

Oracle® Real-Time Decisions

Release Notes

Release 3.2

E52407-01

June 2014

Oracle Real-Time Decisions Release Notes, Release 3.2

E52407-01

Copyright © 2003, 2014, Oracle and/or its affiliates. All rights reserved.

Primary Author: Oracle Corporation

Contributors: Oracle Business Intelligence development, product management, and quality assurance teams

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

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

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

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

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

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

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

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

Contents

Preface	v
Audience	v
Documentation Accessibility	v
Related Documents	v
Conventions	vi
What's New in Oracle Real-Time Decisions, Release 3.2	vii
New Decision Analytics Reporting	vii
Improved Cluster Management	vii
New Query Service	vii
Improved Learning Service	viii
Learning On Choice Attributes and Transient Session Attributes	viii
Max Input Cardinality/Buckets Defined At the Choice Attribute Level	viii
Scoring Rule Externalization	viii
New Certifications	viii
1 How to Use These Release Notes	
2 Upgrade Steps	
2.1 Upgrade Assumptions	2-1
2.2 Shut Down Oracle RTD Release 3.0.0.1	2-1
2.3 Backup Existing Data and Configuration Files	2-2
2.4 Unpack Oracle RTD Release 3.2	2-2
2.5 Upgrade the Oracle RTD Decision Studio Workplace	2-2
2.6 Upgrade the Oracle RTD Database	2-2
2.7 Upgrade the Application Server (Optional)	2-3
2.8 Install and Configure Oracle RTD 3.2	2-3
2.9 Verify Installation	2-3
2.10 Recompile and Redeploy the Inline Services	2-4
2.11 Upgrade Issues	2-4
2.11.1 XML Response from Advisor Invalid With Special Characters	2-4
2.11.2 Issues in 3.0.0.1 with <Choice Group>.getChoice() Have Been Fixed	2-4

3	General Issues and Workarounds	
3.1	Certification Information	3-1
3.2	Installation Issues.....	3-1
3.2.1	Client Examples Available Only in Windows Installation .Zip Files, Not .CPIO.....	3-1
3.3	General Issues.....	3-1
3.3.1	Ensuring Unique Batch Names Across a Cluster.....	3-2
3.3.2	Transient Likelihood Problems.....	3-2
3.3.3	Issue Opening .csv Files in Excel.....	3-2
3.3.4	Do Not Customize Study Names	3-2
3.3.5	Use Development Deployment State Only	3-2
4	Configuration Issues	
4.1	Decision Service Only Accepts Requests from the Local Host by Default.....	4-1
5	Integration Issues	
5.1	Java Smart Client Initial Timeout	5-1
6	Oracle RTD Decision Studio Issues	
6.1	Mapping Array Attributes in a Multi-Level Entity Hierarchy.....	6-1
6.2	Issue with Oracle JDBC driver version 11.2.0.3.....	6-1
6.3	Caching With a Procedural Data Source Does Not Work	6-1
6.4	Error Importing a Stored Procedure With a Column of Timestamp Data Type	6-1
6.5	ChoiceModelInterface API issues.....	6-2
6.6	Some Log Messages Do Not Show Up In Rtd Studio.....	6-2
6.7	Issues When Trying to Enable Caching for Some Entities.....	6-2
7	Oracle RTD Decision Center Issues	
7.1	Choice Group Performance Count Rollup Reports Not Available for Dynamic Choice Folder Names 7-1	
7.2	IPv6 Support in Decision Center	7-1
8	Externalized Objects Management Issues	
8.1	Improvement to Dynamic Choices Definition.....	8-1
9	Globalization Issues	
9.1	Oracle RTD User Interfaces Only Available in English	9-1
10	Documentation Errata	
10.1	Materialized Views.....	10-1
10.2	Creating Tablespaces.....	10-2

Preface

Oracle Real-Time Decisions (Oracle RTD) enables you to develop adaptive enterprise software solutions. These adaptive solutions continuously learn from business process transactions while they execute and optimize each transaction, in real time, by way of rules and predictive models.

These release notes describe known issues and workarounds for Oracle RTD Release 3.2.

For information about installing Oracle RTD, consult *Oracle Real-Time Decisions Installation and Administration Guide*, which applies to Version 3.2. It is available on the Oracle RTD Documentation Web site:

<http://www.oracle.com/technetwork/middleware/real-time-decisions/documentation/index.html>

Audience

This document is intended for personnel implementing Oracle Real-Time Decisions.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit

<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> or visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> if you are hearing impaired.

Related Documents

For more information, see the following documents in the Oracle Real-Time Decisions Release 3.2 documentation set:

- *Oracle Real-Time Decisions Installation and Administration Guide*
- *Oracle Real-Time Decisions Platform Developer's Guide*
- *Oracle Real-Time Decisions Decision Center User's Guide*

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
<i>italic</i>	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
monospace	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

What's New in Oracle Real-Time Decisions, Release 3.2

Oracle Real-Time Decisions (Oracle RTD), Release 3.2, introduces the following new features:

- [New Decision Analytics Reporting](#)
- [Improved Cluster Management](#)
- [New Query Service](#)
- [Improved Learning Service](#)
- [Learning On Choice Attributes and Transient Session Attributes](#)
- [Max Input Cardinality/Buckets Defined At the Choice Attribute Level](#)
- [Scoring Rule Externalization](#)
- [New Certifications](#)

New Decision Analytics Reporting

RTD 3.2 now has the option to log detailed decision and choice event information in its schema for transformation into a star schema. RTD provides reporting on top of that star schema with OBIEE dashboards.

Improved Cluster Management

RTD now uses a database based topology instead of JGroups. This simplifies RTD cluster configuration and improves stability of large RTD deployments.

New Query Service

RTD has a new service called the Query Service. This service is in charge of providing data for Decision Center and Decision Manager reports. This decouples the handling of Decision Center and Decision Manager Model queries from the active Learning Server and allows the Learning Server to focus on its own tasks. Multiple servers in an RTD cluster can have the Query Service, further enhancing scalability of our reporting infrastructure.

Improved Learning Service

Further scalability improvements have been made on top of the improvements made in 3.0.0.1.11 and later PSUs.

Learning On Choice Attributes and Transient Session Attributes

This feature allows to learn on session data at the time of the event while saving everything in one learning record at the end of the session. This reduces the number of learning records needed without any functionality loss. This in turn allows greater scalability on the learning server side.

Max Input Cardinality/Buckets Defined At the Choice Attribute Level

Max Input Cardinality (for string attributes) and Max Input Buckets (for numeric attributes) can now be overridden at the choice attribute level. This lets people tune how much information is needed in the model at the attribute level.

Scoring Rule Externalization

Scoring Rules can now be externalized in a similar fashion as eligibility and filtering rules.

New Certifications

Refer to the RTD 3.2 certification matrix to see the new certifications in this release. See "Oracle Fusion Middleware Supported System Configurations" for Release 3.2, at:

<http://www.oracle.com/technetwork/middleware/ias/downloads/fusion-certification-100350.html>

How to Use These Release Notes

These release notes are updated periodically as new information becomes available. To ensure that you are reading the latest version of the release notes, check the Oracle RTD Documentation Web site:

<http://www.oracle.com/technetwork/middleware/real-time-decisions/documentation/index.html>

Upgrade Steps

This chapter describes the steps for upgrading to Oracle RTD Release 3.2 from Release 3.0.0.1. It consists of the following topics:

- [Section 2.1, "Upgrade Assumptions"](#)
- [Section 2.2, "Shut Down Oracle RTD Release 3.0.0.1"](#)
- [Section 2.3, "Backup Existing Data and Configuration Files"](#)
- [Section 2.4, "Unpack Oracle RTD Release 3.2"](#)
- [Section 2.5, "Upgrade the Oracle RTD Decision Studio Workplace"](#)
- [Section 2.6, "Upgrade the Oracle RTD Database"](#)
- [Section 2.7, "Upgrade the Application Server \(Optional\)"](#)
- [Section 2.8, "Install and Configure Oracle RTD 3.2"](#)
- [Section 2.9, "Verify Installation"](#)
- [Section 2.10, "Recompile and Redeploy the Inline Services"](#)

2.1 Upgrade Assumptions

This section describes how to upgrade from Oracle RTD Release 3.0.0.1 to Oracle RTD Release 3.2.

The assumptions are:

- Oracle RTD Release 3.0.0.1 was installed at `C:\OracleBI\RTD`.
- The Oracle RTD Release 3.0.0.1 database was initialized.
- The Model Snapshots tables, if used, were initialized.
- Your Inline Services were deployed on Oracle RTD Release 3.0.0.1.
- The Oracle RTD Release 3.0.0.1 Decision Studio workspace is at the default location. On Windows, the default location is `%USERPROFILE%\Oracle RTD Studio`, for example, `C:\Documents and Settings\your user name\Oracle RTD Studio`.

2.2 Shut Down Oracle RTD Release 3.0.0.1

To shut down Oracle RTD Release 3.0.0.1:

1. Exit from Oracle RTD Decision Studio.
2. Exit from Oracle RTD Load Generator.

3. Open your J2EE application server's administration console.
4. Stop Oracle RTD.

2.3 Backup Existing Data and Configuration Files

To back up your existing database and configuration files:

1. Rename C:\OracleBI\RTD\ to C:\OracleBI\RTD.30.
2. Copy C:\Documents and Settings*your user name*\Oracle RTD Studio to C:\Documents and Settings*your user name*\Oracle RTD Studio.30.
3. Back up your Oracle RTD database.

Refer to your database vendor's documentation for information about backing up a database.

2.4 Unpack Oracle RTD Release 3.2

To unpack Oracle RTD Release 3.2:

1. Expand the Oracle RTD Release 3.2 distribution archive to the desired location, for example, C:\OracleBI\RTD.

2.5 Upgrade the Oracle RTD Decision Studio Workplace

To upgrade the Oracle RTD Decision Studio Workplace:

1. Start Oracle RTD Decision Studio.
2. If you have any compile errors, select **Project > Clean > Clean all projects**.

2.6 Upgrade the Oracle RTD Database

If you made changes to `SDTablespaceMap.txt` (located in `RTD_HOME/scripts/sql/Oracle`) in RTD 3.0.0.1, make sure to apply these changes in the RTD 3.2 version you are installing.

To upgrade your Oracle RTD database:

1. Start `SDDDBTool`.

For information about `SDDDBTool`, see *Oracle Real-Time Decisions Installation and Administration Guide*.

2. When the option to Initialize or Upgrade appears, select **Upgrade**.
3. If you had model snapshot tables in Release 3.0.0.1, then, from the `RTD_HOME/scripts` directory, run the command that creates the model snapshot tables:

```
sdexec com.sigmadynamics.tools.SDDDBTool.SDDDBTool -f -u -I InitSnapshotDb.ctl
db_type db_host db_port db_name db_runtime_user db_admin_user db_admin_password
```

Note: If you are using DC_Demo, you need to re-run InitAppDB for DC_Demo, as specified in section 2.4, "Populating the DC_Demo Example Data" in *Oracle Real Time Decisions Installation and Administration Guide*.

2.7 Upgrade the Application Server (Optional)

If necessary, upgrade the application server you are using. Ensure you are running a supported version of the application server.

- To upgrade Oracle WebLogic Server, see *Upgrading Oracle WebLogic Server*.
- To upgrade WebSphere, see the WebSphere Application Server documentation library at:

<http://www-01.ibm.com/software/webservers/appserv/was/library/>

2.8 Install and Configure Oracle RTD 3.2

Next, you will install Oracle RTD 3.2; however, before installing it, you need to undeploy Oracle RTD 3.0.0.1. You will then need to reapply all the changes you have made in the previous version:

- The cluster configuration has changed and no longer uses JGroups. Follow the new cluster setup instructions in *Oracle Real-Time Decisions Installation and Administration Guide*.
- Classpath arguments have changed. Review the new classpath settings in the *Oracle Real-Time Decisions Installation and Administration Guide*.
- JMX MBean attributes have changed location and need to be specified again.
- Custom Roles have to be recreated.
- SSL has to be reconfigured.
- You no longer are required to modify `RTD.ear` in order to add new data sources.
- In order to see log files, configure this with JMX console because, with RTD Release 3.2, we deploy the `RTD.ear` file instead of expanding it.

Install a fresh version of Oracle Real Time Decision 3.2 but point it to the upgraded SDDS, rather than a new one. For instructions, see the *Oracle Real Time Decisions Installation and Administration Guide*.

2.9 Verify Installation

Start Oracle RTD and verify that the inline services are loading properly, by either:

- In the `RTD server.log` file, check that your existing Inline Services loaded successfully, by examining the `[AppFactory]` entries.

For example,

```
2007-11-26 17:42:46,725 INFO [AppFactory] Loaded Inline
Service DC_Demo:4 in deployment state Development.
```

- Alternatively, you can check the status of your Inline Services through JMX. In JConsole, navigate to **Mbeans > Oracle RTD > Inline Service Manager > Inline Service Name**.

Expand the *Inline Service Name* to show the deployment state of interest, and note the Status of the deployment state. If the Status attribute has a value of **Loadable**, then the Inline Service for that deployment state has been loaded properly and can now accept requests. If the server was unable to upgrade and load your existing Inline Service, the Status attribute will show a value of **Failed**.

In this case, you can deploy a working version of the Inline Service from the Decision Studio. If you no longer have the source code for an Inline Service, you can download the Inline Service, using **Project > Download**.

2.10 Recompile and Redeploy the Inline Services

1. In Decision Studio, open up all your Inline Services.
2. Recompile the Inline Services, using **Project > Download**, then **Clean all projects**.
3. Deploy each Inline Service to the RTD Server.

2.11 Upgrade Issues

This section describes issues known to exist with the upgrade processes described in this chapter.

2.11.1 XML Response from Advisor Invalid With Special Characters

Special chars (like "&" or "<") were not properly escaped in the response from an advisor (for example "&" is not transformed to "&") in Releases 3.0.0.1.10 PSU and 3.0.0.1.11 PSU.

If you implemented a workaround of escaping the response in the inline service for one of these versions, you should remove this workaround as part of your RTD 3.2.0.0.0 upgrade.

With this new release, 3.2, if a choice attribute includes XML there is no need to include it in a CDATA section.

Any XML in choice attributes will be escaped in the response to the client such that the client (e.g. Java Smart Client) receives it correctly.

If a CDATA section is included in the choice attribute value, then that CDATA section will also pass through to the client; that is, it will not be removed by the client's XML parser.

2.11.2 Issues in 3.0.0.1 with `<Choice Group>.getChoice()` Have Been Fixed

Issues in Release 3.0.0.1 with `<Choice Group>.getChoice()` have been fixed and the previously recommended workaround of using `<Choice Group>.getPrototype()` is no longer needed if you modify your Inline Service as described above. It is strongly advised you make the recommended changes to your Inline Services as `<Choice Group>.getPrototype()` is an internal method and has its own set of issues when used with a deep choice group hierarchy. For more details, see [Section 8.1, "Improvement to Dynamic Choices Definition"](#).

General Issues and Workarounds

This chapter describes general issues and workarounds for Oracle RTD. It contains the following topics:

- [Section 3.1, "Certification Information"](#)
- [Section 3.2, "Installation Issues"](#)
- [Section 3.3, "General Issues"](#)

3.1 Certification Information

For the latest certification information, which supersedes that in the Oracle RTD documentation, see "Oracle Fusion Middleware Supported System Configurations" for Release 3.2, at:

<http://www.oracle.com/technetwork/middleware/ias/downloads/fusion-certification-100350.html>

3.2 Installation Issues

This section provides release notes on installing Oracle RTD. It contains the following topics:

- [Section 3.2.1, "Client Examples Available Only in Windows Installation .Zip Files, Not .CPIO"](#)

3.2.1 Client Examples Available Only in Windows Installation .Zip Files, Not .CPIO

Client example files are not available in the `.cpio` installation files. They may be retrieved from the Windows installation `.zip` files.

3.3 General Issues

This section contains the following topics:

- [Section 3.3.1, "Ensuring Unique Batch Names Across a Cluster"](#)
- [Section 3.3.2, "Transient Likelihood Problems"](#)
- [Section 3.3.3, "Issue Opening .csv Files in Excel"](#)
- [Section 3.3.4, "Do Not Customize Study Names"](#)
- [Section 3.3.5, "Use Development Deployment State Only"](#)

3.3.1 Ensuring Unique Batch Names Across a Cluster

It is a standard requirement for Inline Service names and batch names to be unique within a cluster.

After an Inline Service is copied within a cluster, batch names registered in the Inline Service can be changed manually to achieve uniqueness. This manual step is not required if the batch registration code in the Inline Service automatically generates the batch name from the new Inline Service name, as in the following generic examples:

```
batchAgent.registerBatch(  
    batchAgent.getInlineService().getName() + ".FeedBackBatchJob",  
    "com.<mycompany>.rtd.batch.FeedBackBatchJob",  
    FeedBackBatchJob.description,  
    FeedBackBatchJob.paramDescriptions,  
    FeedBackBatchJob.paramDefaults);
```

or

```
batchAgent.registerBatch(  
    Application.getApp().getName() + ".FeedBackBatchJob",  
    "com.<mycompany>.rtd.batch.FeedBackBatchJob",  
    FeedBackBatchJob.description,  
    FeedBackBatchJob.paramDescriptions,  
    FeedBackBatchJob.paramDefaults);
```

Note that all programs that may call the batch need to be made aware of the new batch name.

3.3.2 Transient Likelihood Problems

There are cases where non-mature predictive models will return an actual likelihood rather than NaN during the early life of a model. There are also some cases where mature predictive models will return NaN as opposed to actual likelihoods when the Randomize Likelihood option is enabled for a model. These situations are transient in nature and, in the latter case, do not outweigh the benefits of the Randomize Likelihood feature.

3.3.3 Issue Opening .csv Files in Excel

Oracle RTD Platform .csv files use tabs as delimiter instead of comma. Some versions of MS Excel do not recognize the delimiter. To work around this, rename the file as a .txt file and open it again in MS Excel, using the import wizard to specify tab as the delimiter.

Also check the formatting of columns with dates and percentages to make sure they are displayed properly.

3.3.4 Do Not Customize Study Names

When deploying an inline service, do not customize the study name. In particular, two inline services should not share the same study.

3.3.5 Use Development Deployment State Only

Deployment states will be deprecated in a future release. Avoid using any deployment state other than Development.

Configuration Issues

This chapter provides release notes related to Oracle RTD configuration. It contains the following topics:

- [Section 4.1, "Decision Service Only Accepts Requests from the Local Host by Default"](#)

4.1 Decision Service Only Accepts Requests from the Local Host by Default

The Decision Service by default will only accept (integration point) requests from the computer that hosts the Decision Service.

For production environments where Real-Time Decision Server is installed on a separate host, administrators can set up a list of trusted hosts from which the Decision Service can receive requests.

Note: Remote Decision Center access (browser access) is not affected by this feature.

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

1. If you are using 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 *Installation and Administration of Oracle RTD* for more information about accessing JConsole.
2. In the Remote Process section, enter the appropriate port number (typically 12345) and the administrator credentials you created during installation and click **Connect**.
3. Click the **MBean** tab, then go to the **OracleRTD > Cluster > Decision Service** 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. If **RestrictDSClients** is set to `false`, any machine can send Decision Service requests to the RTD server.
4. In the **OracleRTD > Cluster > Decision Service** 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.

Integration Issues

This chapter provides release notes on Oracle RTD integration. It contains the following topics:

- [Section 5.1, "Java Smart Client Initial Timeout"](#)

5.1 Java Smart Client Initial Timeout

The timeout property in `sdclient.properties` is ignored, instead it always defaults to 10 seconds.

Oracle RTD Decision Studio Issues

This chapter provides release notes for Oracle RTD Decision Studio. It contains the following topics:

- [Section 6.1, "Mapping Array Attributes in a Multi-Level Entity Hierarchy"](#)

6.1 Mapping Array Attributes in a Multi-Level Entity Hierarchy

A "parent-child" entity hierarchy is typically defined with the child entity as an attribute of the parent entity. When the child entity is an Array attribute, be careful to map the child Array attribute to its data source in the direct parent entity only.

Assume, for example, a Customer with many Accounts, each of which has many Contracts. In Decision Studio, the Customer entity's Mapping tab may display the complete hierarchy as follows:

- The Customer entity contains the Account entity as an Array attribute
- The Account attribute contains the Contract entity as an Array attribute

The Account entity's Mapping tab contains the Contract entity as an Array attribute.

With this example, only map the Contract attribute in the Account entity's Mapping tab. Do not map the Contract attribute in the Customer entity's Mapping tab.

6.2 Issue with Oracle JDBC driver version 11.2.0.3

When importing a Table in Decision Studio, an Integer column may incorrectly be mapped as a Double column. This is due to a bug in the Oracle JDBC driver version 11.2.0.3. In order to fix this, replace the JDBC driver that your application server uses with version 11.2.0.4.

6.3 Caching With a Procedural Data Source Does Not Work

Caching on the set entity using a table data source works fine. But replacing the table data source with a procedural data source gives errors. A workaround may be to use a table data source on top of a view.

6.4 Error Importing a Stored Procedure With a Column of Timestamp Data Type

Importing a stored procedure from an Oracle database with a column of timestamp data type results in an error. To work around this, use a date data type.

6.5 ChoiceModelInterface API issues

The following choice model methods defined in ChoiceModelInterface do not work and should not be used:

- `public float getChoiceModelQuality(String choiceId)`
- `public int getChoiceModelCount(ModelCount modelCount, String choiceId)`
- `public float getChoiceModelLift(ModelLift modelLift, String choiceId)`
- `public float getChoiceModelError(ModelError modelError, String choiceId)`

6.6 Some Log Messages Do Not Show Up In Rtd Studio

When calling `logInfo()` from a function in your inline service, the log message may not appear in the log window in RTD Studio, but it does appear in the server logs. To work around this, look at the tail of the log file.

6.7 Issues When Trying to Enable Caching for Some Entities

There are known issues, which may result in compilation and runtime errors, with enabling entity caching where a number of design factors coincide:

- You have a multi-level entity hierarchy, and a child entity is an array attribute of a parent entity.
- The data source for the child entity has no Input column defined.
- You want to enable caching on the child entity.
- If possible, simplify the structure of the child entity, specifically the attributes that are mapped to the sources. If this is not possible, contact Oracle Support with details of your particular design configuration.

Oracle RTD Decision Center Issues

This chapter provides release notes for Oracle RTD Decision Center. It contains the following topic:

- [Section 7.1, "Choice Group Performance Count Rollup Reports Not Available for Dynamic Choice Folder Names"](#)
- [Section 7.2, "IPv6 Support in Decision Center"](#)

7.1 Choice Group Performance Count Rollup Reports Not Available for Dynamic Choice Folder Names

For Choice Group Performance Count reports, rollup does not apply to the dynamic choice folder names, whose format is *<dynamic_choice_1>...<dynamic_choice_2>*.

7.2 IPv6 Support in Decision Center

When accessing Decision Center, you cannot use an IPv6 address in the URL. Instead use the host name of this machine.

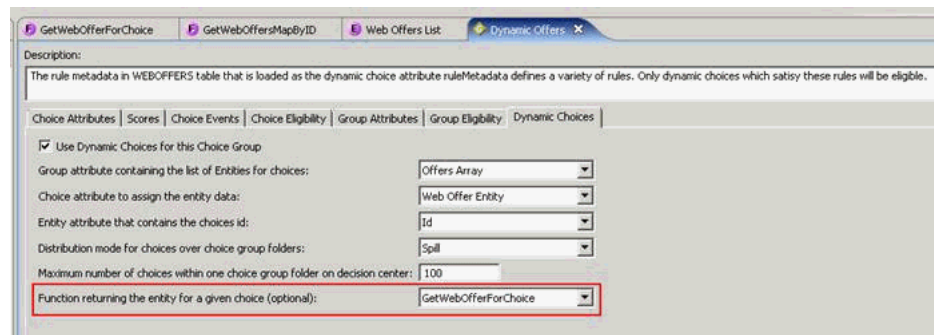
Externalized Objects Management Issues

This chapter provides release notes on externalized objects management issues. It contains the following topics:

- [Section 8.1, "Improvement to Dynamic Choices Definition"](#)

8.1 Improvement to Dynamic Choices Definition

RTD 3.2 introduces a new function which may be configured against a dynamic choice group, indicated in the following screenshot:

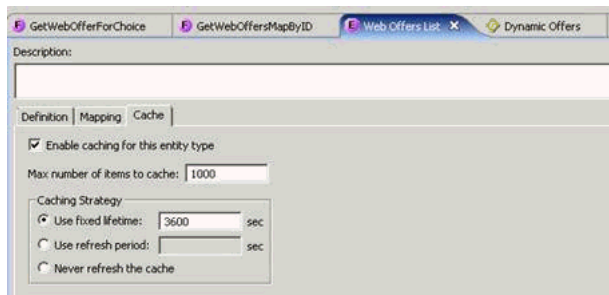


The function is optional. If provided, it must take two String parameters, Choice ID and Choice Group in that order, and return an entity object.

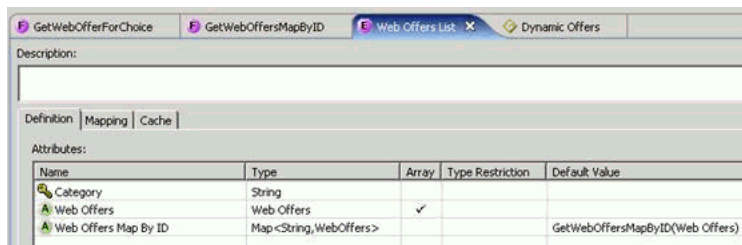
The new function has been introduced as part of fixing `<Choice Group>.getChoice()` for dynamic choices. If the new function is not provided, then `getChoice()` will (a) call the function configured against the list of entities on the Group Attributes tab (b) iterate through all entities to find the matching one. This iteration may be sub-optimal, especially when there is a large number of dynamic choices, hence the possibility for the ILS developer to supply the new function which may be more performant. If the new function is provided, it will be used in preference and should return the entity corresponding to the specified dynamic Choice ID. The Choice Group parameter may help the ILS developer improve the function's performance.

DC_Demo has been extended to include an example implementation of the new dynamic choice function. The changes made are:

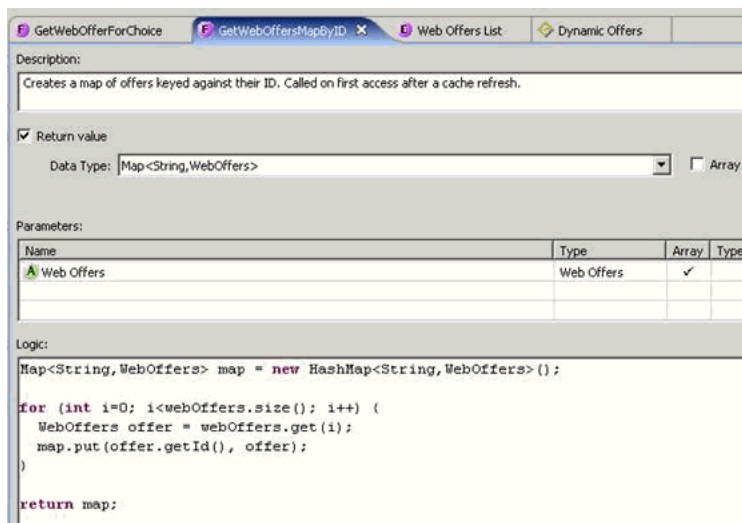
- Caching enabled for the Web Offers List entity:



- A Map attribute added to the Web Offers List attribute, populated on first access after a cache refresh, which provides an efficient way to look up an offer given its ID.



A new function supplies the map on first access:



- A function to return the entity given a Choice ID is provided and configured against the dynamic choice group:

The screenshot displays the configuration for a dynamic choice function. The interface includes a description, return value settings, a parameters table, and a code block.

Description:
Returns the web offer corresponding to a given dynamic choice ID.

Return value:
 Return value
Data Type: Array
Type Restriction:
Call Template:

Parameters:

Name	Type	Array	Type Restriction	Default Value
<input checked="" type="checkbox"/> Choice ID	String			
<input checked="" type="checkbox"/> Choice Group ID	String			

Logic:

```
WebOffersList list = new WebOffersList();  
  
// In practice, the Choice Group ID could be used to determine the category for Web Offers List. Here it is always "DynamicOffersCG"  
list.setCategory("DynamicOffersCG");  
  
return list.getWebOffersMapById().get(choiceId);
```

Globalization Issues

This chapter provides release notes related to Oracle RTD globalization. It contains the following topic:

- [Section 9.1, "Oracle RTD User Interfaces Only Available in English"](#)

9.1 Oracle RTD User Interfaces Only Available in English

Although Oracle RTD supports localized strings in data and metadata, its user interfaces are currently only available in English. In addition, there are several known issues relating to locale-sensitive data, involving the incorrect display format for numbers and dates.

Documentation Errata

This chapter provides information about errata in Oracle RTD documentation.

This section contains the following topics:

- [Section 10.1, "Materialized Views"](#)
- [Section 10.2, "Creating Tablespaces"](#)

10.1 Materialized Views

This section applies to the *Oracle Real-Time Decisions Platform Developer's Guide*. In Section 18.2.5, "Report Schema", a new subsection, 18.2.5.4 "Materialized Views", that contains the following information, should be added:

Four pre-defined materialized views have been created for out of the box deployment:

- MV\$_REC0 - This materialized view caters to the Generic Choice Performance Dashboard and is enabled by default
- MV\$_REC1 - This materialized view caters to Campaign and Creative Performance Dashboards of the Base Marketing application
- MV\$_REC2 - This materialized view caters to Channels, Offers and Placements Performance Dashboards of the Base Marketing application
- MV\$_REC3 - This materialized view caters to Slot and Slot Type Performance Dashboards of the Base Marketing application

Refreshing MV\$_REC1, MV\$_REC2 and MV\$_REC3 is disabled by default. You can enable them by setting the `hierarchyenabled` flag to true in the `SDReportOptions` table of the `rtd_rep` user. This type of refresh is incremental in nature and so for the very first time after enabling the flag, a complete refresh should be performed by the Administrator. Subsequently the refresh of Materialized views are automatic.

The command to perform a complete refresh is:

```
Begin
DBMS_SNAPSHOT.REFRESH( 'MV$_REC1', 'C' );
DBMS_SNAPSHOT.REFRESH( 'MV$_REC2', 'C' );
DBMS_SNAPSHOT.REFRESH( 'MV$_REC3', 'C' );
End;
/
```

10.2 Creating Tablespaces

In Section 2.2.1, "Creating the Oracle RTD Database", of the *Oracle Real Time Decisions Installation and Administration Guide*, the following should be done before the steps in Section 2.2.2, "Initializing the Oracle RTD Database Using SDDDBTool":

Prior to installing this release of Oracle RTD, and if you wish to use database partitions to improve performance, the `CreateTablespaces.sql` script must be run on the Oracle instance, as SYS.

Both the file names for the partitions and the file sizes may be varied. The tablespace names must remain invariant as these names are used in our SQL code.

Additionally, if you are using Oracle Database 12c, set the quota for the tablespaces used by the run-time user to an appropriate level, or to unlimited, depending on your database policies. This is because, unlike 11g, with Oracle Database 12c the Resource role does not include the UNLIMITED TABLESPACE privilege.

The tablespaces created by this script are needed as part of the SDLearning, SDDDecisionLog and SDChoiceEventLog partitioned table definition. If the user does not have the Oracle DB partitioning option enabled and licensed, the SDLearning table can be created without partitioning while running SDDDBTool, thus the creation of tablespaces via this script (`CreateTablespaces.sql`) can be bypassed.

Note: Note that the Oracle DB partitioning option is required for the report schema created in Section 18.3.1, "Post Oracle RTD-Installation Steps" of the *Oracle Real-Time Decisions Platform Developer's Guide*.
