TIBCO ActiveMatrix® Adapter Service Engine for Siebel

Configuration and Deployment

Software Release 6.0
April 2010

Important Information

SOME TIBCO SOFTWARE EMBEDS OR BUNDLES OTHER TIBCO SOFTWARE. USE OF SUCH EMBEDDED OR BUNDLED TIBCO SOFTWARE IS SOLELY TO ENABLE THE FUNCTIONALITY (OR PROVIDE LIMITED ADD-ON FUNCTIONALITY) OF THE LICENSED TIBCO SOFTWARE. THE EMBEDDED OR BUNDLED SOFTWARE IS NOT LICENSED TO BE USED OR ACCESSED BY ANY OTHER TIBCO SOFTWARE OR FOR ANY OTHER PURPOSE.

USE OF TIBCO SOFTWARE AND THIS DOCUMENT IS SUBJECT TO THE TERMS AND CONDITIONS OF A LICENSE AGREEMENT FOUND IN EITHER A SEPARATELY EXECUTED SOFTWARE LICENSE AGREEMENT, OR, IF THERE IS NO SUCH SEPARATE AGREEMENT, THE CLICKWRAP END USER LICENSE AGREEMENT WHICH IS DISPLAYED DURING DOWNLOAD OR INSTALLATION OF THE SOFTWARE (AND WHICH IS DUPLICATED IN LICENSE.PDF) OR IF THERE IS NO SUCH SOFTWARE LICENSE AGREEMENT OR CLICKWRAP END USER LICENSE AGREEMENT, THE LICENSE(S) LOCATED IN THE “LICENSE” FILE(S) OF THE SOFTWARE. USE OF THIS DOCUMENT IS SUBJECT TO THOSE TERMS AND CONDITIONS, AND YOUR USE HEREOF SHALL CONSTITUTE ACCEPTANCE OF AND AN AGREEMENT TO BE BOUND BY THE SAME.

This document contains confidential information that is subject to U.S. and international copyright laws and treaties. No part of this document may be reproduced in any form without the written authorization of TIBCO Software Inc.

TIB, TIBCO, TIBCO Adapter, TIBCO ActiveMatrix, Predictive Business, Information Bus, The Power of Now, TIBCO ActiveMatrix BusinessWorks, TIBCO Rendezvous, TIBCO Administrator, TIBCO Designer, TIBCO Runtime Agent, TIBCO Hawk, TIBCO Enterprise Message Service, TIBCO Designer Add-in for TIBCO Business Studio, TIBCO ActiveMatrix Service Grid, TIBCO ActiveMatrix Service Bus, TIBCO ActiveMatrix BusinessWorks Service Engine, TIBCO ActiveEnterprise, and TIBCO Business Studio, are either registered trademarks or trademarks of TIBCO Software Inc. in the United States and/or other countries.

All other product and company names and marks mentioned in this document are the property of their respective owners and are mentioned for identification purposes only.

THIS SOFTWARE MAY BE AVAILABLE ON MULTIPLE OPERATING SYSTEMS. HOWEVER, NOT ALL OPERATING SYSTEM PLATFORMS FOR A SPECIFIC SOFTWARE VERSION ARE RELEASED AT THE SAME TIME. SEE THE README.TXT FILE FOR THE AVAILABILITY OF THIS SOFTWARE VERSION ON A SPECIFIC OPERATING SYSTEM PLATFORM.

THIS DOCUMENT IS PROVIDED “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT.

THIS DOCUMENT COULD INCLUDE TECHNICAL INACCURACIES OR TYPOGRAPHICAL ERRORS. CHANGES ARE PERIODICALLY ADDED TO THE INFORMATION HEREIN; THESE CHANGES WILL BE INCORPORATED IN NEW EDITIONS OF THIS DOCUMENT. TIBCO SOFTWARE INC. MAY MAKE IMPROVEMENTS AND/OR CHANGES IN THE PRODUCT(S) AND/OR THE PROGRAM(S) DESCRIBED IN THIS DOCUMENT AT ANY TIME.

THE CONTENTS OF THIS DOCUMENT MAY BE MODIFIED AND/OR QUALIFIED, DIRECTLY OR INDIRECTLY, BY OTHER DOCUMENTATION WHICH ACCOMPANIES THIS SOFTWARE, INCLUDING BUT NOT LIMITED TO ANY RELEASE NOTES AND "READ ME" FILES.

Copyright © 2008-2010 TIBCO Software Inc. ALL RIGHTS RESERVED.

TIBCO Software Inc. Confidential Information

Contents

Figures

Tables

Preface

This document describes how to configure and deploy TIBCO ActiveMatrix Adapter for Siebel.

Topics

Related Documentation

This section lists documentation resources you may find useful.

TIBCO ActiveMatrix Adapter for Siebel Documentation

The following documents form the TIBCO ActiveMatrix Adapter for Siebel documentation set:

- *TIBCO ActiveMatrix Adapter Service Engine for Siebel Installation* Read this manual to learn how to install TIBCO ActiveMatrix Adapter Service Engine for Siebel.
- *TIBCO ActiveMatrix Adapter Service Engine for Siebel Configuration and Deployment* Read this manual for instructions on how to create, configure, and deploy adapter projects.
- *TIBCO ActiveMatrix Adapter Service Engine for Siebel Examples* Read this manual to work through the examples provided with the adapter.
- *TIBCO ActiveMatrix Adapter Service Engine for Siebel Release Notes* Read this document for information about new features, deprecated features, and known and closed issues.

Before TIBCO ActiveMatrix Adapter for Siebel can be installed and used, you have to install TIBCO ActiveMatrix Adapter for Siebel. The following documents form the TIBCO ActiveMatrix Adapter for Siebel documentation set:

- *TIBCO ActiveMatrix Adapter for Siebel Concepts* Read this manual to familiarize yourself with the product and its uses.
- *TIBCO ActiveMatrix Adapter for Siebel Installation* Read this manual to learn how to install TIBCO ActiveMatrix Adapter for Siebel.
- *TIBCO ActiveMatrix Adapter for Siebel Configuration and Deployment* Read this manual for instructions on how to create, configure, and deploy adapter projects.
- *TIBCO ActiveMatrix Adapter for Siebel Examples* Read this manual to work through the examples provided with the adapter.
- *TIBCO ActiveMatrix Adapter for Siebel Release Notes* Read this document for information about new features, deprecated features, and known and closed issues.

Other TIBCO Product Documentation

You may find it useful to read the documentation for the following TIBCO products:

- TIBCO Designer™
- TIBCO Administrator™
- TIBCO ActiveMatrix® Administrator
- TIBCO ActiveMatrix BusinessWorks™
- TIBCO ActiveMatrix BusinessWorks™ Service Engine
- TIBCO Rendezvous®
- TIBCO Enterprise Message Service™
- TIBCO Hawk®
- TIBCO Adapter™ SDK
- TIBCO Runtime Agent™
- TIBCO Business Studio™

Third-Party Documentation

You may also find it useful to read the *Siebel Bookshelf*, available from Siebel Systems Inc.

Typographical Conventions

The following typographical conventions are used in this manual.

Table 1 General Typographical Conventions

Convention	Use
TIBCO_HOME *ENV_HOME* *AMX_HOME*	Many TIBCO products must be installed within the same home directory. This directory is referenced in documentation as *TIBCO_HOME*. The value of *TIBCO_HOME* depends on the operating system. For example, on Windows systems, the default value is `C:\tibco`. Other TIBCO products are installed into an installation environment. Incompatible products and multiple instances of the same product are installed into different installation environments. The directory into which such products are installed is referenced in documentation as *ENV_HOME*. The value of *ENV_HOME* depends on the operating system. For example, on Windows systems the default value is C:\tibco. TIBCO ActiveMatrix is installed into a directory inside *ENV_HOME*. This directory is referenced in documentation as *AMX_HOME*. The value of *AMX_HOME* depends on the operating system. For example, on Windows systems the default value is C:\tibco\amx. TIBCO ActiveMatrix Adapter for Siebel is installed in a directory inside *AMX_HOME*.
`code font`	Code font identifies commands, code examples, filenames, pathnames, and output displayed in a command window. For example: Use `MyCommand` to start the foo process.
`bold code font`	Bold code font is used in the following ways: • In procedures, to indicate what a user types. For example: Type **`admin`**. • In large code samples, to indicate the parts of the sample that are of particular interest. • In command syntax, to indicate the default parameter for a command. For example, if no parameter is specified, `MyCommand` is enabled: `MyCommand [`**`enable`**` | disable]`

Table 1 General Typographical Conventions (Cont'd)

Convention	Use
italic font	Italic font is used in the following ways: • To indicate a document title. For example: See *TIBCO ActiveMatrix BusinessWorks Concepts*. • To introduce new terms For example: A portal page may contain several portlets. *Portlets* are mini-applications that run in a portal. • To indicate a variable in a command or code syntax that you must replace. For example: `MyCommand` *PathName*
Key combinations	Key name separated by a plus sign indicate keys pressed simultaneously. For example: Ctrl+C. Key names separated by a comma and space indicate keys pressed one after the other. For example: Esc, Ctrl+Q.
	The note icon indicates information that is of special interest or importance, for example, an additional action required only in certain circumstances.
	The tip icon indicates an idea that could be useful, for example, a way to apply the information provided in the current section to achieve a specific result.
	The warning icon indicates the potential for a damaging situation, for example, data loss or corruption if certain steps are taken or not taken.

Terminology and Acronyms

The following acronyms are used in this manual:

Acronym	Meaning
API	Application Programming Interface
QOS	TIBCO Rendezvous quality of service
GUI	Graphical User Interface
RV	Refers to TIBCO Rendezvous reliable message quality of service, as opposed to certified message
RVCM	Refers to TIBCO Rendezvous certified message quality of service
RVDQ	Refers to TIBCO Rendezvous distributed queue
RPC	Remote Procedural Call
TRA	TIBCO Runtime Agent
JMS	Java Message Service
Component Interface (CI)	A Component Interface is a PeopleTools object that you create in PeopleSoft Application Designer. It exposes a PeopleSoft component for synchronous access from another application. External applications need not be concerned with the details of page structures and component definitions in order to access the underlying data and business logic through Component Interfaces.
TIB Work Page	This page is bundled with the `TIB_PS8_ADAPTER` project. It implements the logic for capturing desired fields for publication at runtime. The page must be attached to the component from which data is captured.
Inbound	Events coming into the adapter. It refers to the inflow of data into the PeopleSoft application. It is applicable to Subscription and Request-Response services.
Outbound	Events going out from the adapter. It refers to data being captured from the PeopleSoft application and being sent out. It is applicable to the Publication Service.
IB	Integration Broker

Acronym	Meaning
VPD	Vital Product Database

How to Contact TIBCO Support

For comments or problems with this manual or the software it addresses, please contact TIBCO Support as follows.

- For an overview of TIBCO Support, and information about getting started with TIBCO Support, visit this site:

  http://www.tibco.com/services/support

- If you already have a valid maintenance or support contract, visit this site:

  https://support.tibco.com

  Entry to this site requires a user name and password. If you do not have a user name, you can request one.

Chapter 1 Introduction

This chapter provides an introduction to TIBCO ActiveMatrix Adapter Service Engine for Siebel.

Topics

Overview

TIBCO ActiveMatrix Adapter Service Engine for Siebel is a gateway for TIBCO ActiveMatrix Adapter for Siebel to the Service Oriented Architecture (SOA) world. The TIBCO ActiveMatrix Adapter Service Engine for Siebel provides an ActiveMatrix container to deploy adapter projects using the TIBCO ActiveMatrix Administrator.

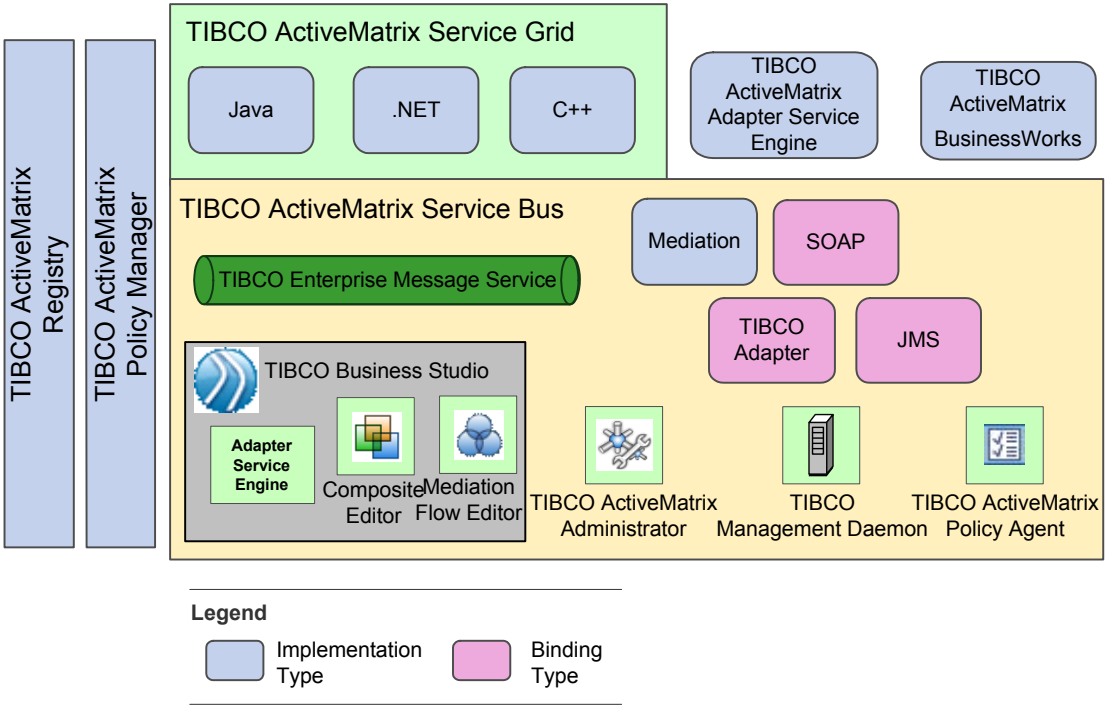
This manual describes the process of creating and configuring an adapter instance, packaging it into a service assembly and then deploying the services.

The design-time tool used is TIBCO Business Studio and the services are deployed using TIBCO ActiveMatrix Administrator.

Before proceeding you should familiarize yourself with the concepts and the terminology used by TIBCO ActiveMatrix Adapter Service Engine for Siebel and TIBCO ActiveMatrix platform.

The following figure shows the TIBCO ActiveMatrix product family.

Figure 1 TIBCO ActiveMatrix Product Family



Adapter Service Engine Project Lifecycle

This section describes the high-level steps required to configure and deploy an adapter. Each of these steps are described in details in subsequent chapters. Adapter projects are configured using TIBCO Business Studio.

Configuration

Task A Creating an Adapter Project

The adapter project at this point contains the basic structure to which you later add adapter service engine configurations.

See Creating an Adapter Project on page 8 for more information.

Task B Adding an Adapter Service Engine Configuration

The adapter service engine configuration instance contains information about schema definitions, basic adapter configuration, adapter services, and log sinks.

See Chapter 4, Configuring an Adapter Service Engine Instance, on page 39 for more information.

Task C Creating the Service Assembly

A service assembly is an unit of packaging in the ActiveMatrix world. Before you can deploy the adapter project it has to be packaged into a service assembly.

See Creating a Service Assembly and a Service Assembly Archive on page 84 for more information.

Deployment

Task D Deploying and Starting the Service Assembly

The service assembly is deployed using TIBCO ActiveMatrix Administrator.

See Chapter 6, Deploying the Service Assembly Archive, on page 85 for more information.

Chapter 2 Working with TIBCO Business Studio

The TIBCO ActiveMatrix development tools consist of TIBCO Business Studio Workbench and a set of TIBCO ActiveMatrix plug-ins. This chapter describes how to start TIBCO Business Studio, create a new adapter project, and import projects and files into the project.

For information on TIBCO Business Studio, refer to the *Workbench User Guide* in the Workbench online help. To view the online help, select **Help > Help Contents**.

Topics

Starting TIBCO Business Studio

To start TIBCO Business Studio, follow these steps:

When you start TIBCO Business Studio for the first time after installing TIBCO ActiveMatrix Adapter Service Engine for Siebel, make sure you use the `-clean` option.

1. Execute one of the following platform-specific commands to open the Workspace Launcher window.
   - — On Microsoft Windows

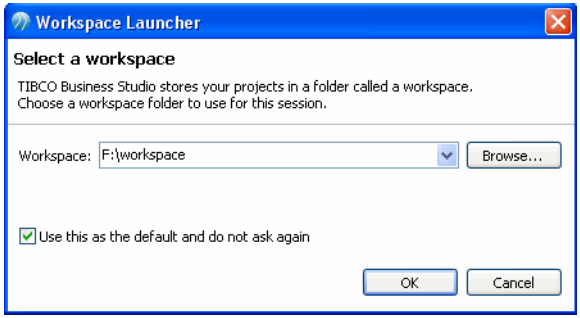
     From the Start menu, select **All Programs > TIBCO > TIBCO Business Studio <*version number*> > TIBCO Business Studio**.

     or

     From the command line, run *TIBCO_HOME*`\BusinessStudio\TIBCOBusinessStudio.exe`.
   - — On UNIX

     Run *TIBCO_HOME*`/BusinessStudio/TIBCOBusinessStudio.sh`.
2. The Workspace Launcher dialog appears, as shown below. Accept the default workspace or browse to create a new workspace. Click the **OK** button.

Figure 2 Workspace Launcher

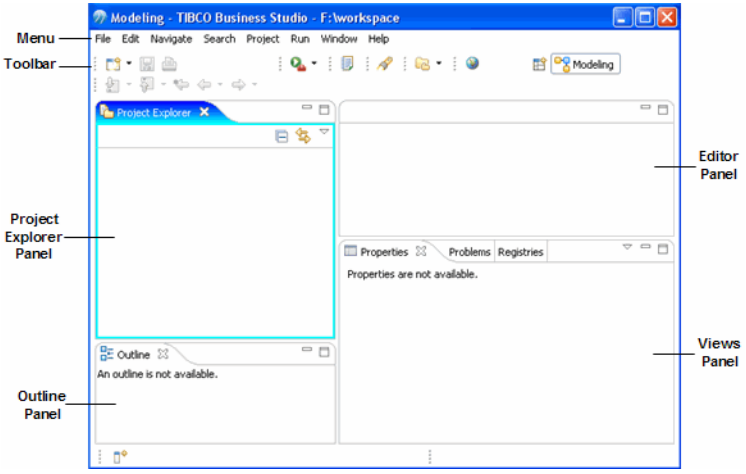


3. The Workbench window appears, as shown below. The first time a new workspace is selected, a Welcome screen displays on the window.

The Layout of the Workbench Window

The Workbench window contains the following areas: Menu, Toolbar, Project Explorer Panel, Outline Panel, Editor Panel, and Views Panel. For detailed information, refer to **Help** > **Help Content** > *Workbench User Guide*.

Figure 3 The Workbench Window



- **Menu**

  Contains menu items such us File, Edit, Navigate, Project, Run, Window, and Help.

- **Toolbar**

  Contains buttons for the most frequently used commands.

- **Project Explorer Panel**

  Displays a tree containing all the project resources such as project folders, shared resource definition files, WSDL files, composite files, service assembly files, and so on.

- **Editor Panel**

  Displays editors for the objects currently being edited. You can switch between editors by clicking tabs at the top of the Editor area. The Composite Editor contains a canvas on which you can drop elements and a palette that organizes the elements that you have added to the composite. Other editors allow you to configure shared resources and service assemblies.

- **Outline Panel**

  Provides an overview of the Composite Editor canvas. You can easily navigate from one part of a composite to another.

The Outline view also displays a content tree structure that contains the composite elements inside the composite. In this view you can delete the contents of the composite. When you select a composite element in the Outline tree, the corresponding artifact in the composite becomes selected.

- **Views Panel**

  Displays under the Editor Area and contains the following views by default:

  — Properties

  Displays property sheets for editing composites and composite elements. When you select a composite or composite element in the Composite Editor canvas, this view shows the properties of the selected object in a vertical tabbed notebook.

  — Problems

  Displays validation and other errors.

  — Registries

  Lists UDDI registries and the WSDL files returned from searching a registry.

  You open a view by selecting **Window > Show View**.

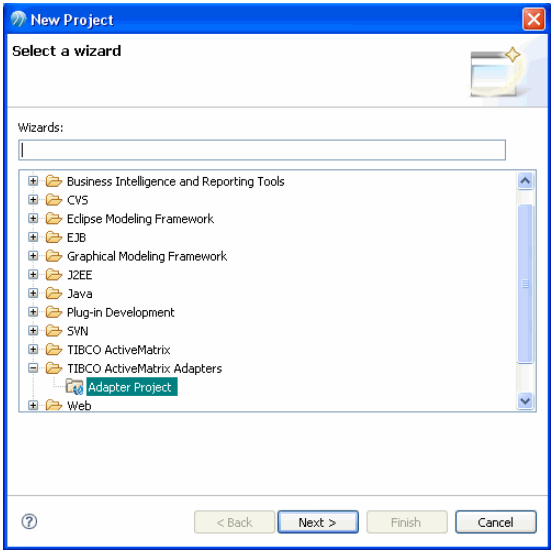
Creating an Adapter Project

An adapter project contains adapter service engine configurations, related shared resources, and business functionality which must be enabled for the adapter service engine to function. Adapter projects are created using TIBCO Business Studio.

To create an Adapter project, follow these steps:

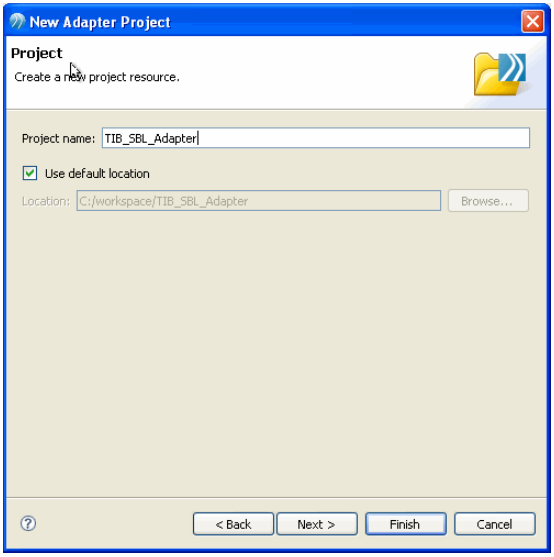
1. Start TIBCO Business Studio.
2. Select **File > New > Project** from the Menu to open the New Project window.
3. Select **TIBCO ActiveMatrix Adapters > Adapter Project** in the Select a wizard page and click the **Next** button.

Figure 4 Create a Project - Select a Wizard



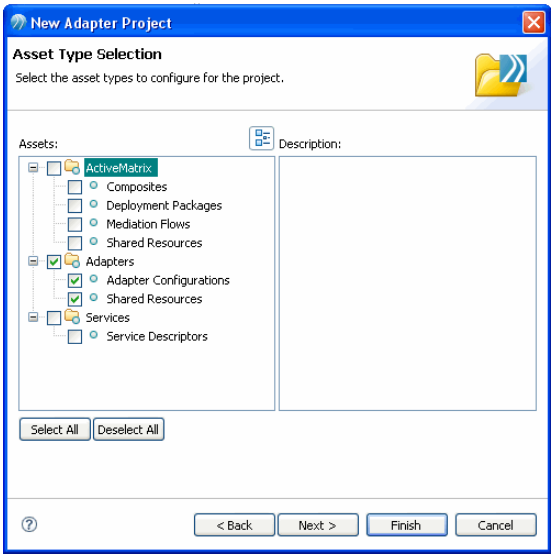
4. Type the adapter project name (For example, *TIB_SBL_Adapter*) in the Project name field and click the **Next** button.

Figure 5 Create a Project - Project



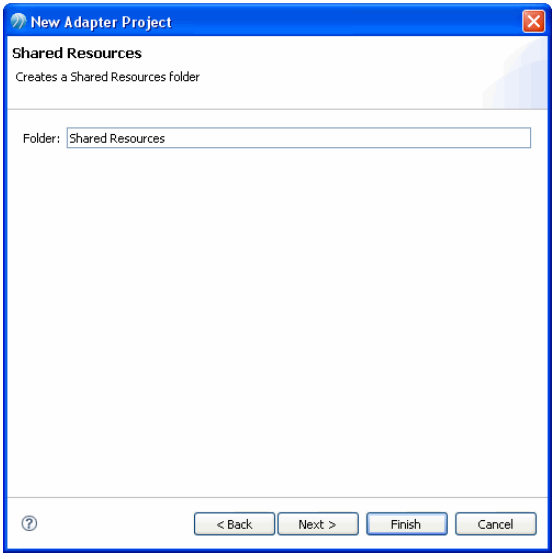
5. Select the asset types to configure for the project in the Asset Type Select page and click the **Next** button.

Figure 6 Create a Project - Asset Type Selection



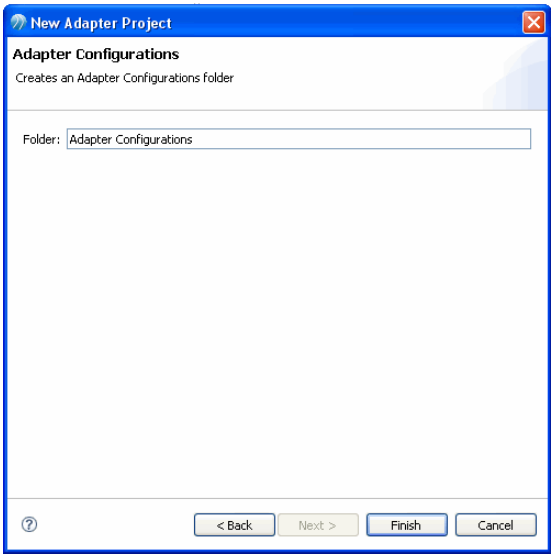
6. Type a name for the Shared Resources folder in the Folder field and click the **Next** button.

Figure 7 Create a Project - Shared Resources



7. Type a name for the Adapter Configurations folder in the Folder field and click the **Finish** button.

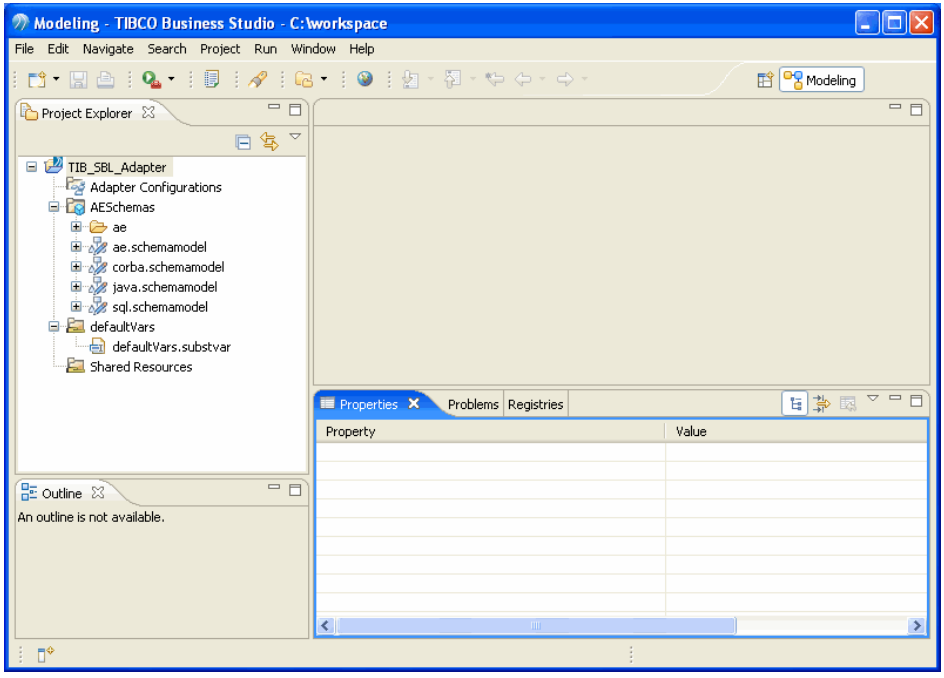
Figure 8 Create a Project - Adapter Configurations



The newly created project is shown in the Project Explorer, as shown below. It contains the following folders:

- Adapter Configurations
- AESchemas
- defaultVars
- Shared Resources

Figure 9 The Project Structure



Importing TIBCO Designer Adapter Projects

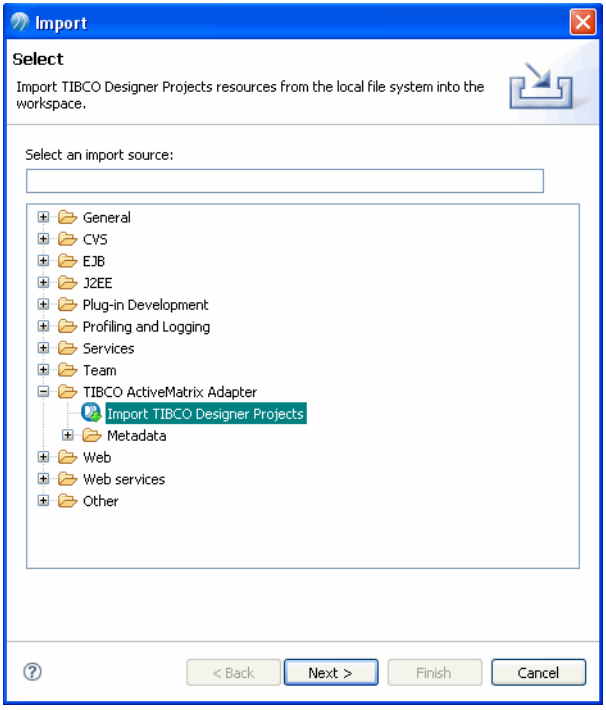
For the standalone component of the adapter, TIBCO ActiveMatrix Adapter for Siebel, the adapter projects created using TIBCO Designer can also be imported and used in the TIBCO Business Studio environment.

For detailed information about directly creating a TIBCO Designer project in TIBCO Business Studio, refer to Creating a TIBCO Designer Project in TIBCO Business Studio on page 17.

To import a TIBCO Designer adapter project, follow these steps:

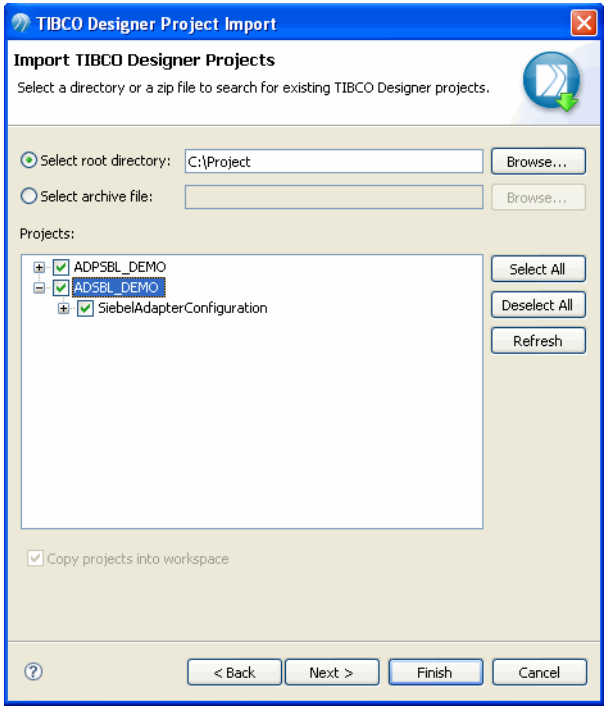
1. Start TIBCO Business Studio.
2. Select **File > Import...** from the Menu to open the Import window.
3. Expand the **TIBCO ActiveMatrix Adapters** folder and select **Import TIBCO Designer Projects** in the Select page, as shown below, and then click the **Next** button.

Figure 10 Import a Project - Select



4. Select a directory or an archive file to search for the existing TIBCO Designer projects, as shown below, and then click the **Next** button.

Figure 11 Import a Project - Import TIBCO Designer Projects



a. Choose between **Select root directory** and **Select archive file** and click the **Browse...** button to navigate to the folder where the Designer adapter project is located or select the archive containing the Designer adapter projects. Click the **OK** button.

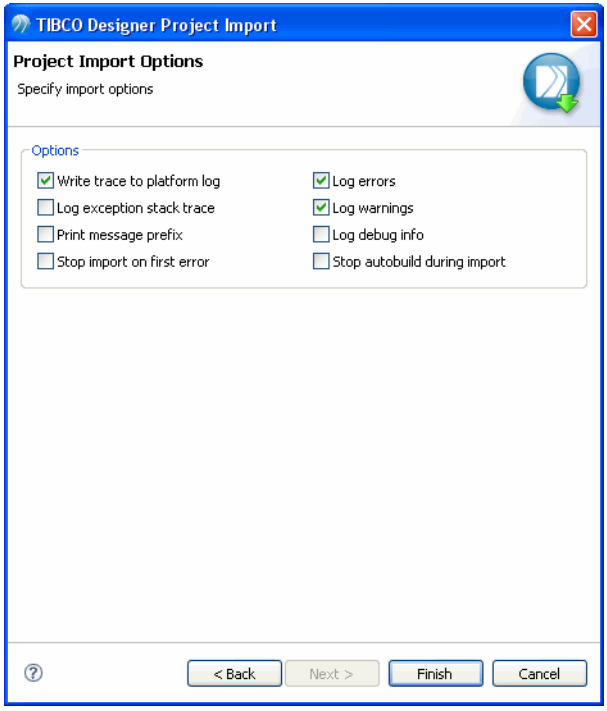
b. In the Projects pane, check the check box for the project you want to import as well as the services contained in the project.

c. The **Copy projects into workspace** check box is checked by default and the projects are copied into the current workspace.

d. Click the **Next** button.

5. Select the import options from the **Project Import Options** page, as shown below, and then click the **Finish** button.

Figure 12 Create a Project - Asset Type Selection

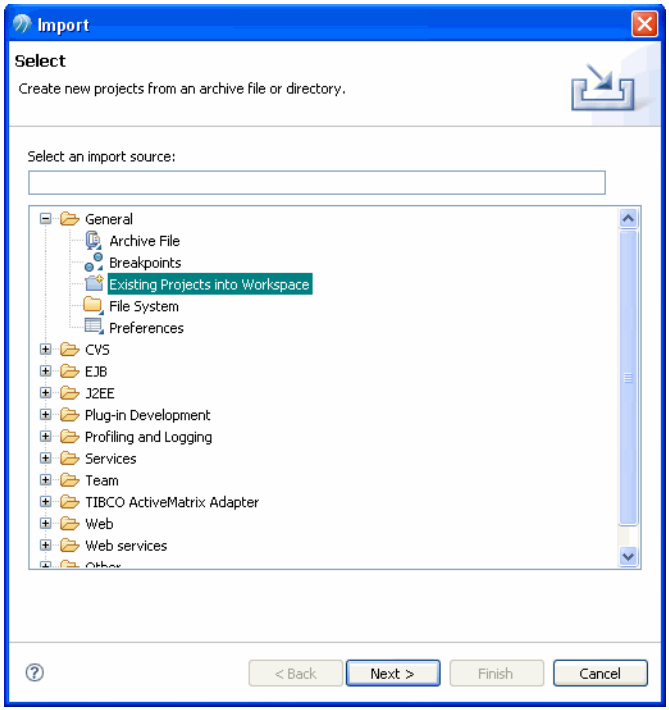


Importing Projects into Workspace

Follow these steps to import projects created in another workspace to the current workspace:

1. Start TIBCO Business Studio.
2. Select **File > Import...** from the Menu to open the Import dialog.
3. Select **General > Existing Projects into Workspace** in the Select page, as shown below, and then click the **Next** button.

Figure 13 Import a Project into Workspace - Select



4. Select a directory or an archive file to search for the existing TIBCO Designer projects in another workspace, and then click the **Next** button.
   - Choose between Select root directory and Select archive file and click the **Browse...** button to navigate to the folder where the Designer adapter

project is located or select the archive containing the Designer adapter projects and then click the **OK** button.

— In the Projects pane, check the check box for the project you want to import as well as the services contained in the project.

— Select the **Copy projects into workspace** check box if you want to copy the projects into the current workspace.

5. Click the **Finish** button.

Working with TIBCO ActiveMatrix BusinessWorks Service Engine

This section introduces procedures that have to be carried out to integrate the adapter with TIBCO ActiveMatrix BusinessWorks Service Engine.

Before installing TIBCO ActiveMatrix BusinessWorks Service Engine, you need to install TIBCO Designer Add-in for TIBCO Business Studio.

To configure a TIBCO ActiveMatrix BusinessWorks Service Engine example, do the following configurations:

1. Creating an Adapter Project and Configuring an Adapter Instance, page 17
2. Creating an ActiveMatrix SOA Project, page 17
3. Creating a TIBCO Designer Project in TIBCO Business Studio, page 17
4. Generating a WSDL File, page 18
5. Adding and Configuring Composite Elements, page 18
6. Importing the Schema folder into the TIBCO Designer Project, page 18
7. Configuring the BusinessWorks Process, page 19
8. Creating the Service Assembly, page 19

Creating an Adapter Project and Configuring an Adapter Instance

To create an adapter project and configure an adapter instance, refer to Configuring an Adapter Service Engine Instance on page 23.

Creating an ActiveMatrix SOA Project

To create an SOA project, refer to Creating an ActiveMatrix SOA Project on page 31.

Creating a TIBCO Designer Project in TIBCO Business Studio

To create a TIBCO Designer Project, follow these steps:

1. Select **File > New > Project** from the Menu to open the New Project window.
2. Select **TIBCO Designer > TIBCO Designer Project** in the Select a wizard page, and then click the **Next** button.

3. Type the designer project name (For example, *TIB_SBL_Adapter_DESIGNER*) in the Project name field and select the **Create new project in the workspace** radio button.
4. Click the **Finish** button.

Generating a WSDL File

After creating the ActiveMatrix SOA project, generate a WSDL file from the adapter instance that you have configured in Creating an Adapter Project and Configuring an Adapter Instance on page 17, and save the WSDL file into the SOA project. For detailed information about how to generating a WSDL file, refer to Generating an Adapter WSDL on page 32.

Adding and Configuring Composite Elements

To add and configure composite elements, follow these steps:

1. Click the `TIB_SBL_Adapter_SOA` project in the Project Explorer Panel, and then double click `TIB_SBL_Adapter_SOA.composite` under the Composites folder to launch the Composite Editor on the right.
2. Drag and drop the **SiebelAdapter** icon from the Components Palette into the Components canvas. Configure the Siebel Adapter Component and add a service to it.

   For detailed information about configuring a Siebel Adapter Component, refer to Creating a Siebel Adapter Component on page 34.
3. Drag and drop a service icon from the Services Palette, SOAP, AMX, or JMS, into the Services canvas. For detailed information about configuring a service, refer to Adding a Shared Resource to a Composite on page 79 for details.

Importing the Schema folder into the TIBCO Designer Project

To import the schema folder into the TIBCO Designer project, follow these steps:

1. Open the newly created TIBCO Designer project.
2. Select **File > Import...** from the Menu to open the Import dialog.
3. Select **General > File System** in the Import dialog. Click the **Next** button and browse to the location of the schema folder.
4. Click **Finish** to import the schema folder.

Configuring the BusinessWorks Process

For detailed information on how to configure a BusinessWorks process, refer to *TIBCO ActiveMatrix Adapter Service Engine for Siebel Examples*.

Creating the Service Assembly

All the services and components that were created in the previous tasks need to be packaged in a service assembly, and then they can be deployed using TIBCO ActiveMatrix Administrator.

For detailed information about how to create a service assembly, refer to Creating a Service Assembly Archive on page 87.

Chapter 3 Getting Started

This chapter describes the basic steps to configure and deploy TIBCO ActiveMatrix Adapter Service Engine for Siebel.

Topics

Overview

This chapter provides a short exercise that shows you how to configure and deploy TIBCO ActiveMatrix Adapter Service Engine for Siebel with a Publication Service, the publisher event being Siebel Business Components. Details for each step described here are provided later in the manual.

Prerequisites

Before starting the configuration, complete the prerequisites described in this section. See *TIBCO ActiveMatrix Adapter Service Engine for Siebel Installation* for details.

- Install all required software.
- Install the adapter software.
- Import the Custom Siebel Business Service, `EAI TIBCO HTTP Agent` into your Siebel System using Siebel Tools. For example, the file name and location for Siebel 8.0.0 and higher systems are as follows:

`<Adapter_Home>\<version_num>\siebel\`
`8TIBSiebelAdapterHTTPAgentBusinessService_escript.sif.`

It is assumed that you already know how to drag and drop icons in TIBCO Business Studio and save projects. If you are not familiar with these topics, refer to the *Workbench User Guide*, which can be accessed from the **Help > Help Contents** menu option in TIBCO Business Studio.

Configuring an Adapter Service Engine Instance

Each adapter project contains one or more instances of the adapter service engine configuration. This configuration is accessed whenever an adapter application is started.

A typical sequence for creating an adapter project and configuring an Adapter Service Engine instance is as follows:

1. Starting TIBCO Business Studio, page 23
2. Creating an Adapter Project, page 23
3. Creating and Configuring a Siebel Connection, page 24
4. Importing Siebel Business Objects, page 26
5. Creating a Siebel Adapter Configuration, page 27
6. Adding and Configuring a Siebel Adapter Service, page 29

Starting TIBCO Business Studio

Refer to Starting TIBCO Business Studio on page 5 for how to start TIBCO Business Studio.

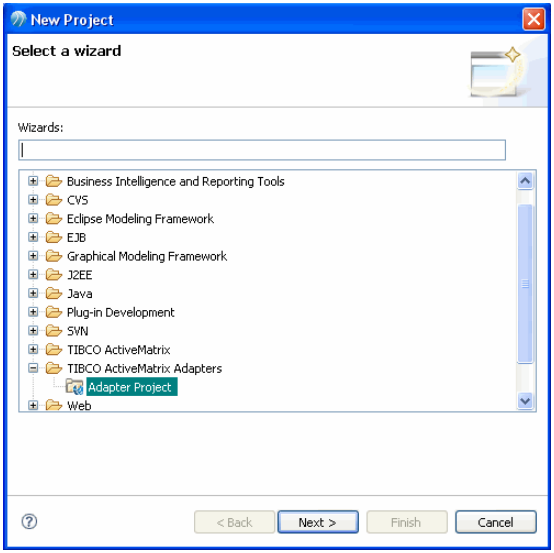
Creating an Adapter Project

An adapter project contains the configuration files that define runtime options for the Adapter Service Engine.

To create an Adapter project, follow these steps:

1. Start TIBCO Business Studio.
2. Select **File > New > Project** from the Menu to open the New Project window.
3. Select **TIBCO ActiveMatrix Adapters > Adapter Project** in the Select a wizard page, as shown below, and then click the **Next** button.

Figure 14 Create a Project - Select a Wizard



4. Enter **`TIB_SBL_Adapter`** as the adapter project name in the Project name field and then click the **Finish** button. The newly created project is shown in the Project Explorer.

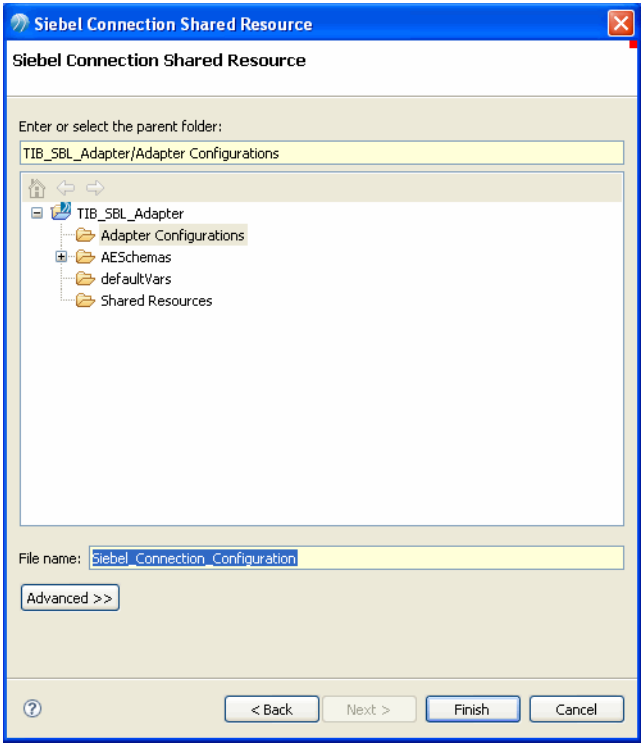
For detailed information about creating a new adapter project, refer to Creating an Adapter Project on page 8.

Creating and Configuring a Siebel Connection

After creating an adapter project, follow these steps to create and configure a Siebel Connection:

1. Create a Siebel Connection shared resource.
   a. Click the `TIB_SBL_Adapter` project in the Project Explorer Panel, and then select **File > New > Adapter Resources...** from the Menu to open the Create New Adapters Resource dialog.
   b. Expand the **Shared Resources** folder and select **Siebel Connection**. Click the **Next** button to open the Siebel Connection Shared Resource dialog, as shown below.

Figure 15 TIB_SBL_Adapter Create a Siebel Connection Shared Resource



c. Click the **Finish** button to close the dialog to open the connection configuration file `Siebel_Connection_Configuration.sharedadsblconn` under the `TIB_SBL_Adapter` project folder in the Project Explorer Panel.

2. Double-click `Siebel_Connection_Configuration.sharedadsblconn` to open the configuration tabs on the right Editor Panel. Specify the value for each parameter listed below:
   - Host
   - Enterprise Server
   - Siebel Server
   - Object Manager
   - Language
   - Transport
   - Encryption
   - Compression
   - User Name
   - Password

For detailed information about creating and configuring a Siebel Connection, refer to Creating a Siebel Connection on page 40 and Configuring a Siebel Connection on page 41.

Importing Siebel Business Objects

TIBCO ActiveMatrix Adapter Service Engine for Siebel introduces the concept of a *Business Object (BO)*. In the simplest sense, a BO is the representation of the data model of the business entities like invoice, orders, customers or vendors.

To import Siebel Business Objects (BOs), follow these steps:

1. Click the `TIB_SBL_Adapter` project in the Project Explorer Panel, and then select **File > Import** from the Menu.
2. Expand the **TIBCO ActiveMatrix Adapters** folder and select **Metadata > Siebel Business Objects**. Click the **Next** button to open the Siebel Business Objects Importer Wizard dialog.
3. Enter **`ADSBL_BO`** in the Name field and browse to select the download location and adapter connection in the dialog, and then click the **Next** button.
4. Select the Siebel BOs you want to retrieve from the Siebel Application Server.
5. Click the **Finish** button to close the dialog. After successfully importing Siebel BOs, a business object file `ADSBL_BO.bo` will be automatically generated in the `TIB_SBL_Adapter` project.

6. Double-click `ADSBL_BO.bo` in the Project Explorer Panel to view imported SiebelBOs. The Siebel Adapter Business Object Editor panel appears on the right.
7. Select the Siebel Field, for instance, Name, from the Siebel Component view by clicking on it. The Field Priorities panel appears on the right. Check the Use box and select `Account.Name` in the drop-down list of Match Field. Repeat the process to select all the fields that are to be included in the Business Event Schema.
8. Click the **Save** button  to save your configuration.

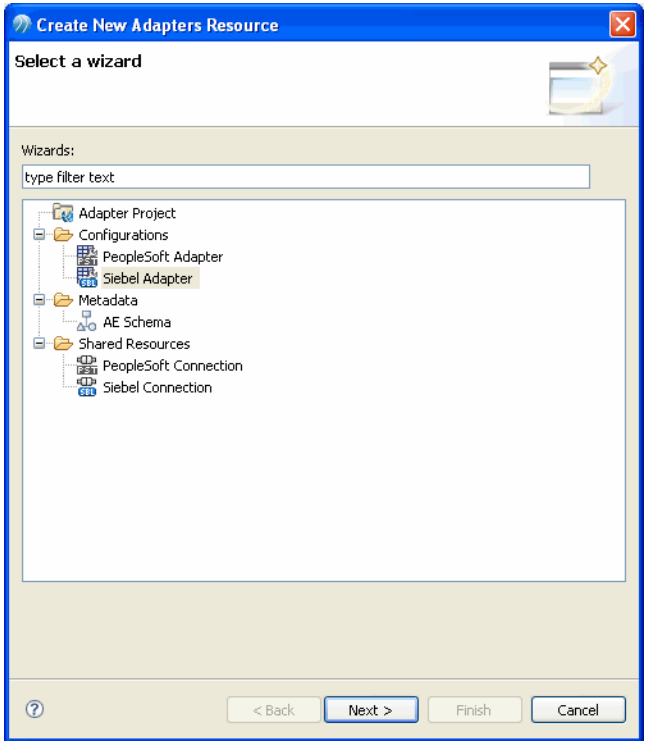
For detailed information about importing Siebel Business Objects, refer to Importing Siebel Objects on page 43.

Creating a Siebel Adapter Configuration

After creating a Siebel Connection and importing Siebel Business Objects, follow these steps to create a Siebel Adapter Configuration:

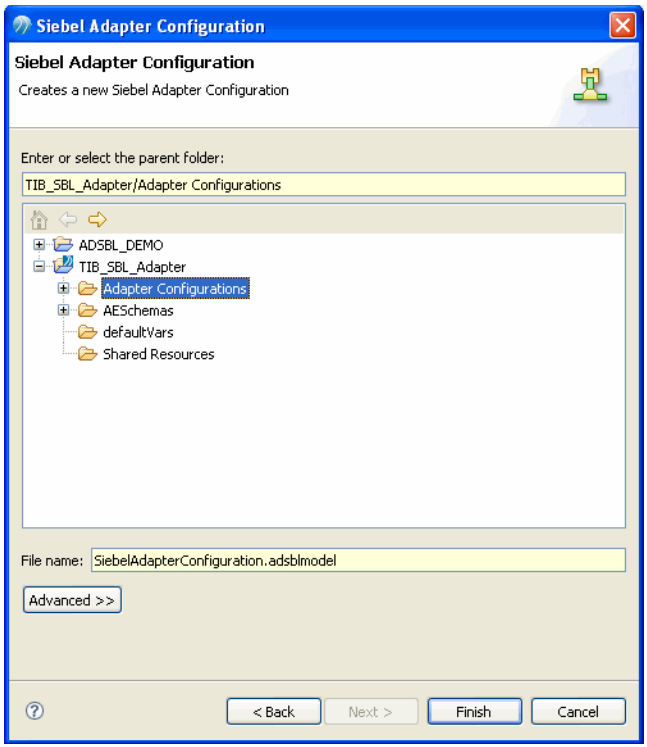
1. Click the `TIB_SBL_Adapter` project in the Project Explorer Panel, and then select **File > New > Adapter Resources...** from the Menu. The Create New Adapters Resource dialog appears, as shown below.

Figure 16 Create New Adapters Resource Dialog



2. Expand the **Configurations** folder and select **Siebel Adapter**. Click the **Next** button to open the Siebel Adapter Configuration dialog, as shown below.

Figure 17 Siebel Adapter Configuration



3. Select `Adapter_Configurations` as the parent folder and click the **Finish** button to close the dialog. The configuration file `SiebelAdapterConfiguration.adsblmodel` appears under the `Adapter_Configurations` project in the Project Explorer Panel and the configuration tabs are shown on the right Editor Panel.

For detailed information about creating a Siebel Adapter Configuration, refer to Siebel Adapter Configuration on page 50.

Adding and Configuring a Siebel Adapter Service

After creating a Siebel Adapter Configuration, adapter services can be added and configured in the Adapter Service Engine.

The following steps show how to add and configure a Publication Service. Details on the adapter services are shown in:

- Publisher Service, page 55
- Subscriber Service, page 56

- RPC Server Service, page 57
- RPC Client Service, page 59

1. Double-click `SiebelAdapterConfiguration.adsblmodel` in the Project Explorer Panel to open the configuration tabs on the right Editor Panel.
2. Click the **Configuration** tab and make sure the Connection Configuration is correct.
3. Click the **Adapter Services** tab.
4. Click the **Add Publisher** button in the All Siebel Adapter Services pane to open the parameter configuration panes in the tab.
5. Choose a Publisher Event From the drop-down list under the Publisher Option pane.
6. Select **Browse** from the Class Reference from the BO drop-down list under the Schema pane to open the Select Child dialog. Select a `.bo` file from the Matching Resources list and select an adapter project from the In Container list. Select from the siebel components in the Available Nodes list.
7. Click the **OK** button to close the dialog. A schema class file (.schemamodel) will be generated in `/AESchemas/as/Siebel/` under the `TIB_SBL_Adapter` project.
8. Click the **Save** button  to save your configuration.

For detailed information about configuring the Adapter Services, refer to Adapter Services on page 55.

Working with an ActiveMatrix SOA Project

Before the Adapter Service Engine instance can be deployed, it must be packaged in a service assembly.

To package an Adapter Service Engine instance in a service assembly, complete the following tasks:

1. Creating an ActiveMatrix SOA Project, page 31
2. Generating an Adapter WSDL, page 32
3. Creating a JMS or HTTP Server Shared Resource, page 33
4. Configuring a Composite, page 33
5. Creating a Service Assembly and a Service Assembly Archive, page 36

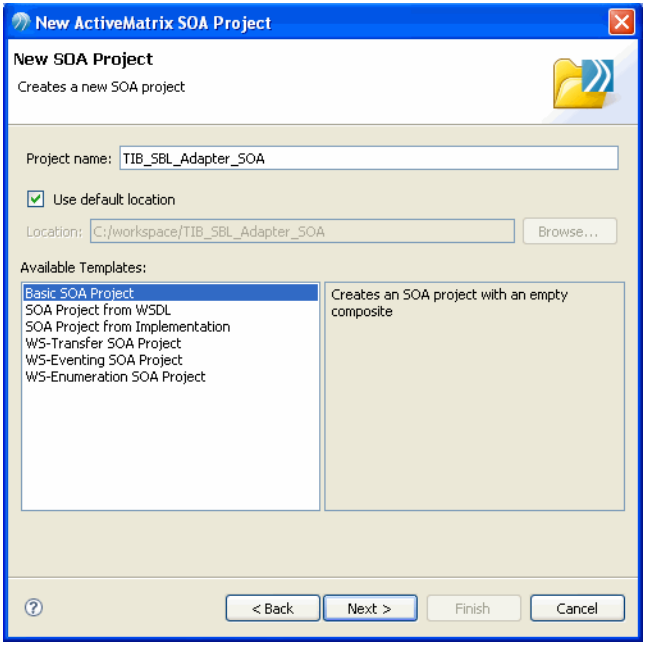
For detailed information, refer to Chapter 5, Working with ActiveMatrix SOA Projects, on page 65.

Creating an ActiveMatrix SOA Project

To create an ActiveMatrix SOA project, follow these steps:

1. Start TIBCO Business Studio.
2. Select **File > New > Project** from the Menu to open the New Project window.
3. Expand the **TIBCO ActiveMatrix** folder and select **ActiveMatrix SOA Project** in the Select a wizard page, and then click the **Next** button.
4. Complete the following operations in the New SOA Project page, as shown in the figure below.
   - Enter **`TIB_SBL_Adapter_SOA`** as the SOA project name in the Project name field.
   - Check the **Use default location** check box if you want to save the SOA project in your default workspace.
   - Select **Basic SOA Project** from the Available Templates list.

Figure 18 Create an ActiveMatrix SOA Project - New SOA Project



5. Click the **Next** button and accept all default names for the selected assets.
6. Click the **Finish** button. The `TIB_SBL_Adapter_SOA` project is shown in the Project Explorer panel.

For detailed information about creating an ActiveMatrix SOA project, refer to Creating an ActiveMatrix SOA Project on page 66.

Generating an Adapter WSDL

TIBCO ActiveMatrix services are described using WSDL files.

To generate a WSDL file, follow these steps:

1. Right-click `SiebelAdapterConfiguration.adsblmodel` under the `TIB_SBL_Adapter` project in the Project Explorer Panel, and then select **Generate Adapter WSDL** from the pop-up menu to open the Target Project dialog.
2. Select `TIB_SBL_Adapter_SOA` as the matching resource, and then click the **OK** button.

For detailed information about generating an adapter WSDL, refer to Generating an Adapter WSDL on page 70.

Creating a JMS or HTTP Server Shared Resource

After creating an ActiveMatrix SOA project, a composite file for this project is automatically generated. You will need to add a service into it. In order for the composite service to be accessible from an external client, add a TIBCO Shared Resource.

To add a TIBCO Shared Resource, follow these steps:

1. Select the `TIB_SBL_Adapter_SOA` project in the Project Explorer Panel, and then select **File > New > TIBCO Shared Resources...** from the Menu to open the TIBCO Resource Wizard dialog.
2. Select a shared resource according to the composite service type, and click the **Next** button.
   - For JMS service, select **JMS**.
   - For SOAP service, select **HTTP Server**.

If HTTP Server is selected as the shared resource, each SOA service must be assigned to a separate HTTP Port.

Select **JMS** as the example in the following steps.

3. Select `TIB_SBL_Adapter_SOA` as the parent folder for the newly created TIBCO Shared Resource file and accept the default file name, `JMS Shared Resource`.
4. Click the **Finish** button to close the dialog and open the shared resource file `JMS Shared Resource.sharedjmscon` in the Project Explorer panel.

For detailed information about creating a JMS or HTTP Server Shared Resource, refer to Configuring a TIBCO Shared Resource on page 73.

Configuring a Composite

To configure a composite, complete the following tasks:

- Task A, Launching the Composite Editor, page 34
- Task B, Creating a Siebel Adapter Component, page 34
- Task C, Adding the Shared Resource to the Composite, page 35
- Task D, Creating a Composite Service, page 35

For detailed information about configuring a composite, refer to Configuring Composites on page 76.

Task A Launching the Composite Editor

To launch the Composite Editor, follow these steps:

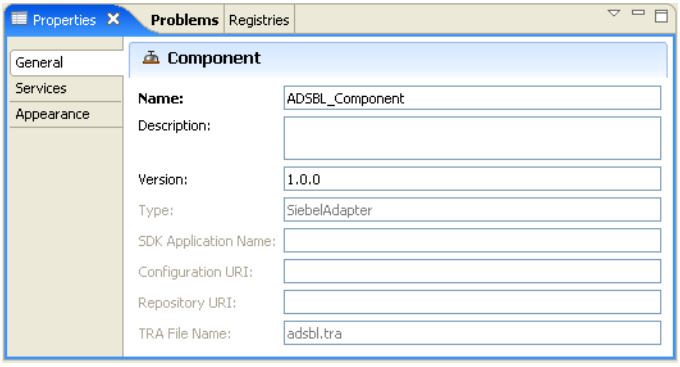
1. Expand **`TIB_SBL_Adapter_SOA`** > **Composites** in the Project Explorer panel.
2. Double-click `TIB_SBL_Adapter_SOA.composite` under the Composites folder to launch the Composite Editor on the right.

Task B Creating a Siebel Adapter Component

To create a Siebel Adapter Component for the `TIB_SBL_Adapter_SOA` project composite, follow these steps in the Composite Editor:

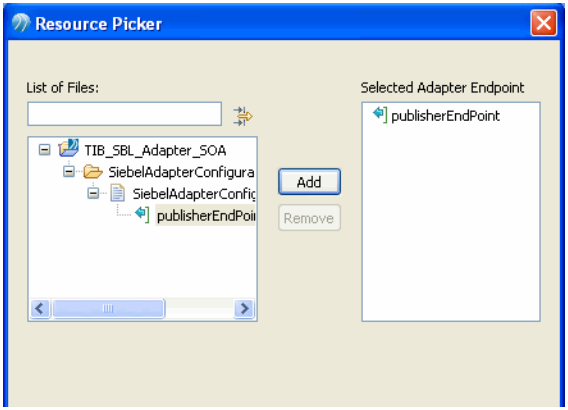
1. Drag the **SiebelAdapter** component from the Palette to the Components column and enter **`ADSBL_Component`** as the component name.
2. Click the `ADSBL_Component` component to display the configuration parameters under the Properties Views panel.

Figure 19 ADSBL_Component Properties Tab



3. Click the **Services** tab, and then Click the **Add** button to open the Resource Picker dialog. Expand the tree in the left pane and select an adapter endpoint (for example, `COUNTRYEndpoint`), and then click the **Add** button to add it into the Selected Adapter Endpoint pane on the right.

Figure 20 Resource Picker Dialog



4. Click the **OK** button. The added service appears under the Services tab.
5. Click the **Save** button to save your configuration.

Task C Adding the Shared Resource to the Composite

In section Creating a JMS or HTTP Server Shared Resource on page 33, you have created the JMS Shared Resource for your composite service. Now you need to add the JMS shared resource to the composite.

To add the JMS Shared Resource to the composite, follow these steps:

1. Click the canvas in the Composite Editor. The Composite Properties Views panel appears under the Composite Editor.
2. Click the **Shared Resource Profiles** tab to add the shared resource.
3. Click the **Save** button to save your configuration.

Task D Creating a Composite Service

To create a composite service in the Composite Editor, follow these steps:

1. Drag a service from the Palette to the Services column and enter **`ADSBL_JMS`** as the service name.
2. Click the `ADSBL_JMS` service, the configuration parameters are displayed under the Properties Views panel.
   - Click the **Target** tab and click the radio button next to the target component service. A wire is drawn between the `ADSBL_JMS` service and the `ADSBL_Component` component.
   - Click the **Binding** tab to add the shared resource profile to the Endpoint pane.

3. Click the **Save** button  to save your configuration.

Creating a Service Assembly and a Service Assembly Archive

The adapter services that have been created in the design phase have to be packaged into a service assembly before they can be deployed.

To create a service assembly and a service assembly archive, follow these steps:

1. Select the `TIB_SBL_Adapter_SOA` project in the Project Explorer panel and expand the **Composites** folder.
2. Right-click `TIB_SBL_Adapter_SOA.composite` under the `Composites` folder and select **Service Assembly** from the pop-up menu.
3. Save the composite if prompted `TIB_SBL_Adapter_SOA.saf` is created in the `Deployment` Packages folder.
4. Expand the **Deployment Packages** folder under the `TIB_SBL_Adapter_SOA` project in the Project Explorer. Right-click `TIB_SBL_Adapter_SOA.saf` and select **Build Archive** from the pop-up menu.
5. Save the service assembly file if prompted.

After completing the process, a ZIP file, `TIB_SBL_Adapter_SOA.zip`, is created in the `Deployment Packages` folder. This service assembly archive can now be deployed and started using TIBCO ActiveMatrix Administrator.

For detailed information about creating a service assembly and a service assembly archive, refer to Creating a Service Assembly Archive on page 87.

Deploying a Service Assembly Archive

When you are ready to deploy your service assembly archive, you can upload the archive to TIBCO ActiveMatrix Administrator to deploy the associated application.

The basic steps are listed below:

1. Start and log in to TIBCO ActiveMatrix Administrator.
2. Upload your service assembly archive. Refer to Uploading a Service Assembly Archive on page 91
3. Configure the service unit and install the shared resource. Refer to Configuring the Service Assembly on page 91
4. Deploy your service assembly. Refer to Deploying the Service Assembly on page 93.
5. Start your service assembly. Refer to Starting the Service Assembly, page 93.

For detailed information, refer to Deploying a Service Assembly Archive on page 91.

Chapter 4 Configuring an Adapter Service Engine Instance

This chapter explains how to configure an Adapter Service Engine instance. All configuration tasks are performed in TIBCO Business Studio.

Topics

Siebel Connection Shared Resource

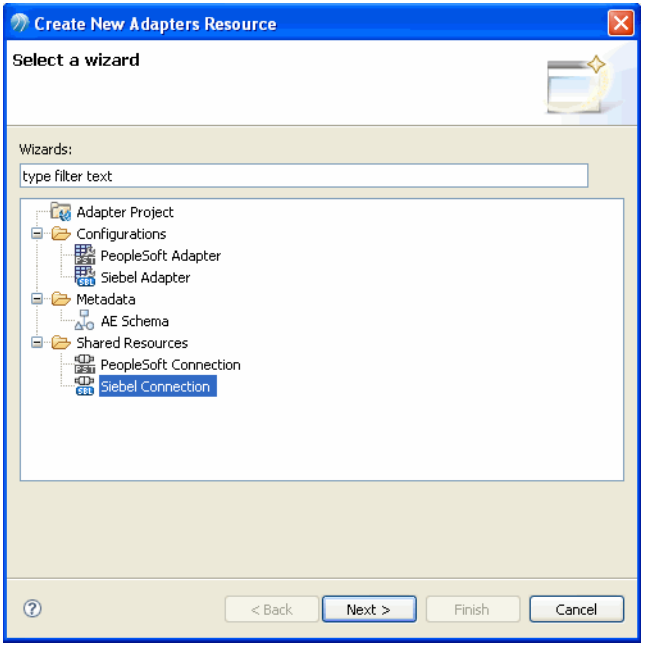
For a newly created adapter project, you need to create a Siebel Connection and configure it to access the PeopleSoft application server.

Creating a Siebel Connection

To create a Siebel Connection, follow these steps:

1. Select a project in the Project Explorer Panel, and then select **File > New > Adapter Resources...** from the Menu. The Create New Adapters Resource dialog appears, as shown below.

Figure 21 Create a Siebel Connection



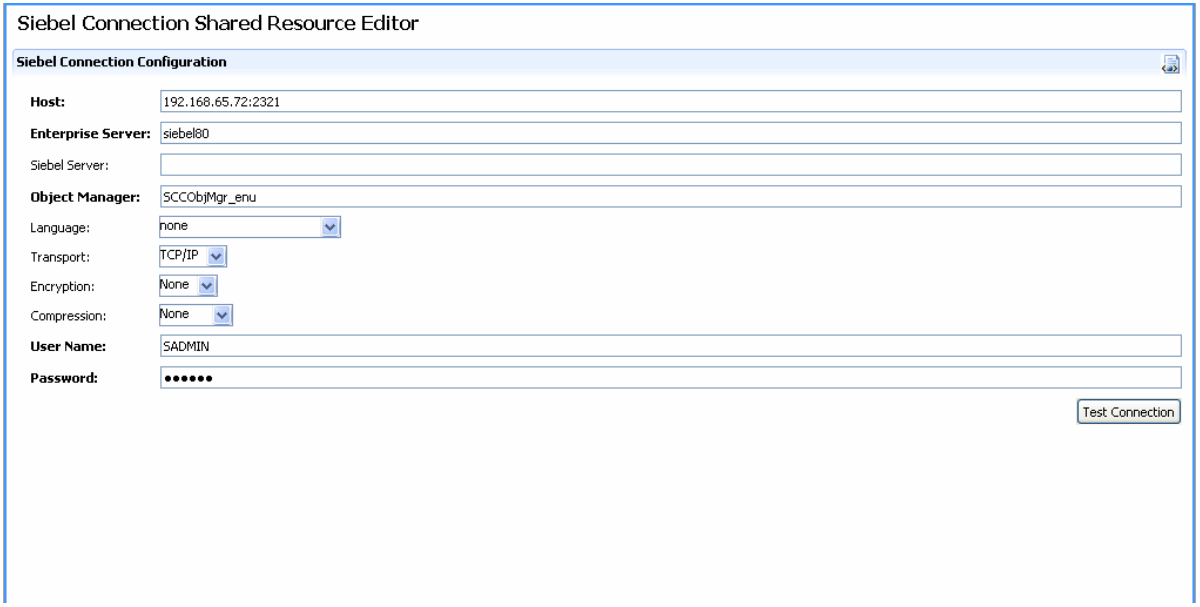
2. Expand the **Shared Resources** folder and select **Siebel Connection**. Click the **Next** button to open the Siebel Connection Shared Resource dialog.
3. Enter or select the parent folder for the newly created connection configuration file and enter the connection configuration file name in the **File name** field. The default connection configuration file name is `Siebel_Connection_Configuration`.
4. Click the **Finish** button to close the dialog.

Configuring a Siebel Connection

To configure a Siebel Connection, follow these steps:

1. Double-click the connection configuration file under the adapter project folder in the Project Explorer Panel. The Siebel Connection Shared Resource Editor panel appears on the right.

Figure 22 Siebel Connection Shared Resource Editor



2. Specify the value for each parameter listed in Table 2.

Table 2 Siebel Connection Configuration Parameters

Field	Description
Host	The name of the machine on which the gateway server is installed along with the port number. If the gateway server is running on a non-default (2320) port, then the parameter, Host should have a value <hostname>:<Port>. For example, myGatewayHost:7666.
Enterprise Server	The Siebel Enterprise under which the Siebel Server is installed.
Siebel Server	The Siebel Server to connect to.

Table 2 Siebel Connection Configuration Parameters (Cont'd)

Field	Description
Object Manager	The name of the Application Object Manager that you want to access. This can be a user-defined component or one of the predefined components, `SCCObjMgr_enu`, `SSEObjMgr`, `ISSObjMgr`, `SSVObjMgr`. (For more information, see *Siebel Server Administration Guide*).
Language	Enter the language to be used while connecting to the Siebel application or click the **Browse** button to select from the list of language parameters. Some of the possible values are enu, chs, cht, csy, dan. You should provide the same value given at the time of the Siebel application's installation.
Transport	Choose one of the following values: `tcpip` or `http`.
Compression	The type of compression for network communications (Possible values are `none`, `zlib`, `or pkware`). If compression is required on both the client and server side when using Resonate, it must be specified for both sides. Each side compresses communications as defined. When Resonate is not used, the server side drives the compression and any client side settings are ignored.
User Name	Specify the username for the account to access the application. Sample value - `sadmin`.
Password	Specify the password for the account to access the application. Sample value - `sadmin`.

Test Connection Button

The Test Connection button allows you to test whether the specified configuration fields result in a valid connection to the Siebel Application Server.

Siebel Objects

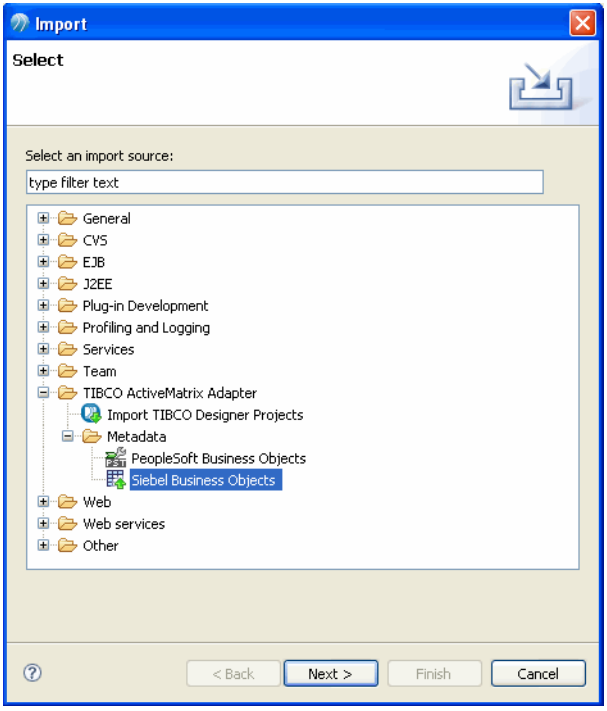
Siebel Objects include Siebel Business Objects and Siebel Integration Objects. This section introduces how to import Siebel Objects and how to convert Siebel Objects into AESchema.

Importing Siebel Objects

To import Siebel Objects, follow these steps:

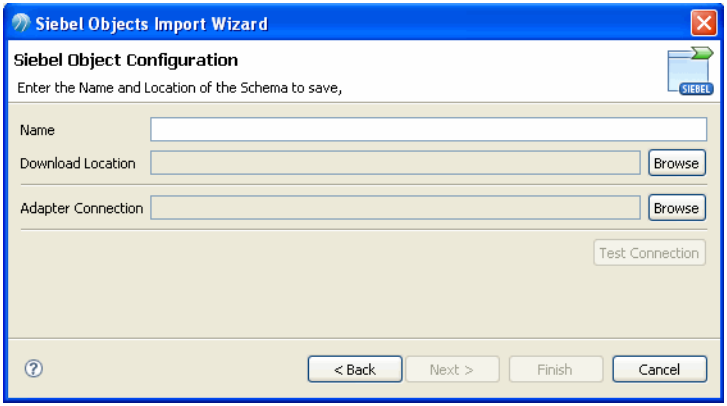
1. Select an adapter project in the Project Explorer Panel, and then select **File > Import** from the Menu. The Import dialog appears.

Figure 23 Import a Siebel Object



2. Expand the **TIBCO ActiveMatrix Adapters** folder and select **Metadata > Siebel Business Objects**. Click the **Next** button to open the Siebel Objects Importer Wizard dialog.

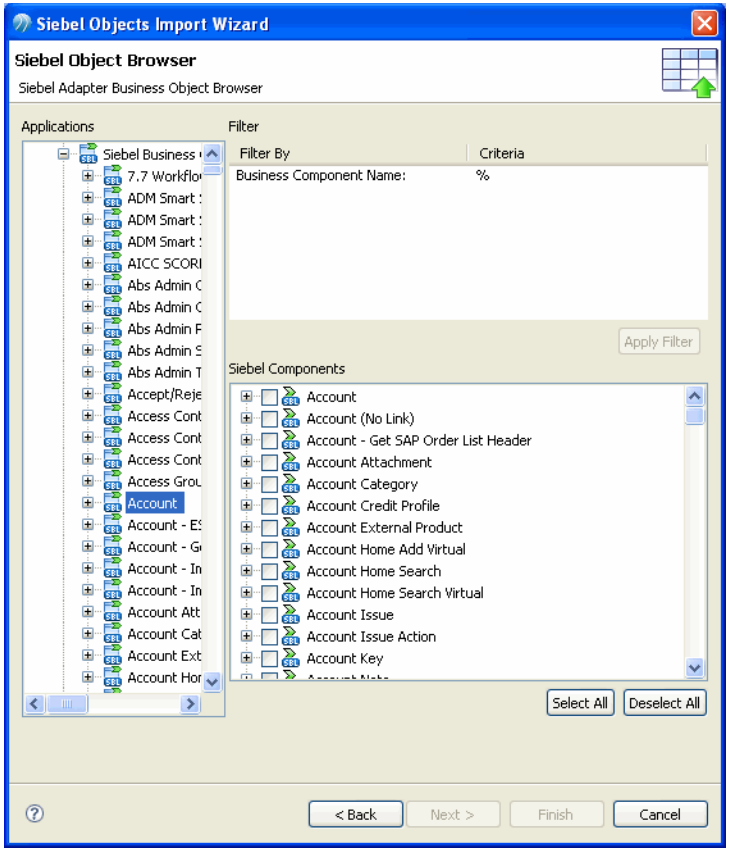
Figure 24 Siebel Objects Importer Wizard - General Configuration



3. Specify the value of each parameter in the dialog, and then click the **Next** button.
   - **Name**: Enter a name to be saved for the schema.
   - **Download Location**: Click the **Browse** button to select a location to save the schema.
   - **Adapter Connection**: Click the **Browse** button to select a pre-configured Siebel Connection.

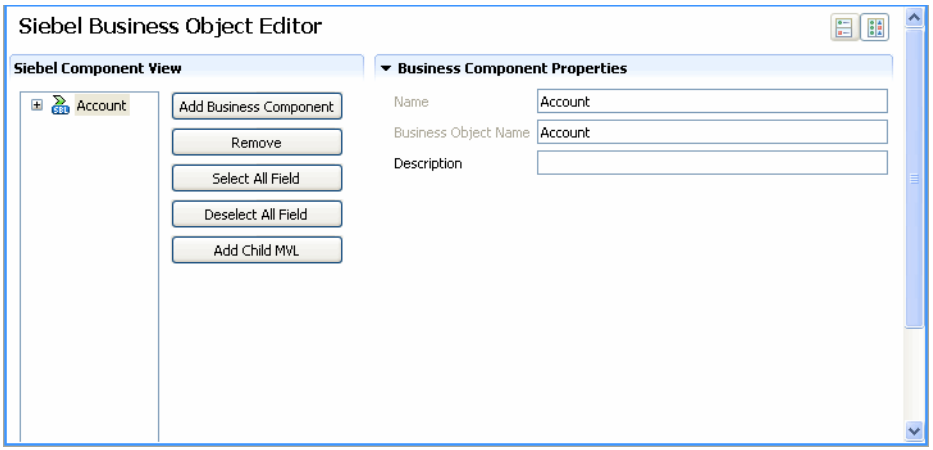
   Click the **Test Connection** button to test whether the specified configuration fields result in a valid connection to the Siebel Application Server.
4. Select the Siebel Objects you want to retrieve from the Siebel Application Server.

Figure 25 Siebel Objects Importer Wizard - Retrieve Siebel Objects



5. Click the **Finish** button to close the dialog. After successfully importing Siebel BOs, a business object file (`.bo`) will be automatically generated in the adapter project.
6. Double-click the business object file under the adapter project folder in the Project Explorer Panel to view the imported Siebel Objects. The Siebel Adapter Business Object Editor panel appears on the right.

Figure 26 Siebel Adapter Business Object Editor



7. View and specify the value for each parameter shown in Figure 26.

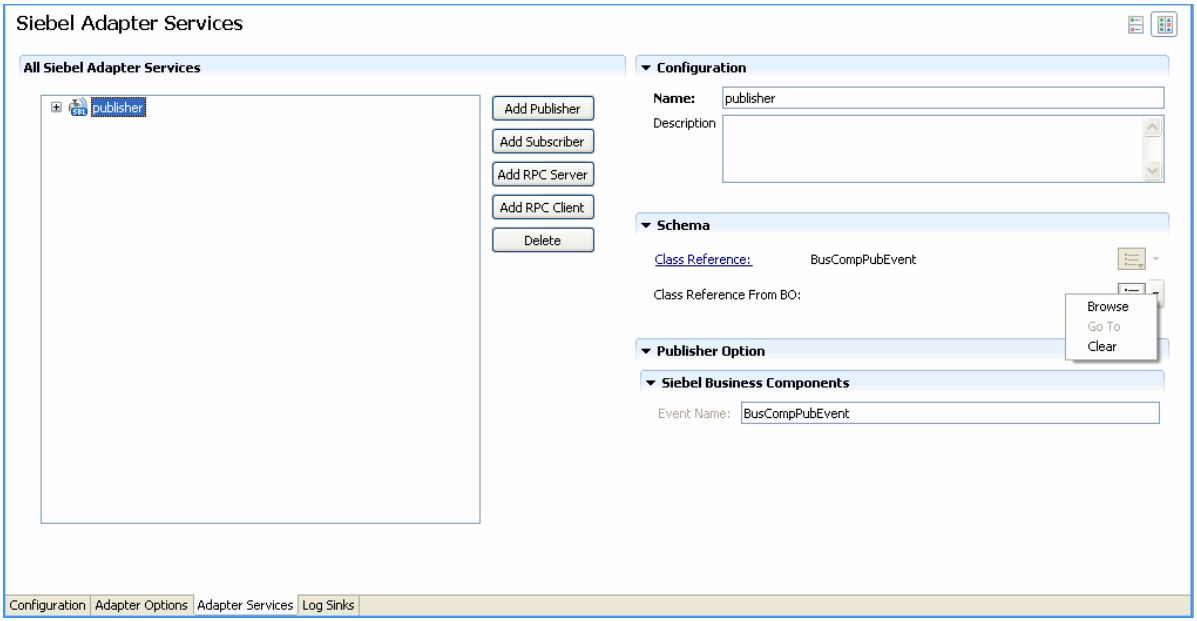
Converting Siebel Objects into AESchemas

After generating a Siebel object file (`.bo`), you need to convert Siebel Objects into AESchemas.

To convert Siebel Objects into AESchemas, follow these steps:

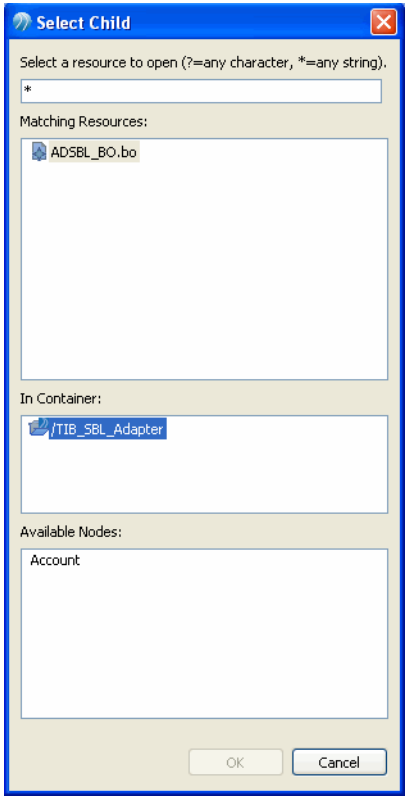
1. Create a Siebel Adapter Configuration for the adapter project. Refer to Siebel Adapter Configuration on page 50 for details.
2. Double-click the configuration file under the adapter project folder in the Project Explorer Panel. The Editor Panel appears on the right with the configuration tabs. Click the **Adapter Services** tab.

Figure 27 Adapter Services Tab



3. Select **Browse** from the Class Reference From the BO drop-down list. The Select Child dialog appears. Select a `.bo` file from the Matching Resources list and select an adapter project from the In Container list. Select the table listed in the Available Nodes list.

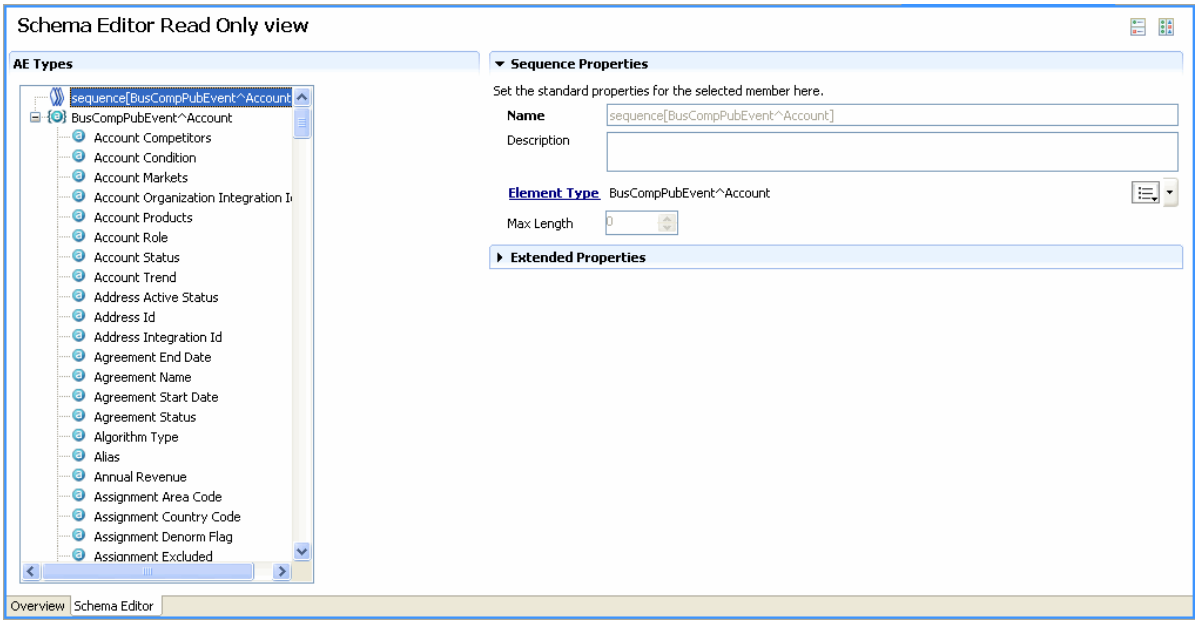
Figure 28 Select Child Dialog



4. Click the **OK** button to close the dialog. A schema class file (.schemamodel) will be generated in `/AESchemas/as/Siebel/ within` the adapter project.

5. Double-click the schema class file in the Project Explorer Panel. The Schema Editor panel appears on the right.

Figure 29 Schema Editor



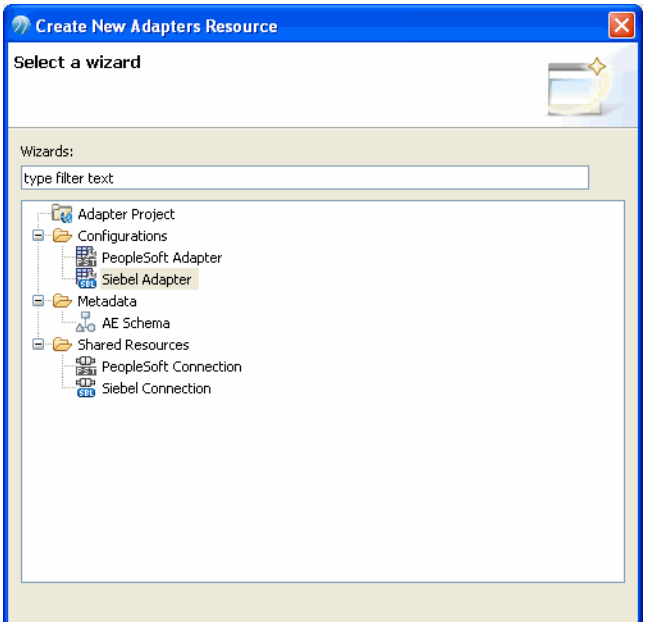
Siebel Adapter Configuration

For a newly created adapter project, create a Siebel Adapter Configuration and configure it to perform tasks.

To create and configure a Siebel Adapter Configuration, follow these steps:

1. Click the newly created project, for example `TIB_SBL_Adapter`, in the Project Explorer Panel, and then select **File > New > Adapter Resources...** from the Menu. The Create New Adapters Resource dialog appears.

Figure 30 Create New Adapters Resource Dialog



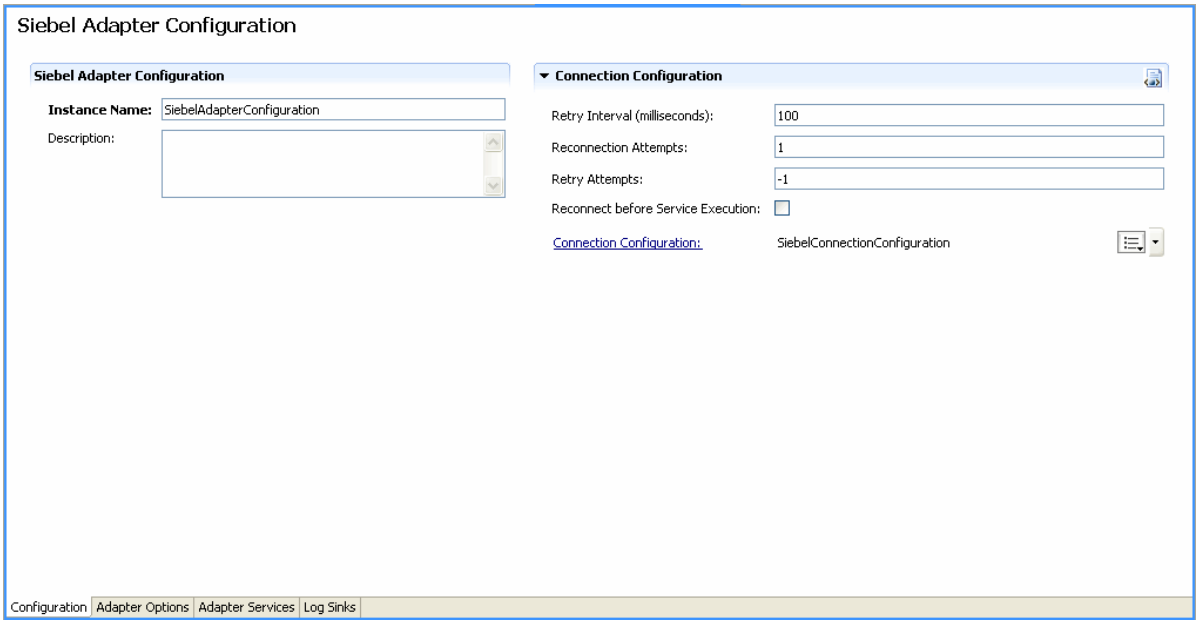
2. Expand the **Configurations** folder and select **Siebel Adapter**. Click the Next button to open the Siebel Adapter Configuration dialog.
3. Enter or select the parent folder for the newly created configuration file and enter the configuration file name in the File name field. The default configuration file name is `SiebelAdapterConfiguration.adsblmodel`.
4. Click the **Finish** button to close the dialog.
5. Double-click the configuration file under the adapter project folder in the Project Explorer Panel. The Editor Panel appears on the right with the configuration tabs.

6. Specify the value for each parameter in the following configuration tabs:
   - Configuration, page 51
   - Adapter Options, page 53
   - Adapter Services, page 55
   - Log Sinks, page 60

Configuration

The Configuration tab provides options to modify basic configuration information.

Figure 31 Configuration Tab



The configuration parameters are listed in Table 3, Configuration Parameters.

Table 3 Configuration Parameters

Field	Description
Siebel Adapter Configuration	
Instance Name	The name of the Adapter Service Engine instance. Use the default name or replace it with a name of your choice. See Guidelines for Choosing an Instance Name on page 52 for more details.
Description	(Optional) A short description of the Adapter Service Engine instance.
Connection Configuration	
Retry Interval	Specify the time interval in milliseconds to elapse between each reconnection attempt.
Reconnection Attempts	Specify the number of reconnection attempts to make before suspending a run-time adapter or adapter service.
Retry Attempts	The total number of reconnection attempts to make before the runtime adapter or adapter service is stopped. The default value of -1 means reconnection attempts will continue indefinitely.
Connection Configuration	Associate the adapter instance with a predefined connection configuration. See Siebel Connection Shared Resource on page 40 for how to define a connection configuration. The options for a connection configuration are listed in Table 2, Siebel Connection Configuration Parameters, on page 41.

Guidelines for Choosing an Instance Name

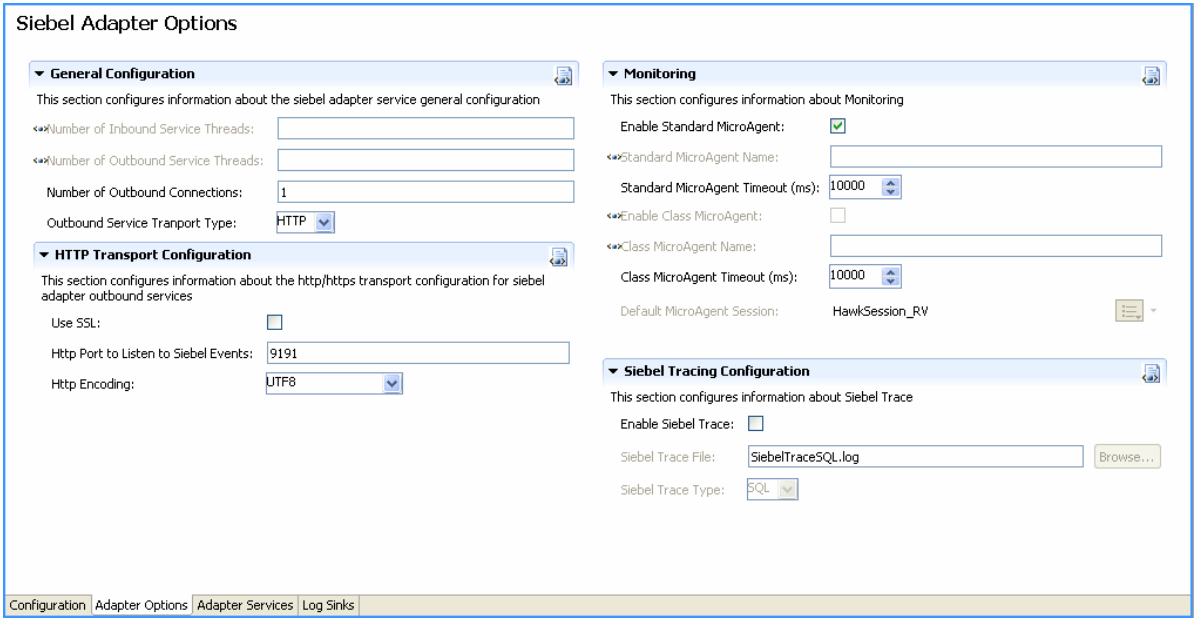
- An instance name must use alphanumeric characters. An underscore (_) character can be used. The entire instance name must be less than 80 characters. The space character cannot be used in an instance name.
- An instance name cannot use global variables.
- An instance name must be unique with respect to other adapter instances for the same adapter in the project. The same instance name can be used to name an adapter instance for a different adapter in the same project. For example, a Siebel adapter instance named `TEST` and a Siebel adapter instance named `TEST` can coexist in the same project.
- Each instance name must be unique per adapter within a project even if each instance is defined in a different folder. That is, configuring same-named adapter instances in different folders will not make their names unique.

When you create an Adapter Service Engine instance, several resources are automatically created for it. The names of these resources derive from the name of the instance they belong to. Changing the adapter instance name results in an automatic regeneration of the resources names. If you manually modify any resource name, that particular name will *not* be automatically regenerated next time you rename the adapter instance.

Adapter Options

The Adapter Options tab provides options to modify base configuration information and monitoring information.

Figure 32 Adapter Options Tab



The configuration parameters for Siebel Adapter Options are listed in Table 4, Adapter Options Parameters.

Table 4 Adapter Options Parameters

Field	Description
Base Configuration	
Language Encoding	Select the encoding from the drop down menu. The adapter may support other encodings not shown.

Table 4 Adapter Options Parameters (Cont'd) (Cont'd)

Field	Description
Polling Interval	The polling interval in milliseconds. This is how often an adapter with a publication service checks the publishing table (`TIB_CI_MQUEUE`) for new rows. The default setting is `5000`, or once every five seconds. Note: This field is applicable only for the CI Publication Service.
Number of Threads	The number of connections to be opened with the Siebel application server by the Adapter Service Engine.
Monitoring (TIBCO HAWK functionality is not supported in the current release.)	
Enable Standard MicroAgent	Check this check box to enable the standard TIBCO Hawk microagent.
Standard MicroAgent Name	The name for the standard microagent that will be registered with the TIBCO Hawk system. In most cases the default value is used.
Standard MicroAgent Timeout (ms)	The timeout for the standard microagent.
Enable Class MicroAgent	Check this check box to enable instance or class specific standard TIBCO Hawk Microagent.
Class MicroAgent Name	The name for the class microagent that will be registered with the TIBCO Hawk system. In most cases the default value is used.
Class MicroAgent Timeout (ms)	The timeout for the class microagent.
Default MicroAgent Session	(Read only) The value of this field is set to `HawkSession`.

Adapter Services

The Adapter Service tab provides options for adding and configuring the following adapter services, as shown in Figure 33.

- Publisher Service, page 55
- Subscriber Service, page 56
- RPC Server Service, page 57
- RPC Client Service, page 59

Figure 33 Adapter Services Tab

To add a service, follow these steps:

1. Click the **Add *Service Name*** button to add a desired service into the All Siebel Adapter Services pane. The parameter configuration panes appear in the tab.
2. Specify the value of each parameter for the added service.

For more information on each service, refer to *TIBCO ActiveMatrix Adapter for Siebel Concepts*.

Publisher Service

The configuration parameters for Publisher Service are listed in Table 5.

Table 5 Publisher Service Configuration parameters

Field	Description
Configuration	

Table 5 Publisher Service Configuration parameters (Cont'd)

Field	Description
Name	The name of Publisher Service. A service name must use alphanumeric characters. An underscore (_) character can be used. The entire instance name must be less than 80 characters. The space character cannot be used in an instance name. A service name cannot use substitution variables.
Description	(Optional) Additional information about the service being configured.
Schema	
Class Reference	The AESchema that is automatically generated from the selected Siebel Business Object when the adapter configuration is saved.
Class Reference From BO	Convert Siebel Business Objects into AESchemas. Click **Browse** in the drop-down list to select a business object child from any of the previously imported BOs. Refer to Converting Siebel Objects into AESchemas on page 46 for details.
Publisher Option (Component Interface)	
Publisher Event:	The Publication Service supports publication of the following components.
Siebel Business Components	The adapter queries the Siebel Business Components based on the search criteria specified and publishes them on appropriate subject names configured.
Non-Siebel Business Components	The adapter publishes the data that comes from the Siebel application, which invokes the adapter without querying Siebel.
Siebel Integration Components	The adapter queries the Siebel Integration Components based on the search criteria specified and publishes them on appropriate subject names configured.
Send Complete Integration Object	The adapter publishes the complete integration object received from the Siebel application in the form of an http request, without querying Siebel application.

Subscriber Service

The configuration parameters for Subscriber Service are listed in Table 6.

Table 6 Subscriber Service Configuration Parameters

Field	Description
Configuration	

Table 6 Subscriber Service Configuration Parameters (Cont'd)

Field	Description
Name	The name of the Subscriber Service. A service name must use alphanumeric characters. An underscore (_) character can be used. The entire instance name must be less than 80 characters. The space character cannot be used in an instance name. A service name cannot use substitution variables.
Description	(Optional) Additional information about the service being configured.
Schema	
Class Reference	The AESchema that is automatically generated from the selected Siebel Business Object when the adapter configuration is saved.
Class Reference From BO	Convert Siebel Business Objects into AESchemas. Click **Browse** in the drop-down list to select a business object child from any of the previously imported BOs. Refer to Converting Siebel Objects into AESchemas on page 46 for details.
Subscriber Option (Component Interface)	
Subscriber Event:	The Subscription Service supports Subscription of the following components.
Siebel Business Components	The adapter upserts, updates, inserts, or deletes Siebel Business Components when it receives a message on a subject.
Siebel Integration Components	The adapter upserts, updates, inserts, or deletes Siebel Integration Components when it receives a message on a subject.
Invoke Siebel Workflow	The adapter invokes the Siebel Workflow when it receives a message on a subject.
Invoke Siebel Business Service	The adapter invokes the Business Service when it receives a message on a subject.

RPC Server Service

The configuration parameters for RPC Server service are listed in Table 7, RPC Server Service Configuration Parameters.

Table 7 RPC Server Service Configuration Parameters

Field	Description
Configuration	

Table 7 RPC Server Service Configuration Parameters (Cont'd)

Field	Description
Name	The name of the CI RPC Server Service. A service name must use alphanumeric characters. An underscore (_) character can be used. The entire instance name must be less than 80 characters. The space character cannot be used in an instance name. A service name cannot use substitution variables.
Description	(Optional) Additional information about the service being configured.
Schema	
Class Reference	The AESchema that is automatically generated from the selected Siebel Business Object when the adapter configuration is saved.
Class Reference From BO	Convert Siebel Business Objects into AESchemas. Click **Browse** in the drop-down list to select a business object child from any of the previously imported BOs. Refer to Converting Siebel Objects into AESchemas on page 46 for details.
RPC Server Option (Component Interface)	
Query Siebel Business Components	The adapter allows an external application to query Siebel Business Components
Insert/Update Siebel Business Components	The adapter allows an external application to insert or update Siebel Business Components
Query Siebel Integration Components	The adapter allows an external application to query Siebel Integration Components.
Insert/Update Siebel Integration Components	The adapter allows an external application to insert or update Siebel Integration Components
Invoke Siebel Workflow	The adapter allows an external application to invoke Siebel Workflow.
Invoke a Siebel Business Service	The adapter allows an external application to invoke a Siebel Business Service.

RPC Client Service

The configuration parameters for RPC Client service are listed in Table 8, RPC Client Service Configuration Parameters.

Table 8 RPC Client Service Configuration Parameters

Field	Description
Configuration	
Name	The name of the RPC Client Service. A service name must use alphanumeric characters. An underscore (_) character can be used. The entire instance name must be less than 80 characters. The space character cannot be used in an instance name. A service name cannot use substitution variables.
Description	(Optional) Additional information about the service being configured.
Schema	
Class Reference	The AESchema that is automatically generated from the selected Siebel Business Object when the adapter configuration is saved.
Class Reference From BO	Convert Siebel Business Objects into AESchemas. Click **Browse** in the drop-down list to select a business object child from any of the previously imported BOs. Refer to Converting Siebel Objects into AESchemas on page 46 for details.
RPC Server Option (Component Interface)	
Query Non Siebel Business Component	The adapter generates the request to the external application with the data from the Siebel application that invokes the adapter, without querying Siebel.
Query Siebel Business Components	The adapter queries Siebel Business Component data and generates a request to the external application.
Query Siebel Integration Components	The adapter queries Siebel Integration Component data and generates a request to the external application.
Query Siebel Complete Integration Object	The adapter queries Siebel Complete Integration Object data and generates a request to the external application.

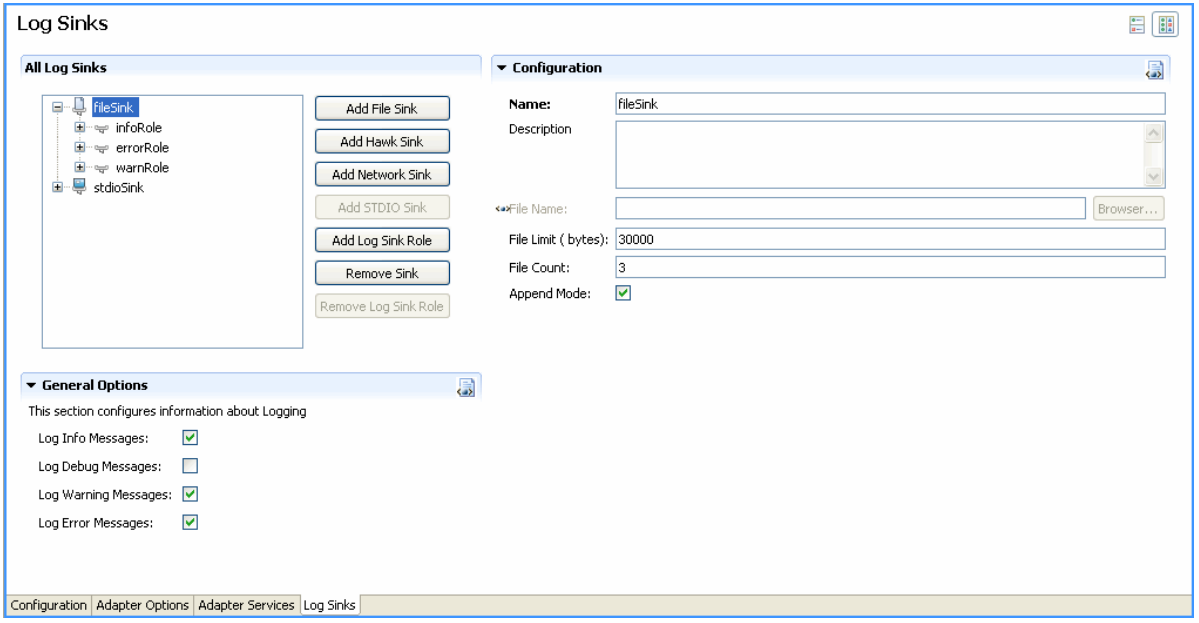
Log Sinks

The Log Sinks tab specify the output destination (sinks) for the trace messages and set the tracing level for the roles selected, as shown in Figure 34.

The following operations are introduced in this section:

- Adding a Sink, page 60
- Removing a Sink, page 62
- Adding Roles to a Sink, page 62
- Removing Roles from a Sink, page 63

Figure 34 Log Sinks Tab



Adding a Sink

To add a sink, follow these steps:

1. Click the **Add *Sink Type*** button to add a desired sink into the All Log Sinks pane. The parameter configuration panes appear in the tab.

2. Specify the value of each parameter for the added sink, as shown in Table 9.

Table 9 Log Sink Configuration Parameters

Field	Description
General Options	
Trace messages from the selected level(s) will be collected in the configured log sinks. You can configure what levels of trace messages you want logged.	
Log Info Messages	Check this checkbox to log Informational messages.
Log Debug Messages	Check this checkbox to log Debug messages. This trace level is reserved and should not be enabled unless requested by the TIBCO Support. This trace level writes a lot of information to the log file and significantly reduces the speed of the Adapter Service Engine.
Log Warning Messages	Check this checkbox to log Warning messages.
Log Error Messages	Check this checkbox to log Error messages.
File Sink	
Name	The name and description for the File sink.
Description	
File Name	The path and name of the trace file.
File Limit (bytes)	Maximum size of the file, in bytes. The default value is 30000.
File Count	The number of rollover files. The default value is 3.
Append Mode	If checked, traces are added to the existing file at startup. If unchecked, the existing file is overwritten at startup if one of the same name exists. Only true and false are legal values.
STDIO Sink	
Name	The name and description for the STDIO sink.
Description	
Output Stream	Choose between `out` and `error`.

Table 9 Log Sink Configuration Parameters (Cont'd)

Field	Description
Hawk Sink	
The Hawk sink uses the Hawk session, created and used by the Adapter Service Engine for monitoring purposes, to send tracing messages to the TIBCO Hawk monitor or Display. (TIBCO HAWK functionality is not supported in the current release.)	
Name	The name and description for the Hawk sink.
Description	
Microagent Name	The name of the microagent for traces from this Hawk sink.
Network Sink	
The Network sink is used to publish tracing message.	
Name	The name and description for the Network sink.
Description	
Subject	Subject of the TIBCO Rendezvous messages to be sent.
Session Reference	Click **Browse** from the drop-down list to select one of the previously defined sessions.

Removing a Sink

To remove a log sink, follow these steps:

1. Select the log sink you want to remove in the All Log Sinks pane.
2. Click the **Remove Sink** button to remove the sink.

Adding Roles to a Sink

To add roles to a sink, follow these steps:

1. Select the log sink you want to add roles into in the All Log Sinks pane.
2. Click the **Add Log Sink Role** button to add additional roles. You can add debug, error, warning, or informational roles to sinks by selecting the corresponding roles from the Role drop-down list.

Removing Roles from a Sink

To remove roles from a sink, follow these steps:

1. Select the role you want to remove from a sink in the All Log Sinks pane.
2. Click the **Remove Log Sink Role** button to remove the role.

Chapter 5 Working with ActiveMatrix SOA Projects

This chapter describes the process of working with ActiveMatrix SOA projects.

Topics

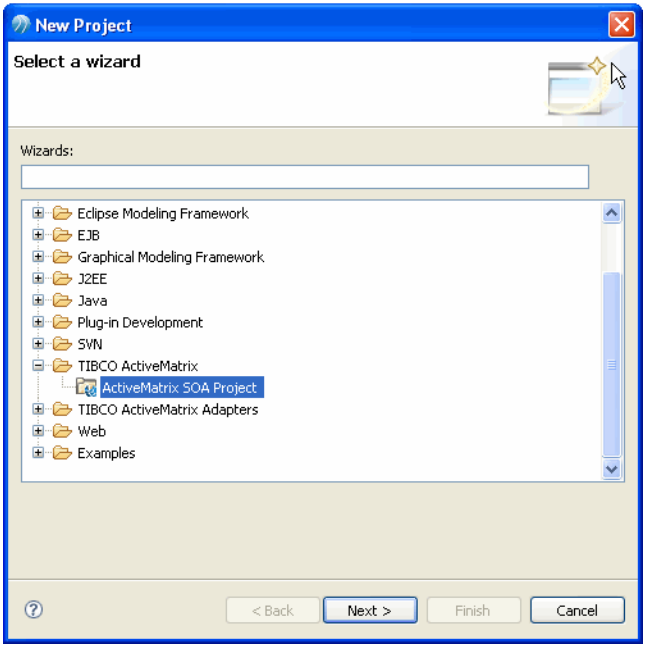
Creating an ActiveMatrix SOA Project

ActiveMatrix SOA projects contain assets that you can develop with TIBCO Business Studio. Before creating any type of asset you must first create an ActiveMatrix SOA project using TIBCO Business Studio.

To create an ActiveMatrix SOA project, follow these steps:

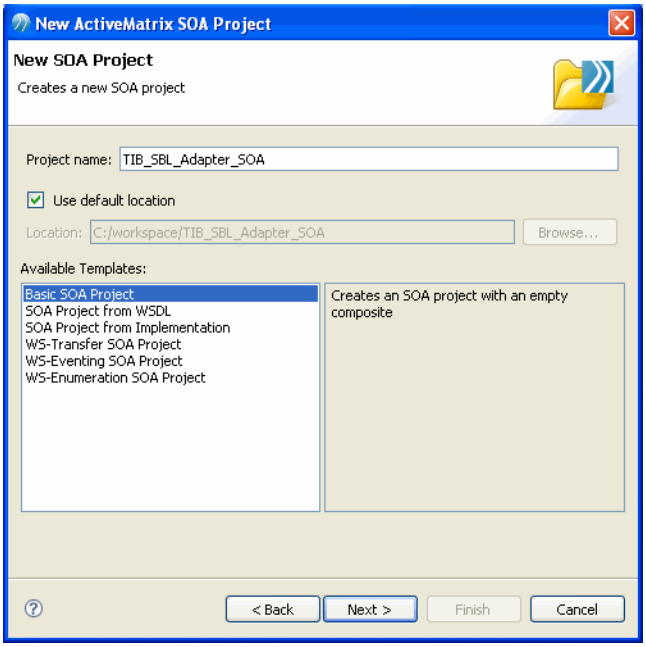
1. Start TIBCO Business Studio.
2. Select **File > New > Project** from the Menu to open the New Project window.
3. Select **TIBCO ActiveMatrix > ActiveMatrix SOA Project** in the Select a wizard page and then click the **Next** button.

Figure 35 Create an ActiveMatrix SOA project - Select a Wizard



4. Do the following operations in the New SOA Project page and then click the **Next** button.
   - Type the SOA project name (For example, *TIB_SBL_Adapter_SOA*) in the Project name field.
   - Check the **Use default location** check box if you want to save the SOA project to your default workspace.
   - Select **Basic SOA Project** from the Available Templates list. The Basic SOA Project template creates a project with an empty composite.

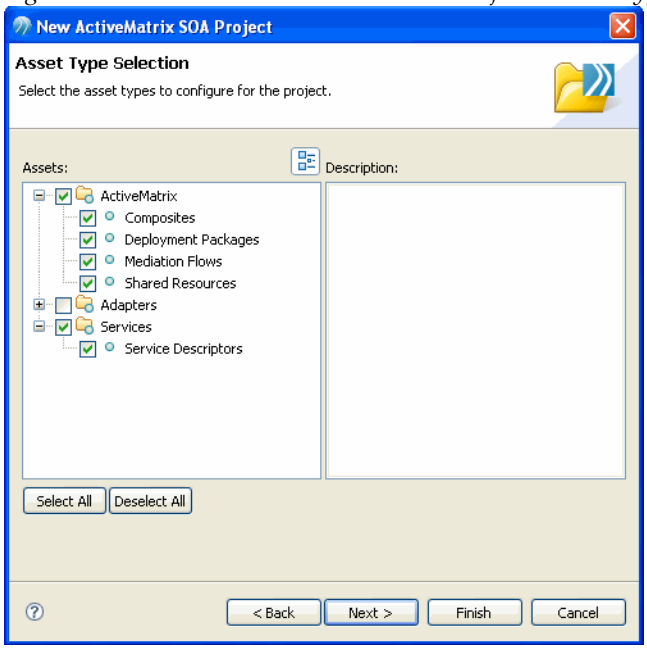
Figure 36 Create an ActiveMatrix SOA Project - New SOA Project



5. Select the asset types to configure for the project in the Asset Type Select page.

Do not check the **Adapters** check box in the Asset Type Select page.

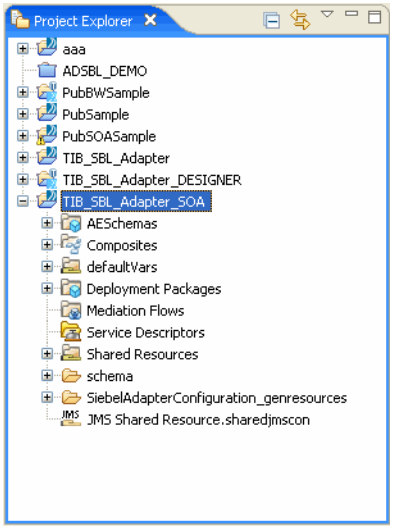
Figure 37 Create an ActiveMatrix SOA Project - Asset Type Selection



6. Specify the names for the selected assets.
   - Click the **Next** button to enter the desired name for each asset in the corresponding Folder field.
   - Click the **Finish** button to accept all default names for the selected assets.

The newly created ActiveMatrix SOA project is shown in the Project Explorer panel.

Figure 38 Newly Created ActiveMatrix SOA Project



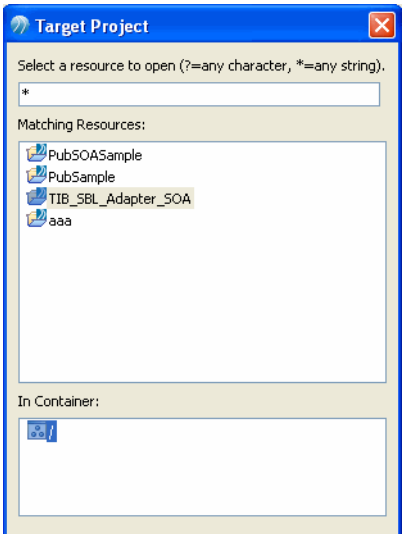
Generating an Adapter WSDL

TIBCO ActiveMatrix services are described using WSDL files.

To generate the WSDL file, follow these steps:

1. Right-click the Siebel Adapter Configuration file (`.adsblmodel`) for the Adapter project you want to deploy in the Project Explorer panel and select **Generate Adapter WSDL** from the pop-up menu. The Target Project dialog appears.

Figure 39 Target Project Dialog



2. Select the ActiveMatrix SOA project you want to use, and then click the **OK** button.

After the generating process is completed, the following two folders are created automatically.

- `schema`: this folder contains the adapter XSD schema.

- *SiebelAdapterConfigurationFilename*_`genresources`: this folder contains the following supporting resources:
  - *SiebelAdapterConfigurationFilename_number*.`dat`: contains information used by the Adapter container.
  - *SiebelAdapterConfigurationFilename*.`adendpoints`: the adapter service endpoints file which contains information about the service endpoints supported by the adapter. The file is used later to select service types.
  - *SiebelAdapterConfigurationFilename*.`substvar`: contains substitution variables that are imported from the adapter configuration file.
  - *SiebelAdapterConfigurationFilename*.`wsdl`: the generated WSDL file which references the schema and `.dat` files contains the service definition endpoints.

After using a WSDL in a composite, ensure that you do not overwrite the WSDL.

Creating a TIBCO Shared Resource

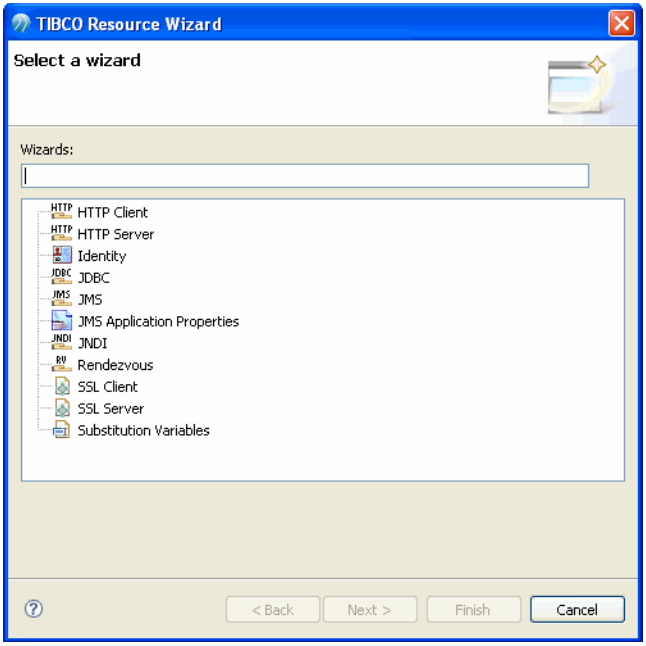
After creating an ActiveMatrix SOA project, a composite file for this project is automatically generated. For detailed information about configuring the composite, refer to Configuring Composites on page 76.

Composites must have a service added to it. For detailed information about how to add a composite service, refer to Creating a Composite Service on page 80. In order for the composite service to be accessible from the external client, you need to add a TIBCO Shared Resource.

To add a TIBCO Shared Resource, follow these steps:

1. Select your ActiveMatrix SOA project in the Project Explorer Panel, and then select **File > New > TIBCO Shared Resources...** from the Menu. The TIBCO Resource Wizard dialog appears.

Figure 40 TIBCO Resource Wizard - Select a Wizard



2. Select a shared resource according to the composite service type, and click the **Next** button.
   - For JMS service, select **JMS**.
   - For SOAP service, select **HTTP Server**.

For TIBCO ActiveMatrix Adapter Service Engine for Siebel, only JMS and SOAP services are available.

3. Enter or select the parent folder for the newly created TIBCO Shared Resource file and enter the name for the shared resource file in the File name field. The default shared resource file name is `HTTP Server Shared Resource` or `JMS Shared Resource`.
4. Click the **Finish** button to close the dialog. The newly created TIBCO Shared Resource is displayed in the Project Explorer panel.

Configuring a TIBCO Shared Resource

To configure a TIBCO Shared Resource, follow these steps:

1. Double-click the shared resource file under the ActiveMatrix SOA project folder in the Project Explorer Panel. The Shared Resource Editor panel appears on the right.

2. Specify the values for the configuration parameters.
   — For JMS Shared Resource, the configuration parameters are listed in Table 10, JMS Shared Resource Configuration Parameters.

Table 10 JMS Shared Resource Configuration Parameters

Field		Description
Configuration		
Name		The name and description for the Shared Resource. The name is case sensitive. The default name is `JMS Shared Resource`.
Description		
Connection Type		
Direct	Provider URL	The server URL used to create the connection factory. This field is required only for connections to a TIBCO Enterprise Message Service server. See documentation for `TibjmsConnectionFactory` in TIBCO Enterprise Message Service product documentation (Javadoc) for more information. The default value is `tcp://localhost:7222`.
	Connection Factory	The factory for creating connections to the JMS server. Set to `com.tibco.tibjms.TibjmsConnectionFactory` for Enterprise Message Service or `progress.message.jclient.ConnectionFactory` for SonicMQ. The default value is `com.tibco.tibjms.TibjmsConnectionFactory`.
JNDI	JNDI Configuration Name	A JNDI shared resource definition that specifies the JNDI connection information. Click **Browse** from the drop-down list to select a JNDI configuration resource.
	Connection Factory	The `ConnectionFactory object` stored in JNDI. This object is used to create a topic or queue connection with a JMS application. See your JNDI provider documentation for more information about creating and storing `ConnectionFactory` objects. The default value is `GenericConnectionFactory`. To create an SSL connection to the JMS server, specify SSLGenericConnectionFactory or any other connection factory with SSL configuration provisioned on the JNDI server.
	Client Certificate Password	The password of the client certificate used to verify the identity of SSL clients.
More Options		

Table 10 JMS Shared Resource Configuration Parameters (Cont'd)

Field	Description
User Identity	User identity of the Username/Password type. If the JMS provider does not require access control, this field can be empty. Not all JMS servers require user names and passwords. Refer to your JMS provider documentation and consult your system administrator to determine if your JMS server requires a user name and password.
Client ID	Client ID for the connection. Typically JMS providers have a provider-specific format for client IDs. See your JMS provider's documentation for more information about client IDs. If specified, the client ID must be unique within the scope of connections within the server.

— For HTTP Server Shared Resource, the configuration parameters are listed in Table 11.

Table 11 HTTP Server Shared Resource Configuration Parameters

Field	Description
Configuration	
Name	The name and description for the Shared Resource. The name is case sensitive. The default name is `HTTP Server Shared Resource`.
Description	
Host	Specify the name of the host that accepts the incoming requests.
Port	The port number on which to listen for incoming HTTP requests. The default value is 80.
More Options	
SSL Configuration	The SSL server shared resource used for the connection. The field label is a link to the resource. Click **Browse** from the drop-down list to select a SSL configuration resource.
Minimum Threads	The minimum number of threads available for incoming HTTP requests. The HTTP server creates the number of threads specified by this parameter when it starts up. The default value is 10.
Maximum Threads	The maximum number of threads available for incoming HTTP requests. The HTTP server will not create more than the number of threads specified by this parameter. When a client sends a request that cannot be processed because no threads are available, the ActiveMatrix node returns the ConnectionRefused exception to the client. The default value is 75.

Configuring Composites

In the ActiveMatrix platform, a *composite* exports a cohesive set of business functions as services. A composite consists of one or more components. A *component* contains a configured implementation, where the implementation provides the business functions.

The component configures the implementation using specific values for custom properties and by connecting component references to component services or composite references. The connections between references and services are represented by wires.

To configure a composite, complete the following tasks:

1. Launching the Composite Editor, page 76
2. Creating a Siebel Adapter Component, page 77
3. Adding a Shared Resource to a Composite, page 79
4. Creating a Composite Service, page 80

For more information, refer to the TIBCO ActiveMatrix Service Grid documentation.

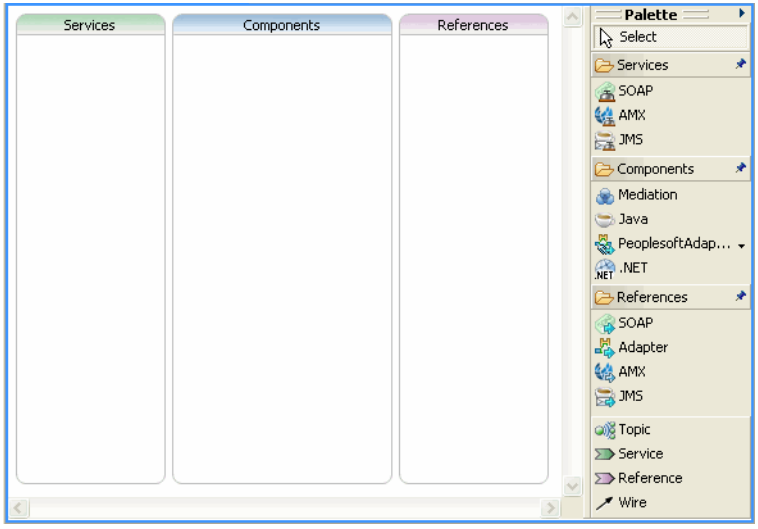
Launching the Composite Editor

The Composite Editor is a graphical editor for developing composites.

To launch the Composite Editor, follow these steps:

1. Expand ***SOAProject*** **> Composites** in the Project Explorer panel. For example, if the name of the ActiveMatrix SOA project you created is *TIB_SBL_Adapter_SOA*, expand **TIB_SBL_Adapter_SOA > Composites** in the Project Explorer panel.
2. Double-click the *SOAProject*`.composite` file under the Composites folder to launch the Composite Editor on the right.

Figure 41 The Composite Editor



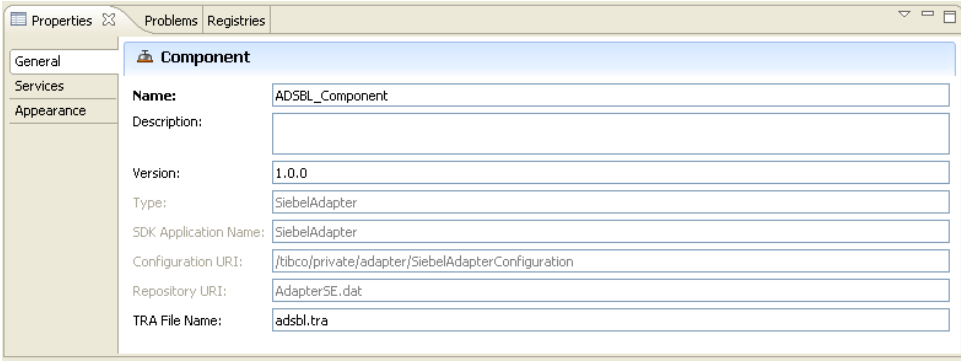
Creating a Siebel Adapter Component

A component contains a configured implementation, where the implementation provides the business functions.

To create a Siebel Adapter Component for an ActiveMatrix SOA project composite, follow these steps in the Composite Editor:

1. Drag the **SiebelAdapter** component from the Palette to the Components column and enter a name for the component (for example, `ADSBL_Component`).
2. Click the component, the configuration parameters for the component are displayed under the Properties Views panel.

Figure 42 Component Properties Tab



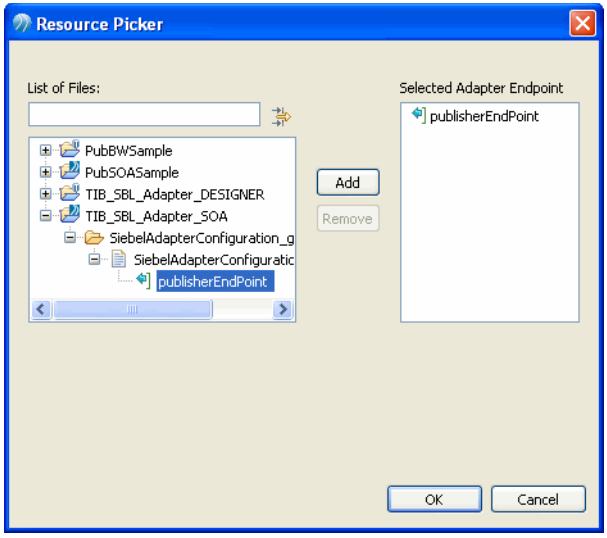
— Click the **General** tab to configure the component. The configuration parameters are listed in Table 12.

Table 12 Component Configuration Parameters

Field	Description
Name	The name of the adapter component.
Description	(Optional) A short description of the adapter component.
Version	The version number of the adapter component.
Type	(Read-only) The adapter type.
SDK Application Name	(Read-only) The instance name of your adapter configuration. Gets populated after adding services to the component.
Configuration URI	(Read-only) The adapter configuration that the service uses. Gets populated after adding services to the component.
Repository URI	(Read-only) This file contains information used by the container to communicate with the adapter service engine. Gets populated after adding services to the component.
TRA File Name	(Read-only) The TRA file to be used.

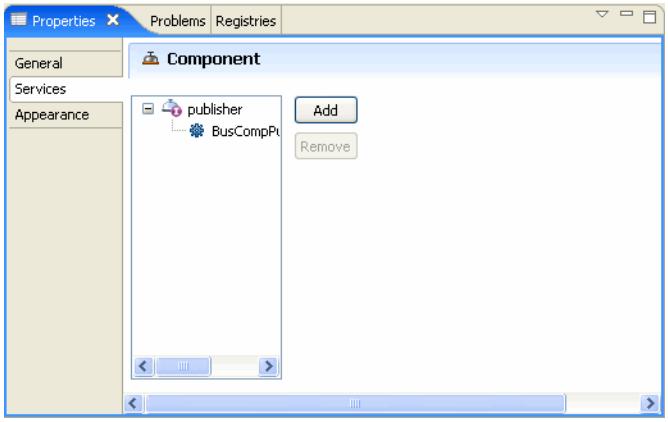
— Click the **Services** tab, and then Click the **Add** button to open the Resource Picker dialog. Expand the tree in the left pane and select an adapter endpoint, and then click the **Add** button to add it into the Selected Adapter Endpoint pane on the right.

Figure 43 Resource Picker Dialog



3. Click the **OK** button. The added service appears under the Services tab.

Figure 44 Add a Service to the Component



4. Click the **Save** button to save your configuration.

Adding a Shared Resource to a Composite

In section Creating a TIBCO Shared Resource on page 72, you have created the HTTP Server Shared Resource or JMS Shared Resource for your composite service. Now you need to add the shared resources to the composite.

To add the HTTP Server Shared Resource or JMS Shared Resource to the composite, follow these steps:

1. Click the canvas in the Composite Editor. The Composite Properties Views panel appears under the Composite Editor.
2. Click the **Shared Resource Profiles** tab to add the shared resource.
   - Click the **Add** button to add a shared resource.

     To change the name of the shared resource, click on its name in the Name column.

     Select **HTTP** or **JMS** in the Type column according to the type of the shared resource you created.

     Click the **Browse** button in the Target column to select a shared resource in the Select Shared Resource dialog.
   - Select an added shared resource and click the **Delete** button to delete it from the list.
3. Click the **Save** button to save your configuration.

Creating a Composite Service

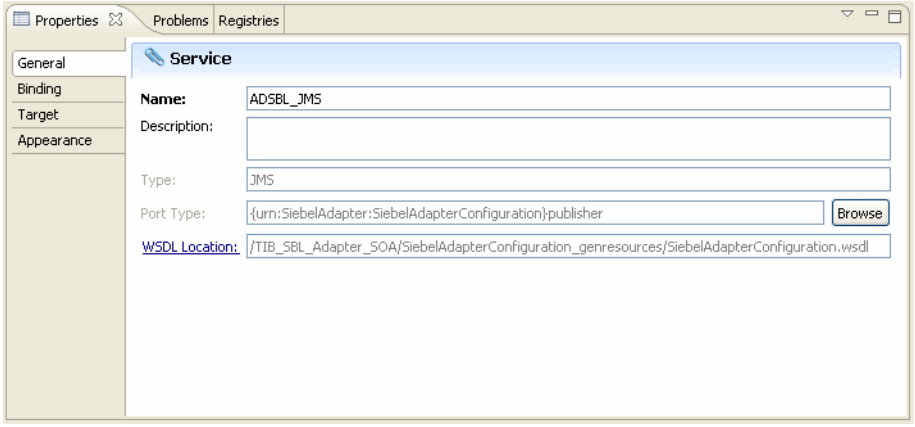
A composite service represents a port type and it enables consumers outside the ActiveMatrix environment to access port types implemented within ActiveMatrix.

For TIBCO ActiveMatrix Adapter Service Engine for Siebel, only JMS and SOAP services are available.

To create a composite service in the Composite Editor, follow these steps:

1. Drag a service from the Palette to the Services column and enter a name for the service (for example, `ADSBL_SOAP` or `ADSBL_JMS`).
2. Click the service, the configuration parameters of the service are displayed under the Properties Views panel.

Figure 45 Service Properties Tab



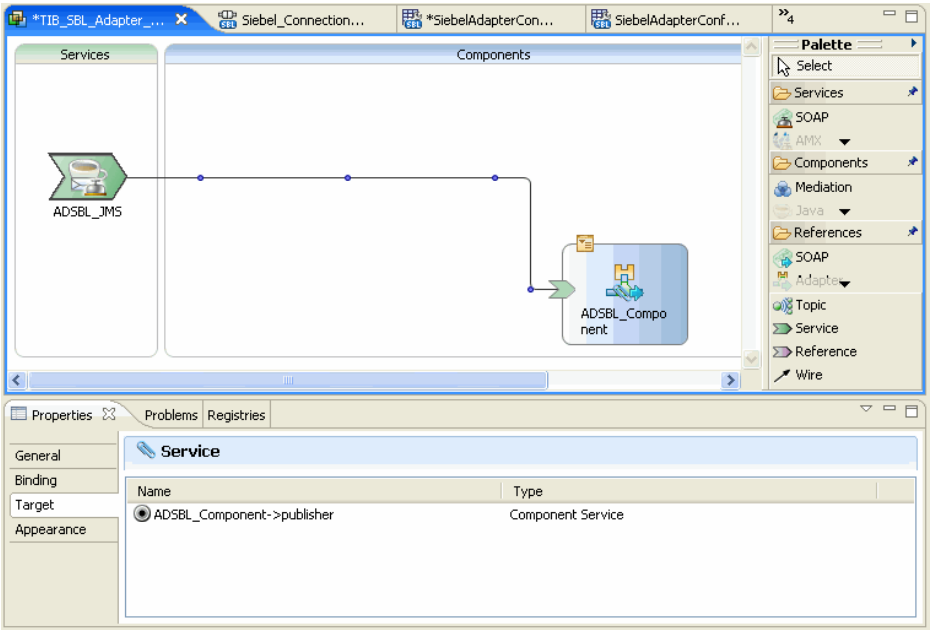
— Click the **General** tab to configure the service. The configuration parameters are listed in Table 13.

Table 13 Service Configuration Parameters

Field	Description
Name	The name of the composite service.
Description	(Optional) A short description of the composite service.
Type	(Read-only) The composite service type.
Port Type	(Read-only) The port type defined in the WSDL file.
WSDL Location	(Read-only) The location of the WSDL file existed in the ActiveMatrix SOA project.

— Click the **Target** tab and click the radio button next to the target component service. A wire is drawn between the service and the component service.

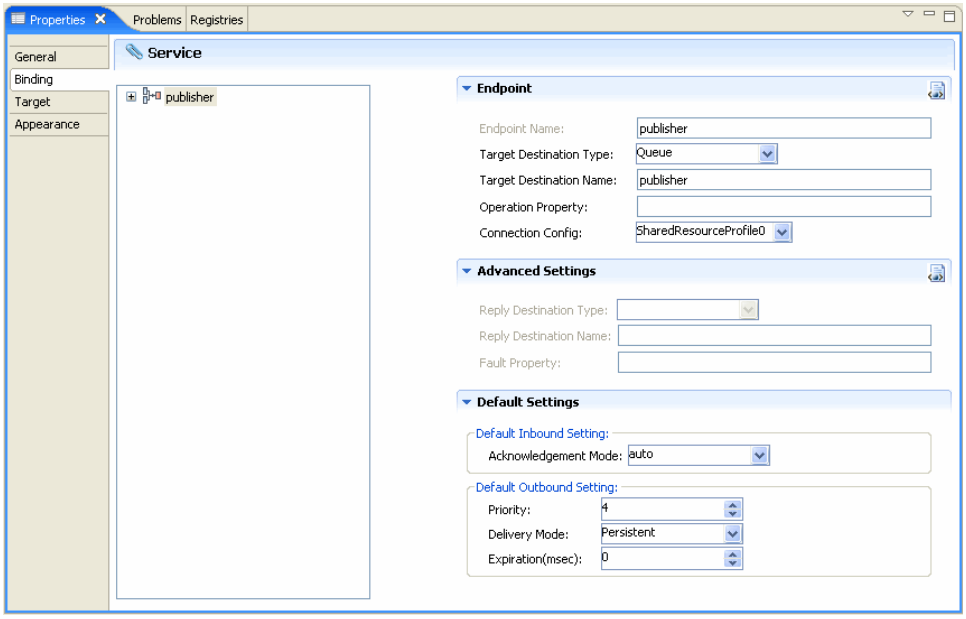
Figure 46 Component and Composite Service Binding



— Click the Binding tab to add the shared resource profile.

For JMS service, specify the shared resource profile in the Endpoint pane.

Figure 47 Endpoint Configuration for the JMS service



For SOAP service, specify the transport type and shared resource profile in the Transport Configuration pane, as shown in step 3. Click the **Generate WSDL** button to generate another WSDL file in the Composites folder of the ActiveMatrix SOA project. This WSDL file will later be used by an external client to access the service within ActiveMatrix.

3. Click the **Save** button to save your configuration.

Creating a Service Assembly and a Service Assembly Archive

The adapter services that have been created in the design phase have to be packaged into a service assembly before they can be deployed.

A service assembly contains service units and a descriptor that indicates the container into which each service unit is to be deployed. Then it is used to create a deployable archive called the service assembly archive.

For detailed information about how to create a service assembly and a service assembly archive, refer to Creating a Service Assembly Archive on page 87.

Chapter 6 Deploying the Service Assembly Archive

This chapter describes how to deploy a service assembly archive using TIBCO ActiveMatrix Administrator.

Topics

Overview

The adapter Service Engine enables the adapter to run in the ActiveMatrix environment as a container and to provide services.

Adapter projects created using TIBCO Designer can also be packaged for deployment in the ActiveMatrix environment.

Adapter projects can be designed to run in the following modes:

- Wired Mode

  The adapter component is wired with other composite elements in the ActiveMatrix SOA project. Projects running in this mode interact with other components in the ActiveMatrix environment.

  Projects created using TIBCO Business Studio always run in the wired mode.

  For more information about creating the service assembly archive in TIBCO Business Studio, refer to Creating a Service Assembly and a Service Assembly Archive on page 84.

- Unwired Mode

  The adapter component in the ActiveMatrix SOA project is not wired with other composite elements and does not participate in the ActiveMatrix environment. Projects running in the unwired mode do not provide services to or consume services from other ActiveMatrix components. The adapter configurations merely run in the ActiveMatrix node but the deployments can be life-cycled using TIBCO ActiveMatrix Administrator.

  Projects imported using the EAR file always run in the unwired mode.

  For more information about creating a service assembly archive from an EAR file, refer to Creating a Service Assembly Archive on page 87.

Creating a Service Assembly Archive

This section introduces how to create a service assembly archive for different adapter projects.

- For Adapter Projects Running in the Wired Mode, page 87
- For Adapter Projects Running in the Unwire Mode (Using EAR2SA Tool), page 89

For Adapter Projects Running in the Wired Mode

Adapter projects created using TIBCO Business Studio always run in the Wired Mode. For detailed information about how to create an adapter project in TIBCO Business Studio, refer to Chapter 4, Configuring an Adapter Service Engine Instance, on page 39.

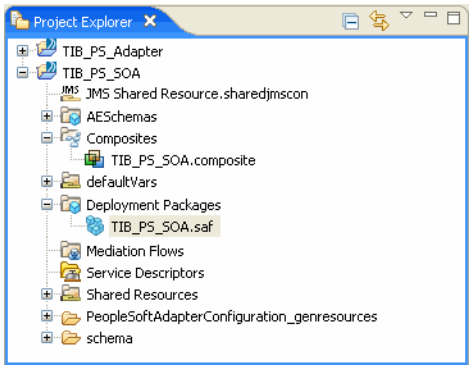
Creating a Service Assembly

To create a service assembly, follow these steps:

1. Select an ActiveMatrix SOA project in the Project Explorer panel and expand the **Composites** folder.
2. Right-click the *SOAProject*`.composite` file under the `Composites` folder and select **Service Assembly** from the pop-up menu.
3. Save the composite if prompted.

After completing the process, the deployment package, *SOAProject*`.saf`, is created in the `Deployment Packages` folder.

Figure 48 A Service Assembly



Refreshing a Created Service Assembly

To refresh a service assembly already created, follow these steps:

1. Select an ActiveMatrix SOA project in the Project Explorer panel and expand the **Deployment Packages** folder.
2. Right-click the service assembly file, *SOAProject*.`saf`, under the `Deployment Packages` folder, and select **Refresh Service Assembly** from the pop-up menu.

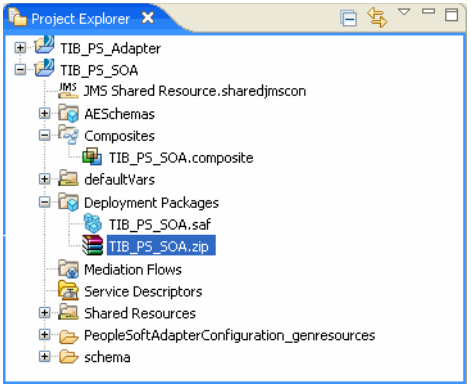
Creating a Service Assembly Archive

To create a service assembly archive, follow these steps:

1. Select an ActiveMatrix SOA project in the Project Explorer panel and expand the **Deployment Packages** folder.
2. Right-click the service assembly file, *SOAProject*.`saf`, under the `Deployment Packages` folder, and select **Build Archive** from the pop-up menu.
3. Save the service assembly file if prompted.

After completing the process, a ZIP file, *SOAProject*.`zip`, is created in the `Deployment Packages` folder, as shown in Figure 49.

Figure 49 A Service Assembly Archive



This service assembly archive can now be deployed and started using TIBCO ActiveMatrix Administrator. For detailed information, refer to Deploying a Service Assembly Archive on page 91.

For Adapter Projects Running in the Unwire Mode (Using EAR2SA Tool)

Adapter projects created using TIBCO Designer always run in the Unwired Mode. For detailed information about how to create an adapter project in TIBCO Designer, refer to *TIBCO ActiveMatrix Adapter for Siebel Configuration and Deployment*.

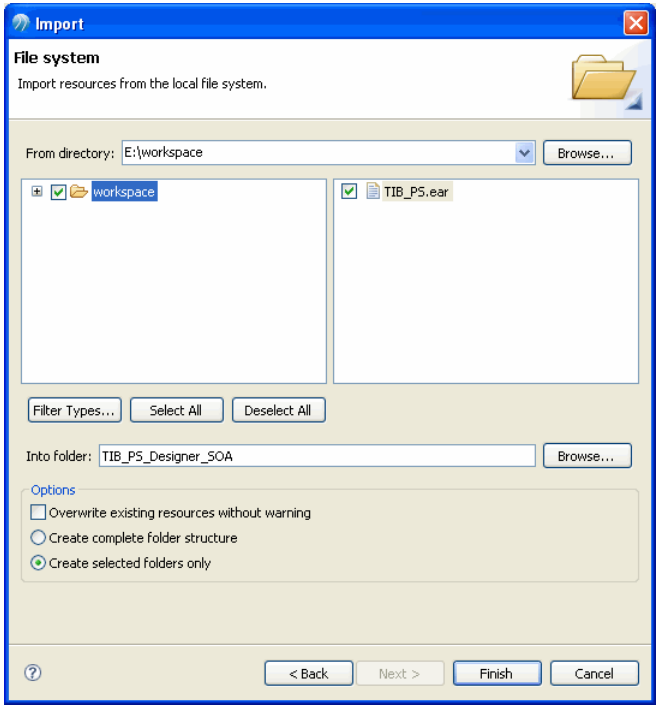
Importing an EAR File

An adapter EAR file created using TIBCO Designer for the adapter project must be imported into TIBCO Business Studio.

To import an EAR file, follow these steps:

1. Create an ActiveMatrix SOA project. Refer to Creating an ActiveMatrix SOA Project on page 66.
2. Select the ActiveMatrix SOA project in the Project Explorer panel and select **Import...** from the pop-up menu. The Import dialog appears.
3. Expand the **General** folder and double-click the **File System** folder. The File System page appears in the Import dialog, as shown in Figure 50.

Figure 50 Select an EAR File to Import



4. Click the **Browse** button next to the From directory field to navigate to the directory where the EAR file is located. Click the **Browse** button next to the Into folder field to select a folder where the EAR file is imported into.
5. Click the **Finish** button to close the dialog.

After completing the process, the EAR file is imported in the ActiveMatrix SOA project.

Creating a Service Assembly Archive

To create a service assembly archive, follow these steps:

1. Right-click the EAR file under the ActiveMatrix SOA project, and select **Build Archive** from the pop-up menu.
2. Save the EAR file if prompted.

After completing the process, a ZIP file, *EAR_File*.`zip`, is created in the `Deployment Packages` folder.

This service assembly archive can now be deployed and started using TIBCO ActiveMatrix Administrator. For detailed information, refer to Deploying a Service Assembly Archive on page 91.

Deploying a Service Assembly Archive

This section describes the steps necessary to configure and deploy a service assembly. These tasks are performed using TIBCO ActiveMatrix Administrator. Deploying a service assembly archive involves the following activities:

- Uploading a Service Assembly Archive, page 91
- Configuring the Service Assembly, page 91
- Deploying the Service Assembly, page 93
- Starting the Service Assembly, page 93

Before proceeding, ensure that the environment for TIBCO ActiveMatrix has been configured correctly. Refer to TIBCO ActiveMatrix documentation for more information about how to configure the environment.

Uploading a Service Assembly Archive

The following steps describe how to upload the service assembly archive:

1. Start and log in to TIBCO ActiveMatrix Administrator.
2. Select **Deploy to an Environment** from the Perspective drop-down list and **development** from the Environment drop-down list.
3. Click **Upload Service Assembly**.
4. Complete the following operations to upload your service assembly archive:
   a. Enter a name for your service assembly in the **Name** field,
   b. Click **Browse...** to select an archived ZIP file (the SA project) which was generated in TIBCO Business Studio.
   c. Select the **Import Shared Resource Definitions** check box to create the shared resources definitions from the definitions in the service assembly archive.
   d. Click the **OK** button.

The service assembly will appear in the Service Assemblies table with a status of `Not Deployed`.

Configuring the Service Assembly

To configure a service assembly, complete the following tasks:

Task A Configure the Service Unit

Specify the nodes on which the service units in the service assembly will be deployed.

1. In the Service Assemblies table, select the service assembly.
2. Click the **Service Units** button to view the service units in the selected service assembly.
3. For each service unit in the service assembly, complete the following operations:
   a. Select a service unit in the Service Units table.
   b. In the Node Mapping tab, click the **Edit** button, select the nodes that you want to deploy the service unit to in the Available Nodes list, and then click the right arrow button. The nodes will appear in the Mapped Nodes list.
   c. Click the **Save** button.

The service unit properties are displayed in the following tabs:

- **Node Mapping** — you can map the nodes to service units. See *TIBCO ActiveMatrix Administration* for details.
- **Substitution Variables** — you can specify the value of the service unit's substitution variables. See *TIBCO ActiveMatrix Administration* for details.

Task B Start Node and Install Shared Resources

If an ActiveMatrix Service (JMS or SOAP) is included in the SA project, you need to install the corresponding Shared Resource for the appropriate node.

1. Select **Configure an Environment** from the Perspective drop-down list and **development** from the Environment drop-down list.
2. Select the appropriate node in the Nodes table, and then click the **Start** button.
3. In the Node Detail panel, select **Shared Resources**.
4. Select the corresponding shared resource, and then click the **Edit** button.
5. In the Enabled at Startup? column, select the **Yes** radio button, and then click the **Save** button.
6. On the top of the Shared Resources table, click the **Install** button. The status of the resource changes to `Installed`.

Deploying the Service Assembly

A service assembly is ready to be deployed once the service assembly is uploaded and configured. Before proceeding with deploying the service assembly, verify the following:

- All nodes required for deployment are running.
- All containers required for deployment are activated.
- All shared resources are installed.

To deploy a service assembly:

1. In the Service Assemblies table, select the service assembly.
2. Click the **Deploy** button.

Once the service assembly is deployed, the status changes to `Deployed`.

Starting the Service Assembly

After a service assembly has been successfully deployed, you can start it, and if required, stop it.

1. In the Service Assemblies table, select one or more service assemblies.
2. To start the selected service assemblies, click the **Start** button. The status changes to `Running`.
3. To stop the selected service assemblies, click the **Stop** button. The status changes to `Deployed`.

Index

T