



ORACLE® ENTERPRISE PERFORMANCE
MANAGEMENT SMART SPACE

RELEASE 9.3.1

DEVELOPER'S GUIDE

ORACLE® | Hyperion®

Smart Space Developer's Guide, 9.3.1

Copyright © 2007, Oracle and/or its affiliates. All rights reserved.

Authors: Kathleen Day

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 is a registered trademark 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.

Copyright © 2002-2003 Dominic Cooney Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions: The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software. THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Contents

Chapter 1. Installing Oracle® Enterprise Performance Management Smart Space Development Kit	5
Prerequisites	5
Installing the SDK	6
Uninstalling the SDK	7
Chapter 2. Creating Gadgets	9
Using the Smart Space SDK Interfaces	9
Hello World Example	10
Debugging and Testing	11
Repository Example	12
Building	13
Signing and Publishing	14
Adding Providers	15
Index	17

1

Installing Oracle® Enterprise Performance Management Smart Space Development Kit

In This Chapter

Prerequisites.....	5
Installing the SDK	6
Uninstalling the SDK	7

This guide assumes you are familiar with the following:

- The .NET framework
- Web services development and related standards such as SOAP, and WSDL
- Oracle® Enterprise Performance Management Smart Space features and technical requirements
- Familiarity with C# or VB.NET
- Familiarity with Microsoft® Visual Studio® 2005

For more information on Smart Space, see <http://www.oracle.com/technology/products/bi/epm/smart-space/index.html>.

Prerequisites

To use the Smart Space Development Kit (SDK) you must have these items on your development computer:

- Microsoft® Visual Studio® 2005

Note:

You also need Microsoft® .NET 2.0 which is installed with Visual Studio 2005.

- Administrative privileges
- Smart Space client for debugging and testing. To install, see *Smart Space User's Guide*.
- Smart Space Administration Utility for signing and publishing To install, see *Smart Space Administrator's Guide*.

Note:

Access documentation using this URL:<http://server:port/SmartSpace/Start.html> substitute your server and port in *server:port*.

Installing the SDK

The SDK consists of:

- .NET interfaces
- Gadget templates in C# and VB.NET, which are blank gadgets you start with to create gadgets within Visual Studio.

During installation these items are installed:

Table 1 SDK Installed Files

Items	Description
C:\Program Files\Hyperion Solutions\Hyperion Smart Space GDK	A folder containing Installation dependency files (.NET assemblies) for building gadgets.
Add-in files located in: C:\Documents and Settings\username\My Documents\Visual Studio 2005\Addins <ul style="list-style-type: none">• GadgetPublishingAddin.AddIn• Hyperion.SmartSpace.SDK.GadgetPublishingAddin.dll• Hyperion.SmartSpace.Framework.Utilities.dll	Add-in components, required for the template to execute within Visual Studio.
Project templates located in: C:\Documents and Settings\your user name\My Documents\Visual Studio 2005\Templates\ProjectTemplates <ul style="list-style-type: none">• Visual C#• Visual Basic	Gadget templates for C# and VB.NET.
HKEY_CURRENT_USER\Software\Oracle EPM Smart Space\SmartSpaceSDK	Registry entry to locate SDK install path for dependent assemblies during build process.

Caution!

Close Visual Studio before you install the SDK.

➤ To install the SDK:

- 1 Click this installation URL `http://host:port/SmartSpace/Installations/SDKWizard/Setup.msi` , substitute your server and port into `server:port`.
- 2 Click **Run** to download and start the installation.
- 3 Review the publisher to ensure it is valid and click **Run**.
- 4 From the Welcome screen, click **Next**.
- 5 Enter an installation folder name or accept the default and click **Next**.
- 6 Click **Next** to start the installation.

Wait for the installation to complete.

- 7 From the Installation Complete screen, click **Close**.

➤ To verify the SDK installation:

- 1 Look for the registry entry `HKEY_CURRENT_USER\Software\Oracle EPM Smart Space\SmartSpaceSDK`
- 2 If the entry exists the SDK is installed.

Uninstalling the SDK

Caution!

Close Visual Studio before you uninstall the SDK.

➤ To uninstall the SDK:

- 1 From the Start menu, select **Start > Settings > Control Panel > Add and Remove Programs**.
- 2 Select **Oracle EPM Smart Space Development Kit**.
- 3 Select **Remove**.

2

Creating Gadgets

In This Chapter

Using the Smart Space SDK Interfaces.....	9
Hello World Example.....	10
Debugging and Testing.....	11
Repository Example.....	12
Building.....	13
Signing and Publishing.....	14
Adding Providers.....	15

The SDK enables you to build your own gadgets to access the Smart Space framework and Hyperion back-end applications. For more information on Smart Space, see <http://www.oracle.com/technology/products/bi/epm/smart-space/index.html>.

► To create and publish gadgets:

- 1 Create the gadget, see “Hello World Example” on page 10 and “Repository Example” on page 12.
- 2 Debug and test the gadget, see “Debugging and Testing” on page 11.
- 3 Build the gadget, see “Building” on page 13.
- 4 Sign and publish the gadget, see “Signing and Publishing” on page 14.

Using the Smart Space SDK Interfaces

Detailed interface descriptions are provided in the Help installed with the SDK. To access the help from Visual Studio, select the Help menu item. These are the main Smart Space .NET interfaces:

- Appearance - to modify the gadget’s appearance details, such as size, and to manipulate a gadget’s menus and toolbars.
- Security – to logon, logout, and obtain the user’s SSO token, user-id, and logon name.

Note:

This interface does not expose the user's password.

- Repository – to iterate and manipulate the repository artifacts, including the ability to obtain a SmartCut to launch a file.

Tip:

A SmartCut is a link to a repository item, in URL form.

- Content Viewer – to launch the Content Viewer with a specific URL or SmartCut.
- Toast Messaging – to display toast messages locally.
- Collaboration – to interact programmatically with the Oracle® Enterprise Performance Management Smart Space Collaborator. These features are included:
 - Send & receive messages to individual and multiple users.
 - Obtain a list of users from the current user's selected community.
 - Obtain a list of users from all Smart Space Collaborator users and the ability to search the list, as done in My Community.
 - Detect presence information for the current user or other users in the Oracle® Enterprise Performance Management Smart Space Collaborator.
- Notifications – to send and receive notifications.

Hello World Example

This section describes how to create a C# gadget that displays the toast message “Hello World” on your computer.

► To create the example:

- 1 Open Visual Studio and select **File > New > Project**.
- 2 From Visual C#, **Templates**, click **Smart Space Gadget**.
- 3 **Optional:** Update the gadget name, location and solution name, and click **OK**.
- 4 Enter a name space and click **Finish**.

Tip:

If you change the name space later, you must edit the type field in the `gadgets.xml` file to match.

A blank gadget is created that inherits from the base gadget.

- 5 From the Solution Explorer, double-click `gadget.cs`.

6 Select View > Code.

7 At the top of the file under the existing using statements, add the following 2 lines of code:

```
using Hyperion.SmartSpace.Framework.Model.Shell;  
using Hyperion.SmartSpace.Gdk.Interfaces;
```

Tip:

This code defines the types for the Oracle EPM Smart Space components.

8 Select View > Designer.

9 From the Toolbox, drag a button onto the form and click the button once to select it .

10 In the Properties window, set:

- a. Name to displayToastButton
- b. Text to Display Toast Message
- c. Width to 145

Tip:

Click the Size property, then update the Width.

11 Double-click the display Toast Message button to add an event handler for the Click event.

The Code Editor will open with the insertion point placed within the event handler.

12 Insert this C# code between the brackets:

```
this.Services.Toast.Show("Hello world!");
```

13 To save, select File > Save Gadget.cs.

14 Debug and Test your gadget, see [“Debugging and Testing” on page 11.](#)

Debugging and Testing

Use debug to run your code. Debug automatically starts the Smart Space client with your new gadget, so you can test it.

Tip:

You must have the Smart Space client installed on your computer to debug.

➤ To debug the gadget:

1 Select Start > Programs > Oracle EPM > Oracle EPM Smart Space.

- 2 From the Smart Space client connection screen, enter your user id and password.

Tip:

Close Smart Space and restart it before you start each debug session. Otherwise the system will close Smart Space and prompt you to restart it.

- 3 Setup your breakpoints.
- 4 Select **Debug > Start Debugging**.

Tip:

If you run without debugging, you cannot test your gadget in the Smart Space client.

Your gadget appears in the Smart Space palette.

- 5 Double-click the gadget to add it to a Smart Space desktop and click the **displayToastButton**.

The toast message Hello World! appears in the bottom right corner of your screen.

- 6 Double-click the gadget to add it to a Smart Space desktop, and do one or more of these steps:

- To test the Hello World! example, click **Display Toast Message**.

The toast message Hello World! appears in the bottom right corner of your screen.

- To test the repository example, click **Display Repository Node**.

The toast message The first Node in Reporting and Analysis repository is: SmartSpace appears in the bottom right corner of your screen.

- 7 Select **Debug > Stop Debugging**.
- 8 Do one of these steps:
 - a. Add more functionality to your gadget, see [“Repository Example” on page 12](#).
 - b. To publish the gadget so other users can use it, see [“Building” on page 13](#) and [“Signing and Publishing” on page 14](#).

Repository Example

This section describes how to display the first node name from the Oracle's Hyperion® Reporting and Analysis – System 9 repository using a toast message. You need to create the Hello World! example first.

- To create the example:

- 1 Open Visual Studio and open the gadget you created, see [“Hello World Example” on page 10](#).
- 2 Select **View > Designer**.

- 3 From the Toolbox, drag a button onto the form, and click the button once to select it .
- 4 In the Properties window, set:
 - a. Name to `displayNodeName`
 - b. Text to Display Repository Node
 - c. Width to 145

Tip:

Click the Size property, then update the Width.

- 5 Double-click the button to add an event handler for the Click event.

Tip:

The Code Editor will open with the insertion point placed within the event handler.

- 6 Insert the following C# code between the brackets:

```
IRepository repositoryService = this.Services.Repositories[0];

RepositoryObject firstItem =repositoryService.GetChildren
(repositoryService.RootDirectory) [0];

this.Services.Toast.Show("The first Node in Reporting and Analysis
repository is: " + firstItem.Path);
```

- 7 To save, select **File > Save Gadget.cs**
- 8 Debug and test your gadget, see [“Debugging and Testing” on page 11.](#)

Building

After you create, debug and test a gadget you can build it within Visual Studio, which makes it available for publishing.

➤ To build:

- 1 From Visual Studio **Solution Explorer**, right-click the gadget and select **Build Gadget**.
- 2 Specify a location to store the gadget.

The `gadgetname.SmartSpaceGadget` file is created, where `gadgetname` is the name of your gadget.

- 3 To make this gadget available to users publish it to an existing Smart Space client installation URL, see [“Signing and Publishing” on page 14.](#)

Signing and Publishing

Use the Administration Utility to sign and publish your gadget to an existing Smart Space Client Installation URL.

Note:

To install and configure the Administration Utility, see the *Smart Space Administrator's Guide*.

► To publish and sign gadgets:

- 1 **The Smart Space services must be running on the Smart Space server.**
- 2 **To verify that the services are running on Windows:**
 - a. From the Smart Space server, type Ctrl+Shift+Esc and click the Processes tab.
 - b. Ensure that the services are running:
 - Oracle EPM Smart Space Web Services
 - Oracle EPM Smart Space Collaborator
 - c. If the services are not running, start them:
 - i. Select **Start > Programs > Oracle EPM Smart Space > Start Web Services**.
 - ii. Select **Start > Programs > Oracle EPM Smart Space > Start Collaborator**.
- 3 **For UNIX, verify that the Smart Space services are running and start them if needed.**
- 4 **Select **Start > Programs > Oracle EPM Smart Space > Smart Space Administration Utility**.**

You must have a certificate selected.
- 5 **To select a certificate, click **Settings**, and perform an action:**
 - To use your own certificate:
 - a. Click **Select from Store**.
 - b. Select your certificate, and click **OK**.

Tip:

Use the Certificates snap-in from Microsoft Management Console to install your company certificate.

- To use a test certificate (only for testing and development environments):
 - a. Click **Select from File**, and browse to C:\Program Files\Hyperion Solutions\Smart Space Administration Utility.
 - b. Select `TestCertificate.pfx`, and click **Open**.
 - c. For the password, enter `Test`, and click **OK**.

Note:

Passwords are case-sensitive.

d. Click OK.

6 Select **Sign and Publish** and click **Add**, browse to your gadget file and click **Open**.

7 **Optional:** To automatically install the gadget into the user's Smart Space client, set the **Recommend** checkbox.

8 Click **Settings** and select a certificate.

9 From the action column, select **Publish** and click **Apply**.

The published gadget is copied to the Smart Space server installation.

10 To use the gadget, open your Smart Space client and install the gadget.

Adding Providers

Use the Administration Utility to add a providers; then you can build gadgets to access the providers. A provider is a back-end system or other connection stored globally for access by all users.

Note:

When the Administration Utility connects to Smart Space the Providers tab is updated with the Reporting and Analysis provider which is set up during Smart Space server installation.

For example, suppose you create a provider for external data and you want to build a gadget to access it. You must add your provider and define the name, type, and name value pairs. Then using the Oracle® Enterprise Performance Management Smart Space SDK, you can build a custom gadget to access the provider.

Table 2 Provider and Provider Type Tabs

Field	Description
Provider Name	Unique name to identify the provider
Provider Type	Select a defined type or add a new one: <ul style="list-style-type: none">● Reporting and Analysis - defines the Oracle's Hyperion® Reporting and Analysis – System 9 repository.● Oracle's Hyperion® Essbase® – System 9 – reserved for future.● Define a new type.
Property Name and Default Value	Use the defined name/value pairs or add a new one.

You must set up the provider type first and then add the provider.

- To setup a provider type:
 - 1 From the Administration Utility, click the **Provider Types** tab.
 - 2 Click **Add** to add a provider type name.
 - 3 Click **Add** to add a name for a name/value pair.

- To add a provider:
 - 1 From the Administration Utility, click the **Providers** tab.
 - 2 Click **Add**, then select the provider type and enter a name.
 - 3 Click the **Providers Type** tab, to edit the values of the name/value pairs.

Index

B

building
gadgets, [13](#)
in Visual Studio, [13](#)

C

certificates
installing, [14](#)
selecting, [14](#)
test, [14](#)
creating
gadgets, [9](#)
toast example, [10](#)
creating:
repository example, [12](#)

D

debugging, [11](#)

E

example
SDK, [10](#)
SDK repository, [12](#)

G

gadgets
building, [13](#)
creating, [9](#)
publishing, [14](#)
signing, [14](#)

I

installation
prerequisites, [5](#)
SDK, [5](#)

SDK files, [6](#)
interfaces, [9](#)

P

publishing, [14](#)

R

removing the SDK, [7](#)

S

SDK
debugging and testing, [11](#)
interfaces, [9](#)
repository example, [12](#)
toast example, [10](#)
signing, [14](#)

T

testing, [11](#)

U

uninstalling, [7](#)
uninstalling the SDK, [7](#)

B C D E G I P R S T U