracleAS Adapter JCA is deployed to a standard JCA container and serves as host container to the adapters. The connector is configured with a repository.

Oracle Application Explorer, a design-tool that works in conjunction with the connector, is used to configure adapter connections, browse EIS objects, configure services, and configure listeners to listen for EIS events. Metadata created while you perform these operations are stored in the repository by the connector.

Figure 1–2 Oracle Application Server Adapter Generic JCA Architecture



1 Use either the default file repository or an Oracle database as your repository.

The Siebel Application Model

The Siebel Enterprise application defines a data abstraction layer that removes dependencies on the underlying database. It accomplishes this by using intermediate Business Components and Business Objects that represent database structures. A Business Component usually represents a table in a database. A Business Object is a group of related business components.

From a given business component, you can navigate the relationships defined for that component to another component. The path you use to traverse component relationships is called the navigation path. For example, if you want to obtain all addresses for a particular account, you can traverse the parent/child relationship between Account and Address to obtain those addresses. By using navigation paths, you can traverse nearly all of the business component relationships defined in the Siebel system.

In Siebel, Integration Objects are similar to Siebel Business Components but describe more complex hierarchal data relationships.

Integration with Siebel

You can use the OracleAS Adapter for Siebel to invoke a Siebel business process, such as add/update account, or you can use the adapter as part of an integration effort to connect Siebel and non-Siebel systems. The OracleAS Adapter for Siebel is bidirectional and can detect an event from Siebel by receiving a Siebel XML document emitted by Siebel.

When integrating with Siebel using Siebel XML documents, the adapter application developer must use existing Siebel Integration Objects or create new Siebel Integration Objects to use within a Siebel Workflow. The Workflow processes inbound or outbound Siebel XML and uses various transports such as MQSeries, File, and HTTP to exchange transactions with external systems. The Siebel Workflow is usually created by the Siebel administrator or developer using Siebel Workflow Administration screens.

When integrating with Siebel directly using the Java Data Bean or COM Data Interface, the OracleAS Adapter for Siebel does not require a Siebel Integration Object or Siebel Workflow. Instead, it executes Siebel Business Services and Siebel Business Components directly.

The following table lists Siebel objects and processes.

Table 1–1 Siebel Objects and Processes

Siebel Objects	API or Transport	Process
Business Services	Java Data Bean (Siebel Version 6.3 or higher) Com Data Interface (Siebel Version 6.2 or lower)	N/A
Business Components	Java Data Bean (Siebel Version 6.3 or higher) Com Data Interface (Siebel Version 6.2 or lower)	N/A
Integration Objects	File	Event, Service
	HTTP	Event, Service
	MQSeries	Event, Service
	MQ Read	Service

Integrating with Siebel EAI Architecture

Siebel provides for integration with other applications and systems using its Siebel EAI (Enterprise Application Integration) framework and its Business Integration Manager facility. The OracleAS Adapter for Siebel uses the Siebel EAI framework and leverages various integration access methods to provide the greatest amount of flexibility and functionality while working within the Siebel framework.

The OracleAS Adapter for Siebel supports the following integration access methods:

- Siebel Java Data Bean for services involving Siebel Business Components or Siebel Business Services.
- Siebel COM Data Interface for services involving Siebel Business Components or Siebel Business Services.
- Siebel XML for events and services involving Siebel Integration Objects.

Using Application Explorer with the OracleAS Adapter for Siebel

The Application Explorer uses an explorer metaphor for browsing the Siebel system for Business Services, Business Objects, Business Components, and Integration Objects. The explorer enables you to create XML schemas and Web services for the associated object. External applications that access Siebel through the OracleAS Adapter for Siebel use either XML schemas or Web services to pass data between the external application and the adapter.

Application Explorer uses interfaces provided by Siebel and in-depth knowledge of the Siebel application systems to access and browse business object metadata. After an object is selected, Application Explorer can generate an XML schema or Web service to define the object for use in conjunction with the OracleAS Adapter for Siebel.

Key features of Application Explorer include:

- The ability to connect to and explore a variety of application systems.
- Access to application system object metadata.
- A point-and-click process for generating XML schemas and Web services.

See Also:

- *Oracle Application Server Adapter Concepts*
- *Oracle Application Server Adapters Installation Guide*

2

Adapter Configuration Using OracleAS Adapter Application Explorer

This chapter provides the information you require to create schemas for Siebel Business Objects.

This chapter discusses the following topics:

- Starting Application Explorer
- Configuring Settings for BSE or JCA
- Creating a Repository Project
- Establishing a Connection (Target) for Siebel
- Viewing Application System Objects
- Creating an XML Schema
- Creating a Web Service or Business Service
- Configuring an Event Adapter

Starting Application Explorer

Prerequisites

Before starting Application Explorer and using the OracleAS Adapter for Siebel, you must create \endorsed directories under your *OracleAS_home* directory and place a copy of the `xalan.jar` file in those directories.

1. Navigate to the *OracleAS_home*`\j2ee\home\connectors\jca-app-adapter\jca-app-adapter` directory and copy the `xalan.jar` file.
2. Create an \endorsed subdirectory in the following two directories:
    - *OracleAS_home*`\jre\1.4.2\lib\`
    - *OracleAS_home*`\jdk\jre\1.4.2\lib\`
3. Place a copy of the `xalan.jar` file into the new subdirectories.

    When you are done, ensure that you have the `xalan.jar` file in the following two directories:

    - *OracleAS_home*`\jre\1.4.2\lib\endorsed`
    - *OracleAS_home*`\jdk\jre\1.4.2\lib\endorsed`

Launching Application Explorer

The server must be started where Application Explorer is deployed.

To start Application Explorer:

1. Ensure the server is started where Application Explorer is deployed.
2. On Windows, invoke `iaexplorer.exe`, found under *`OracleAS_home`*\adapters\application\tools or on UNIX, invoke the iwae script, `iwae.sh`, found under *`OracleAS_home`*/adapters/application/tools.

OracleAS_home
Is the directory where the Oracle Application Server is installed.

Application Explorer opens. You are ready to define new targets to your Enterprise Information System (EIS).

Configuring Settings for BSE or JCA

Before a repository project can be created, you must configure OracleAS Adapter Business Services Engine (BSE). You need not configure the OracleAS Adapter JCA because the `ra.xml` file is configured automatically during installation.

Configuring the OracleAS Adapter Business Services Engine

After the OracleAS Adapter Business Services Engine (BSE) is deployed to Oracle Application Server, you can configure it through the BSE configuration page.

To configure BSE:

1. Open the following page in your browser:

```
http://hostname:port/bse
```

hostname
Is the hostname of the Oracle Application Server.

port
Is the HTTP port for the Oracle Application Server.

For example,

```
http://localhost:7777/bse
```

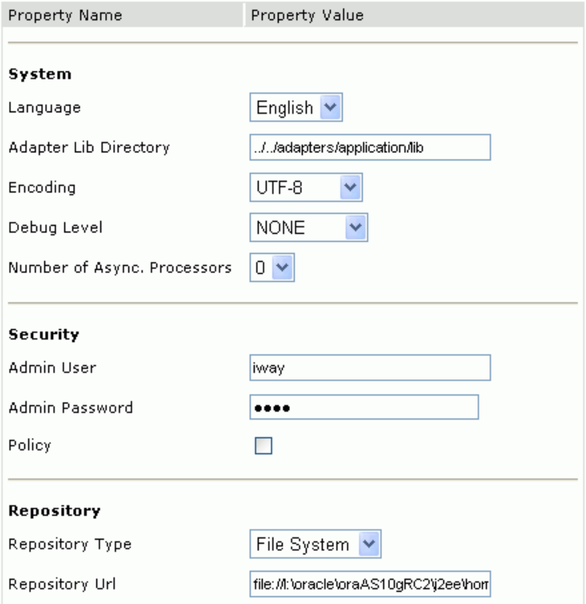
Note: The first time you access this page, it may take time to load.

2. When prompted, log on.

   When first installed, the user ID and the password are:

   - User name: admin
   - Password: admin

The BSE configuration page opens.



Property Name	Property Value
System	
Language	English
Adapter Lib Directory	../../adapters/application/lib
Encoding	UTF-8
Debug Level	NONE
Number of Async. Processors	0
Security	
Admin User	iway
Admin Password	••••
Policy	□
Repository	
Repository Type	File System
Repository Url	file:///I:\oracle\oraAS10gRC2\j2ee\hom

3. Ensure the Adapter Lib Directory parameter specifies the path to the lib directory, for example:

   OracleAS_home\adapters\application\lib

   After you specify the path, adapters in the lib directory are available to BSE.

4. For security purposes, type a new password in the **Admin Password** field.

   > **Note:** The Repository URL field specifies where the file system repository is located. To use a database repository, you must enter the repository connection information. For the initial verification, use a file system repository. See "Configuring an Oracle Repository" on page 2-6 for information on switching to a database repository.

5. Click **Save.**

Configuring BSE System Settings

To configure Business Services (BSE) system settings:

1. Open the BSE configuration page by entering the following URL:

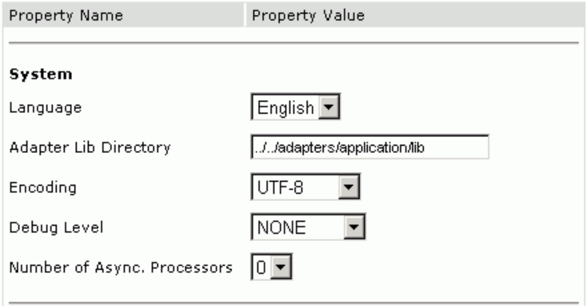
   http://*hostname*:*port*/ibse/IBSEConfig

hostname
Is the machine where BSE is installed.

port
Is the port number on which BSE is listening.

> **Note:** The server to which BSE is deployed must be running.

The BSE settings window opens.

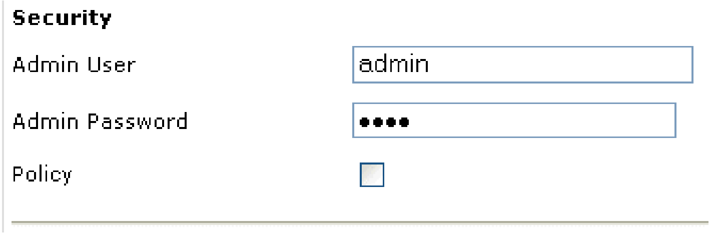


Property Name	Property Value
System	
Language	English
Adapter Lib Directory	../../adapters/application/lib
Encoding	UTF-8
Debug Level	NONE
Number of Async. Processors	0

2. Configure the system settings by providing information for the parameters according to the following table.

Parameter	Description
Language	Specify your required language.
Adapter Lib Directory	Type the full path to the directory where the adapter jar files reside
Encoding	Specify the default encoding from one of the following options: UTF-8 EBCDIC-CP-US ISO-88859-1 Shift JIS UNICODE
Debug Level	Specify the debug level from one of the following options: None Fatal Error Warning Info Debug
Number of Async. Processors	Select the number of asynchronous processors.

The following image illustrates the Security pane of the window.

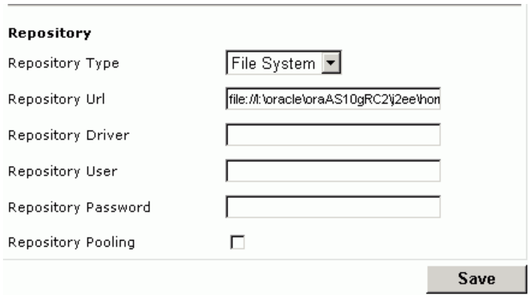


Security	
Admin User	admin
Admin Password	●●●●
Policy	□

3. Configure the security settings by providing information for the parameters according to the following table.

Parameter	Description
Admin User	Provide an BSE administrator ID.
Admin Password	Type the password associated with the BSE administrator ID.
Policy	Select the check box to enable policy security.

The following image illustrates the Repository pane of the window.



Repository

Repository Type: File System

Repository Url: file://I:\oracle\oraAS10gRC2\j2ee\hon

Repository Driver:

Repository User:

Repository Password:

Repository Pooling: □

4. Configure the repository settings by providing information for the parameters according to the following table.

   BSE requires a repository to store transactions and metadata required for the delivery of Web services.

   See "Configuring a File System Repository" on page 2-5 and "Configuring an Oracle Repository" on page 2-6 for more information.

Parameter	Description
Repository Type	Select one of the following repositories from the list: Oracle File
Repository URL	Type the URL to use when opening a connection to the database.
Repository Driver	Provide the driver class to use when opening a connection to the database (optional).
Repository User	Type the user ID to use when opening a connection to the database.
Repository Password	Type the password associated with the user ID.
Repository Pooling	Select the check box to enable pooling.

5. Click **Save.**

Configuring a File System Repository

If you do not have access to a database for the repository, you can store repository information in an XML file on your local machine. However, a file system repository is less secure and efficient than a database repository. When BSE is first installed, it is automatically configured to use a file system repository.

The default location for the repository on Windows is:

```
OracleAS_home\j2ee\OC4J_CONTAINER\applications\ws-app-adapter
\ibse\ibserepo.xml
```

On other platforms, use the corresponding location.

If you are using a file system repository, you are not required to configure any additional BSE components.

Configuring an Oracle Repository

To configure an Oracle repository:

1. Contact your database administrator to obtain an Oracle user ID and password to create the BSE repository.

   This user ID should have rights to create and modify tables as well as the ability to create and execute stored procedures.

2. Open a command prompt and navigate to the setup directory. Its default location on Windows is:

   ```
   OracleAS_home\adapters\application\etc\setup
   ```

   For other platforms, see the corresponding location.

   This directory contains SQL to create the repository tables in the following file:

   ```
   iwse.ora
   ```

   > **Note:** If Oracle is not on the same machine as the Oracle Application Server, copy the iwse.ora file to the Oracle machine. Then, from a command prompt on the Oracle machine, navigate to the directory containing the iwse.ora file.

3. Type the following command:

   ```
   sqlplus userid/password @database @ iwse.ora
   ```

Creating a Repository Project

Before you use Application Explorer with the OracleAS Adapter for Siebel, you must create a repository project. You can create two kinds of repository projects, Web services and JCA, depending on the container to which the adapter is deployed.

At design time, the repository is used to store metadata created when using Application Explorer to configure adapter connections, browse EIS objects, configure services, and configure listeners to listen for EIS events. The information in the repository is also referenced at runtime.

A default JCA repository is created for the default ManagedConnectionFactory. The name of this project is `jca_sample`.

Web services and the OracleAS Adapter Business Services Engine (BSE) refer to the same type of deployment. See "Adapter Features" on page 1-1 for more information.

Creating a Web Service Project for the Web Service Adapter

To create a Web service project for BSE using Application Explorer, you must first define a new configuration.

Defining a New Configuration for BSE

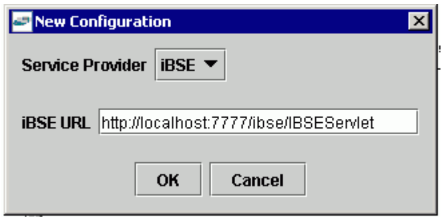
To create a new repository project for BSE:

1. Right-click **Configurations** and select **New.**

   The New Configuration dialog box opens.

2. Enter a name for the new configuration (for example, SampleConfig) and click **OK.**

   The following opens.



iBSE URL http://localhost:7777/ibse/IBSEServlet

3. From the **Service Provider** list, select **BSE.**
4. In the **BSE URL** field, accept the default URL or replace it with a different URL with the following format:

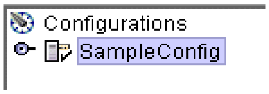
   ```
   http://hostname:port/ibse/IBSEServlet
   ```

hostname
Is the machine where your application server resides.

port
Is the port number where the application server is listening.

5. Click **OK.**

   A node representing the new configuration appears beneath the root Configurations node.



The Web service repository project file is stored in *OracleAS_home*\j2ee\home\applications\ws-app-adapter\ibse.

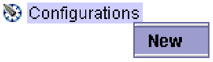
Creating a Repository Project for the OracleAS Adapter JCA

To create a repository project for the OracleAS Adapter JCA using Application Explorer, you must first define a new configuration.

Defining a New Configuration for JCA

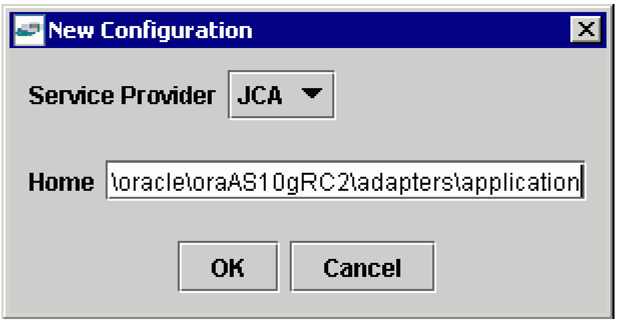
To define a new configuration for JCA:

1. Right-click **Configurations** and select **New.**



   The New Configuration dialog box opens.

2. Enter a name for the new configuration (for example, SampleConfig) and click **OK.**



Service Provider: JCA

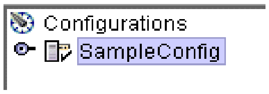
Home: \oracle\oraAS10gRC2\adapters\application

3. From the **Service Provider** list, select **JCA.**
4. In the **Home** field, enter a path to your JCA configuration directory where the repository, schemas, and other information is stored, for example:

   OracleAS_home\adapters\application

5. Click **OK.**

   A node representing the new configuration appears beneath the root Configurations node.



The OracleAS Adapter JCA repository project file is stored in *OracleAS_home*\adapters\application\config\configuration_name

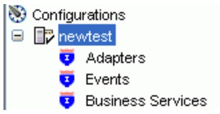
configuration_name
Is the name of the configuration you created; for example, SampleConfig.

Connecting to a New Configuration

To connect to a new configuration:

1. Right-click the configuration to which you want to connect, for example, SampleConfig.
2. Select **Connect.**

   Nodes appear for **Adapters, Events,** and **Business Services** (also known as Web services).



- Use the **Adapters** folder to create inbound interaction with Siebel. For example, you use the Siebel node in the Adapters folder to configure a service that updates Siebel.
- Use the **Events** folder to configure listeners that listen for events in Siebel.
- Use the **Business Services** folder to test Web services created in the Adapters folder. You can also control security settings for the Web services by using the security features of the Business Services folder.

You are now ready to define new targets to Siebel.

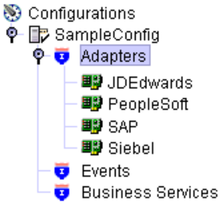
Establishing a Connection (Target) for Siebel

To browse the Siebel Business Services, Business Components, and Integration Objects, you must define a target to Siebel. After you define the target, the parameters are automatically saved. However, you must supply the password to Siebel every time you connect to the target.

Defining a Target to Siebel

To define a target to Siebel:

1. In the left pane, expand the **Adapters** node.



2. Right-click the **Siebel** node and select **Add Target.**

   The Add Target dialog box opens.

   a. In the **Name** field, type a name for the new target.

   b. In the **Description** field, type a description (optional).

   c. From the **Target Type** list, select the type of target; for Siebel 6.0, choose **Siebel 6.2 or lower (COM).** For Siebel 6.3 or higher, choose **Java Bean Data Connection.**

3. Click **OK.**

   - When you select Siebel 6.2 or lower (COM):

   a. In the **User Agent File** field, type the name of the configuration file.

   b. In the **Username** field, type the user name.

   c. In the **Password** field, type the password associated with the user name.

   d. In the **Repository** field, type the Siebel Repository where Application Explorer looks for metadata describing Business Services, Business Objects, and Integration Objects.

      If no repository is specified, a full list of objects from all available repositories will be returned. If a specified repository is not found, an empty list of objects will be returned.

   - When you select 6.3 or higher (JDB):

   a. In the **Gateway Server** field, type the name of the server.

   b. In the **Enterprise** Name field, type the appropriate name.

   c. In the **Siebel Server** field, type the name of your Siebel server.

      Do not populate this field for Siebel 7.7.1.

   d. In the **User** field, type the user name.

   e. In the **Password** field, type the password associated with the user name.

   f. Click the **Advanced** tab and verify the following:

Language

Object Manager

For Siebel 7.0.3, the default Object Manager is EAIObjMgr. For Siebel 7.7, the default is EAIObjMgr_enu.

Repository Manager

If no repository is specified, a full list of objects from all available repositories will be returned. If a specified repository is not found, an empty list of objects will be returned.

The configuration parameters supplied are those used by Siebel client applications to connect to the Siebel system. For more information about these parameters, see your Siebel documentation or ask your Siebel system administrator.

4. Click **OK.**

   In the left pane, the target you create appears under the Siebel node.

Connecting to a Defined Target

To connect to a defined target:

1. Expand the **Siebel** node and click the target name to which you want to connect.
2. In the right pane, enter the password for that target.
3. In the left pane, right-click the target name and select **Connect.**

   The icon changes, and a plus sign(+) next to the target icon indicates that you are connected to the Siebel system.

   You can browse available Business Objects, Business Services, and Integration Objects in the Siebel system.

Disconnecting From Siebel

Although you can maintain multiple open connections to different application systems, it is good practice to close connections when not in use.

To disconnect from Siebel:

1. In the left pane, select the target to which you are connected.
2. Right-click the target and select **Disconnect.**

   Disconnecting from the application system drops the target, but the node remains. The SiebelConnection node in the left pane changes to reflect that the target is closed.

Editing a Target

To edit a target:

1. In the left pane, ensure the target you wish to edit is disconnected.
2. Right-click the target and select **Edit.**

   The **Edit** pane opens on the right.
3. Modify the target information.
4. Click **OK.**

Deleting a Target to Siebel

You can delete a target, rather than just disconnecting and closing it. When you delete the target, the node disappears from the list of Siebel targets in the left pane of the explorer.

To delete a target:

1. In the left pane, select the target.
2. Right-click the target and select **Delete.**

   A confirmation dialog box opens.
3. Click **OK** to delete the target you selected.

   The SiebelConnection node disappears from the left pane.

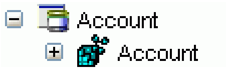
Viewing Application System Objects

Application Explorer gives you the flexibility of viewing all Siebel application system objects. One benefit of this flexibility is that you can gain an understanding of the Siebel data structure. You can review parameters, data types, and other attributes of the Siebel data in the right pane.

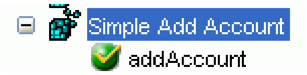
Viewing Metadata

To view metadata:

1. If you have not started the explorer, start Application Explorer and connect to your Siebel system.
2. In the left pane, expand the **Business Object** or **Business Service** containing the component for which you want to generate schema.
3. Expand the **Business Object** or **Business Service** node.
4. Expand the **Business Component** or the **Business Service** node to view the objects under it.
   - For a **Business Component,** select the node in which you are interested, for example, **Account.**

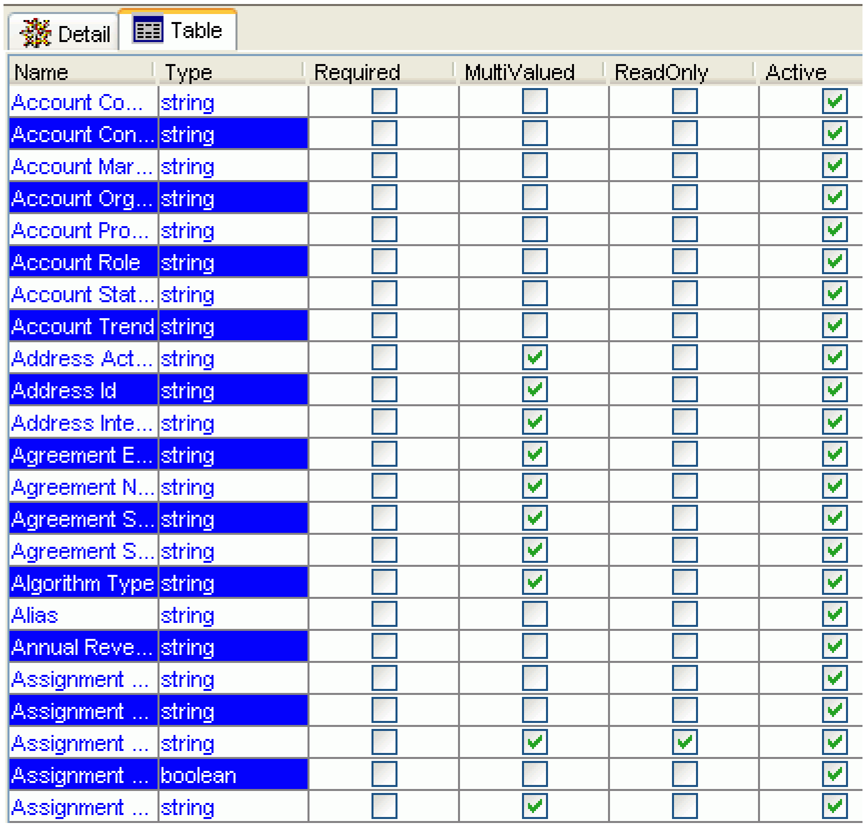


   - For a **Siebel Business Service,** select the object in which you are interested, for example, **addAccount.**



5. In the right pane, click the ellipsis (...) in the **Table** row of the properties table.

   The metadata table appears in the right pane.



Name	Type	Required	MultiValued	ReadOnly	Active
Account Co...	string	□	□	□	☑
Account Con...	string	□	□	□	☑
Account Mar...	string	□	□	□	☑
Account Org...	string	□	□	□	☑
Account Pro...	string	□	□	□	☑
Account Role	string	□	□	□	☑
Account Stat...	string	□	□	□	☑
Account Trend	string	□	□	□	☑
Address Act...	string	□	☑	□	☑
Address Id	string	□	☑	□	☑
Address Inte...	string	□	☑	□	☑
Agreement E...	string	□	☑	□	☑
Agreement N...	string	□	☑	□	☑
Agreement S...	string	□	☑	□	☑
Agreement S...	string	□	☑	□	☑
Algorithm Type	string	□	☑	□	☑
Alias	string	□	□	□	☑
Annual Reve...	string	□	□	□	☑
Assignment ...	string	□	□	□	☑
Assignment ...	string	□	□	□	☑
Assignment ...	string	□	☑	☑	☑
Assignment ...	boolean	□	□	□	☑
Assignment ...	string	□	☑	□	☑

Creating an XML Schema

You can create service schemas for Business Services and Business Components using Application Explorer.

The following topic, "Creating an XML Schema for a Siebel Business Object or Business Service" describes how to create schemas for the adapter when you deploy the OracleAS Adapter for Siebel for use either in a JCA (J2EE Connector Architecture) environment or a Web services environment.

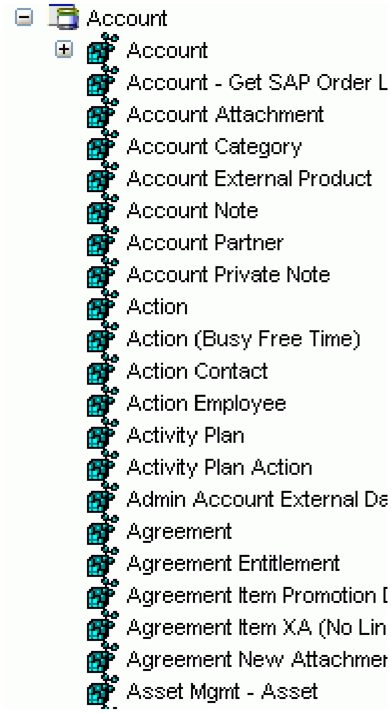
See "Creating a Web Service or Business Service" on page 2-14 if you plan to deploy the OracleAS Adapter for Siebel in a Web services environment.

Creating an XML Schema for a Siebel Business Object or Business Service

You create schemas for Siebel Business Service methods (for example, the Add method) and Business Components using Application Explorer. After you create a schema, you can use it to generate service request and response schemas for the Business Service or Business Component.

Siebel Business Objects contain one or more Siebel Business Components. You can view Business Components by clicking the associated Business Object.

The following illustration shows the Account Business Object expanded to display all Business Components.

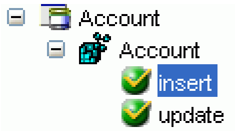


Account
Account
Account - Get SAP Order L
Account Attachment
Account Category
Account External Product
Account Note
Account Partner
Account Private Note
Action
Action (Busy Free Time)
Action Contact
Action Employee
Activity Plan
Activity Plan Action
Admin Account External Da
Agreement
Agreement Entitlement
Agreement Item Promotion [
Agreement Item XA (No Lin
Agreement New Attachmer
Asset Mgmt - Asset

Creating an XML Schema for a Siebel Business Component or Business Service

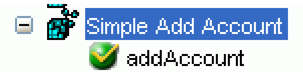
To generate service request and response schemas for a Business Component or Business Service:

1. If you have not started the explorer, start **Application Explorer** and connect to your Siebel system.
2. In the left pane, expand the **Business Object** or the **Business Service** node.
3. Expand the **Business Component** or **Business Service** to view the objects under it.
   - For a **Business Component,** expand the component containing the node you wish and select the method for which you want to create a schema, for example, insert.



Account
Account
insert
update

   - For a **Siebel Business Service,** expand the **Business Service** node containing the object for which you want to create schema.

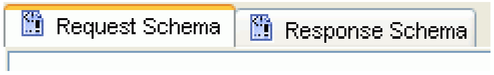


Simple Add Account
addAccount

4. Right-click the node and select **Generate Schema.**

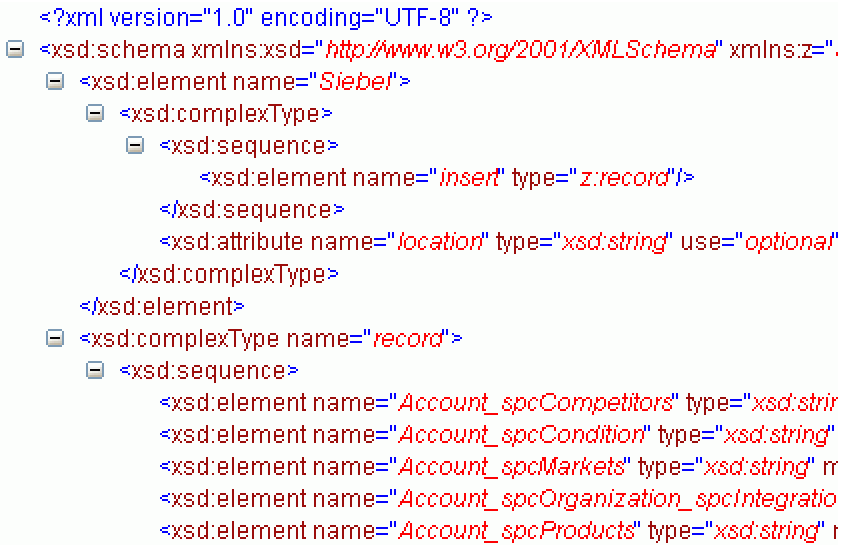
   Application Explorer accesses the Siebel repository and builds schemas.

   Schema tabs similar to the following appear in the right pane.



5. To view a schema, click the ellipsis tab corresponding to the schema you want to view.

   The schema appears on the right.



```
  <?xml version="1.0" encoding="UTF-8" ?>
- <xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:z="
  - <xsd:element name="Siebel">
    - <xsd:complexType>
      - <xsd:sequence>
          <xsd:element name="insert" type="z:record"/>
        </xsd:sequence>
        <xsd:attribute name="location" type="xsd:string" use="optional"
      </xsd:complexType>
    </xsd:element>
  - <xsd:complexType name="record">
    - <xsd:sequence>
        <xsd:element name="Account_spcCompetitors" type="xsd:strir
        <xsd:element name="Account_spcCondition" type="xsd:string"
        <xsd:element name="Account_spcMarkets" type="xsd:string" m
        <xsd:element name="Account_spcOrganization_spcIntegratio
        <xsd:element name="Account_spcProducts" type="xsd:string" r
```

Creating a Web Service or Business Service

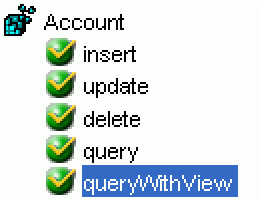
You can generate a **business service** (also known as a Web service) for Siebel objects you wish to use with your adapter after you have properly configured the servlet BSE.

> **Note:** In a J2EE Connector Architecture (JCA) implementation of adapters, Web services are not available. When the adapters are deployed to use the OracleAS Adapter JCA, the Common Client Interface provides integration services using the adapters.

As an example, the following procedure describes the creation of a Web service for a Business Object.

To generate a Web service for a Siebel Business Object:

1. If you have not already connected, connect to your Siebel system.
2. Expand a **Business Object** node.
3. Expand the **Business Component** for which you want to create a Web service.



4. Expand the object and select a method for creating the Web service, for example, QueryWithView under Account.
5. Right-click the node from which you want to create a business service and select **Create Business Service.**

The Create Web Service dialog box opens.

You can add the business object as a method for a new Web service or as a method for an existing one.

a. From the **Existing Service Names** list, select either **<new service>** or an existing service.

b. Specify a service name if you are creating a new service. This name identifies the Web service in the list of services under the **Business Services** node.

c. Type a description for the service (optional).

d. Select one of the available licenses.

6. Click **Next.**

   The License and Method dialog box opens.

   a. In the **License** field, select one or more license codes to assign to the Web service. To select more than one, hold down the **Ctrl** key and click the licenses.

   b. In the **Method Name** field, type a descriptive name for the method.

   c. In the **Description** field, type a brief description of the method.

7. Click **OK.**

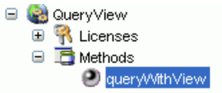
   Application Explorer switches the view to the **Business Services** node, and the new Web service appears in the left pane.

Testing a Web Service

After you create a Web service for the Siebel Business Object, test it to ensure it functions properly. Application Explorer includes a test tool for testing a Web service.

Testing a Web Service for a Business Object

1. In the left pane of Application Explorer, expand the **Business Services** node.
2. Expand the **Services** node.
3. Select the name of the business service you want to test.



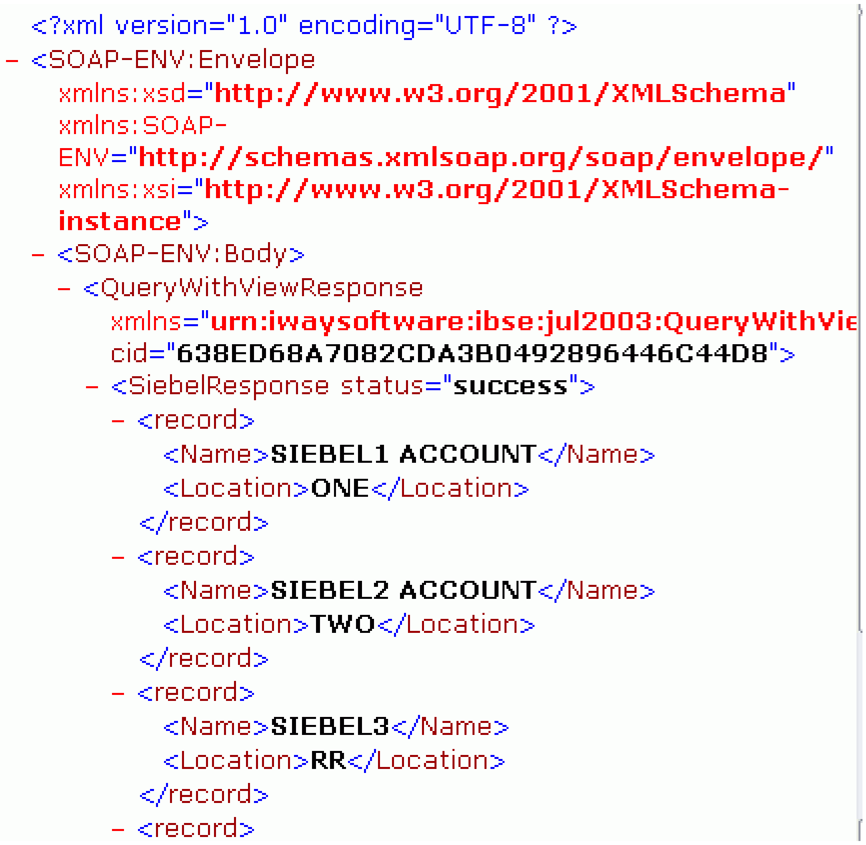
4. Expand the **Methods** node under the service and select the method you want to test.

   The test option appears in the right pane.

   If you are testing a Web service that requires XML input, an input field appears.

5. Click **Invoke.**

   Application Explorer displays the results in the results pane.



```
  <?xml version="1.0" encoding="UTF-8" ?>
- <SOAP-ENV:Envelope
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    xmlns:SOAP-
    ENV="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-
    instance">
  - <SOAP-ENV:Body>
    - <QueryWithViewResponse
        xmlns="urn:iwaysoftware:ibse:jul2003:QueryWithVie
        cid="638ED68A7082CDA3B0492896446C44D8">
      - <SiebelResponse status="success">
        - <record>
            <Name>SIEBEL1 ACCOUNT</Name>
            <Location>ONE</Location>
          </record>
        - <record>
            <Name>SIEBEL2 ACCOUNT</Name>
            <Location>TWO</Location>
          </record>
        - <record>
            <Name>SIEBEL3</Name>
            <Location>RR</Location>
          </record>
        - <record>
```

Testing a Web Service for a Business Service

After you create a Web service for the Siebel Business Service, test it to ensure it functions properly. Application Explorer includes a test tool for testing a Web service.

1. If it is not expanded, expand the **Business Services** node.
2. Expand the **Services** node.
3. Select the name of the business service you want to test.
4. Expand the **Methods** node and select the name of the method you want to test.

   The test option appears in the right pane.

   If you are testing a Web service that requires XML input, an input field appears.
5. Provide the appropriate input.
6. Click **Invoke.**

   Application Explorer displays the results in the results pane.

Configuring an Event Adapter

Events are generated as a result of a specific business condition being satisfied or triggered in the Siebel system. You can use events to trigger an action in your application. For example, an update to a database can reflect an update to customer information. If your application must perform when this happens, your application is a consumer of this event.

After you create a connection to your application system, you can add events using Application Explorer. To configure an event, you must create a port and a channel.

- Port

  A port associates a particular business object exposed by an adapter with a particular disposition. A disposition defines the protocol and location of the event data. The port defines the end point of the event consumption. See "Creating an Event Port" on page 2-21 for more information.

- Channel

  A channel represents configured connections to particular instances of back-end or other types of systems. A channel binds one or more event ports to a particular listener managed by an adapter. See "Creating a Channel" on page 2-23 for more information.

> **Note:** OC4J currently conforms to JCA 1.0, which does not call for event capabilities. When conforming to JCA 1.0, only service interactions are supported.

Creating an Event Schema for a Siebel Integration Object

You can listen for Siebel Integration Object events by configuring ports and channels. Before you create the ports and channels, you must create a schema. To create XML schemas for Siebel Integration Objects, you must first generate XDR schemas using the Siebel Tools Schema Wizard.

The XDR schema is used as input to Application Explorer when generating schemas for integration objects. After you generate the XDR schema, Application Explorer uses the XDR file to generate the XML schema.

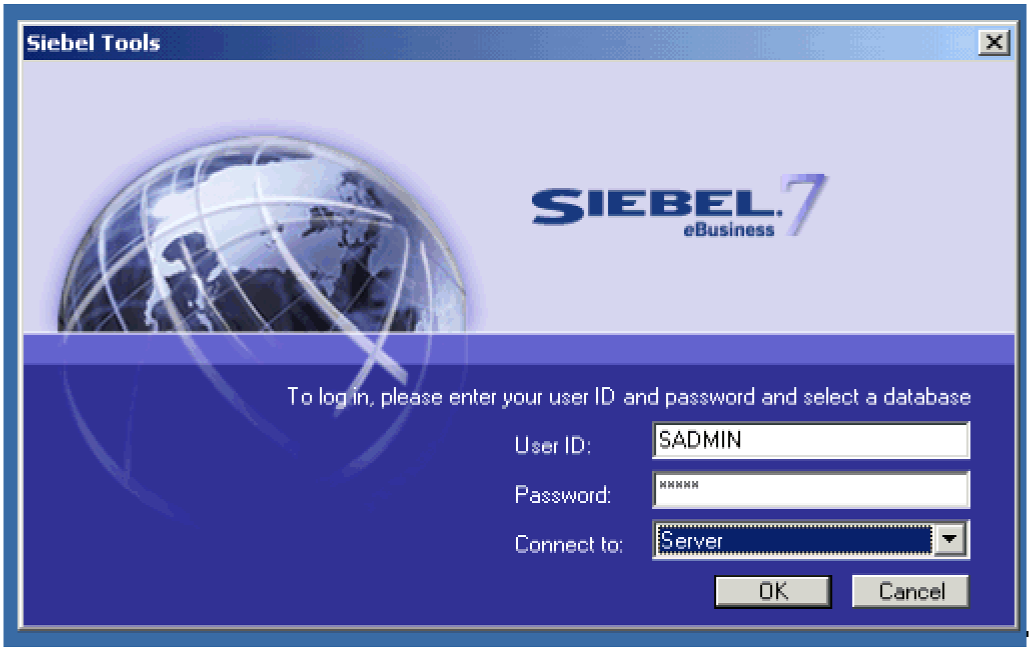
The OracleAS Adapter for Siebel supports access to Siebel Integration Objects by using Siebel XML to handle events. Using Siebel Integration Objects through supported transports requires Siebel workflows.

> **Note:** For releases prior to Siebel 6.3, the Siebel Tools Schema Wizard creates only DTD schemas. You must transform these schemas manually, or by using other tools, into XDR files before Application Explorer can use them as input to create XML schemas. In addition, you must include the SiebelMessage tag reference in your XDR file.

Creating a Siebel XDR Schema for a Siebel Integration Object

To generate a Siebel XDR schema:

1. Log on to Siebel Tools.



To log in, please enter your user ID and password and select a database

User ID: SADMIN

Password: *****

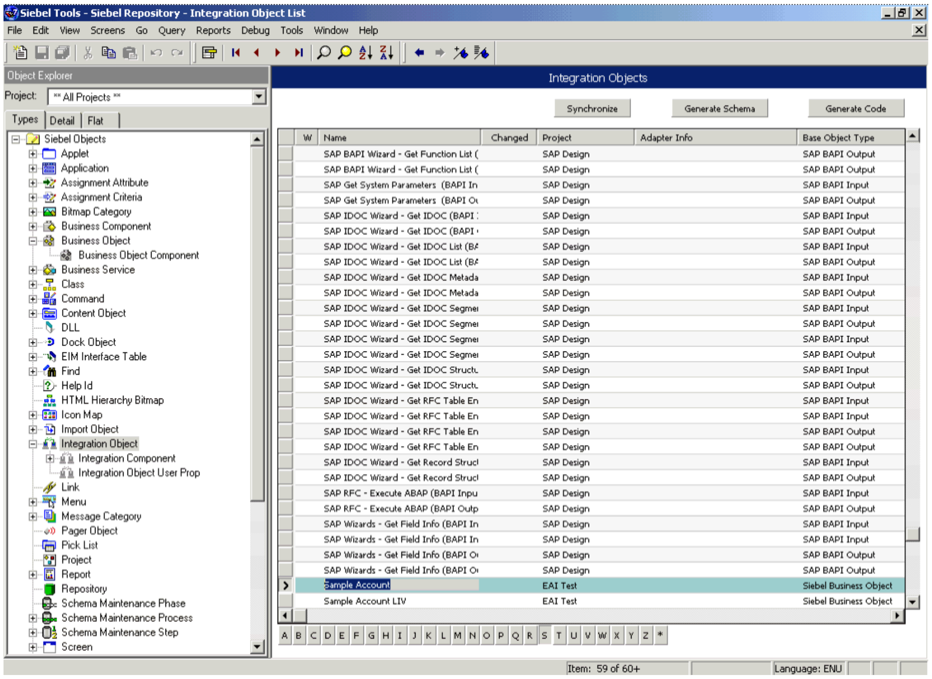
Connect to: Server

a. Type your user ID and password.

b. Select a database from the list.

2. Click **OK.**

The Siebel Tools window opens. Integration Objects appear in the right pane.

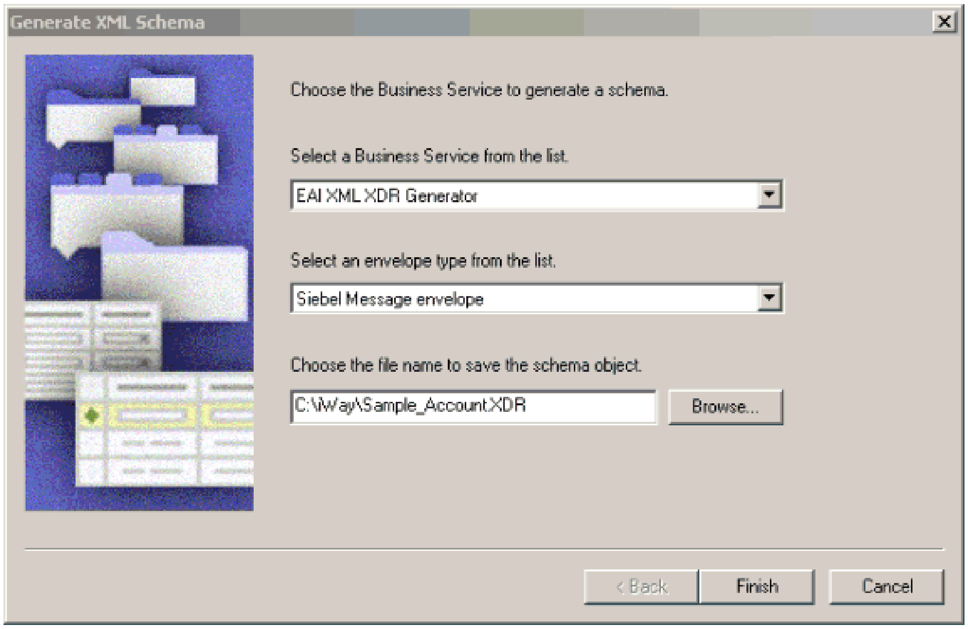


W	Name	Changed	Project	Adapter Info	Base Object Type
	SAP BAPI Wizard - Get Function List (		SAP Design		SAP BAPI Output
	SAP BAPI Wizard - Get Function List (		SAP Design		SAP BAPI Output
	SAP Get System Parameters (BAPI In		SAP Design		SAP BAPI Input
	SAP Get System Parameters (BAPI Ou		SAP Design		SAP BAPI Output
	SAP IDOC Wizard - Get IDOC (BAPI :		SAP Design		SAP BAPI Input
	SAP IDOC Wizard - Get IDOC (BAPI ‹		SAP Design		SAP BAPI Output
	SAP IDOC Wizard - Get IDOC List (BA		SAP Design		SAP BAPI Input
	SAP IDOC Wizard - Get IDOC List (BA		SAP Design		SAP BAPI Output
	SAP IDOC Wizard - Get IDOC Metada		SAP Design		SAP BAPI Input
	SAP IDOC Wizard - Get IDOC Metada		SAP Design		SAP BAPI Output
	SAP IDOC Wizard - Get IDOC Segme		SAP Design		SAP BAPI Input
	SAP IDOC Wizard - Get IDOC Segme		SAP Design		SAP BAPI Output
	SAP IDOC Wizard - Get IDOC Segme		SAP Design		SAP BAPI Input
	SAP IDOC Wizard - Get IDOC Segme		SAP Design		SAP BAPI Output
	SAP IDOC Wizard - Get IDOC Structu		SAP Design		SAP BAPI Input
	SAP IDOC Wizard - Get IDOC Structu		SAP Design		SAP BAPI Output
	SAP IDOC Wizard - Get RFC Table En		SAP Design		SAP BAPI Input
	SAP IDOC Wizard - Get RFC Table En		SAP Design		SAP BAPI Input
	SAP IDOC Wizard - Get RFC Table En		SAP Design		SAP BAPI Output
	SAP IDOC Wizard - Get RFC Table En		SAP Design		SAP BAPI Output
	SAP IDOC Wizard - Get Record Struc		SAP Design		SAP BAPI Input
	SAP IDOC Wizard - Get Record Struc		SAP Design		SAP BAPI Output
	SAP RFC - Execute ABAP (BAPI Inpu		SAP Design		SAP BAPI Input
	SAP RFC - Execute ABAP (BAPI Outp		SAP Design		SAP BAPI Output
	SAP Wizards - Get Field Info (BAPI In		SAP Design		SAP BAPI Input
	SAP Wizards - Get Field Info (BAPI In		SAP Design		SAP BAPI Input
	SAP Wizards - Get Field Info (BAPI Ou		SAP Design		SAP BAPI Output
	SAP Wizards - Get Field Info (BAPI Ou		SAP Design		SAP BAPI Output
	Sample Account		EAI Test		Siebel Business Object
	Sample Account LIV		EAI Test		Siebel Business Object

3. To create a schema, select an Integration Object, for example, Sample Account.

4. Click **Generate Schema.**

The Generate XML Schema wizard opens.



Choose the Business Service to generate a schema.

Select a Business Service from the list.

EAI XML XDR Generator

Select an envelope type from the list.

Siebel Message envelope

Choose the file name to save the schema object.

C:\iWay\Sample_Account.XDR

a. From the **Select a Business Service** list, select **EAI XML XDR Generator.**

b. From the **Select an envelope type** list, select **Siebel Message envelope.**

c. In the **Choose the file name** field, specify a file name for the XDR schema and a directory where it can be accessed by Application Explorer.

Note: The XDR schema file must be saved to a directory on the same computer as Application Explorer.

5. Click **Finish.**

   Now you can use Application Explorer to generate XML schemas for the Siebel Integration Object.

Creating a Schema from a Siebel XDR Schema

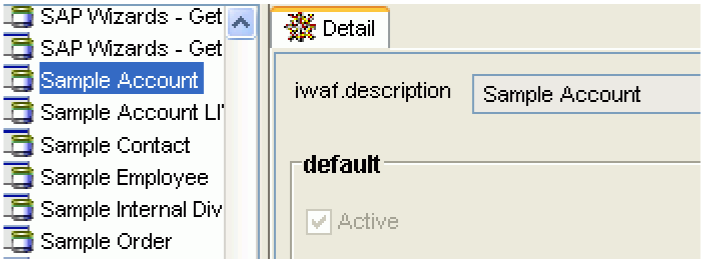
After you create the Siebel XDR schema for a selected Siebel Integration Object, you can create an XML schema using Application Explorer.

You must supply Application Explorer with the location of the previously created Siebel XDR schema for the particular integration object selected.

Note: The XDR file must be on the same computer as Application Explorer or be available through a mapped connection to another drive or machine.

To create a schema from a Siebel XDR Schema:

1. In Application Explorer, expand the **Integration Objects** node to browse the Integration Objects in the Siebel system.



iwaf.description: Sample Account

default

☑ Active

2. Scroll down and select an integration object, for example, Sample Account.
3. To generate event schema, right-click the object and choose **Add event.**

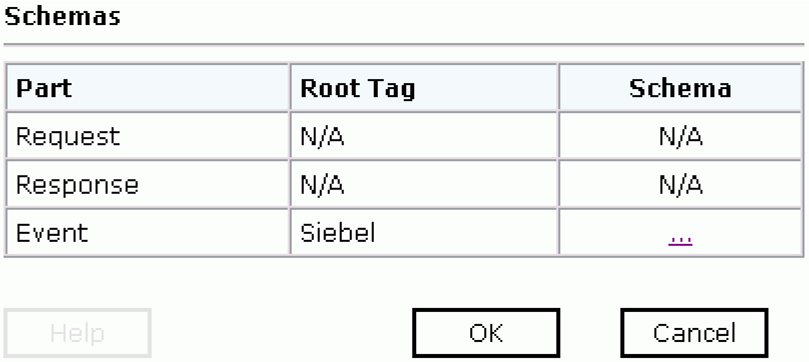
   The Add Event dialog box opens.

   a. In the **Node** name field, type a name for the node to be created under Sample Account.

   b. Type or browse to the location of the XDR schema where Application Explorer generates the service or event schema.

4. Click **Add.**

   The Schemas pane opens.

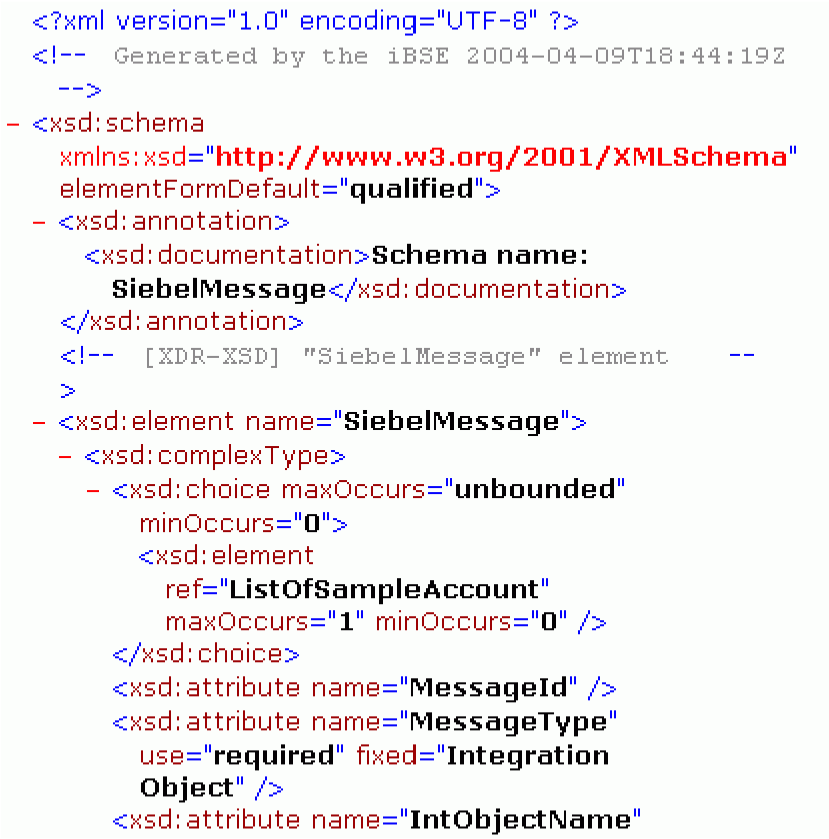


Schemas

Part	Root Tag	Schema
Request	N/A	N/A
Response	N/A	N/A
Event	Siebel	...

5. To view the XML for a schema, click the **Event Schema** tab.

   The results appear in the right pane.



```
 <?xml version="1.0" encoding="UTF-8" ?>
 <!--  Generated by the iBSE 2004-04-09T18:44:19Z
   -->
- <xsd:schema
   xmlns:xsd="http://www.w3.org/2001/XMLSchema"
   elementFormDefault="qualified">
 - <xsd:annotation>
     <xsd:documentation>Schema name:
       SiebelMessage</xsd:documentation>
   </xsd:annotation>
   <!--  [XDR-XSD] "SiebelMessage" element    --
   >
 - <xsd:element name="SiebelMessage">
   - <xsd:complexType>
     - <xsd:choice maxOccurs="unbounded"
         minOccurs="0">
         <xsd:element
           ref="ListOfSampleAccount"
           maxOccurs="1" minOccurs="0" />
       </xsd:choice>
       <xsd:attribute name="MessageId" />
       <xsd:attribute name="MessageType"
         use="required" fixed="Integration
         Object" />
       <xsd:attribute name="IntObjectName"
```

6. Click the browser **Back** button to return.

   A directory structure is created to store the schemas.

   You are now ready to configure ports and channels for Siebel Integration Objects.

Creating an Event Port

To create an event port for Siebel Integration Objects, you must first indicate the location of the XDR schema for that object.

You can create an event port from the Events node for Siebel Integration Objects. To create event ports for Siebel Business Objects and Siebel Business Services, you must use the Events node.

Creating an Event Port from the Adapters Node

For Siebel Integration Objects, you can bypass the Events node and create an event port directly from the Adapters node.

1. Select the Integration Object event you created.
2. Right-click the Integration Object event and select **Add Port.**

   The Add Port dialog box opens.

   a. Type a name for the event port and provide a brief description.

   b. From the list, select the required disposition, for example, RMI.

   c. In the URL field, type the disposition url.
3. Click **OK.**

Creating an RMI Event Port From the Events Node

The following procedure describes how to create an event port from the Events node for an RMI disposition using Application Explorer.

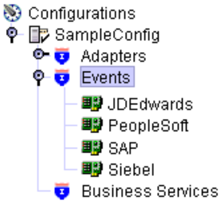
To create an event port for Siebel Integration Objects, you must first indicate the location of the XDR schema for that object. See "Creating a Schema from a Siebel XDR Schema" on page 2-19 for more information.

You also can create an event port directly from the Adapters node. See "Creating an Event Port from the Adapters Node" on page 2-21 for more information.

Creating an Event Port for RMI

To create a specific event port for RMI:

1. Click the **Events** node.



2. Expand the **Siebel** node.
3. Right-click the **Ports** node and select **Add Port.**

   The Add Port dialog box opens.

   a. Type a name for the event port and provide a brief description.
   b. From the **Protocol** list, select **RMI.**
   c. In the **URL** field, specify a destination file to which the event data is written using the following format:

   ```
   rmi://host:port;RemoteObject=[APPNAME];errorTo=[pre-defined port name or
   another disposition url]
   ```

   d. From the **Disposition** protocol list, select **RMI.**

   The following table defines the parameters for the disposition.

Parameter	Description
host	The host name or IP address from which the RMI server accepts RMI requests. If you omit this attribute, the RMI server will accept RMI requests from any host.
port	The port number on which the RMI server listens for RMI requests.
RemoteObject	A home or Enterprise JavaBeans (EJB) object.
errorTo	Predefined port name or another disposition URL to which error logs are sent.

4. Click **OK.**

   The port appears under the ports node in the left pane. In the right pane, a table appears that summarizes the information associated with the event port you created.

You are ready to associate the event port with a channel. See "Creating a Channel" on page 2-23 for more information.

Editing an Event Port

To edit an event port:

1. In the left pane, select the event port you want to edit.
2. Right-click the port and select **Edit.**

   The Edit Port pane opens.
3. Make the required changes and click **OK.**

Deleting an Event Port

To delete an event port:

1. Select the event port you want to delete.
2. Right-click the port and select **Delete.**

   A confirmation dialog box opens.
3. To delete the event port you selected, click **OK.**

   The event port disappears from the list in the left pane.

Creating a Channel

The following procedure describes how to create a channel for your event. All defined event ports must be associated with a channel. Three channel types are available:

- HTTP
- MQ Series
- File

> **Note:** OC4J currently conforms to JCA 1.0, which does not call for event capabilities. When conforming to JCA 1.0, only service interactions are supported.

Creating an HTTP Channel

To create a channel:

1. Click the **Events** node.

   The Events window opens. The adapters that appear in the left pane support events.
2. In the left pane, expand the **Siebel** node.

   The ports and channels nodes appear.
3. Right-click **channels** and select **Add channel.**

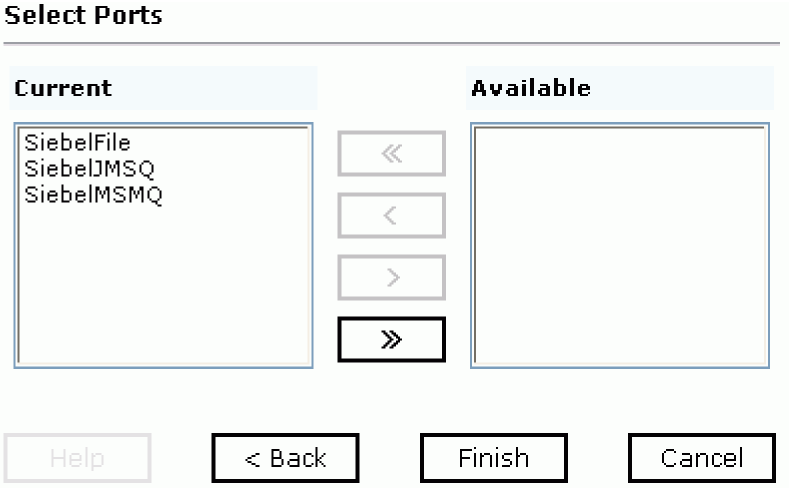
   The Add Channel dialog box opens.

   a. Type a name for the channel, for example, NewChannel.

   b. Type a brief description.

c. From the list, select **HTTP Listener.**

The following image shows the Select Ports dialog box.



d. Select an event port from the list of current ports.

e. To transfer the port to the list of available ports, click the double right (>>) arrow. To associate all the event ports, control-click to select each port or click one port and press **Control+A.** Then, click the double right (>>) arrow.

4. Click **Next.**

5. When the dialog box opens, type the system information as specified in the following table.

Parameter	Description
Port	Port on which to listen for Siebel event data.
Server port	Port on which the host database is listening.

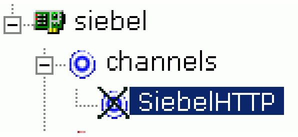
Synchronization Type	Possible values are:
	`REQUEST`
	`REQUEST_RESPONSE`
	`REQUEST_ACK`

6. Click **OK.**

The summary pane opens.

A summary provides the channel description, channel status, and available ports. All the information is associated with the channel you created.

The channel also appears under the channels node in the left pane



An X over the icon indicates that the channel is currently disconnected. You must start the channel to activate your event configuration.

7. Right-click the channel and select **Start the channel.**

   The channel you created becomes active.

   The X that was over the icon in the left pane disappears.

8. To stop the channel, right-click the channel and select **Stop the channel.**

Creating an MQ Series Listener

To create an MQ Series listener:

1. Click the **Events** node.

   The Events window opens. The adapters that appear in the left pane support events.

2. In the left pane, expand the **Siebel** node.

   The ports and channels nodes appear.

3. Right-click the **channels** node and select **Add channel.**

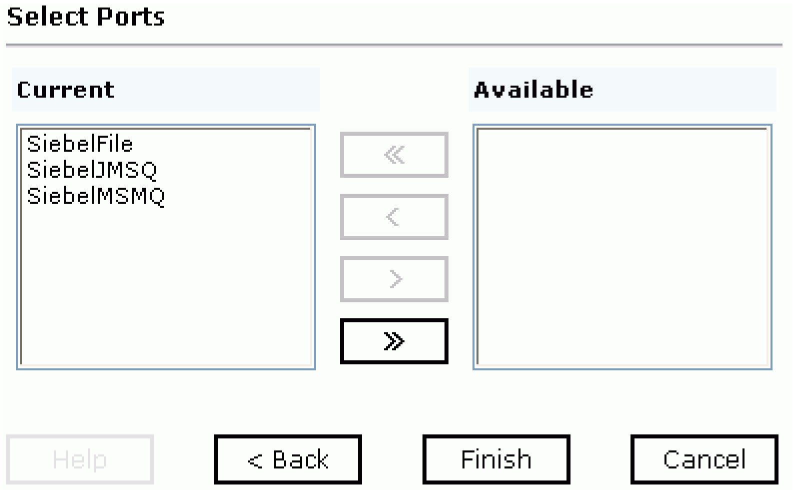
   The Add a new channel pane opens.

   a. Type a name for the channel, for example, NewChannel.

   b. Type a brief description.

   c. From the list, select **MQ Series Listener.**

      The following image shows the Select Ports dialog box.



Select Ports

Current

SiebelFile
SiebelJMSQ
SiebelMSMQ

Available

   d. Select an event port from the list of current ports.

   e. To transfer the port to the list of available ports, click the **double right (>>) arrow.**To associate all the event ports, control-click to select each port or click one port and press **Control+A.** Then, click the double right (>>) arrow.

4. Click **Next.**

5. When the dialog box opens, type the system information as follows.

   a. In the **Request** tab, enter values for the following parameters:

Parameter	Description
Queue manager name	The host on which the MQ Server is located (MQ Client only).
MQ server host for MQClient operation	Port on which the host database is listening.

MQ server port for MQClient operation	The number to connect to an MQ Server queue manager (MQ client only).
	`REQUEST`
	`REQUEST_RESPONSE`
	`REQUEST_ACK`

MQ server channel for MQClient operation	The case-sensitive name of the channel that connects with the remote MQ Server queue manager (MQ client only). The default channel name for MQSeries is SYSTEM.DEF.SVRCONN.
Document type XML	Leave the default selection.

Request queue name	Queue where the message is routed and where request documents are received. The name of the queue is case-sensitive and conforms to the following format:
	`Host\queue type$\qName`
	Host
	Is the machine name where the MQ Series queuing system is running.
	queue type
	Private queues are queues that are not published in Active Directory and appear only on the local computer where they reside. Private queues are accessible only by Message Queuing applications that recognize the full path name or format name of the queue.
	qName
	Is the name of the queue where messages are placed, for example,
	`iwaykxc1\Private$\siebel`

b. In the **Response** tab, enter values for the following parameters:

Parameter	Definition
Synchronization Type	Choose from three options: ■ REQUEST ■ REQUEST_RESPONSE ■ REQUEST_ACK

c. In the **Advanced** tab, enter values for the following parameters.

Parameter	Definition
Error Directory	Directory to which documents with errors are written.
Message wait interval (msec)	The interval (in milliseconds) when to check for new input. The default is 3 seconds. Optional.

Parameter	Definition
Mode of operation	Choose Sequential or Threaded. ■ Sequential indicates single processing of requests. ■ Threaded indicates processing of multiple requests simultaneously.
Thread limit	If you selected threaded processing, indicate the maximum number of requests that can be processed simultaneously.

6. Click **OK.**

   The summary pane opens.

   A summary provides the channel description, channel status, and available ports. All the information is associated with the channel you created. The channel also appears under the channels node in the left pane

   An X over the icon indicates that the channel is currently disconnected. You must start the channel to activate your event configuration.

7. Right-click the channel and select **Start the channel.**

   The channel you created becomes active.

   The X that was over the icon in the left pane disappears.

8. To stop the channel, right-click the channel and select **Stop the channel.**

Creating a File Listener

1. Click the **Events** node.

   The Events window opens. The adapters that appear in the left pane support events.

2. In the left pane, expand the **Siebel** node.

   The ports and channels nodes appear.

3. Right-click the **channels** node and select **Add Channel.**

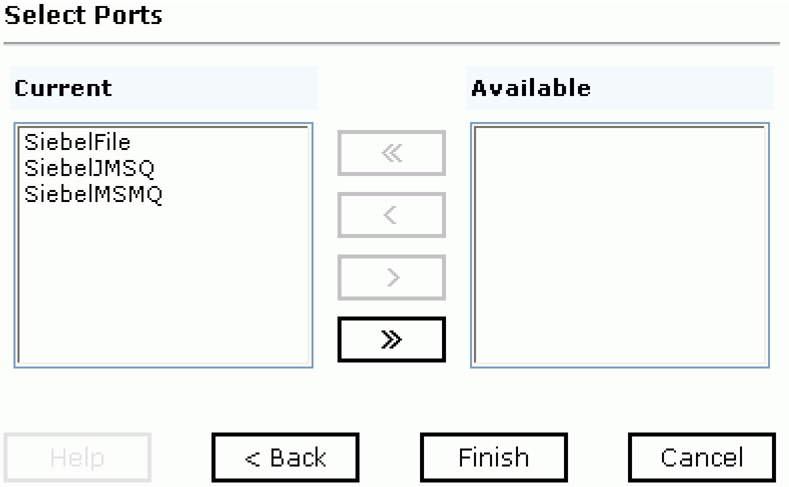
   The Add Channel dialog box opens.

   a. Type a name for the channel, for example, NewChannel.

   b. Type a brief description.

   c. From the list, select **File Listener.**

      The following image shows the Select Ports dialog box.



d. Select an event port from the list of current ports.

e. To transfer the port to the list of available ports, click the double right (>>) arrow button. To associate all the event ports, control-click to select each port or click one port and press **Control+A.** Then, click the double right (>>) arrow button.

4. Click **Next.**

5. When the dialog box opens, type the system information as follows.

a. In the **Request** tab, enter values for the following parameters:

Parameter	Description
Polling Location	The target file system location for the Siebel XML file.
File Mask	The file name to be used for the output file generated as a result of this operation.

b. In the **Response** tab, enter values for the following parameters:

Parameter	Definition
Synchronization Type	Choose from three options: ▪ REQUEST ▪ REQUEST_RESPONSE ▪ REQUEST_ACK
Response/Ack Directory	Directory where responses or acknowledgments are sent.

c. In the **Advanced** tab, enter values for the following parameters:

Parameter	Definition
Error Directory	Directory to which documents with errors are written.
Poll interval (msec)	The interval (in milliseconds) when to check for new input. The default is 3 seconds. Optional.

Parameter	Definition
Processing Mode	Choose Sequential or Threaded. ■ Sequential indicates single processing of requests. ■ Threaded indicates processing of multiple requests simultaneously.
Thread limit	If you selected threaded processing, indicate the maximum number of requests that can be processed simultaneously.

6. Click **OK.**

   The summary pane opens.

   A summary provides the channel description, channel status, and available ports. All the information is associated with the channel you created.

   The channel also appears under the channels node in the left pane

   An X over the icon indicates that the channel is currently disconnected. You must start the channel to activate your event configuration.

7. Right-click the channel and select **Start the channel.**

   The channel you created becomes active.

   The X that was over the icon in the left pane disappears.

8. To stop the channel, right-click the channel and select **Stop the channel.**

Editing a Channel

To edit a channel:

1. In the left pane, select the channel you want to edit.
2. Right-click the channel and select **Edit.**

   The Edit channels pane opens.

3. Make the required changes to the channel configuration and click **OK.**

Deleting a Channel

To delete a channel:

1. In the left pane, select the channel you want to delete.
2. Right-click the channel and select **Delete.**

   A confirmation dialog box opens.

3. To delete the channel you selected, click **OK.**

   The channel disappears from the list in the left pane.

3

Deployment and Integration

This chapter describes Oracle Containers for J2EE (OC4J) deployment and integration with OracleAS Integration InterConnect.

This chapter discusses the following topics:

- OC4J Integration
- OracleAS Adapter BSE Integration with OracleAS Integration InterConnect

See Also:

- *Oracle Application Server Integration InterConnect User's Guide*
- *Oracle Application Server Containers for J2EE User's Guide*

OC4J Integration

The following topic shows the basic commands for using CCI with packaged application adapters.

See Also:

- "OC4J Containers" in *Oracle Application Server Adapter Concepts*
- "Deployment and Integration through J2CA" in *Oracle Application Server Adapter Concepts*

Application Development Using the CCI API

The following example shows the code structure for using CCI with packaged application adapters. The code sample is shown in five steps.

Step 1. Obtain the Connection Factory

The connection factory is obtained by JNDI lookup.

```
InitialContext context = new InitialContext();
ConnectionFactory cf = (ConnectionFactory)context.lookup(iwayJndi)
```

Step 2. Obtaining a Connection for the Adapter

IWAFConnectionSpec is an implementation of ConnectionSpec used for creating a design time or runtime service adapter connection. The ConnectionSpec has seven parameters. Connection Pooling is fully supported and established based on these parameters, except log level.

Parameter Name	Description
adapterName	Name of the packaged application adapter.
config -	Adapter configuration name. NOT REQUIRED FOR IWAEAdapter.
language	Default is en.
country	Default is us.
userName	User name. If provided, it overwrites configuration.
password	Password. If provided, it overwrites configuration.
logLevel	It overwrites the level set by the ManagedConnectionFactory property.

Note: Currently the OracleAS Adapter JCA supports only basic security mapping. The DEBUG log level provides detailed information on the mapping behavior. It functions as follows:

- If the userName and password are not set, and no security is provided by the application server, the OracleAS Adapter JCA will still let it pass and rely on the adapter configuration security information.
- If userName and password are set, these values will overwrite the adapter configuration. The OracleAS Adapter JCA compares this information with the security information provided by the application server and log in case the values do not match. However, it still allows the information through.

The iWAFConnectionSpec can be made to invoke an interaction with Siebel by specifying the adapter name and configuration parameters in the ConnectionSpec. For example,

```
iWAFConnectionSpec cs = new IWAFConnectionSpec();
  cs.setAdapterName(ADAPTER);
  cs.setConfig(TARGET);
  cs.setLogLevel(LOG_LEVEL);  // Adapter layer log level
  Connection c = cf.getConnection(cs);// where cf is the connection factory
```

In this snippet, `ADAPTER` and `TARGET` refer to the adapter being invoked, in this case Siebel, and the name of a target defined in Application Explorer. See"Complete Code Sample" on page 3-3 for more information.

Step 3. Create interaction with interactionSpec for runtime

```
Interaction i = c.createInteraction();
  IWAFInteractionSpec is = new IWAFInteractionSpec();
  is.setFunctionName(IWAFInteractionSpec.PROCESS);
```

Two functions can be set: PROCESS and IWAE. PROCESS is used at runtime. IWAE is used when you are using the IAEAdapter at design time.

Step 4. Create Input Record and Execute Interaction

In this case, to complete the EIS invocation, a Siebel message is referenced. The schema is provided by Application Explorer.

A standard JCA Indexed Record is used in this example:

```
// Use JCA IndexRecord, named "input" for runtime processing.
IndexedRecord rIn = cf.getRecordFactory().createIndexedRecord("input");
rIn.add(msg_run);
  IndexedRecord rOut = (IndexedRecord)i.execute(is, rIn);
System.out.println((String)rOut.get(0));
```

A special record is supported in this example:

```
//IWAFRecord rIn = new IWAFRecord("input");
//rIn.setRootXML(msg_run);
//IWAFRecord response = executeRunInteraction(c, rIn);
  //IWAFRecord rOut = (IWAFRecord)i.execute(is, rIn);
//System.out.println(rOut.getRootXML());
```

msg_run

Is an instance XML document generated from the schema created by Application Explorer.

Complete Code Sample

The following is a sample of the complete code:

```
import javax.resource.cci.*;
import com.ibi.afjca.cci.*;
import com.ibi.afjca.spi.*;

/**
 * The purpose of this sample is to illustrate how to use the IWAF Universal
 * JCA connector.
 *
 * Author: Marcelo Borges
 * Date: August, 2004
 */
public class IWAFJCASimple {

private static String  HOME    = "c:/iway/xfoc/components/iwafcont/dist";
private static String  CONFIG    = "base";
private static String  LOG_LEVEL = "FATAL";

private static String ADAPTER = "Siebel";
private static String TARGET  = "Siebel_connection";

// Input Message
private static String msg_run = "<Siebel/>";

  public static void main(String[] args) throws Exception {

// 1. Getting the Connection factory through JNDI lookup
// ------------------------------------------------------------
  InitialContext context = new InitialContext();
  ConnectionFactory cf = (ConnectionFactory)context.lookup(iwayJndi)
  // 2. Getting a connection for a particular adapter target, in this case Siebel
// ------------------------------------------------------------
  IWAFConnectionSpec cs = new IWAFConnectionSpec();
  cs.setAdapterName(ADAPTER);
  cs.setConfig(TARGET);
  cs.setLogLevel(LOG_LEVEL);  // Adapter layer log level
  Connection c = cf.getConnection(cs);// where cf is the connection factory

  // 3. Create interaction with interactionSpec for RUNTIME
// -------------------------------------------------------
  Interaction i = c.createInteraction();
  IWAFInteractionSpec is = new IWAFInteractionSpec();
  is.setFunctionName("PROCESS");

  // 4. Create input Record and execute interaction
// -------------------------------------------------------

  // 4.1 Using JCA standard Indexed Record
// Use JCA IndexRecord, named "input" for runtime processing.
IndexedRecord rIn = cf.getRecordFactory().createIndexedRecord("input");
rIn.add(msg_run);
  IndexedRecord rOut = (IndexedRecord)i.execute(is, rIn);
System.out.println((String)rOut.get(0));

  // 4.2 Our own Record is supported here
//IWAFRecord rIn = new IWAFRecord("input");
//rIn.setRootXML(msg_run);
//IWAFRecord response = executeRunInteraction(c, rIn);
  //IWAFRecord rOut = (IWAFRecord)i.execute(is, rIn);
//System.out.println(rOut.getRootXML());

  } // main()

}
```

Creating a Managed Connection Factory

The `OC4J-ra.xml` descriptor provides OC4J-specific deployment information for resource adapters. For example, the default jca_sample configuration in Application Explorer is represented in the `OC4J-ra.xml` file as follows:

```
<?xml version="1.0"?>
<!DOCTYPE oc4j-connector-factories PUBLIC "-//Oracle//DTD Oracle Connector
9.04//EN" "http://xmlns.oracle.com/ias/dtds/oc4j-connector-factories-9_04.dtd">
<oc4j-connector-factories>
  <connector-factory location="eis/OracleJCAAdapter/DefaultConnection"
connector-name="IWAFJCA10">
  <config-property name="IWayHome" value="../../adapters/application"/>
  <config-property name="IWayConfig" value="jca_sample"/>
   <config-property name="IWayRepoURL" value=""/>
   <config-property name="IWayRepoUser" value=""/>
   <config-property name="IWayRepoPassword" value=""/>
   <config-property name="logLevel" value="debug"/>
   </connector-factory>
</oc4j-connector-factories>
```

The parameters are defined in the following:

Parameter Name	Description
IWayHome	The base installation directory for the OracleAS packaged application adapter.
IWayConfig	The adapter configuration name as defined in Application Explorer. For example, the OracleAS Adapter for PeopleSoft has a preconfigured jca_sample configuration in the Application Explorer.

Parameter Name	Description
IWayRepoURL	The URL to use when opening a connection to the database. This is necessary only when using an Oracle database as the BSE repository. See"Configuring BSE System Settings" in on page 2-3 for more information.
IWayRepoUser	User name to use when connecting to the database. This is necessary only when using an Oracle database as the BSE repository. See"Configuring BSE System Settings" in on page 2-3 for more information.
IWayRepoPassword	Password. If provided, it overwrites configuration. This is necessary only when using an Oracle database as the BSE repository. See"Configuring BSE System Settings" in on page 2-3 for more information.
loglevel	It overwrites the level set by the ManagedConnectionFactory property.

Creating Multiple Managed Connection Factories

To establish Multiple Managed Connection Factories, you must edit the `OC4J-ra.xml` file to add the required information. The file can contain more than one `<connector-factory>` element. By adding more `<connector-factory>` elements, you can create Multiple Managed connection factories. For example, the default jca_sample configuration in Application Explorer is represented in the `OC4J-ra.xml` file as follows:

```
<?xml version="1.0"?>
<!DOCTYPE oc4j-connector-factories PUBLIC "-//Oracle//DTD Oracle Connector
9.04//EN" "http://xmlns.oracle.com/ias/dtds/oc4j-connector-factories-9_04.dtd">
<oc4j-connector-factories>
   <connector-factory location="eis/OracleJCAAdapter/DefaultConnection"
connector-name="IWAFJCA10">
   <config-property name="IWayHome" value="../../adapters/application"/>
   <config-property name="IWayConfig" value="jca_sample"/>
   <config-property name="IWayRepoURL" value=""/>
   <config-property name="IWayRepoUser" value=""/>
   <config-property name="IWayRepoPassword" value=""/>
   <config-property name="logLevel" value="debug"/>
   </connector-factory>
</oc4j-connector-factories>
```

To create Multiple Managed Connection Factories, you must add new <connector-factory> nodes in the file. For example:

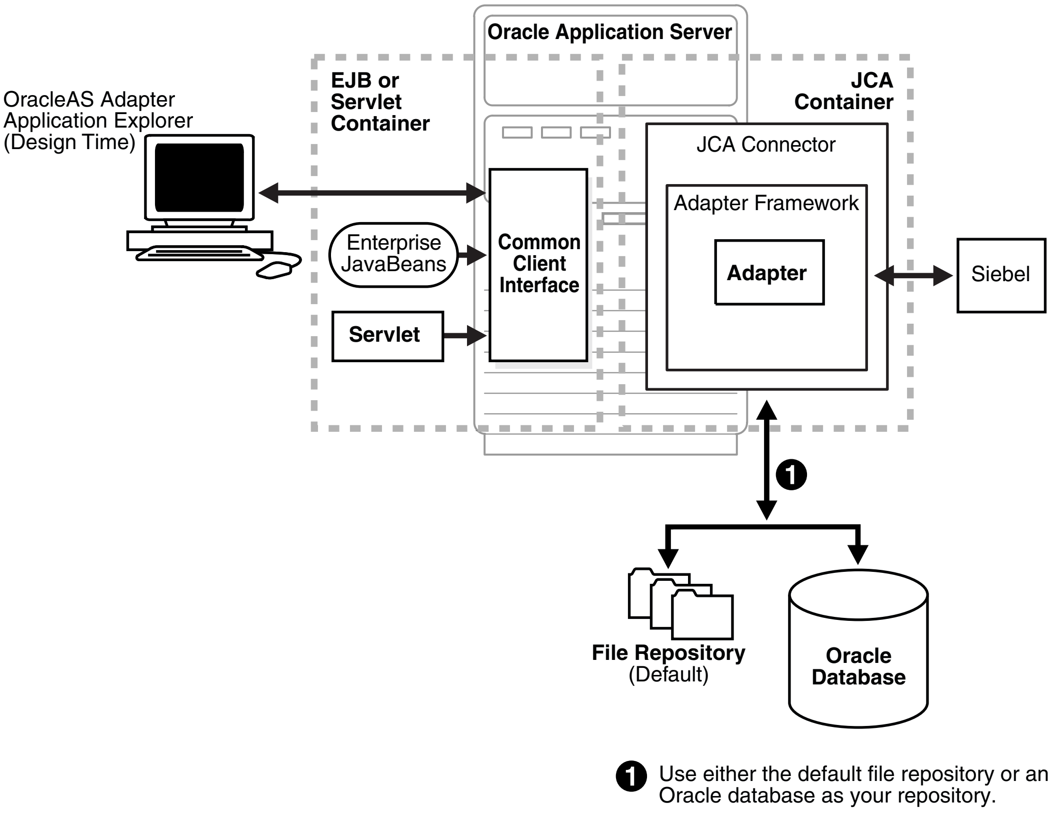
```
<?xml version="1.0"?>
<!DOCTYPE oc4j-connector-factories PUBLIC "-//Oracle//DTD Oracle Connector
9.04//EN" "http://xmlns.oracle.com/ias/dtds/oc4j-connector-factories-9_04.dtd">
<oc4j-connector-factories>
   <connector-factory location="eis/OracleJCAAdapter/DefaultConnection1"
connector-name="IWAFJCA10">
   <config-property name="IWayHome" value="../../adapters/application"/>
   <config-property name="IWayConfig" value="jca_sample"/>
   <config-property name="IWayRepoURL" value=""/>
   <config-property name="IWayRepoUser" value=""/>
   <config-property name="IWayRepoPassword" value=""/>
   <config-property name="logLevel" value="debug"/>
   </connector-factory>
```

```
<connector-factory location="eis/OracleJCAAdapter/DefaultConnection2"
connector-name="IWAFJCA10">
   <config-property name="IWayHome" value="../../adapters/application"/>
   <config-property name="IWayConfig" value="jca_sample2"/>
   <config-property name="IWayRepoURL" value=""/>
   <config-property name="IWayRepoUser" value=""/>
   <config-property name="IWayRepoPassword" value=""/>
   <config-property name="logLevel" value="debug"/>
   </connector-factory>
</oc4j-connector-factories>
```

Oracle Application Server Adapter JCA Architecture

Figure 3–1 shows deployment of the Connector to the Oracle Application Server. In a runtime service scenario, an EJB, Servlet, or Java program client makes CCI calls to JCA resource adapters. The adapters process the calls as requests and send them to the EIS. The EIS response is then sent back to the client.

Figure 3–1 Oracle Application Server Adapter JCA Architecture



OracleAS Adapter BSE Integration with OracleAS Integration InterConnect

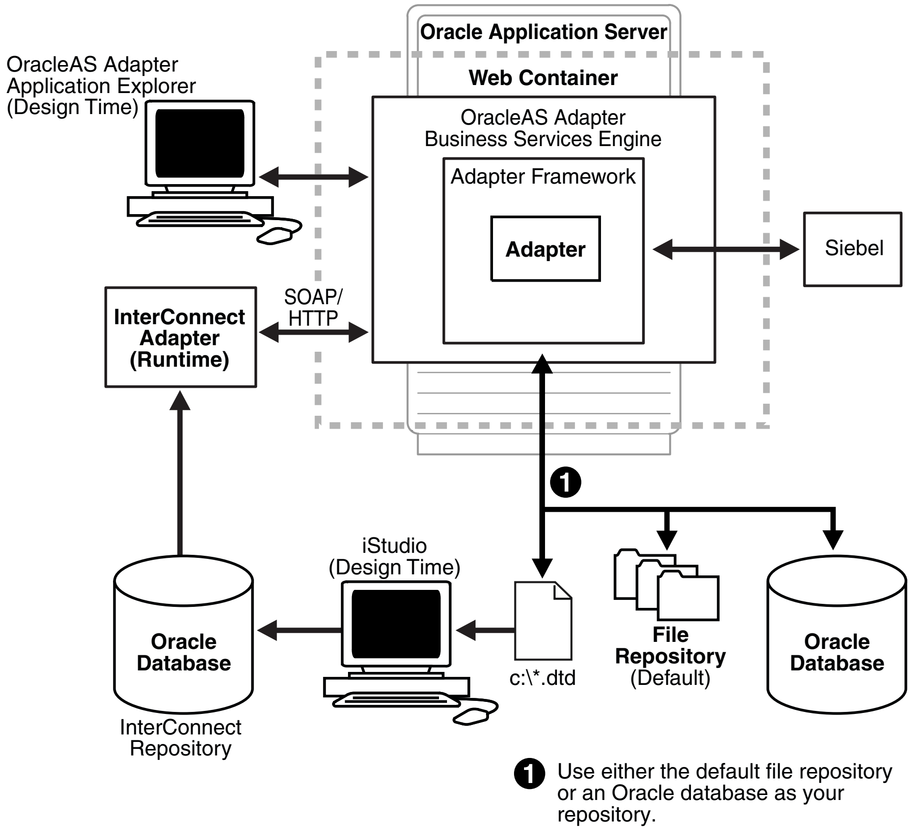
See Also: "Deployment and Integration through OracleAS Web Services" in *Oracle Application Server Adapter Concepts*

BSE Architecture as Deployed to Oracle Application Server

Figure 3–2 shows adapter framework deployment with BSE to OracleAS Integration InterConnect. In a runtime service scenario, the OracleAS Integration InterConnect EIS Adapter Plugin (EIS Adapter Plugin) receives DTD-compliant XML from the **agent** component of the EIS Adapter Plugin. The EIS Adapter Plugin strips runtime information from the XML, wraps the XML in a SOAP envelope, and sends the result to BSE, including the runtime information in the SOAP request. BSE receives the request, removes the envelope, retrieves Web service method metadata, including adapter and connection information from the repository, and makes the adapter request.

BSE receives the adapter response, wraps the response XML in a SOAP envelope, and returns it to the EIS Adapter Plugin. The EIS Adapter Plugin then strips the SOAP envelope, strips the namespace prefix, if present, and passes the DTD-compliant XML to the **agent** component of the EIS Adapter Plugin.

Figure 3–2 BSE Architecture as Deployed to Oracle Application Server


Oracle Application Server
Web Container
OracleAS Adapter Application Explorer (Design Time)
OracleAS Adapter Business Services Engine
Adapter Framework
Adapter
Siebel
SOAP/ HTTP
InterConnect Adapter (Runtime)
1
iStudio (Design Time)
Oracle Database
InterConnect Repository
c:\*.dtd
File Repository (Default)
Oracle Database

1 Use either the default file repository or an Oracle database as your repository.

Upon installation of the Oracle Web Services Adapter, an `adapter.ini` file is created. The file consists of all the initialization parameters that the adapter reads at startup. Some of the parameters in this file are configurable.

See Also: *Oracle Application Server Adapters Installation Guide*

4 Examples

This chapter contains examples of service and event integration with Siebel.

- Siebel Service Integration
- Creating Events
- Siebel Event Integration

The service and event scenarios shown depend on the following prerequisites and configuration steps:

Prerequisites

- OracleAS Adapter for Siebel installed on the Oracle Application Server.
- OracleAS Database adapter deployed and configured.
- OracleAS Integration InterConnect Adapter Plugin for EIS installed and running.

> **See Also:** *Oracle Application Server Adapters Installation Guide*

Configuration Steps

The examples present all the configuration steps necessary for demonstrating service integration with Siebel. The following cross references identify where more information can be obtained.

1. Configure the OracleAS Adapter for Siebel for services. See Chapter 2, "Adapter Configuration Using OracleAS Adapter Application Explorer" for more information.
2. Configure OracleAS Integration InterConnect iStudio for service interactions. For more information, see the following service and event steps.

Siebel Service Integration

This topic illustrates Siebel service integration. The procedures describe design time and runtime.

OracleAS Integration InterConnect Design Time

The following procedures describe how to start the repository and create a common view and then, define invoked and implemented procedures. Then, it describes how to export PL/SQL code from iStudio.

Starting the Repository

To start the repository, double-click the `start.bat` file located in the following directory:

```
D:\oracle\ora92InterCon\oai\9.0.4\repository\start.bat
```

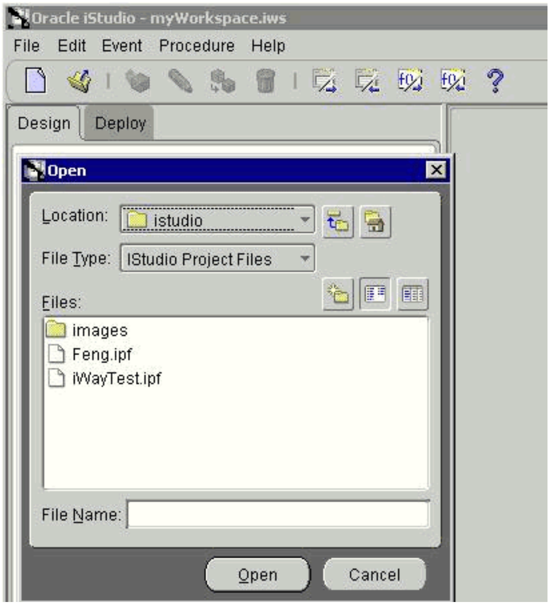
Creating a Common View

To create a Common View:

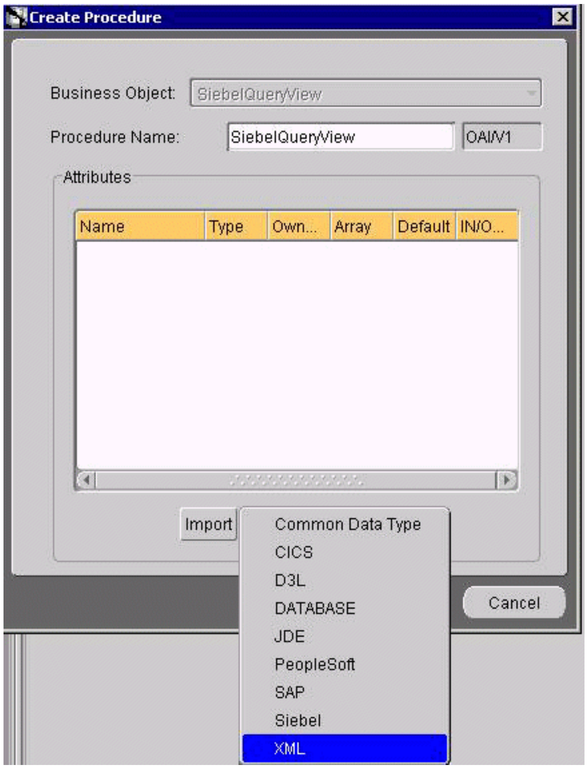
1. Start Oracle iStudio by double-clicking the `start.bat` file located in the following directory:

   ```
   D:\oracle\ora92iStudio\oai\9.0.4\istudio\iStudio.bat
   ```

   iStudio opens.



2. Open a project.
3. Open **Common Views** and **Business Objects.**
4. Create a Business Object called **SiebelQueryView.**



Create Procedure

Business Object: SiebelQueryView

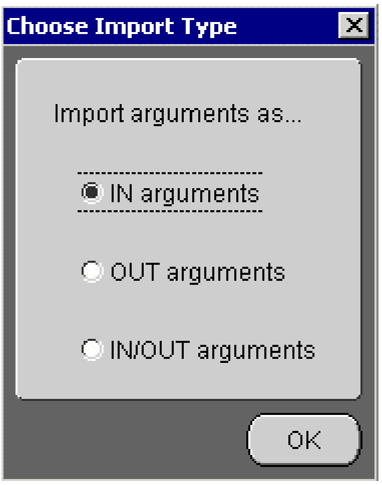
Procedure Name: SiebelQueryView OAI/V1

Attributes

Name	Type	Own...	Array	Default	IN/O...

5. Create a new procedure under `SiebelQueryView` and type `SiebelQueryView` as the procedure name.
6. Open the DTD generated from Application Explorer and load it.

   The Choose Root Element dialog box opens.
7. Select the root element, `Siebel`, in this example.
8. Click **OK.**



Choose Import Type

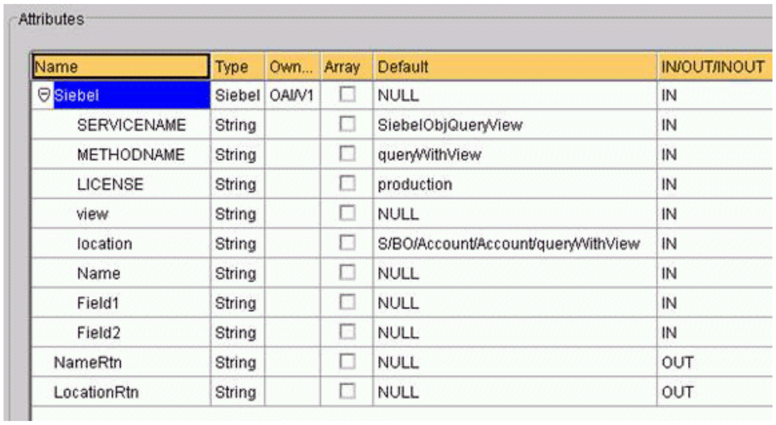
Import arguments as...

● IN arguments

○ OUT arguments

○ IN/OUT arguments

9. Select **IN arguments** as the import type for the request DTD and click **OK.**
10. Manually enter all the OUT parameters as shown in the following image.
11. Select **OUT arguments** as the import type for the response DTD and click **OK.**



Attributes

Name	Type	Own...	Array	Default	IN/OUT/INOUT
Siebel	Siebel	OAI/V1	□	NULL	IN
SERVICENAME	String		□	SiebelObjQueryView	IN
METHODNAME	String		□	queryWithView	IN
LICENSE	String		□	production	IN
view	String		□	NULL	IN
location	String		□	S/BO/Account/Account/queryWithView	IN
Name	String		□	NULL	IN
Field1	String		□	NULL	IN
Field2	String		□	NULL	IN
NameRtn	String		□	NULL	OUT
LocationRtn	String		□	NULL	OUT

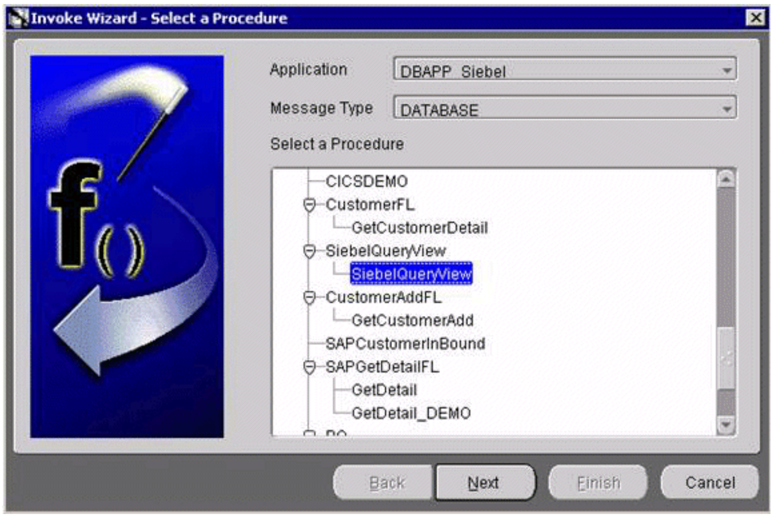
12. To save the new procedure, click **Save.**

Creating an Invoked Procedure

To create an invoked procedure:

1. Create a new application called **DBAPP_Siebel.**
2. Right-click **Invoked Procedures** and select **New.**

   The Invoke Wizard - Select a Procedure window opens.



Invoke Wizard - Select a Procedure

Application: DBAPP_Siebel

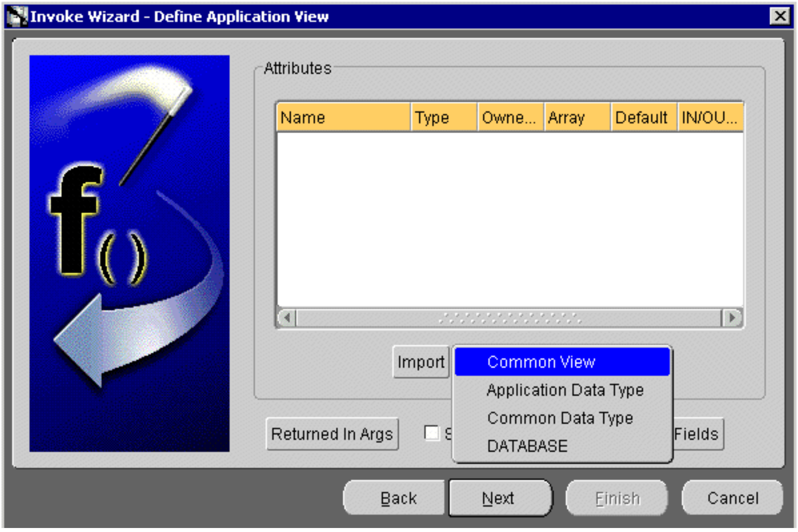
Message Type: DATABASE

Select a Procedure

   a. From the **Message Type** list, select **DATABASE.**
   b. Expand the `SiebelQueryView` business object as the event and select **SiebelQueryView.**

3. Click **Next.**

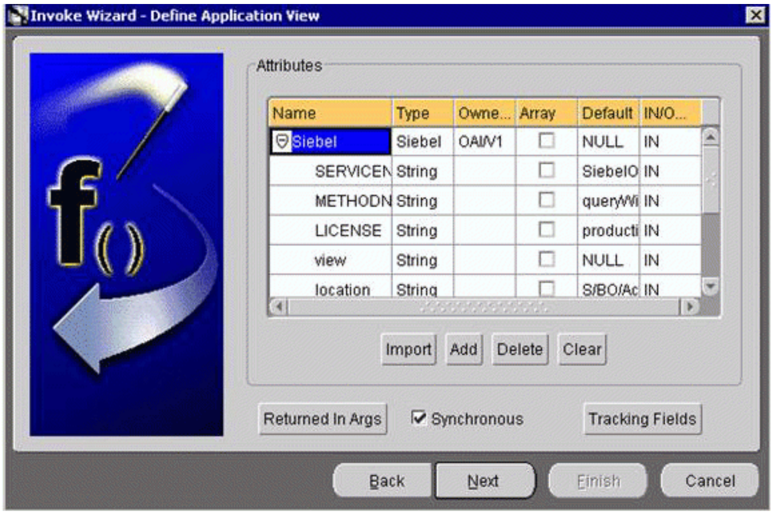
   The Invoke Wizard - Define Application View window opens.



a. Click **Import.**

b. Select **Common View.**

Information appears in the right pane.



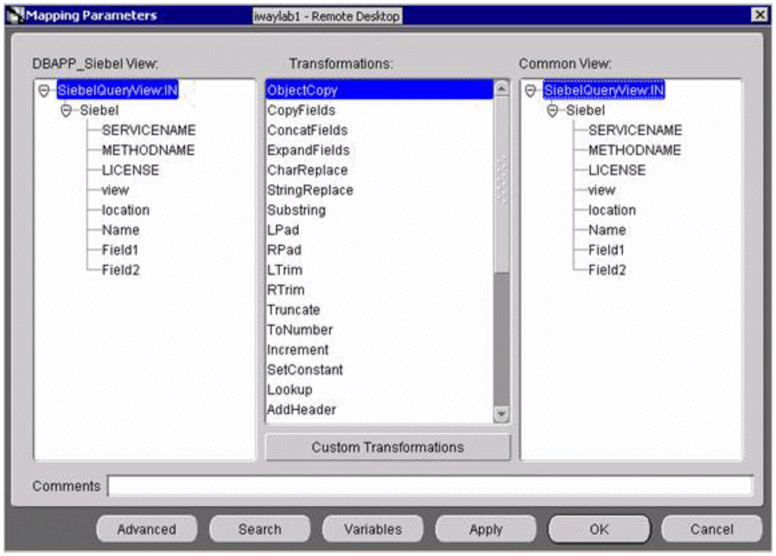
Name	Type	Owne...	Array	Default	IN/O...
Siebel	Siebel	OAI/V1	□	NULL	IN
SERVICEN	String		□	SiebelO	IN
METHODN	String		□	queryWi	IN
LICENSE	String		□	producti	IN
view	String		□	NULL	IN
location	String		□	S/BO/Ac	IN

☑ Synchronous

c. Select the **Synchronous** check box, as this is a request and a response.

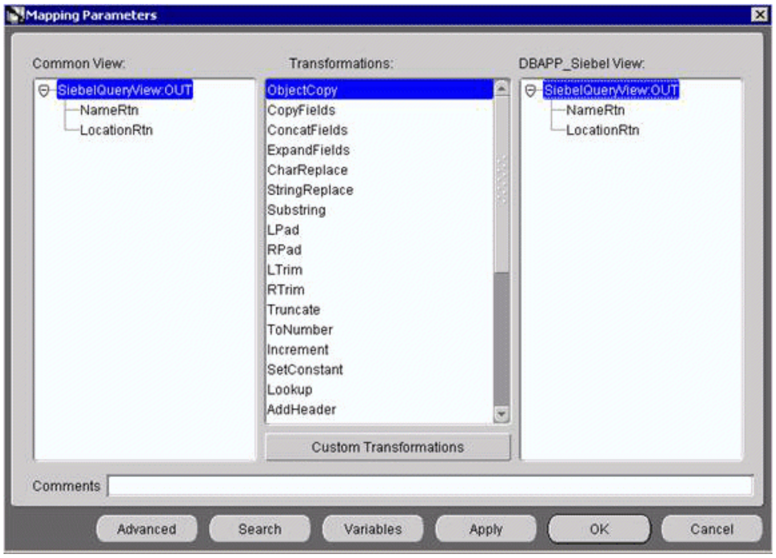
4. Click **Next.**

5. Click **New** to create a mapping between the Common View and the Application View for the IN parameters.



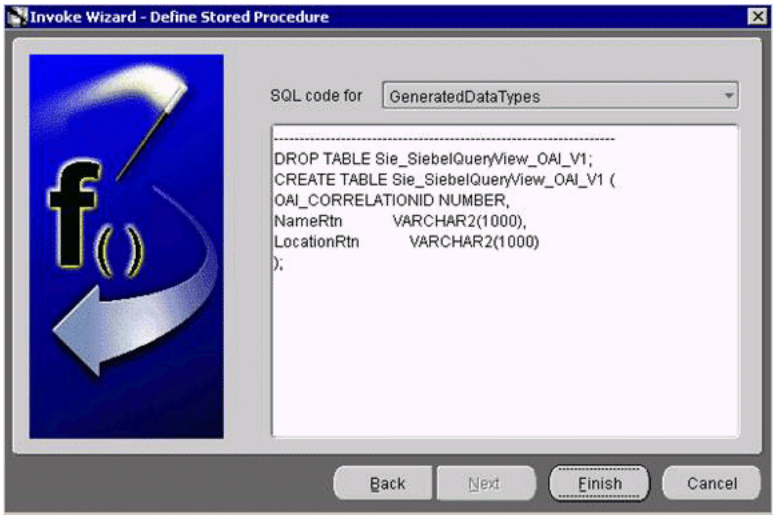
In this example, the Application View and the Common View have the same structure. All the attributes can be mapped by using ObjectCopy Transformation.

6. Click **Apply** and then, **OK.**
7. Click **Next.**
8. Use ObjectCopy Transformation to map the attributes between Application View and Common View for the OUT parameters.



9. Click **Finish.**

   iStudio generates SQL code.



SQL code for: GeneratedDataTypes

```
-----------------------------------------------------------------
DROP TABLE Sie_SiebelQueryView_OAI_V1;
CREATE TABLE Sie_SiebelQueryView_OAI_V1 (
OAI_CORRELATIONID NUMBER,
NameRtn          VARCHAR2(1000),
LocationRtn         VARCHAR2(1000)
);
```

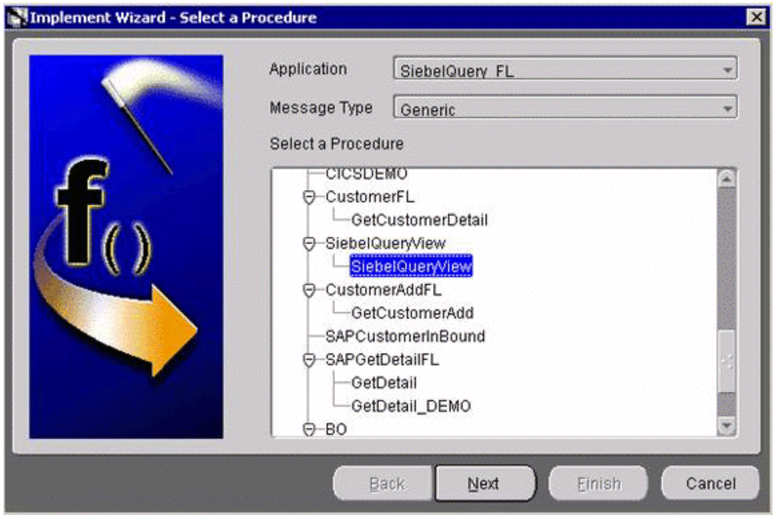
The application definition for the invoked procedure is now complete.

Defining an Implemented Procedure

To define an implemented procedure:

1. Create a new application called `SiebelQuery_FL`.
2. Expand **SiebelQueryView.**
3. Right-click **Implemented Procedure** and select **New.**

   The Implement Wizard - Select a Procedure window opens.



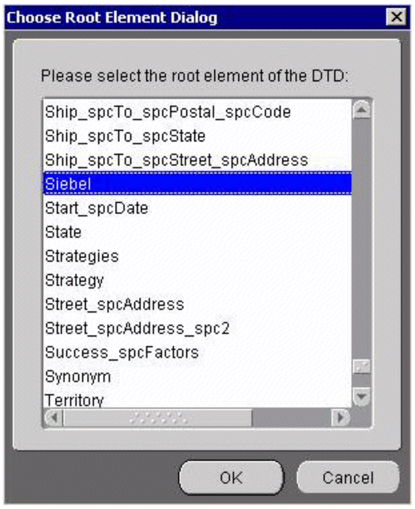
Application: SiebelQuery_FL

Message Type: Generic

   a. From the **Message Type** list, select **Generic.**
   b. Expand the `SiebelQueryView` business object and select **SiebelQueryView** as the procedure.

4. Click **Next.**
   a. Click **Import.**
   b. Select **XML.**
   c. Select the request and response DTDs generated by Application Explorer.

5. Import the request and response DTDs into iStudio.



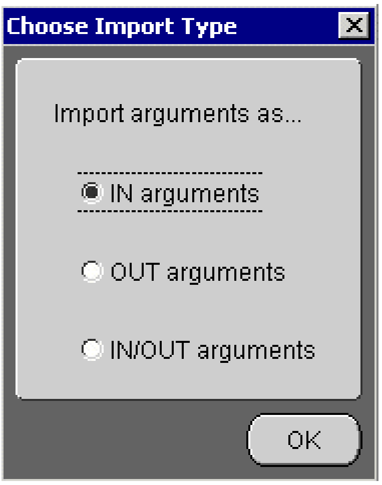
Choose Root Element Dialog

Please select the root element of the DTD:

Ship_spcTo_spcPostal_spcCode
Ship_spcTo_spcState
Ship_spcTo_spcStreet_spcAddress
Siebel
Start_spcDate
State
Strategies
Strategy
Street_spcAddress
Street_spcAddress_spc2
Success_spcFactors
Synonym
Territory

6. Select **Siebel** as the root element of the request DTD.

   The Choose Import Type dialog box opens.



Choose Import Type

Import arguments as...

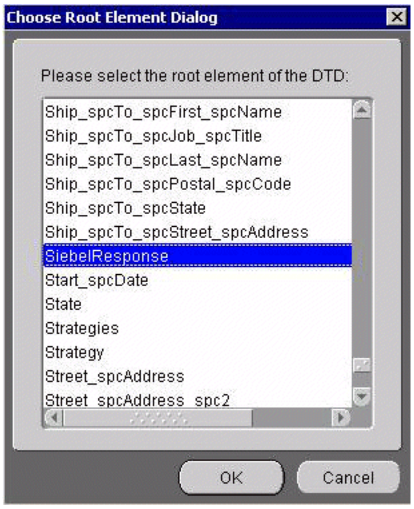
● IN arguments
○ OUT arguments
○ IN/OUT arguments

7. Select **IN arguments** as the import type for the request DTD and click **OK.**

   The Implement Wizard - Define Application View window opens.

8. Type the root element of the request DTD in the Object Name field, if it is not automatically populated after importing the request DTD and click **Import.**

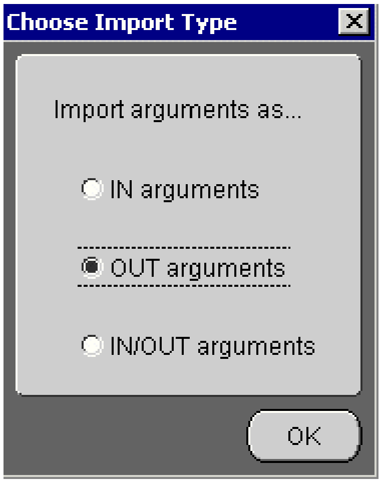
   The Choose Root Element dialog box opens.



Please select the root element of the DTD:

- Ship_spcTo_spcFirst_spcName
- Ship_spcTo_spcJob_spcTitle
- Ship_spcTo_spcLast_spcName
- Ship_spcTo_spcPostal_spcCode
- Ship_spcTo_spcState
- Ship_spcTo_spcStreet_spcAddress
- SiebelResponse
- Start_spcDate
- State
- Strategies
- Strategy
- Street_spcAddress
- Street_spcAddress_spc2

9. Select **SiebelResponse** as the root element and click **OK.**



Import arguments as...

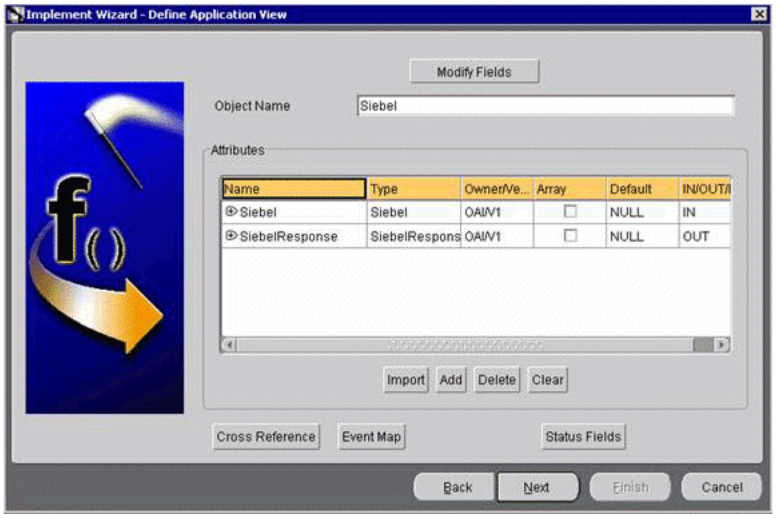
○ IN arguments

● OUT arguments

○ IN/OUT arguments

10. Select **OUT arguments** as the import type for the response DTD and click **OK.**

    Both the request and response DTDs are now imported into iStudio.

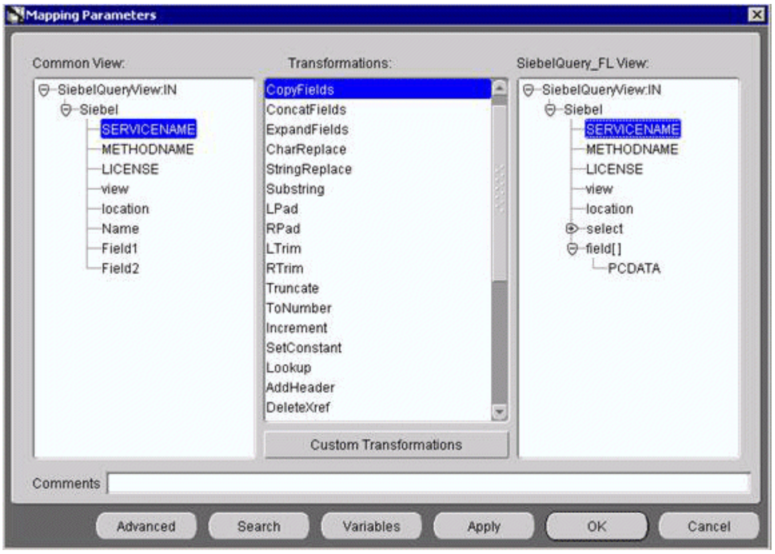


Object Name: Siebel

Attributes

Name	Type	Owner/Ve...	Array	Default	IN/OUT/I
⊕Siebel	Siebel	OAI/V1	□	NULL	IN
⊕SiebelResponse	SiebelRespons	OAI/V1	□	NULL	OUT

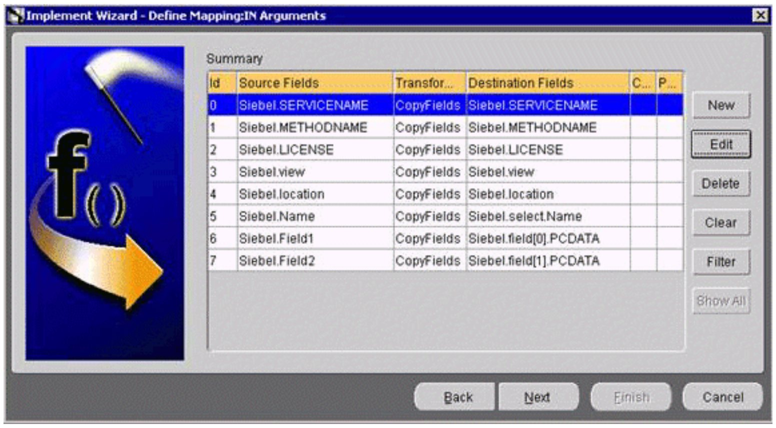
11. To define a mapping between the Application View and the Common View, click **Next** and then, **New.**



12. Use the **CopyFields** transformation to map each field.

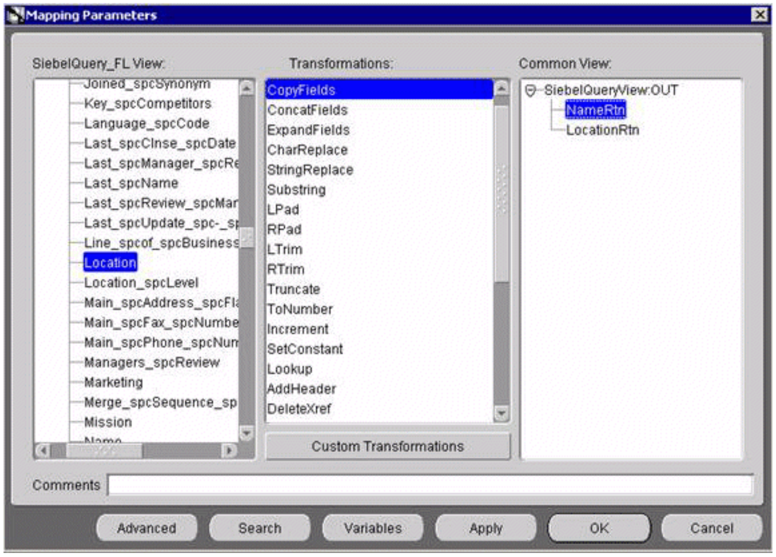
13. Click **Apply** and then **OK.**

When complete, your mapping should look similar to the following example.



Id	Source Fields	Transfor...	Destination Fields	C...	P...
0	Siebel.SERVICENAME	CopyFields	Siebel.SERVICENAME		
1	Siebel.METHODNAME	CopyFields	Siebel.METHODNAME		
2	Siebel.LICENSE	CopyFields	Siebel.LICENSE		
3	Siebel.view	CopyFields	Siebel.view		
4	Siebel.location	CopyFields	Siebel.location		
5	Siebel.Name	CopyFields	Siebel.select.Name		
6	Siebel.Field1	CopyFields	Siebel.field[0].PCDATA		
7	Siebel.Field2	CopyFields	Siebel.field[1].PCDATA		

14. To define the mapping for the OUT arguments, click **Next.**

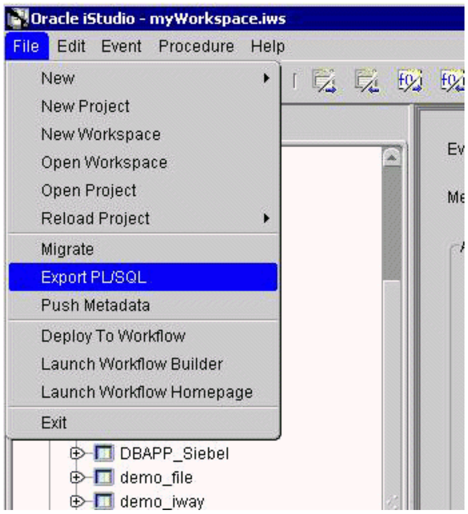


15. To complete the definition of the implemented procedure, click **Next** and then, **Finish.**

Exporting PL/SQL Code from iStudio

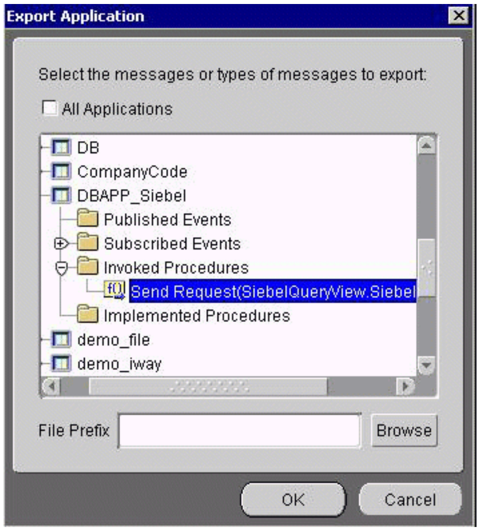
You must export the PL/SQL code created in "Defining an Implemented Procedure" on page 4-7 and execute it against the appropriate schema. In this example, the schema used is DBAPP_Siebel.

To export PL/SQL code from iStudio:



1. In iStudio, click **File** and **Export PL/SQL.**

The Export Application dialog box opens.



a. Select the application from which to export PL/SQL.

b. Type or browse to the file prefix (path to the application).

2. Click **OK.**

   In this example, two SQL scripts are created:

   - `DBAPP_Siebel_SiebelQueryViewTYPES.sql`
   - `DBAPP_Siebel_SiebelQueryView.sql`

3. Log on to the database with the appropriate privileges (in this example, DBAPP_Siebel) and execute the following in the order given:

   - `DBAPP_Siebel_SiebelQueryViewTYPES.sql`
   - `DBAPP_Siebel_SiebelQueryView.sql`

4. Create another stored procedure, `SIEBELQUERYVIEW_EXE`, in the same schema. It executes at runtime to create the database message that is sent to the hub.

```
CREATE OR REPLACE  PROCEDURE "DBAPP_SIEBEL"."SIEBELQUERYVIEW_EXE"
    (
servicename LONG,
methodname  LONG,
license     LONG,
viewmode    LONG,
location    LONG,
name        LONG,
field1      LONG,
field2      LONG
)
AS
  moid NUMBER;
  aoid NUMBER;
  coid NUMBER;
  siebelid NUMBER;
  namertn LONG;
  locationrtn LONG;
BEGIN
  SiebelQueryView.crMsg_SiebelQueryView_OAI_V1(moid, aoid);
  siebelid := SiebelQueryView.cr_Siebel_
Siebel(servicename,methodname,license,viewmode,location,name,field1,field2,moid,ao
id);
  coid := SiebelQueryView.inv_SiebelQueryView_OAI_V1(moid,'DBAPP_
Siebel','',namertn,locationrtn);
  insert into customer
  (customerid,address)
  values
  (namertn,locationrtn);
  COMMIT;
END;
```

Editing the adapter.ini File

To edit the adapter.ini file:

1. Open the `adapter.ini` file.
2. Add the following two lines to `adapter.ini` for the adapter:

```
// Bridge class
bridge_class=com.iwaysoftware.iwbridge.IWBridge

// IBSE URL
ibse_url=http://lee.ibi.com:7777/ibse/IBSEServlet/XDSOAPRouter
```

lee.ibi.com
Is the URL of the server.

7777
Is the port number.

OracleAS Integration InterConnect Runtime

The following topic describes how to verify service integration using the OracleAS Adapter for Siebel.

Verifying Service Integration

To verify service integration:

1. Start the Oracle Application Server or ensure that the server is running.
2. Restart OC4J, if required, by executing the following command:

```
\OracleAS_home\opmn\bin\opmnctl  stopproc process-type=home
\OracleAS_home\opmn\bin\opmnctl  startproc process-type=home
```

3. Check the status of OC4J by executing the following command:

```
\OracleAS_home\opmn\bin\opmnctl  status
```

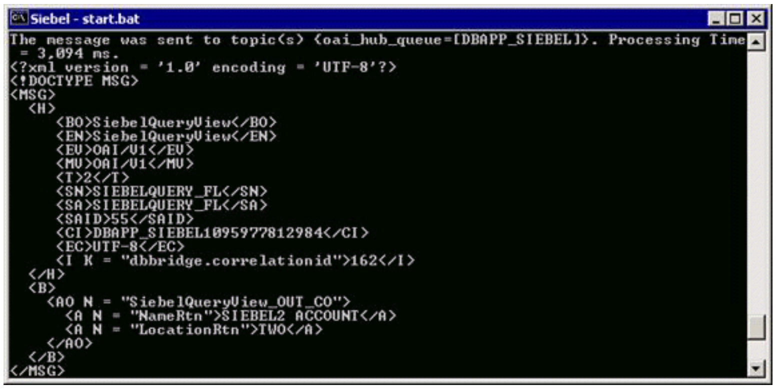
4. Invoke and implement the adapter by executing the following commands:

```
\InterConnect_HOME\oai\9.0.4\adapters\ SiebelQuery_FL\start.bat
\InterConnect_HOME\oai\9.0.4\adapters\DBAPP_Siebel\start.bat
```

5. Log on to SQL*Plus with `DBAPP_Siebel` and execute the following command:

```
exec
siebelqueryview_
exe('SiebelObjQueryView','queryWithView','production','AllView',
'S/BO/Account/Account/queryWithView','SIEBEL2 ACCOUNT','Location','Name');
```

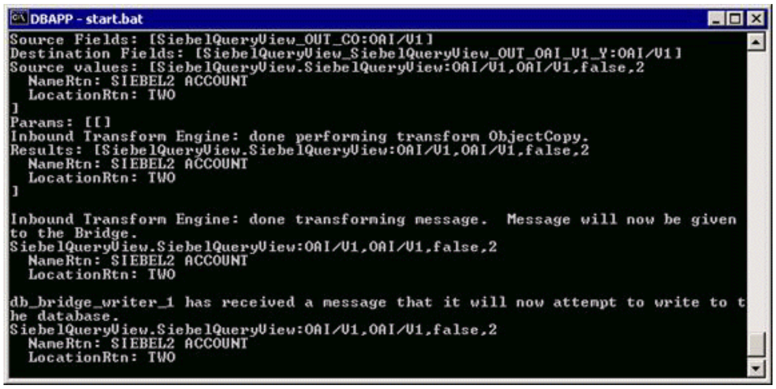
The following image shows the SiebelQuery_FL example. It receives a reply from Siebel and returns the reply to the hub.



Siebel - start.bat

```
The message was sent to topic(s) {oai_hub_queue=[DBAPP_SIEBEL]}. Processing Time
 = 3,094 ms.
<?xml version = '1.0' encoding = 'UTF-8'?>
<!DOCTYPE MSG>
<MSG>
  <H>
     <BO>SiebelQueryView</BO>
     <EN>SiebelQueryView</EN>
     <EV>OAI/V1</EV>
     <MV>OAI/V1</MV>
     <T>2</T>
     <SN>SIEBELQUERY_FL</SN>
     <SA>SIEBELQUERY_FL</SA>
     <SAID>55</SAID>
     <CI>DBAPP_SIEBEL1095977812984</CI>
     <EC>UTF-8</EC>
     <I K = "dbbridge.correlationid">162</I>
  </H>
  <B>
    <AO N = "SiebelQueryView_OUT_CO">
      <A N = "NameRtn">SIEBEL2 ACCOUNT</A>
      <A N = "LocationRtn">TWO</A>
    </AO>
  </B>
</MSG>
```

The following image shows the DBAPP_Siebel example. It receives a reply from the hub and writes the data to the database table.



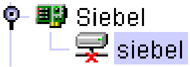
DBAPP - start.bat

```
Source Fields: [SiebelQueryView_OUT_CO:OAI/V1]
Destination Fields: [SiebelQueryView_SiebelQueryView_OUT_OAI_V1_Y:OAI/V1]
Source values: [SiebelQueryView.SiebelQueryView:OAI/V1,OAI/V1,false,2
  NameRtn: SIEBEL2 ACCOUNT
  LocationRtn: TWO
]
Params: [[]
Inbound Transform Engine: done performing transform ObjectCopy.
Results: [SiebelQueryView.SiebelQueryView:OAI/V1,OAI/V1,false,2
  NameRtn: SIEBEL2 ACCOUNT
  LocationRtn: TWO
]

Inbound Transform Engine: done transforming message.  Message will now be given
to the Bridge.
SiebelQueryView.SiebelQueryView:OAI/V1,OAI/V1,false,2
  NameRtn: SIEBEL2 ACCOUNT
  LocationRtn: TWO

db_bridge_writer_1 has received a message that it will now attempt to write to t
he database.
SiebelQueryView.SiebelQueryView:OAI/V1,OAI/V1,false,2
  NameRtn: SIEBEL2 ACCOUNT
  LocationRtn: TWO
```

Creating Events

The following example describes how to create events for Siebel and how to create a channel and a port.

1. Open Application Explorer.
2. Expand the **Adapters** node.



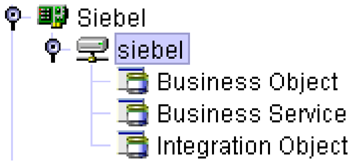
   a. Expand the **Siebel** node.

   b. Click the target name, for example, siebel, under the **Siebel** node.

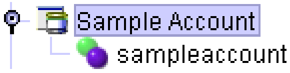
   The Connection dialog box opens, populated with values you entered.

3. Verify your connection parameters. If required, provide the password.
4. Right-click the target name and select **Connect.**

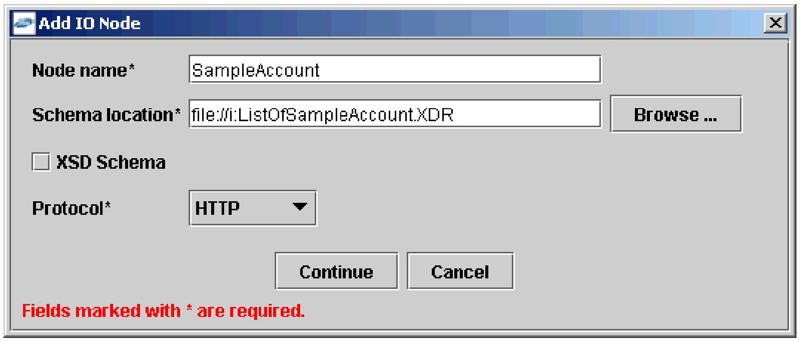
   The x icon disappears, indicating that the node is connected.



5. Expand the **Integration Object** node and select **Sample Account.**



6. Right-click the **Sample Account** node and select **Add IO Node.**


Add IO Node

Node name* SampleAccount

Schema location* file://i:ListOfSampleAccount.XDR

□ XSD Schema

Protocol* HTTP

Fields marked with * are required.

7. Enter a node name, for example SampleAccount in the **Node name** field and a path to the Sample Account XDR file in the **Schema location** field.

8. Select a protocol from the **Protocol** list.

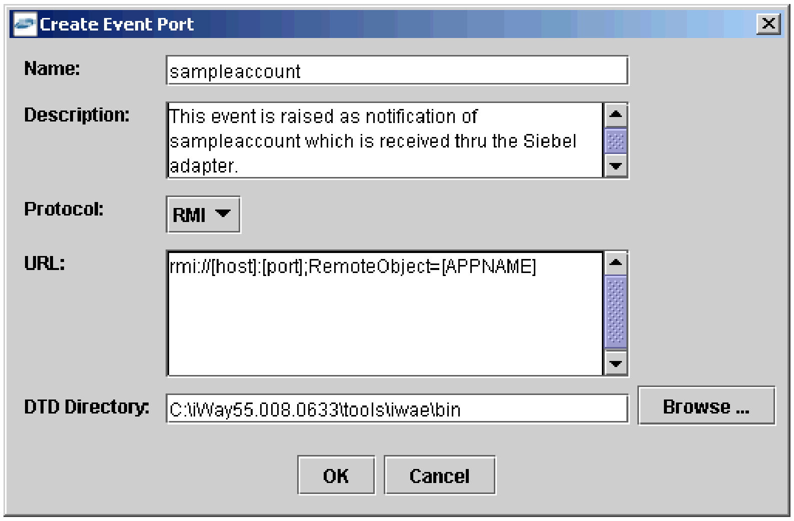
9. Click **Continue.**

Creating Events for Sample Account

To create events for Sample Account:

1. Right-click the **Sample Account** node and select **Create Event Port.**

   The Create Event Port dialog box opens.


Create Event Port

Name: sampleaccount

Description: This event is raised as notification of sampleaccount which is received thru the Siebel adapter.

Protocol: RMI

URL: rmi://[host]:[port];RemoteObject=[APPNAME]

DTD Directory: C:\iWay55.008.0633\tools\iwae\bin

   a. Type a name for the event port and provide a brief description.

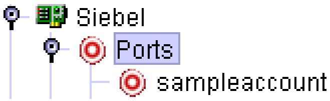
b. From the list, select the required disposition, for example, RMI.

c. Type the disposition URL.

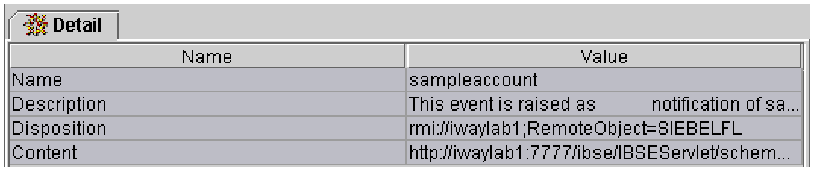
d. Type (or browse to) the path containing the DTD directory.

2. Click **OK.**

   The port appears under the ports node in the left pane.



   In the right pane, a table appears that summarizes the information associated with the event port you created.



Name	Value
Name	sampleaccount
Description	This event is raised as notification of sa...
Disposition	rmi://iwaylab1;RemoteObject=SIEBELFL
Content	http://iwaylab1:7777/ibse/IBSEServlet/schem...

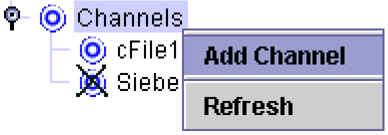
   You can now associate the event port with a channel.

Creating a Channel

To create a channel:

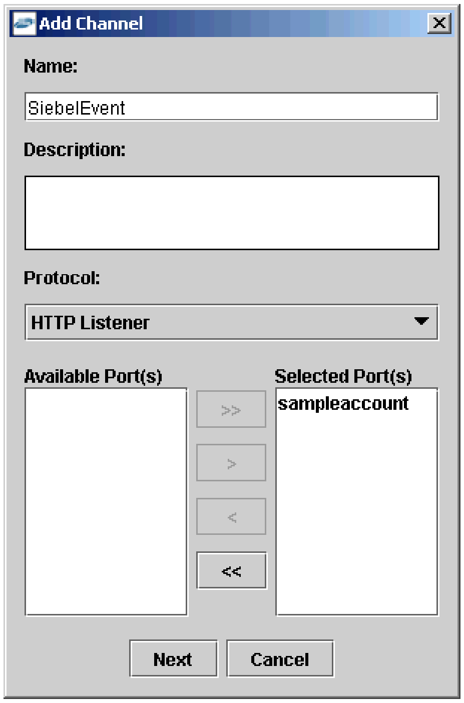
1. In the left pane, click the **Events** node.
2. Expand the **Siebel** node.

   The ports and channels nodes appear in the left pane.



3. Right-click **Channels** and select **Add Channel.**

   The Add Channel dialog box opens.



Add Channel

Name: SiebelEvent

Description:

Protocol: HTTP Listener

Available Port(s)

Selected Port(s): sampleaccount

a. Type a name for the channel, for example, `SiebelEvent`.

b. Type a brief description.

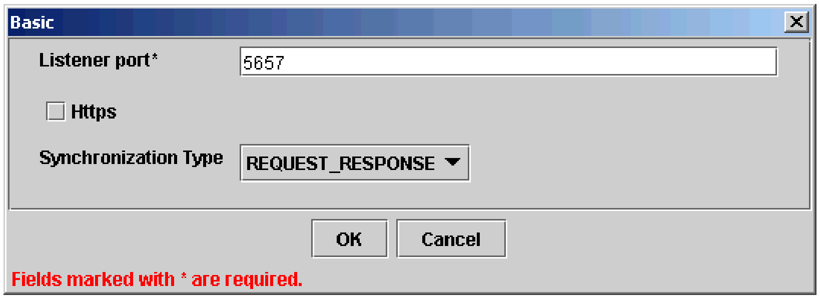
c. From the **Protocol** list, select **HTTP Listener, MQ Series Listener,** or **File Listener.**

d. Select an event port from the list of available ports. To select more than one, hold down the **Ctrl** key and click the ports.

e. To transfer the ports to the list of selected ports, click the **double right (>>)** arrow button.

4. Click **Next.**

   The Basic dialog box opens.



Basic

Listener port* 5657

□ Https

Synchronization Type REQUEST_RESPONSE

Fields marked with * are required.

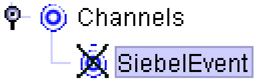
5. Enter a port number in the **Listener port** field.

6. Select **REQUEST_RESPONSE** from the Synchronization Type drop-down list.

7. Click **OK.**

   The channel appears under the channels node in the left pane.

   An X over the icon indicates that the channel is currently disconnected. You must start the channel to activate your event configuration.



8. Right-click the channel node and select **Start.**

   The channel you created becomes active.

   The X that was over the icon in the left pane disappears.

9. To stop the channel, right-click the connected channel node and select **Stop.**

Siebel Event Integration

This topic illustrates Siebel event integration. The procedures describe design time and runtime.

OracleAS Integration InterConnect Design Time

The following procedures describe how to start the repository and create a common view and then, publish and subscribe an event.

Starting the Repository

To start the repository, double-click the `start.bat` file located in the following directory:

```
D:\oracle\ora92InterCon\oai\9.0.4\repository\start.bat
```

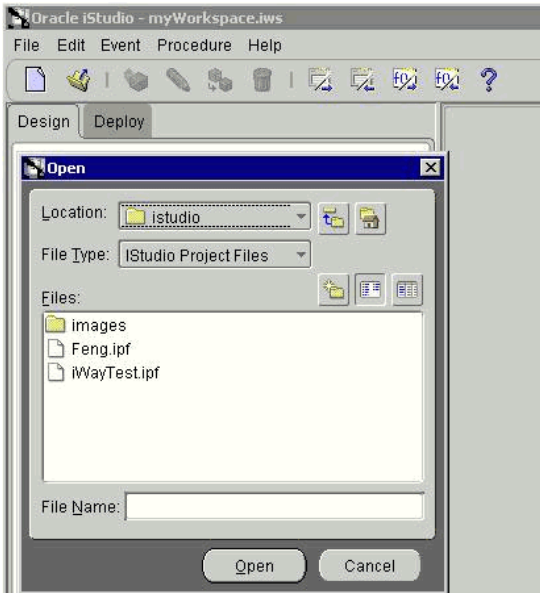
Creating a Common View

To create a Common View:

1. Start Oracle iStudio by double-clicking the `start.bat` file located in the following directory:

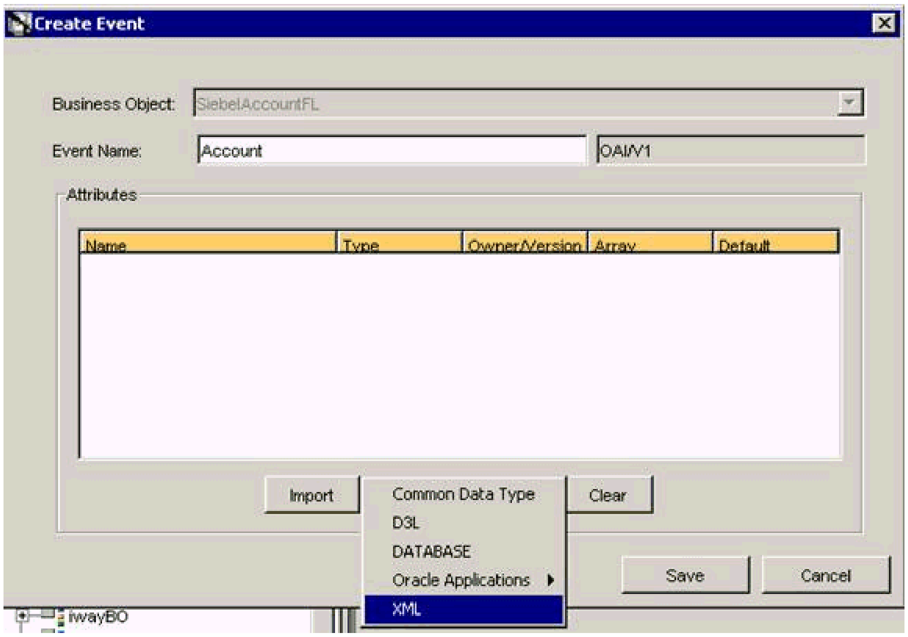
   ```
   D:\oracle\ora92iStudio\oai\9.0.4\istudio\iStudio.bat
   ```

   iStudio opens.

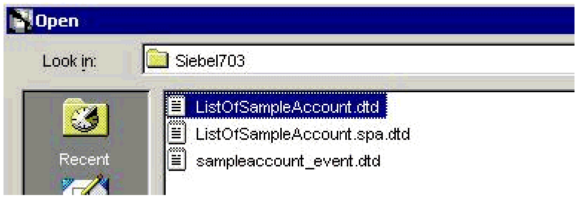

Oracle iStudio - myWorkspace.iws
File Edit Event Procedure Help
Design Deploy
Open
Location: istudio
File Type: IStudio Project Files
Files:
images
Feng.ipf
iWayTest.ipf
File Name:
Open
Cancel

2. Open a project.
3. Open **Common Views** and **Business Objects.**
4. Create a Business Object called **CustomerFL** and a new event under `CustomerFL`.

> **Note:** The event name must be the root element of the DTD generated from Application Explorer. In this example, the root element in the DTD is `Account`.

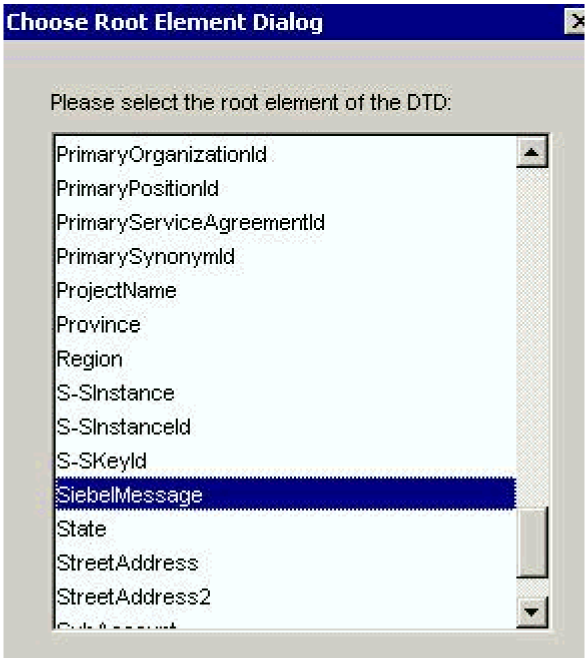


5. Click **Import** and select **XML** from the list.



6. Open the DTD generated from Application Explorer and load it.

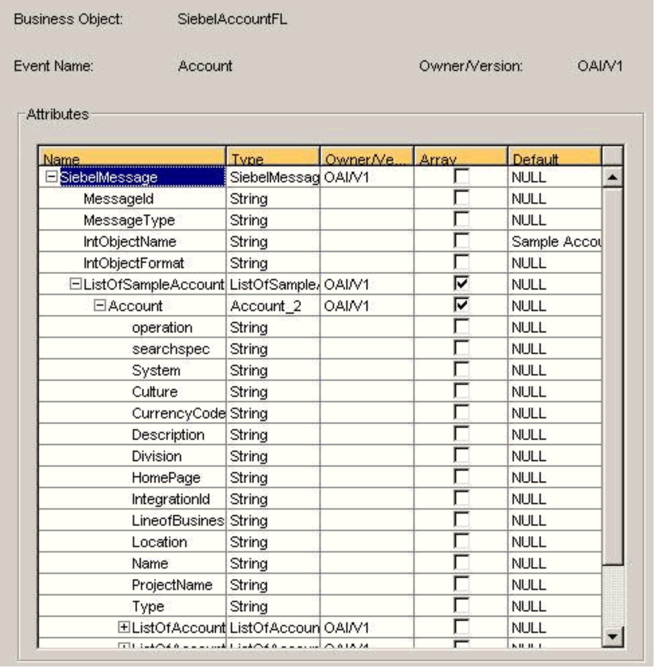
   The Choose Root Element dialog box opens.



Please select the root element of the DTD:

PrimaryOrganizationId
PrimaryPositionId
PrimaryServiceAgreementId
PrimarySynonymId
ProjectName
Province
Region
S-SInstance
S-SInstanceId
S-SKeyId
SiebelMessage
State
StreetAddress
StreetAddress2

7. Select the root element, which must be identical to the event name specified earlier.



Business Object: SiebelAccountFL

Event Name: Account

Owner/Version: OAI/V1

Name	Type	Owner/Ve...	Array	Default
⊟SiebelMessage	SiebelMessag	OAI/V1	☐	NULL
MessageId	String		☐	NULL
MessageType	String		☐	NULL
IntObjectName	String		☐	Sample Accou
IntObjectFormat	String		☐	NULL
⊟ListOfSampleAccount	ListOfSample/	OAI/V1	☑	NULL
⊟Account	Account_2	OAI/V1	☑	NULL
operation	String		☐	NULL
searchspec	String		☐	NULL
System	String		☐	NULL
Culture	String		☐	NULL
CurrencyCode	String		☐	NULL
Description	String		☐	NULL
Division	String		☐	NULL
HomePage	String		☐	NULL
IntegrationId	String		☐	NULL
LineofBusines	String		☐	NULL
Location	String		☐	NULL
Name	String		☐	NULL
ProjectName	String		☐	NULL
Type	String		☐	NULL
⊞ListOfAccount	ListOfAccoun	OAI/V1	☐	NULL

8. Click **Save.**

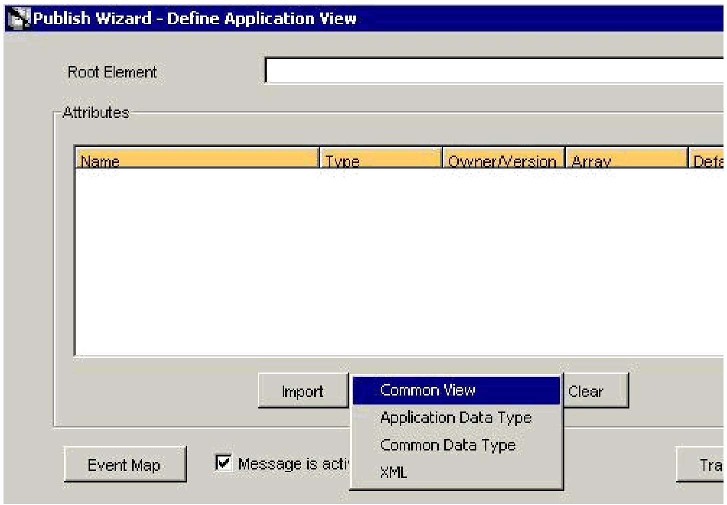
Publishing an Event

To publish an event:

1. Create a new application called **SIEBELFL.**
2. Expand **SIEBELFL.**
3. Right-click **Publish Events** and select **New.**

4. Select **XML** as the message type and select **Account** under the SiebelAccountFL business object as the event.

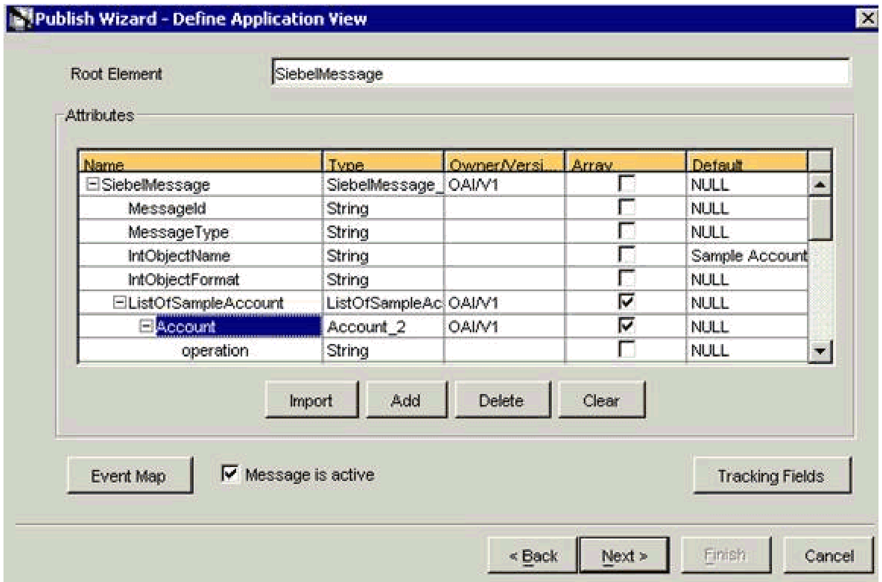
5. Click **Next.**



Name	Type	Owner/Version	Array	Defa

6. Click **Import** and select **Common View** from the list.

   The structure from the selected Common View loads.



Root Element: SiebelMessage

Name	Type	Owner/Versi	Array	Default
⊟SiebelMessage	SiebelMessage_	OAI/V1	☐	NULL
MessageId	String		☐	NULL
MessageType	String		☐	NULL
IntObjectName	String		☐	Sample Account
IntObjectFormat	String		☐	NULL
⊟ListOfSampleAccount	ListOfSampleAc	OAI/V1	☑	NULL
⊟Account	Account_2	OAI/V1	☑	NULL
operation	String		☐	NULL

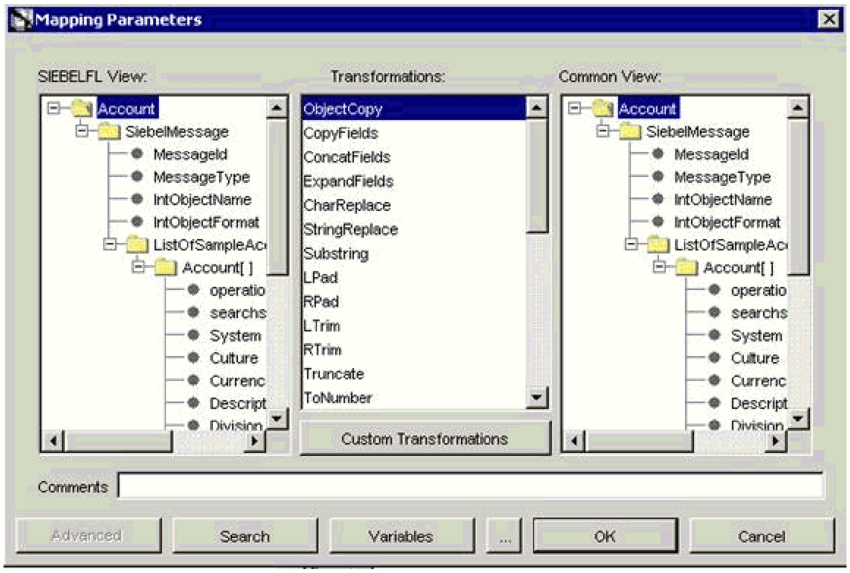
☑ Message is active

7. Ensure you enter the root element of the XML message, for example, `SiebelMessage`.

8. Click **Next.**

9. Click **New** to create a mapping between the Common View and Application View.

   In this example, the Application View and Common View have the same structure. All the attributes can be mapped by using the ObjectCopy transformation.



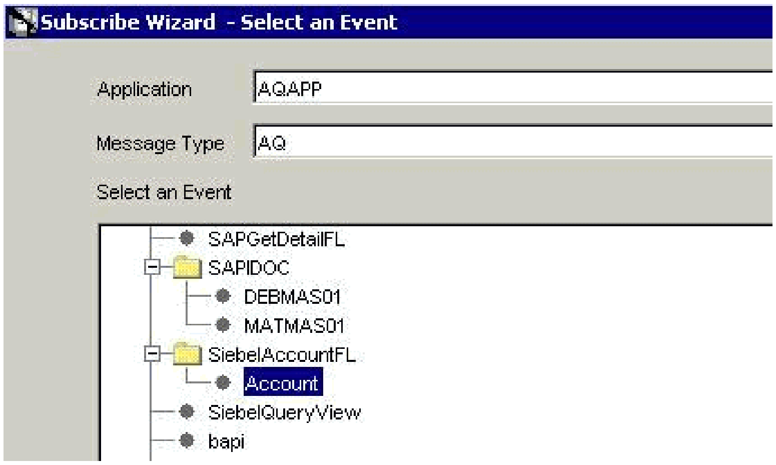
10. Click **Apply, OK,** and then, **Finish.**

    The application definition for a published event is now complete.

Subscribing an Event

To subscribe an event:

1. Create a new application called **AQAPP.**
2. Expand **AQAPP.**
3. Right-click **Subscribe Events** and select **New.**



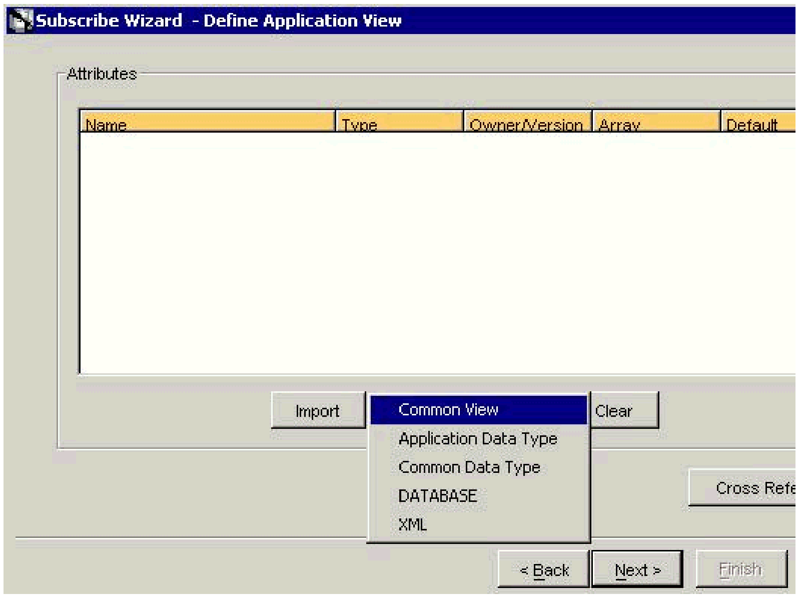
Subscribe Wizard - Select an Event

Application: AQAPP

Message Type: AQ

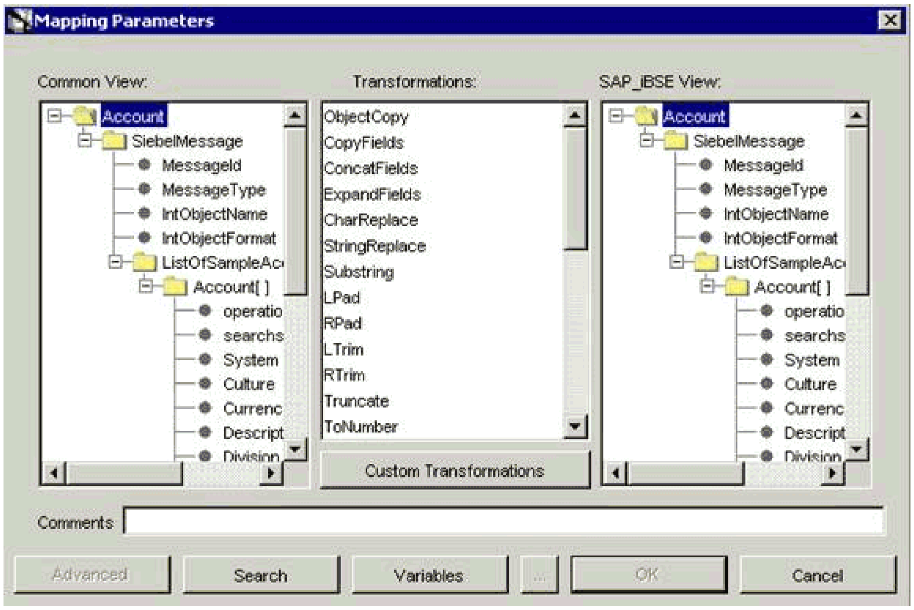
   a. Select **AQ** as the message type.
   b. Under the `SiebelAccountFL` business object, select **Account** as the event.

4. Click **Next.**



5. Click **Next.**

6. Click **New** to define mapping between the Application View and the Common View.



   a. Define the mapping for each field.

   b. After you map each field, click **Apply.**

7. When you have finished the mapping process, click **OK.**

   The application definition for a subscribed event is now complete.

OracleAS Integration InterConnect Runtime

The following topic describes how to trigger an event in Siebel to verify event integration using the OracleAS Adapter.

Verifying Event Integration

To verify event integration:

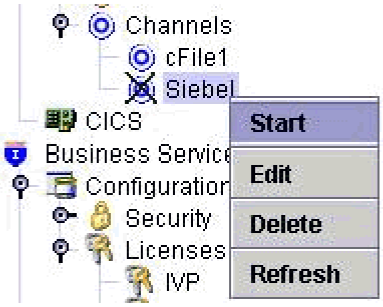
1. Start the Oracle Application Server or ensure that the server is running.
2. Restart OC4J, if required, by executing the following command:

```
\OracleAS_home\opmn\bin\opmnctl  stopproc process-type=home
\OracleAS_home\opmn\bin\opmnctl  startproc process-type=home
```

3. Check the status of OC4J by executing the following command:

```
\OracleAS_home\opmn\bin\opmnctl  status
```

4. Start the channel in Application Explorer.



5. Start the publishing adapter and the subscribing adapter, by executing the following commands:

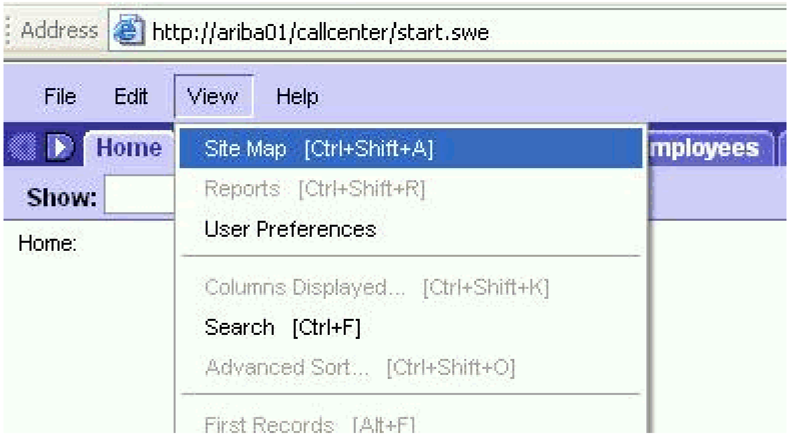
```
I:\oracle\ora10gRC3\integration\interconnect\adapters\AQAPP\start.bat
I:\oracle\ora10gRC3\integration\interconnect\adapters\SIEBELFL\start.bat
```

Triggering an Event

To trigger an event in Siebel:

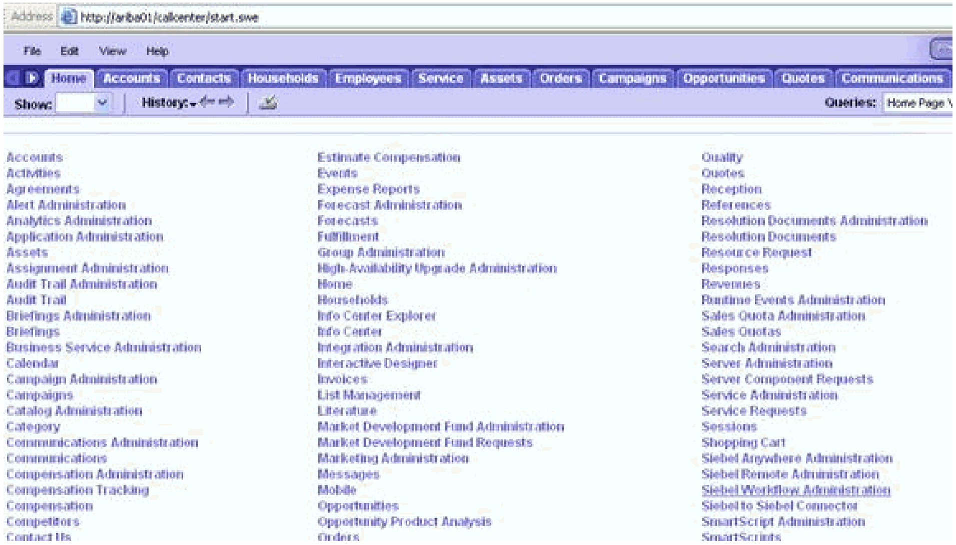
1. Start the Siebel Call Center by entering the following URL in a browser:

```
http://ariba01/callcenter/start.swe
```



2. Click **View** and select **Site Map** from the list.

   The Site Map view is displayed.



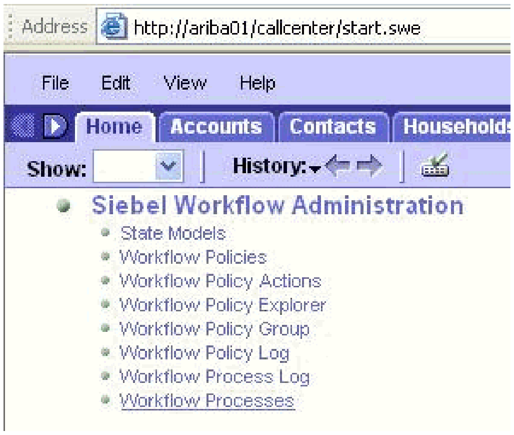
Accounts
Activities
Agreements
Alert Administration
Analytics Administration
Application Administration
Assets
Assignment Administration
Audit Trail Administration
Audit Trail
Briefings Administration
Briefings
Business Service Administration
Calendar
Campaign Administration
Campaigns
Catalog Administration
Category
Communications Administration
Communications
Compensation Administration
Compensation Tracking
Compensation
Competitors
Contact Us

Estimate Compensation
Events
Expense Reports
Forecast Administration
Forecasts
Fulfillment
Group Administration
High-Availability Upgrade Administration
Home
Households
Info Center Explorer
Info Center
Integration Administration
Interactive Designer
Invoices
List Management
Literature
Market Development Fund Administration
Market Development Fund Requests
Marketing Administration
Messages
Mobile
Opportunities
Opportunity Product Analysis
Orders

Quality
Quotes
Reception
References
Resolution Documents Administration
Resolution Documents
Resource Request
Responses
Revenues
Runtime Events Administration
Sales Quota Administration
Sales Quotas
Search Administration
Server Administration
Server Component Requests
Service Administration
Service Requests
Sessions
Shopping Cart
Siebel Anywhere Administration
Siebel Remote Administration
Siebel Workflow Administration
Siebel to Siebel Connector
SmartScript Administration
SmartScripts

3. Click **Siebel Workflow Administration.**

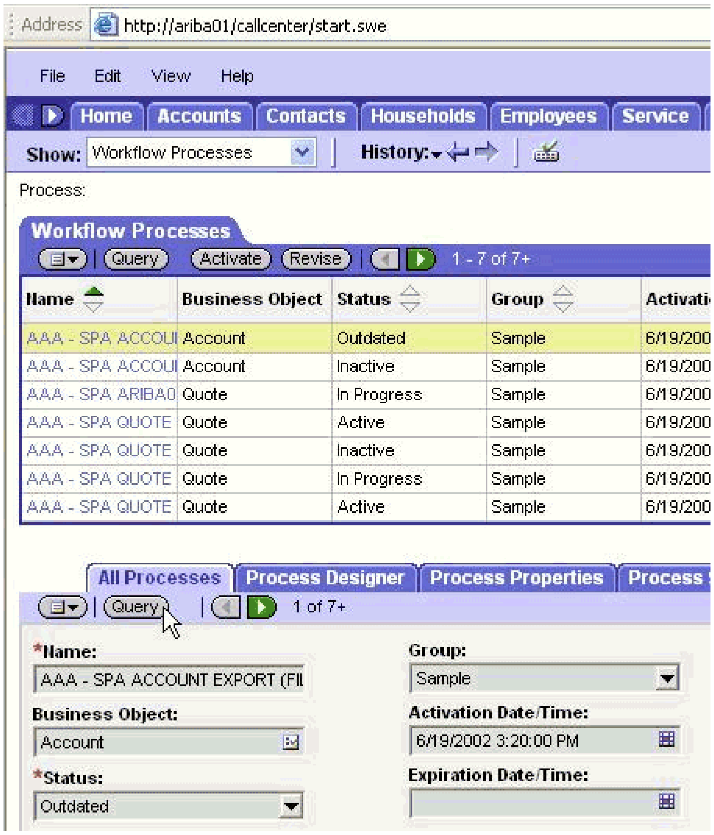
   The Siebel Workflow Administration page opens.



- Siebel Workflow Administration
  - State Models
  - Workflow Policies
  - Workflow Policy Actions
  - Workflow Policy Explorer
  - Workflow Policy Group
  - Workflow Policy Log
  - Workflow Process Log
  - Workflow Processes

4. Click **Workflow Processes.**

   The Workflow Processes page opens.



Name	Business Object	Status	Group	Activati
AAA - SPA ACCOUI	Account	Outdated	Sample	6/19/200
AAA - SPA ACCOUI	Account	Inactive	Sample	6/19/200
AAA - SPA ARIBA0	Quote	In Progress	Sample	6/19/200
AAA - SPA QUOTE	Quote	Active	Sample	6/19/200
AAA - SPA QUOTE	Quote	Inactive	Sample	6/19/200
AAA - SPA QUOTE	Quote	In Progress	Sample	6/19/200
AAA - SPA QUOTE	Quote	Active	Sample	6/19/200

*Name: AAA - SPA ACCOUNT EXPORT (FIL

Group: Sample

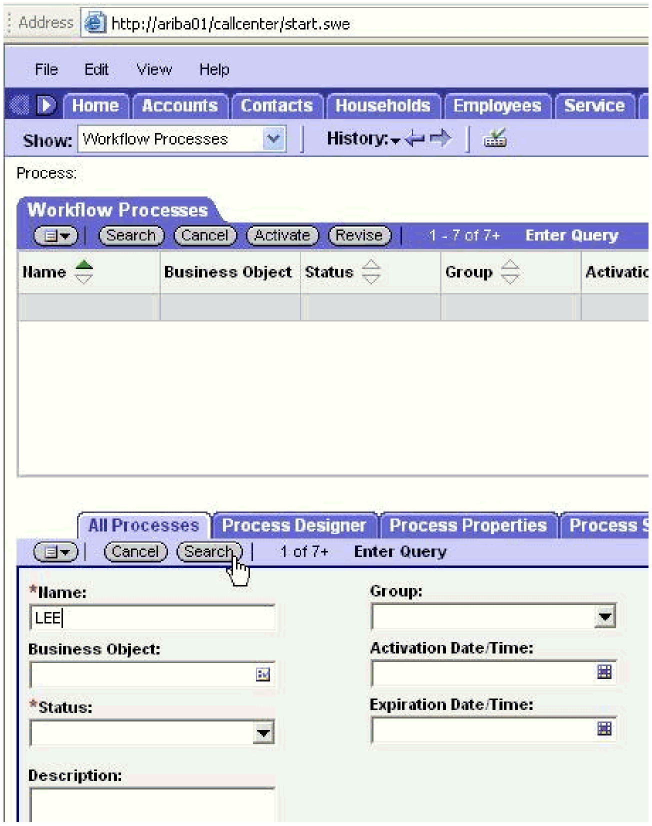
Business Object: Account

Activation Date/Time: 6/19/2002 3:20:00 PM

*Status: Outdated

Expiration Date/Time:

5. Click **Query** to search for the Workflow needed to trigger a Siebel event.



*Name: LEE

Group:

Business Object:

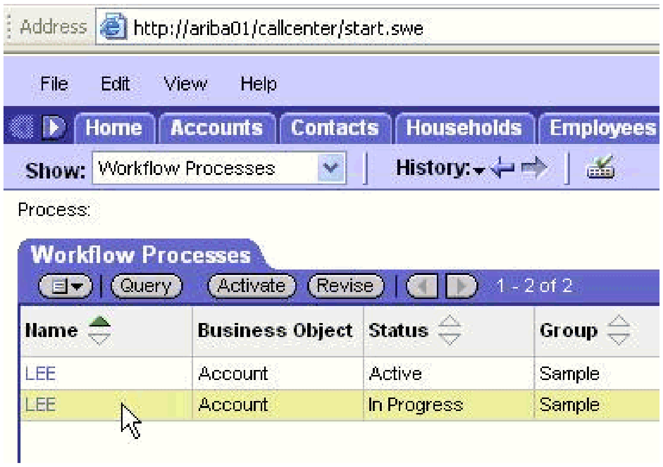
Activation Date/Time:

*Status:

Expiration Date/Time:

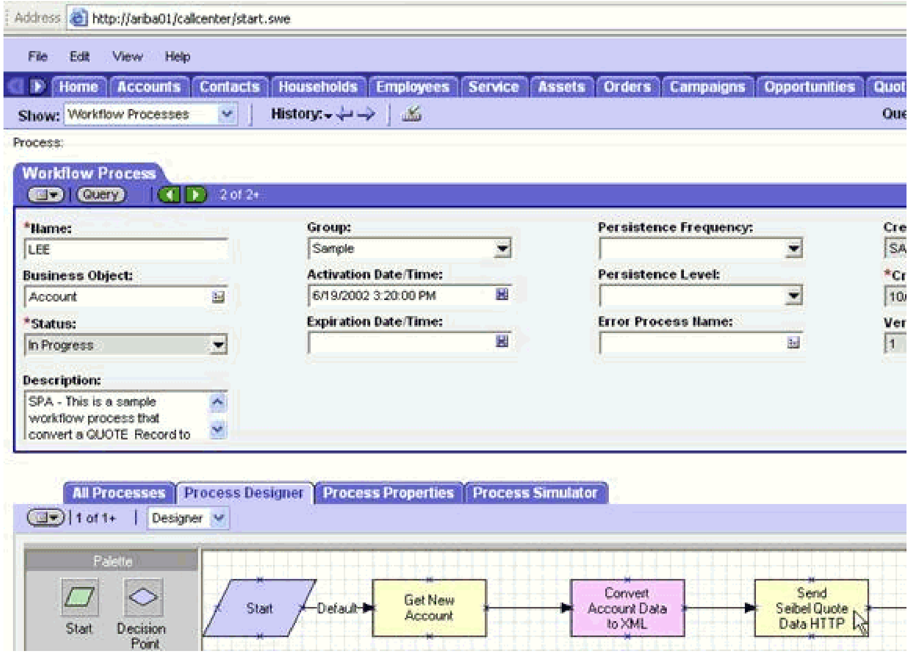
Description:

6. Type a Siebel workflow name, for example, **LEE** and click **Search.**



Name	Business Object	Status	Group
LEE	Account	Active	Sample
LEE	Account	In Progress	Sample

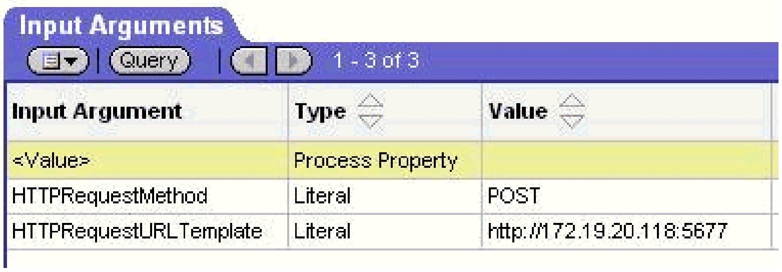
7. Select the workflow, for example, **LEE.**



*Name: LEE	Group: Sample	Persistence Frequency:	Cre SA
Business Object: Account	Activation Date/Time: 6/19/2002 3:20:00 PM	Persistence Level:	*Cr 10
*Status: In Progress	Expiration Date/Time:	Error Process Name:	Ver 1
Description: SPA - This is a sample workflow process that convert a QUOTE Record to			

8. Click the **Process Designer** tab and double-click the **Send Siebel Quote Data HTTP** workflow element.

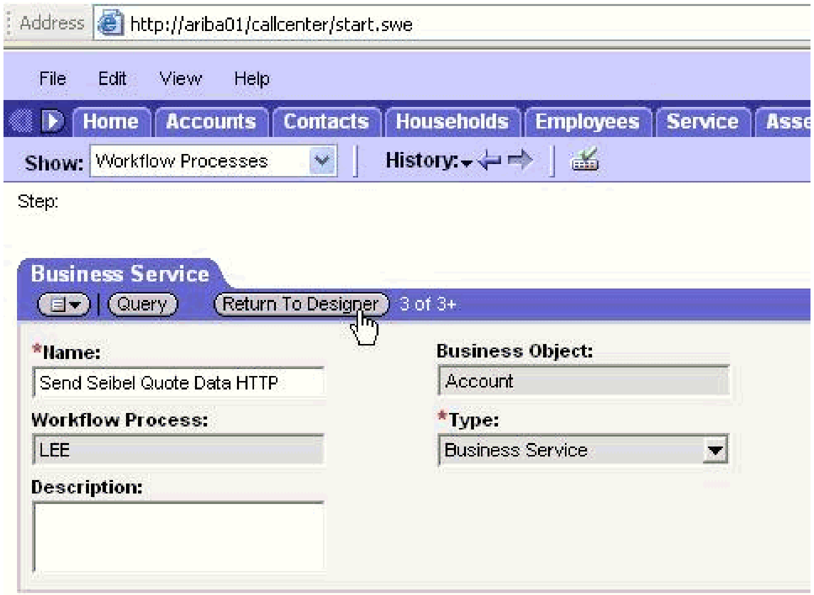
   The Input Arguments tab displays.



Input Argument	Type	Value
<Value>	Process Property	
HTTPRequestMethod	Literal	POST
HTTPRequestURLTemplate	Literal	http://172.19.20.118:5677

9. Type the IP address and port for the HTTPRequestURLTemplate input argument.

10. Click **Return To Designer.**


Address http://ariba01/callcenter/start.swe
File Edit View Help
Home Accounts Contacts Households Employees Service Asse
History:
Step:
Business Service
Query
Return To Designer
3 of 3+

Show: Workflow Processes

*Name: Send Seibel Quote Data HTTP

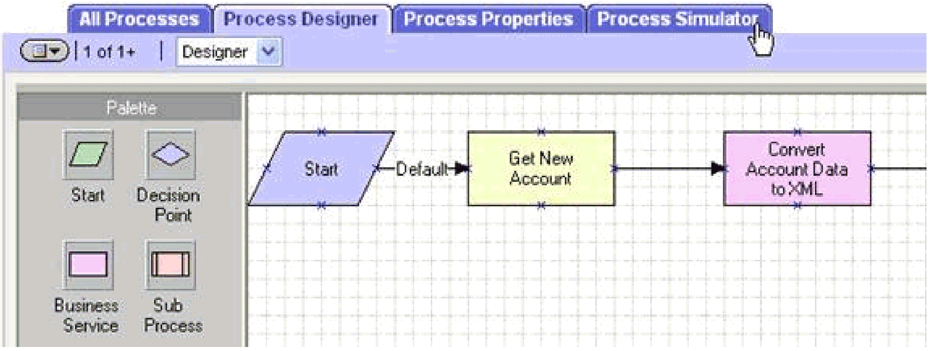
Business Object: Account

Workflow Process: LEE

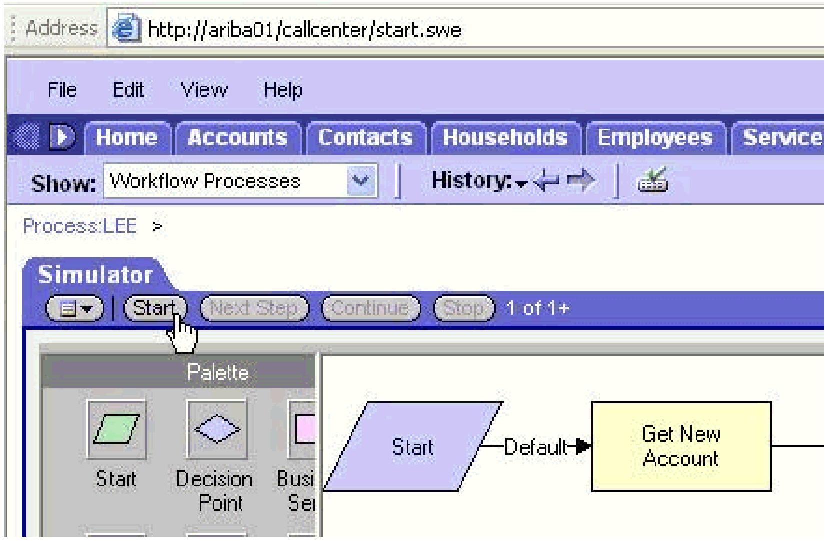
*Type: Business Service

Description:

11. Click the Process Simulator tab.



The Simulator tab opens.



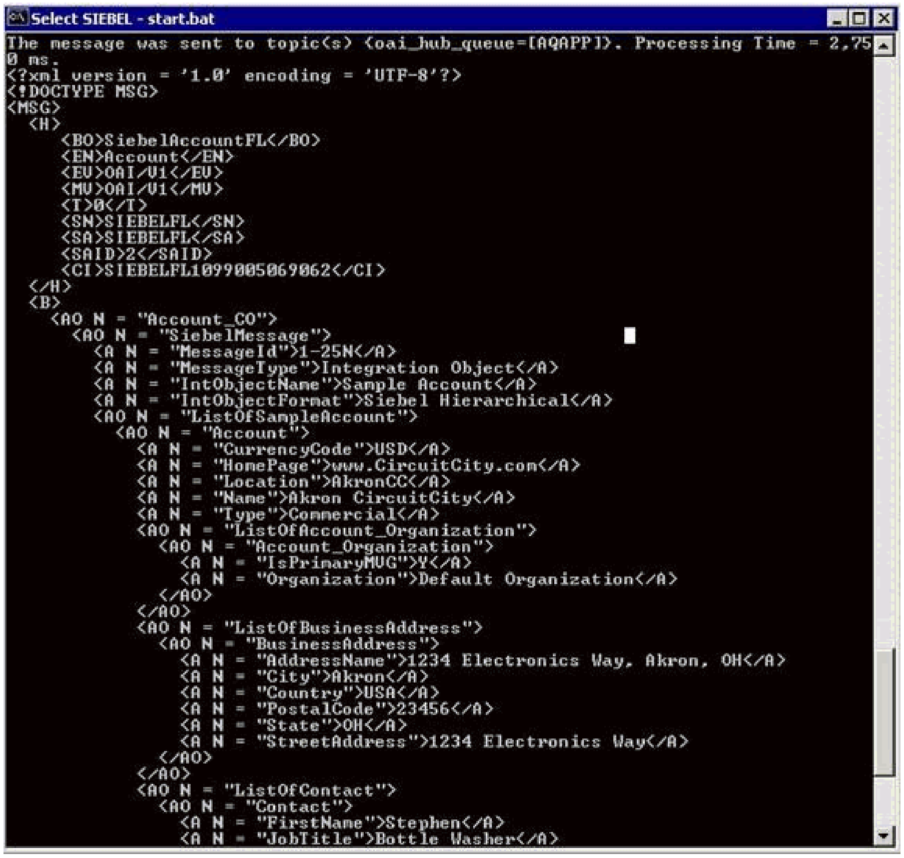
Show: Workflow Processes

12. Click **Start** then **Continue** to complete the Siebel event triggering process.

Verifying Results

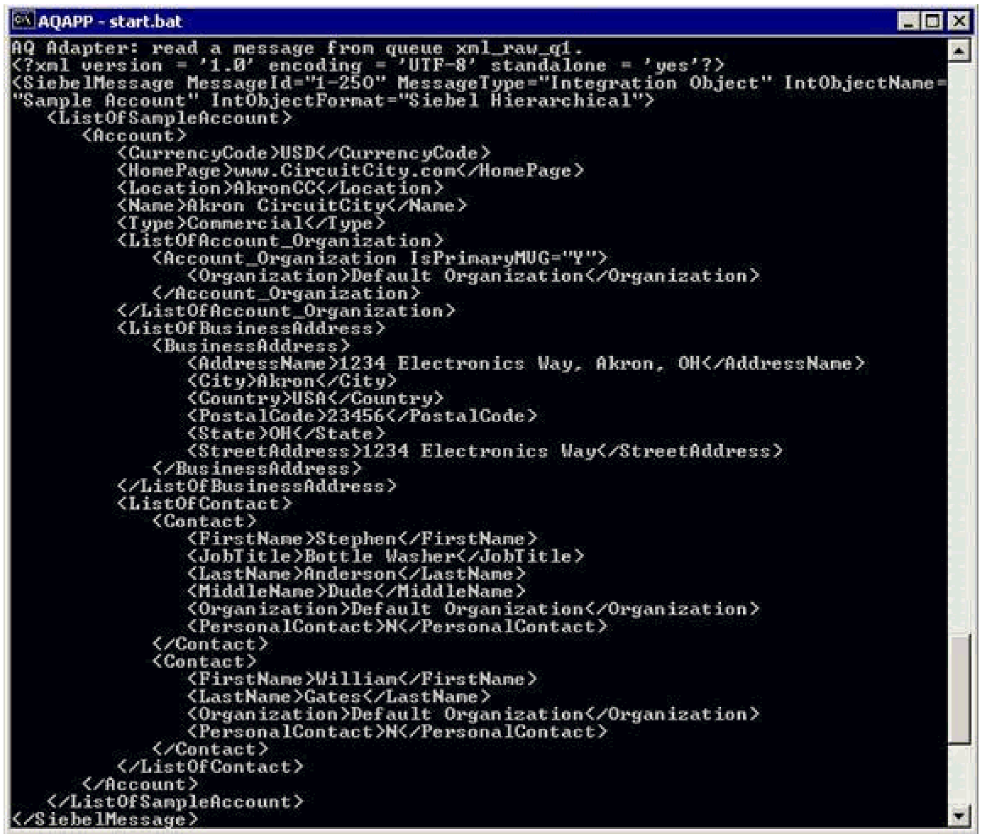
The following section includes a sample publishing adapter (SIEBELFL) console log and subscribing adapter (AQAPP) console log.

Sample Publishing Adapter (SIEBELFL) Console Log



```
The message was sent to topic(s) (oai_hub_queue=[AQAPP]). Processing Time = 2,75
0 ms.
<?xml version = '1.0' encoding = 'UTF-8'?>
<!DOCTYPE MSG>
<MSG>
  <H>
     <BO>SiebelAccountFL</BO>
     <EN>Account</EN>
     <EV>OAI/V1</EV>
     <MV>OAI/V1</MV>
     <T>0</T>
     <SN>SIEBELFL</SN>
     <SA>SIEBELFL</SA>
     <SAID>2</SAID>
     <CI>SIEBELFL1099005069062</CI>
  </H>
  <B>
    <AO N = "Account_CO">
      <AO N = "SiebelMessage">
        <A N = "MessageId">1-25N</A>
        <A N = "MessageType">Integration Object</A>
        <A N = "IntObjectName">Sample Account</A>
        <A N = "IntObjectFormat">Siebel Hierarchical</A>
        <AO N = "ListOfSampleAccount">
          <AO N = "Account">
            <A N = "CurrencyCode">USD</A>
            <A N = "HomePage">www.CircuitCity.com</A>
            <A N = "Location">AkronCC</A>
            <A N = "Name">Akron CircuitCity</A>
            <A N = "Type">Commercial</A>
            <AO N = "ListOfAccount_Organization">
              <AO N = "Account_Organization">
                <A N = "IsPrimaryMVG">Y</A>
                <A N = "Organization">Default Organization</A>
              </AO>
            </AO>
            <AO N = "ListOfBusinessAddress">
              <AO N = "BusinessAddress">
                <A N = "AddressName">1234 Electronics Way, Akron, OH</A>
                <A N = "City">Akron</A>
                <A N = "Country">USA</A>
                <A N = "PostalCode">23456</A>
                <A N = "State">OH</A>
                <A N = "StreetAddress">1234 Electronics Way</A>
              </AO>
            </AO>
            <AO N = "ListOfContact">
              <AO N = "Contact">
                <A N = "FirstName">Stephen</A>
                <A N = "JobTitle">Bottle Washer</A>
```

Sample Subscribing Adapter (AQAPP) Console Log



```
AQ Adapter: read a message from queue xml_raw_q1.
<?xml version = '1.0' encoding = 'UTF-8' standalone = 'yes'?>
<SiebelMessage MessageId="1-250" MessageType="Integration Object" IntObjectName=
"Sample Account" IntObjectFormat="Siebel Hierarchical">
   <ListOfSampleAccount>
      <Account>
         <CurrencyCode>USD</CurrencyCode>
         <HomePage>www.CircuitCity.com</HomePage>
         <Location>AkronCC</Location>
         <Name>Akron CircuitCity</Name>
         <Type>Commercial</Type>
         <ListOfAccount_Organization>
            <Account_Organization IsPrimaryMVG="Y">
               <Organization>Default Organization</Organization>
            </Account_Organization>
         </ListOfAccount_Organization>
         <ListOfBusinessAddress>
            <BusinessAddress>
               <AddressName>1234 Electronics Way, Akron, OH</AddressName>
               <City>Akron</City>
               <Country>USA</Country>
               <PostalCode>23456</PostalCode>
               <State>OH</State>
               <StreetAddress>1234 Electronics Way</StreetAddress>
            </BusinessAddress>
         </ListOfBusinessAddress>
         <ListOfContact>
            <Contact>
               <FirstName>Stephen</FirstName>
               <JobTitle>Bottle Washer</JobTitle>
               <LastName>Anderson</LastName>
               <MiddleName>Dude</MiddleName>
               <Organization>Default Organization</Organization>
               <PersonalContact>N</PersonalContact>
            </Contact>
            <Contact>
               <FirstName>William</FirstName>
               <LastName>Gates</LastName>
               <Organization>Default Organization</Organization>
               <PersonalContact>N</PersonalContact>
            </Contact>
         </ListOfContact>
      </Account>
   </ListOfSampleAccount>
</SiebelMessage>
```

5

Troubleshooting and Error Messages

This chapter explains the limitations and workarounds when connecting to Siebel. The following topics are discussed:

- Troubleshooting
- BSE Error Messages

Troubleshooting

This topic provides troubleshooting information for Siebel, separated into four categories:

- Application Explorer
- Siebel
- OracleAS Adapter JCA
- BSE

> **Note:** Log file information that can be relevant in troubleshooting can be found in the following locations:
>
> - The OracleAS Adapter JCA trace information can be found under the `OracleAS_home\opmn\logs` directory.
> - BSE trace information can be found under the `OracleAS_home\j2ee\home\applications\ws-app-adapter\ibse\ibselogs` directory.
> - The log file for Application Explorer can be found under the `OracleAS_home\adapters\application\tools` directory.

Application Explorer

To use Application Explorer on Windows for debugging or testing purposes, invoke the ae batch script, `ae.bat`, found under `OracleAS_home\adapters\application\tools` or on UNIX invoke the ae script, `ae.sh`, found under `OracleAS_home/adapters/application/tools`.

Error	Solution
Siebel does not appear in the Application Explorer Adapter node list.	Ensure that the Siebel jar files supplied with your Siebel distribution media have been placed in the `*OracleAS_home*\adapters\application\lib` directory. For example, for Siebel 7.03 environments, the `SiebelJI_Common.jar` and `SiebelJI_enu.jar` should be placed in the `*OracleAS_home*\adapters\application\lib` directory.
Target Type drop down contains only Java Data Bean Connection and COM connection type is desired.	Ensure that the Siebel thin client is installed correctly on the machine hosting Application Explorer so that appropriate COM environment is available.
An error message that includes the name of the Siebel Gateway server appears when you try to connect to a Siebel target. For example, `Problem activating adapter (ariba0x). Check logs for more information.`	Ensure that the name of the Siebel Gateway server is correctly defined for the target you are using.
You receive the following error when trying to connect to a Siebel target: `Problem activating adapter. (You have entered an invalid set of logon parameters. Please type in your logon parameters again.). Check logs for more information.`	Ensure that the User ID and password parameter values to connect to your Siebel system are correct.
You receive the following error when trying to connect to a Siebel target: `Problem activating adapter. (Couldn't get nameserver connection). Check logs for more information.`	Check on network connectivity to Siebel environment. Correct networking problem and retry connection.
You receive the following error when attempting to connect to a Siebel target: `Problem activating adapter. (NSReadKey request failed (no error information)...). Check logs for more information.`	Ensure that the values defined for Siebel Server, Enterprise Name, and Object Manager for the target you are using are correct, and retry the connection
You receive the following error when attempting to connect to a Siebel target: `Problem activating adapter. (Error loading translatable messages: com.siebel.locale.enux.messages.SSAMessages_enux). Check logs for more information`	Ensure that the value of the Language parameter on the Advanced tab is defined correctly for the target you are using to connect to your Siebel system (for example, enu for English).
A successful connection is made to Siebel environment but no values are available in Business Object, Business Service, and Integration Object nodes in Application Explorer tree.	The Repository Name specified on the Advanced tab in the Siebel target configuration is either void or empty of any components in the targeted Siebel environment or that Repository Name is not valid for the targeted Siebel environment. Verify that the Repository Name is valid and contains components for interrogation then re-connect.

Siebel

The error messages listed can occur when using the adapter with either a BSE or OracleAS Adapter JCA repository project.

Error	Solution
A successful connection is made to Siebel environment but no values are available in Business Object, Business Service, and Integration Object nodes in Application Explorer tree.	The Repository Name specified on the Advanced tab in the Siebel Target configuration is either void or empty of any components in the targeted Siebel environment or that Repository Name is not valid for the targeted Siebel environment. Verify that the Repository Name is valid and contains components for interrogation then re-connect.
When executing a request, the following error message appears: `AdapterException: Unsupported Action: {0} Tquery`	Verify that method is available for specific request by verifying schema.
When executing a request, the following error message appears: `AdapterException: Field 'NFame' does not exist in definition for business component 'Account'. Please ask your systems administrator to check your application configuration.`	Ensure that field names are valid within request document by referring to schema for that specific object, and then re-submit the request.
When connecting to releases prior to Siebel 7.7 using the Java Data Bean Interface, you cannot reconnect after initial connection loss. This might occur when Application Explorer experiences a brief loss of network connection or if the Siebel Server or Gateway Service is restarted while Application Explorer is logged into the Siebel application.	Restart OC4J and Application Explorer in order to log in successfully to the Siebel application. This is a known Siebel API issue. See Siebel Alert 984 for more information.

OracleAS Adapter JCA

Error	Solution
In Application Explorer, the following error message appears when you attempt to connect to an OracleAS Adapter JCA configuration: `Could not initialize JCA`	In the Details tab in the right pane, ensure that the directory specified in the Home field points to the correct directory, for example, *`OracleAS_home`*\adapters\application

BSE Error Messages

This topic discusses the different types of errors that can occur when processing Web services through Business Services Engine (BSE).

General Error Handling in BSE

Business Services Engine (BSE) serves as both a SOAP gateway into the adapter framework and as the engine for some of the adapters. In both design time and execution time, various conditions can cause errors in BSE when Web services that use adapters are running. Some of these conditions and resulting errors are exposed the same way, regardless of the specific adapter; others are exposed differently, based on the adapter being used. This topic explains what you can expect when you encounter some of the more common error conditions on an adapter-specific basis.

Usually the SOAP gateway (agent) inside BSE passes a SOAP request message to the adapter required for the Web service. If an error occurs, how it is exposed depends on the adapter and the API or interfaces that the adapter uses. A few scenarios cause the SOAP gateway to generate a SOAP fault. In general, anytime the SOAP agent inside BSE receives an invalid SOAP request, a SOAP fault element is generated in the SOAP response. The SOAP fault element contains fault string and fault code elements. The fault code contains a description of the SOAP agent error.

The following SOAP response document results when BSE receives an invalid SOAP request:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">

 <SOAP-ENV:Body>
      <SOAP-ENV:Fault>
         <faultcode>SOAP-ENV:Client</faultcode>
         <faultstring>Parameter node is missing</faultstring>
      </SOAP-ENV:Fault>
   </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

In this example, BSE did not receive an element in the SOAP request message that is mandatory for the WSDL for this Web service.

Adapter-Specific Error Handling

When an adapter raises an exception during execution, the SOAP agent in BSE produces a SOAP fault element in the generated SOAP response. The SOAP fault element contains fault code and fault string elements. The fault string contains the native error description from the adapter target system. Since adapters use the target system interfaces and APIs, whether or not an exception is raised depends on how the target systems interface or API treats the error condition. If a SOAP request message is passed to an adapter by the SOAP agent in BSE, and that request is invalid based on the WSDL for that service, the adapter may raise an exception yielding a SOAP fault.

While it is almost impossible to anticipate every error condition that an adapter may encounter, the following is a description of how adapters handle common error conditions and how they are then exposed to the Web services consumer application.

Oracle Application Server Adapter for Siebel Invalid SOAP Request

When the Oracle Application Server Adapter for Siebel receives a SOAP request message that does not conform to the WSDL for the Web services being executed, the following SOAP response is generated

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Body>
 <SOAP-ENV:Fault>
  <faultcode>SOAP-ENV:Server</faultcode>
<faultstring>XD[FAIL] Parse failure (IS) 3: org.xml.sax.SAXParseException:
Premature end of file.</faultstring>
 </SOAP-ENV:Fault>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Empty Result From Siebel Request

When the Oracle Application Server Adapter for Siebel cannot connect to Siebel when executing a Web service, the following SOAP response is generated.

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
 <SOAP-ENV:Body>
  <SOAP-ENV:Fault>
  <faultcode>SOAP-ENV:Server</faultcode>
  <faultstring><Exception> - major:4096 minor: -1 message:NSReadKey request 11 was
abandoned
after 37846ms connection:12a due to Connection shutdown request
Connection reset by peer:JVM_recv in socket input stream
stream read DetailedMessage:Unknown</Exception></faultstring>
  </SOAP-ENV:Fault>
 </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Oracle Application Server Integration Adapter

The OracleAS Adapter connects BSE to adapters whose engine is another Oracle server. Therefore, since this adapter is used to connect BSE to many different target systems, the error handling behavior is consistent. Check the user guide for your adapter to see if you require the Oracle Application Server Integration Adapter when running Web services.

Invalid SOAP Request

When the Oracle Application Server Adapter receives a SOAP request message that does not conform to the WSDL for the Web services being executed, the following SOAP response is generated.

```
<?xml version="1.0" encoding="ISO-8859-1"
 ?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Body>
 <SOAP-ENV:Fault>
  <faultcode>SOAP-ENV:Server</faultcode>
  <faultstring>RPC server connection failed: Connection refused:
connect</faultstring>
 </SOAP-ENV:Fault>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Empty Result From Oracle Application Server Adapter Request

When the Oracle Application Server Adapter executes a SOAP request using input parameters passed that do not match records in the target system, the following SOAP response is generated.

> **Note:** The condition for this adapter does not yield a SOAP fault.

```
<SOAP-ENV:Envelope xmlns:xsi="http://www.w3.org/1999/XMLSchema-instance"
xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/1999/XMLSchema">
   <SOAP-ENV:Body>
      <m:RunDBQueryResponse xmlns:m="urn:schemas-iwaysoftware-com:iwse"
        xmlns="urn:schemas-iwaysoftware-com:iwse"
        cid="2A3CB42703EB20203F91951B89F3C5AF">
```

```
            <RunDBQueryResult run="1" />
         </m:RunDBQueryResponse>
      </SOAP-ENV:Body>
    </SOAP-ENV:Envelope>
```

6

Advanced Topics

This chapter includes the following topics:

- Using Web Services Policy-Based Security
- Migrating Repositories

Using Web Services Policy-Based Security

Application Explorer provides a security model called Web services policy-based security. The following topics describe how the feature works and how to configure it.

Web Services Policy-Based Security

Web services provide a layer of abstraction between the back-end business logic they invoke, and the user or application running the Web service. This enables easy application integration but raises the issue of controlling the use and execution of critical and sensitive business logic that is run as a Web service.

Application Explorer controls the use of Web services that use adapters, using a feature called policy-based security. This feature enables an administrator to apply "policies" to Business Services (Web services) to deny or permit their execution.

A policy is a set of privileges dealing with the execution of a Business Service (iBS) that can be applied to an existing or new iBS. When you set specific rights or privileges inside a policy, you do not have to re-create privileges for every iBS that has security concerns in common with other Business Services. Instead, you reuse a policy on multiple Business Services.

The goal of the feature is to secure requests at both the transport and the SOAP request level transmitted on the wire. Some of the policies do not deal with security issues directly, but do effect the run-time behavior of the Web services to which they have been applied.

The iBS administrator creates an "instance" of a policy type, names it, associates individual users or groups (a collection of users), and then applies that policy to one or more Business Services.

You can assign a policy to an iBS, or to a method within an iBS. If a policy is only applied to a method, other methods in that iBS will not be governed by it. However, if a policy is applied to the iBS, all methods are governed by it. At runtime, the user ID and password that are sent to BSE in the SOAP request message are checked against the list of users for all policies applied to that specific iBS. The policy type that is supported is Resource Execution, which dictates who can or cannot execute the iBS.

When a policy is not applied, the default value for an iBS is to "grant all". For example, anybody can execute the iBS, until the Resource Execution policy is associated to the iBS. At that time, only those granted execution permissions, or users not part of the group that has been denied execution permissions, have access to the iBS.

Configuring Web Services Policy-Based Security

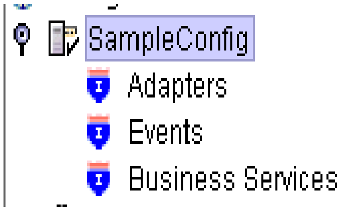
The following procedures describe how to configure Web services policy-based security.

Creating and Associating a User with a Policy

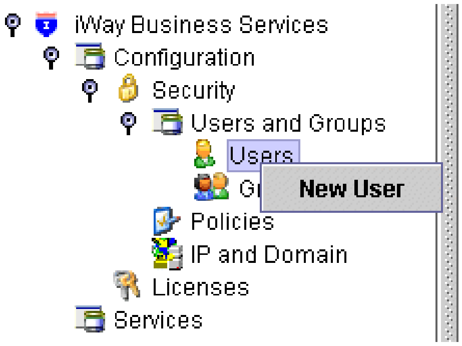
Before you create instances of policies, you must have a minimum of one user or one group to associate to an instance. You can create users and groups using Application Explorer.

1. Open Application Explorer.
2. Right-click the configuration to which you want to connect, for example, SampleConfig. See Chapter 2, "Adapter Configuration Using OracleAS Adapter Application Explorer" for information on creating a new configuration.
3. Select **Connect.**

   Nodes appear for **Adapters, Events,** and **Business Services** (also known as Web services).

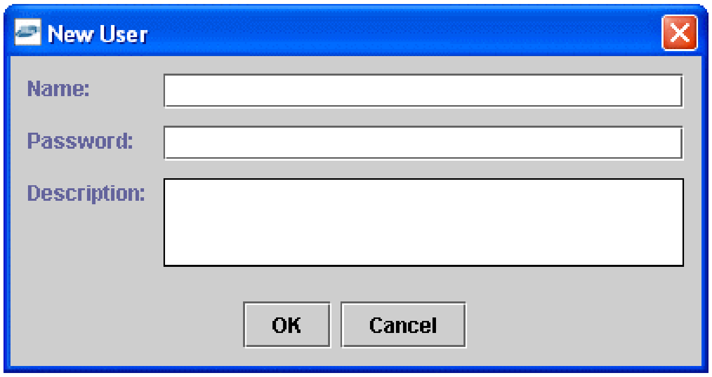


   a. Expand the **Business Services** node by clicking on the plus (+) symbol.
   b. Expand the **Configuration** node by clicking the plus (+) symbol.
   c. Expand the **Security** node by clicking the plus (+) symbol.
   d. Expand the **Users and Groups** node by clicking the plus (+) symbol.



4. Right-click **Users** and click **New User.**

   The New User dialog box opens.

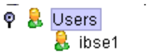


a. In the **Name** field, type a user ID.

b. In the **Password** field, type the password associated with the user ID.

c. In the **Description** field, type a description of the user (optional).

5. Click **OK.**



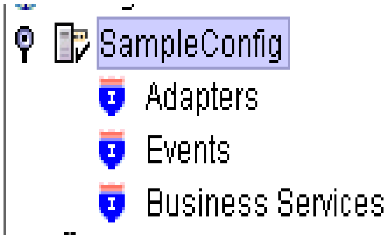
The new user is added under the Users node.

Creating a Group to Use With a Policy

To create a group to use with a policy:

1. Open Application Explorer.
2. Right-click the configuration to which you want to connect, for example, SampleConfig. See Chapter 2, "Adapter Configuration Using OracleAS Adapter Application Explorer" for information on creating a new configuration.
3. Select **Connect.**

   Nodes appear for **Adapters, Events,** and **Business Services** (also known as Web services).

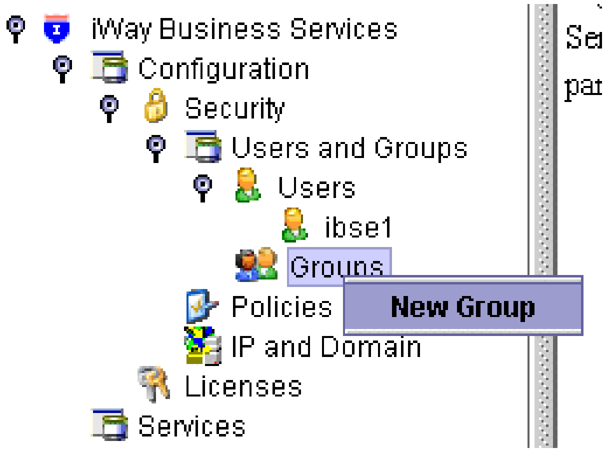


a. Expand the **Business Services** node by clicking on the plus (+) symbol.

b. Expand the **Configurations** node by clicking on the plus (+) symbol.

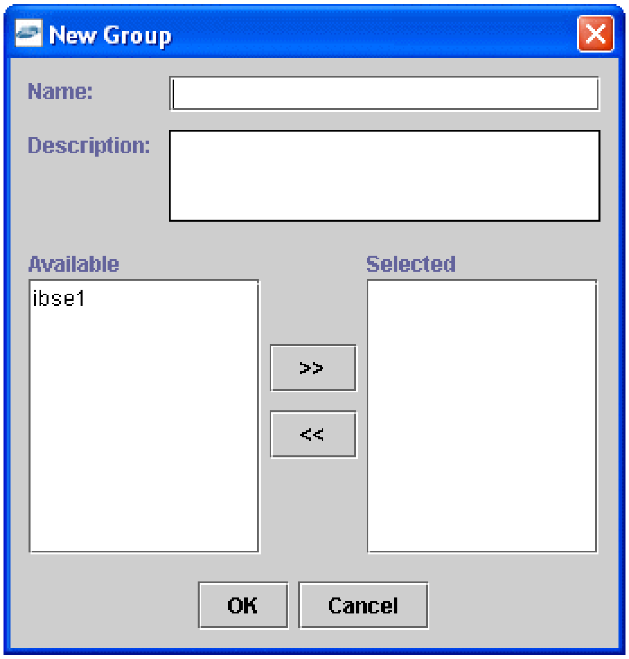
c. Expand the **Security** node by clicking on the plus (+) symbol.

d. Expand the **Users and Groups** node by clicking on the plus (+) symbol.



4. Right-click **Groups** and select **New Group.**

   The New Group dialog box opens.

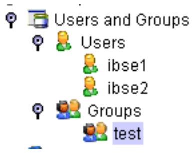


   a. In the Name field, type a a name for the group.

   b. In the Description field, type a description for the group (optional).

   c. From the available list of users in the left pane, select one or more users and add them to the Selected list by clicking the double right facing arrow.

5. When you have selected at least one user, click **OK.**



The new group is added under the Group node.

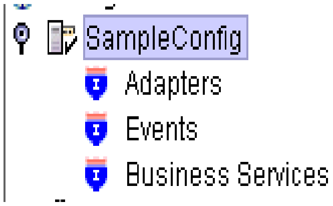
Creating an Execution Policy

An execution policy governs who can execute the Business Services to which the policy is applied.

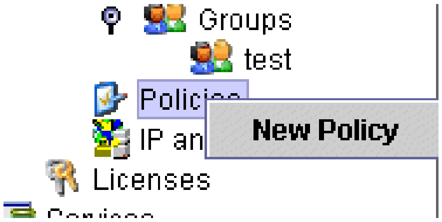
To create an execution policy:

1. Open Application Explorer.
2. Right-click the configuration to which you want to connect, for example, SampleConfig. See Chapter 2, "Adapter Configuration Using OracleAS Adapter Application Explorer" for information on creating a new configuration.
3. Select **Connect.**

   Nodes appear for **Adapters, Events,** and **Business Services** (also known as Web services).

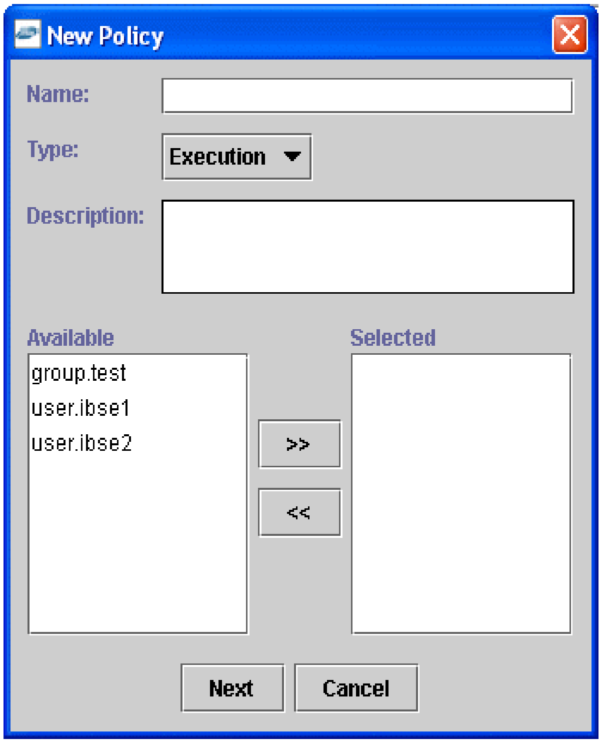


   a. Expand the **Business Services** node by clicking the plus (+) symbol.
   b. Expand the **Configurations** node by clicking the plus (+) symbol.
   c. Expand the **Security** node by clicking the plus (+) symbol.
   d. Expand the **Policies** node by clicking the plus (+) symbol.



4. Right-click **Policies** and select **New Policy.**

   The New policy dialog box opens.



Type: Execution

Available

group.test
user.ibse1
user.ibse2

a. In the **Name** field, type a a name for the policy.

b. From the **Type** list, select **Execution.**

c. In the **Description** field, type a description for the policy (optional).

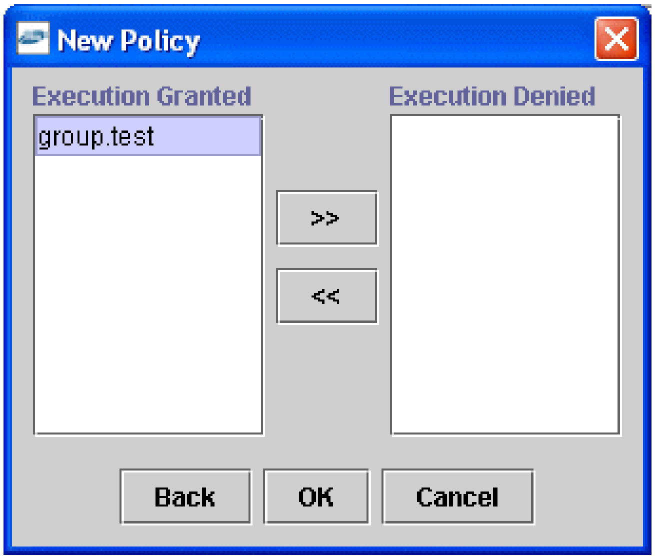
d. From the available list of users in the left pane, select one or more users and add them to the **Selected** list by clicking the double right facing arrow.

> **Note:** This user ID is checked against the value in the user ID element of the SOAP header sent to BSE in a SOAP request.

5. When you have selected at least one user selected, click **OK.**

6. Click **Next.**

The New Policy permissions dialog box opens.



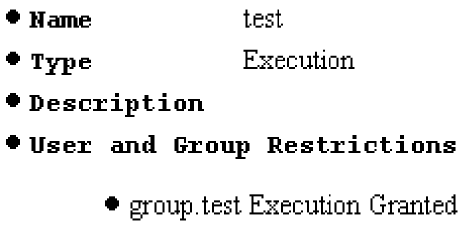
group.test

7. To grant permission to a user or group to execute an iBS, select the user or group and move them into the **Execution Granted** list by selecting the double left facing arrow.

8. To deny permission to a user or group to execute an iBS, select the user or group and move them into the **Execution Denied** list by selecting the double right facing arrow.

9. Click **OK.**

   The following pane summarizes your configuration.



- **Name** test
- **Type** Execution
- **Description**
- **User and Group Restrictions**
  - group.test Execution Granted

Using the IP and Domain Restrictions Policy Type

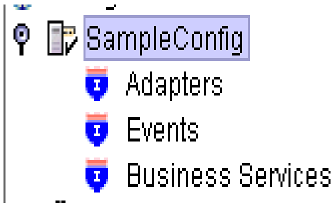
You configure the IP and Domain Restriction policy type slightly differently from other policy types. The IP and Domain Restriction policy type controls connection access to BSE and therefore need not be applied to individual Web services. You need not create a policy, however, you must enable the Security Policy option in Application Explorer.

1. Open Application Explorer.

2. Right-click the configuration to which you want to connect, for example, SampleConfig. See Chapter 2, "Adapter Configuration Using OracleAS Adapter Application Explorer" for information on creating a new configuration.

3. Select **Connect.**

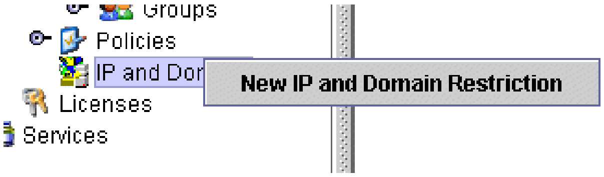
   Nodes appear for **Adapters, Events,** and **Business Services** (also known as Web services).



a. Expand the **Business Services** node by clicking the plus (+) symbol.

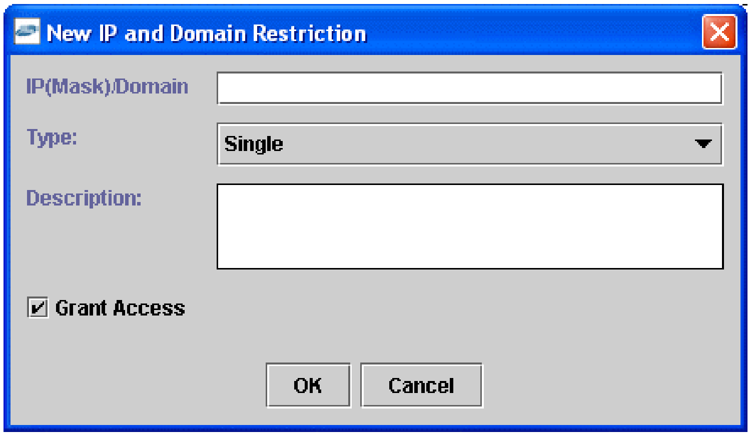
b. Expand the **Configurations** node by clicking the plus (+) symbol.

c. Expand the **Security** node by clicking the plus (+) symbol.



4. Right-click **IP and Domain** and select **New IP and Domain Restriction.**

   The New IP and Domain Restriction dialog box opens.



New IP and Domain Restriction

IP(Mask)/Domain

Type: Single

Description:

☑ Grant Access

a. In the **IP(Mask)/Domain** field, type the IP or domain name using the following guidelines.

   If you select **Single** (Computer) from the **Type** list, you must provide the IP address for that computer. If you only know the DNS name for the computer, click **DNS Lookup** to obtain the IP Address based on the DNS name.

   If you select **Group** (of Computers), you must provide the IP address and subnet mask for the computer group.

   If you select **Domain,** you must provide the domain name, for example, yahoo.com.

b. From the **Type** list, select the type of restriction.

c. In the **Description** field, type a description (optional).

d. To grant access, select the **Grant Access** check box.

5. Click **OK.**

The new domain is added under the IP and Domain node.

The following pane summarizes your configuration.

● **IP Address(Mask)/Domain**	www.yahoo.com
● **Type**	Domain
● **Access**	Denied
● **Description**	

Migrating Repositories

During design time, the Oracle repository is used to store metadata created when using Application Explorer to configure adapter connections, browse EIS objects, configure services, and configure listeners to listen for EIS events. The information in the repository is also referenced at runtime. For management purposes, you can migrate BSE and JCA repositories that are configured for Oracle to new destinations without affecting your existing configuration. For example, you may want to migrate a repository from a test environment to a production environment.

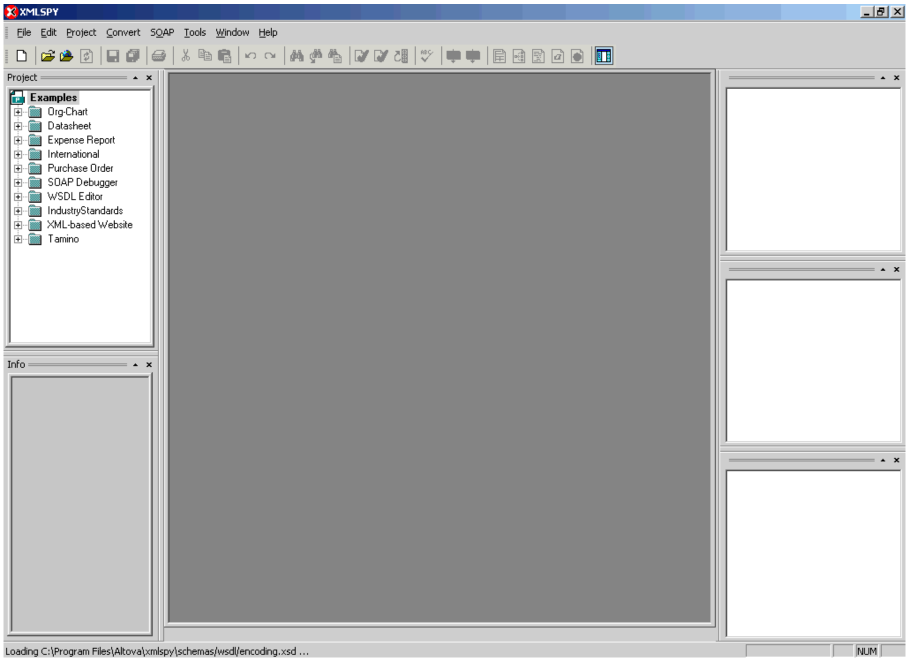
Migrating a BSE Repository

To migrate a BSE repository:

1. Copy the BSE control service URL, for example:

   ```
   http://localhost:7777/ibse/IBSEServlet/admin/iwcontrol.ibs
   ```

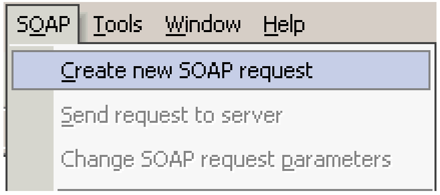
2. Open a third party XML editor, for example, XMLSPY.



Loading C:\Program Files\Altova\xmlspy\schemas\wsdl\encoding.xsd ...

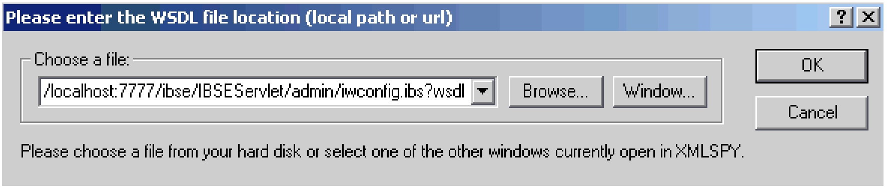
3. In the menu bar, click **SOAP.**

   A list of options appears.



4. Select **Create new SOAP request.**

   The WSDL file location dialog box opens.



Please enter the WSDL file location (local path or url)

Choose a file: /localhost:7777/ibse/IBSEServlet/admin/iwconfig.ibs?wsdl

Please choose a file from your hard disk or select one of the other windows currently open in XMLSPY.

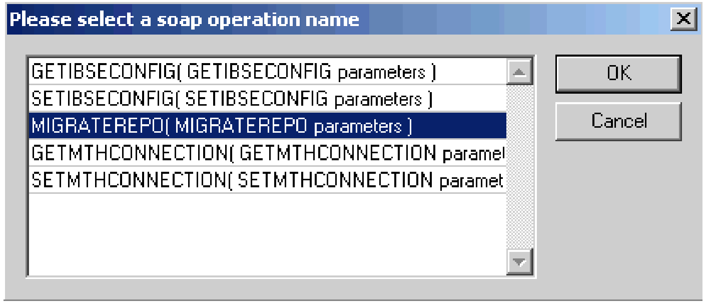
   a. In the **Choose a file** field, paste the BSE control service URL.

   b. Append **?wsdl** to the URL, for example:

```
http://localhost:7777/ibse/IBSEServlet/admin/iwcontrol.ibs?wsdl
```

5. Click **OK.**

   The soap operation name dialog box opens and lists the available control methods.

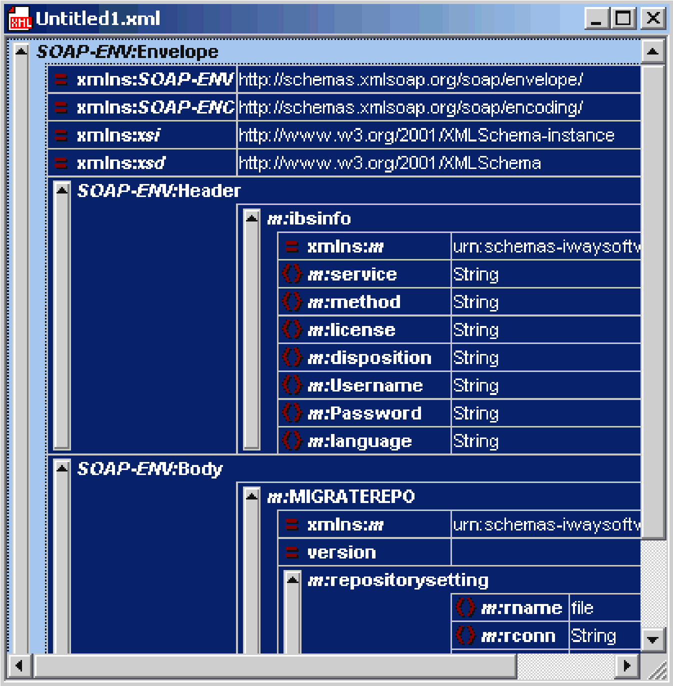


Please select a soap operation name

GETIBSECONFIG(GETIBSECONFIG parameters)
SETIBSECONFIG(SETIBSECONFIG parameters)
MIGRATEREPO(MIGRATEREPO parameters)
GETMTHCONNECTION(GETMTHCONNECTION paramet
SETMTHCONNECTION(SETMTHCONNECTION paramet

6. Select the **MIGRATEREPO(MIGRATEREPO parameters)** control method and click **OK.**

   The following window opens, which shows the structure of the SOAP envelope.



Untitled1.xml

SOAP-ENV:Envelope

xmlns:SOAP-ENV	http://schemas.xmlsoap.org/soap/envelope/
xmlns:SOAP-ENC	http://schemas.xmlsoap.org/soap/encoding/
xmlns:xsi	http://www.w3.org/2001/XMLSchema-instance
xmlns:xsd	http://www.w3.org/2001/XMLSchema

SOAP-ENV:Header

m:ibsinfo

xmlns:m	urn:schemas-iwaysoftw
m:service	String
m:method	String
m:license	String
m:disposition	String
m:Username	String
m:Password	String
m:language	String

SOAP-ENV:Body

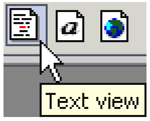
m:MIGRATEREPO

xmlns:m	urn:schemas-iwaysoftw
version	

m:repositorysetting

m:rname	file
m:rconn	String

7. Locate the **Text view** icon in the toolbar.



8. To display the structure of the SOAP envelope as text, click the **Text view** icon.

   The <SOAP-ENV:Header> tag is not required and can be deleted from the SOAP envelope.

9. Locate the following section:

```
<m:MIGRATEREPO xmlns:m="urn:schemas-iwaysoftware-com:jul2003:ibse:config"
version="">
<m:repositorysetting>
<m:rname>oracle</m:rname>
<m:rconn>String</m:rconn>
<m:rdriver>String</m:rdriver>
<m:ruser>String</m:ruser>
<m:rpwd>String</m:rpwd>
</m:repositorysetting>
<m:servicename>String</m:servicename>
</m:MIGRATEREPO>
```

   a. For the <m:rconn> tag, replace the String placeholder with a repository URL where you want to migrate your existing BSE repository.

      The Oracle repository URL has the following format:

      ```
      jdbc:oracle:thin:@[host]:[port]:[sid]
      ```

   b. For the <m:rdriver> tag, replace the String placeholder with the location of your Oracle driver.

c. For the `<m:ruser>` tag, replace the String placeholder with a valid user name to access the Oracle repository.

d. For the `<m:rpwd>` tag, replace the String placeholder with a valid password to access the Oracle repository.

10. Perform one of the following migration options.

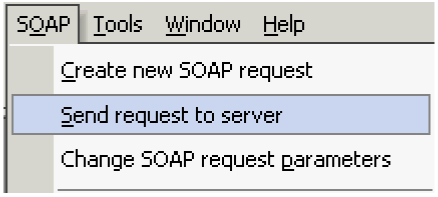
- If you want to migrate a single Web service from the current BSE repository, enter the Web service name in the `<m:servicename>` tag, for example:

```
<m:servicename>SAPService1</m:servicename>
```

- If you want to migrate multiple Web services from the current BSE repository, duplicate the `<m:servicename>` tag for each Web service, for example:

```
<m:servicename>SAPService1</m:servicename>
<m:servicename>SAPService2</m:servicename>
```

- If you want to migrate all Web services from the current BSE repository, remove the `<m:servicename>` tag.



11. In the menu bar, click **SOAP** and select **Send request to server.**

Your BSE repository and any Web services you specified are now migrated to the new Oracle repository URL you specified.

Migrating a JCA Repository

To migrate a JCA repository:

1. Navigate to the location of your JCA configuration directory where the repository schemas and other information is stored, for example:

```
OracleAS_home\adapters\application
```

2. Locate and copy the `repository.xml` file.

3. Place this file in a new JCA configuration directory to migrate the existing repository.

Your JCA repository is migrated to the new JCA configuration directory.

A

Using Siebel Workflows

When using Siebel XML to integrate with Siebel Integration Objects, the interface uses a Siebel Workflow.

> **Note:** This section is intended as a supplement to the documentation designed for the OracleAS Adapter for Siebel user and is not intended as a substitute for Siebel documentation. For complete and up-to-date information on Siebel Workflow and policy topics, see the Siebel Bookshelf for your Siebel system.

Overview

A Siebel Workflow is defined within Siebel to emit or to receive Siebel XML. In either case, emitting or receiving is handled by Siebel transport services for MQSeries, File, or HTTP. The following topics discuss the use and creation of workflows that employ the supported transport services.

Siebel Workflows

A Siebel Workflow is a series of Siebel Business Services linked together to accomplish a business task. You create workflows using the Siebel Client Workflow Administration screens. Workflows are invoked through one of the following methods:

- Using a workflow policy
- Using a runtime event (Siebel Event)
- Using a script (eScript or Siebel VB)

The following topic briefly describes how to invoke the workflow through a policy condition.

> **See Also:**
>
> Siebel Bookshelf documentation for more information on policy and other methods.

Using a Policy to Invoke a Siebel EAI Workflow

A workflow policy is defined by a set of conditions that executes a set of defined actions. A Siebel workflow policy consists of:

- Conditions that define circumstances, based on changes in the state of a Siebel database.
- Actions that define steps taken when conditions are fulfilled.

Creating a policy to invoke a workflow as an action involves the following steps:

1. Define an action to be executed after a policy is triggered. Use the **Run Integration Process** program.
2. Create a policy by setting conditions and selecting appropriate policy groups and actions.
3. Activate the policy by choosing an activation date.
4. Run the **Generate Triggers** server task from **Server Administration** windows to set the conditions to be monitored.
5. Start the **Workflow Monitor** agent after editing with the appropriate policy group (to which your policy belongs) to evaluate whether to perform an action.
6. Start the **Workflow Action Agent** server task from **Server Administration** windows to perform the action.

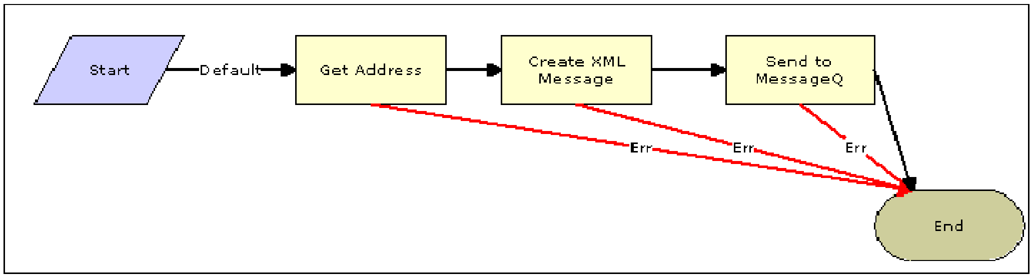
Siebel Workflow - Outbound

When a Siebel Workflow is triggered based on a Siebel policy, runtime, or script (eScript or Siebel VB) event, the result is the generation of a Siebel XML document that is placed on one of the Siebel transports. For example, when you add a new account in the Siebel Call Center application, you can design and configure a workflow to be triggered on the account transaction. You can design the workflow to extract the data for the new record, convert it to Siebel XML, and then, place it on an MQSeries message queue.

In this example, the Siebel Workflow process executes the following series of Siebel Business Services:

1. Calls the Siebel EAI Siebel Adapter that queries for the newly updated account record and places the data in its original internal structure into memory.
2. Calls the Siebel EAI XML Converter that converts the data into an XML message.
3. Calls the Siebel EAI MQSeries Transport that places the newly created XML message into the appropriate MQSeries message queue

   After the message is placed in the message queue, it is retrieved by an OracleAS Adapter for Siebel 6.3 and higher. The following Workflow sequence illustrates the previous steps.



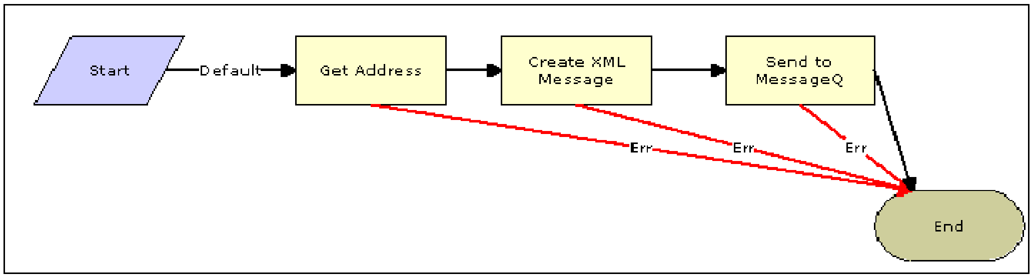
Siebel Workflow - Inbound

A Siebel Workflow that is triggered by an external event begins by receiving a Siebel XML document placed on one of its transports. The result might be the update of a Siebel record using the XML as input, for example, when a new account is added in another CRM system but also must be updated in the Siebel Call Center application. You can design and configure a Workflow to receive or listen on an MQSeries message queue. Upon receipt of the XML message, the Workflow processes the transaction into the Siebel system to update the record.

In this example, upon receipt of the Siebel XML message in the message queue, the Siebel MQSeries Receiver server task initiates a Siebel Workflow process, which in turn executes a series of Siebel Business Services as follows:

1. Calls the Siebel EAI XML Converter, which converts the XML message into Siebel internal format.
2. Calls the Siebel EAI Siebel Adapter, which applies the newly updated account record based on the methods defined in its service.

The following is a sample of the Workflow process.



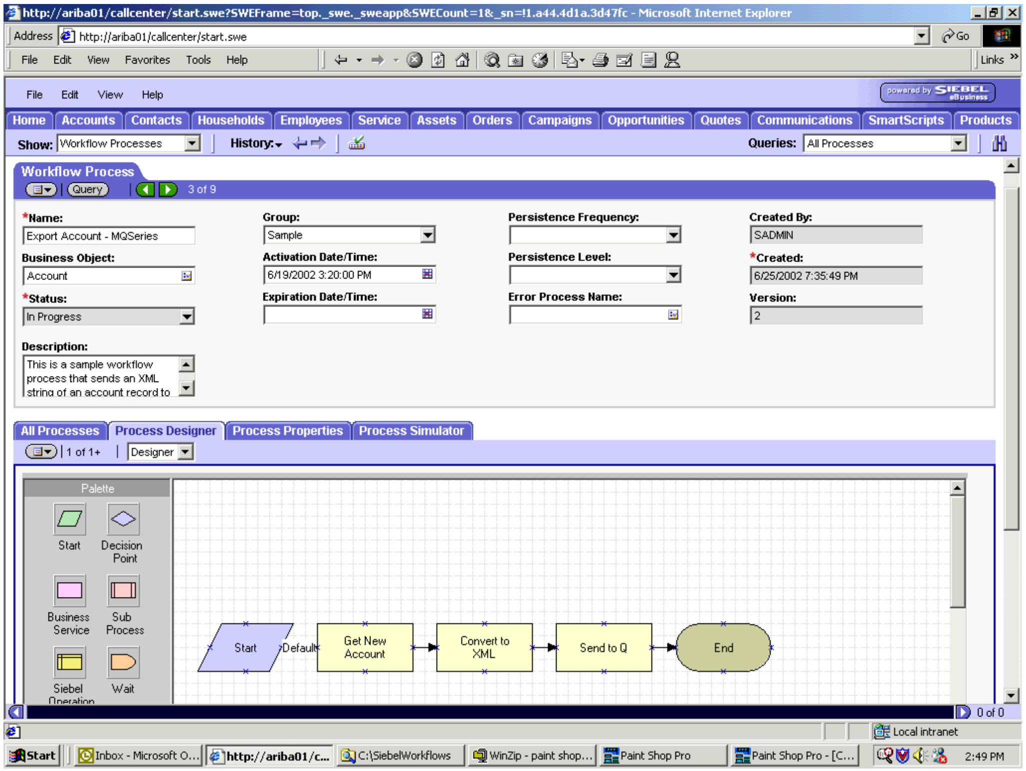
Creating a Siebel Workflow

The following topics include procedures for creating Siebel Workflows in the Siebel Workflow Administration window.

Creating a Siebel Workflow for an Event Using MQSeries Transport

The following procedure is an example of a Siebel Workflow illustrated in the Siebel Workflow Administration window. The Workflow was designed for exporting Siebel Account record information using the MQSeries transport.

The following is a Siebel Workflow Administration window.



*Name: Export Account - MQSeries

Business Object: Account

*Status: In Progress

Description: This is a sample workflow process that sends an XML string of an account record to

Group: Sample

Activation Date/Time: 6/19/2002 3:20:00 PM

Expiration Date/Time:

Persistence Frequency:

Persistence Level:

Error Process Name:

Created By: SADMIN

*Created: 6/25/2002 7:35:49 PM

Version: 2

The following procedure describes how to create a Siebel Workflow that generates Siebel XML when an Account record is updated in the Siebel Call Center application. The Workflow is then placed on an MQSeries message queue.

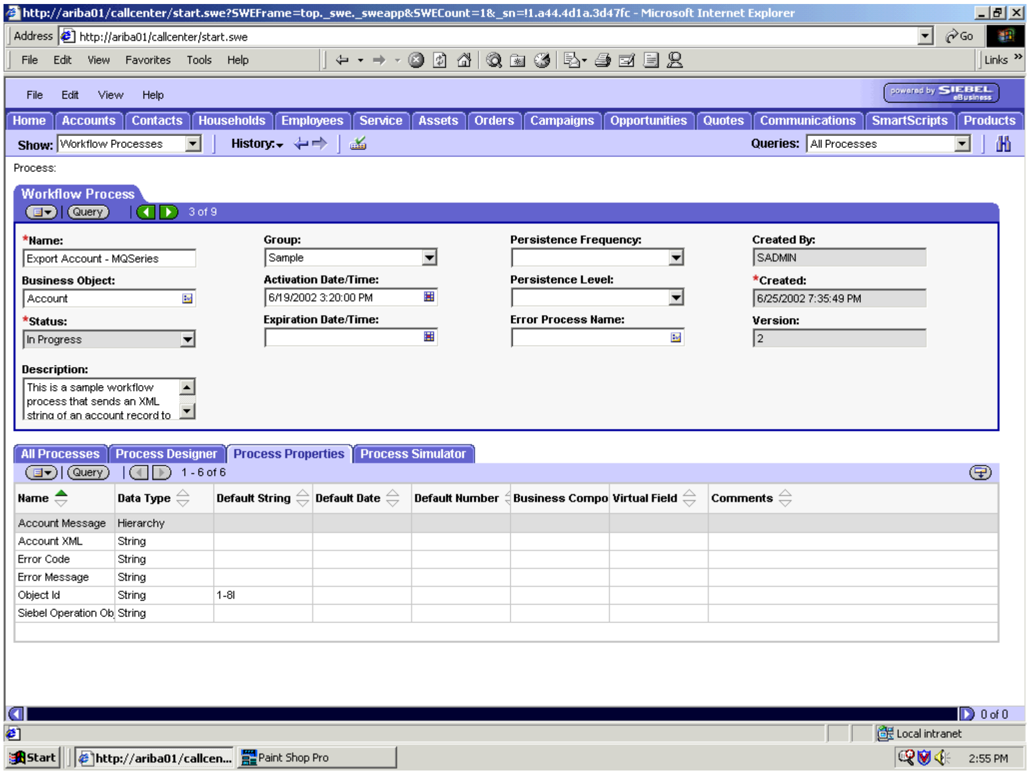
To create a Siebel Workflow:

1. In the **Process Properties** tab of the **Workflow Process** window, define the Account message and Account XML process properties.

   The Account message contains Siebel Account data in hierarchical format.

   Account XML specifies the Siebel Account data that the workflow has converted to XML.

   The following window opens showing the Process Properties tab active.


http://ariba01/callcenter/start.swe?SWEFrame=top._swe._sweapp&SWECount=1&_sn=!1.a44.4d1a.3d47fc - Microsoft Internet Explorer
Address http://ariba01/callcenter/start.swe
File Edit View Favorites Tools Help
Go
Links
File Edit View Help
powered by SIEBEL eBusiness
Home Accounts Contacts Households Employees Service Assets Orders Campaigns Opportunities Quotes Communications SmartScripts Products
History:
Query
Process Designer
Process Simulator
All Processes
Start
http://ariba01/callcen...
Paint Shop Pro
Local intranet
2:55 PM

Show: Workflow Processes

Queries: All Processes

Process:

Workflow Process

3 of 9

*Name: Export Account - MQSeries	Group: Sample	Persistence Frequency:	Created By: SADMIN
Business Object: Account	Activation Date/Time: 6/19/2002 3:20:00 PM	Persistence Level:	*Created: 6/25/2002 7:35:49 PM
*Status: In Progress	Expiration Date/Time:	Error Process Name:	Version: 2

Description:
This is a sample workflow process that sends an XML string of an account record to

Process Properties

1 - 6 of 6

Name	Data Type	Default String	Default Date	Default Number	Business Compo	Virtual Field	Comments
Account Message	Hierarchy						
Account XML	String						
Error Code	String						
Error Message	String						
Object Id	String	1-8I					
Siebel Operation Ob	String						

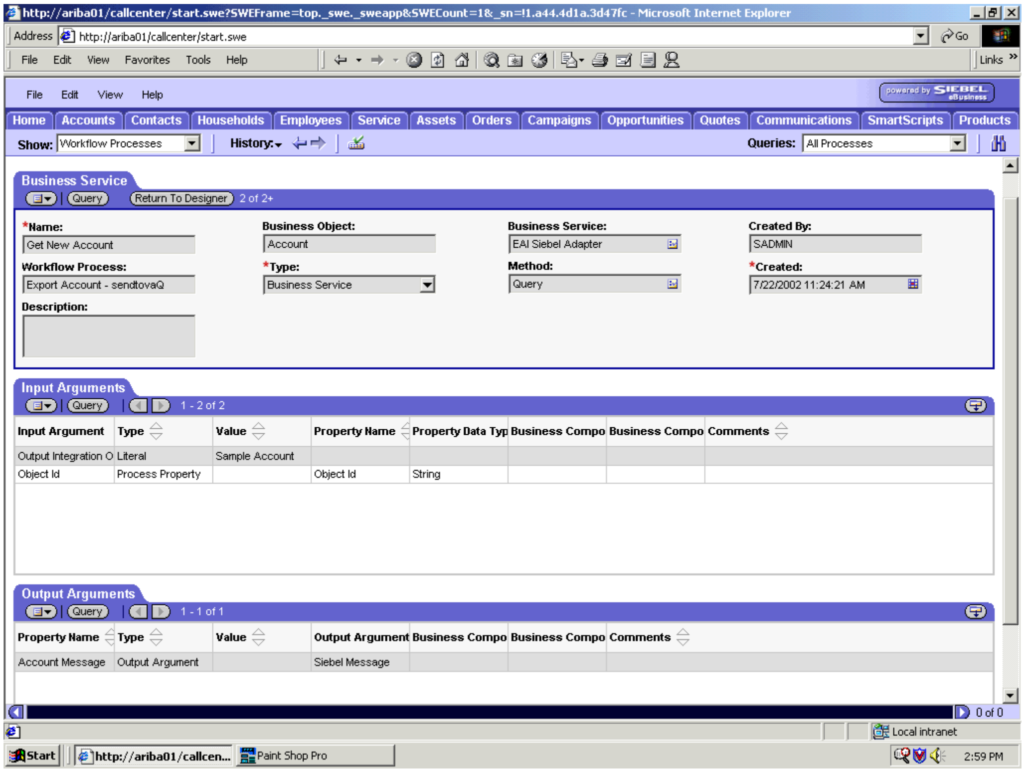
0 of 0

2. Use the **Siebel Workflow Administration** windows to create a Workflow.

3. Define an **EAI Siebel Adapter Business Service** step to receive an instance of Account data and call it `Get New Account`.

   The Business Service obtains the Account information from Siebel using the Query method.

   Output from this Business Service is generated in hierarchical format.


http://ariba01/callcenter/start.swe?SWEFrame=top._swe._sweapp&SWECount=1&_sn=!1.a44.4d1a.3d47fc - Microsoft Internet Explorer
Address http://ariba01/callcenter/start.swe
File Edit View Favorites Tools Help
Links
File Edit View Help
Home Accounts Contacts Households Employees Service Assets Orders Campaigns Opportunities Quotes Communications SmartScripts Products
Show: Workflow Processes
History:
Queries: All Processes
Query
Return To Designer
Local intranet
Start
http://ariba01/callcen...
Paint Shop Pro
2:59 PM

Business Service

2 of 2+

*Name: Get New Account	Business Object: Account	Business Service: EAI Siebel Adapter	Created By: SADMIN
Workflow Process: Export Account - sendtovaQ	*Type: Business Service	Method: Query	*Created: 7/22/2002 11:24:21 AM
Description:			

Input Arguments

1 - 2 of 2

Input Argument	Type	Value	Property Name	Property Data Typ	Business Compo	Business Compo	Comments
Output Integration O	Literal	Sample Account					
Object Id	Process Property		Object Id	String			

Output Arguments

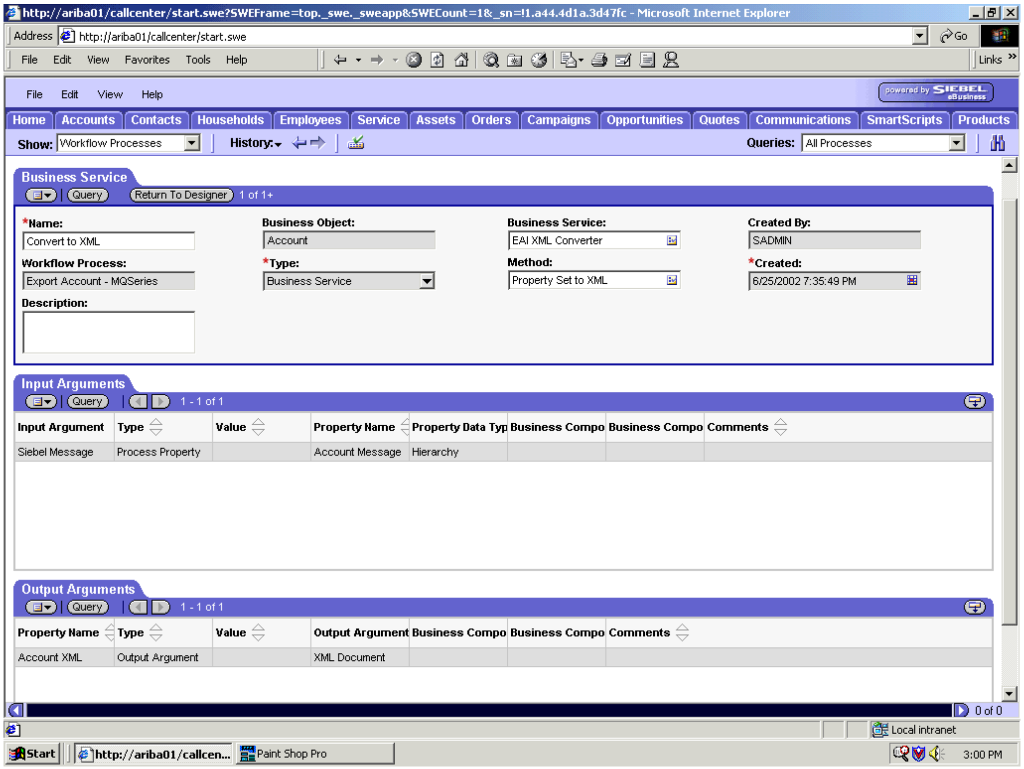
1 - 1 of 1

Property Name	Type	Value	Output Argument	Business Compo	Business Compo	Comments
Account Message	Output Argument		Siebel Message			

0 of 0

4. Define an **EAI XML Converter Business Service** step and call it `Convert to XML.`

   It is defined to receive the Account data from the EAI Siebel Adapter Business Service in hierarchical format and convert it to XML format.


http://ariba01/callcenter/start.swe?SWEFrame=top._swe._sweapp&SWECount=1&_sn=!1.a44.4d1a.3d47fc - Microsoft Internet Explorer
Address http://ariba01/callcenter/start.swe
File Edit View Favorites Tools Help
Links
File Edit View Help
Home Accounts Contacts Households Employees Service Assets Orders Campaigns Opportunities Quotes Communications SmartScripts Products
Show: Workflow Processes
History:
Queries: All Processes
Query
Return To Designer
Local intranet
Start
http://ariba01/callcen...
Paint Shop Pro
3:00 PM

Business Service

1 of 1+

*Name: Convert to XML	Business Object: Account	Business Service: EAI XML Converter	Created By: SADMIN
Workflow Process: Export Account - MQSeries	*Type: Business Service	Method: Property Set to XML	*Created: 6/25/2002 7:35:49 PM
Description:			

Input Arguments

1 - 1 of 1

Input Argument	Type	Value	Property Name	Property Data Typ	Business Compo	Business Compo	Comments
Siebel Message	Process Property		Account Message	Hierarchy			

Output Arguments

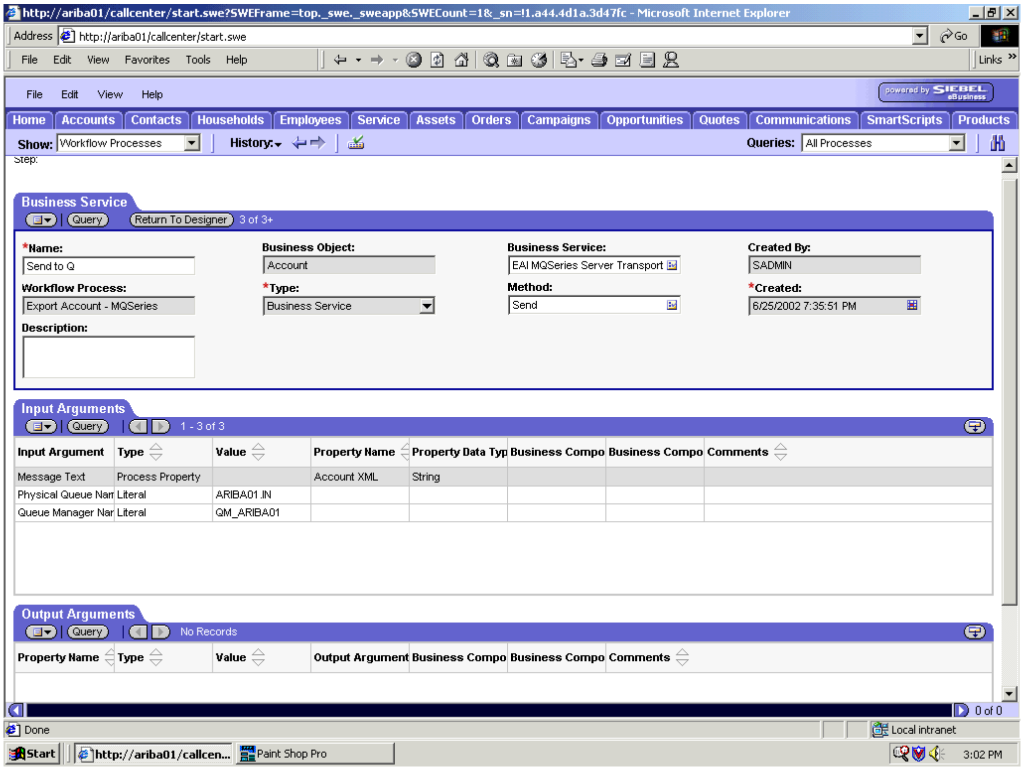
1 - 1 of 1

Property Name	Type	Value	Output Argument	Business Compo	Business Compo	Comments
Account XML	Output Argument		XML Document			

0 of 0

5. Define an **EAI MQSeries** server transport Business Service step and call it `Send to Q`.

   It is defined to receive the Account data from the EAI XML Converter Business Service in Siebel XML format and send the Account XML to MQSeries using the Send method.


http://ariba01/callcenter/start.swe?SWEFrame=top._swe._sweapp&SWECount=1&_sn=!1.a44.4d1a.3d47fc - Microsoft Internet Explorer
Address http://ariba01/callcenter/start.swe
File Edit View Favorites Tools Help
Links
File Edit View Help
powered by SIEBEL eBusiness
Home Accounts Contacts Households Employees Service Assets Orders Campaigns Opportunities Quotes Communications SmartScripts Products
Show: Workflow Processes
History:
Queries: All Processes
Step:
Business Service
Query
Return To Designer
3 of 3+
Input Arguments
1 - 3 of 3
Output Arguments
No Records
0 of 0
Done
Local intranet
Start
http://ariba01/callcen...
Paint Shop Pro
3:02 PM

*Name: Send to Q	Business Object: Account	Business Service: EAI MQSeries Server Transport	Created By: SADMIN
Workflow Process: Export Account - MQSeries	*Type: Business Service	Method: Send	*Created: 6/25/2002 7:35:51 PM
Description:			

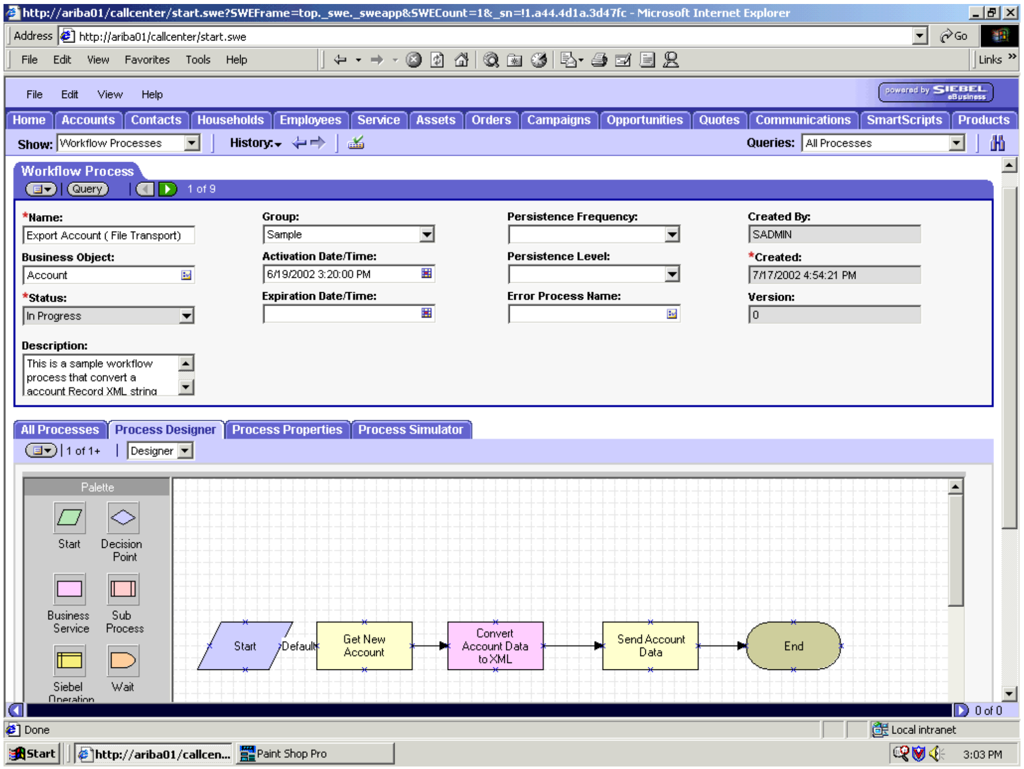
Input Argument	Type	Value	Property Name	Property Data Typ	Business Compo	Business Compo	Comments
Message Text	Process Property		Account XML	String			
Physical Queue Nam	Literal	ARIBA01.IN					
Queue Manager Nar	Literal	QM_ARIBA01					

Property Name	Type	Value	Output Argument	Business Compo	Business Compo	Comments

Creating a Siebel Workflow for an Event Using File Transport

The following procedure is an example of a Siebel Workflow illustrated in the Siebel Workflow Administration window. The Workflow was designed for exporting Siebel Account record information using the File transport.

The following window opens with the Process Designer tab active.


http://ariba01/callcenter/start.swe?SWEFrame=top._swe._sweapp&SWECount=1&_sn=!1.a44.4d1a.3d47fc - Microsoft Internet Explorer
Address http://ariba01/callcenter/start.swe
File Edit View Favorites Tools Help
Links
File Edit View Help
Home Accounts Contacts Households Employees Service Assets Orders Campaigns Opportunities Quotes Communications SmartScripts Products
Show: Workflow Processes
History:
Queries: All Processes
Workflow Process
Query
1 of 9

*Name: Export Account (File Transport)

Business Object: Account

*Status: In Progress

Description: This is a sample workflow process that convert a account Record XML string

Group: Sample

Activation Date/Time: 6/19/2002 3:20:00 PM

Expiration Date/Time:

Persistence Frequency:

Persistence Level:

Error Process Name:

Created By: SADMIN

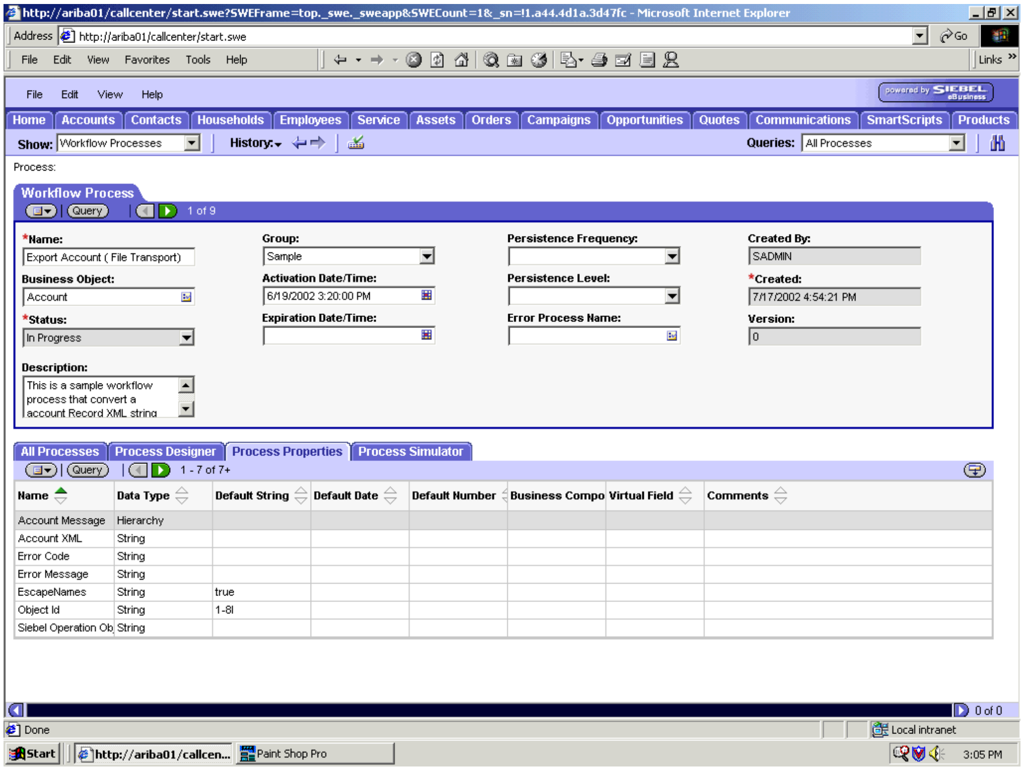
*Created: 7/17/2002 4:54:21 PM

Version: 0

All Processes
Process Designer
Process Properties
Process Simulator
1 of 1+
Designer
Palette
Start
Decision Point
Business Service
Sub Process
Siebel Operation
Wait
Start
Default
Get New Account
Convert Account Data to XML
Send Account Data
End
0 of 0
Done
Local intranet
Start
http://ariba01/callcen...
Paint Shop Pro
3:03 PM

This procedure describes how to create a Siebel Workflow that generates Siebel XML when an Account record is updated in the Siebel Call Center application and then places Siebel XML on the file system.

To create a Siebel Workflow:


http://ariba01/callcenter/start.swe?SWEFrame=top._swe._sweapp&SWECount=1&_sn=!1.a44.4d1a.3d47fc - Microsoft Internet Explorer
Address http://ariba01/callcenter/start.swe
File Edit View Favorites Tools Help
Links
File Edit View Help
Home Accounts Contacts Households Employees Service Assets Orders Campaigns Opportunities Quotes Communications SmartScripts Products
Show: Workflow Processes
History:
Queries: All Processes
Process:
Workflow Process
Query
1 of 9

*Name: Export Account (File Transport)

Business Object: Account

*Status: In Progress

Description: This is a sample workflow process that convert a account Record XML string

Group: Sample

Activation Date/Time: 6/19/2002 3:20:00 PM

Expiration Date/Time:

Persistence Frequency:

Persistence Level:

Error Process Name:

Created By: SADMIN

*Created: 7/17/2002 4:54:21 PM

Version: 0

All Processes
Process Designer
Process Properties
Process Simulator
Query
1 - 7 of 7+

Name	Data Type	Default String	Default Date	Default Number	Business Compo	Virtual Field	Comments
Account Message	Hierarchy						
Account XML	String						
Error Code	String						
Error Message	String						
EscapeNames	String	true					
Object Id	String	1-8I					
Siebel Operation Ob	String						

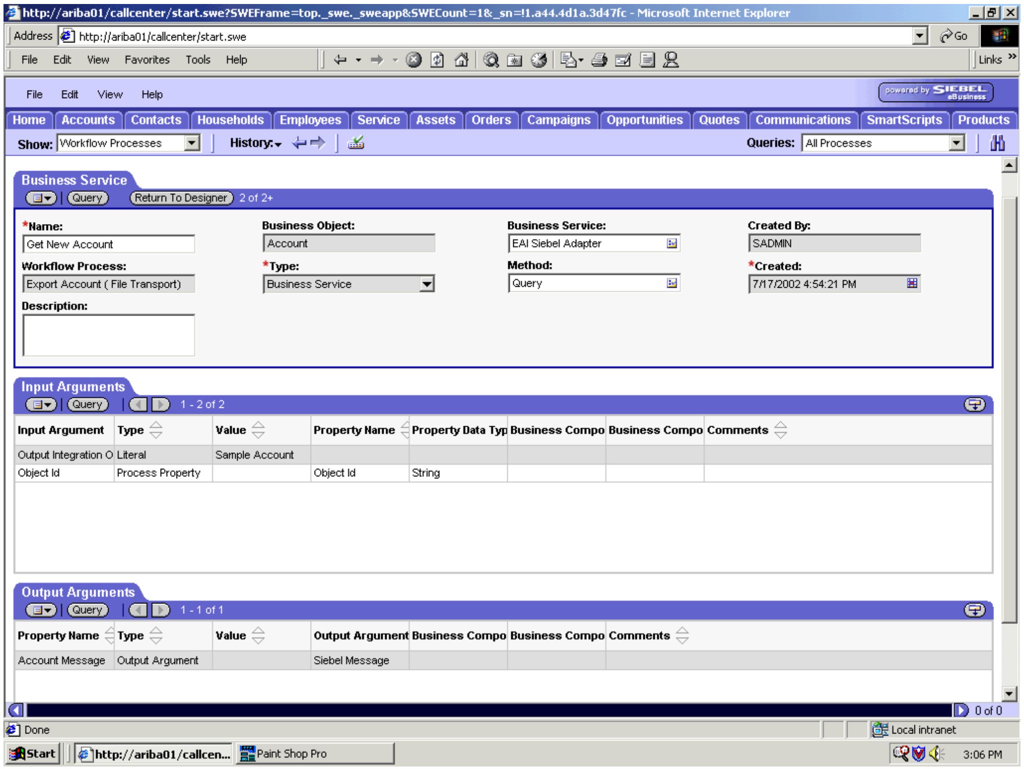
1. On the **Process Properties** tab of the **Workflow Process** window, define the Account message and Account XML process properties.

   Account message contains the Siebel Account data in hierarchical format.

   Account XML specifies which Siebel Account data the Workflow converted to XML.

2. Use the **Siebel Workflow Administration** windows to create a Workflow.

   The following is an example of a Siebel Workflow Administration window.



http://ariba01/callcenter/start.swe?SWEFrame=top._swe._sweapp&SWECount=1&_sn=!1.a44.4d1a.3d47fc - Microsoft Internet Explorer
Address http://ariba01/callcenter/start.swe
File Edit View Favorites Tools Help
Links
File Edit View Help
powered by SIEBEL eBusiness
Home Accounts Contacts Households Employees Service Assets Orders Campaigns Opportunities Quotes Communications SmartScripts Products
Show: Workflow Processes
History:
Queries: All Processes
Query
Return To Designer
2 of 2+

Business Service

*Name: Get New Account	Business Object: Account	Business Service: EAI Siebel Adapter	Created By: SADMIN
Workflow Process: Export Account (File Transport)	*Type: Business Service	Method: Query	*Created: 7/17/2002 4:54:21 PM
Description:			

Input Arguments

Input Argument	Type	Value	Property Name	Property Data Typ	Business Compo	Business Compo	Comments
Output Integration O	Literal	Sample Account					
Object Id	Process Property		Object Id	String			

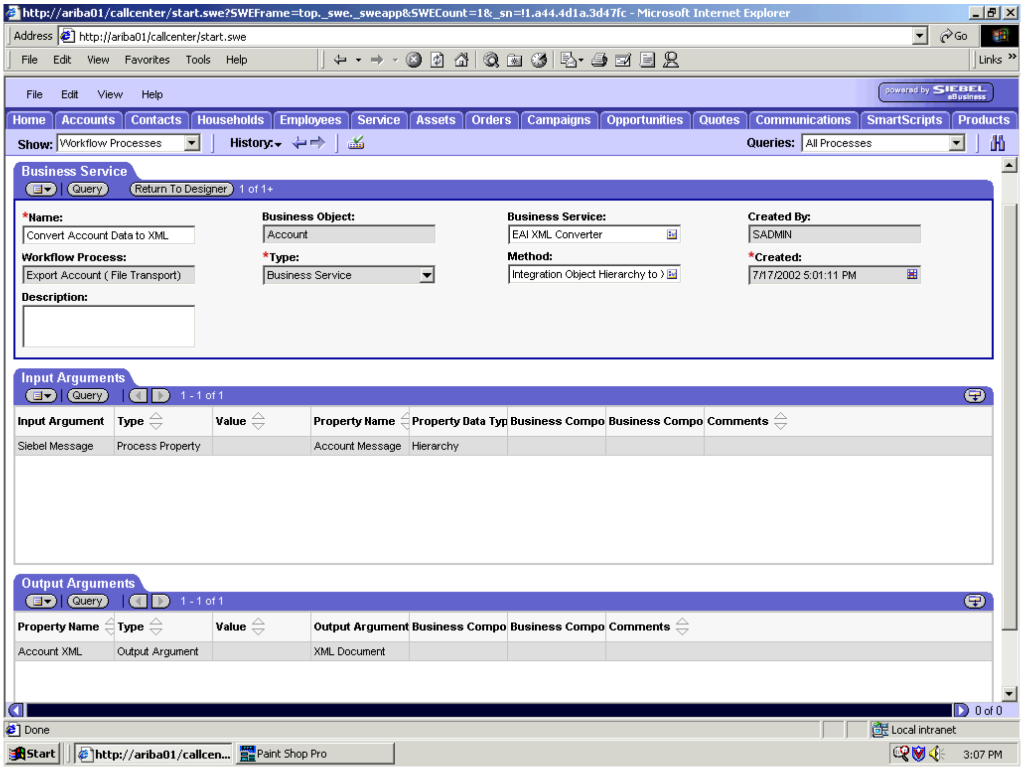
Output Arguments

Property Name	Type	Value	Output Argument	Business Compo	Business Compo	Comments
Account Message	Output Argument		Siebel Message			

3. Define an **EAI Siebel Adapter Business Service** step to receive an instance of Account data and call it `Get New Account`.

   The Business Service obtains the Account information from Siebel using the Query method.

   Output from this Business Service is generated in hierarchical format.


http://ariba01/callcenter/start.swe?SWEFrame=top._swe._sweapp&SWECount=1&_sn=!1.a44.4d1a.3d47fc - Microsoft Internet Explorer
Address http://ariba01/callcenter/start.swe
File Edit View Favorites Tools Help
Links
File Edit View Help
powered by SIEBEL eBusiness
Home Accounts Contacts Households Employees Service Assets Orders Campaigns Opportunities Quotes Communications SmartScripts Products
Show: Workflow Processes History: Queries: All Processes
Query Return To Designer 1 of 1+
Done Local intranet
Start http://ariba01/callcen... Paint Shop Pro 3:07 PM

Business Service

***Name:** Convert Account Data to XML	**Business Object:** Account	**Business Service:** EAI XML Converter	**Created By:** SADMIN
Workflow Process: Export Account (File Transport)	***Type:** Business Service	**Method:** Integration Object Hierarchy to X	***Created:** 7/17/2002 5:01:11 PM
Description:			

Input Arguments 1 - 1 of 1

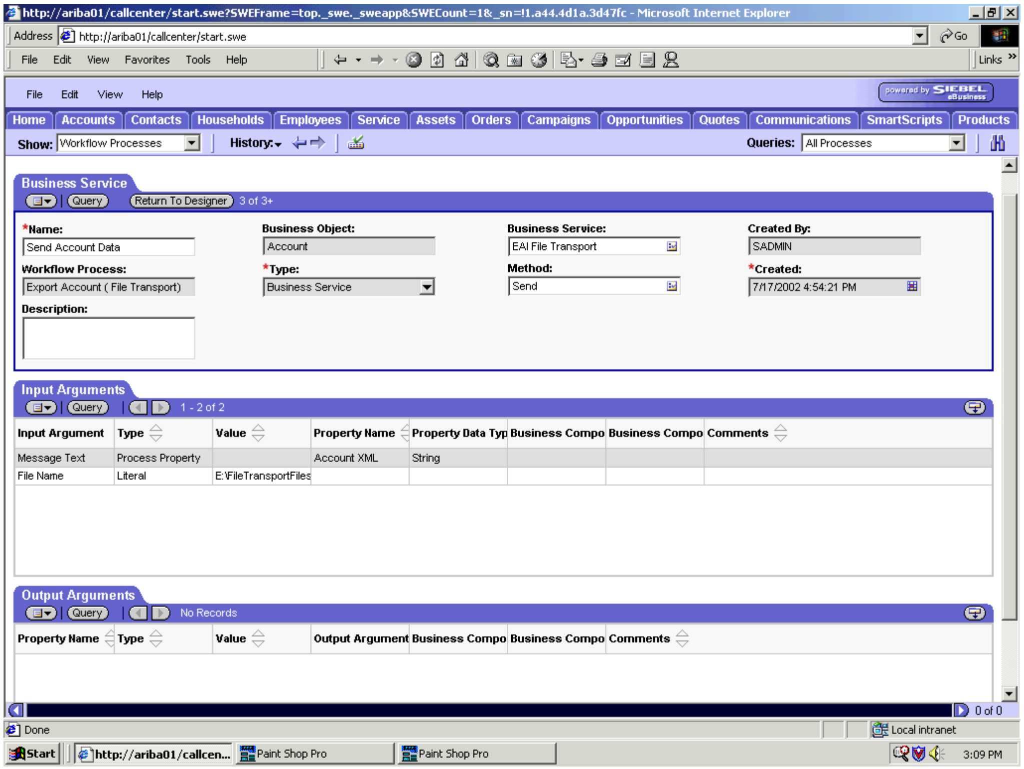
Input Argument	Type	Value	Property Name	Property Data Typ	Business Compo	Business Compo	Comments
Siebel Message	Process Property		Account Message	Hierarchy			

Output Arguments 1 - 1 of 1

Property Name	Type	Value	Output Argument	Business Compo	Business Compo	Comments
Account XML	Output Argument		XML Document			

4. Define an **EAI XML Converter Business Service** step and call it `Convert Account Data to XML`.

   This Business Service is defined to receive the Account data from the EAI Siebel Adapter Business Service in hierarchical format and convert it to XML format.


http://ariba01/callcenter/start.swe?SWEFrame=top._swe._sweapp&SWECount=1&_sn=!1.a44.4d1a.3d47fc - Microsoft Internet Explorer
Address http://ariba01/callcenter/start.swe
File Edit View Favorites Tools Help
Links
File Edit View Help
powered by SIEBEL eBusiness
Home Accounts Contacts Households Employees Service Assets Orders Campaigns Opportunities Quotes Communications SmartScripts Products
Show: Workflow Processes History: Queries: All Processes
Query Return To Designer 3 of 3+
Done Local intranet
Start http://ariba01/callcen... Paint Shop Pro Paint Shop Pro 3:09 PM

Business Service

***Name:** Send Account Data	**Business Object:** Account	**Business Service:** EAI File Transport	**Created By:** SADMIN
Workflow Process: Export Account (File Transport)	***Type:** Business Service	**Method:** Send	***Created:** 7/17/2002 4:54:21 PM
Description:			

Input Arguments 1 - 2 of 2

Input Argument	Type	Value	Property Name	Property Data Typ	Business Compo	Business Compo	Comments
Message Text	Process Property		Account XML	String			
File Name	Literal	E:\FileTransportFiles					

Output Arguments No Records

Property Name	Type	Value	Output Argument	Business Compo	Business Compo	Comments

5. Define an **EAI File Transport Business Service** step and call it `Send Account Data.`

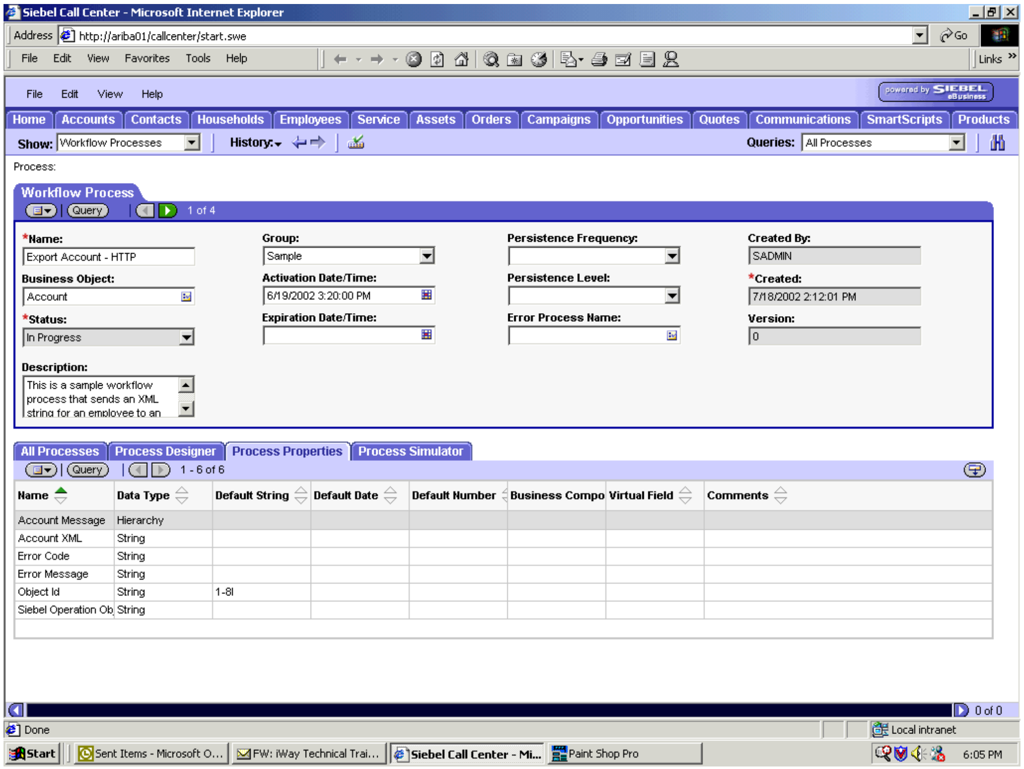
   This Business Service is defined to receive the Account data from the EAI XML Converter Business Service in Siebel XML format and send the Account XML to the file system in a specified directory using the Send method.

Creating a Siebel Workflow for an Event Using HTTP Transport

The following procedure is an example of a Siebel Workflow illustrated in the Siebel Workflow Administration window. The Workflow was designed for exporting Siebel Account record information using the HTTP transport.

This procedure describes how to create a Siebel Workflow that generates Siebel XML when an Account record is updated in the Siebel Call Center application.

To create a Siebel Workflow:


Siebel Call Center - Microsoft Internet Explorer
Address http://ariba01/callcenter/start.swe
Go
File Edit View Favorites Tools Help
Links
File Edit View Help
powered by SIEBEL eBusiness
Home Accounts Contacts Households Employees Service Assets Orders Campaigns Opportunities Quotes Communications SmartScripts Products
History:
Workflow Process
Query
1 of 4
All Processes Process Designer Process Properties Process Simulator
Query
1 - 6 of 6
0 of 0
Done
Local intranet
Start
Sent Items - Microsoft O...
FW: iWay Technical Trai...
Siebel Call Center - Mi...
Paint Shop Pro
6:05 PM

Show: Workflow Processes

Queries: All Processes

Process:

*Name: Export Account - HTTP	Group: Sample	Persistence Frequency:	Created By: SADMIN
Business Object: Account	Activation Date/Time: 6/19/2002 3:20:00 PM	Persistence Level:	*Created: 7/18/2002 2:12:01 PM
*Status: In Progress	Expiration Date/Time:	Error Process Name:	Version: 0

Description: This is a sample workflow process that sends an XML string for an employee to an

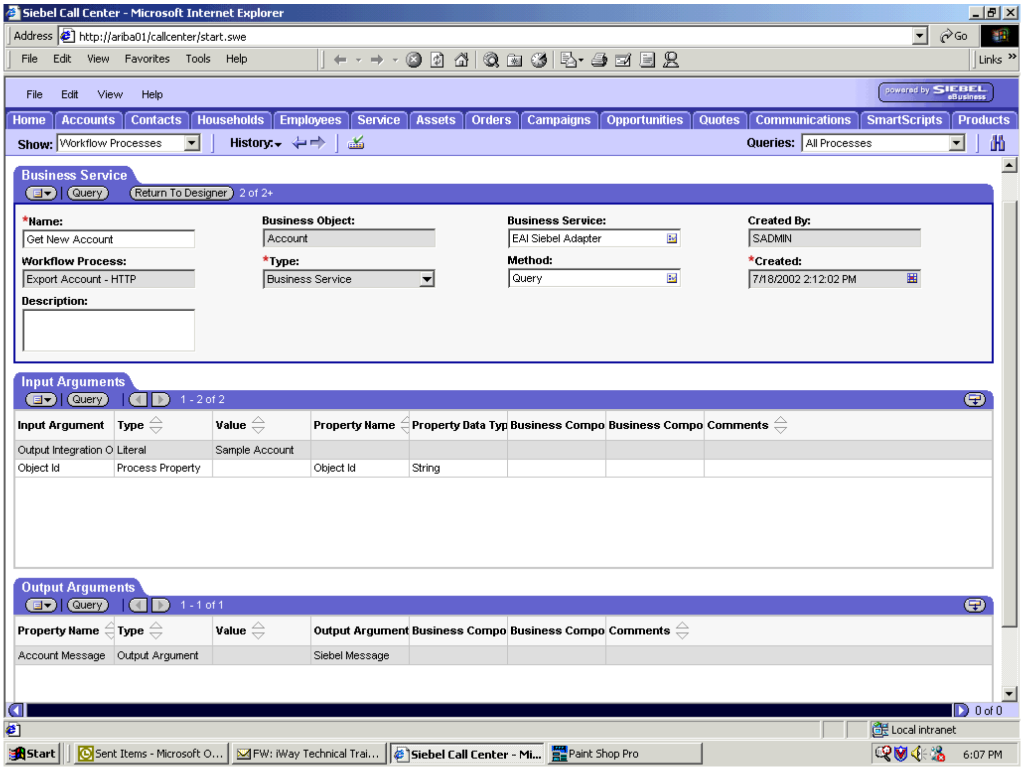
Name	Data Type	Default String	Default Date	Default Number	Business Compo	Virtual Field	Comments
Account Message	Hierarchy						
Account XML	String						
Error Code	String						
Error Message	String						
Object Id	String	1-8I					
Siebel Operation Ob	String						

1. In the **Process Properties** tab of the **Workflow Process** window, define the Account message and Account XML process properties.

   Account message contains the Siebel Account data in hierarchical format.

   Account XML specifies the Siebel Account data that the Workflow has converted to XML.

2. Use the **Siebel Workflow Administration** windows to create a Workflow.


Siebel Call Center - Microsoft Internet Explorer
Address http://ariba01/callcenter/start.swe
File Edit View Favorites Tools Help
Links
File Edit View Help
powered by SIEBEL eBusiness
Home Accounts Contacts Households Employees Service Assets Orders Campaigns Opportunities Quotes Communications SmartScripts Products
History:
Query
Return To Designer
2 of 2+
1 - 2 of 2
1 - 1 of 1
0 of 0
Local intranet
Start
Sent Items - Microsoft O...
FW: iWay Technical Trai...
Siebel Call Center - Mi...
Paint Shop Pro
6:07 PM

Show: Workflow Processes

Queries: All Processes

Business Service

*Name: Get New Account	Business Object: Account	Business Service: EAI Siebel Adapter	Created By: SADMIN
Workflow Process: Export Account - HTTP	*Type: Business Service	Method: Query	*Created: 7/18/2002 2:12:02 PM
Description:			

Input Arguments

Input Argument	Type	Value	Property Name	Property Data Typ	Business Compo	Business Compo	Comments
Output Integration O	Literal	Sample Account					
Object Id	Process Property		Object Id	String			

Output Arguments

Property Name	Type	Value	Output Argument	Business Compo	Business Compo	Comments
Account Message	Output Argument		Siebel Message			

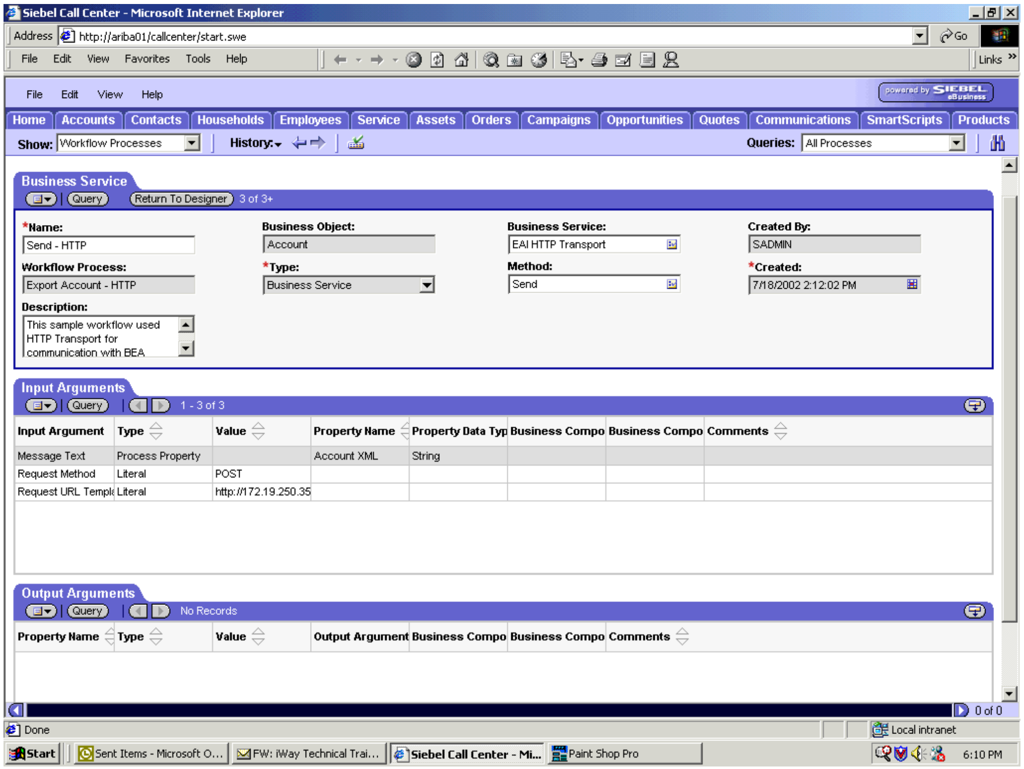
3. Define an **EAI Siebel Adapter Business Service** step to receive an instance of Account data and call it `Get New Account`.

   The Business Service obtains the Account information from Siebel using the Query method.

   Output from this Business Service is generated in hierarchical format.

4. Define an **EAI XML Converter Business Service** step and call it `Convert to XML`.

   This Business Service is defined to receive the Account data from the EAI Siebel Adapter Business Service in hierarchical format and convert it to XML format.



Siebel Call Center - Microsoft Internet Explorer
Address http://ariba01/callcenter/start.swe
File Edit View Favorites Tools Help
File Edit View Help
Home Accounts Contacts Households Employees Service Assets Orders Campaigns Opportunities Quotes Communications SmartScripts Products
Show: Workflow Processes
History:
Queries: All Processes
Business Service
Query
Return To Designer
3 of 3+
Input Arguments
1 - 3 of 3
Output Arguments
No Records
0 of 0
Done
Local intranet
Start
Sent Items - Microsoft O...
FW: iWay Technical Trai...
Siebel Call Center - Mi...
Paint Shop Pro
6:10 PM

*Name: Send - HTTP	Business Object: Account	Business Service: EAI HTTP Transport	Created By: SADMIN
Workflow Process: Export Account - HTTP	*Type: Business Service	Method: Send	*Created: 7/18/2002 2:12:02 PM
Description: This sample workflow used HTTP Transport for communication with BEA			

Input Argument	Type	Value	Property Name	Property Data Typ	Business Compo	Business Compo	Comments
Message Text	Process Property		Account XML	String			
Request Method	Literal	POST					
Request URL Templ	Literal	http://172.19.250.35					

Property Name	Type	Value	Output Argument	Business Compo	Business Compo	Comments

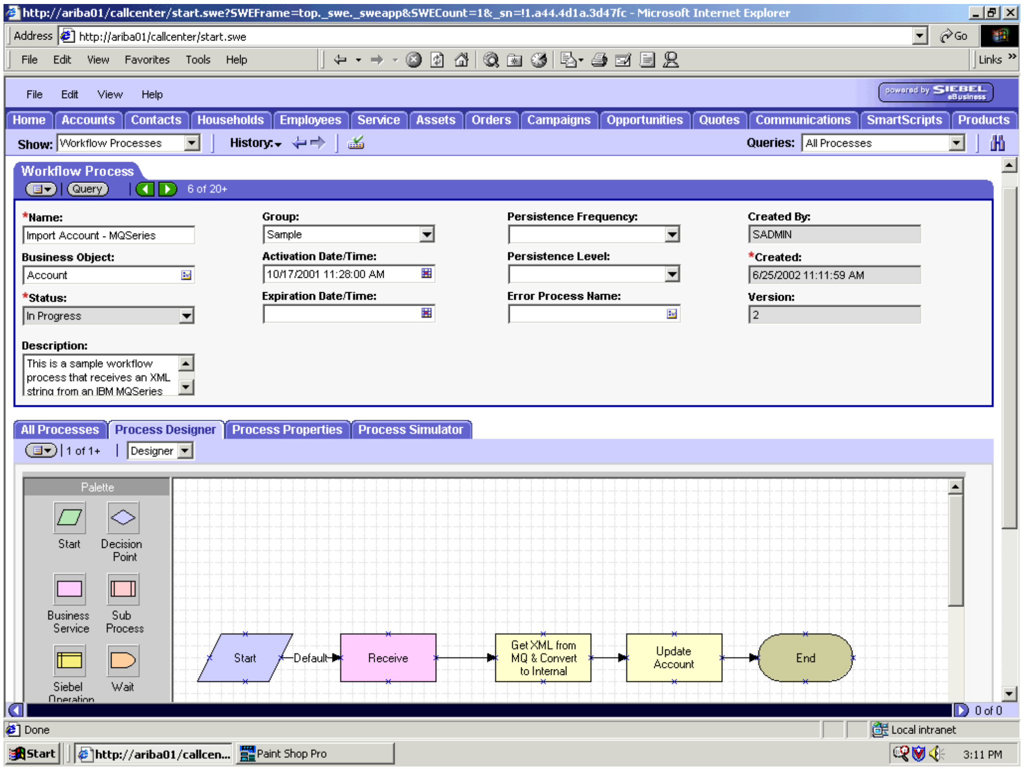
5. Define an **EAI HTTP Transport Business Service** step and call it `Send - HTTP`.

   This Business Service is defined to receive the Account data from the EAI XML Converter Business Service in Siebel XML format and send the Account XML to HTTP using the Send method.

Creating a Siebel Workflow for a Service Using MQSeries Transport

The following procedure is an example of a Siebel Workflow illustrated in the Siebel Workflow Administration window. The Workflow was designed for importing Siebel Account record information through the MQSeries Transport.

The following is a sample Siebel Workflow Administration window.

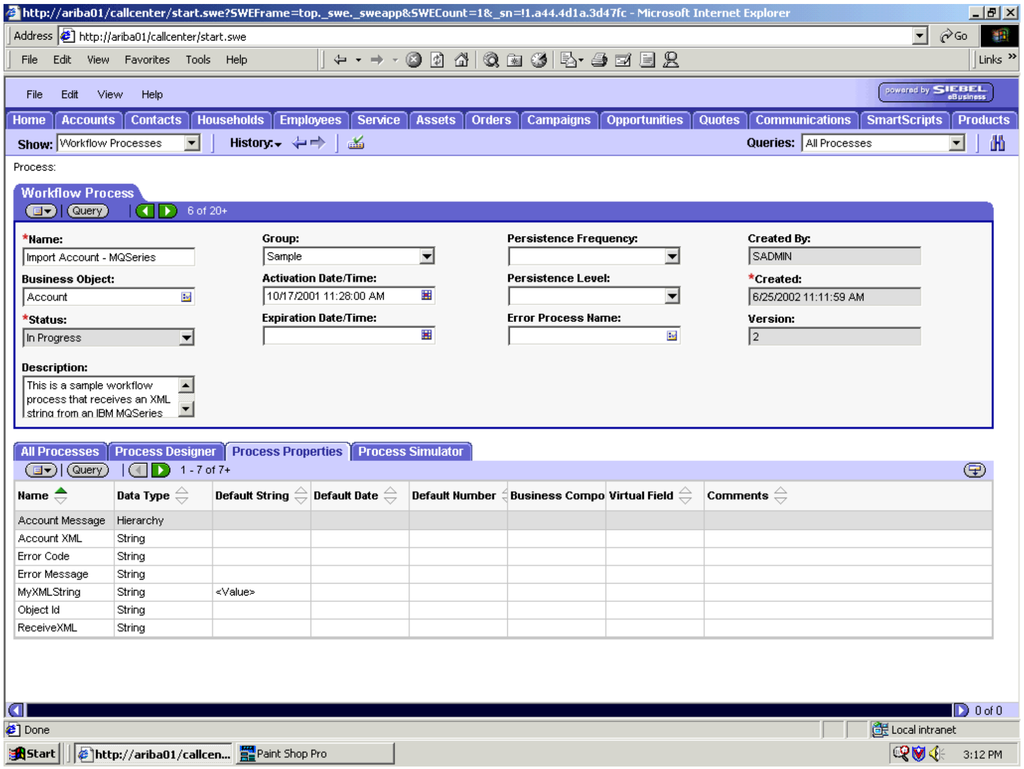

http://ariba01/callcenter/start.swe?SWEFrame=top._swe._sweapp&SWECount=1&_sn=!1.a44.4d1a.3d47fc - Microsoft Internet Explorer
Address http://ariba01/callcenter/start.swe
File Edit View Favorites Tools Help
Home Accounts Contacts Households Employees Service Assets Orders Campaigns Opportunities Quotes Communications SmartScripts Products
Show: Workflow Processes
History:
Queries: All Processes
Workflow Process
Query
6 of 20+
All Processes
Process Designer
Process Properties
Process Simulator
1 of 1+
Designer
Palette
Start
Decision Point
Business Service
Sub Process
Siebel Operation
Wait
Start
Default
Receive
Get XML from MQ & Convert to Internal
Update Account
End
0 of 0
Done
Local intranet
Start
http://ariba01/callcen...
Paint Shop Pro
3:11 PM

*Name: Import Account - MQSeries	Group: Sample	Persistence Frequency:	Created By: SADMIN
Business Object: Account	Activation Date/Time: 10/17/2001 11:28:00 AM	Persistence Level:	*Created: 6/25/2002 11:11:59 AM
*Status: In Progress	Expiration Date/Time:	Error Process Name:	Version: 2

Description: This is a sample workflow process that receives an XML string from an IBM MQSeries

This procedure describes how to create a Siebel Workflow that generates Siebel XML when an Account record is updated in the Siebel Call Center application.

To create a Siebel Workflow:


http://ariba01/callcenter/start.swe?SWEFrame=top._swe._sweapp&SWECount=1&_sn=!1.a44.4d1a.3d47fc - Microsoft Internet Explorer
Address http://ariba01/callcenter/start.swe
File Edit View Favorites Tools Help
Home Accounts Contacts Households Employees Service Assets Orders Campaigns Opportunities Quotes Communications SmartScripts Products
Show: Workflow Processes
History:
Queries: All Processes
Process:
Workflow Process
Query
6 of 20+
All Processes
Process Designer
Process Properties
Process Simulator
Query
1 - 7 of 7+
0 of 0
Done
Local intranet
Start
http://ariba01/callcen...
Paint Shop Pro
3:12 PM

*Name: Import Account - MQSeries	Group: Sample	Persistence Frequency:	Created By: SADMIN
Business Object: Account	Activation Date/Time: 10/17/2001 11:28:00 AM	Persistence Level:	*Created: 6/25/2002 11:11:59 AM
*Status: In Progress	Expiration Date/Time:	Error Process Name:	Version: 2

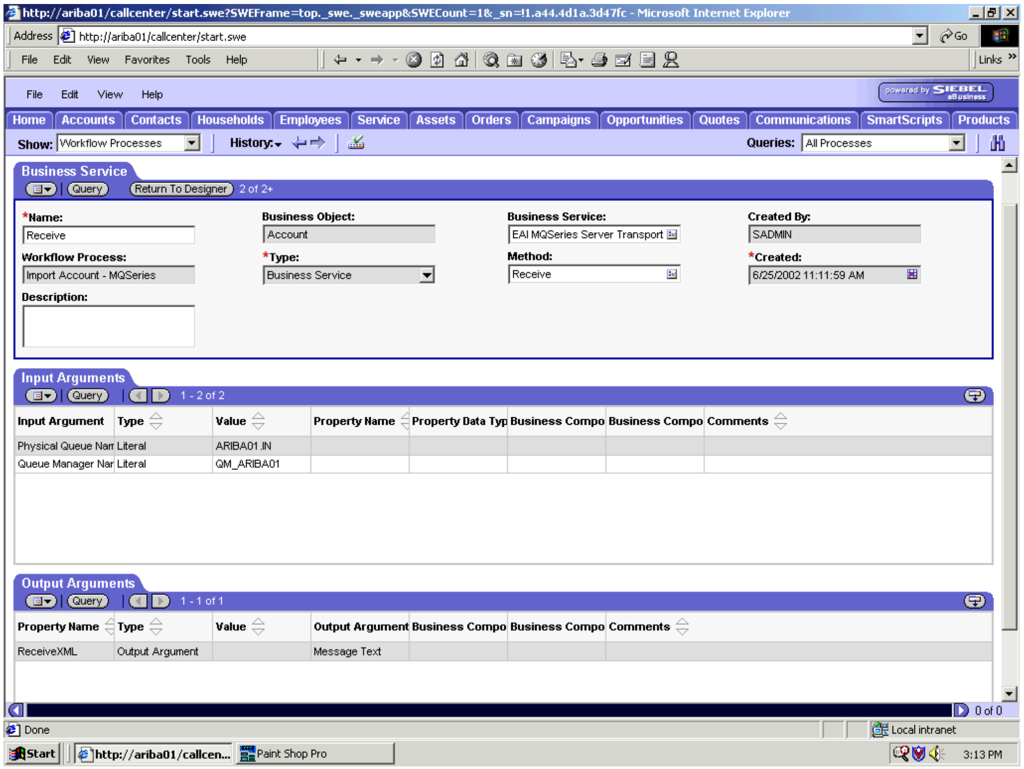
Description: This is a sample workflow process that receives an XML string from an IBM MQSeries

Name	Data Type	Default String	Default Date	Default Number	Business Compo	Virtual Field	Comments
Account Message	Hierarchy						
Account XML	String						
Error Code	String						
Error Message	String						
MyXMLString	String	<Value>					
Object Id	String						
ReceiveXML	String						

1. In the **Process Properties** tab of the **Workflow Process** window, define the Account message and Account XML process properties.

   Account message contains the Siebel Account data in hierarchical format.

   Account XML specifies the Siebel Account data that the Workflow converted to XML.


http://ariba01/callcenter/start.swe?SWEFrame=top._swe._sweapp&SWECount=1&_sn=!1.a44.4d1a.3d47fc - Microsoft Internet Explorer
Address http://ariba01/callcenter/start.swe
Go
File Edit View Favorites Tools Help
Links
File Edit View Help
powered by SIEBEL eBusiness
Home Accounts Contacts Households Employees Service Assets Orders Campaigns Opportunities Quotes Communications SmartScripts Products
Show: Workflow Processes
History:
Queries: All Processes
Business Service
Query
Return To Designer
2 of 2+
Input Arguments
Query
1 - 2 of 2
Output Arguments
Query
1 - 1 of 1
0 of 0
Done
Local intranet
Start
http://ariba01/callcen...
Paint Shop Pro
3:13 PM

Business Service

*Name: Receive	Business Object: Account	Business Service: EAI MQSeries Server Transport	Created By: SADMIN
Workflow Process: Import Account - MQSeries	*Type: Business Service	Method: Receive	*Created: 6/25/2002 11:11:59 AM
Description:			

Input Arguments

Input Argument	Type	Value	Property Name	Property Data Typ	Business Compo	Business Compo	Comments
Physical Queue Nar	Literal	ARIBA01.IN					
Queue Manager Nar	Literal	QM_ARIBA01					

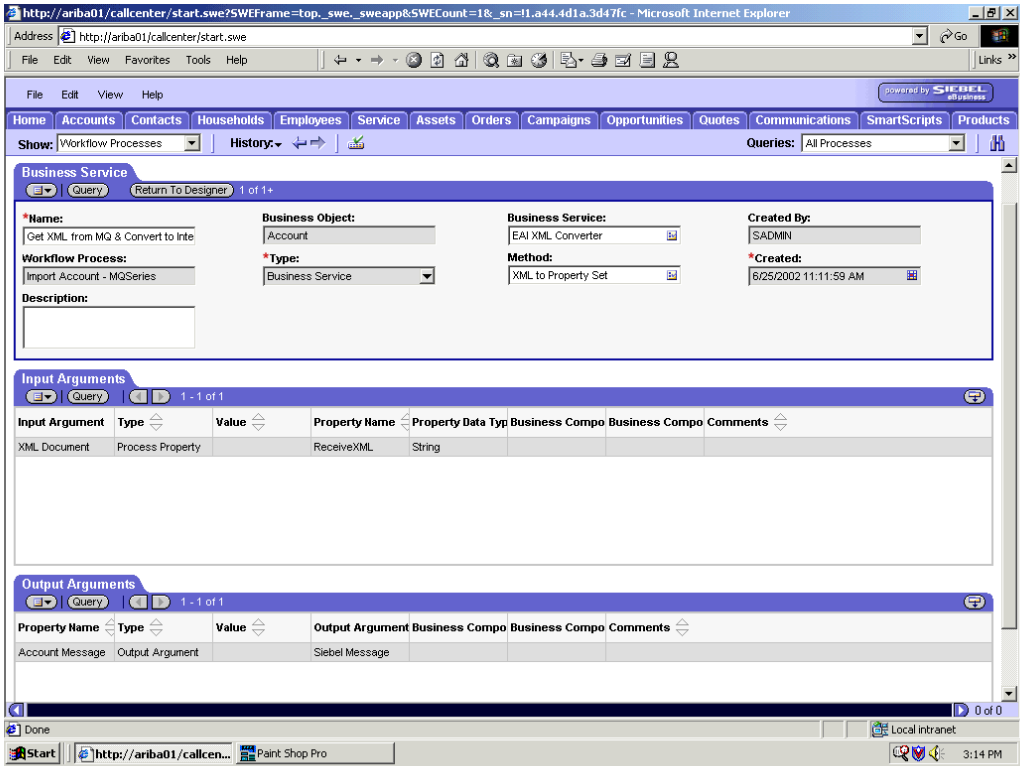
Output Arguments

Property Name	Type	Value	Output Argument	Business Compo	Business Compo	Comments
ReceiveXML	Output Argument		Message Text			

2. Define an **EAI MQSeries Server Transport Business Service** step and call it `Receive.`

   The Business Service is defined to receive the Account data from the MQSeries message queue.

   The EAI MQSeries Server Transport Business Service receives the Account data in Siebel XML format and sends it to the EAI XML Converter Business Service.



http://ariba01/callcenter/start.swe?SWEFrame=top._swe._sweapp&SWECount=1&_sn=!1.a44.4d1a.3d47fc - Microsoft Internet Explorer

Show: Workflow Processes | Queries: All Processes

Business Service — Return To Designer — 1 of 1+

*Name: Get XML from MQ & Convert to Inte	Business Object: Account	Business Service: EAI XML Converter	Created By: SADMIN
Workflow Process: Import Account - MQSeries	*Type: Business Service	Method: XML to Property Set	*Created: 6/25/2002 11:11:59 AM
Description:			

Input Arguments — 1 - 1 of 1

Input Argument	Type	Value	Property Name	Property Data Typ	Business Compo	Business Compo	Comments
XML Document	Process Property		ReceiveXML	String			

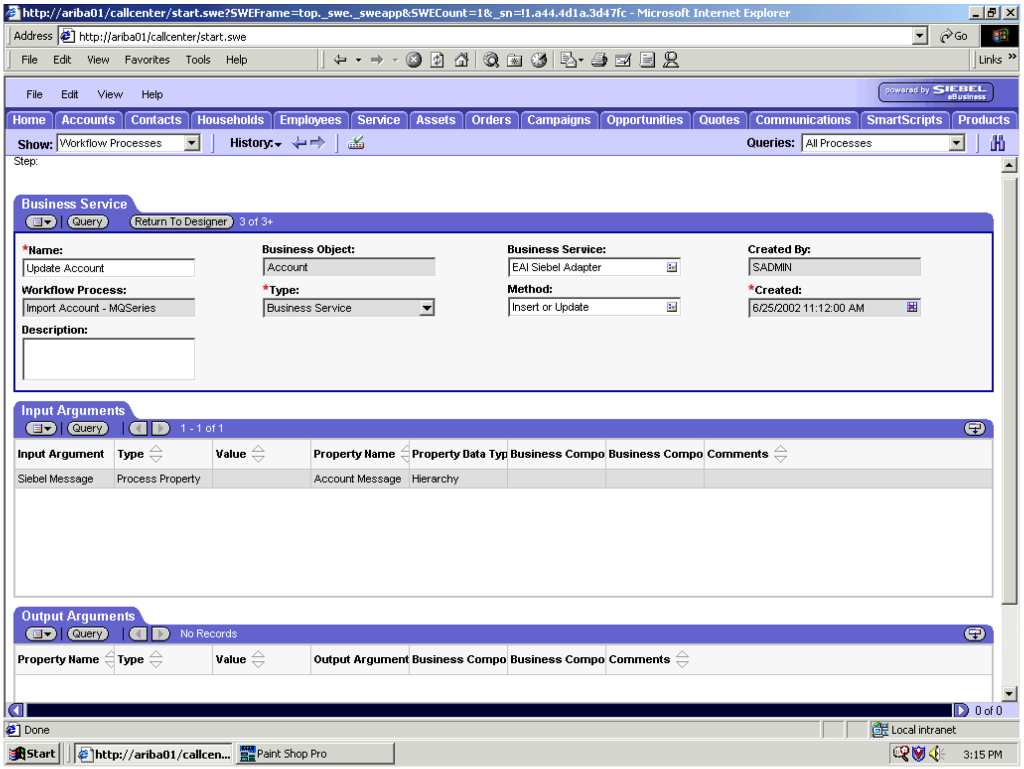
Output Arguments — 1 - 1 of 1

Property Name	Type	Value	Output Argument	Business Compo	Business Compo	Comments
Account Message	Output Argument		Siebel Message			

File Edit View Favorites Tools Help
Home Accounts Contacts Households Employees Service Assets Orders Campaigns Opportunities Quotes Communications SmartScripts Products
Done
Local intranet
Start
Paint Shop Pro
3:14 PM

3. Define an **EAI XML Converter Business Service** step and call it `Get XML from MQ & Convert to XML`.

   This Business Service is defined to receive the Account data from the EAI MQSeries Server Transport Business Service in XML format and convert it to hierarchical format.



http://ariba01/callcenter/start.swe?SWEFrame=top._swe._sweapp&SWECount=1&_sn=!1.a44.4d1a.3d47fc - Microsoft Internet Explorer

Show: Workflow Processes | Queries: All Processes

Step:

Business Service — Return To Designer — 3 of 3+

*Name: Update Account	Business Object: Account	Business Service: EAI Siebel Adapter	Created By: SADMIN
Workflow Process: Import Account - MQSeries	*Type: Business Service	Method: Insert or Update	*Created: 6/25/2002 11:12:00 AM
Description:			

Input Arguments — 1 - 1 of 1

Input Argument	Type	Value	Property Name	Property Data Typ	Business Compo	Business Compo	Comments
Siebel Message	Process Property		Account Message	Hierarchy			

Output Arguments — No Records

Property Name	Type	Value	Output Argument	Business Compo	Business Compo	Comments

File Edit View Favorites Tools Help
Home Accounts Contacts Households Employees Service Assets Orders Campaigns Opportunities Quotes Communications SmartScripts Products
Done
Local intranet
Start
Paint Shop Pro
3:15 PM

4. Define an **EAI Siebel Adapter Business Service** step and call it `Update Account.`

   This Business Service is defined to receive from the EAI XML Converter Business Service the instance of Account data in hierarchical format.

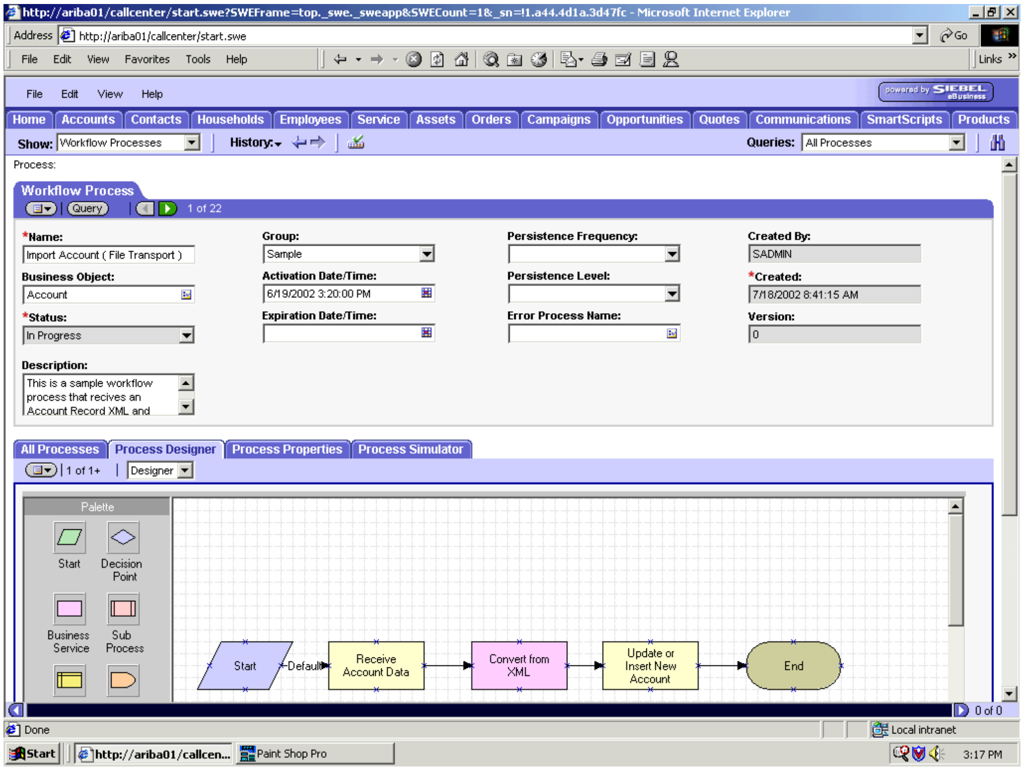
   The Business Service applies the Account information into Siebel using the Insert or Update method.

Creating a Siebel Workflow for a Service Using File Transport

The following procedure is an example of a Siebel Workflow illustrated in the Siebel Workflow Administration window. The workflow was designed for importing Siebel Account record information through the File transport

This procedure describes how to create a Siebel Workflow that generates Siebel XML when an Account record is updated in the Siebel Call Center application and then places Siebel XML on the file system.

The following is a Siebel Workflow Administration window with the Process Designer tab active.



Show: Workflow Processes

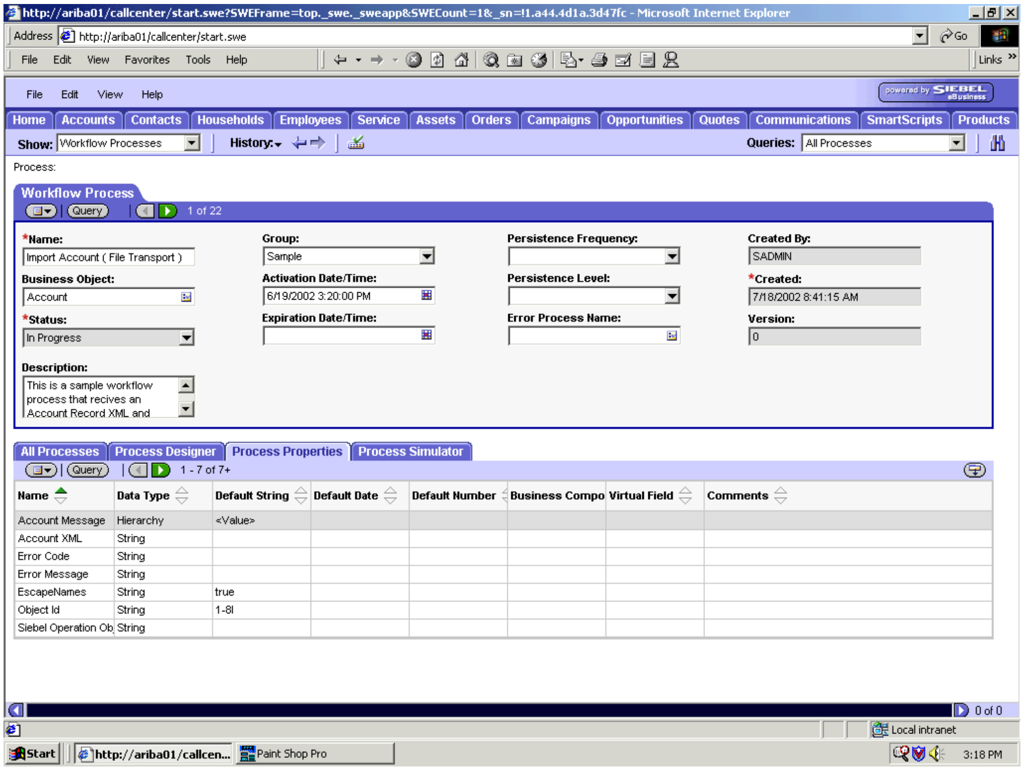
Queries: All Processes

*Name: Import Account (File Transport)	Group: Sample	Persistence Frequency:	Created By: SADMIN
Business Object: Account	Activation Date/Time: 6/19/2002 3:20:00 PM	Persistence Level:	*Created: 7/18/2002 8:41:15 AM
*Status: In Progress	Expiration Date/Time:	Error Process Name:	Version: 0

Description: This is a sample workflow process that recives an Account Record XML and

Designer

To create a Siebel Workflow:


http://ariba01/callcenter/start.swe?SWEFrame=top._swe._sweapp&SWECount=1&_sn=!1.a44.4d1a.3d47fc - Microsoft Internet Explorer
Address http://ariba01/callcenter/start.swe
File Edit View Favorites Tools Help
Links
File Edit View Help
powered by SIEBEL eBusiness
Home Accounts Contacts Households Employees Service Assets Orders Campaigns Opportunities Quotes Communications SmartScripts Products
History
Query
All Processes
Process Designer
Process Properties
Process Simulator
Local intranet
Start
http://ariba01/callcen...
Paint Shop Pro
3:18 PM
0 of 0

Show: Workflow Processes

Queries: All Processes

Process:

Workflow Process

1 of 22

*Name: Import Account (File Transport)	Group: Sample	Persistence Frequency:	Created By: SADMIN
Business Object: Account	Activation Date/Time: 6/19/2002 3:20:00 PM	Persistence Level:	*Created: 7/18/2002 8:41:15 AM
*Status: In Progress	Expiration Date/Time:	Error Process Name:	Version: 0

Description: This is a sample workflow process that recives an Account Record XML and

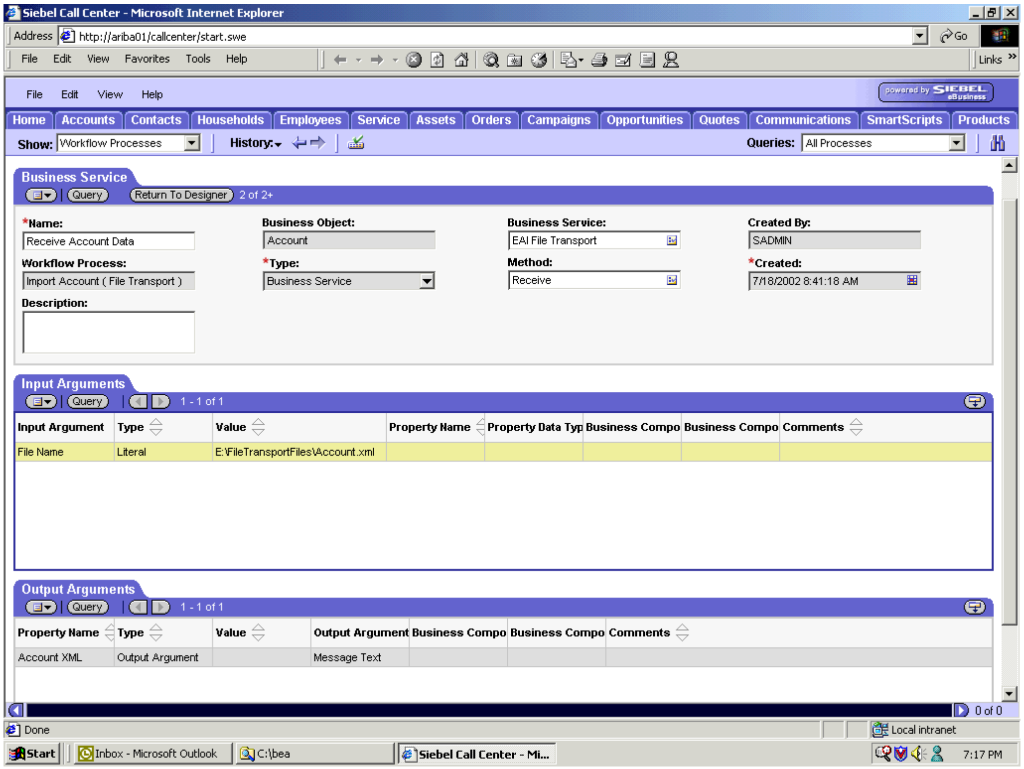
1 - 7 of 7+

Name	Data Type	Default String	Default Date	Default Number	Business Compo	Virtual Field	Comments
Account Message	Hierarchy	<Value>					
Account XML	String						
Error Code	String						
Error Message	String						
EscapeNames	String	true					
Object Id	String	1-8I					
Siebel Operation Ob	String						

1. In the **Process Properties** tab of the **Workflow Process** window, define the Account message and Account XML process properties.

   Account message contains the Siebel Account data in hierarchical format.

   Account XML specifies the Siebel Account data that the workflow converted to XML.


Siebel Call Center - Microsoft Internet Explorer
Address http://ariba01/callcenter/start.swe
File Edit View Favorites Tools Help
Links
File Edit View Help
powered by SIEBEL eBusiness
Home Accounts Contacts Households Employees Service Assets Orders Campaigns Opportunities Quotes Communications SmartScripts Products
History
Query
Return To Designer
2 of 2+
1 - 1 of 1
0 of 0
Done
Local intranet
Start
Inbox - Microsoft Outlook
C:\bea
Siebel Call Center - Mi...
7:17 PM

Show: Workflow Processes

Queries: All Processes

Business Service

*Name: Receive Account Data	Business Object: Account	Business Service: EAI File Transport	Created By: SADMIN
Workflow Process: Import Account (File Transport)	*Type: Business Service	Method: Receive	*Created: 7/18/2002 8:41:18 AM
Description:			

Input Arguments

Input Argument	Type	Value	Property Name	Property Data Typ	Business Compo	Business Compo	Comments
File Name	Literal	E:\FileTransportFiles\Account.xml					

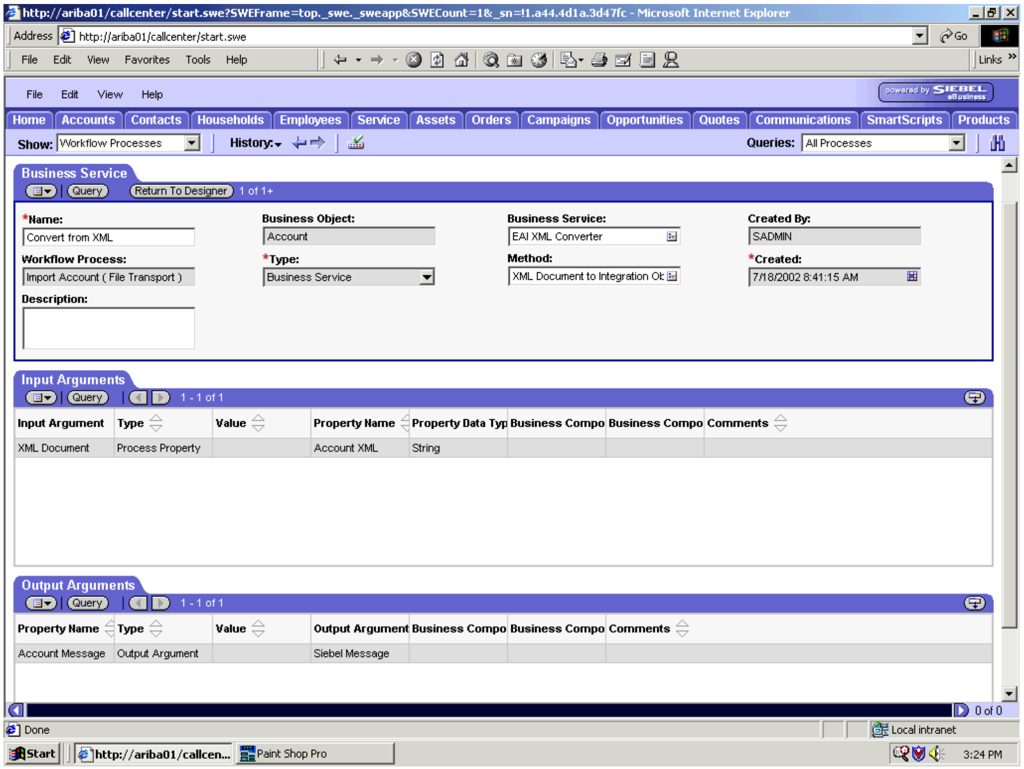
Output Arguments

Property Name	Type	Value	Output Argument	Business Compo	Business Compo	Comments
Account XML	Output Argument		Message Text			

2. Define an **EAI FileTransport Business Service** step and call it `Receive Account Data.`

   The Business Service is defined to receive the Account data from the file system.

   The EAI File Transport Business Service receives the Account data in Siebel XML format and sends it to the EAI XML Converter Business Service.



http://ariba01/callcenter/start.swe?SWEFrame=top._swe._sweapp&SWECount=1&_sn=!1.a44.4d1a.3d47fc - Microsoft Internet Explorer

Show: Workflow Processes | Queries: All Processes

Business Service — Return To Designer — 1 of 1+

*Name: Convert from XML	Business Object: Account	Business Service: EAI XML Converter	Created By: SADMIN
Workflow Process: Import Account (File Transport)	*Type: Business Service	Method: XML Document to Integration Ob	*Created: 7/18/2002 8:41:15 AM
Description:			

Input Arguments — 1 - 1 of 1

Input Argument	Type	Value	Property Name	Property Data Typ	Business Compo	Business Compo	Comments
XML Document	Process Property		Account XML	String			

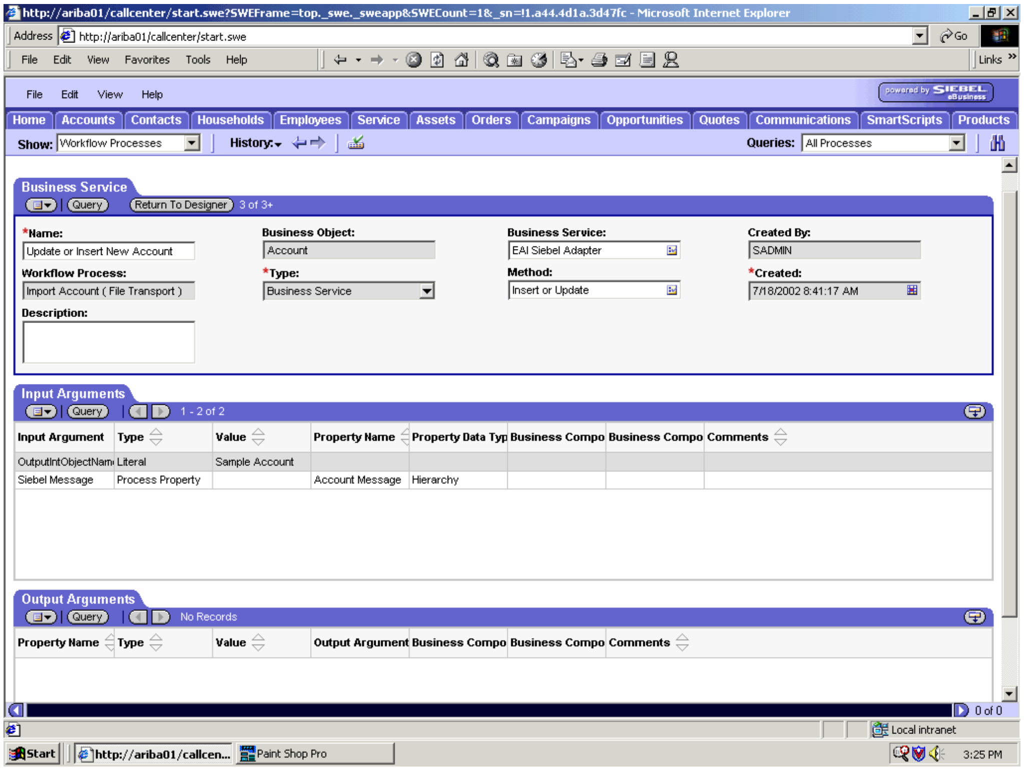
Output Arguments — 1 - 1 of 1

Property Name	Type	Value	Output Argument	Business Compo	Business Compo	Comments
Account Message	Output Argument		Siebel Message			

0 of 0

3. Define an **EAI XML Converter Business Service** step and call it `Convert from XML`.

   This Business Service is defined to receive the Account data from the EAI File Transport Business Service in XML format and convert it to hierarchical format.



http://ariba01/callcenter/start.swe?SWEFrame=top._swe._sweapp&SWECount=1&_sn=!1.a44.4d1a.3d47fc - Microsoft Internet Explorer

Show: Workflow Processes | Queries: All Processes

Business Service — Return To Designer — 3 of 3+

*Name: Update or Insert New Account	Business Object: Account	Business Service: EAI Siebel Adapter	Created By: SADMIN
Workflow Process: Import Account (File Transport)	*Type: Business Service	Method: Insert or Update	*Created: 7/18/2002 8:41:17 AM
Description:			

Input Arguments — 1 - 2 of 2

Input Argument	Type	Value	Property Name	Property Data Typ	Business Compo	Business Compo	Comments
OutputIntObjectNam	Literal	Sample Account					
Siebel Message	Process Property		Account Message	Hierarchy			

Output Arguments — No Records

Property Name	Type	Value	Output Argument	Business Compo	Business Compo	Comments

0 of 0

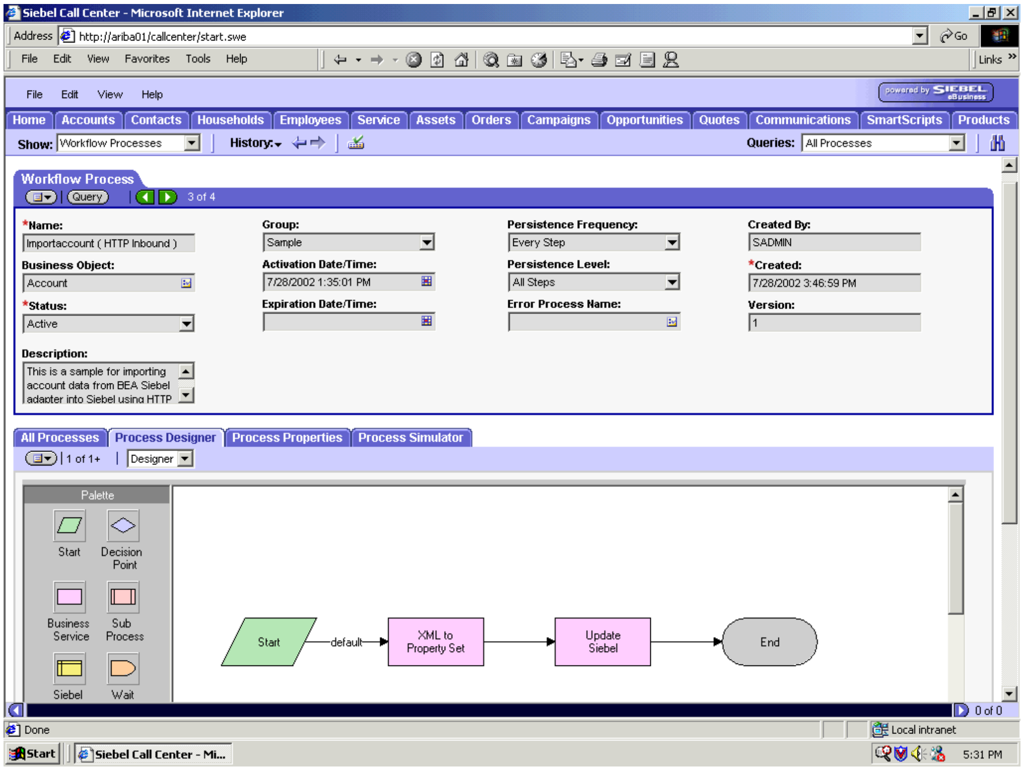
4. Define an **EAI Siebel Adapter Business Service** step and call it `Update` or `Insert New Account.`

   This Business Service is defined to receive from the EAI XML Converter Business Service the instance of Account data in hierarchical format.

   The Business Service applies the Account information into Siebel using the Insert or Update method.

Creating a Siebel Workflow for a Service Using HTTP Transport

The following procedure is an example of a Siebel workflow illustrated in the Siebel Workflow Administration window. The Workflow was designed for importing Siebel Account record information through the HTTP transport.


Siebel Call Center - Microsoft Internet Explorer
Address http://ariba01/callcenter/start.swe
Go
File Edit View Favorites Tools Help
Links
File Edit View Help
powered by SIEBEL eBusiness
Home Accounts Contacts Households Employees Service Assets Orders Campaigns Opportunities Quotes Communications SmartScripts Products
History:
Workflow Process
Query
3 of 4
All Processes
Process Designer
Process Properties
Process Simulator
1 of 1+
Designer
Palette
Start
Decision Point
Business Service
Sub Process
Siebel
Wait
Start
default
XML to Property Set
Update Siebel
End
0 of 0
Done
Local intranet
Start
Siebel Call Center - Mi...
5:31 PM

Show: Workflow Processes

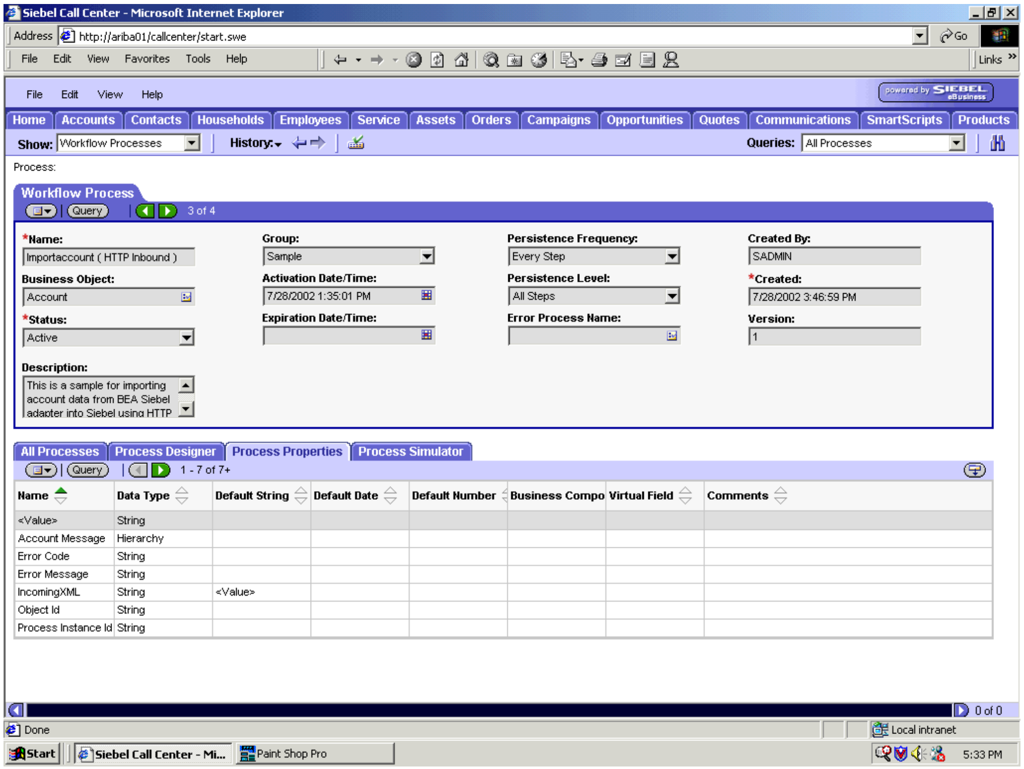
Queries: All Processes

*Name: Importaccount (HTTP Inbound)	Group: Sample	Persistence Frequency: Every Step	Created By: SADMIN
Business Object: Account	Activation Date/Time: 7/28/2002 1:35:01 PM	Persistence Level: All Steps	*Created: 7/28/2002 3:46:59 PM
*Status: Active	Expiration Date/Time:	Error Process Name:	Version: 1

Description: This is a sample for importing account data from BEA Siebel adapter into Siebel using HTTP

The following procedure describes how to create a Siebel Workflow that generates Siebel XML when an Account record is updated in the Siebel Call Center application and then places Siebel XML on the file system.

To create a Siebel Workflow:


Siebel Call Center - Microsoft Internet Explorer
Address http://ariba01/callcenter/start.swe
File Edit View Favorites Tools Help
Links
File Edit View Help
powered by SIEBEL eBusiness
Home Accounts Contacts Households Employees Service Assets Orders Campaigns Opportunities Quotes Communications SmartScripts Products
History:
Query
3 of 4
All Processes Process Designer Process Properties Process Simulator
Query
1 - 7 of 7+
0 of 0
Done
Local intranet
Start
Siebel Call Center - Mi...
Paint Shop Pro
5:33 PM

Show: Workflow Processes

Queries: All Processes

Process:

Workflow Process

*Name: Importaccount (HTTP Inbound)	Group: Sample	Persistence Frequency: Every Step	Created By: SADMIN
Business Object: Account	Activation Date/Time: 7/28/2002 1:35:01 PM	Persistence Level: All Steps	*Created: 7/28/2002 3:46:59 PM
*Status: Active	Expiration Date/Time:	Error Process Name:	Version: 1

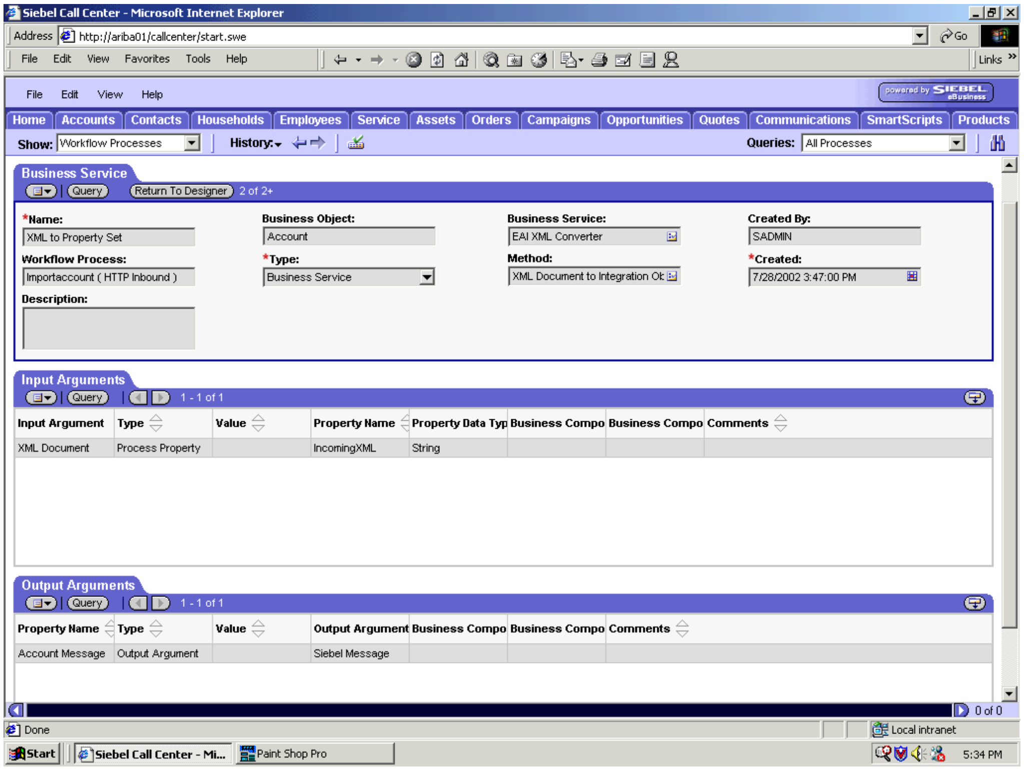
Description: This is a sample for importing account data from BEA Siebel adapter into Siebel using HTTP

Name	Data Type	Default String	Default Date	Default Number	Business Compo	Virtual Field	Comments
<Value>	String						
Account Message	Hierarchy						
Error Code	String						
Error Message	String						
IncomingXML	String	<Value>					
Object Id	String						
Process Instance Id	String						

1. In the **Process Properties** tab of the **Workflow Process** window, define the Account message and Account XML process properties.

   Account message contains the Siebel Account data in hierarchical format.

   Account XML specifies the Siebel Account data that the workflow converted to XML.


Siebel Call Center - Microsoft Internet Explorer
Address http://ariba01/callcenter/start.swe
File Edit View Favorites Tools Help
Links
File Edit View Help
powered by SIEBEL eBusiness
Home Accounts Contacts Households Employees Service Assets Orders Campaigns Opportunities Quotes Communications SmartScripts Products
History:
Query
Return To Designer
Done
Local intranet
Start
Siebel Call Center - Mi...
Paint Shop Pro
5:34 PM

Show: Workflow Processes

Queries: All Processes

Business Service

2 of 2+

*Name: XML to Property Set

Workflow Process: Importaccount (HTTP Inbound)

Description:

Business Object: Account

*Type: Business Service

Business Service: EAI XML Converter

Method: XML Document to Integration Ot

Created By: SADMIN

*Created: 7/28/2002 3:47:00 PM

Input Arguments

1 - 1 of 1

Input Argument	Type	Value	Property Name	Property Data Typ	Business Compo	Business Compo	Comments
XML Document	Process Property		IncomingXML	String			

Output Arguments

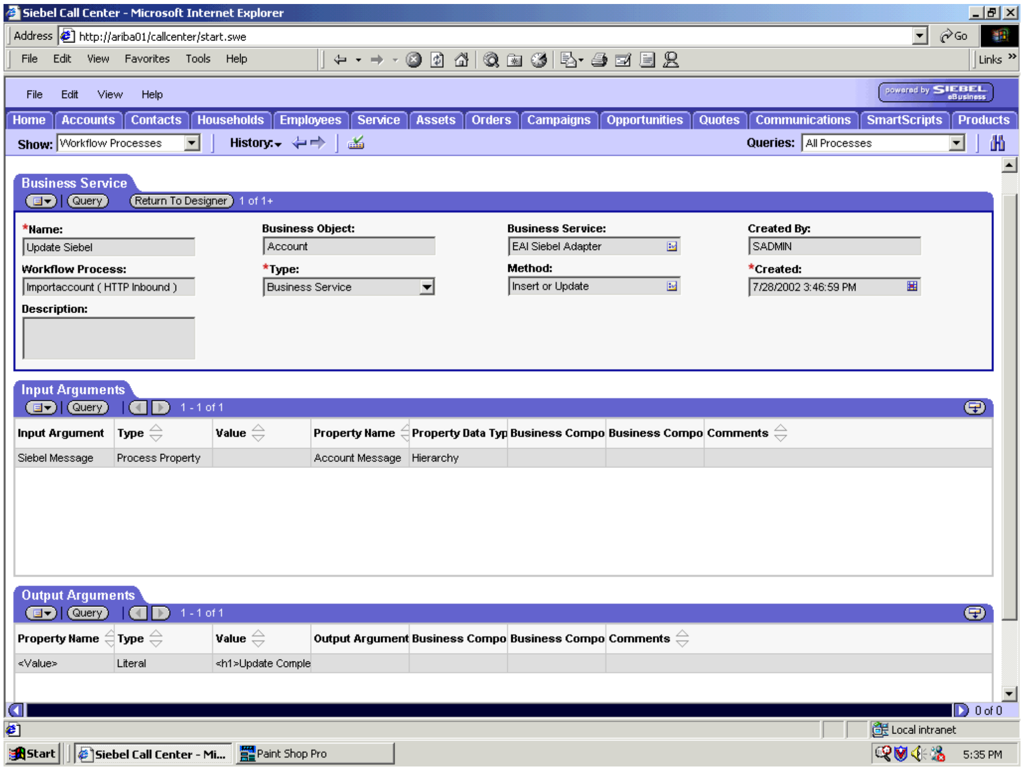
1 - 1 of 1

Property Name	Type	Value	Output Argument	Business Compo	Business Compo	Comments
Account Message	Output Argument		Siebel Message			

0 of 0

2. Define an **EAI XML Converter Business Service** step and call it `XML to Property Set.`

   The Business Service is defined to receive the Account data from the EAI HTTP Transport Business Service in XML format and convert it to hierarchical format.


Siebel Call Center - Microsoft Internet Explorer
Address http://ariba01/callcenter/start.swe
File Edit View Favorites Tools Help
Links
File Edit View Help
powered by SIEBEL eBusiness
Home Accounts Contacts Households Employees Service Assets Orders Campaigns Opportunities Quotes Communications SmartScripts Products
History:
Query
Return To Designer
Local intranet
Start
Siebel Call Center - Mi...
Paint Shop Pro
5:35 PM

Show: Workflow Processes

Queries: All Processes

Business Service

1 of 1+

*Name: Update Siebel

Workflow Process: Importaccount (HTTP Inbound)

Description:

Business Object: Account

*Type: Business Service

Business Service: EAI Siebel Adapter

Method: Insert or Update

Created By: SADMIN

*Created: 7/28/2002 3:46:59 PM

Input Arguments

1 - 1 of 1

Input Argument	Type	Value	Property Name	Property Data Typ	Business Compo	Business Compo	Comments
Siebel Message	Process Property		Account Message	Hierarchy			

Output Arguments

1 - 1 of 1

Property Name	Type	Value	Output Argument	Business Compo	Business Compo	Comments
<Value>	Literal	<h1>Update Comple				

0 of 0

3. Define an **EAI Siebel Adapter Business Service** step and call it `Update Siebel.`

   The Business Service is defined to receive from the EAI XML Converter Business Service the instance of Account data in hierarchical format.

   The Business Service applies the Account information into Siebel using the Insert or Update method.

Glossary

adapter

Provides universal connectivity by enabling an electronic interface to be accommodated (without loss of function) to another electronic interface.

agent

Supports service protocols in listeners and documents.

business service

Also known as a Web service. A Web service is a self-contained, modularized function that can be published and accessed across a network using open standards. It is the implementation of an interface by a component and is an executable entity.

channel

Represents configured connections to particular instances of back-end systems. A channel binds one or more event ports to a particular listener managed by an adapter.

listener

A component that accepts requests from client applications.

port

Associates a particular business object exposed by the adapter with a particular disposition. A disposition is a URL that defines the protocol and location of the event data. The port defines the end point of the event consumption.

Index