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
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

<table>
<thead>
<tr>
<th>Items</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>C:\Program Files\Hyperion Solutions \Hyperion Smart Space GDK</td>
<td>A folder containing installation dependency files (.NET assemblies) for building gadgets.</td>
</tr>
<tr>
<td>Add-in files located in: C:\Documents and Settings\username\My Documents\Visual Studio 2005\Addins</td>
<td>Add-in components, required for the template to execute within Visual Studio.</td>
</tr>
<tr>
<td>- GadgetPublishingAddin.AddIn</td>
<td></td>
</tr>
<tr>
<td>- Hyperion.SmartSpace.SDK.GadgetPublishingAddin.dll</td>
<td></td>
</tr>
<tr>
<td>- Hyperion.SmartSpace.Framework.Utilities.dll</td>
<td></td>
</tr>
<tr>
<td>Project templates located in: C:\Documents and Settings\your user name\My Documents \Visual Studio 2005\Templates \ProjectTemplates</td>
<td>Gadget templates for C# and VB.NET.</td>
</tr>
<tr>
<td>- Visual C#</td>
<td></td>
</tr>
<tr>
<td>- Visual Basic</td>
<td></td>
</tr>
<tr>
<td>HKEY_CURRENT_USER\Software\Oracle EPM Smart Space\SmartSpaceSDK</td>
<td>Registry entry to locate SDK install path for dependent assemblies during build process.</td>
</tr>
</tbody>
</table>

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.
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:
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:

```csharp
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:

```csharp
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:

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

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

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Provider Name</td>
<td>Unique name to identify the provider</td>
</tr>
<tr>
<td>Provider Type</td>
<td>Select a defined type or add a new one:</td>
</tr>
<tr>
<td></td>
<td>● Reporting and Analysis - defines the Oracle's Hyperion® Reporting and Analysis - System 9 repository.</td>
</tr>
<tr>
<td></td>
<td>● Oracle's Hyperion® Essbase® - System 9 - reserved for future.</td>
</tr>
<tr>
<td></td>
<td>● Define a new type.</td>
</tr>
<tr>
<td>Property Name and Default Value</td>
<td>Use the defined name/value pairs or add a new one.</td>
</tr>
</tbody>
</table>

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

certificates
  installing, 14
  selecting, 14
  test, 14
creating
  gadgets, 9
toast example, 10
creating:
  repository example, 12

debugging, 11

e
example
  SDK, 10
  SDK repository, 12

g
gadgets
  building, 13
  creating, 9
  publishing, 14
  signing, 14

installation
  prerequisites, 5
  SDK, 5

P
publishing, 14

R
removing the SDK, 7

S
SDK
  debugging and testing, 11
  interfaces, 9
  repository example, 12
  toast example, 10
  signing, 14

testing, 11

U
uninstalling, 7
uninstalling the SDK, 7