

Oracle® Real-Time Decisions for
Siebel Intelligent Offer Generation
Installation and Reference Guide

Version 3.0

January 2011

Copyright © 2011, Oracle. All rights reserved.

The Programs (which include both the software and documentation) contain proprietary information; they are provided under a license agreement containing restrictions on use and disclosure and are also protected by copyright, patent, and other intellectual and industrial property laws. Reverse engineering, disassembly, or decompilation of the Programs, except to the extent required to obtain interoperability with other independently created software or as specified by law, is prohibited.

The information contained in this document is subject to change without notice. If you find any problems in the documentation, please report them to us in writing. This document is not warranted to be error-free. Except as may be expressly permitted in your license agreement for these Programs, no part of these Programs may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose.

If the Programs are delivered to the United States Government or anyone licensing or using the Programs on behalf of the United States Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the Programs, including documentation and technical data, shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement, and, to the extent applicable, the additional rights set forth in FAR 52.227-19, Commercial Computer Software--Restricted Rights (June 1987). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

The Programs are not intended for use in any nuclear, aviation, mass transit, medical, or other inherently dangerous applications. It shall be the licensee's responsibility to take all appropriate fail-safe, backup, redundancy and other measures to ensure the safe use of such applications if the Programs are used for such purposes, and we disclaim liability for any damages caused by such use of the Programs.

Oracle, JD Edwards, PeopleSoft, and Siebel are registered trademarks of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

The Programs may provide links to Web sites and access to content, products, and services from third parties. Oracle is not responsible for the availability of, or any content provided on, third-party Web sites. You bear all risks associated with the use of such content. If you choose to purchase any products or services from a third party, the relationship is directly between you and the third party. Oracle is not responsible for: (a) the quality of third-party products or services; or (b) fulfilling any of the terms of the agreement with the third party, including delivery of products or services and warranty obligations related to purchased products or services. Oracle is not responsible for any loss or damage of any sort that you may incur from dealing with any third party.

1	What's New in This Release	5
2	Overview	7
2.1	Architectural Overview	7
2.2	IOG installation and configuration	9
2.3	Inbound Offer Management	10
3	Configuring IOG Inline Service in Oracle RTD Platform	11
3.1	Install and configure the Oracle RTD Server	11
3.2	Overview	11
3.3	Configuring JDBC Data Sources.....	11
3.4	Creating the Entity Views in the Siebel OLTP database	12
3.5	Setting up Oracle RTD server Authentication.....	14
3.5.1	Setting up Oracle RTD Platform (DB) Authentication.....	14
3.5.2	Setting up Siebel Object Manager Authentication.....	14
3.6	Specifying the list of trusted hosts for Decision Service requests	15
3.7	Deploying the Siebel IOG Inline Service to the Oracle RTD Server	16
3.7.1	Importing a Siebel IOG Inline Service into Oracle RTD Decision Studio	16
3.7.2	Verifying OLTP data source is accessible by Oracle RTD	17
3.7.3	Deploying IOG Inline Service to Oracle RTD server.....	19
3.8	Configuring RTD Dynamic Choices	19
3.9	Testing the Deployed Inline Service via Web Service Calls	26
4	Configuring IOG in Siebel Enterprise Server 8.1.....	29
4.1	About Oracle Real-Time Decisions for Call Center	29
4.2	Scenario for Intelligent Offer Generation in Call Center.....	29
4.3	Process of Configuring the IOG Application for Siebel Call Center	30
4.3.1	Checking prerequisites for IOG.....	30
4.3.2	Configuring IOG with Siebel Views for Siebel Call Center.....	30
4.3.3	Setting Siebel Enterprise Server Parameters for IOG	31
4.3.4	Configuring Siebel Run-Time Events for Oracle RTD for Siebel Call Center.....	32
4.3.5	Customizing Call Reasons for Oracle RTD for Siebel Call Center	32
4.4	Process of Setting Up Real-Time Intelligent Offers for Siebel Call Center	33
4.4.1	Defining the Campaign for IOG for Siebel Call Center	33
4.4.2	Creating and Modifying IOG Offers for Siebel Call Center.....	33
4.4.3	Enabling IOG Offers for Siebel Call Center.....	33
4.4.4	Opening Decision Center for Editing for IOG for Siebel Call Center	34
4.4.5	Specifying Eligibility for Offers for Oracle RTD for Siebel Call Center	34
4.4.6	Adding Eligibility Rules	35
4.4.7	Redeploy the RTD Inline Service if there is Any Change On It.....	36
4.4.8	IOG Views for Call Center Agents.....	37
5	Configuration Instructions for Siebel IOG	39
5.1	Seed Data.....	40
5.2	Siebel Objects Changes (Repository Changes).....	40
5.2.1	Embedding offer recommendation applet in multiple views	40
5.2.2	Showing offer details along with Offer list	44
5.2.3	Manually triggering offer recommendations from RTD.....	47
5.2.4	Customizing customer responses to offers	48

- 5.2.5 Configuring the offer attributes passed from RTD to Siebel58
- 5.2.6 Selecting RTD Advisor59
- 5.2.7 Selecting RTD inline service60
- 5.2.8 Customizing the session key.....61

- 6 Optional Setups63
 - 6.1 Setting up Demo Intelligent Cross Sell and Retention Management.....63
 - 6.1.1 Single-customer demo tests63
 - 6.1.2 Bulk customers demo tests65
 - 6.2 Utilizing analytics data via OLAP tables.....66
 - 6.2.1 Create and populate the OLAP Tables for History Entities66
 - 6.2.2 Configure OLAP data source.....68
 - 6.2.3 Test the OLAP data source69
 - 6.2.4 Modify IOG Inline Services to include the history attributes69

- 7 Appendix73
 - 7.1 Appendix A: Siebel IOG user properties.....73
 - 7.1.1 Response Button Disable.....73
 - 7.1.2 Interested73
 - 7.1.3 NotInterested73
 - 7.1.4 Contact Required Flg.....74
 - 7.1.5 Reason74
 - 7.2 Appendix B: RTD Internationalization Consideration76

1 What's New in This Release

This guide documents the installation, configuration and administration tasks for the Oracle Real-Time Decisions for Siebel Intelligent Offer Generation product. The product is an integration effort among the Siebel Enterprise Server (Call Center and Marketing applications), the Oracle Real-Time Decisions (RTD) platform, and application logic in both systems to enable intelligent offer generation and retention management.

The Siebel Intelligent Offer Generation product is referred to as IOG or IOG application in this document.

IOG 3.0 provides the following enhancements to previous version 2.2.1:

Topic	Description
Configuring RTD Dynamic Choices	<p>Support for RTD Dynamic Choice. The offer synchronization function in IOG 2.2.1 is obsolete in this release. IOG 3.0 comes with the support for RTD Dynamic Choices so that RTD can directly access Siebel database for offer attributes. Included are instructions on how to set up the dynamic choice in RTD Decision Studio for IOG.</p>
<ul style="list-style-type: none"> • Embedding offer recommendation applet in multiple views • Showing offer details along with Offer list. • Manually triggering offer recommendations from RTD. • Customizing customer responses to offers. • Configuring the offer attributes passed from RTD to Siebel. • Selecting RTD Advisor. • Selecting RTD inline service. • Customizing the session key. 	<p>Enhanced customizability. You can configure IOG 3.0 to meet your own requirements for these tasks.</p>

2 Overview

IOG uses real-time predictive analytics and dynamic business logic to:

- Predict attrition / churn indicators of a contact or account.
- Predict the optimal retention treatment in case of high attrition / churn risk.
- Predict the optimal cross and up sell offers in lack of high attrition / churn risk.

IOG also enables marketing users to:

- Create and manage marketing offers and retention treatments in Siebel Marketing, and make these offers available for real-time optimization using the Oracle RTD platform.
- Track drivers of offer response behavior directly through embedded intelligence reports in Siebel Marketing and Call Center.

2.1 Architectural Overview

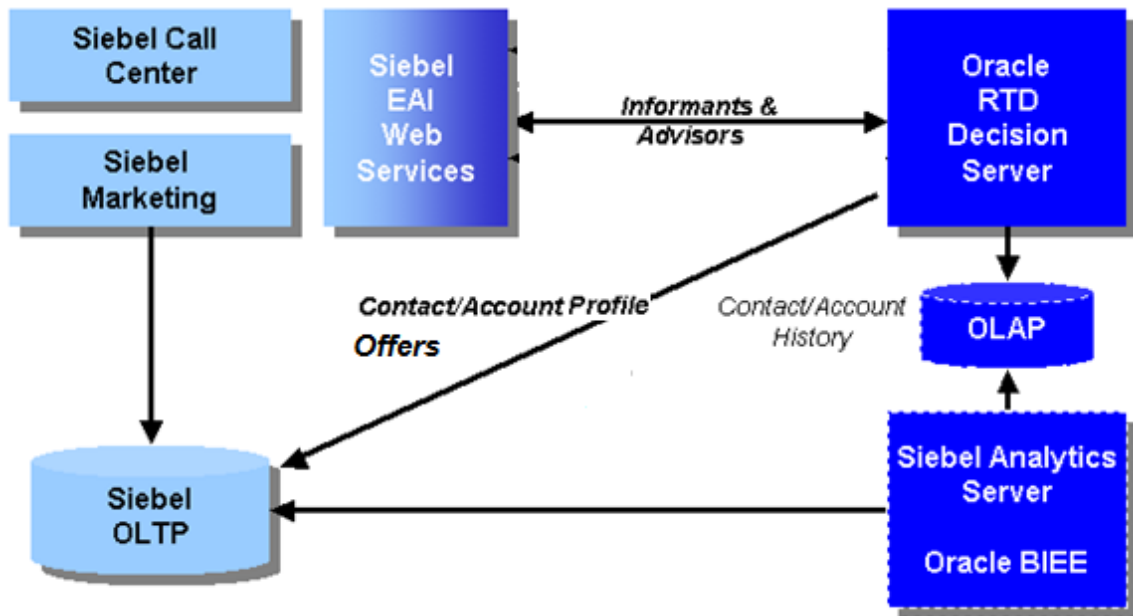


Figure 1. IOG architecture.

Figure 1 shows the high level IOG architecture:

- The Object Manager of Siebel Call Center and Marketing communicates with Oracle RTD platform via Informants and Advisors – these are Oracle RTD integration points used by Siebel to submit contextual data and request offer recommendations from Oracle RTD. For more information about Informant and Advisor integration, refer to the Oracle RTD platform document *Oracle® Real-Time Decisions Platform Developer's Guide, Version 3.0*. This document is available on Oracle Technology Network (OTN) http://download.oracle.com/docs/cd/E13853_01/index.htm.
- RTD 3.0 introduced dynamic choices feature which enables RTD to directly

access offer data in Siebel. Siebel no longer has to synchronize the offer data to RTD as it did in previous releases.

- Oracle RTD uses historical and transactional data from the Siebel OLTP to assist in the offer recommendation process. This data is not stored by the Oracle RTD database (not shown in the figure).
- If Siebel Analytics / Oracle BIEE data is available, it can be utilized as part of the IOG integration via an OLAP JDBC data source.
- Web service requests to Oracle RTD are handled by an Oracle RTD Inline Service, which contains metadata specific to the IOG use-cases, and consists of informants and advisors logic, and definitions of models, rules, data sources, decisions, among other RTD application objects. Oracle RTD Inline Services operate on top of the Oracle RTD platform.

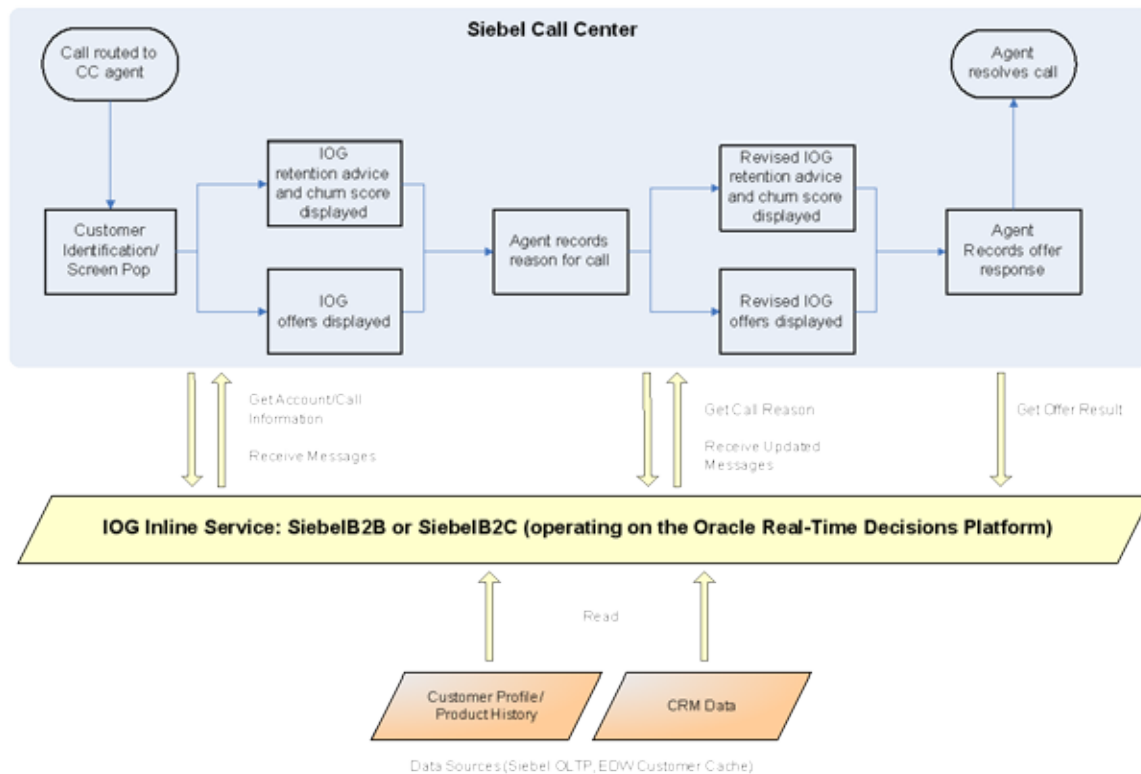


Figure 2. IOG incorporated into business workflow.

Figure 2 shows IOG incorporated into the Siebel Call Center workflow. In the Account or Contact view, after the customer is identified, web service calls are made to Oracle RTD to first submit contextual information about the customer and then to request offers and other information, i.e. churn score of the customer. Optionally, the call center agent may record reasons for the call or other activities, which then triggers additional web service calls to Oracle RTD in order to update the offers displayed. Finally, the agent records the response to the offers displayed which triggers a final web service call to Oracle RTD, in order to close the loop.

2.2 IOG installation and configuration

The installation and configuration of IOG consists of two components:

- Configuration of the IOG Inline Service on an Oracle RTD platform, and
- Configuration of a Siebel Enterprise Server to enable IOG features

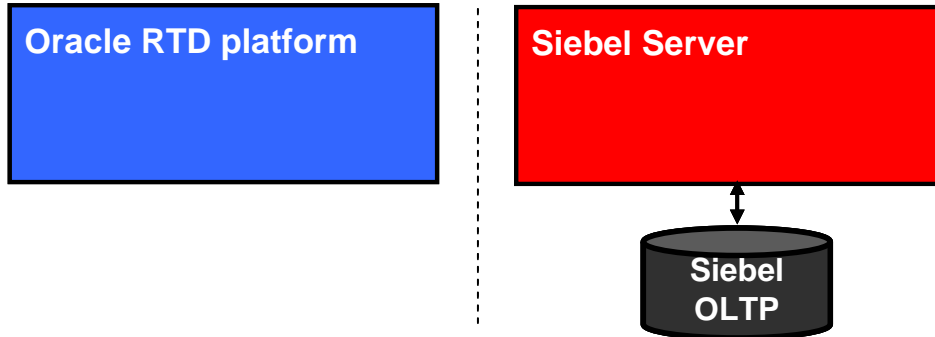


Figure 3. Oracle RTD platform and Siebel Enterprise Server before IOG is configured

In Figure 3, an Oracle RTD platform and a Siebel Enterprise Server are shown as separate entities. Before installing IOG, make sure both server systems are installed and operational. For instructions on installing Oracle RTD platform or Siebel Enterprise Server, see their respective documentation.

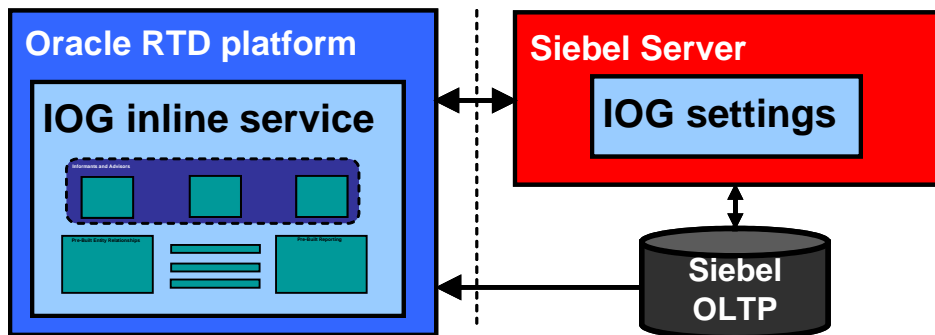


Figure 4. Oracle RTD platform and Siebel Enterprise Server after IOG is configured.

In Figure 4, IOG has been enabled by 1) installing the IOG Inline Service on the Oracle RTD platform, and 2) configuring IOG settings in the Siebel Enterprise Server. Instructions for these two components are given in different chapters, as described in the following table.

Oracle RTD platform version	Siebel Enterprise Server version	IOG configuration in Oracle RTD platform	IOG configuration in Siebel Enterprise Server
3.0	8.1	Chapter 3. Configuring IOG Inline Service in Oracle RTD Platform	Chapter 4. Configuring IOG in Siebel Enterprise Server 8.1

2.3 Inbound Offer Management

After IOG has been installed and configured, marketing user can:

- Create and manage marketing offers and retention treatments in Siebel Marketing, and make these offer messages available for real-time optimization using the Oracle RTD platform.
- Track drivers of offer response behavior directly through embedded intelligence reports in Siebel Marketing and Call Center.

RTD 3.0 introduced dynamic choices feature which enables RTD to directly access offer data in Siebel. Siebel on longer has to synchronize the offer data to RTD as it did in previous releases.

3 Configuring IOG Inline Service in Oracle RTD Platform

3.1 Install and configure the Oracle RTD Server

As prerequisites, you must have installed one of the Oracle RTD supported J2EE application servers, and you must have successfully deployed the Oracle RTD server on to this application server. Before starting the installation, verify that Oracle RTD is running and that the server logs do not contain any errors.

For information on installing the Oracle RTD Server, please refer to the Oracle RTD platform document *Installation and Administration of Oracle RTD v3.0*, available at <http://www.oracle.com/technology/documentation/rtd.html>.

This chapter describes how to set up the Oracle RTD-specific portion of the overall IOG application. See Chapter 4 - [Configuring IOG in Siebel Enterprise Server 8.1](#) for details on setting up the Siebel Enterprise Server-specific portion of IOG:

3.2 Overview

Configuring the IOG Inline Service in the Oracle RTD platform consists of the following tasks:

- [3.3 - Configuring JDBC Data Sources](#)
- [3.4 - Creating the Entity Views in the Siebel OLTP database](#) on page 12
- [3.5 - Setting up Oracle RTD server Authentication](#) on page 14
- [3.6 - Specifying the list of trusted hosts for Decision Service requests](#) on page 15
- [3.7 - Deploying the Siebel IOG Inline Service to the Oracle RTD Server](#) on page 16
- [3.8 - Configuring RTD Dynamic Choices](#) on page 19
- [3.9 - Testing the Deployed Inline Service via Web Service Calls](#) on page 26

3.3 Configuring JDBC Data Sources

Oracle RTD Server uses Java Database Connectivity (JDBC) to access contact and account profile information in the Siebel OLTP and OLAP databases. To run the IOG application for Siebel Call Center, you must use an Online Transaction Processing (OLTP) data source.

The Online Analytical Processing (OLAP) data source is needed only if you use Siebel Analytics / Oracle Business Intelligence Server. For instructions on setting up the OLAP data source, see Chapter 6.2 - [Utilizing analytics data via OLAP tables](#).

The following JDBC data sources are required to run the application:

Data Source name	Type	Description
SDDS	System	Stores the deployed projects, run-time sessions, and model learnings. This data source is required. Typically, this data source is created when you install the Oracle RTD Server.
SIEBEL_OLTP	Source	Populates the customer, account, and asset entities

Data Source name	Type	Description
		from the Siebel OLTP. This data source is required.
SIEBEL_OLAP	Source	Uses prepopulated OLAP tables to populate the service request, order history, and financial account entities. This data source is required only if you use Siebel Analytics / Oracle Business Intelligence Server.

The **SDDS** JDBC data source should have already been configured as part of the Oracle RTD platform setup. To create the **SIEBEL_OLTP** data source in the application server and RTD resource references to it, refer to the Oracle RTD platform document *Installation and Administration of Oracle RTD*, with section numbers as specified by the following table.

Application server	Section #	Notes
Oracle AS 10.1.3	Section 7.1	In section 7.1.2, steps 11a and 11b: set the data source and JNDI names to SIEBEL_OLTP
IBM WebSphere 6.1	Section 7.2	In section 7.2.1, steps 13b and 13c: set the data source and JNDI names to SIEBEL_OLTP
BEA WebLogic 9.2	Section 7.3	In section 7.3.2, steps 4a and 4b: set the data source and JNDI names to SIEBEL_OLTP

Restart the application server after the SIEBEL_OLTP JDBC data source has been added.

3.4 Creating the Entity Views in the Siebel OLTP database

At runtime, Oracle RTD Server accesses views defined in the SIEBEL_OLTP database to populate the entities in the SiebelB2C or SiebelB2B inline services. The login script that executes the Data Definition Language (DDL) statements to create the views must have CREATE VIEW privileges in the database. Only create the views for the inline service that you want to install. For example, if you want to install a B2B inline service, include only B2B views. The following table shows the names and functions of the three views for each application type.

Vertical	View Name	Description
B2B	<i>RTD_CME_ACCOUNT</i>	Represents the wireless customer. Corresponds to the <i>Account</i> entity. Contains attributes demographic and geographic profile information.
	<i>RTD_CME_ASSETS</i>	Represents the plans, devices and options associated to an account. Corresponds to the <i>Assets</i> entity.
	<i>RTD_CME_AGENT</i>	Represents the call center agent who answers the call. Corresponds to the <i>Contextual Call Agent</i> entity. Contains attributes such as the agent tenure and skill set.
	<i>RTD_FINS_OFFER</i>	Represents the dynamic choice data source. Used to fill the <i>Offer Choice</i> entity.

Vertical	View Name	Description
B2C	<i>RTD_FINS_CONTACT</i>	Represents the customer. Used to fill the <i>Customer</i> entity. Contains attributes such as Profile, Credit Score, Investment Experience, and Annual Income.
	<i>RTD_FINS_ASSET</i>	Represents the financial account held by a customer. Used to fill the <i>Financial Accounts</i> entity.
	<i>RTD_FINS_AGENT</i>	Represents the call center agent who answers the call. Corresponds to the <i>Contextual Call Agent</i> entity. Contains attributes such as the agent tenure and skill set.
	<i>RTD_FINS_OFFER</i>	Represents the dynamic choice data source. Used to fill the <i>Offer Choice</i> entity.

To create the view, follow these steps:

- 1) Open the text file which contains the statements that create views specific to your OLTP database type and Siebel Enterprise application type. The file can be found in the Oracle RTD Applications directory

\software\Oracle Real-Time Decisions for Siebel Intelligent Offer Generation\

with filename as shown in the following table:

Application Type	Siebel Vertical	Filename
B2B	Siebel Business Applications	\<SiebelVer>\B2B\SQL_HOR_ORACLE.txt \<SiebelVer>\B2B\SQL_HOR_MSSQL.txt \<SiebelVer>\B2B\SQL_HOR_DB2.txt
B2B	Siebel Industry Applications	\<SiebelVer>\B2B\SQL_SIA_ORACLE.txt \<SiebelVer>\B2B\SQL_SIA_MSSQL.txt \<SiebelVer>\B2B\SQL_SIA_DB2.txt
B2C	Siebel Business Applications	\<SiebelVer>\B2C\SQL_HOR_ORACLE.txt \<SiebelVer>\B2C\SQL_HOR_MSSQL.txt \<SiebelVer>\B2C\SQL_HOR_DB2.txt
B2C	Siebel Industry Applications	\<SiebelVer>\B2C\SQL_SIA_ORACLE.txt \<SiebelVer>\B2C\SQL_SIA_MSSQL.txt \<SiebelVer>\B2C\SQL_SIA_DB2.txt

where <SiebelVer> is the Siebel Enterprise Server version, that is, Siebel8.1.

- 2) Find the CREATE VIEW statement in the text file for the view that you want to create.
- 3) Execute the CREATE VIEW statement on your database.
- 4) Test that the CREATE VIEW statement worked correctly. For example, execute a statement similar to the following:

```
SELECT * FROM RTD_FINS_CONTACT
```

Make sure that you replace RTD_FINS_CONTACT with the correct view name.

- 5) Repeat Step 2 to Step 4 for each view that you want to create.

3.5 Setting up Oracle RTD server Authentication

There are three options for Oracle RTD server authentication: Windows, Platform, and Siebel Object Manager. This section describes the steps necessary to set up Platform or Siebel Object Manager authentication.

For Platform authentication, you must create groups and users which will be stored in the Oracle RTD database. For Siebel Object Manager authentication, the users and groups (responsibilities) are created in and reside on the Siebel Enterprise Server, and are configured with appropriate permissions within Oracle RTD server.

After you have configured an Oracle RTD authentication method, update the Symbolic URLs for username and password within Siebel.

3.5.1 Setting up Oracle RTD Platform (DB) Authentication

Configuring Oracle RTD Platform authentication consists of:

- 1) Changing the authentication provider class to DBAuthenticator.
- 2) Creating groups and users.
- 3) Assigning Oracle RTD realm permissions (numeric values of 0 to 6) to these groups.
- 4) Enabling authentication.

Follow the instructions in sections 6.3 and 6.5 of the Oracle RTD platform document *Installation and Administration of Oracle RTD*.

3.5.2 Setting up Siebel Object Manager Authentication

Oracle RTD server is able to authenticate and authorize using the security adapters configured in the Siebel Object Manager. When authenticating a new user, Oracle RTD attempts a login to a specified Siebel Object Manager. If successfully authenticated, Oracle RTD accesses the Responsibility business component to determine the authorization groups. These groups correspond to the groups managed in the Oracle RTD server.

Configuring Siebel Object Manager authentication consists of the following tasks:

- 1) Changing the authentication provider class to SiebelAuthenticator.
- 1) Assigning Oracle RTD realm permissions (numeric values of 0 to 6) to existing Siebel responsibilities and/or users.
- 2) Enabling authentication.

Follow the instructions in sections 6.4 and 6.5 of the Oracle RTD platform document *Installation and Administration of Oracle RTD*.

In section 6.4.2, step 8 of the Oracle RTD platform document *Installation and Administration of Oracle RTD*, the format of the Siebel Object Manager string (SiebelConnectString) should be as follows:

See the Siebel 8.1 Bookshelf for connect string formats if you are implementing IOG with Siebel 8.1.

Configuring access to the Siebel OM JAR files within the J2EE application server.

For OC4J, read the Oracle RTD platform document *Installation and Administration of Oracle RTD*, section 6.4.1.1.

For WebSphere, read the Oracle RTD platform document *Installation and Administration of Oracle RTD*, section 6.4.1.2.

For WebLogic, read the Oracle RTD platform document *Installation and Administration of Oracle RTD*, section 6.4.1.3.

To test that you have setup Siebel Object Manager, access the standalone Oracle RTD Decision Center UI, and verify that you can log in using a Siebel user name and password.

3.6 Specifying the list of trusted hosts for Decision Service requests

If both the Siebel Enterprise Server and the Oracle RTD server are installed on the same physical machine, this section can be bypassed. If the servers reside on different machines (or belong under different IP addresses), the IP address of the Siebel Enterprise Server must be added to a list of trusted hosts for (Oracle RTD) Decision Service requests.

Similarly, if Oracle RTD Decision Studio (or Oracle RTD Load Generator) is located on a different machine from that of the Oracle RTD Server, and you want to send sample requests using the Test View in Decision Studio, then the IP address of the machine that hosts Decision Studio and/or Load Generator must also be added to the list of trusted hosts. The list is stored within the Oracle RTD server and maintained through Oracle RTD's JMX administration console (JConsole), as described next.

To specify a list of trusted hosts for Oracle RTD Decision Service requests:

- 1) If you are using OC4J or WebLogic, open JConsole by running `JAVA_HOME/bin/jconsole.exe`. If you are using WebSphere, run the batch script you created during JConsole configuration. See the Oracle RTD platform document *Installation and Administration of Oracle RTD* for more information about accessing JConsole.
- 2) Click the **Remote** tab. Then, enter the appropriate port number (typically 12345) and the administrator credentials you created during installation and click **Connect**.
- 3) Click the **MBean** tab, then navigate to the **OracleRTD > SDClusterPropertyManager > Cluster** MBean and ensure the **RestrictDSClients** attribute is set to `true`. This attribute ensures that the Decision Service only accepts requests from its own host, or from the list of hosts identified in the **TrustedDSClients** attribute.
- 4) In the **OracleRTD > SDClusterPropertyManager > Cluster** MBean, update the **TrustedDSClients** attribute to include a semicolon-separated list of IP addresses of

the hosts from which you want Decision Service to accept requests. You must specify IP addresses; do not specify host names.

3.7 Deploying the Siebel IOG Inline Service to the Oracle RTD Server

3.7.1 Importing a Siebel IOG Inline Service into Oracle RTD Decision Studio

There are two IOG Inline Services for each of the Siebel Enterprise Server versions: SiebelB2B and SiebelB2C. The Inline Services are packaged in zip files which contain XML and Java files organized in a directory structure. To load, view, and edit the Inline Service, use the Oracle RTD client tool Decision Studio.

To set up the IOG Inline Service, you must import the Inline Service project into Decision Studio, as follows:

- 1) Create a directory for your Inline Services, for example, C:\RTD_ILS.

This directory will be referred to as *RTD_ILS_HOME* in this documentation.

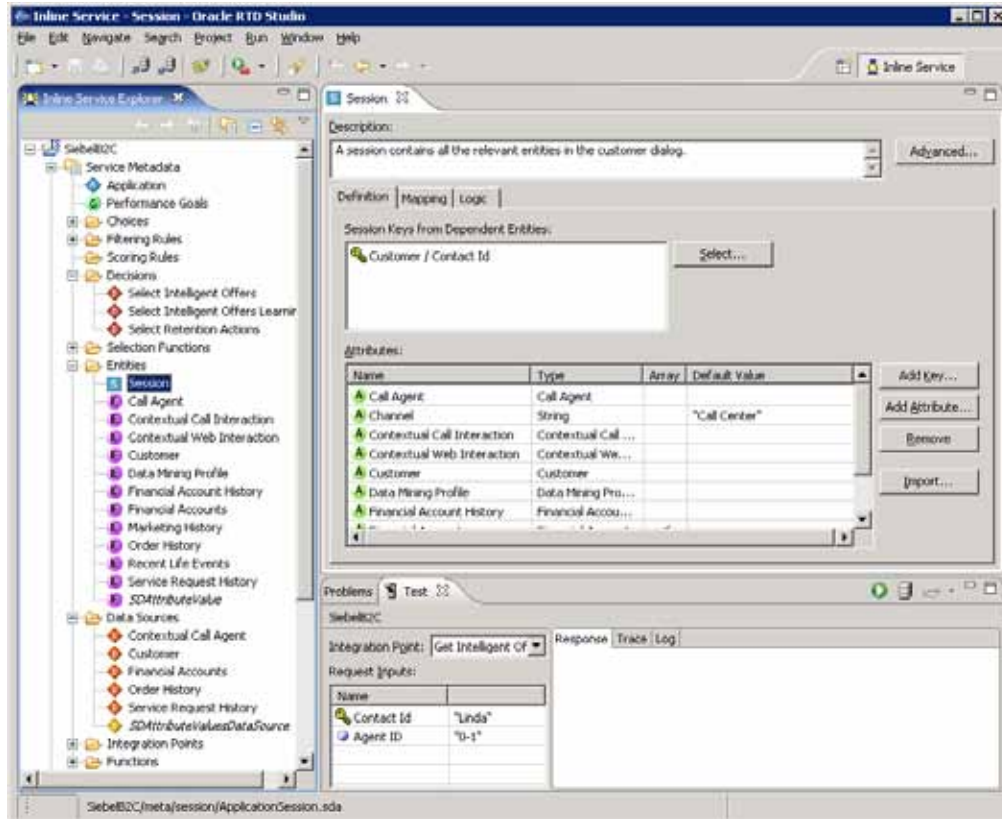
- 2) From the Oracle RTD Applications media pack directory,

\software\Oracle Real-Time Decisions for Siebel Intelligent Offer Generation\

choose the Siebel Enterprise Server version (Siebel8.1), the application type (B2B or B2C), and unzip the application zip file, *SiebelB2B.zip* or *SiebelB2C.zip* into *RTD_ILS_HOME*.

This creates the directory *SiebelB2B* or *SiebelB2C* under *RTD_ILS_HOME*.

- 3) In Decision Studio, select File > Import.
- 4) From the Import window, select **Existing Projects into Workspace**.
- 5) If not already visible in the Projects area of the Import Projects dialog box, click the Browse button, and locate *SiebelB2B* or *SiebelB2C* under *RTD_ILS_HOME*, for example C:\RTD_ILS\SiebelB2B.
- 6) Select the directory *SiebelB2B* or *SiebelB2C* in the file dialog box and click Ok.
- 7) Confirm that the *SiebelB2B* or *SiebelB2C* Inline Service project name appears in the project window and is checked True.
- 8) Click the Finish button to load the project into Decision Studio.
- 9) Click Finish. After the project/Inline Service is imported, Decision Studio displays a complete application tree in the left pane, including Entities, Decisions, Data Sources, etc. similar to what is shown in the following screenshot.

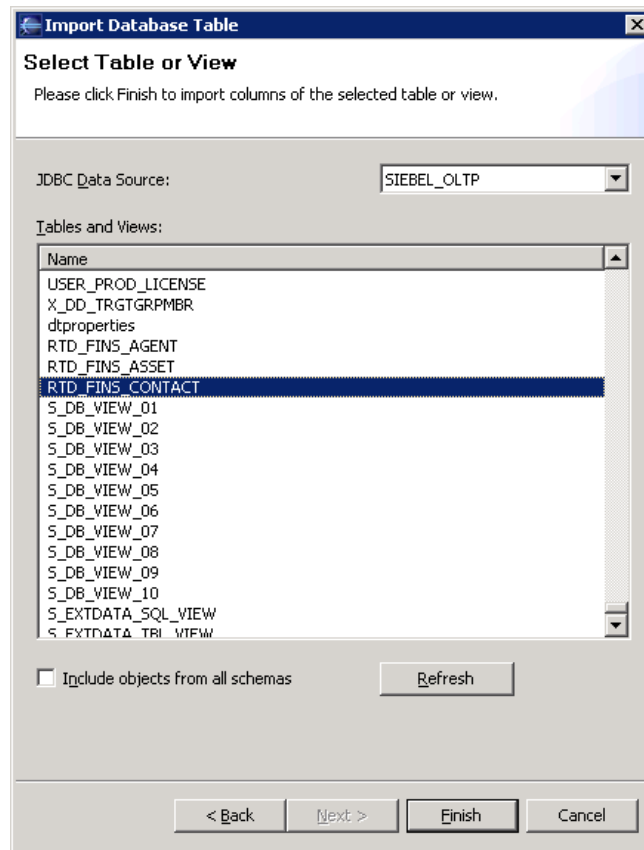


The Inline Service should now be successfully loaded into Oracle RTD Decision Studio.

3.7.2 Verifying OLTP data source is accessible by Oracle RTD

The next step is to verify that the SIEBEL_OLTP data source is accessible by the Oracle RTD server. This verifies that you have setup the Oracle RTD data source properly as described in 3.3 *Configuring JDBC Data Sources*.


- 1) Expand the data sources folder and double click on Customer if your inline service is SiebelB2C and Account if your inline service is SiebelB2B, and then click **Import**.
- 2) Select the running Oracle RTD server and click Next. If the SIEBEL_OLTP JDBC data source was configured correctly, the application displays a list of tables and views with the correct view highlighted for the current data source as shown below:

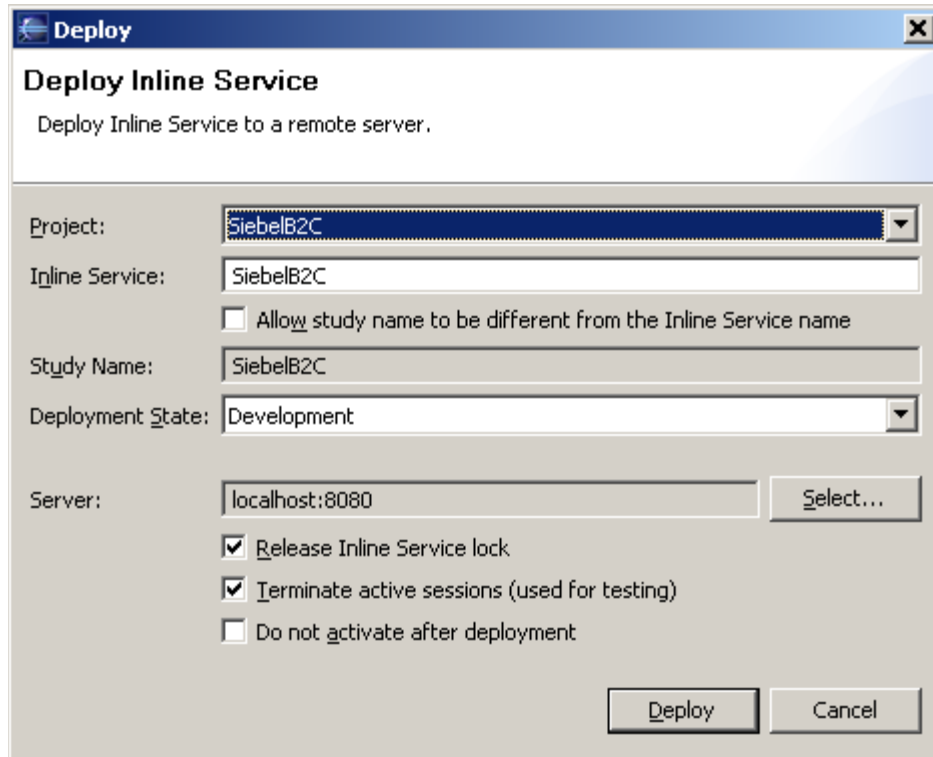


- 3) If you get an error message instead, then double check the JDBC data source configuration as described in *3.3 Configuring JDBC Data Sources*. If you see the list of tables and views properly, click the 'Cancel' button since there is no need to re-import the data source.

3.7.3 Deploying IOG Inline Service to Oracle RTD server

The SiebelB2C or SiebelB2B Inline Service is now ready to be deployed to the Oracle RTD server.

§ Click Deploy, , located in the toolbar. The following dialog appears:



Be sure that the correct SiebelB2C or SiebelB2B project is selected in the project dropdown. The deployment state should be set to Development. Click “Deploy” and if the Inline Service is successfully deployed, a “SiebelB2C deployed successfully” message will be displayed in the bottom status bar.

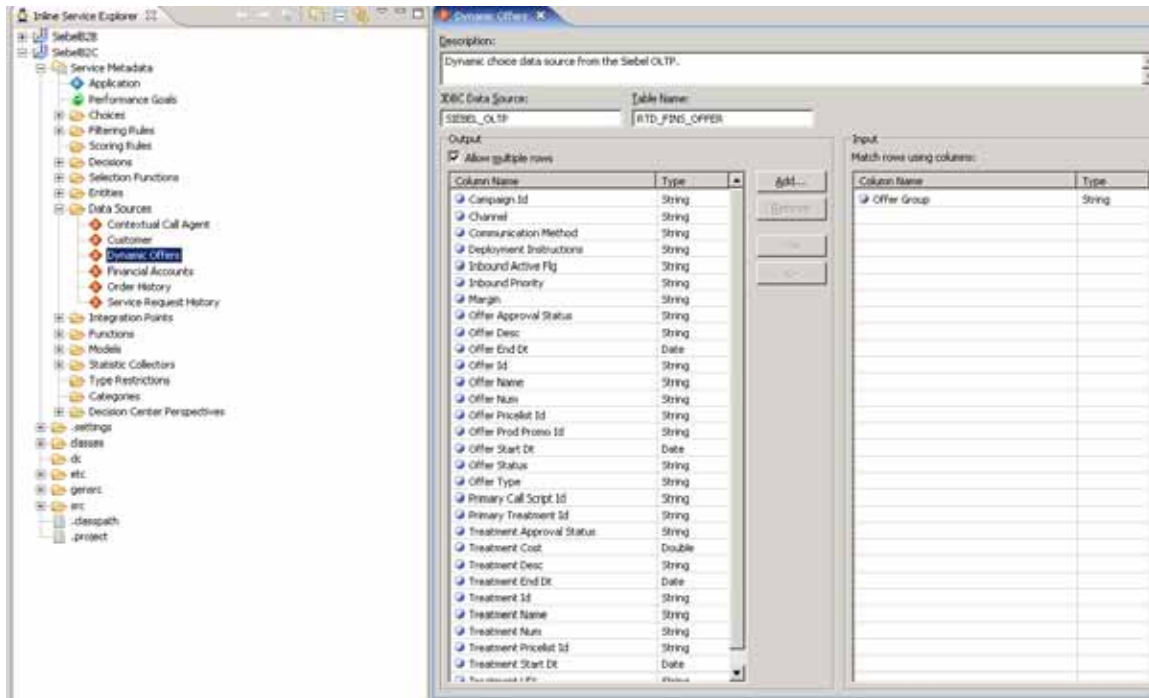
Before the Oracle RTD server can accept requests, the list of trusted hosts needs to be updated to include IP addresses where requests will originate. The instructions are given in the next section.

3.8 Configuring RTD Dynamic Choices

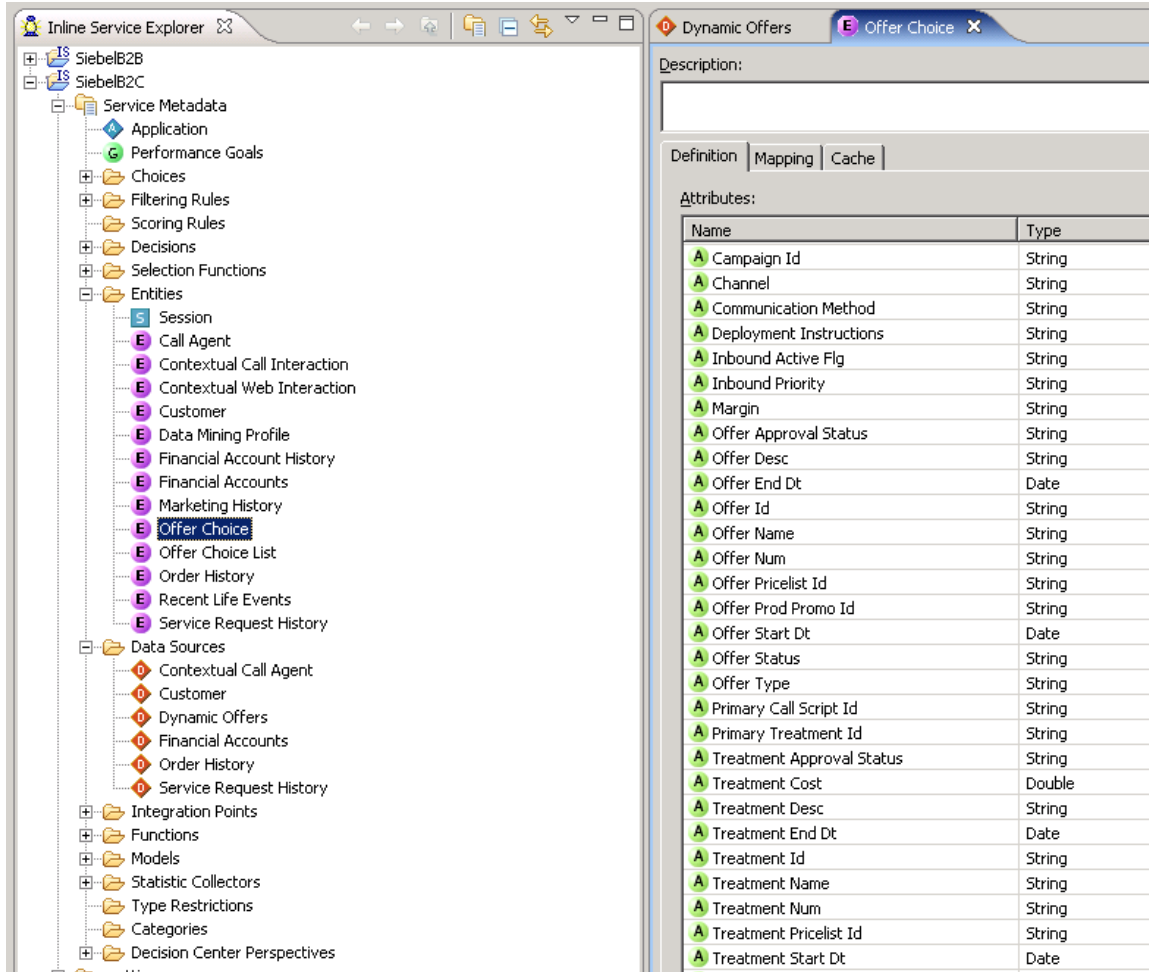
The dynamic choices feature enables RTD to access the Siebel database directly so that IOG no longer needs to synchronize offer data from Siebel to RTD.

The instructions below show how to map RTD entity to the Siebel data source.

- 1) Run the create view script to create RTD_FINS_OFFER view in SIEBEL_OLTP database.
- 2) Open existing SiebelB2C or SiebelB2B inline service project in RTD Studio.
- 3) Create Dynamic Offers Data Source:
 - a. JDBC Data Source: SIEBEL_OLTP
 - b. Table Name: RTD_FINS_OFFER
 - c. Check Allow Multiple Rows.
 - d. Move Offer Group to Input column area.

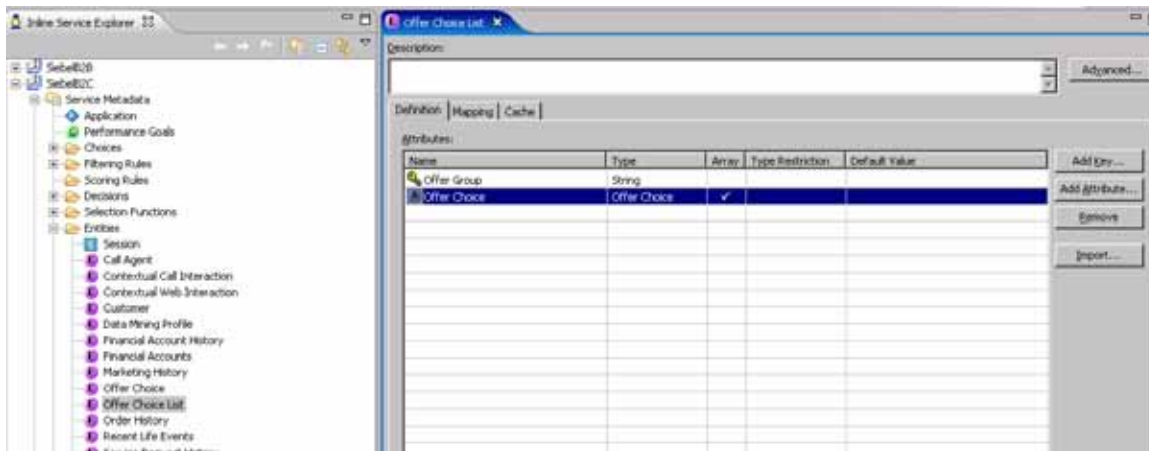


- 4) Create Offer Choice Entity:
 - a. Click Import, and then select Dynamic Offers in the Data Source selection dialog box.
 - b. Uncheck Build Data Mappings for the selected data source, and click OK.



5) Create Offer Choice List Entity:

- a. Create Offer Group as key attribute by clicking Add Key.
- b. Create an Attribute with name Offer Choice, and then mark it as an array.



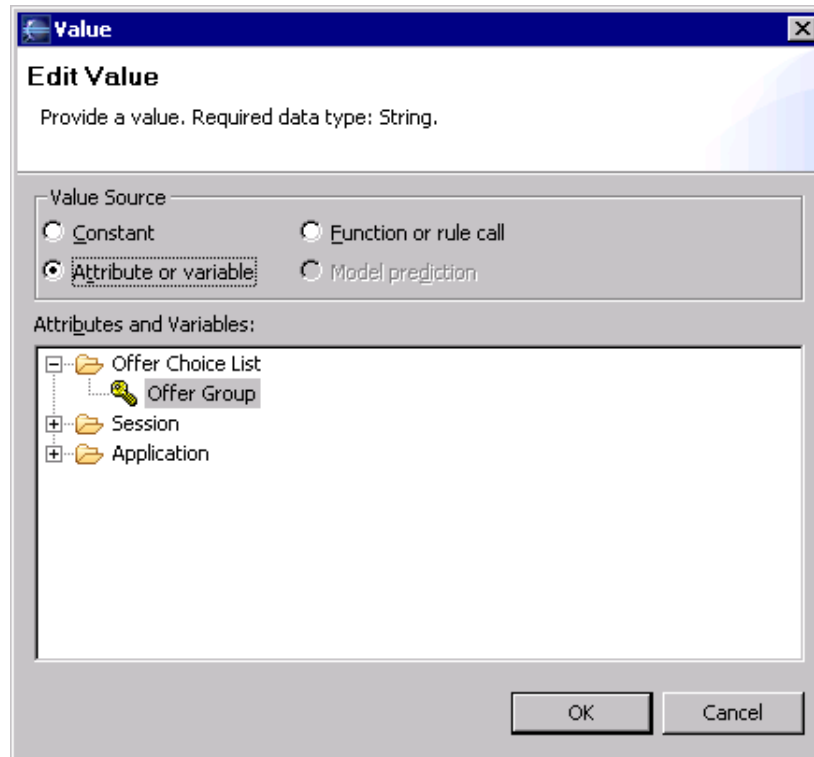
- c. Click the Mapping tab, and map each attribute within the entity-type Attribute to the appropriate column in data source.

The screenshot shows the Oracle Service Explorer interface. On the left, a tree view displays the project structure, with 'Offer Choice List' selected under 'Data Sources'. The main window shows the 'Offer Choice List' entity details. The 'Attributes' section is expanded, showing a list of attributes with their types and sources. Below this, the 'Data Source Input Values' section is visible, showing a table with columns for Data Source, Input Column, Type, and Input Value.

Name	Type	Array	Default Value	Source
Offer Choice	Offer Choice	<input checked="" type="checkbox"/>		Dynamic Offers / Campaign Id
Campaign Id	String			Dynamic Offers / Channel
Channel	String			Dynamic Offers / Communication Method
Communication Method	String			Dynamic Offers / Deployment Instructions
Deployment Instructions	String			Dynamic Offers / Inbound Active Flg
Inbound Active Flg	String			Dynamic Offers / Inbound Priority
Inbound Priority	String			Dynamic Offers / Margin
Margin	String			Dynamic Offers / Offer Approval Status
Offer Approval Status	String			Dynamic Offers / Offer Desc
Offer Desc	String			Dynamic Offers / Offer End Dt
Offer End Dt	Date			Dynamic Offers / Offer Id
Offer Id	String			Dynamic Offers / Offer Name
Offer Name	String			Dynamic Offers / Offer Num
Offer Num	String			Dynamic Offers / Offer Pricelist Id
Offer Pricelist Id	String			Dynamic Offers / Offer Prod Promo Id
Offer Prod Promo Id	String			Dynamic Offers / Offer Start Dt
Offer Start Dt	Date			Dynamic Offers / Offer Status
Offer Status	String			Dynamic Offers / Offer Type
Offer Type	String			Dynamic Offers / Primary Call Script Id
Primary Call Script Id	String			Dynamic Offers / Primary Treatment Id
Primary Treatment Id	String			Dynamic Offers / Treatment Approval Status
Treatment Approval Status	String			Dynamic Offers / Treatment Cost
Treatment Cost	Double			Dynamic Offers / Treatment Desc
Treatment Desc	String			Dynamic Offers / Treatment End Dt
Treatment End Dt	Date			Dynamic Offers / Treatment Id
Treatment Id	String			Dynamic Offers / Treatment Name
Treatment Name	String			Dynamic Offers / Treatment Status
Treatment Status	String			

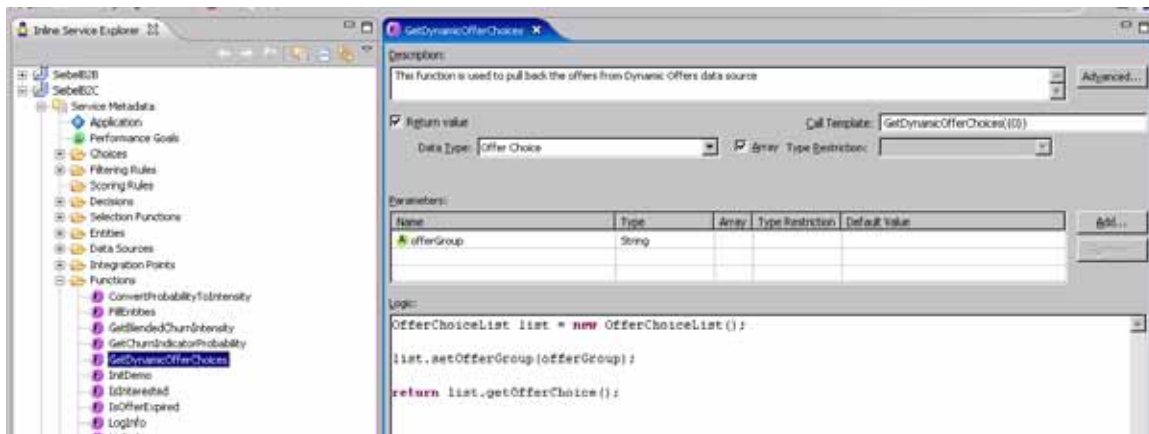
Data Source	Input Column	Type	Input Value
Dynamic Offers	Offer Group	String	Offer Group

- d. In the Data Source Input Values region, for the Input Value of the Data Source, select Offer Group.



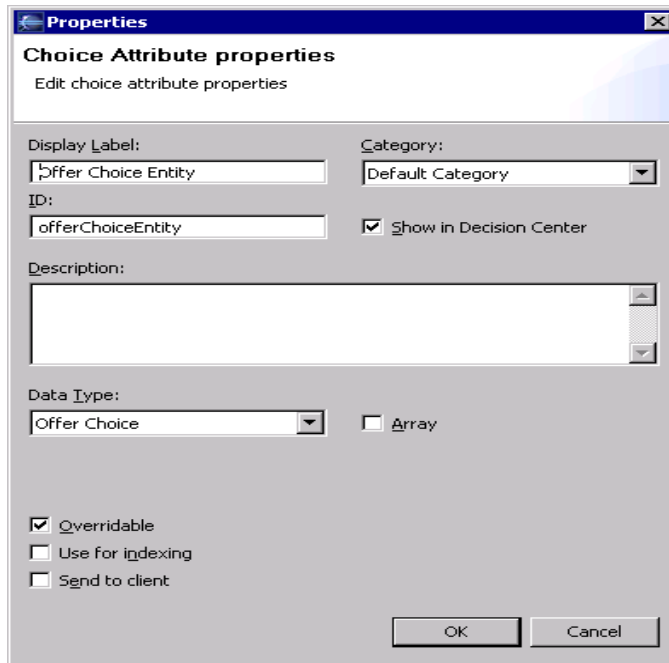
- 6) Create Function: GetDynamicOfferChoices.
 - a. Set Return Value as checked.
 - b. Set Array as checked.
 - c. For the Data Type, select Offer Choice .
 - d. Add offerGroup In Parameters area.
 - e. In the Logic field, enter following code:

```
OfferChoiceList list = new OfferChoiceList();
list.setOfferGroup(offerGroup);
return list.getOfferChoice();
```

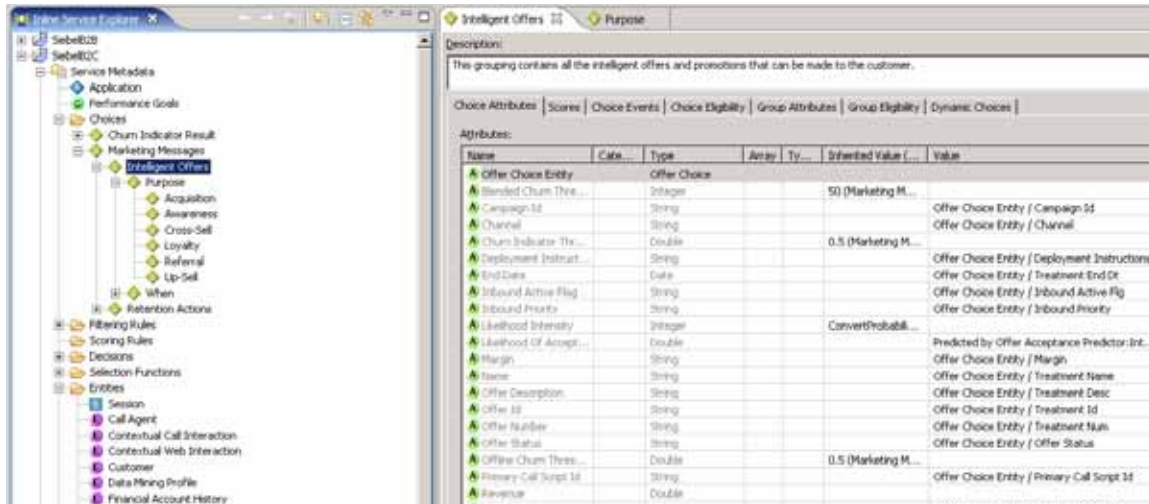


- 7) Create Choice Attribute: Offer Choice Entity.
 - a. Select Intelligent Offers Choice Group.

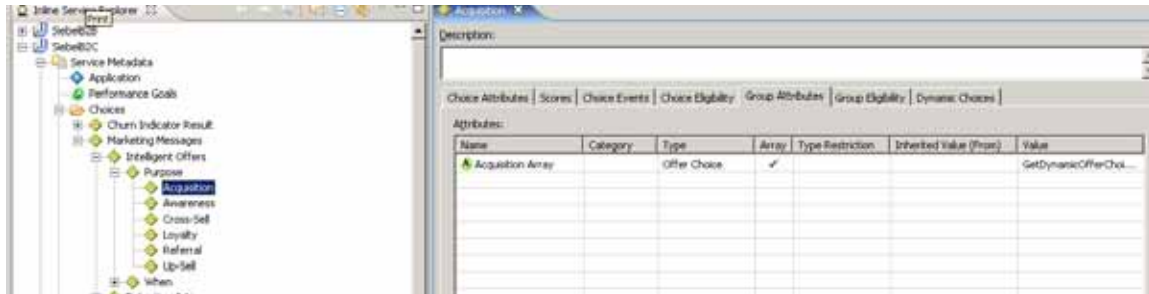
- b. Click the Choice Attributes tab, Add Attribute: Offer Choice Entity.



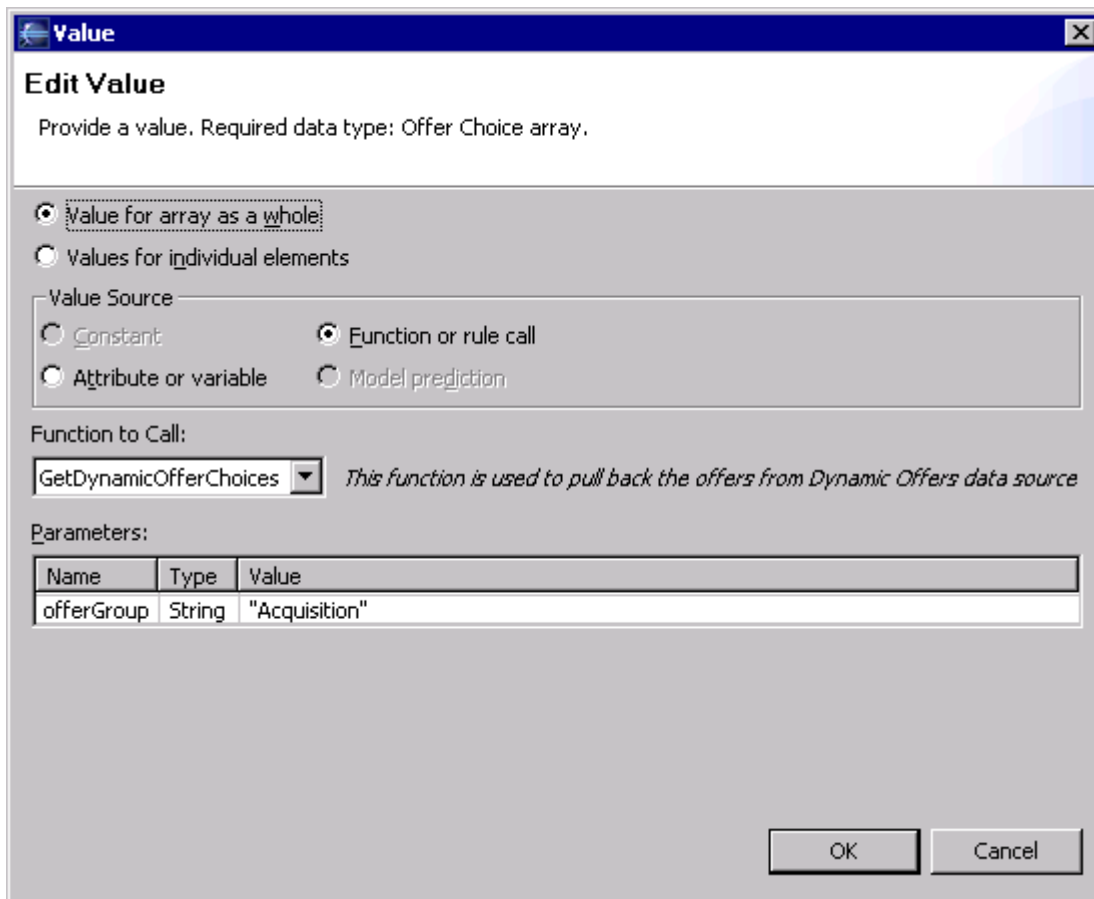
- c. Map each attribute within Choice Attributes to the appropriate Offer Choice Entity attribute in Value area.



- 8) Create a Group Attribute for each Offer Group, such as Acquisition, Cross-Sell, Up-Sell and so on.
 - a. For each Choice Group Create Group Attribute: offer group Array, such as Acquisition Array, Cross-Sell Array, Up-Sell Array, and so on.

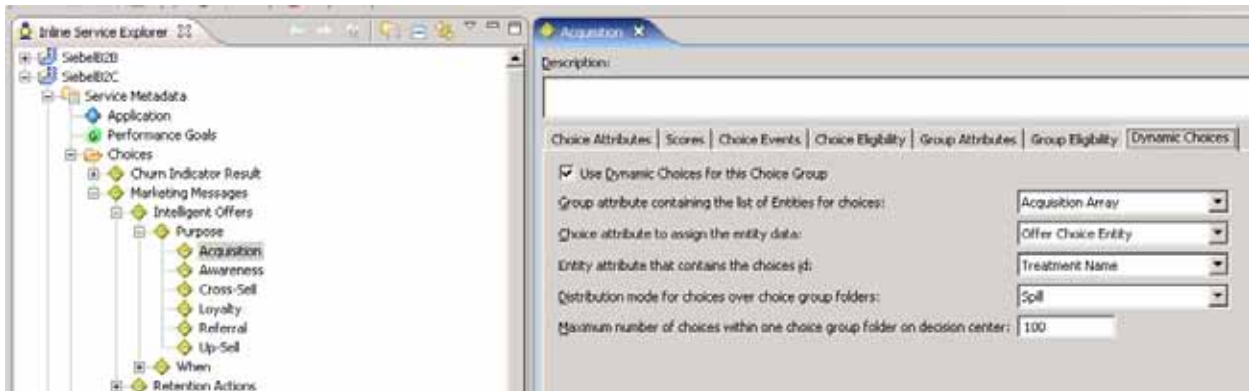


- b. Set Group Attribute value:
 - i. Select the option value for array as a whole.
 - ii. For value source, select Function or rule call, then select GetDynamicOfferChoices function.
 - iii. In Parameters area, set value of Parameter to Offer Group Name i.e. Acquisition or Cross-Sell or Up-Sell, and so on.



- c. Click Dynamic Choices tab.
- d. Select check box option to Use Dynamic Choices for this Choice Group.
- e. For the Group attribute containing the list of Entities for choices, select offer group Array (created at step 8.a).
- f. For the Choice attribute to assign the entity data, select Offer Choice Entity.

- g. For the Entity attribute that contains the choice ID, select Treatment Name.
- h. For the Distribution mode for choices over choice group folder, select spill or even.
- i. Select Maximum number of choices within one choice group folder on decision center to any desired value, such as 100.



- 9) Save All, Build and Deploy the Inline Service Project.
- 10) Click Test, For Integration Point select Get Intelligent Offers.
 - a. Add Request Inputs:
 - i. For Contact ID enter value: Linda
 - ii. For Agent ID enter value: 0-1
 - b. Click Execute Request button.
 - c. Verify the dynamic choice results under Response tab.
 - d. Verify the logs in under Log tab.

3.9 Testing the Deployed Inline Service via Web Service Calls


When the data source connectivity has been verified, and the Inline Service has been deployed, the next step is to do a dry run of the application using the Oracle RTD Decision Studio's built in Test View. To test the **Identify Customer** informant:

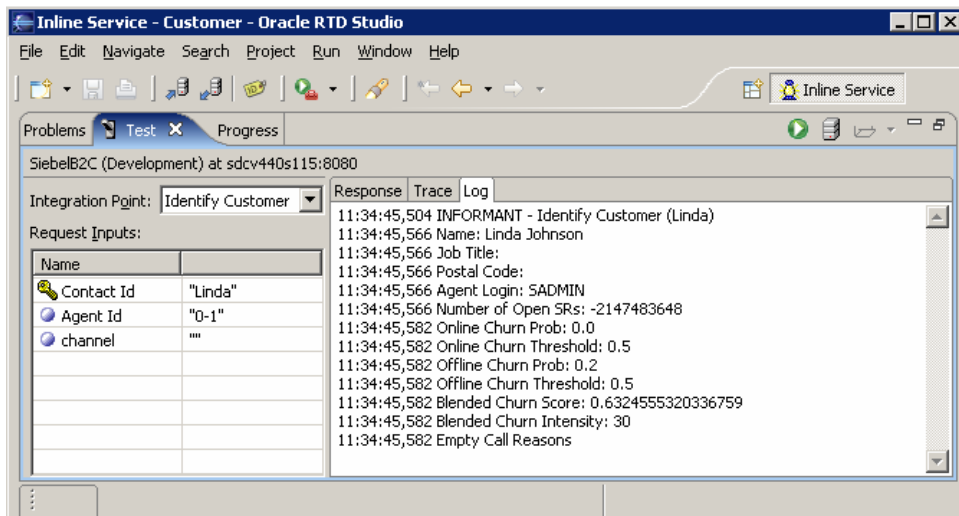
- 1) Navigate to the Test tab on the lower quadrant of Decision Studio. If this tab is hidden, from the Decision Studio's main menu, select Window > Show View > Test to expose it.
- 2) Pick **Identify Customer** from the Integration Point dropdown. If you deployed SiebelB2C, then set:

contactId = Linda
 agentId = 0-1
 channel = web

If you deployed SiebelB2B, then set:

accountId = Shirley
 agentId = 0-1
 channel = web

- 3) Click the  button and a web services call is made to the Oracle RTD Server. The Log tab displays the output results. Verify that output is similar to the following example.



If the test fails, examine the exception stack trace from either the server command window (if one exists), or the Oracle RTD server.log file. This log contains information useful in debugging the problem. For typical locations of the Oracle RTD server.log file, see section 6.1.2 of the Oracle RTD platform document *Installation and Administration of Oracle RTD*.

4 Configuring IOG in Siebel Enterprise Server 8.1

This chapter describes how marketing managers can use and configure the Intelligent Offer Generation (IOG) solution. The IOG solution uses the Oracle Real-Time Decisions (RTD) platform.

4.1 About Oracle Real-Time Decisions for Call Center

The IOG solution provides intelligent offer recommendation and retention management features to call center agents who receive inbound customer communications. This solution uses the decisioning capabilities of the Oracle RTD platform.

Oracle RTD combines predictive analytics technology with dynamic eligibility rules and scoring rules to maximize the value from inbound customer interactions. The criteria for rules can be based on a variety of attributes, for example, the age of a customer, the salary of a customer, recent life events, and so on. Oracle RTD can anticipate customer needs in real time and adapt each interaction to the needs of the customer. The decision logic of Oracle RTD for Call Center and the integration points with Siebel Enterprise are implemented as an inline service. The inline service runs on an Oracle RTD server.

While you are in a view, click Reports to see a list of the available preconfigured reports for the data in that view. From the list you can access individual reports. For more information about reports, see *Siebel Reports Administration Guide*.

4.2 Scenario for Intelligent Offer Generation in Call Center

This topic gives one use case of how IOG could be used. You might use IOG differently, depending on your business model.

A marketing manager has been asked to use the call center to drive proactive retention management and increase revenues from cross- and up-sell offers. The manager decides to implement the Intelligent Offer Generation Application for Siebel Call Center. The marketing manager asks the marketing or system administrator to configure the Oracle RTD server. After the system administrator configures IOG, the marketing manager uses Siebel Marketing to create a marketing campaign specifically for inbound calls.

Then, the marketing manager creates offers within the campaign. Each offer can have one or more associated treatments. A treatment is a form of an offer that is specific to a particular channel. For example, you might have a phone treatment of an offer. After Oracle RTD is synchronized with Siebel Enterprise Server, choices in Oracle RTD correspond to treatments in Siebel Enterprise Server. After the marketing manager creates offers within the campaign, the manager can then change some settings for the treatments associated with the offer. For example, the manager can add a profit margin figure associated with an offer.

The marketing manager can then create rules, or groups of rules, which determine the eligibility of customers for an offer. For example, the manager might set up rules so that an offer is eligible only to customers over the age of 21 years, with a salary of more than \$35,000.

The marketing manager can now add or update offer information, such as the profit margin associated with the offer. Agents can then work with views in Siebel Call Center as they receive calls from customers.

4.3 Process of Configuring the IOG Application for Siebel Call Center

To configure IOG, perform the following tasks:

- 1) [Checking prerequisites for IOG](#)
- 2) [Configuring IOG with Siebel Views for Siebel Call Center](#)
- 3) [Setting Siebel Enterprise Server Parameters for IOG](#) on page 31
- 4) [Configuring Siebel Run-Time Events for Oracle RTD for Siebel Call Center](#) on page 32
- 5) [Customizing Call Reasons for Oracle RTD for Siebel Call Center](#) on page 32.

4.3.1 Checking prerequisites for IOG

Before you configure IOG, you must complete the following actions:

- Install the latest version of Oracle RTD platform.
- Install and deploy the SiebelB2B or SiebelB2C inline service on the Oracle RTD server. Obtain the inline service files from the Oracle RTD Applications media pack, folder "Oracle Real-Time Decisions for Siebel Intelligent Offer Generation". You can integrate only one inline service with Siebel Enterprise Server at a time, so you must choose your inline service before you configure IOG.

This task is a step in [Process of Configuring the IOG Application for Siebel Call Center](#).

4.3.2 Configuring IOG with Siebel Views for Siebel Call Center

Siebel Portal Framework enables IOG to integrate effectively with Siebel views for Siebel Call Center. Seed data for this integration is provided with the product, but you must make some updates to see Oracle RTD reports in Siebel views.

This task is a step in [Process of Configuring the IOG Application for Siebel Call Center](#).

To configure IOG with Siebel views

- 1) Navigate to the Administration - Integration screen, then the WI Symbolic URL List view.
- 2) Select Host Administration from the visibility filter.
- 3) Query the list for a virtual name called RTD_SERVER.
- 4) In the record with the virtual name RTD_SERVER, type the host name of the Oracle RTD server in the Name field. For example, you might use a hostname like services.corp.example.com:8080.
- 5) Select Symbolic URL Administration from the visibility filter.

- 6) Query the list for URLs that contain the string RTD.
- 7) For each record that has a URL that contains RTD, use the Symbolic URL Arguments applet to update the Argument Value of the argument to the appropriate inline service. For example, update the argument value of the DecisionCenter record to SiebelB2C or SiebelB2B.
- 8) If you plan to use Oracle RTD Platform Authentication instead of Siebel Object Manager Authentication, you must update the username and password arguments in the following table:

Name	Type	Value
username	Constant	sdsu
password	Constant	sdsu

4.3.3 Setting Siebel Enterprise Server Parameters for IOG

You must configure particular parameters in Siebel Enterprise Server so that Advisor or Informant functionality perform correctly.

This task is a step in [Process of Configuring the IOG Application for Siebel Call Center](#) on page 30.

To configure Advisor/Informant

- 1) Navigate to the Administration - Server Configuration screen, Servers list, and then the Parameters view.
- 2) Query the list for parameters that begin with the string RTD.
- 3) Enter values for each of the Oracle RTD parameters, as described in the following table.

Name	Description	Example
RTD Application Name	Type the name of the Oracle RTD inline service. This parameter is used to provide the following functions between Siebel Enterprise Server and Siebel Call Center RTD: Real-time communication, for example, with integration points.	SiebelB2B
RTD SOAP URL	Type the Simple Object Access Protocol (SOAP) URL for the Oracle RTD server. The Oracle RTD Integration Business service uses this to make Advisor and Informant SOAP calls.	http://services.corp.example.com:8080/rtis/sdwp

RTD Session Cookie Name	Type the name of the cookie for the Oracle RTD Server to use to load balance calls for Web services from the Siebel Server. If you do not want to use any hardware- or software-based load balancing tools, use the default value JSESSIONID.	JSESSIONID
RTD WebService Timeout	Type the interval, in milliseconds, after which you want the Oracle RTD Advisor or Informant Web service calls to time out.	3000

4.3.4 Configuring Siebel Run-Time Events for Oracle RTD for Siebel Call Center

Siebel Enterprise Server uses run-time events to communicate data changes from Siebel Object Manager to Oracle RTD for Siebel Call Center. Seed data for run-time events is provided with the product, but the run-time events are not enabled by default. You must enable Oracle RTD-related events for IOG to function correctly.

This task is a step in [Process of Configuring the IOG Application for Siebel Call Center](#) on page 30.

To enable Oracle RTD-related run-time event

- 1) Navigate to the Administration - Runtime Events screen, then the Action Sets view.
- 2) Query the list for the parameters that begin with the string RTD.
- 3) Select the Active check box for all of the records that correspond with the inline service that you use. For example, if you use SiebelB2C inline service, select the Active check box only for the records that are labelled as B2C.
- 4) Choose Reload Runtime Events from the menu. This action clears and reloads the run-time events cache.

4.3.5 Customizing Call Reasons for Oracle RTD for Siebel Call Center

When a customer calls your business, your agents can record the reason for the call. Your agents record the reasons in the Activities applet in the Contact Summary (Oracle RTD) and Account Summary (Oracle RTD) views.

By default, a number of call reasons are included with Siebel Business Applications. You can customize the list of call reasons. You can deactivate the default call reasons, and add call reasons that are relevant to your business.

This task is a step in [Process of Configuring the IOG Application for Siebel Call Center](#) on page 30.

To customize call reasons

- 1) Navigate to the Administration - Data view, then the List of Values screen.

- 2) Query the list for the list-of-value types with the Type of TODO_TYPE and with the Parent LIC of Call - Inbound.
- 3) Deselect the Active check box for any of the call reasons that are not relevant to your business.
- 4) Add any list-of-value types that are specific to your business.

4.4 Process of Setting Up Real-Time Intelligent Offers for Siebel Call Center

To set up real-time intelligent offers and retention treatments, perform the following tasks:

- 1) [Defining the Campaign for IOG for Siebel Call Center](#)
- 2) [Creating and Modifying IOG Offers for Siebel Call Center](#)
- 3) [Enabling IOG Offers for Siebel Call Center](#) on page 33
- 4) [Opening Decision Center for Editing for IOG for Siebel Call Center](#) on page 34
- 5) [Specifying Eligibility for Offers for Oracle RTD for Siebel Call Center](#) on page 34
- 6) [Adding Eligibility Rules](#) on page 35
- 7) [Redeploy the RTD Inline Service if there is Any Change on It](#) on page 36

4.4.1 Defining the Campaign for IOG for Siebel Call Center

In the Siebel Marketing application, offers are presented to customers in the context of marketing campaigns. Before you can implement IOG offers, you must create a campaign in Siebel Marketing. For information about how to create a campaign, see *Siebel Marketing User Guide*.

This task is a step in [Process of Setting Up Real-Time Intelligent Offers for Siebel Call Center](#).

4.4.2 Creating and Modifying IOG Offers for Siebel Call Center

Offers are a way to present incentives on products and services to current and potential customers as part of a campaign. Offers are presented to customers in the context of marketing campaigns.

Before you can implement IOG offers, you must create them within a campaign in Siebel Marketing. Add the treatments that you require to the offers. For more information about offers and treatments, see *Siebel Marketing User Guide*.

This task is a step in [Process of Setting Up Real-Time Intelligent Offers for Siebel Call Center](#).

4.4.3 Enabling IOG Offers for Siebel Call Center

Before you can use an IOG offer, you must change some settings for the treatments associated with the offer.

This task is a step in [Process of Setting Up Real-Time Intelligent Offers for Siebel Call Center](#) on page 33.

To enable an IOG offer

- 1) Navigate to the Offers screen, then the Phone Treatments list, or Offers screen, then the Web Treatments list.
- 2) Select the offer that you want to work with.
- 3) In the Treatments view, drill down on the treatment you want to work with.
Alternatively, create a new treatment, then drill down on the new treatment.
- 4) On the treatment screen, click the Real-Time Details tab.
- 5) Select Enabled from the Stage drop-down list.
- 6) Select the group with which you want to associate the offer from the Offer Group drop-down list.
- 7) Select a campaign with which to associate the offer from the Campaign field, and click the Sync button.

4.4.4 Opening Decision Center for Editing for IOG for Siebel Call Center

Before you use IOG, you can update offer-related information. For example, you can specify offer eligibility rules. You make these changes in the Decision Center, but before you make the changes you must open the Decision Center for edit operations.

This task is a step in [Process of Setting Up Real-Time Intelligent Offers for Siebel Call Center](#) on page 33.

To open Decision Center for edit operations

- 1) Navigate to the Administration - Marketing screen, then the Decision Center view.
- 2) Click the Open button in the Decision Center view.
- 3) Select the inline service with which you want to work, check Open the Inline Service for editing, and click OK.
- 4) Click the Perspective button, select the Design perspective, and then click OK.

4.4.5 Specifying Eligibility for Offers for Oracle RTD for Siebel Call Center

By default, all offers are eligible for all customers. You can set up rules, or groups of rules, to determine the eligibility of customers for an offer.

An *eligibility rule* is a relational expression which, when evaluated, determines the eligibility of a customer for an offer. For example, you can set up rules so that an offer is eligible only to customers over the age of 21 years, with a salary of more than \$35,000. A *rule set* is a group of eligibility rules that you can apply as a unit.

This task is a step in [Process of Setting Up Real-Time Intelligent Offers for Siebel Call Center](#) on page 33.

To specify eligibility rules for an offer

- 1) Navigate to the Administration - Marketing screen, then the Decision Center view.
- 2) Open the Decision Center view for edit operations.

For information about how to open decision center for edit operations, see [Opening Decision Center for Editing for IOG for Siebel Call Center](#) on page 34.

- 3) In the tree structure in the left pane, select the offer that you want to work with in Decision Process, then Marketing Messages.
- 4) In the right pane, click the Definition tab, then click the Eligibility tab.
- 5) To add a rule, click the Add Rule toolbar button in the Eligibility tab view.
For more information about how to add a rule, see [Adding Eligibility Rules](#).
- 6) To add a rule set, click the Add Rule Set toolbar button in the Eligibility tab view.
For more information about how to add a rule set, see [Adding Eligibility Rules](#).
- 7) Click the Save toolbar button in the Eligibility tab view.

4.4.6 Adding Eligibility Rules

You can add rules, or groups of rules, to determine the eligibility of customers for an offer.

To add an eligibility rule

- 1) Click the Add Rule toolbar button in the Eligibility tab view.
A blank rule is displayed.
- 2) Click the field on the left side of the blank rule, then click the selection button at the right side of the field.
- 3) Use the Edit value dialog box to select the attribute on which you want to base the rule.
For example, to base the rule on the age of the customer, select Attribute in the Edit value dialog box, then use the tree structure to locate the customer age attribute in the session attributes tree.
- 4) Click the equal sign, then click the selection button at the right side of the field.
Select an operator from the drop-down menu.
For example, if the offer applies to customers over the age of 21, select the IS GREATER THAN OR EQUAL TO (>=) operator.
- 5) Click the field on the right side of the blank rule, then click the selection button at the right side of the field.

- 6) Use the Edit value dialog box to select the value against which you want to evaluate the attribute.

For example, if the offer applies to customers over the age of 21, select the Constant option, then select Integer from the Data Type drop-down list. Type 21 in the Value text field.

For more information about how to work with rules, see the Oracle Real-Time Decisions Documentation Library on Oracle Technology Network .

To add a set of eligibility rules

- 1) Click the Add Rule Set toolbar button in the Eligibility tab view.
- 2) Click the field on the left side of the blank rule, then click the selection button at the right side of the field.
- 3) Use the Edit value dialog box to select the attribute on which you want to base the rule.
- 4) Click the equal sign, then click the selection button at the right side of the field. Select an operator from the drop-down menu.
- 5) Click the field on the right side of the blank rule, then click the selection button at the right side of the field.
- 6) Use the Edit value dialog box to select the value against which you want to evaluate the attribute.

For more information about how to work with rule sets, see the Oracle Real-Time Decisions Documentation Library on Oracle Technology Network.

4.4.7 Redeploy the RTD Inline Service if there is Any Change On It

To make the changes to the offers available in Siebel Call Center, you must first redeploy the modified inline service and then optionally promote it.

This task is a step in [Process of Setting Up Real-Time Intelligent Offers for Siebel Call Center](#) on page 33.

To redeploy the inline service

- 1) Navigate to the Administration - Marketing screen, then the Decision Center view.
- 2) Open the Decision Center view in edit mode.

For information about how to open decision center in edit mode, see [Opening Decision Center for Editing for IOG for Siebel Call Center](#) on page 34.

- 3) Make changes to the required offers, as described in [Process of Setting Up Real-Time Intelligent Offers for Siebel Call Center](#) on page 33.
- 4) Click the Redeploy button.
- 5) If you want to continue to work with the current inline service, select the Keep the current inline service opened for editing check box.

6) Click OK.

To promote the inline service

- 1) Navigate to the Administration - Marketing screen, then the Decision Center view.
- 2) Open the Decision Center view in edit mode.
 For information about how to open decision center in edit mode, see [Opening Decision Center for Editing for IOG for Siebel Call Center](#) on page 34.
- 3) Make changes to the required offers, as described in [Process of Setting Up Real-Time Intelligent Offers for Siebel Call Center](#) on page 33.
- 4) Redeploy the inline service, with the option to keep the current inline service opened for editing selected.
- 5) Click the Promote button.
- 6) Select the deployment state to which you want to promote the inline service from the To deployment state drop-down list.
- 7) If you want to continue to work with the current inline service, select the Keep the current inline service opened for editing check box.
- 8) Click OK.

4.4.8 IOG Views for Call Center Agents

IOG Application for Siebel Call Center provides views that expose offer recommendations by Oracle RTD for use by call center agents. The content of these views varies, depending on your business model.

The following following table lists the views that call center agents can use.

View	To Access, Navigate To...
Account Summary (RTD)	Accounts, Accounts List, and then Account Summary (RTD)
Contact Summary (RTD)	Contacts, Contacts List, and then Contact Summary (RTD)

5 Configuration Instructions for Siebel IOG

This chapter contains instructions if the user would like to further customize the Siebel IOG to:

- Embed the offer recommendation applet in multiple views
- Show offer details along with the offer list
- Manually trigger offer recommendations from RTD
- Customize customer responses to the offers
- Configure the offer attributes passed from RTD to Siebel
- Select RTD advisor
- Select RTD inline service to be used by IOG
- Customize the session key

For users on SIA, users need to manually apply the following changes in Siebel Tools.

For users on HOR, users have the alternative to directly import the Siebel repository files included in file CR_12-1YM4GZ9_SIF.zip, which includes:

Applets:

1. *Contact_Prospect Response Detail List Applet- Non Admin.sif*
2. *Response List Applet.sif*
3. *RTD Activity Intelligent Offers List Applet (B2B).sif*
4. *RTD Intelligent Offers Form Applet (B2C).sif*
5. *RTD Intelligent Offers List Applet (B2B).sif*
6. *RTD Intelligent Offers List Applet (B2C).sif*
7. *RTD Offer Status Form Applet (Phone Offer).sif*
8. *RTD Offer Status Form Applet (Web Offer).sif*
9. *RTD Retention Actions Form Applet (B2B).sif*
10. *RTD Retention Actions Form Applet (B2C).sif*

Business Components:

1. *RTD Intelligent Offers (B2B).sif*
2. *RTD Intelligent Offers (B2C).sif*
3. *RTD Retention Actions (B2B).sif*
4. *RTD Retention Actions (B2C).sif*
5. *Action.sif*

Business Objects:

1. *Action.sif*

Links:

1. *Action_RT D Intelligent Offers (B2B).sif*
2. *Action_RT D Intelligent Offers (B2C).sif*
3. *Contact_RT D Intelligent Offers (B2C).sif*
4. *Contact_RT D Retention Actions (B2C).sif*

Screen:

1. *Contacts Screen.sif*

View:

1. *Activity Attachment View.sif*
2. *Contact Summary View (RTD).sif*
3. *RTD Intelligent Offers Detail View.sif*

5.1 Seed Data

- System Preference

System Preference Name	System Preference Value	Description
RTD_B2B_APPLICATION_NAME	SiebelB2B	RTD Inline Service name for B2B Application
RTD_B2C_APPLICATION_NAME	SiebelB2C	RTD Inline service name for B2C Application

- Repository Objects

5.2 Siebel Objects Changes (Repository Changes)

Follow below configuration steps to apply the Siebel repository changes manually.

5.2.1 Embedding offer recommendation applet in multiple views

IOG 3.0 embeds Intelligent Offers applet in different views, although offers shown in the applet are the same: on the Contact Summary view, on the Account Summary view, on the Account/Service Request view, and so forth.

To integrate the Intelligent Offers Applet in a different view that is based on an object other than Account or Contact Business Object, the following configuration changes need to be done.

The configuration steps below explain how to integrate RTD Intelligent Offers List Applet (B2B) in the Activity Attachment View.

5.2.1.1 Link

Since the session key value will be fetched from the link definition, you need to ensure that the link definition is correct while creating a new link.

- Action/RTD Intelligent Offers (B2B) (New)

Name	Project	Parent Business Component	Child Business Component	Source Field	Destination Field	No Update No Delete No Insert
Action/RTD Intelligent Offers (B2B)	RTD Integration	Action	RTD Intelligent Offers (B2B)	Account Id	accountId	No Update = TRUE No Delete = TRUE No Insert = TRUE

5.2.1.2 Business Object

Follow below mentioned configuration steps to add the business component “RTD Intelligent Offers (B2B)” to business object “Action”.

- Action (Mod)

Action	
Primary BC	Action
Project	Activity

Add new Business Object component “RTD Activity Intelligent Offers (B2B)” as shown below.

Business Object Component		
New/Modify	Bus Comp	Link
New	RTD Intelligent Offers (B2B)	Action/RTD Intelligent Offers (B2B)

5.2.1.3 Applet

Create a new applet by copying existing “RTD Intelligent Offers List Applet (B2B)” and renaming it “RTD Activity Intelligent Offers List Applet (B2B)”.

Note: Existing “RTD Intelligent Offers List Applet (B2B)” can also be used instead of creating a new applet, but in a case where there is a need to disable buttons then it is better not to change the existing “RTD Intelligent Offers List Applet (B2B)”. Instead, create a new applet as mentioned below.

- RTD Activity Intelligent Offers List Applet (B2B) (New)

RTD Activity Intelligent Offers List Applet (B2B)	
Class	CSSFrameList
Project	RTD Integration
BusComp	RTD Intelligent Offers (B2B)
No Update	Y

Verify that the following standard controls are present in the Applet Web Template: EditList

Controls			
Name/Field	Caption	HTML Icon Map/HTML Type	Method Invoked Comment
ButtonInterested	Yes ! Tell Me More	HTML Type: MiniButtonEdit	Interested
DeleteRecord	Delete	MiniButton	DeleteRecord
EditRecord	Edit	MiniButton	EditRecord
ExecuteQuery	Go	MiniButtonQuery	ExecuteQuery
UndoQuery	Cancel	MiniButtonQuery	UndoQuery
UndoRecord	Cancel	MiniButtonNew	UndoRecord
WriteRecord	Save	MiniButtonEditNew	WriteRecord
NewQuery	Query	MiniButtonEdit	NewQuery
QueryAssistant	Query Assistant	MiniButtonQuery	ShowQueryAssistant
PositionOnRow		PositionOnRow	PositionOnRow
List		Text	
GotoPreviousSet		RecNavPrv	GotoPreviousSet
GotoNextSet		RecNavNxt	GotoNextSet
ButtonNotInterested	No. Thank You	MiniButtonEdit	NotInterested

Ensure that following List columns are present.

Name Field	Display Name Symbolic String	HTML Icon Map HTML Type	Available ReadOnly Runtime Show Popup	Show In List Comment
Reason Reason	Reason SBL_REASON-1004232329-507	Field	Available: TRUE Runtime: TRUE	Show In List: TRUE

Name Field	Display Name Symbolic String	HTML Icon Map HTML Type	Available ReadOnly Runtime Show Popup	Show In List Comment
			Show Pop-up: FALSE	
contactId contactId	Contact Id SBL_CONTACT_ID-1004225137-08M	Text	Available: TRUE Runtime: TRUE	
Id Id	Id SBL_ID-1004231107-35W	Text	Available: TRUE Runtime: TRUE	
likelihoodAcceptance likelihoodAcceptance	Likelihood Acceptance SBL_LIKELIHOOD_ACCEPTANCE	Text	Available: TRUE Runtime: TRUE	
likelihoodIntensity likelihoodIntensity	Score SBL_SCORE-1004232628-67A	Likelihood Intensity Text	Available: TRUE Runtime: TRUE	Show In List: TRUE
Name Name	Name SBL_NAME-1004224644-1K5	Text	Available: TRUE Runtime: TRUE	Show In List: TRUE
offerDescription offerDescription	Information SBL_INFORMATION-1004225947-01T	TextArea	Available: TRUE Runtime: TRUE Show Pop-up: TRUE	Show In List: TRUE
offered offered	Offer Id SBL_OFFER_ID-1004231600-4D0	Text	Available: TRUE Runtime: TRUE	Show In List: FALSE
offerNumber offerNumber	Offer Code SBL_OFFER_CODE-1004225618-01X	Text	Available: TRUE Runtime: TRUE	Show In List: TRUE
primaryCallScriptId primaryCallScriptId	Call Script SBL_CALL_SCRIPT-1004225158-118	Text	Available: TRUE Runtime: TRUE	Show In List: FALSE

Add a new web template item in the Activity Attachment View as shown below.

5.2.1.4 View

To expose the RTD Activity Intelligent Offers Applet (B2B) in Activity Attachment View, apply following configuration changes.

- Activity Attachment View (Mod)

Activity Attachment View				
Project	Activity (SSE)			
BusObj	Action			
WebTemplate Items				
Name / Applet	Applet Mode	Item Identifier	Display Size Display Visibility	Move Range Position
RTD Activity Intelligent Offers List Applet (B2B)	Edit List	3		
RTD Activity Intelligent Offers List Applet (B2B)				

5.2.1.5 Integrating RTD Intelligent Offers List Applet (B2C) in the Activity Attachment View

The following configuration steps explain how to integrate RTD Intelligent Offers List Applet (B2C) in the Activity Attachment View.

5.2.1.5(A) Business Component

Add a new calculated business component field “Calculated Value”, this field will be used in the link to fetch the session key value.

- Action (Mod)

Action	
Class	CSSBCFINSActivity
Project	Activity

Add a new field as shown below.

Business Component: Action					
Name	Calculated Value	Type	No Copy	Validation	Comments
Calc Contact Id	[Contact Id]	DTYPE_TEXT			This will be used in link definition.

5.2.1.5(B) Create a new Link

Since the session key value will be fetched from link definition we need to ensure that link definition is correct while creating a new link.

Link

Action/ RTD Intelligent Offers (B2C) (*New*)

Name	Project	Parent Business Component	Child Business Component	Source Field	Destination Field	No Update No Delete No Insert
Action/RTD Intelligent Offers (B2C)	RTD Integration	Action	RTD Intelligent Offers (B2C)	Calc Contact Id	contactId	No Update = TRUE No Delete = TRUE No Insert = TRUE

5.2.1.5(C) Business Object

Add the new Business Object component "RTD Intelligent Offers (B2C)" as shown below.

- Action (*Mod*)

Action	
Primary BC	Action
Project	Activity

Add the new Business Object component "RTD Intelligent Offers (B2C)" as shown below.

Business Object Component		
New/Modify	Bus Comp	Link
New	RTD Intelligent Offers (B2C)	Action/RTD Intelligent Offers (B2C)

5.2.1.5(D) View

To expose the "RTD Intelligent Offers List Applet (B2C)" in the Activity Attachment View, apply the following configuration changes.

- Activity Attachment View (Mod)

Activity Attachment View				
Project	Activity (SSE)			
BusObj	Action			
WebTemplate Items				
Name / Applet	Applet Mode	Item Identifier	Display Size Display Visibility	Move Range Position
RTD Intelligent Offers List Applet (B2C)	Edit List	4		
RTD Intelligent Offers List Applet (B2C)				

5.2.2 Showing offer details along with Offer list

IOG 3.0 provides an Offer Details applet and associates it with an Intelligent Offers applet. The Offer Details applet appears on the same screen as the Intelligent Offers applet, side-by-side. When you pick an offer from the Intelligent Offers applet, offer details appear in the Offer Details applet.

Complete the following configuration steps to create an Offer Details applet and associate it with an Offer Recommendation applet. The configuration steps describe creating new detail applets and how to integrate this detail applet in the Contact Summary view (RTD).

5.2.2.1 Applet

Complete the following configuration steps to create an Offer Details applet and associate it with an Offer Recommendation applet. The configuration steps describe creating new detail applets and how to integrate this detail applet in the Contact Summary view (RTD).

- RTD Intelligent Offers Form Applet (B2C) (New)

RTD Intelligent Offers Form Applet (B2C)	
Class	CSSFrame
Project	RTD Integration
BusComp	RTD Intelligent Offers (B2C)
No Update	Y

Add the following controls to the new applet.

Controls			
Name/Field	Caption	HTML Icon Map/HTML Type	Method Invoked Comment ReadOnly Visible
DeleteRecord	Delete	MiniButton	DeleteRecord
EditRecord	Edit	MiniButton	EditRecord
ExecuteQuery	Go	MiniButtonQuery	ExecuteQuery
UndoQuery	Cancel	MiniButtonQuery	UndoQuery
UndoRecord	Cancel	MiniButtonNew	UndoRecord
WriteRecord	Save	MiniButtonEditNew	WriteRecord
NewQuery	Query	MiniButtonEdit	NewQuery

Controls			
Name/Field	Caption	HTML Icon Map/HTML Type	Method Invoked Comment ReadOnly Visible
QueryAssistant	Query Assistant	MiniButtonQuery	ShowQueryAssistant
PositionOnRow		PositionOnRow	PositionOnRow
List		Text	
GotoPreviousSet		RecNavPrv	GotoPreviousSet
GotoNextSet		RecNavNxt	GotoNextSet
contactId contacted	Contact Id SBL_CONTACT_ID-1004225137-08M	HTML Type: Text	ReadOnly: TRUE Visible: TRUE
Id Id	Id SBL_ID-1004231107-35W	HTML Type: Text	ReadOnly: TRUE Visible: TRUE
name name	Name SBL_NAME-1004224644-1K5	HTML Type: Text	ReadOnly: TRUE Visible: TRUE
offerDescription offerDescription	Information SBL_INFORMATION-1004225947-01T	HTML Type: Text	ReadOnly: TRUE Visible: TRUE
offerId offered	Offer Id SBL_OFFER_ID-1004231600-4D0	HTML Type: Text	ReadOnly: TRUE Visible: TRUE
offerNumber offerNumber	Offer Code SBL_OFFER_CODE-1004225618-01X	HTML Type: Text	ReadOnly: TRUE Visible: TRUE

Add following applet Web template.

Applet Web Template		
Name	Type	Web Template
Edit	Edit	Applet Form Grid Layout

Add following Applet web template Item

Applet Web Template Item		
Name	Control	Type
Name	name	Control
offerDescription	offerDescription	Control
offerNumber	offerNumber	Control
offerId	offerId	Control

Add the new applet to the Contact Summary View (RTD) as shown below.

5.2.2.2 View

Add the new “RTD Intelligent Offers Form Applet (B2C)” to Contact Summary View (RTD).

- Contact Summary View (RTD) (Mod)

Contact Summary View (RTD)				
Project	RTD Integration			
BusObj	Contact			
WebTemplate Items				
Name / Applet	Applet Mode	Item Identifier	Display Size Display Visibility	Move Range Position
RTD Intelligent Offers Form Applet (B2C)	Edit	9		

5.2.2.3 Configuring Drilldown

To configure drill down on Offer “Name” control follow below mentioned configuration steps.

Create a new View as described in the following configuration steps.

5.2.2.3(A) View

- RTD Intelligent Offers Detail View (New)

RTD Intelligent Offers Detail View				
Project	RTD Integration			
BusObj	Contact			
Thread Applet	Contact Form Applet			
Thread Field	Last Name			
Thread Title	Contact:			
Add To History	TRUE			
Type	Standard			
WebTemplate	View Detail			
WebTemplate Items				
Name / Applet	Applet Mode	Item Identifier	Display Size Display Visibility	Move Range Position
Contact Form Applet Contact Form Applet	Edit	1		
RTD Intelligent Offers Form Applet (B2C) RTD Intelligent Offers Form Applet (B2C)	Edit	2		

Create a “Drilldown” on Name control as shown below.

Drilldown Object			
Name	Hyperlink Field	View	Business Component
OfferDetail	name	RTD Intelligent Offers Detail View	RTD Intelligent Offers (B2C)

5.2.2.3(B) Screen

Add new view to Contacts Screen.

- Contacts Screen (Mod)

Screen Views				
Name	View	Type/ Sequence	Parent Category	Viewbar Text Menu Text
RTD Intelligent Offers Detail View	RTD Intelligent Offers Detail View	Detail View	Contact List	Offer Details Offer Details

5.2.2.3(C) Seed Data Changes

Add new view in the Application and add the appropriate Responsibility.

Navigate to Administration- Application, and then Views, and create a new View record with following values.

Views		
View Name	Description	Default Local Access
RTD Intelligent Offers Detail View	RTD Intelligent Offers Detail View	TRUE

Add the appropriate Responsibility from Responsibility child applet.

5.2.3 Manually triggering offer recommendations from RTD

To manually trigger the RTD for an offer recommendation, follow the configuration steps below. You can add a Generate Offers button on the Retention Action applet and Intelligent Offers applet that, once clicked, triggers IOG to retrieve offer recommendations from RTD. At the back end, once triggered, IOG calls RTD through Advisor integration point for offer recommendation.

Business Service method “CallAdvisor” can be used to manually trigger offer recommendations from eScript. The Input parameters to make Advisor calls are as follows.

- Business Service: RTD Integration Service

Business Service: RTD Integration Service	
Class	CSSRTDIntegrationService
Project	RTD Integration

- Business Service Method: CallAdvisor

Business Service Method Input Arg			
Input Parameter Name	Data Type	Storage Type	Description
Application	String	Property	Inline Service Name on RTD Server
Attributes	Hierarchy	Hierarchy	Property set containing Name/Value pairs (properties) of additional attributes (optional)
IntegrationPoint	String	Property	Integration Point Name on RTD Server. For offer recommendation the value for IntegrationPoint = GetIntelligentOffers
SessionKeys	Hierarchy	Hierarchy	Property set containing Name/Value pairs (properties) of Session Keys
Trace	String	Property	Enables extra logging (TRUE/FALSE) (optional, Default = FALSE)

Output property contains the following information.

Business Service Method Output Arg			
output Parameter Name	Data Type	Storage Type	Description
Choices	Hierarchy	Property	One or more Child Propertyset of type Choice.
TraceInfo	String	Property	Trace information (Optional)
LogInfo	String	Property	Log information (Optional)
ServiceVersion	String	Property	Service Version (Optional)
ClusterVersion	String	Property	Cluster Version (Optional)

5.2.4 Customizing customer responses to offers

In IOG 3.0, you can create new types of offer responses and modify existing ones in design time. Moreover, when an offer is turned down by the customer, you can record the reason for rejection as well.

Add the following BC user properties in the list of Business Components described below.

5.2.4.1 Business Components

In current IOG implementation, the Yes Tell Me More button captures the response type Requested More Info, and the No Thanks button is used to capture the response type No Interest.

To modify the existing buttons to capture different types of response types, complete the configurations changes described below.

These configuration steps contain an example that explains the configuration steps required to customize the existing No Thank You button in order to capture the Requested Unsubscribe response type on the RTD Intelligent Offers List Applet (B2C).

- RTD Intelligent Offers (B2B)
- RTD Intelligent Offers (B2C)
- RTD Retention Actions (B2B)
- RTD Retention Actions (B2C)

Business Component User Property		
New/Modify	Name	Value
New	NotInterested	No Interest
New	Interested	Requested more info

Add the following BC user property and ensure that the BC user property name is the same as the value specified for Method Invoked on the button control ButtonNotInterested in the RTD Intelligent Offers List Applet (B2C).

BC user property values should contain the Language Independent code of respective Response Type.

To customize the existing Offer Response Capture buttons, follow the configuration steps below.

- Create a new BC user property in respective VBC as shown below.

Business Component User Property		
New/Modify	Name	Value
New	NotInterested	Requested Unsubscribe

Similarly, to customize the existing “Yes tell me more” response capture button, the above steps should be followed with the appropriate BC user property name and BC user property value.

Business Component User Property		
New/Modify	Name	Value
New	Interested	(Enter appropriate LIC of response type)

Configuration changes listed below are required if you want to capture the reasons for rejection from “RTD Intelligent Offers List Applet (B2C)”.

Modify the “No Update” flag value to FALSE from TRUE in below link definition.

5.2.4.2 Link

Apply following configuration changes to make the applet columns editable in order to allow the user to enter the value for reason field.

- Contact/RTD Intelligent Offers (B2C) (Mod)

Modify the link as shown below.

Name	Project	Parent Business Component	Child Business Component	Source Field	Destination Field	No Update No Delete No Insert	Comments
Contact/RTD Intelligent Offers (B2C)						No Update = FALSE	

Follow the configuration steps described below.

5.2.4.3 Business Component

Add a new business component field “Reason”.

- RTD Intelligent Offers (B2C) (Mod)

RTD Intelligent Offers (B2C)	
Class	CSSBCVRTDOffers
Project	RTD Integration

Add a new field as shown below.

Business Component: RTD Intelligent Offers (B2C)			
Name	Type	Immediate Post	PickList
Reason	DTYPE_TEXT	TRUE	Offer Rejection Reason PickList

Make a Pick map entry for “Reason” field as shown below.

Pick Maps		
For the Field 'Reason'		
New/Modify	Field	Picklist Field
New	Reason	Value

Add a BC user property as shown below.

Business Component User Properties		
New/Modify	Name	Value
New	Reason	Reason

5.2.4.4 Applet

Expose the “Reason” control in the UI, follow below mentioned configuration steps.

- RTD Intelligent Offers List Applet (B2C) (Mod)

Update the existing applet as shown below.

RTD Intelligent Offers List Applet (B2C)	
Class	CSSFrameList
Project	RTD Integration
BusComp	RTD Intelligent Offers (B2C)
No Update	N
Comments	Modify "No Update" flag from "Y" to "N".

Add a new List Column to the Applet as shown below.

List Column					
Name Field	Display Name Symbolic String	HTML Icon Map HTML Type	Detail / MVG / Pick Applet	Available ReadOnly Runtime Show Popup	Show In List Comment
Reason Reason	Reason SBL_REASON-1004232329-507	HTML Type: Field		Available: TRUE Runtime: TRUE Show Pop-up: FALSE	TRUE

Add a new Applet Web Template Item as shown below.

Applet Web Template		
Name	Type	Web Template
Edit List	Edit List	Applet List (Base/Edit List)

Applet Web Template Item			
Name	Control	Type	Comments
Reason	Reason	List Item	Reason column is added to enter the reasons of offers rejection.

Modify following existing List Columns as shown below.

List Columns					
Name	Display Name Symbolic String	HTML Icon Map HTML Type	Detail / MVG / Pick Applet	Available ReadOnly Runtime Show Popup	Show In List Comment
contactId				ReadOnly: TRUE	
Id				ReadOnly: TRUE	
likelihoodAcceptance				ReadOnly: TRUE	
likelihoodIntensity				ReadOnly: TRUE	
Name				ReadOnly: TRUE	
offerDescription				ReadOnly: TRUE	
offered				ReadOnly: TRUE	
offerNumber				ReadOnly: TRUE	
primaryCallScriptId				ReadOnly: TRUE	

5.2.4.5 Seed Data Changes

To pass the "Reason" to RTD follow below mentioned configuration steps.

- 1) Navigate to Administration – Runtime Events and click Events.
- 2) Select a record with RTD Intelligent Offers (B2C) as Object Name and with subevent as NotInterested.
- 3) Drill down on Action Set Name and navigate the Personalization Action List Applet child applet.
- 4) Select Set RTD Key List record.
- 5) Navigate to Personalization Action Form Applet and modify the Value field as shown below, and then save the record.

Field	New Value
Value	'RTD Key:contactId;RTD Key:choiceName;RTD Key:eventName;RTD Key:agentId;RTD Key:Reason'

- 6) Click New Button on the Personalization Action List Applet, and enter the values as shown below.

Name	Action Type	Sequence	Active
Set RTD Key:Reason	Attribute Set	7	TRUE

- 7) Go to Personalization Action Form Applet child applet and enter values for fields as shown below and save the record.

Profile Attribute	Set Operator	Value
RTD Key:Reason	Set	[Reason]

Configuration changes below are required to capture the reasons for rejection from “RTD Retention Actions (B2C)”.

5.2.4.6 Link

To expose the new “Reason” field in the UI, follow below mentioned configuration steps

- Contact/RTD Retention Actions (B2C) (Mod)

Modify the link as shown below.

Name	Project	Parent Business Component	Child Business Component	Source Field	Destination Field	No Update No Delete No Insert	Comments
Contact/RTD Retention Actions (B2C)						No Update = FALSE	

5.2.4.7 Business Component

Add a new business component field “Reason”.

- RTD Retention Actions (B2C)

RTD Retention Actions (B2C)	
Class	CSSBCVRTDOffers
Project	RTD Integration

Add a new field as shown below.

Business Component: RTD Intelligent Offers (B2C)						
Name	Type	Immediate Post	PickList	No Copy	Validation	Column
Reason	DTYPE_TEXT	TRUE	Offer Rejection Reason PickList			

Make a Pick map entry as shown below.

Pick Maps	For the Field 'Reason'	
New/Modify	Field	Picklist Field
New	Reason	Value

Add BC user properties as shown below.

Business Component User Properties		
New/Modify	Name	Value
New	Reason	Reason

5.2.4.8 Applet

Expose the "Reason" field in the UI.

- RTD Retention Actions Form Applet (B2C) (Mod)

RTD Retention Actions Form Applet (B2C)	
Class	CSSFrame
Project	RTD Integration
BusComp	RTD Retention Actions (B2C)
No Update	N
Comments	Modified "No Update" flag from "Y" to "N".

Add a new control as shown below.

Control					
Name Field	Caption Caption String	HTML Icon Map HTML Type	Detail / MVG / Pick Applet	ReadOnly Runtime Show Popup	Visible Comment
Reason	Reason SBL_REASO	BC Field		Runtime: TRUE Show Pop-up: FALSE	TRUE
Reason	N- 1004232329- 507				

Add a new control to the Applet Web Template as shown below.

Applet Web Template		
Name	Type	Web Template
Edit	Edit	Applet Form Grid Layout

Applet Web Template Item			
Name	Control	Type	Comments
Reason	Reason	Control	Reason column is added to enter the reason for offer rejection.

Make following controls as Read-Only.

Controls					
Name	Caption Caption String	HTML Icon Map HTML Type	Detail / MVG / Pick Applet	ReadOnly Runtime Show Popup	Visible Comment
contactId				ReadOnly: TRUE	
Id				ReadOnly: TRUE	
Name				ReadOnly: TRUE	
churnIndicatorIntensity				ReadOnly: TRUE	
offerDescription				ReadOnly: TRUE	
offered				ReadOnly: TRUE	

5.2.4.9 Seed Data Changes

To pass the “Reason” to RTD, apply the following changes to Run Time Events.

- 1) Navigate to Administration – Runtime Events, and then click Events.
- 2) Select a record with “RTD Retention Actions (B2C)” as Object Name and subevent as “NotInterested”.
- 3) Drill down on Action Set Name, and navigate to Personalization Action List Applet child applet.
- 4) Select the Set RTD Key List record.
- 5) Go to Personalization Action Form Applet and modify the Value field as shown below, and then save the record.

Field	New Value
Value	'RTD Key:contactId;RTD Key:choiceName;RTD Key:eventName;RTD Key:agentId;RTD Key:Reason'

- 6) Click New Button on Personalization Action List Applet, and enter the values as shown below.

Name	Action Type	Sequence	Active
Set RTD Key:Reason	Attribute Set	7	TRUE

- 7) Go to the Personalization Action Form Applet child applet and enter values for fields as shown below and save the record.

Profile Attribute	Set Operator	Value
RTD Key:Reason	Set	[Reason]

Configuration changes described below are required to capture the reason for rejection from “RTD Intelligent Offers (B2B)”.

5.2.4.10 Link

Apply the following configuration changes to make “RTD Intelligent Offers List Applet (B2B)” applet columns editable in the UI.

- Account/RTD Intelligent Offers (B2B) (Mod)

Modify the link as shown below.

Name	Project	Parent Business Component	Child Business Component	Source Field	Destination Field	No Update No Delete No Insert	Comments
Account/RTD Intelligent Offers (B2B)						No Update = FALSE	

5.2.4.11 Business Component

Add a new field “Reason” to business component “RTD Intelligent Offers (B2B)” as shown in below configuration steps.

- RTD Intelligent Offers (B2B) (Mod)

RTD Intelligent Offers (B2B)	
Class	CSSBCVRTDOffers
Project	RTD Integration

Add a new field as shown below.

Business Component: RTD Intelligent Offers (B2B)					
Name	Join	Calculated Value	Type	Immediate Post	PickList
Reason			DTYPE_TEXT	TRUE	Offer Rejection Reason PickList

Make a Pick map entry as shown below.

Pick Maps		
For the Field 'Reason'		
New/Modify	Field	Picklist Field
New	Reason	Value

Add BC user properties as shown below.

Business Component User Properties		
New/Modify	Name	Value
New	Reason	Reason

5.2.4.12 Applet

To expose the “Reason” field in the UI follow below mentioned configuration steps.

- RTD Intelligent Offers List Applet (B2B) (Mod)

Update the existing applet as shown below.

RTD Intelligent Offers List Applet (B2B)	
Class	CSSFrameList
Project	RTD Integration
BusComp	RTD Intelligent Offers (B2B)
No Update	N
Comments	Modified “No Update” flag from “Y” to “N”.

Add a new List Column as shown below.

List Columns					
Name Field	Display Name Symbolic String	HTML Icon Map HTML Type	Detail / MVG / Pick Applet	Available ReadOnly Runtime Show Popup	Show In List Comment
Reason Reason	Reason SBL_REASON-1004232329-507	Field		Available: TRUE Runtime: TRUE Show Pop-up: FALSE	TRUE

Add a new List Column to the Applet Web Template as shown below.

Applet Web Template		
Name	Type	Web Template
Edit List	Edit List	Applet List (Base/Edit List)

Applet Web Template Item			
Name	Control	Type	Comments
Reason	Reason	List Item	Reason column is added to allow enter the reasons of offers rejection.

Modify existing List Columns as shown below.

List Columns					
Name	Display Name Symbolic String	HTML Icon Map HTML Type	Detail / MVG / Pick Applet	Available ReadOnly Runtime Show Popup	Show In List Comment
contactId				ReadOnly: TRUE	
id				ReadOnly: TRUE	
likelihoodAcceptance				ReadOnly: TRUE	
likelihoodIntensity				ReadOnly: TRUE	
name				ReadOnly: TRUE	
offerDescription				ReadOnly: TRUE	
offerId				ReadOnly: TRUE	
offerNumber				ReadOnly: TRUE	
primaryCallScriptId				ReadOnly: TRUE	

5.2.4.13 Seed Data Changes

To pass the Reason to RTD, apply the following changes to Run Time Events.

- 1) Navigate to Administration – Runtime Events and click Events.
- 2) Select a record with “RTD Intelligent Offers (B2B)” as Object Name and with subevent as “NotInterested”.
- 3) Drill down on Action Set Name, and navigate to Personalization Action List Applet child applet.
- 4) Select the Set RTD Key List record.
- 5) Go to Personalization Action Form Applet and modify the Value field as shown below and save the record.

Field	New Value
Value	'RTD Key:contactId;RTD Key:choiceName;RTD Key:eventName;RTD Key:agentId;RTD Key:Reason'

- 6) Click New Button on the Personalization Action List Applet, and enter the values as shown below.

Name	Action Type	Sequence	Active
Set RTD Key:Reason	Attribute Set	7	TRUE

- 7) Go to Personalization Action Form Applet child applet and enter values for fields as shown below and save the record.

Profile Attribute	Set Operator	Value
RTD Key:Reason	Set	[Reason]

Configuration changes described below are required to capture the reasons for rejection from RTD Retention Actions (B2B).

5.2.4.14 Link

- Account/RTD Retention Actions (B2B) (Mod)

Modify the link as shown below.

Name	Project	Parent Business Component	Child Business Component	Source Field	Destination Field	No Update No Delete No Insert	Comments
Account/RTD Retention Actions (B2B)						No Update = FALSE	

5.2.4.15 Business Component

- RTD Retention Actions (B2B)

RTD Retention Actions (B2B)	
Class	CSSBCVRTDOffer
Project	RTD Integration

Add a new field as shown below.

Business Component: RTD Retention Actions (B2B)								
Name	Join	Calculated Value	Type	Immediate Post	PickList	No Copy	Validation	Column
Reason			DTYPE_TEXT	TRUE	Offer Rejection Reason PickList			

Make a Pick map entry as shown below.

Pick Maps		
For the Field 'Reason'		
New/Modify	Field	Picklist Field
New	Reason	Value

Add BC user properties as shown below.

Business Component User Properties		
New/Modify	Name	Value
New	Reason	Reason

5.2.4.16 Applet

- RTD Retention Actions Form Applet (B2B) (Mod)

RTD Retention Actions Form Applet (B2B)	
Class	CSSFrame
Project	RTD Integration
BusComp	RTD Retention Actions (B2B)
No Update	N
Comments	Modified "No Update" flag from "Y" to "N".

Add a new control as shown below.

Control					
Name Field	Caption Caption String	HTML Icon Map HTML Type	Detail / MVG / Pick Applet	ReadOnly Runtime Show Popup	Visible Comment
Reason Reason	Reason SBL_REASON-1004232329-507	BC Field		Runtime: TRUE Show Pop-up: FALSE	TRUE

Add a new Applet Web Template Item to Applet Web Template as shown below.

Applet Web Template		
Name	Type	Web Template
Edit	Edit	Applet Form Grid Layout

Applet Web Template Item			
Name	Control	Type	Comments
Reason	Reason	Control	Reason column is added to enter the reason for offer rejection.

Make the following controls as read-only.

Controls					
Name	Caption Caption String	HTML Icon Map HTML Type	Detail / MVG / Pick Applet	ReadOnly Runtime Show Popup	Visible Comment
Name				ReadOnly: TRUE	
churnIndicatorIntensity				ReadOnly: TRUE	
offerDescription				ReadOnly: TRUE	

5.2.4.17 Seed Data Changes

To pass the "Reason" to RTD, apply the following changes to Run Time Events.

- 1) Navigate to Administration – Runtime Events and click Events.
- 2) Select a record with "RTD Retention Actions (B2B)" as Object Name and with subevent as "NotInterested".
- 3) Drill down on Action Set Name, and navigate to Personalization Action List Applet child applet.
- 4) Select Set RTD Key List record and go to Personalization Action Form Applet" and modify the Value field as shown below and save the record.

Field	New Value
Value	'RTD Key:contactId;RTD Key:choiceName;RTD Key:eventName;RTD Key:agentId;RTD Key:Reason'

- 5) Click the New Button on Personalization Action List Applet, and enter the values as shown below.

Name	Action Type	Sequence	Active
Set RTD Key:Reason	Attribute Set	7	TRUE

- 6) Go to “Personalization Action Form Applet” child applet and enter values for fields as shown below and save the record.

Profile Attribute	Set Operator	Value
RTD Key:Reason	Set	[Reason]

5.2.4.18 Applet

- Contact/Prospect Response Detail List Applet- Non Admin (*Mod*)

Contact/Prospect Response Detail List Applet- Non Admin	
Class	CSSSWEFrameResponseList
Project	Response
BusComp	Response
Comments	Added a new list column “Reason”.

Add a new List Column to the Applet as shown below.

List Column					
Name Field	Display Name Symbolic String	HTML Icon Map HTML Type	Detail / MVG / Pick Applet	Available ReadOnly Runtime Show Popup	Show In List Comment
Reason	Reason	Field		Available: TRUE Runtime: TRUE Show Pop-up: FALSE	TRUE
Reject Reason Code	SBL_REASON-1004232329-507				

Add a new List Column to the Applet Web Template as shown below.

Applet Web Template		
Name	Type	Web Template
Edit List	Edit List	Applet List (Base/Edit List)

Applet Web Template Item			
Name	Control	Type	Comments
Reason	Reason	List Item	Reason column is added to enter the reasons of offers rejection.

5.2.5 Configuring the offer attributes passed from RTD to Siebel

In IOG 3.0, you can configure the offer attributes passed from RTD to Siebel. For example, you can pass the type of recommendation (up-sell or churn), Account Number, Phone line (service Id), Contract Number (asset number), Campaign Id, or Treatment Id as long as they are defined in the offer object in RTD Decision Studio.

Follow the configuration steps below if user wishes to add channel field to intelligent offers Applet.

Note: Make sure Send to Client is checked for channel attribute in RTD Inline Service.

5.2.5.1 Business Component

- RTD Intelligent Offers (B2C)

RTD Intelligent Offers (B2C)	
Class	CSSBCVRTDOffers
Project	RTD Integration

Add a new field as shown below.

Business Component: RTD Intelligent Offers (B2C)								
Name	Join	Calculated Value	Type	Immediate Post	PickList	No Copy	Validation	Column
channel			DTYPE_TEXT					

5.2.5.2 Applet

- RTD Intelligent Offers List Applet (B2C) ((Mod))

RTD Intelligent Offers List Applet (B2C)	
Class	CSSFrameList
Project	RTD Integration
BusComp	RTD Intelligent Offers (B2C)
Comments	Added a new list column "channel".

Add a new List Column to the Applet as shown below.

List Column					
Name Field	Display Name Symbolic String	HTML Icon Map HTML Type	Detail / MVG / Pick Applet	Available ReadOnly Runtime Show Popup	Show In List Comment
channel	Channel	Text		Available: TRUE Runtime: TRUE Show Pop-up: FALSE	TRUE
channel	SBL_CHANNEL-1004224441-0N9				

Add a new List Column to the Applet Web Template as shown below.

Applet Web Template			
Name	Type	Web Template	
Edit List	Edit List	Applet List (Base/Edit List)	
Applet Web Template Item			
Name	Control	Type	Comments
Channel	Channel	List Item	Media type code information sent by RTD

5.2.6 Selecting RTD Advisor

There might be multiple advisors created in RTD Decision Studio to generate offer recommendations, each with its own goal and offer set. In IOG 3.0, you can specify which advisor to use by configuring the advisor's name in IOG.

Follow the configuration steps below to change the advisor name in order to get intelligent offers for B2C Application

5.2.6.1 Business Component

- RTD Intelligent Offers (B2C) (Mod)

RTD Intelligent Offers (B2C)	
Class	CSSBCVRTDOffers
Project	RTD Integration

Modify a BC user property as shown below.

Business Component User Properties		
New/Modify	Name	Value
Modify	Integration Point	GetIntelligentOffers

Follow below mentioned configuration steps to change the advisor name to get intelligent offers for B2B Application.

5.2.6.2 Business Component

- RTD Intelligent Offers (B2B) (Mod)

RTD Intelligent Offers (B2B)	
Class	CSSBCVRTDOffers
Project	RTD Integration

Modify a BC user property as shown below.

Business Component User Properties		
New/Modify	Name	Value
Modify	Integration Point	GetIntelligentOffers

5.2.7 Selecting RTD inline service

In IOG 3.0, you can select the RTD inline service to use (SiebelB2B or SiebelB2C) in Siebel. Follow the steps below to change RTD inline service name for B2B or B2C Application

- System Preference

System Preference Name	System Preference Value	Description
RTD_B2B_APPLICATION_NAME	SiebelB2B	RTD Inline Service name for B2B Application
RTD_B2C_APPLICATION_NAME	SiebelB2C	RTD Inline service name for B2C Application

5.2.7.1 Business Component (Modify)

Apply following configuration changes to respective business components.

- RTD Intelligent Offers (B2B)

Buscomp User Properties		
Name	Value	Comments
Syspref Application	RTD_B2B_APPLICATION_NAME	To support customizability of RTD B2B Inline Service name from System Preferences

- RTD Retention Actions (B2B)

Buscomp User Properties		
Name	Value	Comments
Syspref Application	RTD_B2B_APPLICATION_NAME	To support customizability of RTD B2B Inline Service name from System Preferences

- RTD Intelligent Offers (B2C)

Buscomp User Properties		
Name	Value	Comments
Syspref Application	RTD_B2C_APPLICATION_NAME	To support customizability of RTD B2C Inline Service name from System Preferences

- RTD Retention Actions (B2C)

Buscomp User Properties		
Name	Value	Comments
Syspref Application	RTD_B2C_APPLICATION_NAME	To support customizability of RTD B2C Inline Service name from System Preferences

5.2.8 Customizing the session key

RTD obtains customer information from the current Siebel session. In IOG 3.0, you can select one of the following attributes as the session key: MSISDN, telephone number, account ID, contact ID.

Follow the configuration steps below if user wishes to change the session key name for B2C Application

5.2.8.1 Business Component

- RTD Intelligent Offers (B2C) *(Mod)*

RTD Intelligent Offers (B2C)	
Class	CSSBCVRTDOffers
Project	RTD Integration

Modify a BC user property as shown below.

Business Component User Properties		
New/Modify	Name	Value
Modify	Session Key	contactId

Edit Server Scripts to set session key value in RTD_SESSION_KEY_VAL profile attribute.

Follow the configuration steps below to change the session key name for B2B Application.

5.2.8.2 Business Component

- RTD Intelligent Offers (B2B) *(Mod)*

RTD Intelligent Offers (B2B)	
Class	CSSBCVRTDOffers
Project	RTD Integration

Modify a BC user property as shown below.

Business Component User Properties		
New/Modify	Name	Value
Modify	Session Key	accountId

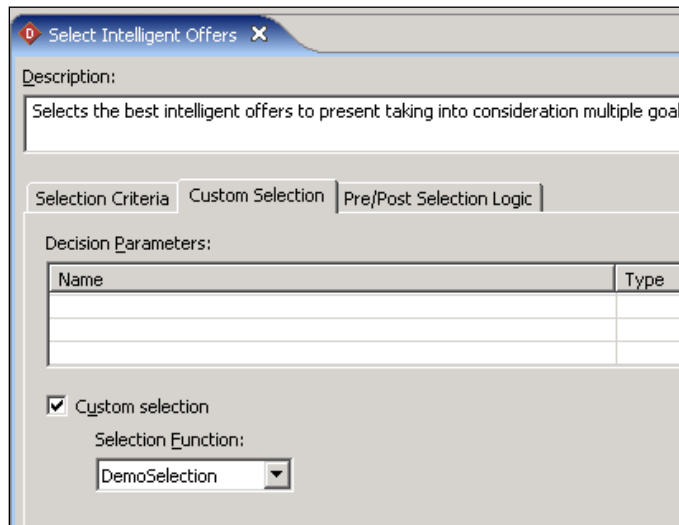
Edit Server Scripts to set session key value in RTD_SESSION_KEY_VAL profile attribute.

6 Optional Setups

6.1 Setting up Demo Intelligent Cross Sell and Retention Management

The SiebelB2C and SiebelB2B RTD metadata have demo intelligent cross sell and retention management scenarios pre-built, but require a small modification to activate the logic.

- 1) Open SiebelB2C (or SiebelB2B) in Oracle RTD Decision Studio. Expand the Service Metadata folder, then the Decisions folder, and double-click on the decision Select Intelligent Offers for editing.
- 2) Make active the Custom Selection tab, check the checkbox labeled Custom selection, and choose **DemoSelection** from the Selection Function dropdown.



- 3) Save the changes and redeploy the Inline Service.

6.1.1 Single-customer demo tests

For SiebelB2C, the retail banking scenario is as follows:

Scenario Type	Description
Cross Sell	Linda Johnson is a 70 year old conservative customer of Bank X. She has a checking and savings account and banks via traditional check writing and branch teller only. Linda calls the bank and the agent clicks on the phone icon to accept the call. The agent is navigated to the customer dashboard view displaying Linda's profile, financial assets, recent activities, etc. The dashboard also indicates that Linda is currently low churn risk and the best cross sell offer at this time is a <i>mortgage</i>

Scenario Type	Description
	<p><i>related offer</i>. Linda then indicates that she is calling to <i>order checks</i>. The agent places the order and creates an activity to log the call reason. With the additional call reason, the dashboard refreshes the best cross sell offers and now recommends a <i>web bill pay offer</i>. The agent delivers the offer and Linda's response is tracked to the campaign and offer.</p>
Retention	<p>Robert Knowles is a young 25 year old day trader. Although his brokerage account is with Bank X, he does not have a checking or savings account there. Robert calls the bank and the agent clicks on the phone icon to accept the call. The agent in navigated to the customer dashboard view displaying Robert's profile, financial assets, recent activity etc. The dashboard also indicated that Robert has moderate churn risk and the best cross sell offer at this time is a <i>Millennium Credit Card offer</i>. Robert then indicates that he is calling <i>to inquire about his recent trading fees</i>. The agent explains the fees and creates an activity to log the call reason. With the additional call reason, the dashboard updates to indicate that Robert has become high churn risk and a retention action is necessary. The agent offers a <i>Free One Time Fees Waiver</i> and Robert's response is tracked to the campaign and offer.</p>

For SiebelB2B, the telecommunications scenario is as follows:

Scenario Type	Description
Cross Sell	<p>Pat Weathers is a real estate who is constantly on the go when making mobile calls. He has an average usage of 1400+ minutes a month. Pat calls the telecom company and the agent clicks on the phone icon to accept the call. The agent is navigated to the customer dashboard view displaying Pat's profile, service items, billing items, recent activities, etc. The dashboard also indicates that Pat is currently low churn risk and the best cross sell offer at this time is a <i>50% Off Hands Free Cell Kit offer</i>. Pat then indicates that he is calling to <i>Order an SMS Plan</i>. Before the agent places the order, he creates an activity to log the call reason. With the additional call reason, the dashboard refreshes the best cross sell offers and now recommends an <i>Upgrade to Blackberry 7230 Unlimited Internet offer</i>. The agent delivers the offer and Pat's response is tracked to the campaign and offer.</p>
Retention	<p>Shirley Roberts is a consultant who runs a home based business. She's a Gold customer in terms of profitability partly due to her Very High monthly cell phone bills. Due to recent</p>

Scenario Type	Description
	<p>network repairs, she has experienced a spike in dropped calls the last 30 days. Shirley calls the telecom company and the agent clicks on the phone icon to accept the call. The agent is navigated to the customer dashboard view displaying Shirley's profile, service items, billing items, recent activities, etc. The dashboard also indicates that Shirley has moderate churn risk and the best cross sell offer at this time is a <i>One Month Trial DSL Program offer</i>. Shirley then indicates that she is calling to <i>complain about recent dropped calls</i>. The agent explains that the reason for dropped calls was due to recent network repairs and creates an activity to log the call reason. With the additional call reason, the dashboard updates to indicate that Shirley has become high churn risk and a retention action is necessary. The agent offers a <i>200 Minute Credit on Dropped Calls</i> and Shirley's response is tracked to the campaign and offer.</p>

The SiebelB2C and SiebelB2B metadata have been configured to demo the above scenarios by 1) checking for the demo customers and hard coding the cross sell and retention scenarios and 2) providing a GenerateLearnings entry point for bulk generating customer interactions that train the Oracle RTD learning models.

6.1.2 Bulk customers demo tests

In order to bulk generate the customer interactions, both SiebelB2C and SiebelB2B provide corresponding load generator scripts to invoke the GenerateLearnings advisor with 10,000 unique customer ids. Each call of the script does the following:

- 1) Generate Learnings advisor integration point is invoked by load generator with unique customer id.
- 2) A new session is started.
- 3) The *Session Started* event is logged on the *Churn Indicator Event* choice.
- 4) Customer or Account entity is filled with demo data.
- 5) A random eligible choice under intelligent offers choice group is selected.
- 6) The *Presented* event is logged on the selected offer.
- 7) *IsInterested* is called to check if customer is interested in selected offer.
- 8) If so, the *Interested* event is logged on the selected offer.
- 9) *IsAttrite* is called to check if the customer would churn.
- 10) If so, the *Churn Indicated* event is logged on the *Churn Indicator Event* choice.
- 11) End session on exiting the integration point and commit the learnings.

In order to run the script, perform the following:

- 1) Start the Oracle RTD Load Generator utility by double-clicking on the loadgen.cmd script in *RTD_HOME*\scripts\loadgen.cmd, where *RTD_HOME* is where you had unzipped the Oracle RTD install (ex: C:\OracleBI\RTD\scripts\loadgen.cmd).
- 2) From Load Generator, open the SiebelB2C_LoadGenerator.xml or SiebelB2B_LoadGenerator.xml script provided in the Oracle RTD Applications media pack directory depending on the Siebel Enterprise Server version and the application type (B2B or B2C), for example:

```
\software\Oracle Real-Time Decisions for Siebel Intelligent Offer  
Generation\Siebel8.1\B2C\SiebelB2B_LoadGenerator.xml
```

- 3) Go to the *General* tab and select the clientHttpEndpoints.properties file in the clients folder of the Oracle RTD installation (ex: C:\OracleBI\RTD\clients\). You may need to first edit this file if the Oracle RTD server is on a different machine than the load generator such as the case when Oracle RTD server is on UNIX.
- 4) Review the setup in the Variable and Edit Script tab. The customer or account id should be a sequence of integers with 1. The script should only invoke the Generate Learnings integration point.
- 5) Start the script from the Run tab and let it run 10,000 requests. Verify that the load generator is contacting the intended server and also check for exceptions in the RT server log. If there are exceptions you need to troubleshoot the script.
- 6) Open the Oracle RTD Decision Center UI and log into the latest deployment state of SiebelB2C or SiebelB2B when prompted. Verify that counts and analysis are being displayed for intelligent offer choices and the churn indicator event choice. The drivers and best-fit reports should model the scenarios described above.

6.2 Utilizing analytics data via OLAP tables

Follow the instructions in this section to enable the use of Siebel Analytics / Oracle BI data as part of the decision logic in the IOG Inline Services SiebelB2B and SiebelB2C. The sequence is as follows:

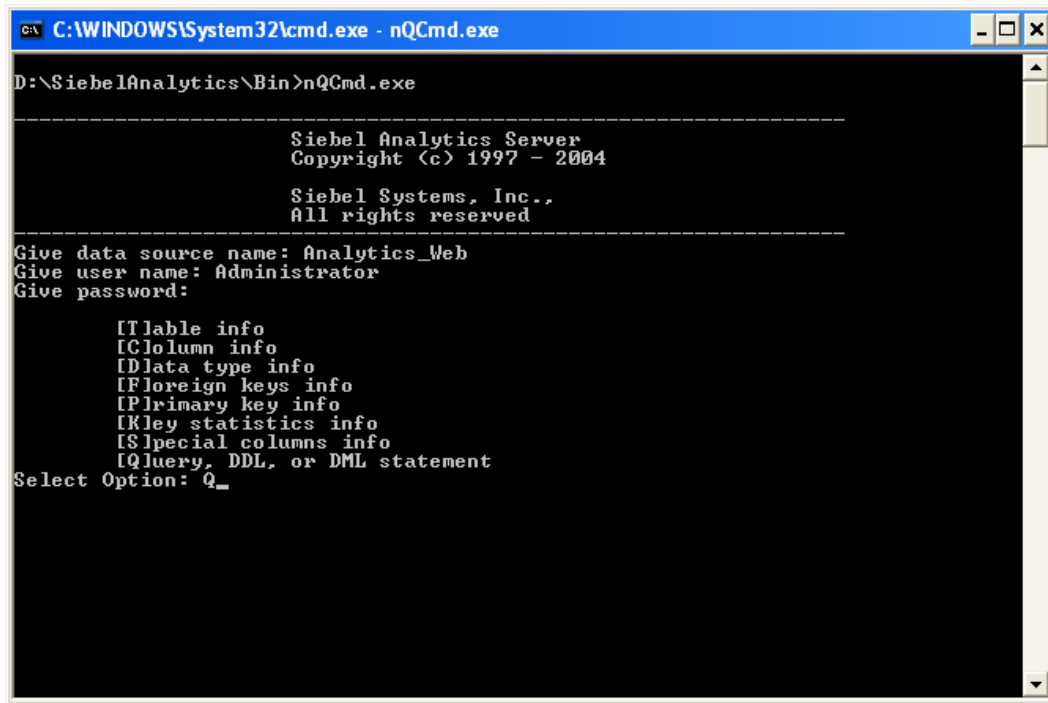
- 1) Create and populate OLAP tables for history entities
- 2) Configure OLAP data source
- 3) Modify IOG Inline Services to include the history attributes

6.2.1 Create and populate the OLAP Tables for History Entities

Siebel Analytics Server / Oracle BI provides behavioral and aggregate metrics that are sourced by Oracle RTD to enrich the customer profile for learning. Siebel Analytics Server / Oracle BI needs to stage the behavioral metrics in a custom OLAP table so that Oracle RTD can access the table using standard DB2, Oracle, SQL Server or Teradata drivers. Siebel Analytics Server / Oracle BI does not support connectivity from a JDBC interface directly. The following steps assume that Analytics Server has already been

installed and is running a Siebel 7.5.x, 7.7.x or 7.8.x RPD against the Siebel RMW. If you are using Oracle BI 10.1.3.2-10.1.3.4, these instructions will still work, please substitute Siebel Analytics paths with Oracle BI paths.

- 1) Using a command window, go to siebelanalytics/bin folder and run nQCmd.exe. Enter the ODBC DSN and administrative username password to connect to the Siebel Analytics Server. Select option [Q]:



- 2) In another window, navigate to the Inline Service specific analytics server query file. This file can be found in the Oracle RTD Applications media pack, in the directory (for example, for Siebel 8.1 and Siebel B2C):

\\software\Oracle Real-Time Decisions for Siebel Intelligent Offer Generation\Siebel8.1\B2C\SQL_ANALYTICS_SERVER.txt

- 3) For each of the POPULATE statements in the file, execute the statement into the nQCmd.exe window. As an example, the statement will retrieve the result set as specified in the logical SQL and insert into the RTD_SRVREQ_HISTORY table through the **Data Warehouse Connection Pool**. The below table lists all the statements:

Service	Entity	OLAP Table	Description
SiebelB2C SiebelB2B	Service Request History	RTD_SRVREQ_HIST	Average Service Quality, # of Critical SRs, Average SR Duration, # of Open SRs
SiebelB2C SiebelB2B	Order History	RTD_ORDER_HIST	% Change Quarter Ago Total Order Revenue, # of Orders

Here an example run of the Service Request History populate statement:

```

C:\WINDOWS\System32\cmd.exe - nQCmd.exe
[Query, DDL, or DML statement
Select Option: Q
Give SQL Statement: POPULATE RTD_SR_HIST MODE <CREATE TABLE CONNECTION POOL "Data Warehouse Connection Pool"> AS SELECT "- Profile".ROW_ID ROW_ID, "- Service Request Facts"."# Assets with Service Requests" NUM_SR_ASSETS, "- Service Request Facts"."Average Service Quality" AUG_SUC_QLTY, "- Service Request Facts"."# of Critical SRs" NUM_CRIT_SR, "- Service Request Facts"."Average SR Duration Days" AUG_SR_DAYS, "- Service Request Facts"."# of Open SRs" NUM_OPEN_SR, "- Service Request History Facts"."% Chg Quarter Ago # of Critical SRs" CRIT_SR_CHG_QTR, "- Service Request History Facts"."Chg Quarter Ago # of Open SRs" OPEN_SR_CHG_QTR FROM "Service Requests"
POPULATE RTD_SR_HIST MODE <CREATE TABLE CONNECTION POOL "Data Warehouse Connection Pool"> AS SELECT "- Profile".ROW_ID ROW_ID, "- Service Request Facts"."# Assets with Service Requests" NUM_SR_ASSETS, "- Service Request Facts"."Average Service Quality" AUG_SUC_QLTY, "- Service Request Facts"."# of Critical SRs" NUM_CRIT_SR, "- Service Request Facts"."Average SR Duration Days" AUG_SR_DAYS, "- Service Request Facts"."# of Open SRs" NUM_OPEN_SR, "- Service Request History Facts"."% Chg Quarter Ago # of Critical SRs" CRIT_SR_CHG_QTR, "- Service Request History Facts"."Chg Quarter Ago # of Open SRs" OPEN_SR_CHG_QTR FROM "Service Requests"
-----
ROW_ID                UM_SR_ASSETS  AUG_SUC_QLTY  NUM_CRIT_SR
AUG_SR_DAYS          NUM_OPEN_SR  CRIT_SR_CHG_QTR  N_SR_CHG_QTR
-----
Row count: 0
-----
Give SQL Statement:

```

- 4) Once the tables have been created, connect to the SIEBEL_OLAP DSN using any SQL tool and verify that the RTD_* tables exist and contain data.

6.2.2 Configure OLAP data source

The following JDBC data source is required in order to utilize analytics data generated in section 6.2.1- *Create and populate the OLAP Tables for History Entities*:

Data Source Name	Type	Description
SIEBEL_OLAP	Source	Used to fill the service request and order history entities using pre-populate OLAP tables.

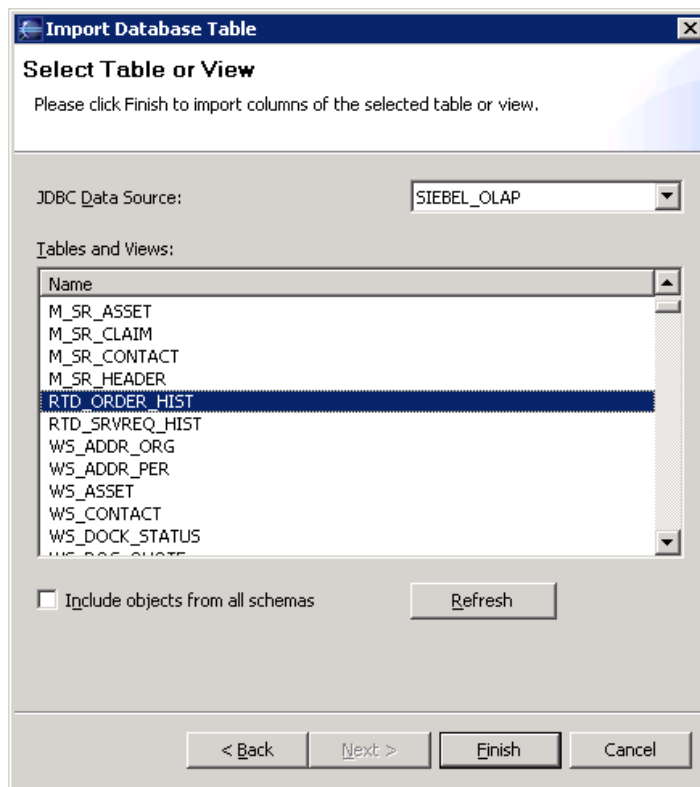
To create the **SIEBEL_OLAP** data source in the application server and RTD resource references to it, refer to the Oracle RTD platform document *Installation and Administration of Oracle RTD*, with section numbers as specified by the following table.

Application server	Section #	Notes
Oracle AS 10.1.3	Section 7.1	In section 7.1.2, steps 11a and 11b: set the data source and JNDI names to SIEBEL_OLAP
IBM WebSphere 6.1	Section 7.2	In section 7.2.1, steps 13b and 13c: set the data source and JNDI names to SIEBEL_OLAP
BEA WebLogic 9.2	Section 7.3	In section 7.3.2, steps 4a and 4b: set the data source and JNDI names to SIEBEL_OLAP

6.2.3 Test the OLAP data source

After you have created the SIEBEL_OLAP data source as described above, verify that the SIEBEL_OLAP data source is accessible by the Oracle RTD server. Start Oracle RTD Decision Studio and make sure SiebelB2B or SiebelB2C is shown in the Inline Service Explorer. For more information on setting up an IOG inline service, see section 3.7 - [Deploying the Siebel IOG Inline Service to the Oracle RTD Server](#).

- 1) Expand the data sources folder and double click on Order History, and then click Import.
- 2) Select the running Oracle RTD server and click Next. If the SIEBEL_OLAP data source was configured correctly, the application displays a list of tables and views with the correct view highlighted for the current data source as shown below:



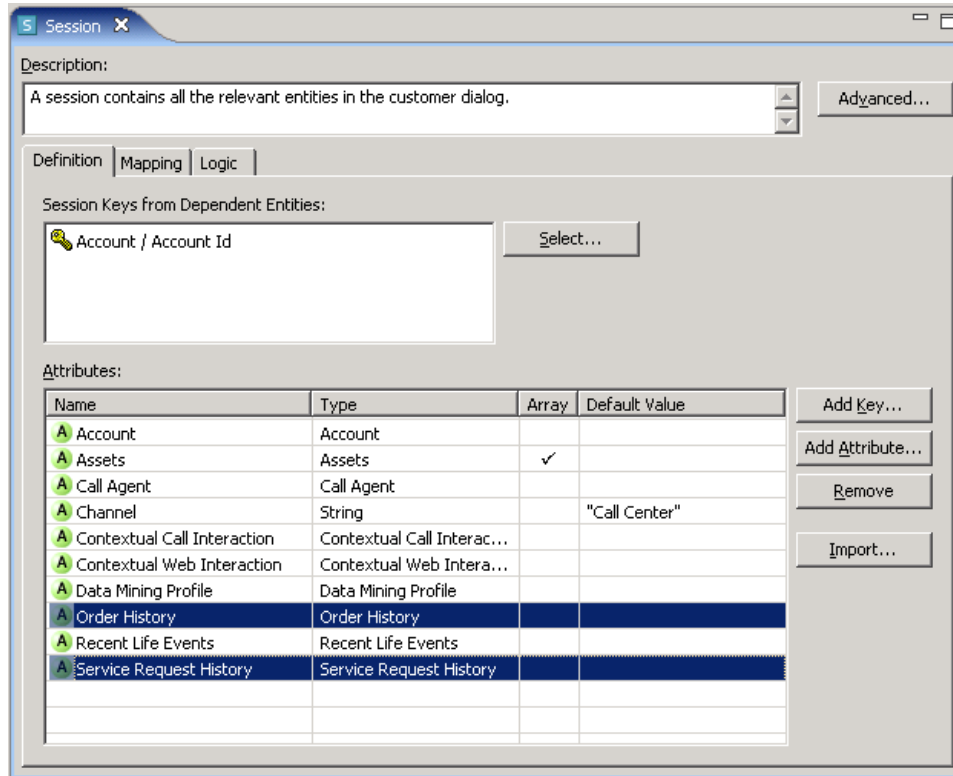
If you get an error message instead, then double check the JDBC data source configuration as described earlier. Click Cancel since there is no need to re-import the data source.

6.2.4 Modify IOG Inline Services to include the history attributes

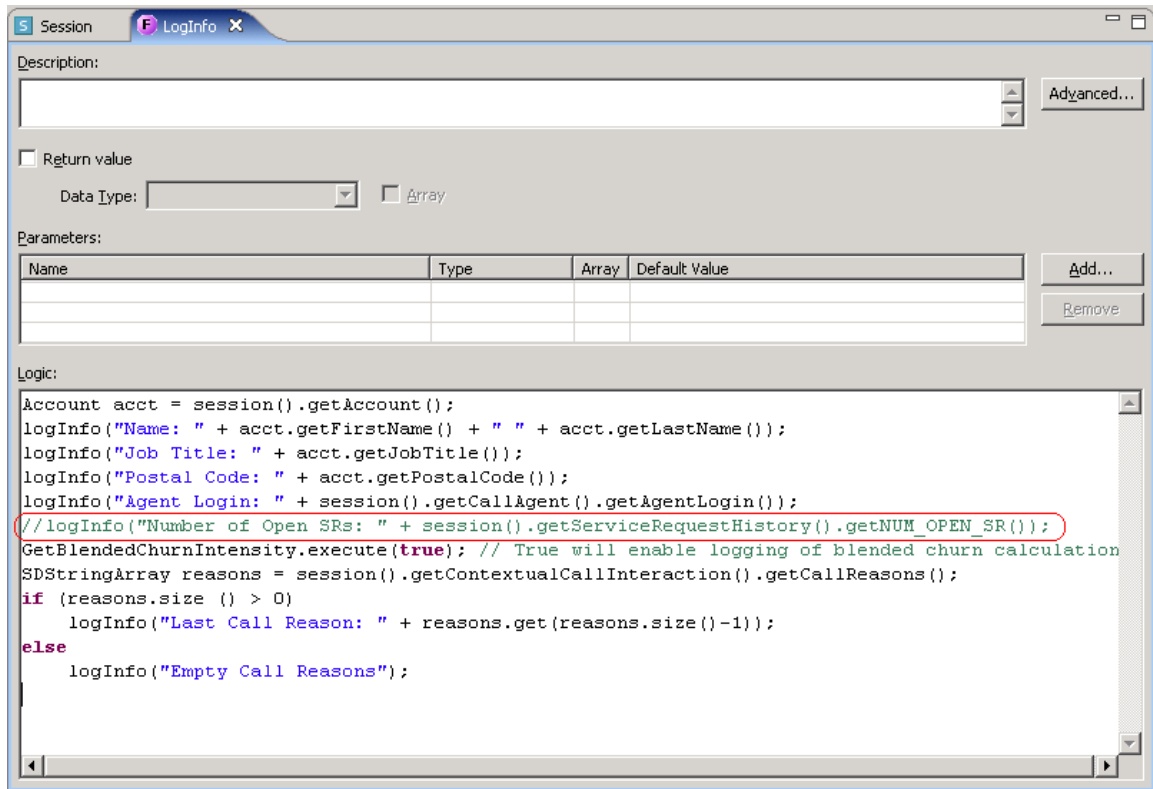
Once you have created the SIEBEL_OLAP data source as described above, several modifications to the IOG Inline Service (SiebelB2B or SiebelB2C) are needed. To make changes to the IOG Inline Service, start Oracle RTD Decision Studio and make sure SiebelB2B or SiebelB2C is shown in the Inline Service Explorer.

- 1) Expand the Entities folder and double click on Session.

- 2) In the Definition tab, click on the button Add Attribute and set the Display Label to **Order History**. The ID will automatically be set to **orderHistory**. In the Data Type dropdown, select **Other**. In the dialog box that pops up, expand the folder Entity Types and select **Order History**. Click OK twice.
- 3) Add another attribute by clicking on the Add Attribute button. Set Display Label to **Service Request History**. The ID will automatically be set to **serviceRequestHistory**. For the Data Type, select entity type of **Service Request History**. Click OK twice. The list of Session attributes should now look like the following:



- 4) Expand the Functions folder and double click on LogInfo. In the Logic section, uncomment the line that prints the number of open service requests:



- 5) Save all changes (Ctrl-Shift-S).
- 6) Redeploy the Inline Service.

7 Appendix

7.1 Appendix A: Siebel IOG user properties

7.1.1 Response Button Disable

Use this BC user property to disable the “Yes! Tell Me More” and “No. Thank You” buttons from below mentioned applets if Treatment name is “No Action Required”.

- RTD Retention Actions Form Applet (B2B)
- RTD Retention Actions Form Applet (B2C)

Value: No Action Required

Parent Object Type: Applicable for following business components,

- RTD Retention Actions (B2B)
- RTD Retention Actions (B2C)

Usage: This user property is used in C++ code to disable the “Yes ! Tell Me More” and “No. Thank You” buttons from Retention applets when Treatment name is “No Action Required”.

Note: If default retention action “Treatment” name changes, then this user property value needs to be updated with appropriate treatment name to disable the buttons.

7.1.2 Interested

Use this property to pass the LIC of Response Type when user clicks on the “**Yes ! Tell Me More**” button.

Value: Requested more info

Parent Object Type: Applicable for following business components,

- RTD Intelligent Offers (B2B)
- RTD Intelligent Offers (B2C)
- RTD Retention Actions (B2B)
- RTD Retention Actions (B2C)

Usage: This user property is used to pass the LIC of Response Type when user clicks on the “**Yes! Tell Me More**” button.

Note: If user adds a new response capture button, a new BC user property needs to be created, ensure that BC user property name should be same as that of “Method Invoked” name, and provide appropriate LIC of Response Type as a user property value.

7.1.3 NotInterested

Use this property to pass the LIC of Response Type when user clicks on the “**No. Thank You**” button.

Value: No Interest

Parent Object Type: Applicable for following business components,

- RTD Intelligent Offers (B2B)
- RTD Intelligent Offers (B2C)
- RTD Retention Actions (B2B)
- RTD Retention Actions (B2C)

Usage: This user property is used to pass the LIC of Response Type when user clicks on the “No. Thank You” button.

Note: If user adds a new response capture button, a new BC user property needs to be created, ensure that BC user property name should be same as that of “Method Invoked” name, and provide appropriate LIC of Response Type as a user property value.

7.1.4 Contact Required Flg

This BC user property is used to make technical decisions within C++ code to allow the response capture for Account if there are contacts associated with it.

Value: TRUE

Parent Object Type: Applicable for following business components,

- RTD Intelligent Offers (B2B)
- RTD Retention Actions (B2B)
- Campaign List Contact
- Response

Usage: This user property is used in C++ code. Responses cannot be created for accounts with no contacts. If user would like to associate responses for account, they need to create a contact first, “Contact Required Flg” BC user property value should always be TRUE.

7.1.5 Reason

This BC user property will be used in C++ code to fetch the field name which holds the value for Reason for rejection.

Value: Reason

Parent Object Type: Applicable for following business components,

- RTD Intelligent Offers (B2B)
- RTD Intelligent Offers (B2C)

- RTD Retention Actions (B2B)

RTD Retention Actions (B2C)

Usage: This BC user property will be used in C++ code to fetch the field name which holds the value for Reason for rejection, if user changes the “Reason” field name then this BC user property should be updated with appropriate field name.

7.2 Appendix B: RTD Internationalization Consideration

There is a known issue regarding the integration between Siebel and RTD when use non-English strings. The offers are not shown correctly if Siebel is non-English while strings defined in RTD is in English.

The root cause of this issue is non-translated strings in RTD Decision Studio which are used in following implementation:

1. Comparison operation under 'Choice Eligibility' rule to ensure treatments with status field – “Enabled” can only participate in selection.
2. Define Group Attribute value for a Choice Group, so that the dynamic choices fall under designated groups i.e. “Acquisition”, ‘Cross-Sell” and “Up-Sell” etc.

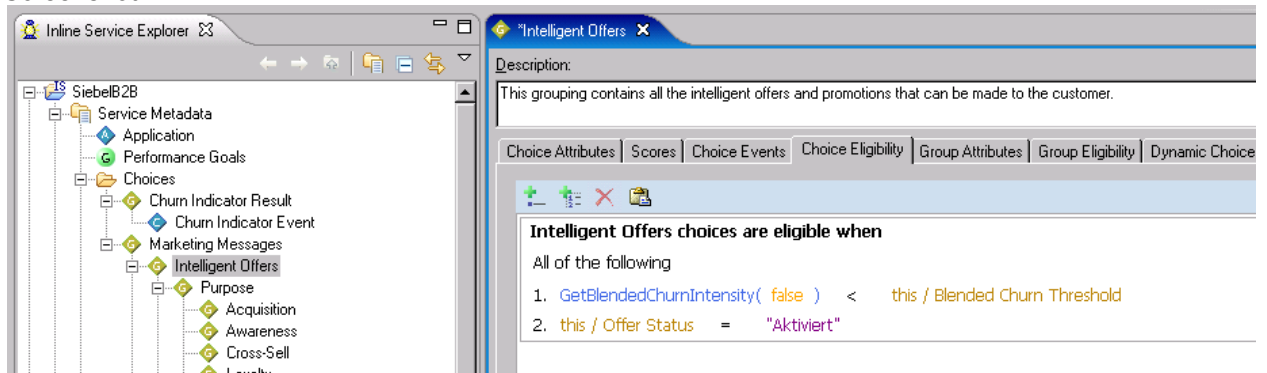
Dynamic choice offer data is pulled from Siebel OLTP database, which is non English in this case (Siebel database is updated with German dump file) and we have only provided English strings in RTD ILS. So, the comparison is failing and none of the offer is selected by RTD hence no offer appears in Siebel UI.

Workaround:

Workaround for this issue is to provide translated strings i.e. language specific strings in RTD Decision studio. These strings should be updated at the following location in SiebelB2B and SiebelB2C Inline Service.

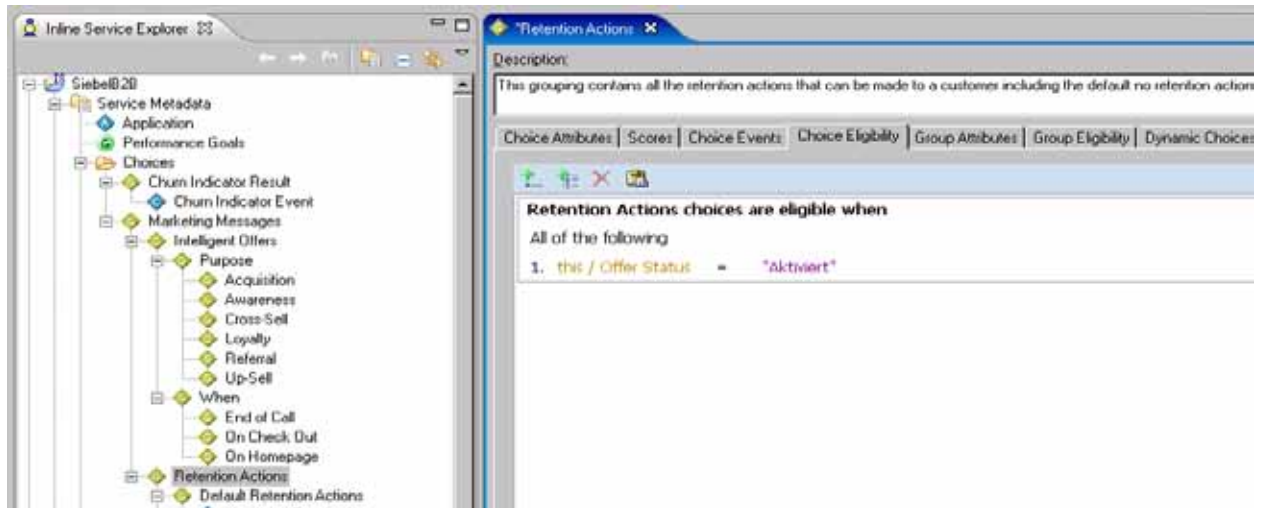
1. Choice Eligibility tab under Intelligent Offers Choice Group:
Edit the eligibility rule and change the comparison value of “this / Offer” Status from “Enabled” to language specific string.
Example: “Aktiviert” in German language.

Screenshot:



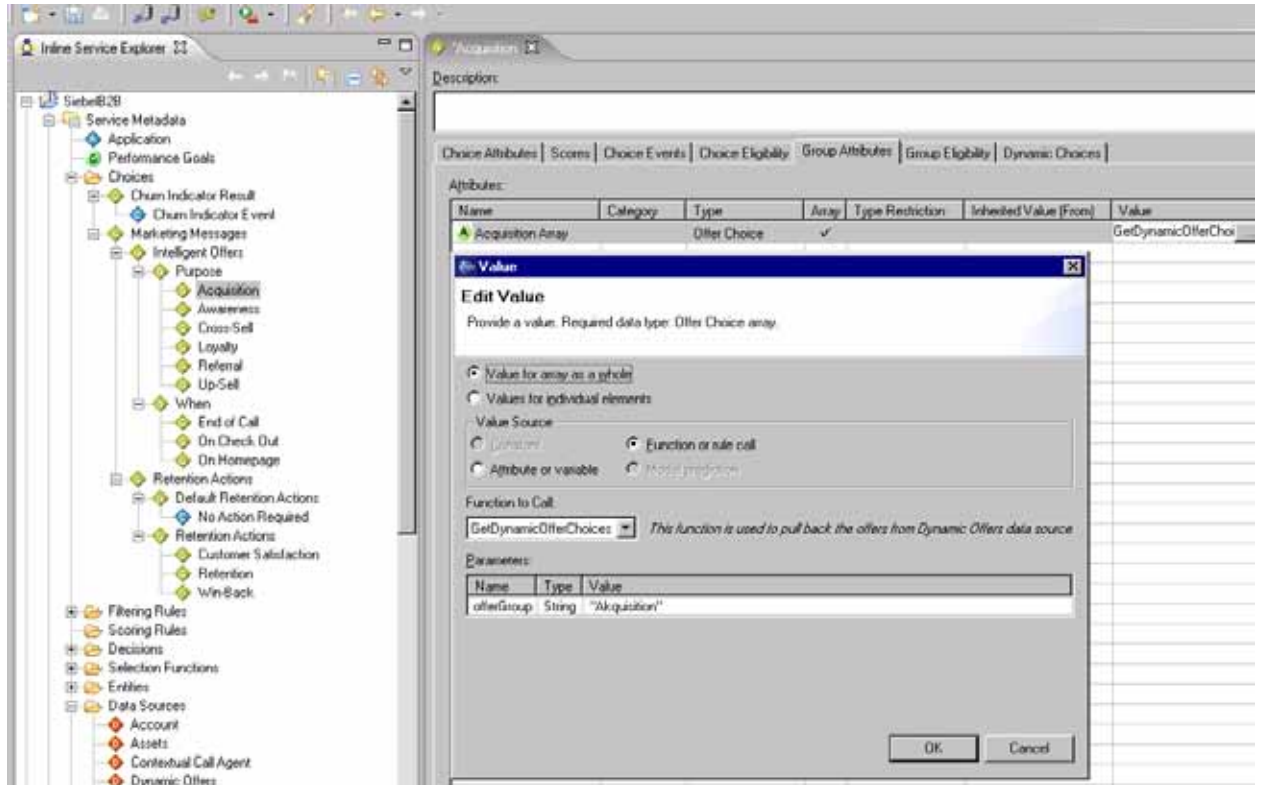
2. Choice Eligibility tab under Retention Actions Choice Group:
Edit eligibility rule by changing the comparison value of “this / Offer Status” from “Enabled” to language specific string,
Example: “Aktiviert” in German language.

Screenshot:



- Group Attributes tab under the Choice Groups having dynamic choices:
 Edit Group Attributes Value section by changing the value of offerGroup parameter to language specific string.
 Example: For Acquisition Choice Group, Acquisition Array attribute value is changed by setting offerGroup parameter value to "Akquisition" in German language

Screenshot:



Similarly, Group Attributes value should be changed for other Choice Groups like Awareness, Cross-Sell, Up-Sell, Loyalty etc with the language specific strings.

