



Siebel RTD

Installation and Administration of Siebel RTD

Copyright

Copyright © 2005 Sigma Dynamics All Rights Reserved.

Restricted Rights Legend

This software and documentation is subject to and made available only pursuant to the terms of the Sigma Dynamics License Agreement and may be used or copied only in accordance with the terms of that agreement. It is against the law to copy the software except as specifically allowed in the agreement. This document may not, in whole or in part, be copied, photocopied, reproduced, translated, or reduced to any electronic medium or machine readable form without prior consent, in writing, from Sigma Dynamics.

Information in this document is subject to change without notice and does not represent a commitment on the part of Sigma Dynamics. THE SOFTWARE AND DOCUMENTATION ARE PROVIDED AS IS WITHOUT WARRANTY OF ANY KIND INCLUDING WITHOUT LIMITATION, ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. FURTHER, Sigma Dynamics DOES NOT WARRANT, GUARANTEE, OR MAKE ANY REPRESENTATIONS REGARDING THE USE, OR THE RESULTS OF THE USE, OF THE SOFTWARE OR WRITTEN MATERIAL IN TERMS OF CORRECTNESS, ACCURACY, RELIABILITY, OR OTHERWISE.

Installation and Administration of Siebel RTD

Section 1:	Preparing for Installation	2
1.1	Before you begin	2
1.2	Preparing your database server	2
1.2.1	Preparing Microsoft SQL Server.....	2
1.2.2	Preparing Oracle.....	2
1.2.3	Preparing DB2	3
Section 2:	Installing the Siebel Real-Time Decision Server	4
2.1	Installing with the JBoss Application Server	4
2.1.1	Running Siebel RTD Server as a Windows Service	6
2.1.2	Changing Java Virtual Machine Parameters	6
2.2	Installing Siebel RTD for IBM WebSphere Application Server	6
2.2.1	Installing and configuring WebSphere for Siebel RTD	7
2.2.2	Installing the Siebel Real-Time Decision Server for WebSphere	7
2.2.3	Configuring Server properties.....	7
2.2.4	Creating a JDBC Provider:	8
2.2.5	Installing Siebel RTD on the Websphere application server.....	11
2.2.6	Configuring Siebel RTD	11
2.2.7	Starting Siebel RTD.....	11
2.2.8	Configuring Secure Sockets Layer (SSL) for Siebel Real-Time Decision Server	11
2.2.9	Configuring WebSphere to use Siebel Object Manager Authentication	13
2.3	Installing and Configuring Siebel RTD for BEA WebLogic Application Server	14
2.3.1	Installing and configuring WebLogic for Siebel RTD	14
2.3.2	Installing the Siebel Real-Time Decision Server for WebLogic	14
2.3.3	Configuring Server properties.....	14
2.3.4	Configuring Secure Sockets Layer (SSL) for Siebel Real-Time Decision Server	17

2.3.5	Configuring WebLogic to use Siebel Object Manager Authentication	18
Section 3:	About Siebel RTD client tools	19
3.1	Overview of Siebel RTD client tools	19
3.2	Installing Siebel RTD client tools	19
Section 4:	Configuring Authentication for Siebel RTD	22
4.1	About Authentication	22
4.2	Managing security for Siebel RTD	22
4.2.1	About Changing Authentication	22
4.2.2	Configuring Windows Authentication.....	22
4.2.3	Configuring Siebel RTD Platform Authentication.....	23
4.2.4	Configuring Siebel Object Manager Authentication.....	23
4.2.5	Managing users and groups for Siebel RTD Platform Authentication	24
Section 5:	Configuring Data Access for Siebel RTD.....	26
5.1	Initializing the database and modifying the SDDS data source	26
5.1.1	Initializing the database	26
5.1.2	Populating the CrossSell example data.....	27
5.1.3	Modifying the SDDS datasource.....	27
5.2	Creating additional JDBC data sources for Siebel Real-Time Decision Server	28
5.2.1	Configuring JDBC data sources for JBoss	28
5.2.2	Configuring JDBC data sources for WebLogic.....	29
5.2.3	Configuring JDBC data sources for WebSphere	31
5.3	Configuring JDBC Data Sources to access Siebel Analytics data	34
5.3.1	Creating the Siebel Analytics JDBC data source for WebLogic	35
5.3.2	Creating the Siebel Analytics Data Source for JBoss.....	36
5.4	Configuring JDBC Data Sources to access Siebel OLTP	36
5.4.1	Creating the Siebel OLTP data source for WebLogic.....	36
5.4.2	Creating the Siebel OLTP data source for WebSphere.....	37
5.4.3	Creating the Siebel OLTP data source for JBoss	37

Section 6:	Additional Configuration Settings.....	39
6.1	Additional Configuration for Decision Center.....	39
6.1.1	Setting Internet Explorer options	39
6.2	Additional Configuration for Siebel Real-Time Decision Servers	39
6.2.1	About Logging Configuration	39
6.2.2	Changing log configuration parameters.....	39
6.2.3	Configuring Services for Servers	39
Section 7:	Production Deployment of Siebel RTD	40
7.1	All-In-One Configuration.....	40
7.2	Full Deployment Configuration	40
Section 8:	JMX Management of Siebel RTD.....	42
8.1	About JMX MBean operations and attributes.....	42
8.2	Siebel RTD cluster level management	42
8.2.1	About SDCluster	43
8.2.2	About SDClusterPropertyManager	44
8.2.3	About SDClusterPropertyManager→Misc	44
8.2.4	About SDClusterPropertyManager→ClusterManager	45
8.2.5	About SDClusterPropertyManager→Deployment.....	46
8.2.6	About SDClusterPropertyManager→Alert Service	47
8.3	Siebel RTD Member Management	47
8.3.1	About SDCluster→Member	47
8.3.2	About SDCluster→Members→Logger.....	47
8.3.3	About SDCluster→Members→Properties	48
8.3.4	About SDCluster→Members→ DecisionService	49
8.4	Siebel RTD security management	50
8.4.1	About SDCluster→Security.....	50
8.4.2	About SDCluster→Security→SecurityProperties	53
8.4.3	About SDCluster→Security→Authenticator	53

8.5	About Inline Service Manager	54
8.5.1	About SDCluster→InlineServiceManager→ [Inline Service].....	54
8.6	About Deployment States Management	55
8.6.1	About SDCluster→DeploymentStates	55
8.6.2	About Deployment State.....	55
8.7	About Learning Service management	56
8.7.1	About SDCluster→ (Server) → LearningService MBean	56
8.8	About Alert Service Management.....	57
8.8.1	About SDCluster→Alert Service	57

Preface

About this document

This document acts as a reference to administering Siebel RTD. It identifies the means for deploying Inline Services, clustering servers for performance and creating users and groups for maintaining security.




Intended Audience

This document is designed to act as a reference for Administrators of Siebel RTD. Users should have a working knowledge installing and administering enterprise level applications and the use of a JMX console.

How to use this guide

This document is divided into the following sections: **Section 1: Preparing for Installation** outlines tasks that must be performed before you begin; **Section 2: Installing the Siebel Real-Time Decision Server** gives step by step instructions to install and configure the Siebel Real-Time Decision Server on JBoss, WebLogic and WebSphere application servers; **Section 3: About Siebel RTD Client Tools** gives instruction on installing Siebel RTD client tools; **Section 4: Configuring Authentication** provides information on configuring and managing Windows, Platform or Siebel Object Manager authentication; **Section 5: Configuring Data Access for Siebel RTD** explains tools and procedures for setting up JDBC data sources; **Section 6: Additional Configuration Settings** gives additional configurations for Siebel RTD clients and servers; **Section 7: Production Deployment of Siebel RTD** discuss deployment scenarios for production environments; **Section 8: JMX Management of Siebel RTD** describes the JMX MBeans used to manage Siebel RTD.

Document conventions

Convention	Description
<code>monospace</code>	Indicates source code and program output.
bold	Indicates portions of the user interface, including labels, tabs, menus, etc.
<i>italic</i>	Italics indicate user specific values.
'quote'	Indicates input required from the user.
	Indicates additional information that may make the task easier.
	Indicates additional information about the subject.
	Indicates actions that may result in loss of data or errors.

Section 1: Preparing for Installation

Siebel RTD is available on both Windows and Unix platforms and runs with the JBoss, WebLogic and WebSphere application servers. To install Siebel RTD, you will:

1. Prepare your database server.
2. Install the Siebel Real-Time Decision Server.
3. Configure the server using your application server's administration console. This step is not required for the embedded JBoss application server.
4. Install the Siebel RTD client tools.
5. Configure the authentication settings.

1.1 Before you begin

If you are running virus protection software that uses script detection, you should disable it before installing Siebel RTD. Close all Windows applications before you begin.

Also, any real-time indexing software, such as Google™ Desktop, should be disabled before installation.

1.2 Preparing your database server

Install the database of your choice using the installation instructions included with the product. Use the following instructions to prepare the database for the Siebel RTD installation.

1.2.1 Preparing Microsoft SQL Server

After installing SQL Server, use the SQL Server Enterprise Manager to create a new database. By default the installation procedure assumes it will be named 'sd'. If you choose to name your database otherwise, you will need to adjust installation settings. Assign the users described below to this database.

Installation of Siebel RTD on SQL Server requires that you provide two database users: an administration user and a runtime user. The administration user should have the **Server Role membership** 'System Administrator'. The runtime user should have the **Database Role memberships** 'db_ddladmin', 'db_reader', and 'db_writer.'

For assistance in database and user creation, please refer to your Microsoft SQL Server documentation.

You must have SQL Server Client Tools installed on the machine from which you will install Siebel RTD.

1.2.2 Preparing Oracle

Installation of Siebel RTD on Oracle requires that you provide the net service name and a valid user name and password. For assistance setting up net services, please refer to your Oracle documentation.

Installation of Siebel RTD on Oracle requires that you provide two database users: an administration user and a runtime user. The administration user should have the **Database role membership** 'dba'. The runtime user should have the **System privilege** 'Create Session'. The default 'Connect' **System privilege** can be revoked.

The **Quota** for the runtime user's tablespace should be set to an appropriate level or to 'unlimited' depending on your database policies.

The Oracle database must have the **user_lock** package installed. Please refer to the Oracle documentation for instructions on installing optional Oracle packages.

You must have Oracle Administrator Client installed on the machine from which you will install Siebel RTD.

1.2.2.1 About tablespace mapping for Oracle

If you would like to use Oracle tablespaces for the Siebel RTD database, skip the database initialization portion of the Siebel Real-Time Decision Server installation. Edit the script located at `$INSTALLDIR\scripts\sql\Oracle\SDTablespaceMap.txt`. This file allows you to control the allocation of Siebel Real-Time Decision Server tables to your Oracle tablespaces. The entries in this file are key/value pairs. The SQL script that creates Siebel Real-Time Decision Server tables has occurrences of tokens that look like `${key}`.

If the key in the token is a key in this file, then a tablespace clause is inserted and the name of the tablespace is the value from the key/value pair. If the key doesn't match a key in this file, then no tablespace clause is inserted and the token is removed.

Uncomment the key value pair by deleting the '#', and initialize the database using SDDBTool. For usage of SDDBTool, see *Section 5: Configuring Data Access for Siebel RTD*.

1.2.3 Preparing DB2

After installing DB2 Server, use DB2 Control Center to create a new database. By default the installation procedure assumes it will be named 'sd'. If you choose to name your database otherwise, you will need to adjust installation settings. The code set of this database should be set to UTF-8.

Installation of Siebel RTD on DB2 requires that you provide two database users: an administration user and a runtime user. The administration user should have the **Authority** 'Database Administrator' on the database you created. The runtime user should have the **Authority** 'Connect to database'.

For assistance in user and database creation, please refer to your DB2 documentation.

You must have the DB2 Administrative Client installed on the machine from which you will install Siebel RTD.

Section 2: Installing the Siebel Real-Time Decision Server

The Siebel Real-Time Decision Server must be installed on the server where your chosen application server is resident. Installation of Siebel RTD on the WebSphere and WebLogic application servers requires configuration through their respective administration consoles after installation; the Siebel Real-Time Decision Server for JBoss is configured automatically through the installation process.

2.1 Installing with the JBoss Application Server

Siebel RTD runs on the JBoss Application Server, a full featured, standards-based J2EE application server.

- 1 With the database server started, run setup. **Welcome** page appears. Click **Next**.
- 2 **Choose Install Location** page appears. Enter a destination folder or accept the default destination. This is your \$INSTALLDIR. Click **Next**.
- 3 **Select Database Type** appears. Choose your database type: SQL Server, Oracle or DB2. Click **Next**.
- 4 **Database Settings** appears. Enter your database settings:

For SQL Server

Host
Database port
Database Name
Runtime User
Runtime User Password XXXX
Administrative User
Administrative User Password XXXX

For Oracle

Host
Database Port
SID
Runtime User
Runtime User Password XXXX
Administrative User
Administrative User Password XXXX

For DB2

Host
Database port
Database Name
Runtime User
Runtime User Password XXXX
Administrative User
Administrative User Password XXXX



Note: The Administrative User entered here must have rights to create tables and stored procedures on the database. The Runtime user is used to access system data at runtime.

Click **Next**.

- 5 **Initialize/Upgrade** appears. By default the installation will initialize or upgrade your database. If your database has not been prepared, or if you wish to initialize or upgrade later, uncheck the **Initialize/Upgrade** box. You must initialize or upgrade at a later time.
- 6 Choose an Authentication type, Windows Authentication, Siebel RTD Platform Authentication or Siebel Object Manager Authentication. Leave **Only apply security settings if they have not been configured before** checked unless you want to *explicitly override* security settings on an already installed server. Click **Next**.

- If you chose Windows Authentication, enter a domain controller in **Domain Name**. This is a Domain Controller on your network. Enter an **Administrator Group Name** for your Domain Controller. Click **Next**.

Note: The Administrator group name must be part of your Windows Authentication groups. See *4.2.2 Configuring Windows Authentication* below.

- If you chose Siebel RTD Platform Authentication, enter an **Administrator Group Name**. By default this is 'SDAdministrators'. Enter a **User Name** and **Password**. This is your default Siebel RTD Administrator user. Click **Next**.

- If you chose Siebel Object Manager Authentication, enter a:

Connect String: The connect string is a URL containing the information needed to connect to any Siebel Server component. It specifies both the protocol and the details of the Client Application Manager service in the Siebel Servers to which the client connects. The generic syntax for the connect string follows:

For Siebel Server 7.5.3:

```
siebel[.[transport][.[encryption][.[compression]]]:  
//host[:port]/EnterpriseServer/AppObjMgr[/SiebelServer]
```

For Siebel Server 7.7.1:

```
siebel[.[transport][.[encryption]  
[.[compression]]]://host[:port]/EnterpriseServer/  
AppObjMgr
```

For more information about connect strings, see the Siebel Bookshelf document, *Siebel Object Interfaces Reference*.

Login Name: A Siebel User Name. This user must have rights to traverse Siebel's database for user and responsibility definitions. This user should be assigned the Siebel Administrator responsibility.

Login Password: The password for the Siebel user identified by Login Name.

Siebel Administrative Responsibility: A Siebel Responsibility. Responsibilities determine which views users can access. All users that are assigned this responsibility in Siebel will be viewed as administrators in Siebel RTD.

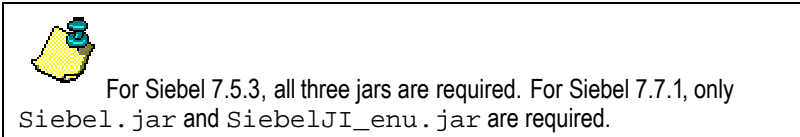
Click **Next**.

Security Setting – Classpaths appears. Browse to the location of the following jar files located in the Siebel Object Manager installation. The default location of these jar files is `\siebel_analytics_install_directory\siebsrvr\CLASSES\`

SiebelJI.jar

SiebelJI_common.jar

SiebelJI_enu.jar

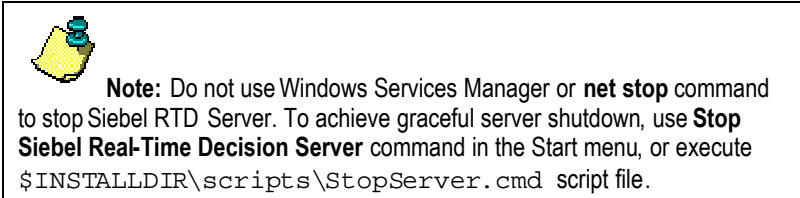


If the Siebel Analytics installation is on a different server from the Jboss installation, you must copy these jar files to a location on the Jboss server. The Classpath should then be updated to refer to the jar files on the Jboss server.

- 7 **Choose Start Menu Folder** appears. Choose a Start Menu or accept the default.
- 8 Click **Install**. The Siebel RTD installation begins.

2.1.1 Running Siebel RTD Server as a Windows Service

To run Siebel RTD Server as a Windows service use Windows Services Manager to change startup type for Siebel RTD Server service from **Manual** to **Automatic**. The service will start automatically every time the OS is restarted.



2.1.2 Changing Java Virtual Machine Parameters

The most common case of changing Java VM parameters is allowing the Siebel RTD Server to use more memory than the default 512MB. Maximum amount of memory is controlled by “-mx” Java VM parameter.

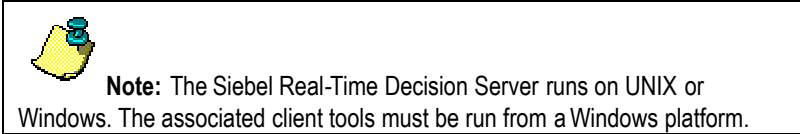
Parameters of Java virtual machine running Siebel RTD Server are specified in the following script files located in \$INSTALLDIR\scripts folder:

- StartSD.cmd Starts Siebel RTD Server in a console window.
- StartSDDebug.cmd Starts Siebel RTD Server in a console window in debug mode.
- InstallService.cmd Installs Siebel RTD Server as a Windows service.

To change the Java virtual machine parameters edit the corresponding script file. In case of InstallService.cmd, the script file has to be executed once to reinstall the service with the new Java VM parameters.

2.2 Installing Siebel RTD for IBM WebSphere Application Server

Siebel RTD is supported on both UNIX and Windows platforms for IBM's WebSphere application server. The following instructions install the Siebel Real-Time Decision Server.



2.2.1 Installing and configuring WebSphere for Siebel RTD

- 1 Install WebSphere according to the WebSphere documentation. When installing, you must be logged into the local machine as a member of the Administrator Group.
- 2 Update the WebSphere installation according to IBM's Recommended Updates, Version 5.1, found at <http://www-1.ibm.com/support/docview.wss?rs=180&uid=swg27004980#51>.
 - Upgrade to the 5.1.1 Fix Pack
 - Install the V5.1.1 Java SDK Service Release
 - Upgrade to the 5.1.1.5 Fix Pack
- 3 After installation and upgrades, update the parameter in the WebSphere properties file, `j2c.properties`, in the WebSphere directory `\websphere_install_directory\AppServer\properties`. Uncomment the property 'logMissingTranContext' and change value to 'false'. This prevents WebSphere from appending to the log file periodically as a result of running the Siebel Real-Time Decision Server.

```
<cm-properties>
    <manageCachedHandles>>false</manageCachedHandles>
    <logMissingTranContext>false</logMissingTranContext>
</cm-properties>
```

- 4 After installation, start the WebSphere server.

2.2.2 Installing the Siebel Real-Time Decision Server for WebSphere

- 1 **For Windows:** Use WinZip to extract the archive file 'sd.zip'.
For Unix: Use gunzip or tar to extract the archive file 'sd.tar.gz'. Note: For Solaris, right-click the archive file and use **Uncompress file** before using tar to extract.
- 2 When you extract the files, a directory called `\SiebelAnalytics\RTD` will contain all of the Siebel Real-Time Decision Server files under the directory you extracted into. This directory is your `$INSTALLDIR`.

2.2.3 Configuring Server properties

Configuration is accomplished through the WebSphere administration console. For more information about the use of the console, please refer to your WebSphere documentation.

- 1 Open a browser and enter the url '`http://websphere_host_machine_name:port/admin`'. The default port is 9090. At the login prompt, enter the default administrator username, 'admin', or the appropriate administrator username and password.
- 2 **Setting up host aliases:**
 - In the tree on the left, expand **Environment** and select **Virtual Hosts**.
 - On the right select **default_host**.
 - Under **Additional Properties**, select **Host Aliases**.

- Select **New**: Use the following values: **Host Name**: '*', **Port**: '8080'. Click **OK**.

3 Setting up the Server properties:

- In the tree on the left, expand **Servers** and select **Application Servers**. On the right select **server1**. Under **Additional Properties** select **Web Container**.
- Select **Http transports** and click **New**. Use the following values: **Host**: '*', **Port**: '8080'. Click **OK**.
- In the tree on the left, expand **Servers** and select **Application Servers**. On the right select **server1**. Select **Process Definition**. Select **Java Virtual Machine**.
- **FOR WINDOWS**: Use the following value:
Generic JVM arguments: '-Xj9 -Xverify:none -Djava.awt.headless=true'
- **FOR UNIX**: Use the following value:
Generic JVM arguments: '-Djava.awt.headless=true'
- Select **Custom Properties**. Select **New**.
Use the following values:

Name: org.eclipse.emf.ecore.ERPackage.Registry.INSTANCE
Value: com.sigmadynamics.emf.util.SDEMFRegistry.

Click **OK**.

4 Setting up the Siebel RTD Install location:

- In the tree on the left, expand **Environment** and select **Manage WebSphere Variables**. Click **New**. Use the following values:

Name: 'SD_HOME'

Value: the directory where Siebel Real-Time Decision Server was installed referred to as \$INSTALLDIR above.

Click **OK**.

2.2.4 **Creating a JDBC Provider:**

1 In the tree on the left expand **Resources** and select **JDBC Providers**. Click **New**.

For SQL Server:

- In **JDBC Providers**, select 'User-defined JDBC Provider'. Click **OK**. Use the following values:

Name: 'SD Data Provider'

Classpath: '\${SD_HOME}\lib\jtds.jar'

Implementation Classname: 'net.sourceforge.jtds.jdbcx.JtdsDataSource'

Click **OK**.

- Select **SD Data Provider**. Select **Data Sources** under **Additional Properties**. Select **New**. In name type 'SDDS'. Press return.

- Select on **J2C Authentication Data Entries**. Select **New**. Use the following values:

Alias 'SDDS-auth'

User ID: your database login

Password: your database password. You may not enter a username with an empty password.

- Click **OK**.
- At the top of the page, click on **Save**. Click the **Save** button to save change to the master configuration.
- Expand **Resources** and select **JDBC Providers**. Select **SD Data Provider**.
- In **Additional Properties** select **Data Sources**. Choose **SDDS**.
- For **Container-managed Authentication Alias**, select 'SDDS-auth'.
For **Component-managed Authentication Alias**, select 'SDDS-auth'.
- Click on **Custom Properties**. Create the following properties:

Name: 'serverName'; **Value:** host name of the database server

Name: 'portNumber'; **Value:** port of the database server

Name: 'databaseName'; **Value:** Name of the database.

- At the top of the page, click on **Save**. Click on the **Save** button.

For Oracle:

- For **JDBC Providers**, select **Oracle JDBC Driver**. Click **OK**. Use the following values:

Name: 'SD Data Provider'

Classpath: '\${SD_HOME}\lib\ojdbc14.jar'.

Click **OK**.

- Select 'SD Data Provider'. Expand **Additional Properties** and select **Data Sources**. Click **New**. Enter 'SDDS' for **Name**. Press ENTER.
- Click on **J2C Authentication Data Entries**. Click **New**. Use the following values:

Alias 'SDDS-auth'

User ID: your database login

Password: your database password.

Click **OK**.

- At the top of the page, click on **Save**. Click on the **Save** button.
- Expand **Resources** and select **JDBC Providers**. Select 'SD Data Provider'.
- In **Additional Properties** select **Data Sources**. Choose 'SDDS'.
- For **Container-managed Authentication Alias**, select 'SDDS-auth'. Click on **Custom Properties**.
- Change the following properties:

URL - jdbc:oracle:thin:@<host name of the Oracle server>:<port of Oracle server:<SID> ,
(the default port is 1521 for Oracle)

driverType - thin

portNumber - 1521

preTestSQLString - SELECT 1

- At the top of the page, click on **Save**. Click on the **Save** button.

For DB2:

- For **JDBC Providers**, select **DB2 Universal JDBC Driver-compliant Provider**. Click **OK**. Use the following values:

Name: 'SD Data Provider'

Classpath: ' \${SD_HOME}\lib\db2jcc.jar' and '\${SD_HOME}\lib\db2jcc_license_cu.jar'.

Click **OK**.

- Select SD Data Provider. Expand **Additional Properties** and select **Data Sources**. Click **New**. Use SDDS for **Name**. Press ENTER.
- Click on **J2C Authentication Data Entries**. Click **New**. Use the following values:

Alias 'SDDS-auth'

User ID: your database login

Password: your database password.

Click **OK**.

- At the top of the page, click on **Save**. Click on the **Save** button.
- Expand **Resources** and select **JDBC Providers**. Select 'SD Data Provider'.
- In **Additional Properties** select **Data Sources**. Choose 'SDDS'.
- For **Container-managed Authentication Alias**, select 'SDDS-auth'.
For **Component-managed Authentication Alias**, select 'SDDS-auth'.

- Click on **Custom Properties**.
- Change the following properties:
 - databaseName** – 'SD' or the alternate name you provided for your database.
 - driverType** – the type of DB2 driver, in this case '4'.
 - serverName** – the name of your DB2 server.
 - portNumber** – the port number of the DB2 server. (This is 50000 by default for DB2)

At the top of the page, click on **Save**. Click on the **Save** button.

2.2.5 Installing Siebel RTD on the Websphere application server

- 1 In the tree on the left, expand **Applications** and click on **Install New Application**.
- 2 Click **Browse** and select 'sd.ear' from 'SD_HOME\package'. (located in \$INSTALLDIR\package\) Click on **Next**. Click on **Next**. Click on **Next**.
- 3 For **existing Resource JNDI name**, select 'jdbc/SDDS' from the drop down. Make sure all of the Modules are selected. Click on **Apply**. Click on **Next**. Click on **Next**. Click on **Next**. Click on **Finish**.
- 4 Click on **Save to Master Configuration**.
- 5 Click on **Save** button.

2.2.6 Configuring Siebel RTD

- 1 In the tree on the left, expand **Applications** and click on **Enterprise Applications**.
- 2 Click on **SD**.
 - Select 'PARENT_LAST' for **Classloader Mode**.
 - Select 'Application' for **WAR Classloader Policy**.

Click on **Apply**. Click on **OK**.
- 3 Click on **Save** link at top of the page. Click on **Save** button.
- 4 Restart the WebSphere server.

2.2.7 Starting Siebel RTD

- 1 In the tree on the left, expand **Applications** and click on **Enterprise Applications**.
- 2 Check **SD**. Click on **Start**.

2.2.8 Configuring Secure Sockets Layer (SSL) for Siebel Real-Time Decision Server

OPTIONAL

To enable secure connections from all Siebel RTD clients to Siebel Real-Time Decision Server, you must configure SSL settings through the administration console.

- 1 Open a browser and enter the url 'http://< webspHERE_host_machinename>:9090/admin'. At the login prompt, enter the default administrator username, 'admin', or the appropriate administrator username and password for WebSphere.
- 2 In the tree on the left expand **Security** and choose **SSL and DefaultSSLSettings**. Modify the following properties: Select Security Center.

Key file name: '\$INSTALLDIR\etc\ssl\sdserver.keystore', where \$INSTALLDIR is the location you installed Siebel Real-Time Decision Server.

Key file password: 'tc-ssl'

Confirm password: 'tc-ssl'

Key file format: JKS

Trust file name: '\$INSTALLDIR\etc\ssl\sdtrust.store', where \$INSTALLDIR is the location you installed Siebel Real-Time Decision Server.

Trust file password: 'tc-ssl'

Confirm Password: 'tc-ssl'

Security Level: HIGH

Click the **Apply** button.

- 3 In the tree on the left, expand **Application Servers** and choose your server. On the right choose Under **Additional Properties**, choose **Web container**. Under **Additional Properties** choose **HTTP transports**.

- 4 Use **New** to add a new port:

Host: '*'

Port: '8443'

SSL enabled: select

Click the **Apply** button.

- 5 In the tree on the left, expand **Environment** and choose **Virtual Hosts**. On the right select **default host**. Under **Additional Properties** select **Host Aliases**.

- 6 Use **New** to add the port to the default virtual host:

Host: '*'

Port: '8443'

Click the **Apply** button.

- 7 On the tool bar, click **Save**. Use the **Save** button to save your changes to the master configuration.

- 8 Restart the WebSphere server. SSL is configured.

2.2.9 Configuring WebSphere to use Siebel Object Manager Authentication

OPTIONAL

If you are planning on using Siebel Object Manager for authentication on Siebel RTD, perform this additional configuration. You must also configure Siebel Object Manager Authentication during the client tools installation, as outlined below in *Section 3: Installing Siebel RTD client tools*.

- 1 Open a browser and enter the url 'http://websphere_host_machine_name:9090/admin'. At the login prompt, enter the default administrator username, 'admin', or the appropriate administrator username and password for WebSphere.
- 2 In the tree on the left, expand **Environment** and select **Shared Libraries**. On the right, select **New**. Use the following values:

Name: 'Siebel Lib',

Classpath:

`%siebel_analytics_install_directory%\siebsrvr\CLASSES\SiebelJI.jar`

`%siebel_analytics_install_directory%\siebsrvr\CLASSES\SiebelJI_common.jar`

`%siebel_analytics_install_directory%\siebsrvr\CLASSES\SiebelJI_enu.jar`



Each classpath component must be entered on a separate line. For Siebel 7.5.3, three jars are required, so the text entered here will appear on three lines.

For Siebel 7.7.1, only `Siebel.jar` and `SiebelJI_enu.jar` are required.

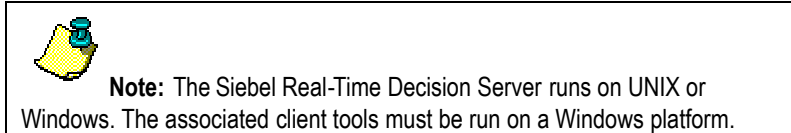
If the Siebel Analytics installation is on a different server from the WebSphere installation, you must copy these jar files to a location on the WebSphere server. The Classpath value should then be updated to refer to the jar files on the WebSphere server.

Click **OK**.

- 3 In the tree on the left, expand **Applications**, select **Enterprise Applications**, and then select **SD**.
- 4 At the bottom of the screen, click **Additional Properties**, then **Libraries**. Click the **Add** button.
- 5 Select **'Siebel Lib'** from the dropdown box. Click **OK**.
- 6 Click **Save** at the top of the screen, then click the **Save** button.
- 7 You must also configure Siebel Object Manager Authentication during the client tools installation, as outlined below in *Section 3: About Siebel RTD client tools*.

2.3 Installing and Configuring Siebel RTD for BEA WebLogic Application Server

Siebel RTD is supported on both UNIX and Windows platforms for the BEA WebLogic application server. The following instructions install and configure the Siebel Real-Time Decision Server for WebLogic.



2.3.1 Installing and configuring WebLogic for Siebel RTD

- 1 Install WebLogic according to the WebLogic documentation. After installation, start the WebLogic server.

2.3.2 Installing the Siebel Real-Time Decision Server for WebLogic

- 1 **For Windows:** Use WinZip to extract the archive file 'sd.zip'.
For Unix: Use gunzip or tar to extract the archive file 'sd.tar.gz'. Note: For Solaris, right-click the archive file and use **Uncompress file** before using tar to extract.
- 2 When you extract the files, a directory called `\SiebelAnalytics\RTD` will contain all of the Siebel Real-Time Decision Server files under the directory you extracted into. This directory is your `$INSTALLDIR`.
- 3 The application's ear file must be exploded before it can be deployed on WebLogic. Use Winzip to extract the `sd.ear` file found under `\SiebelAnalytics\RTD\package`.
Open the archive and create a directory 'exploded' under package to extract the ear file contents.

Your directory structure should look like
`$INSTALLDIR\SiebelAnalytics\RTD\package\exploded`.

2.3.3 Configuring Server properties

Configuration is achieved through the WebLogic administration console. For more information about the use of the console, please refer to your WebLogic documentation.

- 1 Using **Run**→**BEA WebLogic Platform**→**Quick Start**, create a new domain for your Siebel Real-Time Decision Server. The domain configuration template should be 'Basic WebLogic Server Domain'. Use the Custom options on creating the domain to change the listening port to 8080. For more information on creating domains, please refer to your WebLogic documentation.
- 2 Edit the StartWebLogic script, located by default at
`\bea\user_projects\domains\mydomain\startWebLogic.cmd`,
where *mydomain* is the name of your domain.

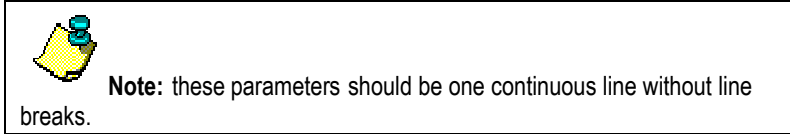
Edit the java startup command to include an additional argument '-Djava.awt.headless=true':

```
%JAVA_HOME%\bin\java %JAVA_VM% %MEM_ARGS% %JAVA_OPTIONS% -  
Dweblogic.Name=%SERVER_NAME% -  
Dweblogic.ProductionModeEnabled=%PRODUCTION_MODE% -
```

```
Djava.security.policy="%WL_HOME%\server\lib\weblogic.policy" -  
Djava.awt.headless=true weblogic.Server
```

- 3 Add the Classpath token to the Classpath parameter, making sure that it precedes the WebLogic classpath:

```
set CLASSPATH=%CLASSPATH%;%WEBLOGIC_CLASSPATH%;%POINTBASE_CLASSPATH%;  
%JAVA_HOME%\jre\lib\rt.jar;%WL_HOME%\server\lib\webservices.jar;
```



- 4 Using the **Advanced** tab on **System Properties**, available by right-clicking **My Computer**, add the following path to your classpath environment variable at the beginning of the string.

For Oracle:

```
$INSTALLDIR\lib\ojdbc14.jar;
```

For SQLServer:

```
$INSTALLDIR\lib\jtds.jar
```

For DB2:

```
$INSTALLDIR\lib\db2jcc.jar;
```

```
$INSTALLDIR\lib\db2jcc_license_cu.jar;
```

- 5 Start the WebLogic administration server for your domain using the Start menu. By default, a 'Start Server' item is added to the BEA Start Menu. For more information on starting the administration server, please refer to your WebLogic documentation.
- 6 Open a browser and enter the url 'http://weblogic_host_machine_name:8080/console'. At the login prompt, enter the administrator username and password.
- 7 In the tree on the left, expand **Services** and **JDBC**. Select **Connection Pool** and, from the right-click menu, **Configure a New JDBC Connection Pool**. Choose the following:

For SQL Server: **Database Type:** 'MS SQL Server' **Database Driver:** 'Other';

For Oracle: **Database Type:** 'Oracle' **Database Driver:** 'Other';

For DB2: **Database Type:** 'DB2' **Database Driver:** 'Other'.

Choose **Continue**. **Define connection properties** appears.

Enter the following properties:

Name: 'SDConnectionPool'

Database Name: The name of the database to connect to.

Host Name: The name or IP address of the database server.

Port: The port on the database server used to connect to the database.

Database User Name: The database account user name used in the physical database connection.

Password and Confirm Password: The database account password used in the physical database connection.

Choose **Continue**. Configure the following parameters:

Driver Classname: The full package name of the JDBC driver class used to create the physical database connections in the connection pool. (Note that this driver class must be in the classpath of any server to which it is deployed.)

MS SQL Server: net.sourceforge.jtds.jdbc.Driver
Oracle: oracle.jdbc.driver.OracleDriver
DB2: com.ibm.db2.jcc.DB2Driver

URL: The URL of the database to connect to. The format of the URL varies by JDBC driver.

MS SQL Server: jdbc:jtds:sqlserver://\${DB_SERVER}:\${DB_PORT}/\${DB_NAME}
Oracle: jdbc:oracle:thin:@\${DB_SERVER}:\${DB_PORT}:\${DB_NAME/SID}
DB2: jdbc:db2://\${DB_SERVER}:\${DB_PORT}/\${DB_NAME}

Properties: The list of properties passed to the JDBC drivers that are used to create physical database connections. Certain properties may have been automatically generated from the previous information.

MS SQL Server: user=database_username serverName=server_name
Oracle: user=database_username
DB2: None

- 8 If necessary, add the driver .jar file(s) to your system CLASSPATH environment variable. The default location for these files is: \$INSTALLDIR\lib

MS SQL Server: jtds.jar
Oracle: ojdbc14.jar
DB2: db2jcc.jar, db2jcc_license.jar

- 9 In the Administration Console, test the connection using **Test Connection** button. You should receive a message confirming the test. If not, follow the instructions given on the console to correct the connection settings. When complete, click **Create and Deploy** to deploy the connection pool.
- 10 *For Oracle only*, select the created **SDConnectionPool** from the tree on the left under **Services**→**JDBC**→**Connection Pools**.

Select the **Connection** tab and use the **Show** link to show **Advanced Options**. Uncheck **Remove Infected Connections Enabled**. Click **Apply**.

- 11 In the left navigation tree, expand **JDBC** under **Services**. Select **JDBC Data Sources** and use the link on the right pane to **Configure a new JDBC Data Source**. Enter the following:

Name: SDDS

JNDI Name: SDDS

Click **Continue**. **Connect to Connection Pool** appears.

- 12 Select your connection pool from **Pool Name**. Click **Continue**. **Target data Source** appears.
- 13 Select the server you created when you created your domain as a target. Click **Create**. The data source is created.
- 14 In the tree on the left, expand your domain. Under **Deployments** choose **Applications**. In the right pane, choose **Deploy a new Application**. **Deploy an Application** appears.
- 15 Use the hyperlink to **localhost** to browse to the location of your exploded archive. When you click **localhost**, you will be presented with a list of local drives. Select the one where the exploded archive is and navigate to
\$INSTALLDIR\SiebelAnalytics\RTD\package\exploded.
- 16 Click **Continue**. The application will be deployed.

2.3.4 Configuring Secure Sockets Layer (SSL) for Siebel Real-Time Decision Server

OPTIONAL

To enable secure connections from all Siebel RTD clients to Siebel Real-Time Decision Server, you must configure SSL settings through the administration console.

- 1 Open a browser and enter the url 'http://< weblogic_host_machinename>:7001/console'. At the login prompt, enter the administrator username and password for WebLogic.
- 2 In the tree on the left, expand **Servers** and select *myserver* where *myserver* is the name of the server you created on your domain. Select the **General** tab. Use the following values:
 - SSL Listen Port Enabled:** select
 - SSL Listen Port:** 8443
- 3 Select the **Keystores and SSL** tab. Click **Change** in the upper right. **Specify Keystore Type** appears. Choose **Custom Identity and Custom Trust**. Click **Continue**.
- 4 In **Custom Identity Key Store File Name**, enter the fully qualified path to the keystore file, '\$INSTALLDIR\etc\ssl\sdserver.keystore', where \$INSTALLDIR is the location you installed Siebel Real-Time Decision Server.
- 5 In **Custom Identity Key Store Type** enter 'JKS'.
- 6 In **Custom Identity Key Store Pass Phrase** and **Confirm Custom Identity Key Store Pass Phrase** enter 'tc-ssl'.
- 7 In **Custom Trust Key Store File Name**, enter the fully qualified path to the trust store file, '\$INSTALLDIR\etc\ssl\sdtrust.store', where \$INSTALLDIR is the location you installed Siebel Real-Time Decision Server.
- 8 In **Custom Trust Key Store Type** enter 'JKS'.

- 9 In **Custom Trust Key Store Pass Phrase** and **Confirm Custom Trust Key Store Pass Phrase** enter 'tc-ssl'. Click **Continue**. **Review SSL Private Key Settings** appears.
- 10 In **Private Key Alias** enter 'sdserver'.
- 11 In **Pass Phrase** and **Confirm Pass Phrase** enter 'tc-ssl'. Click **Continue**. **Notice: You Need to Restart Your Server** appears. Click **Finish**.
- 12 Restart your WebLogic server. SSL is configured.

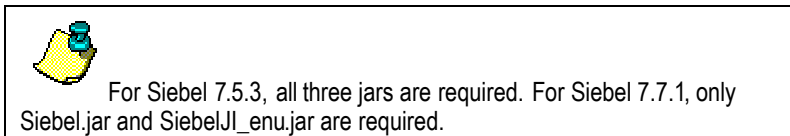
2.3.5 Configuring WebLogic to use Siebel Object Manager Authentication

OPTIONAL

- 1 If you are planning on using Siebel Object Manager for authentication on Siebel RTD, perform this additional configuration step. Add the following values to your system environment variable CLASSPATH:

```
siebel_analytics_install_directory\siebsrv\CLASSES\SiebelJI.jar;  
siebel_analytics_install_directory\siebsrv\CLASSES\SiebelJI_common.jar;  
siebel_analytics_install_directory\siebsrv\CLASSES\SiebelJI_enu.jar
```

Where *siebel_analytics_install_directory* is the location of your Siebel Analytics installation.



If the Siebel Analytics installation is on a different server from the WebLogic installation, you must copy these jar files to a location on the WebLogic server. The system environment variable CLASSPATH should then be updated to refer to the jar files on the WebLogic server.

- 2 You must also configure Siebel Object Manager Authentication during the client tools installation, as outlined below in *4.2.4 Configuring Siebel Object Manager Authentication*.

Section 3: About Siebel RTD client tools

The installation of client tools includes:

- Siebel Decision Studio
- Siebel Load Generator
- Administration (JMX)
- An embedded JBoss application server

3.1 Overview of Siebel RTD client tools

If you are planning on using Siebel RTD client tools on the same host where the JBoss version of Siebel Real-Time Decision Server was installed, no additional client installation is needed; the client tools installation and database initialization were completed when you installed the server.

If you are installing client tools for the WebSphere or WebLogic application server, you must install from a Windows operating system. In addition to installing client tools, the installation provides database initialization and authentication settings for the WebSphere or WebLogic platform.



The WebSphere and WebLogic versions of the Siebel Real-Time Decision Server must be started from within their administration consoles. After installation of the JBoss application server client tools, the Start menu item, **Siebel Analytics**→**RTD**→**Siebel Real-Time Decision Server (JBoss)**, will start the JBoss server.

You can use this application server locally for testing, however, *do not run* the JBoss server and the WebSphere server at the same time on the same machine.

3.2 Installing Siebel RTD client tools

- 1 With the database server started, run setup. **Welcome** page appears. Click **Next**.
- 2 **Choose Install Location** page appears. Enter a destination folder and click **Next**.



Note: If you are installing client tools for a WebSphere or WebLogic server installation, make sure the path you enter is different than the location in which you installed the Siebel Real-Time Decision Server for WebSphere or WebLogic.

We recommend appending JBoss to the directory, to clearly indicate which folders contain the JBoss installation.

For example: C:\SiebelAnalytics\RTD_JBoss\

- 3 **Select Database Type** appears. Choose your database type: SQL Server, Oracle, or DB2. Click **Next**.
- 4 **Database Settings** appears. Enter your database settings:

For SQL Server


Host
Database port
Database Name
Runtime User
Runtime User Password XXXX
Administrative User
Administrative User Password XXXX

For Oracle

Host
Database Port
SID
Runtime User
Runtime User Password XXXX
Administrative User
Administrative User Password XXXX

For DB2

Host
Database port
Database Name
Runtime User
Runtime User Password XXXX
Administrative User
Administrative User Password XXXX




Note: The Administrative User entered here must have rights to create tables and stored procedures on the database. The Runtime user is used to access system data at runtime.

Click **Next**.

- 5 **Initialize/Upgrade** appears. By default the installation will initialize or upgrade your database. If your database has not been prepared, or if you wish to initialize or upgrade later, uncheck the **Initialize/Upgrade** box. You must initialize or upgrade at a later time.
- 6 Choose an Authentication type, Windows Authentication, Siebel RTD Platform Authentication, Siebel Object Manager Authentication, or No Authentication. Leave **Only apply security settings if they have not been configured before** unless you want to explicitly override security setting on an already installed server.

Click **Next**.



Note: If your Siebel Real-Time Decision Server is running on a UNIX platform, you should choose Siebel RTD Platform Authentication, Siebel Object Manager Authentication, or No Authentication.

- 7 If you chose Windows Authentication, enter a domain controller in **Domain Name**. This is a Domain Controller on your network. Enter an **Administrator Group Name** for your Domain Controller.

Note: The Administrator group name must be part of your Windows Authentication groups or users. See *Configuring Windows Authentication* below.

- 8 If you chose Siebel RTD Platform Authentication, enter an **Administrator Group Name**. By default this is SDAAdministrators. Enter a **User Name** and **Password**. This is your default Siebel RTD Administrator user. Click **Next**.

- 9 **Choose Start Menu Folder** appears. Choose a Start Menu or accept the default.
- 10 Click **Install**. The Siebel RTD installation begins. A status bar displays as the process completes.

Section 4: Configuring Authentication for Siebel RTD

Three methods of authentication are available for Siebel RTD: Windows Authentication, Siebel RTD Authentication and Siebel Object Manager Authentication.

4.1 About Authentication

Siebel RTD uses Windows Authentication, Siebel RTD Platform Authentication, or Siebel Object Manager Authentication. Authentication may also be disabled. A method of authentication is chosen during the installation process. During the installation process you also chose a default administrative username and password. For Windows authentication you also chose a domain controller and domain username.

The following section describes steps needed to enable these types of authentication as well as how to change Siebel Real-Time Decision Server authentication type.

4.2 Managing security for Siebel RTD

Siebel RTD has authentication on both the server (cluster) and Inline Service levels. You have the option of using Windows Authentication, Siebel RTD Platform Authentication, or Siebel Object Manager Authentication.

4.2.1 About Changing Authentication

Siebel RTD is installed with Siebel RTD Platform Authentication, Windows Authentication, or Siebel Object Manager Authentication. Authentication changes are made using the JMX Console. For more about JMX see *Section 8: JMX Management of Siebel RTD*.

To change the Authentication type, use JMX to do the following.

- 1 With the Siebel Real-Time Decision Server running, open the JMX Console.
- 2 Navigate to the **SecurityProperties** MBean using **SDCluster**→**Security**→**SecurityProperties**.
- 3 Ensure that **AuthenticationEnabled** is set to true.
- 4 Change **AuthenticationProviderClass** to:
 - com.sigmadynamics.server.security.DBAuthenticator for Siebel RTD Platform Authentication
 - com.sigmadynamics.server.security.WindowsAuthenticator for Windows Authentication
 - com.sigmadynamics.server.security.SiebelAuthenticator for Siebel Object Manager Authentication
- 5 Click **Apply changes**.

The type of authentication you choose must be configured using the instructions below.

4.2.2 Configuring Windows Authentication

If you chose to use Windows authentication during the install process you choose a:

- Domain controller and a domain user

or

- Domain controller and domain group

If you chose a user or group and a domain controller, your system administrator must add the Administrators group that you specified in the installation (SDAdministrators, by default) to the domain. All members of this Administrator group are able to use Siebel RTD.

Two group for Decision Center users must also be added to the domain: 'SDDDecisionCenterUsers' and 'SDDDecisionCenterEditors'. These groups are used to define permissions on perspectives for Decision Center. See *Siebel Decision Studio Reference Guide* for information on defining permissions on perspectives.

To add permissions to additional Windows domain groups, use the JMX console to access **SDCluster→Security→assignPermission()**. For more about JMX see *Section 6: JMX Management of Siebel RTD*.

4.2.3 Configuring Siebel RTD Platform Authentication

If you selected Siebel RTD Platform Authentication, the username and group you specified during installation has been added to the repository of users and groups. This user is your default administrator, and the group specified is the administration group.

Two group for Decision Center users must also be added to the platform repository: 'SDDDecisionCenterUsers' and 'SDDDecisionCenterEditors'. These groups are used to define permissions on perspectives for Decision Center. See *Siebel Decision Studio Reference Guide* for information on defining permissions on perspectives.

To add additional Groups or Users use the JMX Console to access **SDCluster→Security→addGroup()** or **SDCluster→Security→addUser()** as described later in this document. Use **SDCluster→Security→assignPermission()** to assign cluster-level permissions to the users or groups.

4.2.4 Configuring Siebel Object Manager Authentication

If you selected Siebel Object Manager Authentication, the Siebel Object Manager responsibility you specified during installation has been added to the repository of users and groups. Every user having the responsibility within Siebel Object Manager has your default administrative privileges within Siebel RTD.

Two responsibilities for Decision Center users must also be added to Siebel Object Manager: 'SDDDecisionCenterUsers' and 'SDDDecisionCenterEditors'. These responsibilities are used to define permissions on perspectives for Decision Center. See *Siebel Decision Studio Reference Guide* for information on defining permissions on perspectives.

To give additional Siebel Responsibilities privileges, use the JMX Console to access **SDCluster→Security→assignPermission()** to assign cluster-level permissions to the responsibilities. Details about assigning permissions can be found later in this section.

Configuring WebLogic and WebSphere access to Siebel Object Manager jar files

The WebLogic and WebSphere application servers must be configured with the location of certain Siebel Object Manager jar files in order to use Siebel Object Manager Authentication.

See *2.2.8 Configuring WebSphere to use Siebel Object Manager Authentication* or *2.3.4 Configuring WebLogic to use Siebel Object Manager Authentication* for instructions.

Configuring JBoss access to Siebel Object Manager jar files

JBoss is configured automatically by the installer. If you change to Siebel Object Manager Authentication after installation, you must edit the script `SetSDParams.cmd`, located in `$INSTALLDIR\scripts` folder.

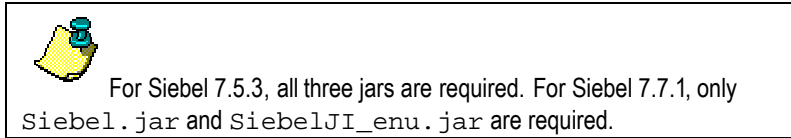
Add the Siebel Object Manager jar files to the line that sets the `SD_SERVER_CLASSPATH` environment variable:

```
set SD_SERVER_CLASSPATH=%SD_SERVER_CLASSPATH% ;
```

The fully qualified path to the following files should be added to the end of this line, preceding the semi-colon:

```
SiebelJI.jar  
SiebelJI_common.jar  
SiebelJI_enu.jar
```

The default location of these files is `\siebel_analytics_install_directory\siebsrvr\CLASSES`. Your files may be in a different location; check with you administrator if you do not know the location of these files.



If the Siebel Analytics installation is on a different server from the Jboss installation, you must copy these jar files to a location on the Jboss server. The `SD_SERVER_CLASSPATH` variable in the script `SetSDParams.cmd` should then be updated to refer to the jar files on the Jboss server.

After making the change, you must restart the Siebel RTD Server. If you are running Siebel RTD Server as a Windows service, you must execute `$INSTALLDIR\scripts\InstallService.cmd` script file to reinstall the service.

4.2.5 Managing users and groups for Siebel RTD Platform Authentication

Siebel RTD Platform Authentication users and groups are added using the JMX Console. For more about JMX see *Section 8: JMX Management of Siebel RTD*.

To manage users or groups for Siebel RTD Platform Authentication do the following:

- 1 With the Server running, open the JMX Console.
- 2 Use the **SDCluster->Security->Principals** and the following operations:
 - Use `createUser()` to create a new user. Note, this is only for Siebel RTD Platform Authentication. Windows Authentication uses the Windows domain usernames and Siebel Object Manager Authentication uses Siebel usernames.
 - Use `createGroup()` to create a new security group; alternatively use the Administration security group you created during the installation process. Windows uses Groups created on the Windows domain.
 - Use `addUserToGroup()` to make a user a direct member of a group.
- 3 Using the **Security->Permissions** Mbean, add permissions to the User and or Group.

Use `listPermissionCodes()` to list possible permissions. Permissions are:

Code	Permission
0	Administrator
1	Open Service
2	Open Service for Reading

3	Deploy Service from Studio
5	Download Service
6	Execute Integration Points from Load Generator
7	Execute Integration Points from Studio

Use `assignPermission()` to assign permissions using the permission code listed in the table above.

Section 5: Configuring Data Access for Siebel RTD

JDBC data sources are used by Siebel RTD to access outside data. Data sources are application server specific, and are used to:

- Identify the 'SDDS' data source that Siebel RTD uses to locate Siebel Real-Time Decision Server system data;
- Identify new JDBC data sources that are to be used as suppliers in Inline Services. These data sources can be RDBMS databases as well as ODBC identified data sources.

5.1 Initializing the database and modifying the SDDS data source

When you install Siebel RTD for JBoss, you are given the option to choose a database for Siebel RTD to initialize for use with the Platform. Initialization consists of two parts: creation of the data Siebel RTD needs to run and creation of a JDBC datasource that the Platform uses to locate the database. If you did not initialize on installation or if you would like to change the location of your JDBC datasource there are scripts provided.

If you want to initialize a database or modify a datasource on a Siebel Real-Time Decision Server installation on a UNIX platform, copy these scripts to a machine running the Windows operating system and run them from there.

5.1.1 Initializing the database

To initialize the database, use the SDDBTool. Before initializing the database, make sure that the Siebel Real-Time Decision Server is not running. SDDBTool is located in \$INSTALLDIR\scripts.

- 1 Run the script SDDBTool. Choose your database type and click **Next**.
- 2 Enter your database settings:

For SQL Server

Host
Database port
Database Name
Runtime User
Runtime User Password XXXX
Administrative User
Administrative User Password XXXX

For Oracle

Host
Database Port
SID
Runtime User
Runtime User Password XXXX
Administrative User
Administrative User Password XXXX

For DB2

Host
Database port
Database Name
Runtime User
Runtime User Password XXXX
Administrative User
Administrative User Password XXXX



Note: The Administrative User entered here must have rights to create tables and stored procedures on the database. The Runtime user is used to access system data at runtime.

Click **Next**.

- 3 Choose **Initialize** or **Upgrade**. Initialize creates the data and datasource needed to run Siebel RTD. Upgrade will upgrade from a previous version to the current.
- 4 Choose **Initialize** or **Upgrade**. Initialize creates the data and datasource needed to run Siebel RTD. Upgrade will upgrade from a previous version to the current.
- 5 After initializing the database, you should add a default administrator group and user using the JMX console. If you are using Windows authentication, use JMX to change the Authentication Provider class. See *7.4.1 About SDCcluster→Security*

5.1.2 **Populating the CrossSell example data**

An example Inline Service is included with the installation. During the JBoss or client tools installation, two tables (CrossSellCustomers and CrossSellResponses) are populated. If you change data sources, or if you need to repopulate these tables for any reason, use the script `InitAppDB` located with the example Inline Service. The script is specific to your database type:

installation_directory\examples\CrossSell\etc\data\SQLServer directory for SQL Server.

installation_directory\examples\CrossSell\etc\data\Oracle for Oracle.

installation_directory\examples\CrossSell\etc\data\DB2 for DB2.

This script takes the following parameters:

```
InitAppDB.cmd $INSTALLDIR $DB_SERVER $DB_PORT $DB_NAME $DB__RUNTIME_USER  
$DB__ADMIN_USER $DB__ADMIN_PASSWORD
```

Where

`$INSTALLDIR` is the path of the *installation_directory*, surrounded by double-quotes (“”);

`$DB_SERVER` is the name of the database server;

`$DB_PORT` is the database port number;

`$DB_NAME` is the name of the database, or for Oracle, the SID;

`$DB__RUNTIME_USER` is the username of the runtime user for the system;

`$DB__ADMIN_USER` is a username of a user that has rights to create tables and stored procedures on the database;

`$DB__ADMIN_PASSWORD` is the password of the administrative user.

5.1.3 **Modifying the SDDS datasource**

The SDDS JDBC data source is used by Siebel RTD to communicate with the data base. JDBC data sources are application server specific.

Modifying the SDDS JDBC data source for JBoss

If you have just initialized the data base, or if you have moved your database to another location, you can modify the JDBC datasource by using ModifySDDS located in \$INSTALLDIR\scripts. Before running the following script, make sure your Siebel Real-Time Decision Server is stopped.

```
ModifySDDS.cmd $DB_TYPE $DB_SERVER $DB_PORT $DB_NAME $DB_USER  
$DB_PASSWORD
```

Where

\$DB_TYPE is sqlserver, oracle, or db2;

\$DB_SERVER is the name of the database server;

\$DB_PORT is the port of the database server;

\$DB_NAME is the name of the database, or for Oracle, the SID;

\$DB_USER is a username on the database; this is a runtime user;

\$DB_PASSWORD is the password associated with \$DB_USER.

After running the script, restart the Siebel Real-Time Decision Server.

Modifying the SDDS JDBC data source for WebLogic

To modify the SDDS data source for WebLogic, refer to the installation and configuration instructions for that platform given in *2.3 Installing and Configuring Siebel RTD for BEA WebLogic Application Server*.

Modifying the SDDS JDBC data source for WebSphere

To modify the SDDS data source for WebSphere, refer to the installation and configuration instructions for that platform given in *2.2.2 Installing the Siebel Real-Time Decision Server for WebSphere*.

5.2 Creating additional JDBC data sources for Siebel Real-Time Decision Server

A JDBC data source is required on your application server in order to allow access to outside data to your Inline Service. JDBC data sources are application server specific.

5.2.1 Configuring JDBC data sources for JBoss

When Siebel RTD Server is running on top of the JBoss application server, the JDBC data source can be configured by using the JMX console. On Windows you can start the console by clicking on **Siebel Analytics**→**RTD**→**Administration (JMX)** from the Start menu. In JMX console open **SDManagement MBean**→**SDCluster** and invoke “addDataSource” method with the following parameters:

Param	Comment
Type	Database type: sqlserver, oracle, db2, or odbc.
DatasourceName	The JNDI name of the data source. For ODBC, any name without spaces.
Server	The server where the data source resides. For ODBC, leave

	blank.
Port	The port on the server that provides access to the data source. For ODBC, leave blank.
Name	The name of the database (or the SID for Oracle). For ODBC, the name of the ODBC data source.
User	The user name used to connect to the server.
Password	The password used to connect to the server.

For more about SDCluster, see 8.2.1 *About SDClusterPropertyManager*.

5.2.2 **Configuring JDBC data sources for WebLogic**

To create a JDBC data source for WebLogic, refer to the WebLogic online help topic *Creating and Deploying JDBC Components—Connection Pools, MultiPools, and Data Sources*.

Once your JDBC data source has been configured within WebLogic, a resource reference must be added to the web.xml file within two archives: soap.war and rtis.war.

These war files are located in the application directory of WebLogic where you exploded the sd.ear file during installation.

Open each archive, soap.war and rtis.war, and extract web.xml from each. Add a resource reference identifier to both web.xml files:

```
<resource-ref id="your-resource-reference-id">
  <res-ref-name>your-jndi-datasource-name</res-ref-name>
  <res-type>javax.sql.DataSource</res-type>
  <res-auth>Container</res-auth>
  <res-sharing-scope>Unshareable</res-sharing-scope>
</resource-ref>
```

Replace *your-resource-reference-id* with the ID of your resource reference and *your-jndi-datasource-name* with the name of your JNDI data source. Note that the ID must be a unique value within the file.

Place your resource reference identifier entry after the existing resource reference identifier. For example:

```
<?xml version="1.0" encoding="ISO-8859-1"?>

<!DOCTYPE web-app PUBLIC
  "-//Sun Microsystems, Inc.//DTD Web Application 2.3//EN"
  "http://java.sun.com/dtd/web-app_2_3.dtd">
<web-app id="SDSOAPWebApp">
  <display-name>Apache-Axis</display-name>

  <servlet>
    <servlet-name>AdminServlet</servlet-name>
    <display-name>Axis Admin Servlet</display-name>
    <servlet-class>
      org.apache.axis.transport.http.AdminServlet
    </servlet-class>
```

```

    <load-on-startup>1</load-on-startup>
</servlet>

<servlet>
  <servlet-name>SOAPMonitorService</servlet-name>
  <display-name>SOAPMonitorService</display-name>
  <servlet-class>
    <!--org.apache.axis.monitor.SOAPMonitorService-->
    com.sigmadynamics.tools.SDSOAPMonitorService
  </servlet-class>
  <init-param>
    <param-name>SOAPMonitorPort</param-name>
    <param-value>5001</param-value>
  </init-param>
  <load-on-startup>2</load-on-startup>
</servlet>

<servlet id="SDAxisServlet">
  <servlet-name>AxisServlet</servlet-name>
  <display-name>SD-Axis Servlet</display-name>
  <servlet-class>
    com.sigmadynamics.services.SDServiceDeployerServlet
  </servlet-class>
  <load-on-startup>3</load-on-startup>
</servlet>

<servlet-mapping>
  <servlet-name>AxisServlet</servlet-name>
  <url-pattern>/AxisServlet</url-pattern>
</servlet-mapping>

<servlet-mapping>
  <servlet-name>AxisServlet</servlet-name>
  <url-pattern>*.jws</url-pattern>
</servlet-mapping>

<servlet-mapping>
  <servlet-name>AxisServlet</servlet-name>
  <url-pattern>/services/*</url-pattern>
</servlet-mapping>

<servlet-mapping>
  <servlet-name>SOAPMonitorService</servlet-name>
  <url-pattern>/SOAPMonitor</url-pattern>
</servlet-mapping>

<!-- uncomment this if you want the admin servlet -->
<servlet-mapping>

```

```

    <servlet-name>AdminServlet</servlet-name>
    <url-pattern>/AdminServlet</url-pattern>
</servlet-mapping>

<!-- currently the W3C havent settled on a media type for WSDL;
    http://www.w3.org/TR/2003/WD-wsdl12-20030303/#ietf-draft
    for now we go with the basic 'it's XML' response -->
<mime-mapping>
    <extension>wsdl</extension>
    <mime-type>text/xml</mime-type>
</mime-mapping>

<mime-mapping>
    <extension>xsd</extension>
    <mime-type>text/xml</mime-type>
</mime-mapping>

<welcome-file-list id="WelcomeFileList">
    <welcome-file>index.html</welcome-file>
    <welcome-file>index.jsp</welcome-file>
    <welcome-file>index.jws</welcome-file>
</welcome-file-list>

<resource-ref id="SDDS_Axis">
    <res-ref-name>SDDS</res-ref-name>
    <res-type>javax.sql.DataSource</res-type>
    <res-auth>Container</res-auth>
    <res-sharing-scope>Unshareable</res-sharing-scope>
</resource-ref>

<resource-ref id="My_Resource_Reference">
    <res-ref-name>NewResource</res-ref-name>
    <res-type>javax.sql.DataSource</res-type>
    <res-auth>Container</res-auth>
    <res-sharing-scope>Unshareable</res-sharing-scope>
</resource-ref>

</web-app>

```

Re-archive the web.xml files.

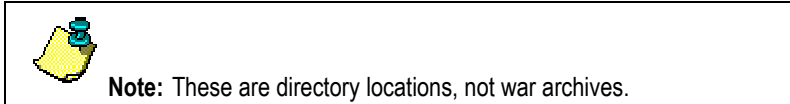
5.2.3 Configuring JDBC data sources for WebSphere

To create a JDBC data source for WebSphere, refer to the WebSphere online help topic *Resource Management: JDBC provider settings*.

Once your JDBC data source has been configured within WebSphere, a resource reference must be added to two files: web.xml and ibm-web-bnd.xmi. These files are located in two separate directory locations:

\websphere_install_directory\AppServer\config\cells\server_name\applications\
SD.ear\deployments\SD\soap.war\WEB-INF\

\websphere_install_directory\AppServer\config\cells\server_name\applications\
SD.ear\deployments\SD\rtis.war\WEB-INF\



5.2.3.1 Changes to the web.xml files

Add the following elements to the web.xml files.

```
<resource-ref id="your-resource-reference-id">  
  <res-ref-name>your-jndi-datasource-name</res-ref-name>  
  <res-type>javax.sql.DataSource</res-type>  
  <res-auth>Container</res-auth>  
</resource-ref>
```

Replace *your-resource-reference-id* with the ID of your resource reference and *your-jndi-datasource-name* with the name of your JNDI data source. Note that the ID must be a unique value within the file.

Place your resource reference identifier entry after the existing resource reference identifier. For example:

```
<?xml version="1.0" encoding="ISO-8859-1"?>  
  
<!DOCTYPE web-app PUBLIC  
  "-//Sun Microsystems, Inc.//DTD Web Application 2.3//EN"  
  "http://java.sun.com/dtd/web-app_2_3.dtd">  
<web-app id="SDSOAPWebApp">  
  <display-name>Apache-Axis</display-name>  
  
  <servlet>  
    <servlet-name>AdminServlet</servlet-name>  
    <display-name>Axis Admin Servlet</display-name>  
    <servlet-class>  
      org.apache.axis.transport.http.AdminServlet  
    </servlet-class>  
    <load-on-startup>1</load-on-startup>  
  </servlet>  
  
  <servlet>  
    <servlet-name>SOAPMonitorService</servlet-name>  
    <display-name>SOAPMonitorService</display-name>  
    <servlet-class>  
      <!--org.apache.axis.monitor.SOAPMonitorService-->  
      com.sigmadynamics.tools.SDSOAPMonitorService  
    </servlet-class>  
    <init-param>  
      <param-name>SOAPMonitorPort</param-name>  
      <param-value>5001</param-value>
```

```

    </init-param>
    <load-on-startup>2</load-on-startup>
</servlet>

<servlet id="SDAxisServlet">
    <servlet-name>AxisServlet</servlet-name>
    <display-name>SD-Axis Servlet</display-name>
    <servlet-class>
        com.sigmadynamics.services.SDServiceDeployerServlet
    </servlet-class>
    <load-on-startup>3</load-on-startup>
</servlet>

<servlet-mapping>
    <servlet-name>AxisServlet</servlet-name>
    <url-pattern>/AxisServlet</url-pattern>
</servlet-mapping>

<servlet-mapping>
    <servlet-name>AxisServlet</servlet-name>
    <url-pattern>*.jws</url-pattern>
</servlet-mapping>

<servlet-mapping>
    <servlet-name>AxisServlet</servlet-name>
    <url-pattern>/services/*</url-pattern>
</servlet-mapping>

<servlet-mapping>
    <servlet-name>SOAPMonitorService</servlet-name>
    <url-pattern>/SOAPMonitor</url-pattern>
</servlet-mapping>

<!-- uncomment this if you want the admin servlet -->
<servlet-mapping>
    <servlet-name>AdminServlet</servlet-name>
    <url-pattern>/AdminServlet</url-pattern>
</servlet-mapping>

<!-- currently the W3C havent settled on a media type for WSDL;
    http://www.w3.org/TR/2003/WD-wsdl12-20030303/#ietf-draft
    for now we go with the basic 'it's XML' response -->
<mime-mapping>
    <extension>wsdl</extension>
    <mime-type>text/xml</mime-type>
</mime-mapping>

<mime-mapping>

```

```

    <extension>xsd</extension>
    <mime-type>text/xml</mime-type>
</mime-mapping>

<welcome-file-list id="WelcomeFileList">
    <welcome-file>index.html</welcome-file>
    <welcome-file>index.jsp</welcome-file>
    <welcome-file>index.jws</welcome-file>
</welcome-file-list>

<resource-ref id="SDDS_Axis">
    <res-ref-name>SDDS</res-ref-name>
    <res-type>javax.sql.DataSource</res-type>
    <res-auth>Container</res-auth>
    <res-sharing-scope>Unshareable</res-sharing-scope>
</resource-ref>

<resource-ref id="My_Resource_Reference">
    <res-ref-name>NewResource</res-ref-name>
    <res-type>javax.sql.DataSource</res-type>
    <res-auth>Container</res-auth>
    <res-sharing-scope>Unshareable</res-sharing-scope>
</resource-ref>

</web-app>

```

5.2.3.2 Changes to the ibm-web-bnd.xmi files

Add a new binding resource reference for each data source to the ibm-web-bnd.xmi file:

```

<resRefBindings xmi:id="your-resource-reference-bindings-id"
    jndiName="your_jndi_name">
    <bindingResourceRef href="WEB-INF/web.xml#resource-reference-id"/>
</resRefBindings>

```

Replace *your-resource-reference-bindings-id* with a unique ID within the file; *your_jndi_name* with the JNDI name of the data source that you defined in WebSphere console, and *resource-reference-id* with the resource reference id that you defined in the web.xml file. For example:

```

<resRefBindings xmi:id="NewResourceRefBindId_123456789"
    jndiName="jdbc/NewResource">
    <bindingResourceRef href="WEB-INF/web.xml#My_Resource_Reference"/>
</resRefBindings>

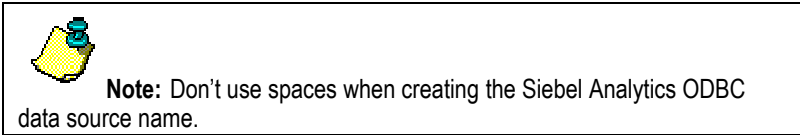
```

Place the binding resource reference after the existing binding resource reference.

5.3 Configuring JDBC Data Sources to access Siebel Analytics data

Before creating a JDBC data source for connecting to Siebel Analytics Server you have to create and configure a Siebel Analytics ODBC data source name as described in *Siebel Analytics Server Administration Guide*.

The ODBC data source has to be created on every machine running the Decision Service.



JDBC data sources are created using application server specific tools.

5.3.1 Creating the Siebel Analytics JDBC data source for WebLogic

- 1 Start the WebLogic administration server for your domain using the Start menu. By default, a 'Start Server' item is added to the BEA Start Menu. For more information on starting the administration server, please refer to your WebLogic documentation.
- 2 Open a browser and enter the url 'http://weblogic_host_machine_name:8080/console'. At the login prompt, enter the administrator username and password.
- 3 In the tree on the left, expand **Services** and **JDBC**. Select **Connection Pool** and, using the link on the right, **Configure a New JDBC Connection Pool**. Choose the type 'Other'.
Choose **Continue**. **Configure a JDBC Connection Pool** appears.

Enter the following properties:

- Name:** The name of your connection pool.
- URL:** 'jdbc:odbc:odbc_name', where *odbc_name* is the name of the ODBC connection you created earlier through Windows Administrative Tools.
- Driver Classname:** 'sun.jdbc.odbc.JdbcOdbcDriver'
- Database user name:** a valid user name on the Siebel Analytics Server.
- Password and Confirm Password:** the password corresponding to the user name above.

- 4 In the Administration Console, test the connection using **Test Connection** button. You should receive a message confirming the test. If not, follow the instructions given on the console to correct the connection settings. When complete, click **Create and Deploy** to deploy the connection pool.
- 5 In the left navigation tree, expand **JDBC** under **Services**. Select **JDBC Data Sources** and use the link on the right pane to **Configure a new JDBC Data Source**. Enter the following:

- Name:** *name your datasource*
- JNDI Name:** *create a JNDI name for your datasource*

Click **Continue**. **Connect to Connection Pool** appears.

- 6 Select your connection pool from **Pool Name**. Click **Continue**. **Target the Data Source** appears.
- 7 Select your server or servers to target and click **Create**.
- 8 Update your web.xml files as referenced in section 5.2.2 *Configuring JDBC data sources for WebLogic*.

5.3.2 Creating the Siebel Analytics Data Source for JBoss

For JBoss, use JMX to configure the data source. The method `addDataSource()` on the JMX MBean **SDCluster** is used to configure the datasource for JBoss. For more about `addDataSource()`, see section 8.2.2 *About SDClusterPropertyManager*.

The parameter values to use to add the datasource are:

Parameter	Parameter Value	Comment
Type	'odbc'	The connection type.
DatasourceName	JDBC_data_source_name	Any name without spaces.
Server		Leave blank.
Port		Leave blank.
Name	ODBC_data_source_name	The name of the Siebel Analytics ODBC data source.
User	username	The user name used to connect to Siebel Analytics Server, for example "Administrator".
Password	password	The password used to connect to Siebel Analytics Server, for example 'SADMIN'.

5.4 Configuring JDBC Data Sources to access Siebel OLTP

To access Siebel OLTP, set up a JDBC data source according to your application server type and database type.

5.4.1 Creating the Siebel OLTP data source for WebLogic

The Siebel OLTP data source is defined just as any other JDBC data source is defined.

Start the WebLogic administration server for your domain using the Start menu. By default, a 'Start Server' item is added to the BEA Start Menu. For more information on starting the administration server, please refer to your WebLogic documentation.

- 1 Open a browser and enter the url 'http://weblogic_host_machine_name:8080/console'. At the login prompt, enter the administrator username and password.
- 2 In the tree on the left, expand **Services** and **JDBC**. Select **Connection Pool** and, using the link on the right, **Configure a New JDBC Connection Pool**. Choose the type specific to the database hosting Siebel OLTP.
Choose Continue. **Configure a JDBC Connection Pool** appears.
- 3 Under Connection Properties, provide the information requested. The required attributes vary by the DBMS and JDBC driver you selected in the previous step. In the Administration Console, test the connection using **Test Connection** button. You should receive a message

confirming the test. If not, follow the instructions given on the console to correct the connection settings. When complete, click **Create and Deploy** to deploy the connection pool.

- 4 In the left navigation tree, expand **JDBC** under **Services**. Select **JDBC Data Sources** and use the link on the right pane to **Configure a new JDBC Data Source**. Enter the following:
 - Name:** 'SIEBEL_OLTP JDBC Driver Provider'.
 - JNDI Name:** 'SIEBEL_OLTP'
- 5 Click Continue. Connect to Connection Pool appears.
- 6 Select your connection pool from **Pool Name**. Click Continue. Target the **Data Source** appears.
 - Select your server or servers to target and click **Create**.
- 7 Stop the server.
- 8 Once your JDBC data source has been configured within WebLogic, a resource reference must be added to the web.xml file within **two** archives: **soap.war** and **rtis.war**. Configure the web.xml files as described in 5.2.2 *Configuring JDBC data sources for WebLogic*.

5.4.2 Creating the Siebel OLTP data source for WebSphere

- 1 Open a browser and enter the url 'http://websphere_host_machine_name:port/admin'. The default port is 9090. At the login prompt, enter the default administrator username, 'admin', or the appropriate administrator username and password.
- 2 In the tree on the left expand **Resources** and select **JDBC Providers**. Click **New**.
- 3 Choose the type of JDBC Provider appropriate to the database hosting Siebel OLTP. **Name:** 'SIEBEL_OLTP JDBC Driver Provider'. Provide the **Classpath** and **Implementation Classname** for the driver specific to your database.
- 4 Select 'SIEBEL_OLTP JDBC Driver Provider'. Select **Data Sources** under **Additional Properties**. Select **New**. In **Name** enter 'SIEBEL_OLTP'. Press return.
- 5 For **Container-managed Authentication Alias**, select a J2C Authentication Data Entries appropriate for Siebel OLTP.
For **Component-managed Authentication Alias**, select a J2C Authentication Data Entries appropriate for Siebel OLTP.
- 6 Configure any **Custom Properties** specific to this data source type.
- 7 At the top of the page, click on **Save**. Click on the **Save** button.
- 8 Stop the WebSphere server using 'Stop the Server' from the WebSphere program menu.
- 9 Once your JDBC data source has been configured within WebSphere, a resource reference must be added to two files: web.xml and ibm-web-bnd.xml. Configure these files as described in 5.2.3 *Configuring JDBC data sources for WebSphere*.
- 10 Restart the server using 'Start the Server' from the WebSphere program menu.

5.4.3 Creating the Siebel OLTP data source for JBoss

For JBoss, use JMX to configure the data source. The values are:

Param	ParamValue	Comment
Type	One of the following database types: sqlserver, oracle, db2, odbc	The connection type.

DatasourceName	'SIEBEL_OLTP'	The JNDI name of the Siebel OLTP datasource
Server	<i>Server name</i>	<i>Server name of the database</i>
Port	<i>Database port</i>	<i>Port on which the database server is listening</i>
Name	'SIEBEL_OLTP JDBC Driver Provider'.	The name of the Siebel OLTP data source.
User	<i>username</i>	The user name used to connect to Siebel OLTP, for example "Administrator".
Password	<i>password</i>	The password used to connect to Siebel OLTP, for example 'SADMIN'.

Section 6: Additional Configuration Settings

Additional configuration settings are available to help maintain and fine tune your system.

6.1 Additional Configuration for Decision Center

Decision Center client browsers should be configured in the following way for optimal performance:

6.1.1 Setting Internet Explorer options

- 1 In Internet Explorer use **Tools**→**Internet Options** to set options.
- 2 On the Advanced tab, uncheck Reuse windows for launching shortcuts.
- 3 Cookies should also be enabled for the browser.

6.2 Additional Configuration for Siebel Real-Time Decision Servers

6.2.1 About Logging Configuration

The Logger attribute of the **SDCluster**→**Server**→**Logger** is a Log4J type logger. For full documentation see the Log4J website at <http://logging.apache.org/log4j>.

6.2.2 Changing log configuration parameters

- 1 From the JMX console home page, click the link **Cluster**.
- 2 From **Servers**, choose the server you would like to adjust logging for.
- 3 From the **Server** click on **View MBean** of the **Logger** attribute. There are three attributes related to logging: **Priority**, **InlineServicePriority**, and **LogFile**.
- 4 The attributes **Priority** and **InlineServicePriority** can be set to one of the following levels and control the level of logging messages that will be written to the server log file:
 - DEBUG
 - INFO
 - WARN
 - ERROR
- 5 The attribute **LogFile** determines the location of the server log and is by default set to `$INSTALLDIR\log\server.log`.
- 6 You can change apply changes to any of the attributes individually by clicking on **Apply** for that attribute, or all at once by clicking on **Apply Changes**.

6.2.3 Configuring Services for Servers

If you are running several servers in your cluster, you may want to assign different services to run on different machines.

- 1 From the JMX console home page, click the link **Cluster**.
- 2 From **Servers**, choose the server you would like to assign the service too. From the **Server** click on **Properties** and then **Misc**.
- 3 Enable or Disable the Services for this server. You must have the Learning Service, Decision Service and Alert Service enabled on at least one server in your cluster.
- 4 Servers that have had changes must be restarted.

Section 7: Production Deployment of Siebel RTD

Once an Inline Service is tested and ready for production deployment, you will deploy it to one or more servers for production. Whether you choose an all-in-one configuration or a full deployment configuration will determine the exact hardware required to support an Inline Intelligence™ solution. System configuration may need to be adjusted based upon specific customer requirements and application characteristics.

7.1 All-In-One Configuration

The All-In-One configuration is to support a proof of concept or pilot type implementation. The intent is not to support a system in production for an unspecified length of time, but to allow the customer to better understand the benefits of deploying an Inline Intelligence solution. The configuration supports approximately 100 account managers, or about 50 call center agents.

Inline Intelligence™ Platform	Minimum Configuration
J2EE Application Server	Embedded JBOSS
OS	Windows
Number of CPUs	2
CPU	2+ GHz
RAM	2 GB
HD	100 GB
Database	SQL Server, Oracle or DB2

7.2 Full Deployment Configuration

This configuration will support an implementation of approximately 1500 B2B Account Managers or 1000 call center agents in a production environment. There are a number of factors that determine the system configuration. They include the peak number of requests per second, the number of choices, the number and size of models, and the number data sources. The Siebel Systems Professional Services Group can assist in determining the most appropriate configuration given the business and technical requirements.

Server	Component	Minimum Configuration
Inline Intelligence Servers (2)	CPU	2 X 2-CPU, 2+GHz, 2GB RAM
	J2EE Application Server	WebSphere, Web Logic, or JBoss
	OS	Windows, Solaris, or Linux
Database Server ¹	CPU	4-CPU, 2+GHz, 8GB RAM
	DBMS	Oracle, SQLServer or DB2

Learning Server ²	CPU	2-CPU, 2+GHz, 2GB RAM
	J2EE Application Server	WebSphere, Web Logic, or JBoss,
	OS	Windows, Solaris, or Linux
<p>Notes:</p> <p>1: The ratio of database server CPUs to Inline Intelligence Server CPUs is approximately 1:1.</p> <p>2: The ratio of Learning Server CPUs to Inline Intelligence Server CPUs is approximately 1:4.</p>		

Section 8: JMX Management of Siebel RTD

Siebel RTD uses the J2EE industry standard, Java Management Extensions (JMX) to manage the Management Service, Decision Service, Learning Service and deployed Inline Services. Siebel RTD comes packaged with a JMX implementation and browser-based console. Third party JMX implementations may also be used to manage Siebel RTD.

JMX MBeans manage various aspects of Siebel RTD including logging, Service configuration, security and users. Siebel RTD is comprised of three Services:

Management Service: manages the generation of Inline Services from configurations that are created using Siebel Decision Studio.

Decision Service: runs Inline Services and integrates to enterprise operational processes.

Learning Service: maintains analytic, self learning models that underlie Inline Services.

Deployments of Siebel RTD are often done across multiple servers as well as in clusters to enhance performance in high transaction environments. A relational database is used by each of these Services for retention of code, transactional data and configurations.

8.1 About JMX MBean operations and attributes

Siebel RTD MBeans are all accessed through the **SDCluster** MBean endpoint. This endpoint appears by default when the JMX Console is started through the **Siebel Analytics**→**RTD** menu.

The JMX Console uses these MBeans to manage the various aspects of Siebel RTD. Each MBean consist of Attributes and Operations that are used for informational and management purposes.

Attributes and operations at the SDCluster level are meant to manage cluster level features. The organization of the MBeans is hierarchical. The attributes of SDCluster are:

1. Properties – SD Cluster Properties Configuration.
2. Members – Members of the cluster. Each Member is listed through this attribute. Member is used to manage local server level properties.
3. Security – Security allows management of various authentication attributes. Many of these are specific to one authentication type and will not appear for other authentication types.
4. InlineServiceManager – Manages deployed Inline Services.
5. DeploymentStates – Allows the setup and ordering of deployment states.
6. LearningService – Administers the Learning Service attributes.
7. AlertService – Administers the Alerts Service.

8.2 Siebel RTD cluster level management

Management at the cluster level is for items that impact the entire cluster of servers. Note that if you have only one server, there is still cluster-level management.


8.2.1 About SDCluster

At the cluster level you can:

- Add and remove JDBC data sources
- Use the **View MBean** links to gain access to:

Attribute	Description
Properties	Cluster Properties Manager
Members	Members of the cluster
Security	Security Manager
InlineServiceManager	Inline Services
DeploymentStates	Define deployment states for Inline Services
LearningService	Learning Service attributes
AlertService	Alert Service attributes

The SDCluster MBean has the following operations:



Note: These two operations are specific to the JBoss application server. To add or remove data sources from WebLogic or WebSphere, refer to *5.2 Creating additional JDBC data sources for Siebel Real-Time Decision Server*.

java.lang.Void addDataSource()

Parameters: Type: one of the following database types: sqlserver, oracle, db2, odbc
DataSource name: JNDI name of the datasource
Server: Server name of the database; for ODBC, leave this blank
Port: Port on which the database server is listening; for ODBC, leave this blank
Name: Database name; for ODBC, this is the ODBC name
User: Database user name
Password: Database password

Purpose: adds a data source for use by Inline Services. Note that for an ODBC data source only Datasource name, Name, User, and Password are required.

java.lang.Void removeDataSource()

Parameters: Datasource name: JNDI name of the data source

Purpose: removes the data source

8.2.2 About SDClusterPropertyManager

From SDClusterPropertyManager you can restore the cluster default properties and gain access to:

Attribute	Description
Misc	Miscellaneous properties
Cluster	Cluster configuration
Deployment	Configuration for deployment

SDClusterPropertyManager has the following operation:

java.lang.Void restoreDefault()

Parameters: none

Purpose: Restores the default installation settings to Siebel RTD. If this command is run on a cluster, then values are restored to cluster defaults.

8.2.3 About SDClusterPropertyManager→Misc

The Misc MBean allows you to adjust the following attributes:

Attribute	Description
DisableBatchDBOperations	Boolean switch that controls batch database operations.
AutoFlushTimeout	Interval in seconds controlling auto flush of database write buffers. Fractional values are supported.
WorkerThreadPoolSize	The number of threads used for general purpose maintenance activities, not for normal Integration Point request processing. Maintenance activities include model maintenance, session timing, and timed-out request processing.
IntegrationPointRequestIOFactor	The percentage of time Integration Point requests spend doing IO, or otherwise waiting for systems external to this VM.
IntegrationPointMaxConcurrentJobs	The maximum number of concurrently executing Integration Point requests. This should normally be set to 0, in which case the value is calculated as follows, where Math.ceil means "round up to the next higher integer". $\text{NumCPUs} * \text{Math.ceil}(1/(1 - \text{DSRequestIOFactor})) + 5$
IntegrationPointQueueSize	The maximum number of Integration Point requests that can wait to execute. If a request tries to exceed this number, the server terminates the request with a "Server Too Busy" error message. This setting should be less than or equal to the number of servlet threads allocated by the servlet container minus the configured or calculated value of IntegrationPointMaxConcurrentJobs. The calculated value of IntegrationPointMaxConcurrentJobs can be seen in SDCConsole/Members/Decision Service.

IntegrationPointGuaranteedRequestTimeout	Guaranteed response time, in milliseconds, for Integration Point requests. (Service Level Guarantee). Zero means don't timeout Integration Point requests - suitable for debugging only.
DBOperationLogThresholdMilliSec	All database operations that take longer than the specified threshold are logged.
DCOperationLogThresholdMilliSec	All decision center requests that take longer than the specified threshold are logged.
SystemDSName	The JNDI name of the datasource
ModelDSName	The JNDI name of the datasource used by the Learning Service
ArchivedModelCatalogRefreshInterval	Refresh interval in seconds for a catalog of archived models. The catalog is used by Discovery Explorer.
ArchivedModelCacheTimeToLive	Maximum time in seconds an archived model is preserved in memory. The cache of archived models is used by Discovery Explorer.
DSQueueThreadCount	The number of JMS Threads listening to the DS request queue
DSSessionIdleTimeoutMilliSec	Decision Service session idle timeout in milliseconds.
DSManagesSessionAffinity	Decision Service manages session affinity. When set to true, the decision service maintains a map of active session keys and, if necessary, will forward Integration Point requests to the cluster host owning the key's session. Should be disabled in single-host installations and in installations where session affinity is perfectly managed by the app server or external load balancer.
CustomLoginLogo	Custom image for login page
CustomHeaderLogo	Custom image for main page
SMTPHost	SMTP Mail server host name. Currently only used to send alert notifications.
SMTPPort	SMTP Mail server port.
SMTPUser	SMTP Mail server user name, if SMTP authorization is required.
SMTPPassword	SMTP Mail server password, if SMTP authorization is required.

8.2.4 About SDClusterPropertyManager→ClusterManager

ClusterManager allows you to adjust the following attributes:

Attribute	Description
GenerateDSCookies	Generate Decision Server HTTP Cookies. Set to true to have Decision Server associate Integration Point requests with HTTP sessions, thus causing the web container to generate container-

	specific session-affinity cookies.
JGroupProtocols	Defines the protocols and properties used for the JGroups channel.
JGroupDSProtocols	Defines the JGroups protocols and properties used for the Decision Server forwarding channel.
SDGroupName	This is the name of the cluster.
LearningServiceInitialWait	The number of milliseconds to wait when the server first starts up before trying to start the Learning Service.
LearningServiceRestartWait	The number of milliseconds to wait after a machine fails or leaves the cluster before trying to restart the AlertService.
AlertServiceInitialWait	The number of milliseconds to wait when the server first starts up before trying to start the Alert Service.
AlertServiceRestartWait	The number of milliseconds to wait after a machine fails or leaves the cluster before trying to restart the Alert Service.
OperationalDataCleanupPeriod	The number of hours (fractions are allowed) between cleanup of the operational data (choice history, statistics, learning data storage) in the database.
StatisticsCleanupChunkSize	The chunk size to use when deleting old statistic records.
ChoiceHistoryCleanupChunkSize	The chunk size to use when deleting old choice history records.
LearningDataStorageCleanupChunkSize	The chunk size to use when deleting old learning data storage records.
StatisticsCleanupThrottle	A number between 0.1 and 1, inclusive. Higher throttle corresponds to higher speed.
ChoiceHistoryCleanupThrottle	A number between 0.1 and 1, inclusive. Higher throttle corresponds to higher speed.
LearningDataStorageCleanupThrottle	A number between 0.1 and 1, inclusive. Higher throttle corresponds to higher speed.

8.2.5 **About SDClusterPropertyManager→Deployment**

Deployment allows you to adjust the following attributes:

Attribute	Description
AppPollingInterval	How frequently, in seconds, the AppFactory polls the SDApps table to see if there are new apps.

8.2.6 About SDClusterPropertyManager→Alert Service

Alert Service allows you to adjust the following attributes:

Attribute	Description
MailFrom	Alert email notifications will be sent from this address.
MailLogo	Alert email notifications will have this logo in their header.
AlertHistoryCleanupIntervalMinutes	The Alert History archive should be scanned at this interval for expiration of old alerts. Units are minutes.
AlertDispatcherSleepIntervalMinutes	The Alert Service should evaluate alerts at this interval. Units are minutes.
AlertDispatcherErrorSleepIntervalMinutes	The Alert Service should fall back to evaluating alerts at this interval if there are any errors. Units are minutes.

8.3 Siebel RTD Member Management

Member level management is for items that impact individual servers. Each server must be administered through the JMX console resident on the machine, even if part of a cluster.

8.3.1 About SDCluster→Member

The Server MBean contains the following MBeans for server level management.

Attribute	Description
Logger	SD Logger Configuration.
Properties	SD Properties Configuration.
DecisionService	Decision Service Configuration

8.3.2 About SDCluster→Members→Logger

Logger allows you to adjust the following attributes:

Attribute	Description
Priority	Current logging priority for the 'com.sigmadynamics' category. Valid values are 'DEBUG', 'ERROR', 'INFO', 'WARN'
InlineServicePriority	Current logging priority for Inline Services. Valid values are 'DEBUG', 'ERROR', 'INFO', 'WARN'
LogFile	Location of the current logfile.

For analysis of values written to the log file, see *Siebel Decision Studio Reference Guide, Section 5: Troubleshooting and Debugging Inline Services*.

8.3.3 About SDCluster→Members→Properties

From SDPropertyManager you can restore the member default properties and gain access to:

Attribute	Description
Misc	Miscellaneous properties
PerformanceMonitoring	Performance Counter Properties

SDCluster→Members→Properties has the following operation:

java.lang.Void restoreDefault()

Parameters: none

Purpose: Restores the default installation settings to Siebel RTD.

Properties changes at the member level are persisted to both the server file system and the database and will survive a re-install. SDCluster Properties override member level properties.

8.3.3.1 About SDCluster → Members → Properties →Performance Monitoring

Performance Monitoring allows you to adjust the following attributes:

Attribute	Description
DSPerfCounterEnabled	Enables the writing of DS performance counters. This should not be enabled indefinitely, because the file grows without limit.
DSPerfCounterAppend	If true, performance data is appended to an existing file, if any, otherwise any existing file is overwritten when the server restarts.
DSPerfCounterLogFile	The tab-separated CSV file into which DS performance counts are periodically appended. If MS Excel is available, ds_perf.xls, supplied in the installation's etc directory provides a convenient view.
DSPerfCounterLogInterval	The update interval in milliseconds for DS performance counts.

For more about using Performance Monitoring, see the *Siebel Decision Studio Reference Guide: Using Performance Monitoring*.

8.3.3.2 About SDCluster → Members → Properties →MISC

The Misc MBean allows you to adjust the following attributes:

Attribute	Description
DecisionServiceEnabled	Whether or not Decision Service should run in this instance.
LearningServiceEnabled	Whether or not Learning Service should run in this instance.
AlertServiceEnabled	Whether or not Alert Service should run in this instance.

WorkerThreadPoolSize	The number of threads used for general purpose maintenance activities, not for normal Integration Point request processing. Maintenance activities include model maintenance, session timing, and timed-out request processing.
NumCPUs	Number of cpus in this host. In MS Windows, for a hyper-threaded machine use half the number of processors shown by Windows Task Manager.
IntegrationPointRequestIOFactor	The percentage of time Integration Point requests spend doing IO, or otherwise waiting for systems external to this VM.
IntegrationPointMaxConcurrentJobs	The maximum number of concurrently executing Integration Point requests. This should normally be set to 0, in which case the value is calculated as follows, where Math.ceil means "round up to the next higher integer". $\text{NumCPUs} * \text{Math.ceil}(1/(1 - \text{DSRequestIOFactor})) + 5$
IntegrationPointQueueSize	The maximum number of Integration Point requests that can wait to execute. If a request tries to exceed this number, the server terminates the request with a "Server Too Busy" error message. This setting should be less than or equal to the number of servlet threads allocated by the servlet container minus the configured or calculated value of IntegrationPointMaxConcurrentJobs. The calculated value of IntegrationPointMaxConcurrentJobs can be seen in SDConsole/Members/Decision Service.
WebServerPort	The port of the webserver. Requires server restart before the property change takes effect.
HTTPSEnabled	Whether or not HTTPS is enabled. Requires server restart before the property change takes effect.

8.3.4 About SDCluster→Members→DecisionService

The DecisionService MBean has the following attributes:

Attribute	Description
MaxAllowedConcurrentRequests	Maximum number of requests that could be allowed to run concurrently.
CurrentRequestsRunning	Number of currently running requests.
CurrentRequestsQueued	Number of currently waiting requests, not yet running.
TotalRequests	Total number of requests seen since server started.
TimedOutRequests	Total number of requests that have timed out.
RequestsForwarded	Total number of requests forwarded to another server in the cluster.

RequestQueueCapacity	Maximum number of requests that could be allowed to wait concurrently.
RequestsQueued	Total number of requests that have had to wait before running.
PeakRequestsQueued	Largest number of requests that have had to wait at any one time.
RequestsWhenQueueFull	Total number of requests rejected because the request queue was full.
CurrentSessions	Number of Decision Service sessions still open.
TotalSessions	Total number of Decision Service sessions created.

8.4 Siebel RTD security management

Security management is for items that impact authentication settings on the cluster. A link to the Security MBean is listed in the SDCluster. Different security settings are displayed depending on the type set in Security Authenticator.

8.4.1 About SDCluster→Security

Security has the following attributes:

Attribute	Description
SecurityProperties	Security properties configuration.
Authenticator	Authenticator properties and test operations.

Security has the following operations:

addGroupMember()

Parameters: group Name: The group's name.
username: The user's name.

Purpose: Makes a user or group a direct member of a group.

assignPermission()

Parameters: userOrGroup: A user name or group name.
permCode: The code of the permission.

Purpose: Assigns the specified cluster permission.

createGroup()

Parameters: groupName: The group's name.
description: Description for the new group.

Purpose: Creates a new security group. Siebel RTD Platform Authentication only.

createUser()

Parameters: username: The user's name.
description: The new user's description.
password: The user's password, in clear text.

Purpose: Creates a new user. Siebel RTD Platform Authentication only.

deleteGroup()

Parameters: groupName: The group's name.

Purpose: Deletes a security group. Siebel RTD Platform Authentication and Siebel Object Manager Authorization only.

deleteUser()

Parameters: username: The user's name.

Purpose: Deletes an existing user. Siebel RTD Platform Authentication only.

listDirectGroupMembers()

Parameters: groupName: The name of the target group.
includeGroups: True or False. If true, include member groups, otherwise include only users.

Purpose: Returns the names of users and groups directly as a member of a specified group.

listDirectlyContainingGroups ()

Parameters: userOrGroup: The user's or group's name.

Purpose: Returns the names of groups containing a specified user or group as a direct member.

listDirectPermissions()

Parameters: userOrGroup: A user name or group name.

Purpose: Returns the permissions directly assigned to a specific user or group.

listEffectivePermissions()

Parameters: userOrGroup: A user name or group name.

Purpose: Returns the permissions directly assigned to a specific user or group or to any of its directly or indirectly containing groups.

listEveryoneHavingDirectPermissions()

Parameters: none

Purpose: Returns the names of users and groups having directly assigned permissions.

listGroupNames()

Parameters: none

Purpose:	Returns the names of all groups to which permissions may be assigned.
listPermissionCodes()	
Parameters:	none
Purpose:	Returns the localized name and integer code for all possible permission types. The codes are used as operands for various operations that require a permission type designation.
listUserNames()	
Parameters:	none
Purpose:	Returns the names of all users to which permissions may be assigned.
removeAllDirectPermissions ()	
Parameters:	userOrGroup: A user name or group name
Purpose:	Removes all permissions directly assigned for the specified user or group.
removeGroupMember()	
Parameters:	containingGroupName: The group to contain the member group. memberGroupName: The group to be added as a member of another group.
Purpose:	Removes a user or group from a containing group.
removePermission()	
Purpose:	Removes a permission from its direct assignment to a user or group. The permission is not removed from any groups containing the specified user or group.
Parameters:	userOrGroup: A user name or group name. permCode: The code of the permission.
renameGroup()	
Parameters:	oldGroupName: The group's old name. newGroupName: The group's new name.
Purpose:	Renames a group.
renameUser()	
Parameters:	oldUserName: The user's old name. newUserName: The user's new name.
Purpose:	Renames a user.
setPassword()	
Parameters:	username: The user's name. password: The new password, in clear text.

Purpose: Changes a user's password.

8.4.2 About SDCluster→Security→SecurityProperties

SecurityProperties has the following attributes:

Attribute	Description
AuthenticationEnabled	When false, password checking and permission checking is turned off.
AuthenticatorLogsInvalidCredentials	When true the Authenticator logs a short message whenever a login failure is believed to have been caused by invalid credentials.
AuthenticatorLogsFailures	When true the Authenticator logs details about each failed login, including an exception stack trace. This level of detail is provided unconditionally for errors deemed by the security provider to be caused by configuration problems; this flag provides similar details for errors that the provider thinks, perhaps incorrectly, are caused by invalid credentials.
AuthenticationProviderClass	The name of the class that performs logins, group membership queries, and optionally manages the definition of users and groups. Use com.sigmadynamics.server.security.DBAuthenticator or com.sigmadynamics.server.security.WindowsAuthenticator

8.4.3 About SDCluster→Security→Authenticator

Authenticator has the following attributes:

Attribute	Description
SecurityManager	Security Manager
ProviderClass	Authentication provider class.

Authenticator has the following methods:

java.lang.String loginUserPassword()

Parameters: username: The user's name.
password: The user's password, in clear text.

Purpose: Login with a user name and password, to receive a login ticket.

java.lang.String getUserName()

Parameters: ticket: The ticket returned from the login.

Purpose: Get the user name associated with the login session.

java.lang.Boolean isUserInRole()

Parameters: ticket: The ticket returned from the login.
userOrGroupName: The name of the user or group being tested for identity or containment, respectively.

Purpose: Returns true if the logged in user is the specified user or is directly or indirectly in the specified group.

java.lang.Void logout()

Parameters: ticket: The ticket returned from the login.

Purpose: Terminates a login session.

8.5 About Inline Service Manager

Inline Service Manager allows you to manage the Inline Services deployed on the cluster. A link to **InlineServiceManager** MBean is listed in the SDCIcluster.

InlineServiceManager has the following operation:

removeAllServices ()

Parameters: none

Purpose: Removes all inline services (loaded, loadable, failed).

Each deployed Inline Service is displayed in the attribute **InlineServices**. Click on each Inline Service name to manage the Inline Service.

8.5.1 About SDCIcluster→InlineServiceManager→ [Inline Service]

InlineServiceManager for a specific Inline Service has the following attributes:

Attribute	Description
ServiceId	Service id
DeploymentState	Deployment state
Flag	Flag
LockStatus	Lock status

InlineServiceManager for a specific Inline Service has the following operations:

Operation name	Parameters	Purpose
unlockService()	none	Unlocks this service.
removeService()	none	Stops an inline service in this server and removes the service from the database.
flushStatistics()	none	Flushes all of the statistics for this service to the database.

deleteStatistics()	none	Flushes and deletes all of the statistics for this service from the database.
deleteChoiceHistory()	none	Deletes all of the choice history for this service from the database.
deleteAllOperationalData()	none	Deletes all of the operational data for this service from the database. This includes choice history, statistics, and the study.
deleteStudy()	none	Removes the study for this service

8.6 About Deployment States Management

Deployment States allows you to add, edit, delete and reorder Deployment States. A link to the Deployment States MBean is listed in the SDC cluster. By viewing the MBean, you see a list of deployment states that are available on the cluster.

8.6.1 About SDC cluster → Deployment States

Deployment States have the following attributes:

Attribute	Description
StateObjectNames	A listing of all Deployment States available on the server.

Deployment States has the following operations:

addDeploymentState ()

Parameters: Name - Deployment state name
 AfterDeploymentState - Name of the deployment state after which the new one should be created (if no value is passed in, the new deployment state will be created at the beginning of the list.
 AllowHotSwapping - Allows hot swapping of Inline Services. If Hot Swapping is enabled for this deployment state, and an Inline Service is re-deployed in this state, the existing Inline Service will remain active until all existing sessions close or timeout. New sessions will be created on the newly deployed Inline Service.

Purpose: Creates a new deployment state.

8.6.2 About Deployment State

Clicking on a Deployment State listed will give you the ability to edit or remove that state. Deployment States have the following attributes:

Attribute	Description
Name	Name of the deployment state.
Id	Id of the deployment state.
AllowHotSwapping	Allow hot swapping of Inline Services with this deployment state in

	Decision Service.
--	-------------------

Deployment States has the following operations:

remove ()

Parameters: None.

Purpose: Removes a deployment state from the cluster.

8.7 About Learning Service management

Management of the Learning Service on the cluster allows you to

8.7.1 About SDCluster→ (Server) → LearningService MBean

The LearningService MBean has the following attributes:

Attribute	Description
Studies	A list of all Studies running on the Learning Server. The models of a Study are viewed by clicking on a Study.

By clicking on a Study link, you see a view of that Study. Studies have the following attributes:

Attribute	Description
Models	Models belonging to this study.
Name	The name of the study.

The Study MBean has the following operations:

Delete()

Parameters: None

Purpose: Deletes this study.

By clicking on a Model link, you see the attributes of that model. Models have the following attributes:

Attribute	Description
Attributes	Names of the model attributes.
Name	The name of the model.

The Model MBean has the following operations:

Delete()

Parameters: None

Purpose: Deletes this model.

DeleteAttributeValue()

Parameters: AttributeName - The name of an attribute.
Value - The value to be deleted.

Purpose: Erases model data collected for a value of an attribute.

DeleteAttributeValueRange()

Parameters: AttributeName - The name of an attribute.
LowestValue - The lowest value to be deleted.
HighestValue - The highest value to be deleted.

Purpose: Erases model data collected for a range of values of an attribute.

StartNewTimeWindow()

Parameters: None

Purpose: Closes the current time window and starts a new one. This operation should not be used in production environment since it may impair future model learning.

8.8 About Alert Service Management

The Alert Service Manager allows you to adjust parameter related to how the Alert Service operates. A link to the Alert Service MBean is available on SDCluster.

8.8.1 About SDCluster→Alert Service

Alert Service allows you to adjust the following attributes:

Attribute	Description
MailFrom	Alert email notifications will be sent from this address.
MailLogo	Alert email notifications will have this logo in their header.
AlertHistoryCleanupIntervalMinutes	The Alert History archive should be scanned at this interval for expiration of old alerts. Units are minutes.
AlertDispatcherSleepIntervalMinutes	The Alert Service should evaluate alerts at this interval. Units are minutes.
AlertDispatcherErrorSleepIntervalMinutes	The Alert Service should fall back to evaluating alerts at this interval if there are any errors. Units are minutes.