Appeon Developer User Guide

Appeon® 6.1 for PowerBuilder®

DOCUMENT ID: DC20238-01-0610-01

LAST REVISED: October 2008

Appeon Corporation, 1/F, Shell Industrial Building, 12 Lee Chung Street, Chai Wan District, Hong Kong.

Contents

1 About This Book

1.1 Audience

This book is written for PowerBuilder developers using Appeon Developer.

Important:

For audience who want to switch deployment options in Appeon Developer 6.1 like in Appeon Developer 3.x or 5.x: Appeon has announced Pure-JavaScript deployment option is end of life at the release of Appeon Developer 6.1, therefore, only Appeon Xcelerator deployment option can be used and it is always automatically used from now on.

Contents related with Pure-JavaScript deployment option has been removed from this guide, therefore, this guide is for Appeon Xcelerator deployment option only from now on, athough the text "Appeon Xcelerator" is not explicitly mentioned throughout the document.

1.2 How to use this book

There are sixteen chapters in this book.

Chapter 1: About This Book

A general description of this book.

Chapter 2: Introduction

An overview of Appeon Developer.

Chapter 3: Before You Begin

Some important instructions that you MUST follow before using Appeon Developer.

Chapter 4: Configuring Appeon Developer

Instructions for configuring Appeon Developer.

Chapter 5: Using UFA Tool

Instructions for analyzing the unsupported features in a PowerBuilder application.

Chapter 6: Deploying PowerBuilder Applications

Instructions for deploying PowerBuilder applications to the Web.

Chapter 7: Debugging Appeon Web Applications

Instructions for using Appeon Debugger to debug an Appeon Web application.

Chapter 8: Running Web Applications

Instructions for running Web applications after deployment.

Chapter 9: Using Information Manager

Instructions for viewing reports and log files generated in the Appeon migration process.

Chapter 10: Packaging and Installing Web Applications

Instructions for using Web application packaging tool to generate installation packages.

Chapter 11: Undeploying Web Applications

Instructions on how to undeploy Appeon Web applications.

Chapter 12: Developing with Code Insight

Instructions on how to use the Appeon Code Insight feature.

Chapter 13: Launching Appeon Enterprise Manager

Instructions for launching Appeon Enterprise Manager from Appeon Developer.

Chapter 14: Appeon for PowerBuilder Help

Instructions for launching the Appeon HTML help file.

Chapter 15: Extended Toolkit

Instructions for using the extended tools provided by Appeon: the DLL/OCX Files Package tool, the Appeon Silent Installer, and the Appeon Developer Toolbar Register tool.

Chapter 16: Technical Support

Important information on technical support.

1.3 Related documents

Appeon provides the following user documents to assist you in understanding Appeon for PowerBuilder and its capabilities:

- *Appeon Demo Applications Tutorial*:

Introduces Appeon's demo applications, including the Appeon Sales Application Demo, Appeon Code Examples, Appeon ACF Demo, and Appeon Pet World, which show Appeon's capability in converting PowerBuilder applications to the Web.

- *Appeon Developer User Guide* (or *Working with Appeon Developer Toolbar*)

Provides instructions on how to use the Appeon Developer toolbar in Appeon 6.1.

Working with Appeon Developer Toolbar is an HTML version of the *Appeon Developer User Guide*.

- *Appeon Server Configuration Guide*

Provides instructions on how to configure Appeon Server Status Monitor, establish connections between Appeon Server and Database Server, and configure AEM for maintaining Appeon Server and Appeon deployed Web applications.

- *Appeon Supported Features Guide* (or *Appeon Features Help*):

Provides a detailed list of what PowerBuilder features are supported and can be converted to the Web with Appeon 6.1 and what features are unsupported.

Appeon Features Help is an HTML version of the *Appeon Supported Features Guide*.

- *Appeon Installation Guide*:

Provides instructions on how to install *Appeon for PowerBuilder* successfully.

- *Appeon Migration Guide*:

A process-oriented guide that illustrates the complete diagram of the Appeon Web migration procedure and various topics related to steps in the procedure, and includes a tutorial that walks the user through the entire process of deploying a small PowerBuilder application to the Web.

- *Appeon Performance Tuning Guide*:

Provides instructions on how to modify a PowerBuilder application to achieve better performance with its *corresponding Web application*.

- *Appeon Troubleshooting Guide*:

Provides information about troubleshooting issues, covering topics such as product installation, Web deployment, AEM, Web application runtime, etc.

- *Introduction to Appeon*:

Guides you through all the documents included in Appeon 6.1 for PowerBuilder.

- *New Features Guide* (or *What's New in Appeon*):

Introduces new features and changes in Appeon 6.1 for PowerBuilder.

What's New in Appeon is an HTML version of the *New Features Guide*.

1.4 If you need help

Each Sybase installation that has purchased a support contract has one or more designated people who are authorized to contact Sybase Technical Support, or an Authorized Sybase Support Partner. If you have any questions about this installation, or if you need assistance during the installation process, ask a designated person to contact Sybase Technical Support, or an Authorized Sybase Support Partner based on your support contract. You may access the Technical Support Web site at http://www.sybase.com/support.

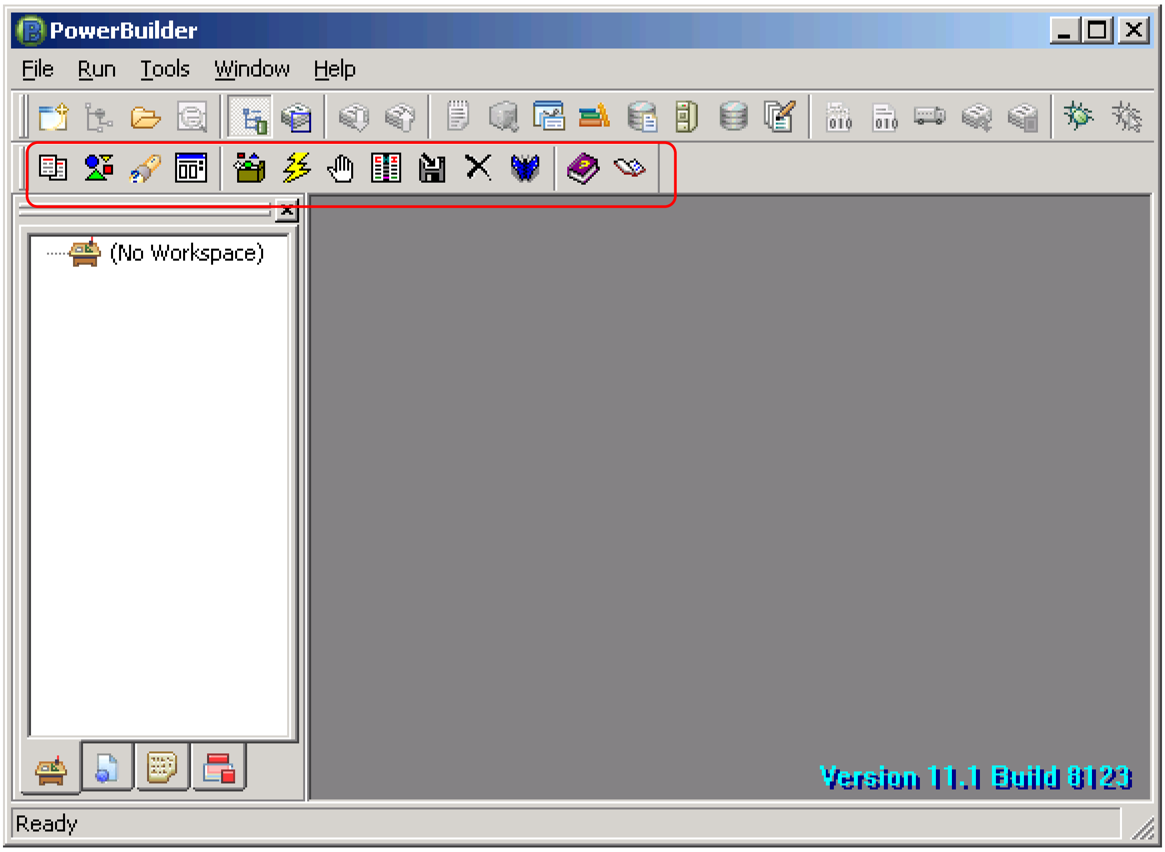
2 Introduction

2.1 Overview

Appeon Developer, a component of Appeon for PowerBuilder, extends the capabilities of PowerBuilder, allowing a new or existing PowerBuilder application to be converted into a *bona fide* Web application using only PowerBuilder skills.

Appeon Developer provides a set of tools that enable the entire PowerBuilder-to-Web process to take place within the PowerBuilder IDE. These tools are accessed via a toolbar in the PowerBuilder IDE, which automatically loads each time PowerBuilder is opened.

Figure 2-1: Appeon Developer toolbar embedded in the PowerBuilder IDE



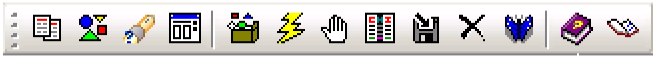
Appeon Developer converts a PowerBuilder application by parsing the PBLs (source code) of PowerBuilder applications and generating a set of corresponding HTML, JavaScript and XML files. When deployed by Appeon Server, these generated files form an exact Web-based replica of the source PowerBuilder application. Any user can open a standard Microsoft Web browser and access the Web-based version of the original PowerBuilder application over the Internet, an intranet, or an extranet.

2.2 Appeon Developer toolbar in PowerBuilder

All Appeon Developer tools conform to the US Government Section 508 Accessibility Guidelines.

Figure 2-2 represents the user interface of the Appeon Developer toolbar.

Figure 2-2: Appeon Developer toolbar



The Appeon Developer toolbar has 13 buttons. The functionality of each button is described in Table 2-1.

Table 2-1: Appeon Developer toolbar functions

Button	Name	Function
📄	Config Wizard	Provides a wizard for quick configuration of Appeon Developer for Web conversion.
⚙	Configure	Provides configuration and management of Appeon Developer for Web conversion.
🔍 or 🔧	Analyze	Analyzes application source code for unsupported features before deployment. This analysis can cover the whole application, at object level or within an inheritance hierarchy. (It displays as 🔧 in PowerBuilder 11.5)
▦	Code Insight	Develops PowerBuilder applications that are free of Appeon unsupported features.
📦	Deploy	Starts the complete process of deploying a PowerBuilder application to the Web.
⚡	Run	Runs the deployed Web application in Internet Explorer.
✋	Appeon Debugger	Debugs the Appeon Web application.
▥	Information	Manages logs and reports.
💾	Package	Packages application files for installation.
✕	Undeploy	Undeploys Web applications from Web Server(s) and Appeon Server(s).
🦋	AEM	Launches Appeon Enterprise Manager (AEM).
📕	Help	Searches, browses, prints, copies and pastes useful information in the Appeon user guides.
🤝	Get Support	Displays the support information.

Apart from Code Insight and Information, only one Appeon Developer toolbar function can be launched at any given time. This prevents an Appeon Developer process from interfering with another.

2.3 Web application development with Appeon Developer

For either developing a new Web application, or migrating an existing PowerBuilder application onto the Web, you need to perform three key tasks in Appeon Developer: Analyze, Modify and Deploy Automatically.

If you use Appeon for PowerBuilder for developing a new Web application, the first step is to write a new PowerBuilder application (refer to Chapter 2: *Web RAD with Appeon for PowerBuilder* in the *Appeon Migration Guide*). It is recommended that the new PowerBuilder application conforms to Appeon's coding styles, as laid out in *Appeon Features Help*, then you will expend less effort in the first two tasks: Analyze and Modify.

- Task 1 – Analyze

  You must analyze the application PBLs for unsupported PowerBuilder objects and features, using the PowerBuilder IDE extended with Appeon Developer. General rules of thumb regarding how the application should be structured are outlined in *Appeon Features Help* under *Basic and Architectural Requirements*. The application must meet these requirements before the analysis can begin.

  The first step results in an analysis report highlighting unsupported PowerBuilder objects and code within the application.

- Task 2 – Modify

  Next, work around or remove unsupported PowerBuilder objects and code that prevent the Web application from running, using standard PowerBuilder programming. *Appeon Features Help* and *Appeon Code Examples* (an Appeon demo PowerBuilder application) will guide you through this process by providing information regarding supported PowerBuilder features as well as example code for implementing these features.

  Then, perform a "full build" of the application in PowerBuilder, ensure that there are no bugs in the PowerBuilder code and that the application functions correctly.

  The second task results in a PowerBuilder application that is ready for automatic conversion to the Web.

- Task 3 – Deploy Automatically

  At the push of a button, you can now automatically generate a precise replica of the PowerBuilder application, that deploys to the Web n-Tier architecture and can be accessed by standard Internet Explorer Web browsers.

  Lastly, deploy the generated Web application files to Appeon Server. The third task results in a *bona fide* Web application with the look and feel of the source PowerBuilder application.

For detailed information about Web conversion, refer to the *Appeon Migration Guide*.

3 Before You Begin

3.1 Installing Appeon for PowerBuilder

Follow the *Appeon Installation Guide* carefully; make sure that both Appeon Developer and Appeon Server have been installed.

Verify that Appeon Developer has been installed to the Developer PC.

Verify that Appeon Server has been installed to the application server, such as, Sybase EAServer, BEA WebLogic or IBM WebSphere. Once Appeon Server has been installed, the machine hosting Appeon Server is then referred to as Appeon Server.

If using a separate Web Server (e.g. Apache) instead of the application server built-in Web server, verify that the Appeon Server Web Component has been installed to the Web Server.

Verify that all system requirements have been met for the Developer PC, Web Server, Appeon Server, and the Database Server. Refer to the *Appeon Installation Guide* for details.

3.2 Configuring application database connectivity

Perform the following database configurations for all prospective PowerBuilder applications that will be deployed with Appeon Developer:

1. Set up the database used by the PowerBuilder application to interface with Appeon Server and the JDBC driver. Refer to the documentation from the appropriate database vendor for instructions.
2. Create the connection cache in Appeon Server. Appeon provides systematic instructions for certified database systems. Refer to the *Appeon Server Configuration Guide* for more information.
3. Map the transaction object in the target PowerBuilder application to the newly created connection cache for the prospective Web application. You can either dynamically set up the mapping via PowerScript following the instructions in the *Appeon Server Configuration Guide* or establish the mapping statically in Appeon Developer or AEM.

3.3 Starting Appeon Server

Appeon Developer interacts with Appeon Server and Web server during Web deployment. Web applications are deployed to one or more Appeon Servers and one or more Web servers. You should verify that the Appeon Server and Web server are running before deployment. If you use the application server built-in Web server as the Web server, you only need to start application server/Appeon Server. For example, if you use EAServer as the Appeon Server and EAServer HTTP Server as the Web server, you only need to start EAServer/Appeon Server. Otherwise, start both the Web server and Appeon Server. Refer to the documents provided by the Web server vendor for how to start the Web server, and refer to the following instructions to start the Appeon Server.

Table 3-1: How to start Appeon Server

Appeon Server	Operating System	Detailed Instructions
EAServer	Windows	Select Windows Start | Programs | Appeon 6.1 for PowerBuilder | Appeon Server | *InstanceName*.
	UNIX/Linux	Run the appeonserverstart.sh file in the $appeon/bin/ folder, for example: $ $JAGUAR/appeon/bin/appeonserverstart.sh You can run EAServer/Appeon Server in different modes, debug and normal, using different Java runtime versions and different Java VMs. For details on how to specify the mode options in the above syntax, refer to the EAServer System Administration Guide.
WebLogic	Windows	Select Windows Start | Programs | Appeon 6.1 for PowerBuilder | Appeon Server for WebLogic | Start WebLogic
	UNIX/Linux	Run the appeonserver.sh file in the $appeon/bin/ folder, for example: $ $BEA_HOME/user_projects/domains/mydomain/appeon/bin/appeonserver.sh
WebSphere	Windows	Select Windows Start | Programs | Appeon 6.1 for PowerBuilder | Appeon Server for WebSphere | Start WebSphere
	UNIX/Linux	Run the appeonserver.sh file in the $appeon/bin/ folder, for example: $ $WebSphere/AppServer/appeon/bin/appeonserver.sh

4 Configuring Appeon Developer

4.1 Overview

The settings configured in Appeon Developer are critical; they are used throughout the entire PowerBuilder-to-Web process. Appeon Developer settings determine which PowerBuilder application will be converted to the Web, and the manner in which it will be deployed.

Before you attempt to use any other functionality on the toolbar, complete the following tasks:

1. Set up an application profile for each of the PowerBuilder applications intended for conversion. Each application profile tells Appeon Developer important information about the application, such as which PBLs compose the PowerBuilder application, the database type, etc.

   During the application profile setup, set up a profile for the database type used by the application. This enables Appeon Developer to generate the correct database syntax.
2. Set up at least one Appeon Server profile and one Web Server profile. This enables Appeon Developer to utilize the Appeon Server and Web Server for deployment.
3. Set up at least one deployment profile, which links at least one Appeon Server and one Web Server together. This tells Appeon Developer where to deploy the Web application.

Appeon provides a *Config Wizard* () to quickly configure an application. With your specifications, the wizard creates the application profile, database type profile, Appeon Server profile, Web server profile, connection cache profile, and transaction object mapping. After creation, all these profiles can be modified in the *Configure* tool (). Besides modifying the above profile settings, the *Configure* tool provides you with advanced settings, such as performance settings, parsing options, Web Services profiles, runtime settings etc.

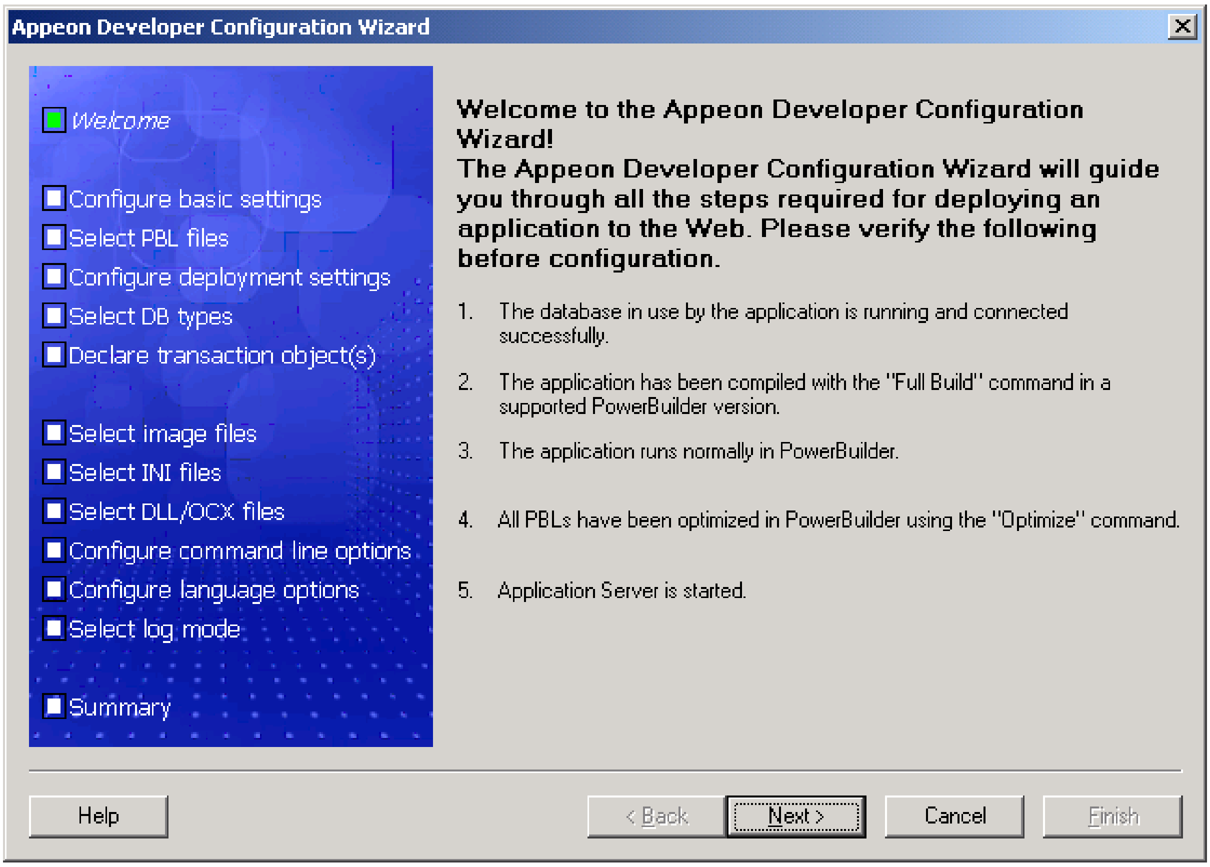
4.2 Using Appeon Developer Configuration Wizard

The settings in Appeon Developer Configuration Wizard are the same as those in the *Configure* tool, therefore, only basic description will be provided in this section. For detailed description, refer to Section 4.3: *Using Configure Tool*.

Click the *Config Wizard* button () on the Appeon Developer toolbar to launch the Appeon Developer Configuration Wizard.

Carefully read the notes and requirements on the Welcome page of the Appeon Developer Configuration wizard (Figure 4-1). Click *Next* to proceed.

Figure 4-1: Welcome page



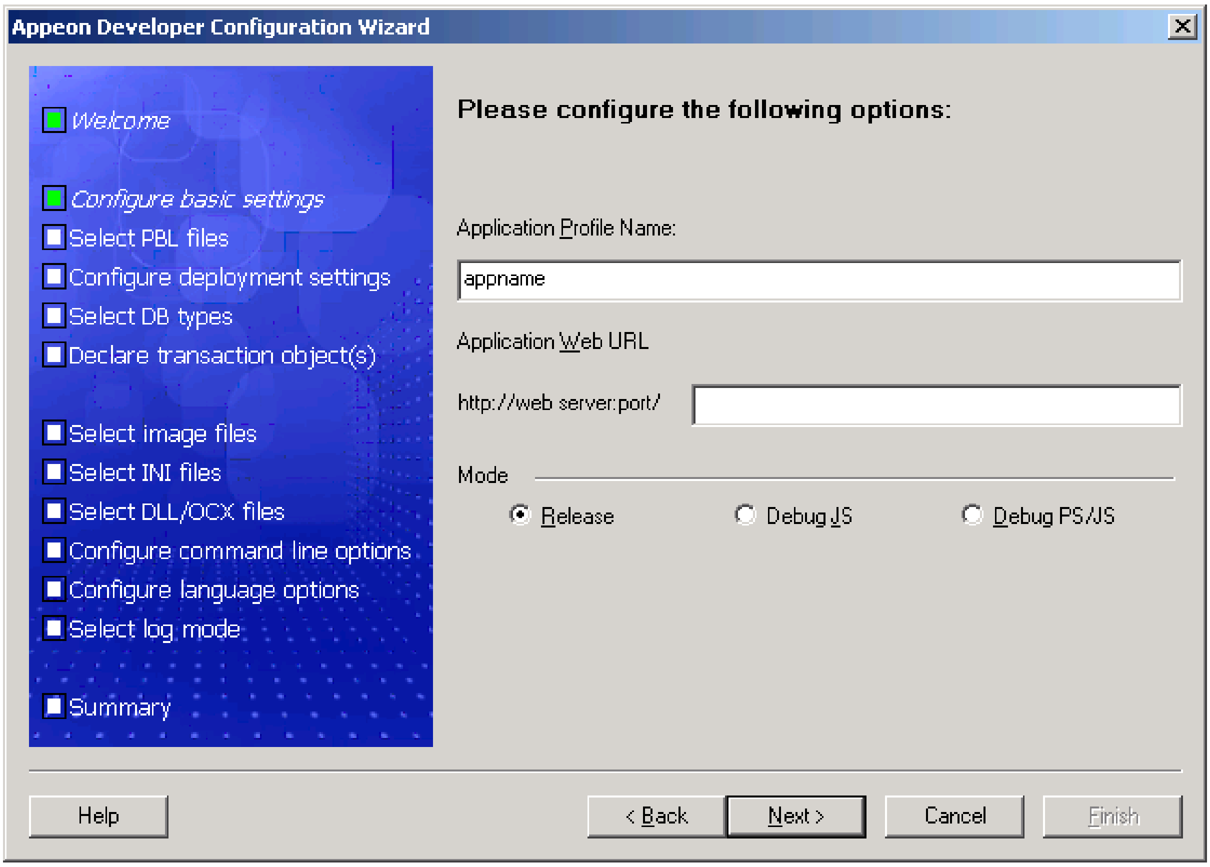
Welcome to the Appeon Developer Configuration Wizard!
The Appeon Developer Configuration Wizard will guide you through all the steps required for deploying an application to the Web. Please verify the following before configuration.

1. The database in use by the application is running and connected successfully.
2. The application has been compiled with the "Full Build" command in a supported PowerBuilder version.
3. The application runs normally in PowerBuilder.
4. All PBLs have been optimized in PowerBuilder using the "Optimize" command.
5. Application Server is started.

4.2.1 Configuring basic settings

The basic settings refer to the settings of an application that are essential for deployment, including the application profile name, Web URL, and file generation mode, as shown in Figure 4-2. For more information about them, refer to the *Basic Settings* section in the *Using Configure Tool* section.

Figure 4-2: Configure basic settings of an application



Please configure the following options:

Application Profile Name: appname

Application Web URL

http://web server:port/

Mode

● Release ○ Debug JS ○ Debug PS/JS

Step 1 – Specify the application profile name.

The application profile name is used to identify the PowerBuilder application during the entire PowerBuilder-to-Web process.

Step 2 – Specify the application Web URL.

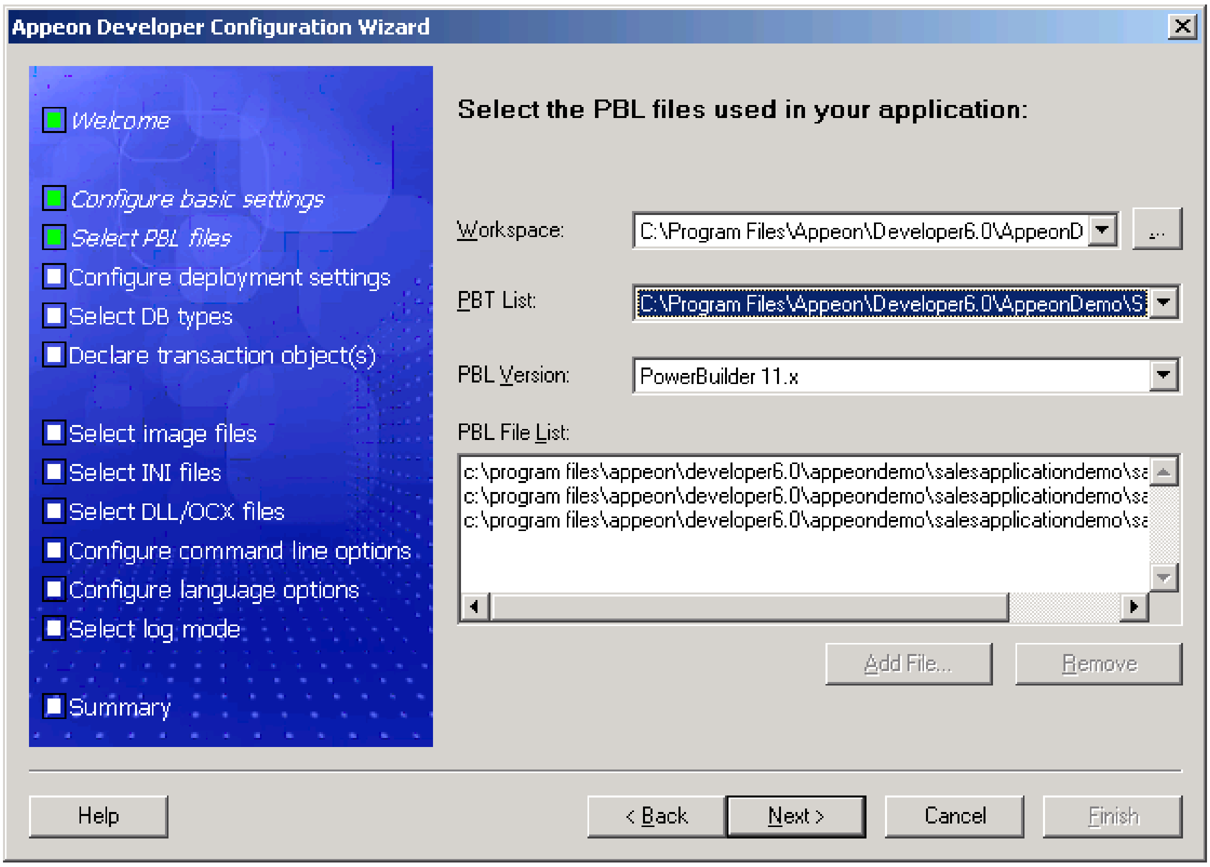
The specified text is also used as the name of the folder created under the Web root of the Web server for storing the Web application files when the application is deployed.

Step 3 – Choose a Web file generation mode.

4.2.2 Selecting PBL file(s)

Specify the version and the location of the PowerBuilder application source code (Figure 4-3).

Figure 4-3: Select PBLs



Appeon Developer Configuration Wizard

Select the PBL files used in your application:

Workspace: C:\Program Files\Appeon\Developer6.0\AppeonD

PBT List: C:\Program Files\Appeon\Developer6.0\AppeonDemo\S

PBL Version: PowerBuilder 11.x

PBL File List:

c:\program files\appeon\developer6.0\appeondemo\salesapplicationdemo\sa
c:\program files\appeon\developer6.0\appeondemo\salesapplicationdemo\sa
c:\program files\appeon\developer6.0\appeondemo\salesapplicationdemo\sa

You can add application PBLs using one of the following methods:

Method 1 – Select the workspace and the PBT file to automatically add PBL files. Detailed steps are as follows:

- Step 1 – Select the workspace. Select a recent workspace from the dropdown list, or click the ellipsis button to select a workspace.

  When a workspace is selected, all PBT files contained in the selected workspace are added to the PBT List.

- Step 2 – Select the PBT file.

  When a PBT file is selected, all PBL files contained in the selected PBT are added to the PBL File List.

This is the recommended method to add PBLs, as it reduces the likelihood of forgetting to add a required PBL.

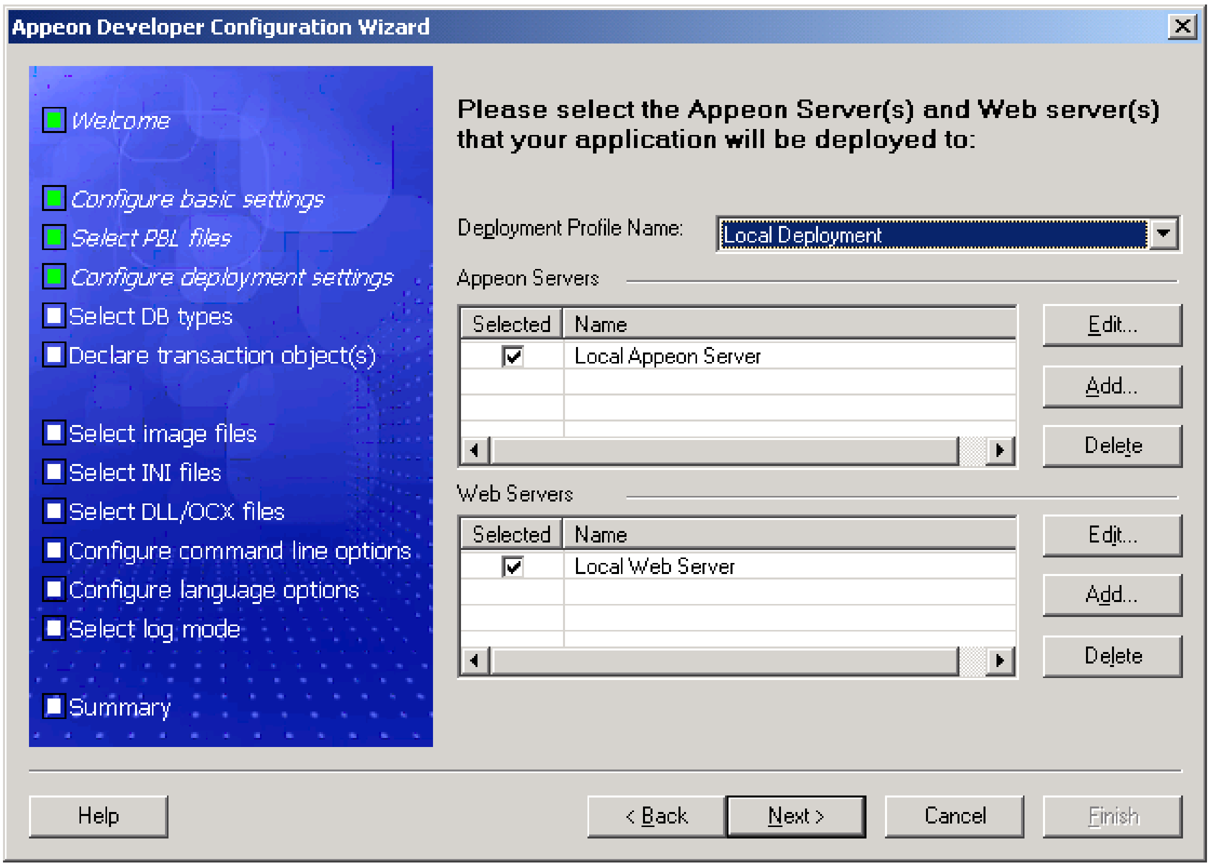
Method 2: You can also add PBL files without specifying the workspace and PBT file first.

- Click *Add File* to add one or multiple PBL files. This is the only method you can take when using the *Configure* tool. Refer to the *Basic Settings* section for detailed instructions.

4.2.3 Configuring deployment settings

The deployment settings associate the Appeon Server(s) and Web server(s) as a group used for application deployment (Figure 4-4).

Figure 4-4: Select the Appeon Server and Web server



Appeon Developer Configuration Wizard

Please select the Appeon Server(s) and Web server(s) that your application will be deployed to:

Deployment Profile Name: Local Deployment

Appeon Servers

Selected	Name
☑	Local Appeon Server

Web Servers

Selected	Name
☑	Local Web Server

Step 1 – Select an existing deployment profile.

To create or modify a deployment profile, you must use the *Configure* tool. For detailed instructions, refer to Section 4.3.4: *Managing deployment profiles*.

Step 2 – Select the check boxes to include the Appeon Server(s) and the Web server(s) in the deployment profile.

All the Appeon Server profiles and Web server profiles you have created are listed. More than one Appeon Server and one Web server can be selected.

If the required Appeon Server profiles or Web server profiles are not available, click the *Add* button to create them. Refer to Section 4.3.3.b: *Appeon Server profile settings* for detailed instructions on creating an Appeon Server profile and refer to Section 4.3.3.c: *Web Server profile settings* for detailed instructions on creating a Web server profile.

Step 3 – Click *Next* to proceed.

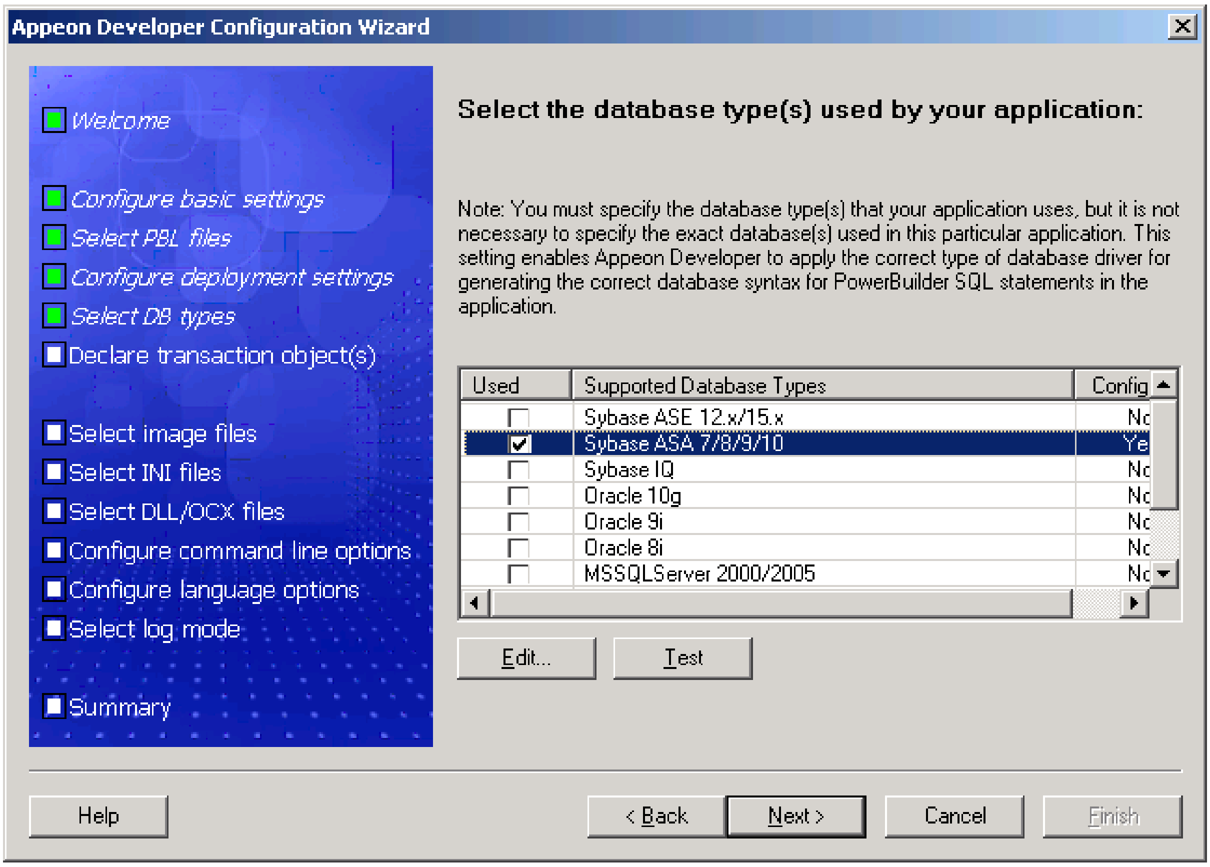
When *Next* is clicked, the wizard will test the connection to the Appeon Server and the Web server. If connection is successful, the deployment profile is saved with the associated Appeon Server(s) and Web server(s).

4.2.4 Selecting DB Type(s)

The database type is required during the application deployment if the application connects to one or more databases.

Select the database type(s) used by the application by selecting the *Used* column of the database type, as shown in Figure 4-5.

Figure 4-5: Database types



Appeon Developer Configuration Wizard

- 🟩 *Welcome*
- 🟩 *Configure basic settings*
- 🟩 *Select PBL files*
- 🟩 *Configure deployment settings*
- 🟩 *Select DB types*
- □ Declare transaction object(s)
- □ Select image files
- □ Select INI files
- □ Select DLL/OCX files
- □ Configure command line options
- □ Configure language options
- □ Select log mode
- □ Summary

Select the database type(s) used by your application:

Note: You must specify the database type(s) that your application uses, but it is not necessary to specify the exact database(s) used in this particular application. This setting enables Appeon Developer to apply the correct type of database driver for generating the correct database syntax for PowerBuilder SQL statements in the application.

Used	Supported Database Types	Config
□	Sybase ASE 12.x/15.x	Nc
☑	Sybase ASA 7/8/9/10	Ye
□	Sybase IQ	Nc
□	Oracle 10g	Nc
□	Oracle 9i	Nc
□	Oracle 8i	Nc
□	MSSQLServer 2000/2005	Nc

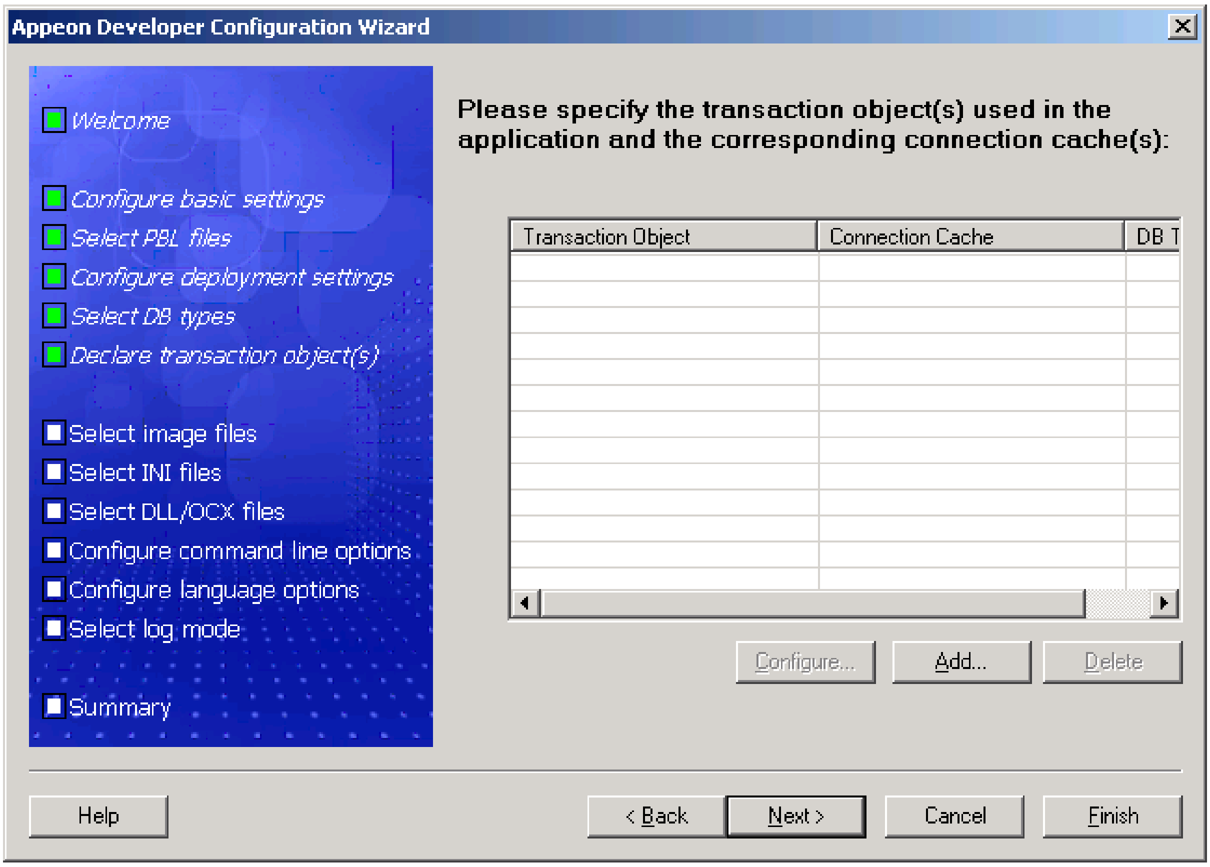
If the database type is not configured (The *Configured* column is indicated with “No”), you should select it and click the *Edit* button to create a profile for it. For detailed instructions on creating a database type profile, refer to Section 4.3.2: *Managing database type profiles*.

4.2.5 Declaring transaction object(s)

When the application is deployed to the Web, Appeon Server handles the database connection using connection caches rather than transaction objects defined in the PowerBuilder application. You must associate the transaction objects used by the PowerBuilder application with proper connection caches created in Appeon Server.

You can create, modify or delete the mapping between transaction objects and connection caches either in Appeon Developer or AEM.

Figure 4-6: Specify transaction objects



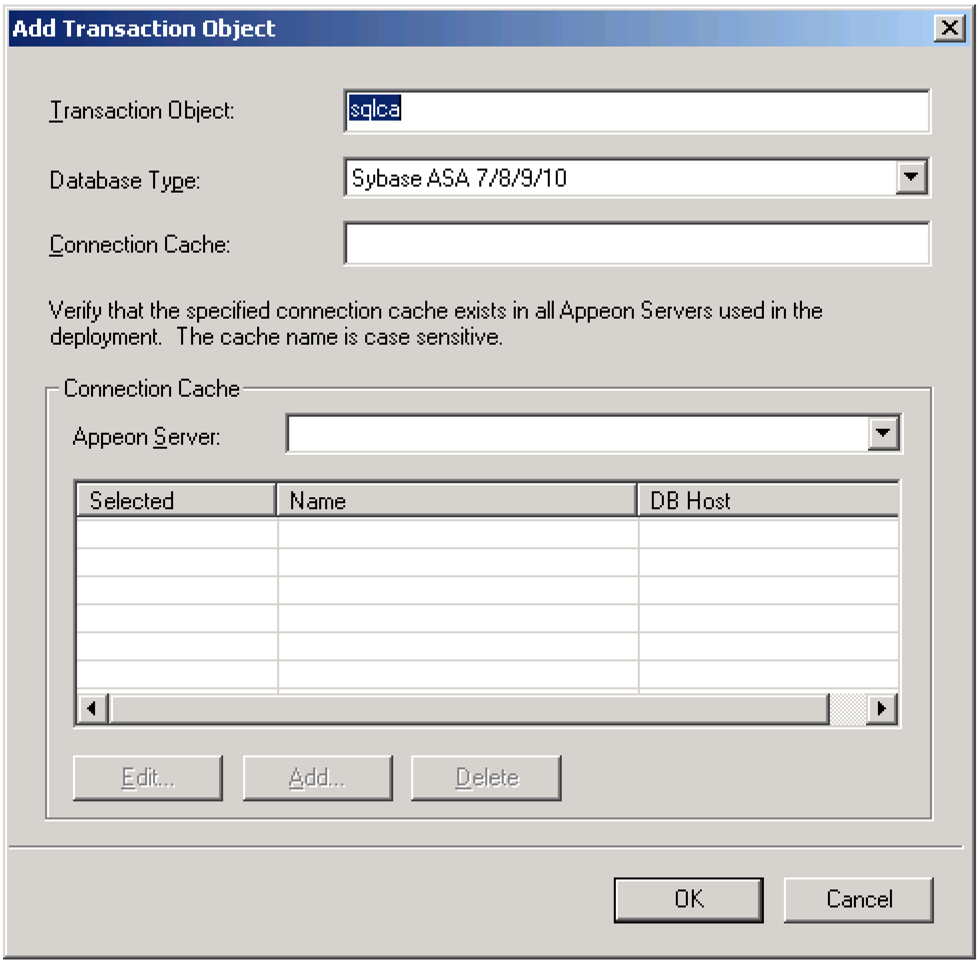
Appeon Developer Configuration Wizard
☒
🟩 Welcome
🟩 Configure basic settings
🟩 Select PBL files
🟩 Configure deployment settings
🟩 Select DB types
🟩 Declare transaction object(s)
□ Select image files
□ Select INI files
□ Select DLL/OCX files
□ Configure command line options
□ Configure language options
□ Select log mode
□ Summary
Configure...
Add...
Delete
Help
< Back
Next >
Cancel
Finish

Please specify the transaction object(s) used in the application and the corresponding connection cache(s):

Transaction Object	Connection Cache	DB T

Click *Add* to create the mapping of the transaction object and the connection cache of an application, as shown in Figure 4-7.

Figure 4-7: Add a transaction object



Add Transaction Object

Transaction Object: sqlca

Database Type: Sybase ASA 7/8/9/10

Connection Cache:

Verify that the specified connection cache exists in all Appeon Servers used in the deployment. The cache name is case sensitive.

Step 1 – Input the transaction object name used by the application to the *Transaction Object* text box.

Step 2 – Select the database type from the *Database Type* list box.

Step 3 – Specify the connection cache name in the *Connection Cache* text box. The connection cache should connect to the same database that the transaction object connects to.

Make sure the connection cache exists in all Appeon Servers selected for this application deployment. If the connection cache does not exist, you can create it by following instructions in Section 4.3.5: *Managing connection cache profiles*. After you create a connection cache, you can click the *Selected* column to associate the connection cache with the transaction object.

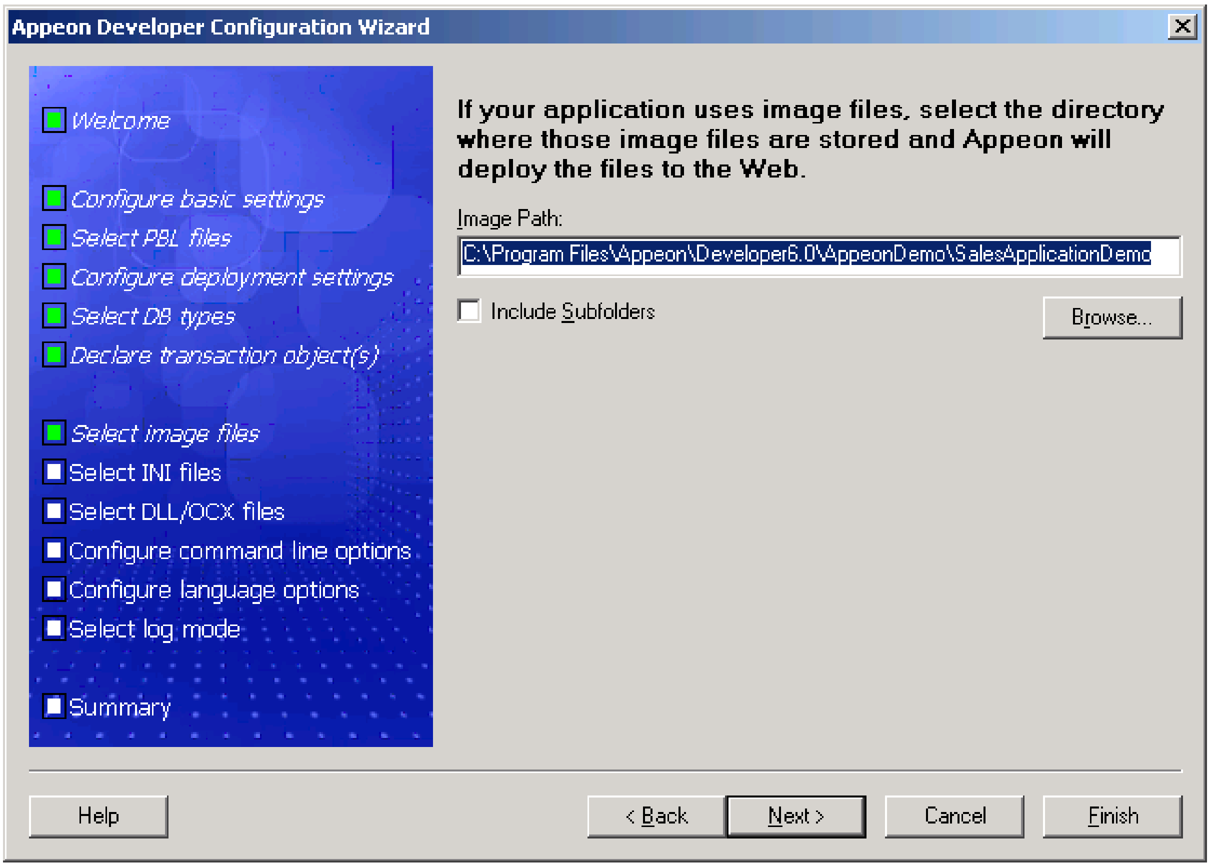
Note: The Connection Cache tool will be available if Appeon Server is installed to EAServer 5.x, and not be available if Appeon Server is installed to EAServer 6.x, WebLogic, or WebSphere. You should go to the corresponding application server administration console to configure the data source for EAServer 6.x, WebLogic, or WebSphere. For detailed instructions, refer to the *Database Connection Setup* chapter in the *Appeon Server Configuration Guide*.

4.2.6 Selecting image files

Click *Browse* to specify which folder contains the image files that will be used in the Web application. Select *Include Subfolders* to deploy the sub-folders under the specified directory (Figure 4-8).

For detailed information about the image files, refer to the *Deploy Images* section.

Figure 4-8: Select image files



Appeon Developer Configuration Wizard

If your application uses image files, select the directory where those image files are stored and Appeon will deploy the files to the Web.

Image Path: C:\Program Files\Appeon\Developer6.0\AppeonDemo\SalesApplicationDemo

□ Include Subfolders

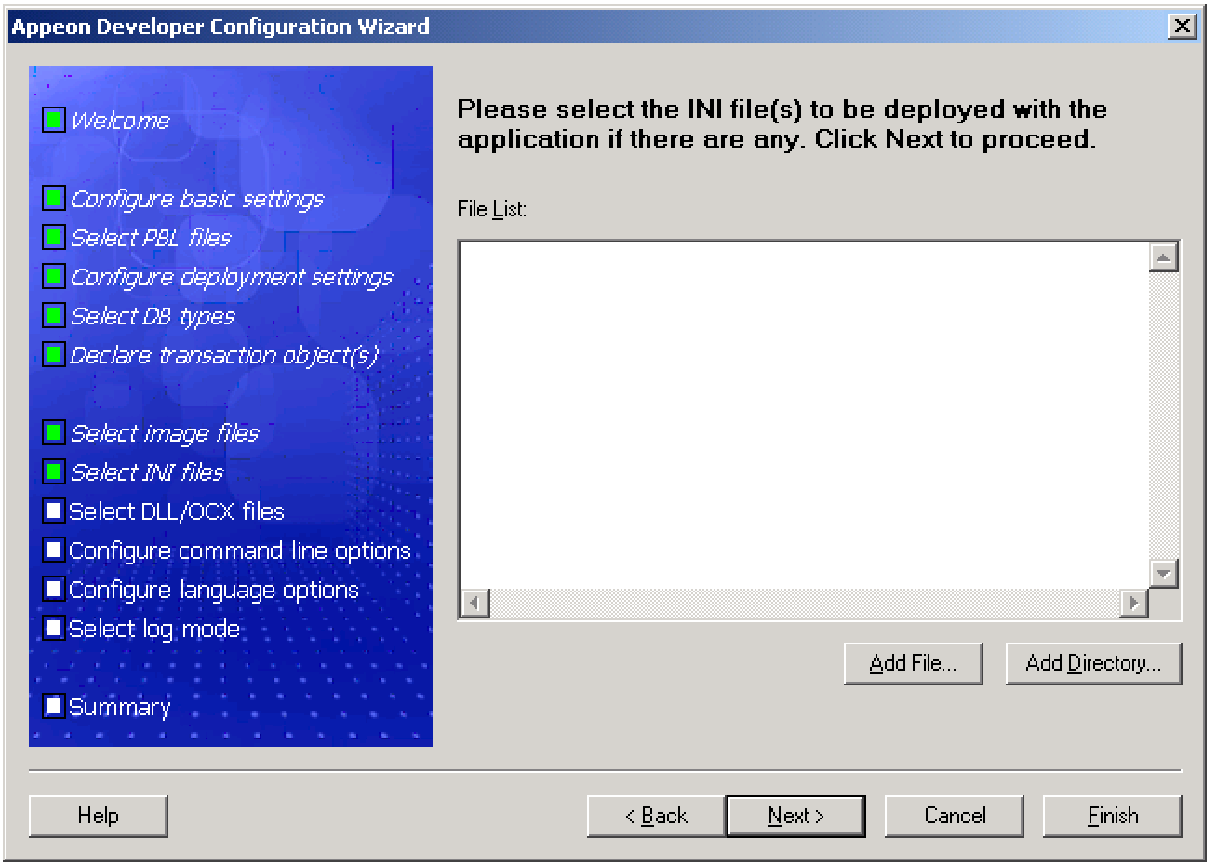
4.2.7 Selecting INI files

If a PowerBuilder application uses an initialization (INI) file which contains user preferences, specify the INI file so that Appeon can deploy the file for the application.

Click *Add* to add one or multiple INI files or click *Add Directory* to add all INI files in the selected directory (Figure 4-9).

For detailed information about the INI files, refer to the *Deploy INI Files* section.

Figure 4-9: Select INI files



Please select the INI file(s) to be deployed with the application if there are any. Click Next to proceed.

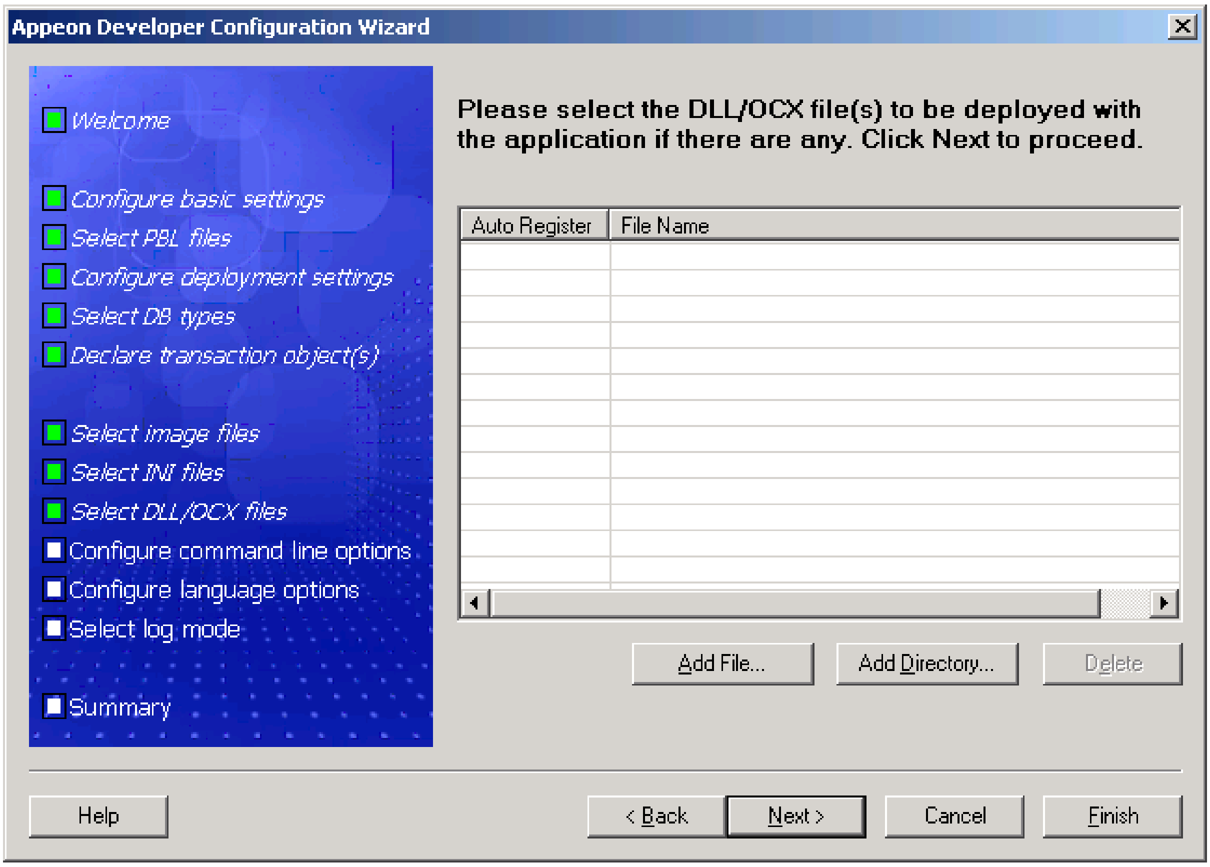
4.2.8 Selecting DLL/OCX Files

If your application calls any custom user DLL/OCX files, you can specify them here and deploy them to the Web server. Or you can use the DLL/OCX File Packaging tool provided by Appeon to package the files and install them to the client. For detailed instructions, refer to Chapter 15: *Extended Toolkit*.

Click *Add File* or *Add Directory* to add DLL/OCX files to the table. Then select the *Auto Register* column if the file needs to be automatically registered after it is downloaded to the client (Figure 4-10). If you choose not to automatically register a file, you can register it manually after it is downloaded.

For more information about the DLL/OCX files, refer to the *Deploy External Non-System DLL/OCX Files* section.

Figure 4-10: Select DLL/OCX files



Please select the DLL/OCX file(s) to be deployed with the application if there are any. Click Next to proceed.

Auto Register	File Name

4.2.9 Configuring command line options

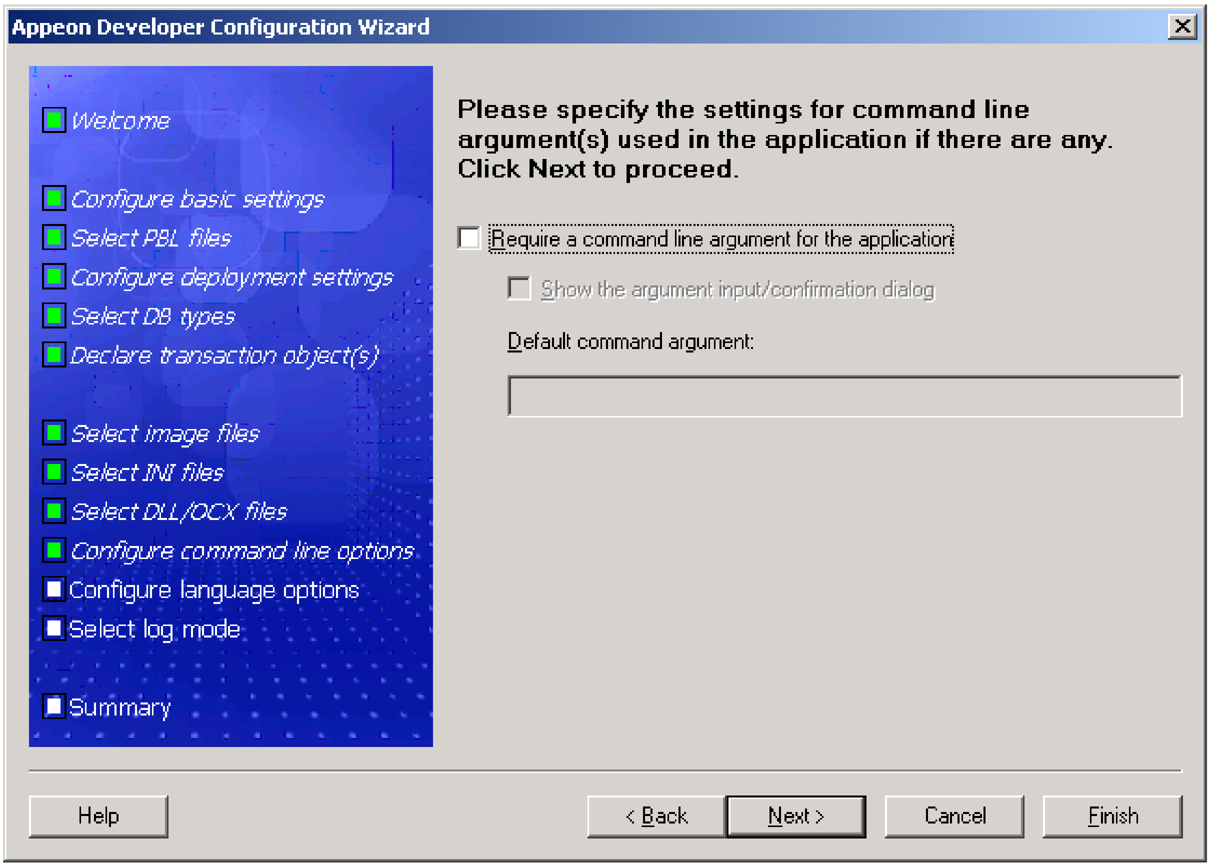
If an application needs to use command line arguments, enable the command line argument option here, as shown in Figure 4-11.

Step 1 – Click the *Require a command line argument for the application* check box to enable the command line argument option.

Step 2 – Select the *Show the argument input/confirmation dialog* option and/or input the default arguments in the *Default command argument* text box.

Note: They are effective only if you run the application from the Appeon Developer toolbar. For more information, refer to the *Command line arguments* section.

Figure 4-11: Command line settings



Please specify the settings for command line argument(s) used in the application if there are any. Click Next to proceed.

☐ Require a command line argument for the application

☐ Show the argument input/confirmation dialog

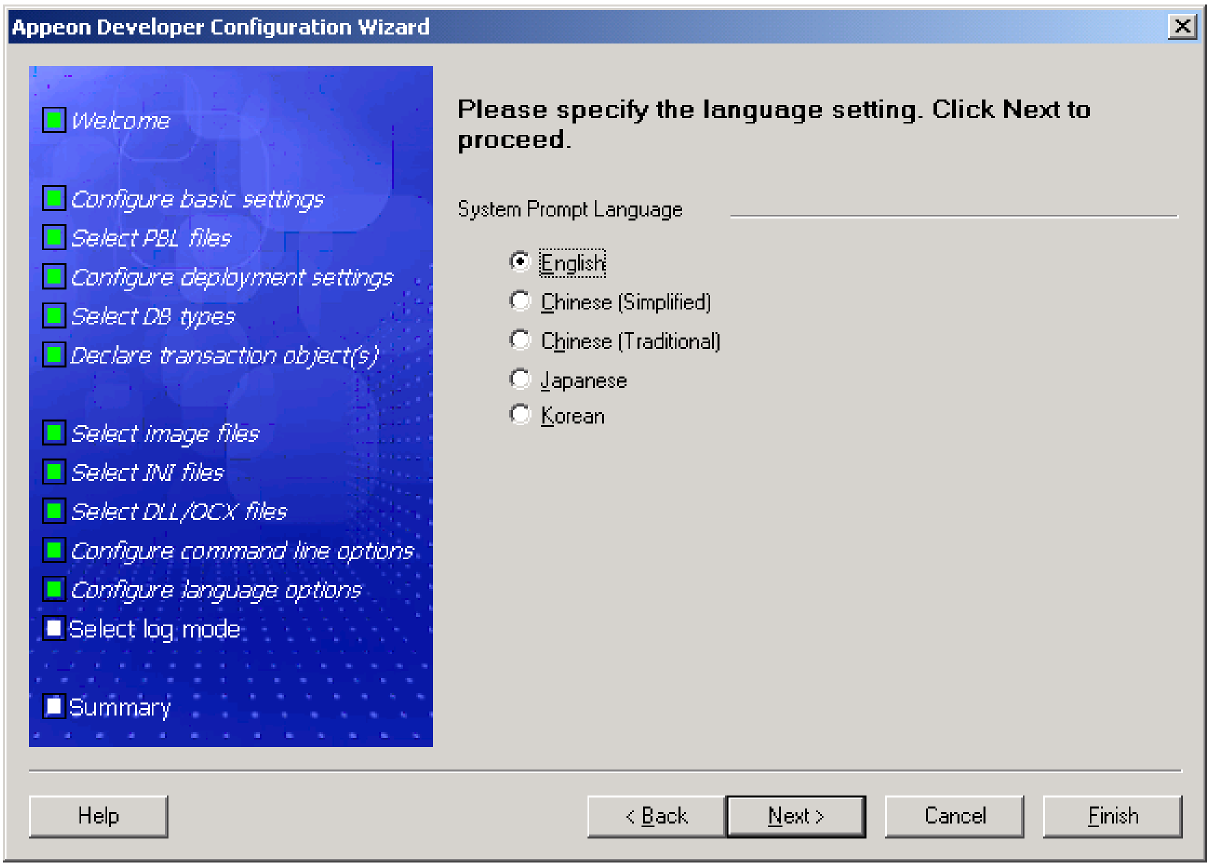
Default command argument:

4.2.10 Configuring language options

Select a language from the *System Prompt Language* dropdown listbox, as shown in Figure 4-12.

System prompts are error messages, informational messages, warnings, notices, and prompts that are displayed when you run Appeon Web applications.

Figure 4-12: System Prompt Language



Appeon Developer Configuration Wizard

Please specify the language setting. Click Next to proceed.

System Prompt Language

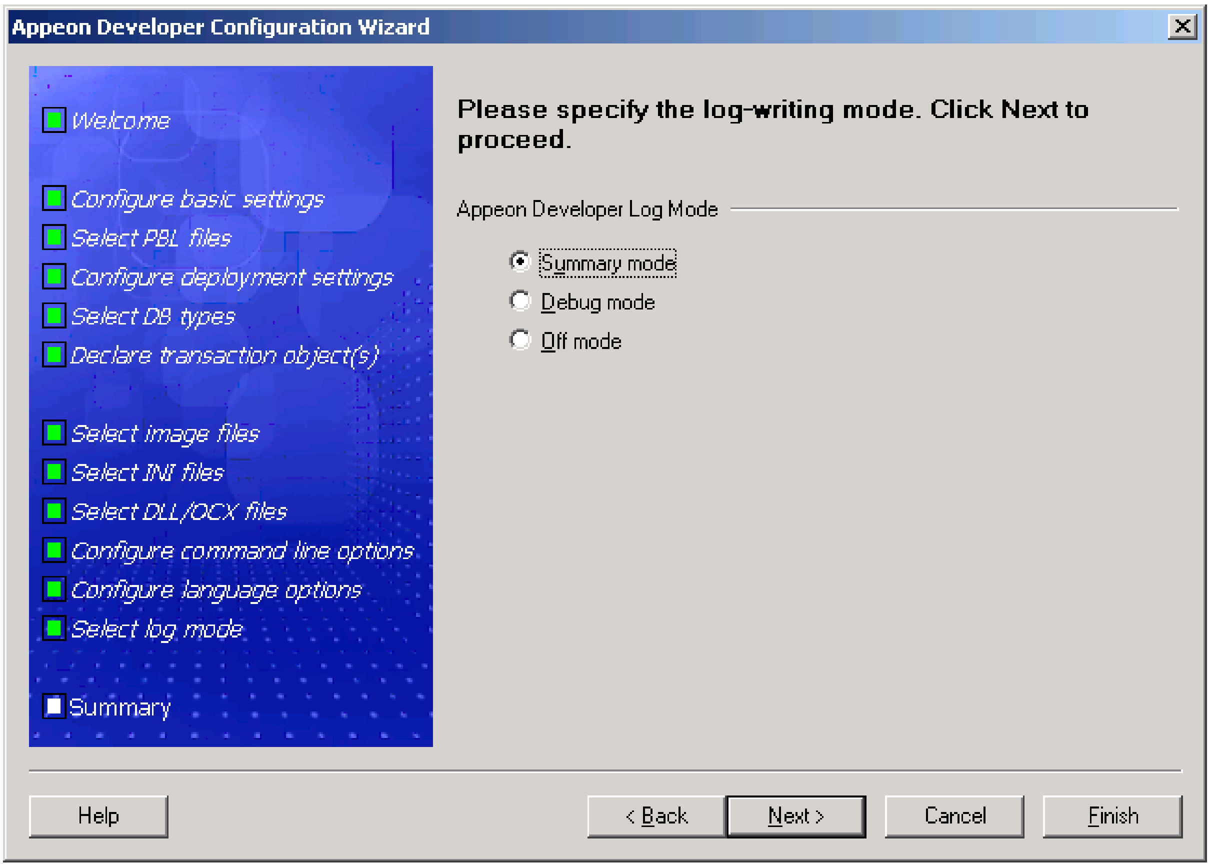
● English
○ Chinese (Simplified)
○ Chinese (Traditional)
○ Japanese
○ Korean

4.2.11 Specifying log Mode

The log-writing mode determines the content of the log files generated by Appeon Developer, including the deployment logs and feature analysis logs.

Select the desired mode from the Appeon Developer Log Mode box, as shown in Figure 4-13. For detailed information of log modes, refer to the *Appeon Developer log-writing mode* section.

Figure 4-13: Log-writing mode



Appeon Developer Configuration Wizard

Please specify the log-writing mode. Click Next to proceed.

Appeon Developer Log Mode

● Summary mode
○ Debug mode
○ Off mode

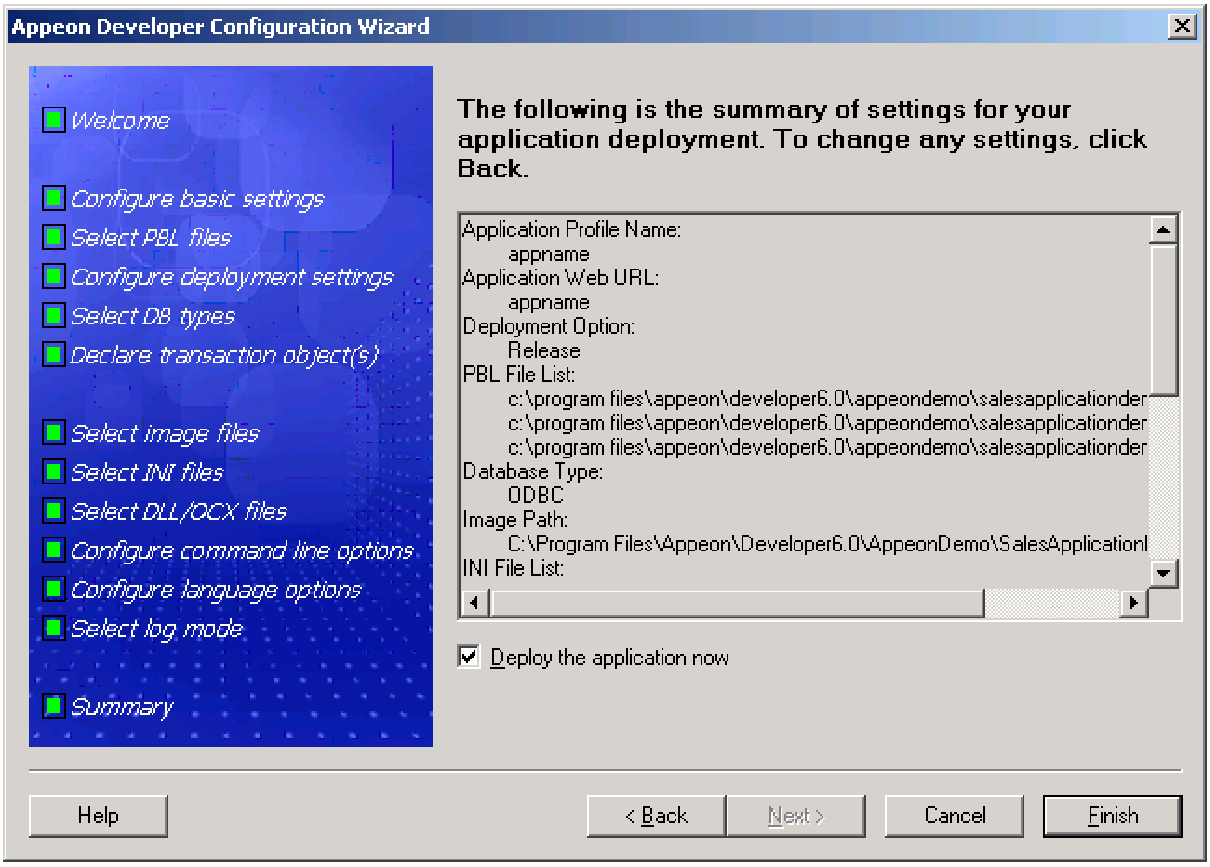
4.2.12 Summary

Review the settings. Click *Finish* to complete the configuration (Figure 4-14).

If the *Deploy the application now* option is selected, the Appeon Deployment Wizard will be launched for you to deploy the application. For detailed information, refer to Chapter 6: *Deploying PowerBuilder Applications*.

After the configuration is complete, the application will be added to the Application Profiles tab of the *Configure* tool where you can manage or change any settings of the application. For detailed instructions, refer to Section 4.3: *Using Configure Tool*.

Figure 4-14: Summary page



The following is the summary of settings for your application deployment. To change any settings, click Back.

```
Application Profile Name:
        appname
Application Web URL:
        appname
Deployment Option:
        Release
PBL File List:
        c:\program files\appeon\developer6.0\appeondemo\salesapplicationder
        c:\program files\appeon\developer6.0\appeondemo\salesapplicationder
        c:\program files\appeon\developer6.0\appeondemo\salesapplicationder
Database Type:
        ODBC
Image Path:
        C:\Program Files\Appeon\Developer6.0\AppeonDemo\SalesApplicationI
INI File List:
```

☑ Deploy the application now

4.3 Using Configure Tool

The *Configure* tool allows you to modify the settings of existing application profiles, database type profiles, Appeon Server profiles, Web server profiles, deployment profiles and connection cache profiles, and specify the advanced settings for an application profile, such as the performance settings, parsing options, Web service profiles, and runtime settings.

4.3.1 Managing application profiles

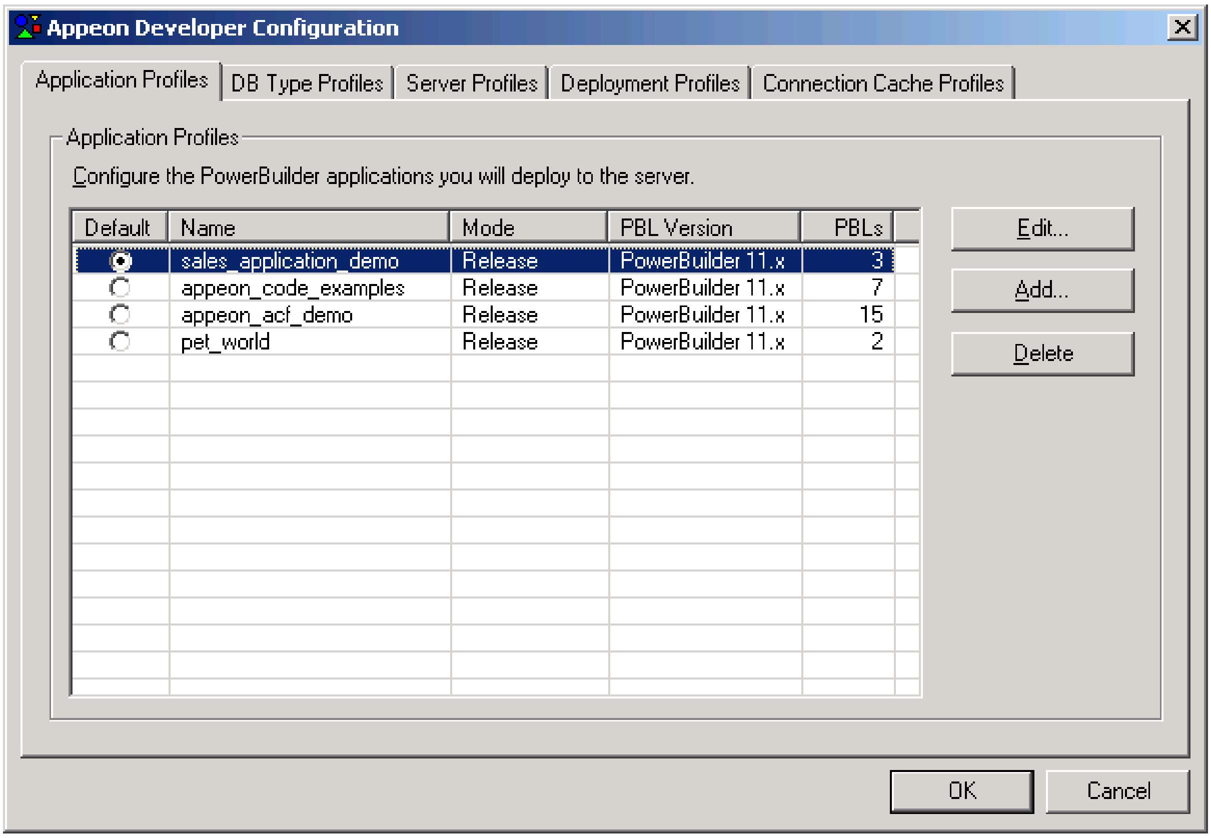
The *Configure* tool provides an *Application Profiles* tab for you to modify the settings specified in the Appeon Developer Configuration Wizard, and use the following advanced functions:

- Specify the default application profile
- Select the Web file generation mode for an application
- Delete an application profile
- Boost the application runtime performance
- Specify parsing options
- Specify runtime settings
- Configure Web Service profiles

4.3.1.a Application Profiles tab page

When you click the *Configure* button () on the Appeon Developer toolbar, the Appeon Developer Configuration window pops up with the *Application Profiles* tab selected by default, as shown in Figure 4-15.

Figure 4-15: Appeon Developer Configuration Window



Configure the PowerBuilder applications you will deploy to the server.

Default	Name	Mode	PBL Version	PBLs
●	sales_application_demo	Release	PowerBuilder 11.x	3
○	appeon_code_examples	Release	PowerBuilder 11.x	7
○	appeon_acf_demo	Release	PowerBuilder 11.x	15
○	pet_world	Release	PowerBuilder 11.x	2

Specifying the default application profile

To specify which application profile will be used as the default application profile, select the *Default* radio button.

The PowerBuilder application defined in the default application profile is selected for unsupported features analysis, PowerBuilder-to-Web conversion, application packaging, and application undeployment.

Selecting a Web file generation mode

To specify a mode for Web file generation, click the *Mode* list box. If you change the mode, you must perform a full deployment to make the changes effective.

Table 4-1: File generation mode

Mode	Description	Use In The…
Debug PS/JS	Generates unencrypted JavaScript files for debug use. The PowerBuilder source code is provided as comments in the JavaScript files for easy reference.	Web application debugging and tuning stage.
Debug JS	Generates unencrypted JavaScript files for debug use. No PowerBuilder source code is provided in the	Web application debugging and tuning

	JavaScript files.	stage.
Release	Generates encrypted JavaScript files.	Production stage.

Deleting an application profile

Click the *Delete* button to delete a selected application profile.

Note: You cannot delete the default application profile.

After the application profile is deleted, delete the temporary folder for the application profile on the Developer machine. The folder has the same name as the application profile and is created in the \Project folder under the root directory where Appeon Developer is installed (e.g. C:\Program Files\Appeon\Developer6.1\).

4.3.1.b Application profile settings

The *Application Profiles Configuration* window provides six tab pages for you to edit the required application information:

Table 4-2: Application Profiles settings

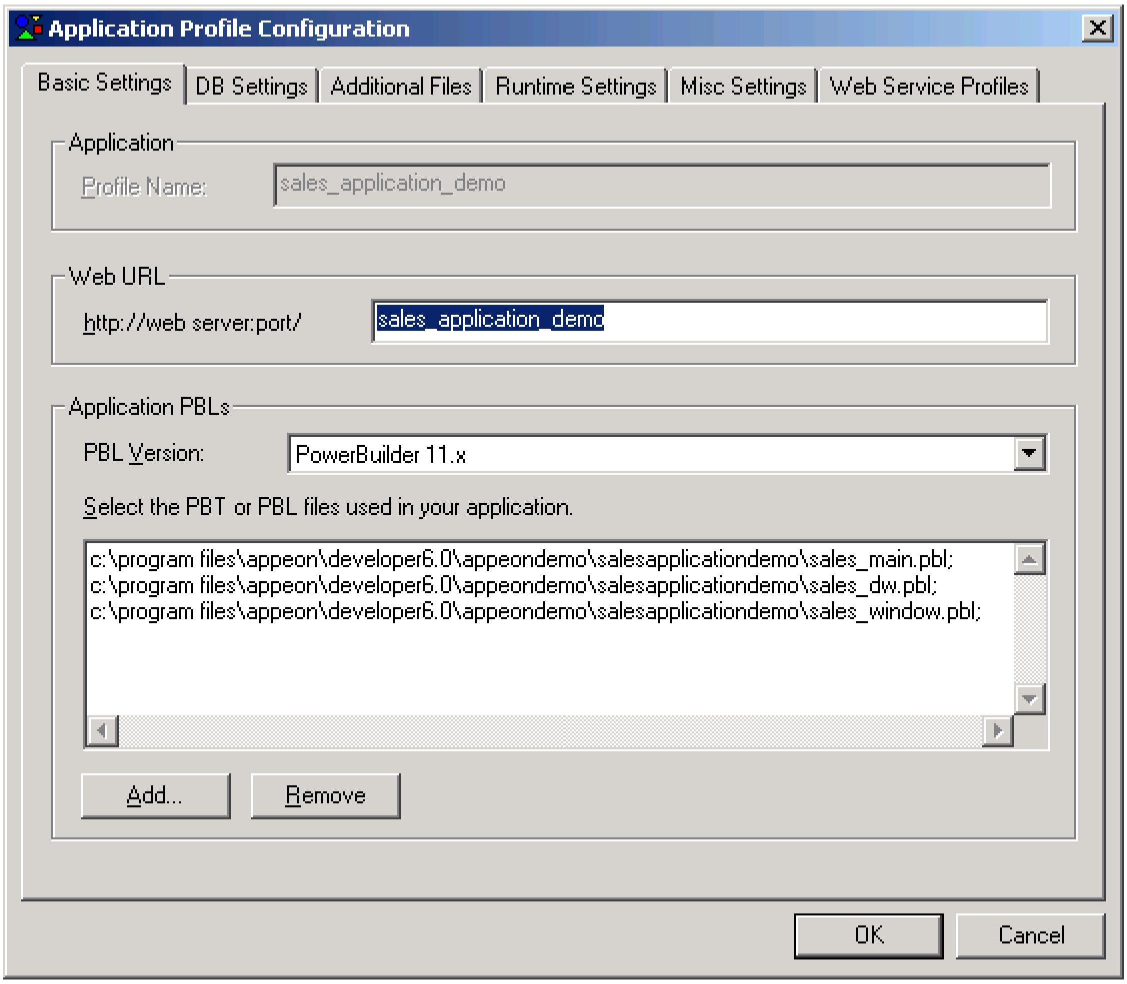
Tab	Settings	To make changes effective, you must …
Basic Settings tab	The application profile name, Web URL, PBL version and PBL location of an application. See the *Basic Settings* section for more details.	Perform a full or incremental deployment on the application.
DB Settings tab	The database types, transaction objects and the corresponding connection caches. See the *DB Settings* section for more details.	Perform a full or incremental deployment on the application.
Additional Files tab	INI files, image files and DLL/OCX files. See the *Additional Files* section for more details.	Perform a full or incremental deployment on the application.
Runtime Settings tab	The application runtime mode, application file storage location, and company information. See the *Runtime Settings* section for more details.	Perform a full or incremental deployment on the application.
Misc Settings tab	Command line arguments, application language, runtime performance, parsing option, and log-writing mode. See the *Misc Settings* section for more details.	Perform a full deployment on the application.
Web Service Profiles tab	The WSDL file, service name and port for the Web service(s). See the *Web Service Profiles* section for more details.	Perform a full or incremental deployment on the application.

Basic Settings

After you click the *Add* or *Edit* button on the *Application Profiles* tab, the *Application Profile Configuration* window is displayed with the *Basic Settings* tab selected by default, as shown in Figure 4-16.

The Basic Settings tab allows you to modify the Web URL, application PBL version and PBL location.

Figure 4-16: Basic Settings



Application

Profile Name: sales_application_demo

Web URL

http://web server:port/ sales_application_demo

Application PBLs

PBL Version: PowerBuilder 11.x

Select the PBT or PBL files used in your application.

c:\program files\appeon\developer6.0\appeondemo\salesapplicationdemo\sales_main.pbl;
c:\program files\appeon\developer6.0\appeondemo\salesapplicationdemo\sales_dw.pbl;
c:\program files\appeon\developer6.0\appeondemo\salesapplicationdemo\sales_window.pbl;

Application Profile Name – The application profile name is used to identify the PowerBuilder application during the entire PowerBuilder-to-Web process. It is also used in the appeondb database on Appeon Server to identify the DataWindow syntax, profile, and registry information for an Appeon Web application.

The Profile Name cannot contain double-byte characters (such as Chinese, Korean, or Japanese characters) or special characters (such as, \, /, :, *, ?, ", <, >, or |).

Web URL – A URL of the Appeon Web application can contain a combination of letters, underscores (“_”), and numbers. The specified URL is also used as the name of the folder created under the Web Root of the Web Server for storing the Web application files when the application is deployed.

Avoid setting application Web URLs as existing folder names in the Web server Web root because: (1) If a Web application uses an existing folder under the Web root as the Web URL, a large number of Web files belonging to the application will be deployed to that folder, making it difficult to use the folder for its original purpose. (2) If a Web application uses a folder that is reserved for the Web server or Appeon Server, some important files belonging to the Web server or Appeon Server may be replaced or even removed when the application is undeployed. The following are folder names reserved for the Web server or Appeon

Server that should not be used as application Web URLs: appeon, classes, docs, images, ir, wst, and WEB-INF.

The Web URL for an application profile can be changed, and the Web files stored at the Web Server can have different versions for the Client to access, but Appeon only remembers the Web URL in the last deployment, and performs the application packaging and application undeployment based on the Web files generated in the corresponding Web folder.

Application PBL files – Select the PBL version and add application PBL files using any of the following methods:

- Click the *Add* button and choose to add an application Target file (*.pbt).

  By adding PBLs using an application Target, all PBLs in the Target are automatically added to the application PBLs list. This is the recommended way to add PBLs, as it reduces the likelihood of forgetting to add a required PBL.

- Click the *Add* button and add PBL files without using an application Target.

  You can add one or multiple PBL files at the same time. To select multiple files, hold down the Ctrl key and choose which files to add.

- Copy and paste (Ctrl+C and Ctrl+V) the path list of PBLs into the PBL list box.

DB Settings

The *DB Settings* tab allows you to specify the database types and the transaction objects that the application uses, as shown in Figure 4-17. When configuring the database types, it is not necessary to specify the actual databases that the application uses, it can be any database of the same type. However, when specifying the transaction objects, you must specify the actual transaction objects that the application uses.

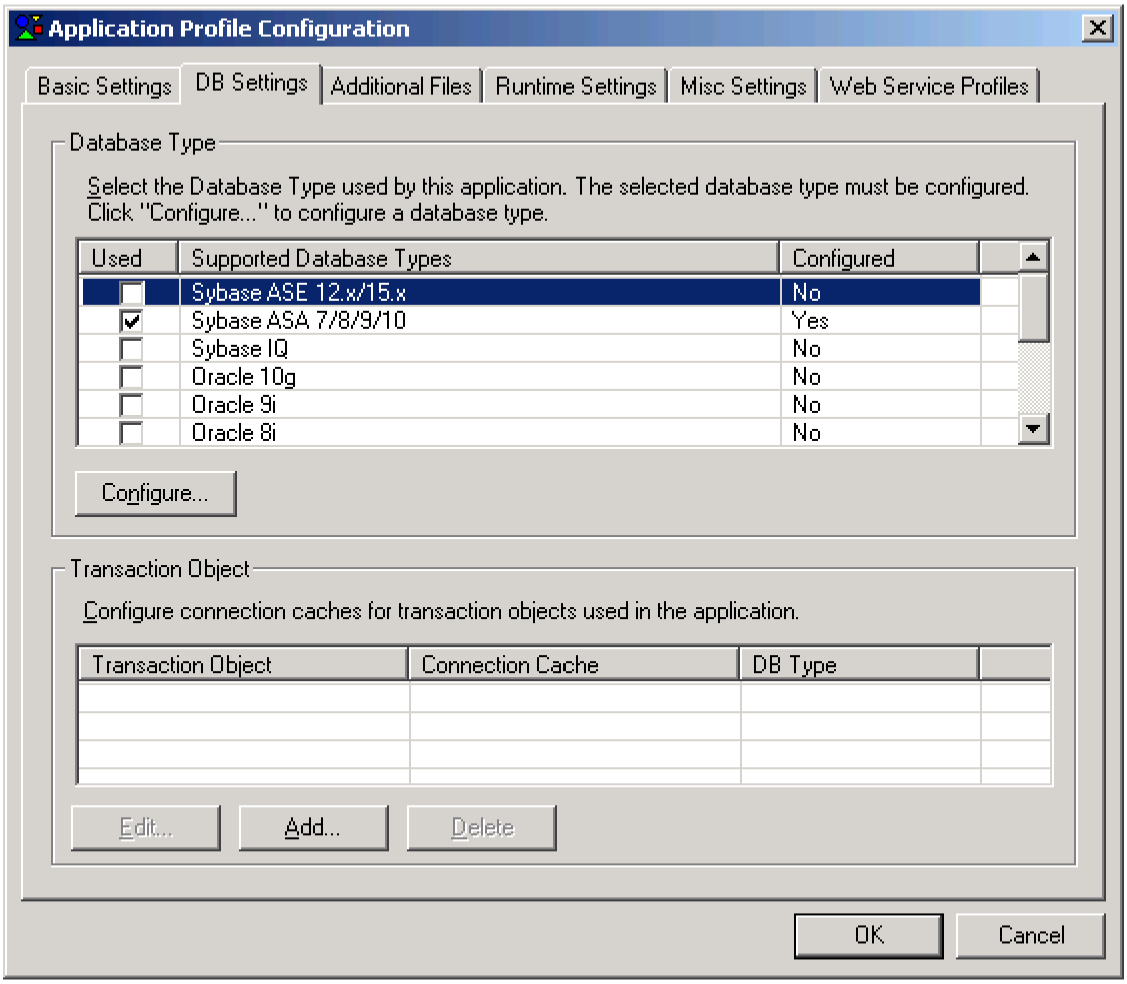
What the database type is used for

PowerBuilder SQL statements must be converted into the correct database syntax before they can be used to retrieve data from the database that the Web application connects to. The database type settings enable Appeon Developer to apply the correct database driver type for generating correct database syntax for PowerBuilder SQL statements.

Set the database types for the application

Select the database type(s) used by the application, as shown in Figure 4-17.

Figure 4-17: Database types



Database Type

Select the Database Type used by this application. The selected database type must be configured. Click "Configure..." to configure a database type.

Used	Supported Database Types	Configured
□	Sybase ASE 12.x/15.x	No
☑	Sybase ASA 7/8/9/10	Yes
□	Sybase IQ	No
□	Oracle 10g	No
□	Oracle 9i	No
□	Oracle 8i	No

Transaction Object

Configure connection caches for transaction objects used in the application.

Transaction Object	Connection Cache	DB Type

Table 4-3: Database type settings

Column	Description
Used	You can click the checkboxes to select the database types used by the application. More than one database type can be selected, but only the configured database types are selectable.
Supported Database Type	This column lists all the database types that are supported by Appeon.
Configured	This column is identical to the Configured column in the DB Type Profiles tab. It indicates whether a profile has been configured for the database type. If this column of the desired database type is "No", you must click the *Configure* button to configure a profile for it before you can use it. Configuring a database type profile in this tab is the same as configuring it in the *DB Type Profiles* tab. Once a database type profile is configured successfully, the Configured column in both this tab and the *DB Type Profiles* tab changes from "No" to "Yes".

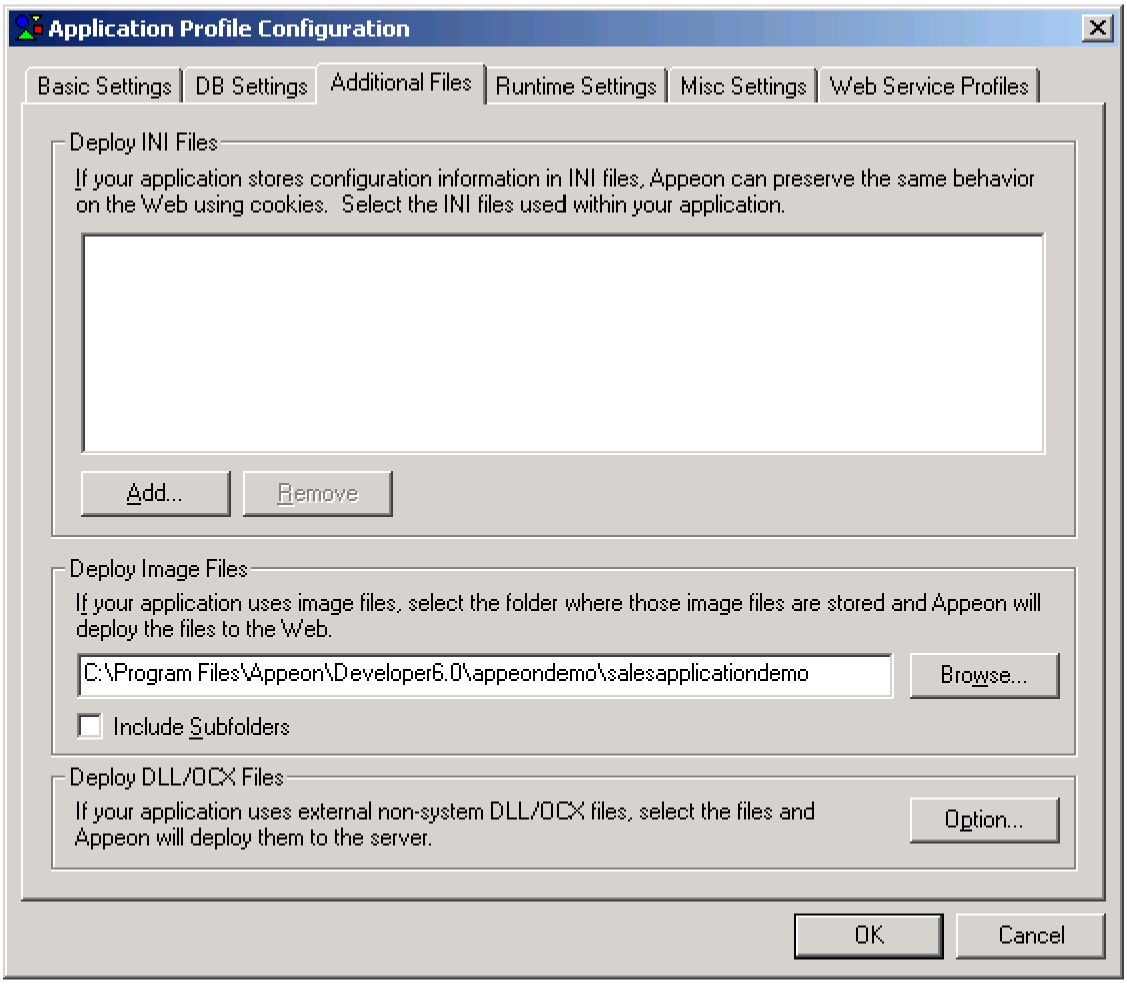
Specify the transaction object and the connection cache

Click *Add* or *Edit* to configure a transaction object and specify the corresponding connection cache. Refer to Section 4.2.5: *Declaring transaction object(s)* for detailed description.

Additional Files

"Additional files" refer to the files that are outside the application source code and are necessary for running the PowerBuilder application, including INI files, image files, and DLL/OCX files.

Figure 4-18: Additional Files tab



Application Profile Configuration

Basic Settings | DB Settings | Additional Files | Runtime Settings | Misc Settings | Web Service Profiles

Deploy INI Files

If your application stores configuration information in INI files, Appeon can preserve the same behavior on the Web using cookies. Select the INI files used within your application.

Deploy Image Files

If your application uses image files, select the folder where those image files are stored and Appeon will deploy the files to the Web.

C:\Program Files\Appeon\Developer6.0\appeondemo\salesapplicationdemo

☐ Include Subfolders

Deploy DLL/OCX Files

If your application uses external non-system DLL/OCX files, select the files and Appeon will deploy them to the server.

Deploy INI Files

If a PowerBuilder application uses an initialization (INI) file which contains user preferences, specify the INI file in this tab so that Appeon can deploy the file for the application.

Click *Add* to add the INI file(s) into the Deploy INI Files list box.

How INI files are supported in Appeon

During deployment, the specified INI file(s) are converted to XML file(s) and deployed to Appeon Server. The XML file(s) act as mock INI file(s). When a Client runs the deployed application for the first time, a copy of the XML file is created in Appeon Server and keeps the Client's profile information.

For a Client to use the correct profile information in its subsequent visits to the application, the Client browser must be cookie-enabled. Otherwise, each time the Client visits the application, a new copy of the XML file is created in Appeon Server, and the new copy only contains the information of the original INI file.

You can delete XML files that are unused for a period using Appeon Enterprise Manager (AEM). For more information on how to maintain XML files in Appeon Server, refer to the *Auto Cleanup* section in the *Appeon Server Configuration Guide*.

Deploy Images

Click *Browse* to specify which folder contains the image files that will be used in the Web application. If you do not specify the path of the image files, the path to the application PBL files will be displayed as the default path of the image files.

The image files can be in any format supported by PowerBuilder, apart from:

- unsupported run-length encoded (RLE), and
- unsupported Windows metafile (WMF)

All the image files in the specified folder will be deployed to the *%Web Deployment Path%\images* folder in the Web Server. *%Web Deployment Path%* stands for the folder storing the Web application files. The image files may fail to be displayed on the Web if the file names are stored in the database or dynamically generated. To resolve this, you can copy the image files directly to the *%Web Deployment Path%\images* folder, but make sure that the file names are all lowercase; otherwise, the Web application may fail to load them.

Deploy External Non-System DLL/OCX Files

If your application calls any custom user DLL/OCX files, which are also called external non-system DLL/OCX files in this tab, you can specify them here and deploy them to the Web server.

Alternatively, you can use the DLL/OCX File Packaging tool provided by Appeon to package the files and install them to the *%SystemRoot%\system32\AppeonPlugin* folder at each Client. %SystemRoot% indicates the folder where Windows is installed (for example, C:\WINNT). For detailed instructions, refer to Chapter 15: *Extended Toolkit*.

Click *Options* in the *Deploy DLL/OCX Files* group box to open the Deploy DLL/OCX Files dialog box (as shown in Figure 4-19) and select the required DLL/OCX files to be deployed.

Figure 4-19: DLL/OCX Automatically Deploy and Download

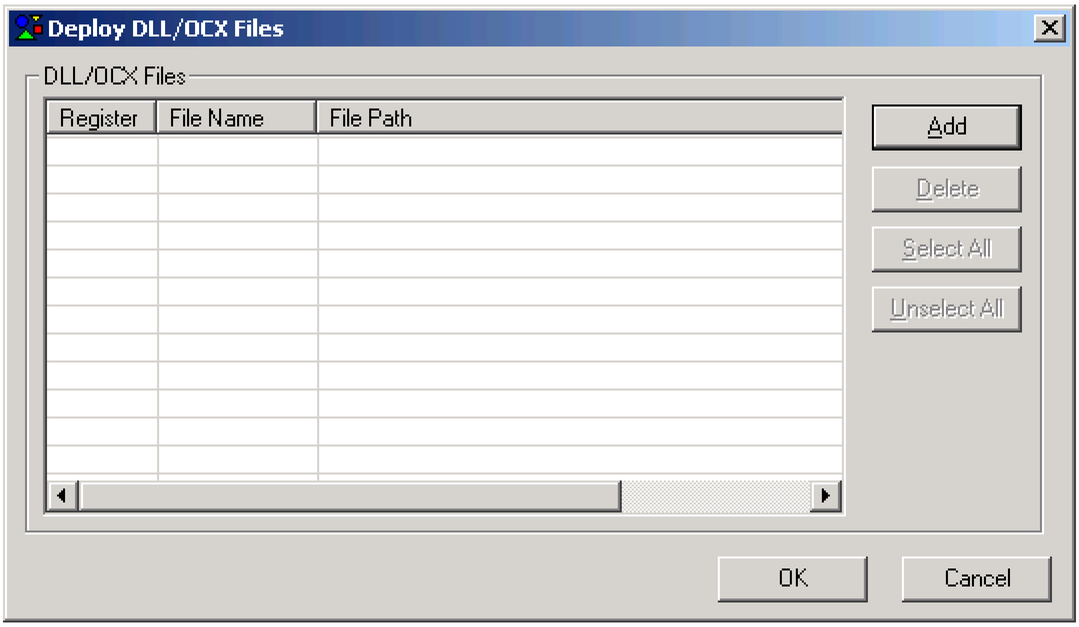


Table 4-4 introduces the elements on the Deploy DLL/OCX Files dialog box.

Table 4-4: Deploy DLL/OCX Files dialog box

column	Description
Register	Allows you to select whether the file will be automatically registered after it is downloaded to the Client at runtime. Click *Select All* to select all files and *Unselect All* to de-select all of them. If you choose not to automatically register a file, you can register it manually after it is downloaded.
File Name	Lists the added files. Click the header of this column to display the files in alphabetical order.
File Path	Displays the file location.

Table 4-5 describes how to use the Deploy DLL/OCX Files dialog box.

Table 4-5: Deploy DLL/OCX Files dialog box

To Do This	Do This
To add files	Click the *Add* button. A standard File Selection dialog box is displayed. Select the files from your local machine. The files can be stored in any location.
To remove files	Select a file or multiple files by pressing Ctrl or Shift keys, and click the *Delete* button to remove the selected files from the list.
To save the settings	Click *OK*. Appeon will copy the files to a temporary folder and then deploy them to the Web server during deployment.

How the deployed DLL/OCX files work on Web

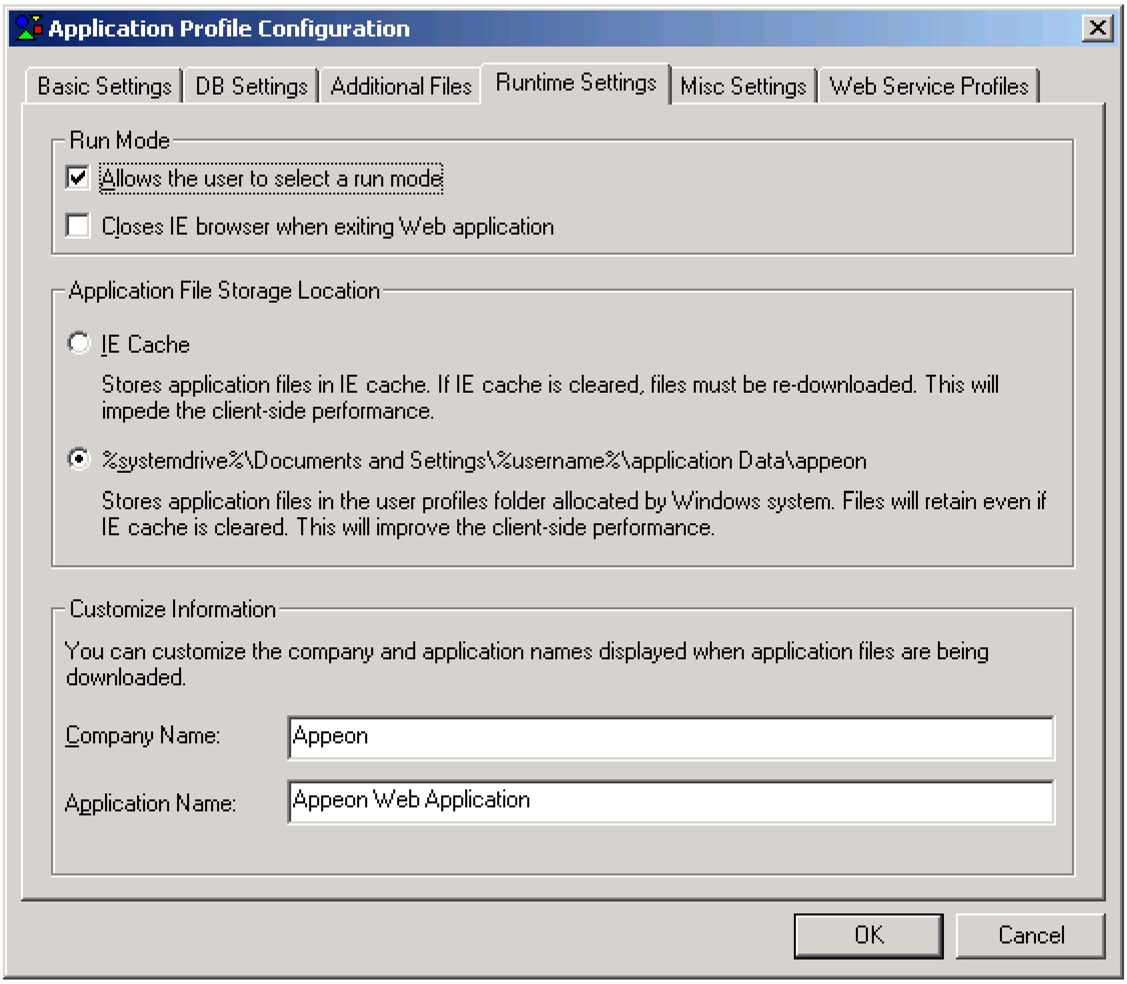
To successfully call the deployed DLL/OCX files on the Web application, the DLL/OCX files must be downloaded and installed to the Client during the initial run of the deployed Web application.

Appeon Enterprise Manager (AEM) provides options for manually or automatically downloading and installing the files. The files should be saved to the *%SystemRoot%\system32\AppeonPlugin* folder at the Client. *%SystemRoot%* indicates the folder where Windows is installed (for example, C:\WINNT). For detailed instructions, refer to the *Appeon Server Configuration Guide*.

Runtime Settings

The Runtime Settings tab (Figure 4-20) determines how an Appeon application starts on the Web.

Figure 4-20: Runtime Settings dialog box



Run Mode

☑ Allows the user to select a run mode

□ Closes IE browser when exiting Web application

Application File Storage Location

○ IE Cache

Stores application files in IE cache. If IE cache is cleared, files must be re-downloaded. This will impede the client-side performance.

● %systemdrive%\Documents and Settings\%username%\application Data\appeon

Stores application files in the user profiles folder allocated by Windows system. Files will retain even if IE cache is cleared. This will improve the client-side performance.

Customize Information

You can customize the company and application names displayed when application files are being downloaded.

Company Name: Appeon

Application Name: Appeon Web Application

Allows the user to select a run mode – This option provides feasible download modes to the end user if they use slow bandwidth. Depending on the bandwidth, the end user can select from the following run modes when the application starts:

- Run Now – The application will run immediately and files will be downloaded when needed. This is the default mode if the “Allows users to select the run mode” option is not selected.
- Run After Download – The application will run after all files are downloaded to the client. This is recommended for clients using low bandwidth.

Closes IE browser when existing Web application – When you close the Web application, the Internet Explorer window does not close, leaving a blank page in it. With this option enabled, the Internet Explorer window can be automatically closed when the application is closed.

Application Files Storage Location – You can select to store the downloaded application files in the following two locations:

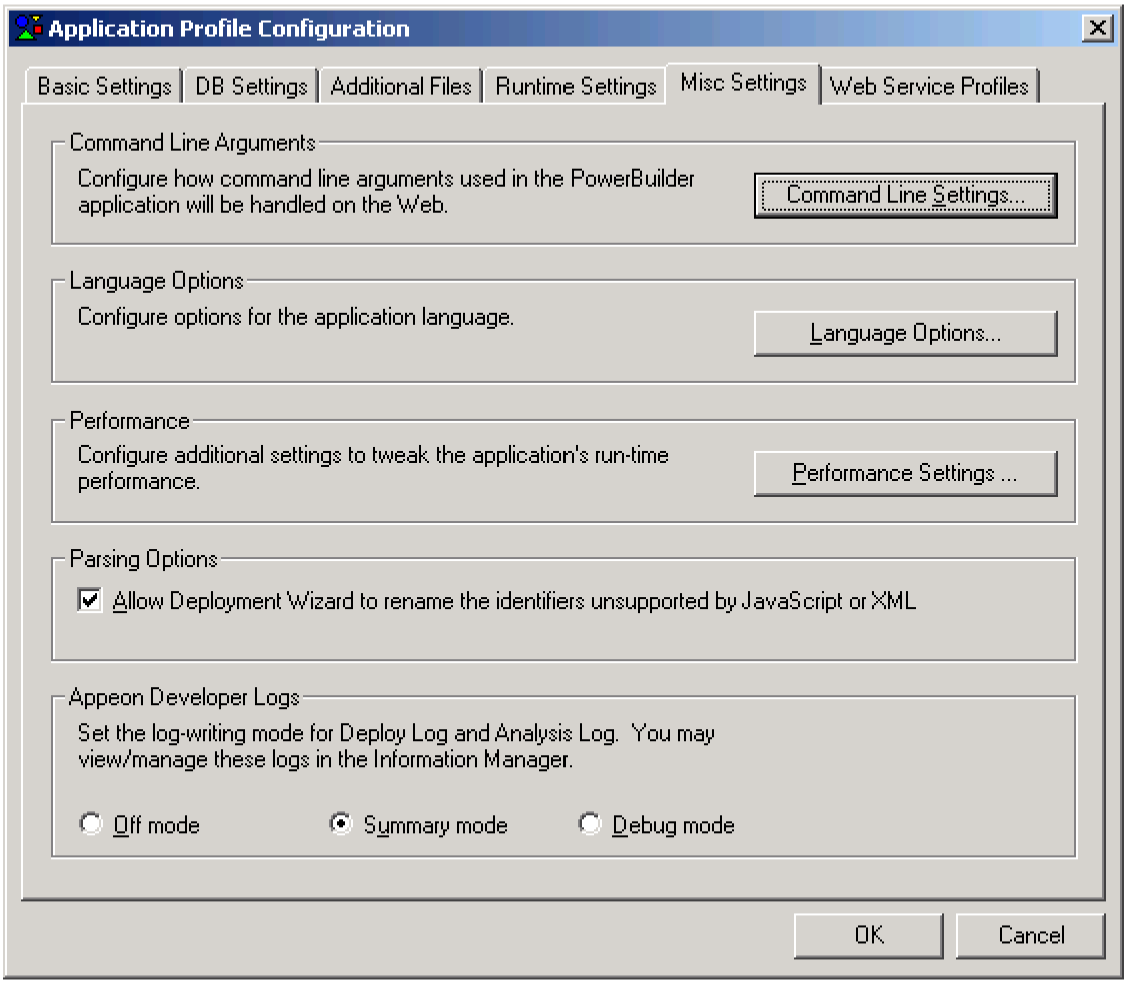
- IE Cache – Files will be downloaded to the IE cache. When IE cache is cleared, files will be downloaded again the next time the application runs.
- Windows user profiles folder – Files will be saved permanently in the user profiles folder: %systemdrive%\Documents and Settings\%username%\Application Data\appeon, unless you remove them manually. This is the recommended storage location.

Customize Information – When the application starts, the company and application names will be displayed on the downloading page. You can specify these names according to your actual needs.

Misc Settings

The *Misc Settings* tab page enables you to modify settings for command line arguments, application language, and log-writing mode, and allows you to configure the runtime performance, and parsing options.

Figure 4-21: Misc settings



Application Profile Configuration

Command Line Arguments

Configure how command line arguments used in the PowerBuilder application will be handled on the Web.

Language Options

Configure options for the application language.

Performance

Configure additional settings to tweak the application's run-time performance.

Parsing Options

☑ Allow Deployment Wizard to rename the identifiers unsupported by JavaScript or XML

Appeon Developer Logs

Set the log-writing mode for Deploy Log and Analysis Log. You may view/manage these logs in the Information Manager.

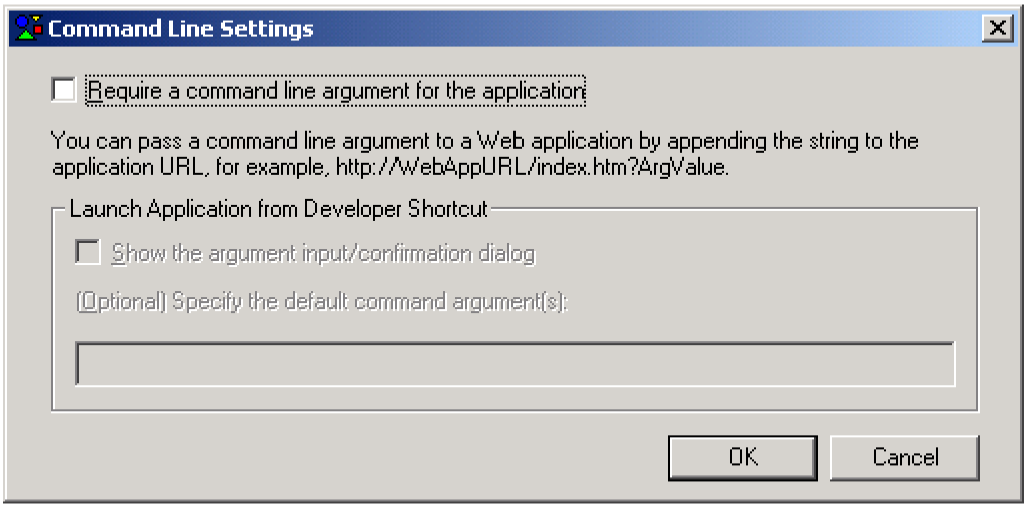
○ Off mode ● Summary mode ○ Debug mode

Command line arguments

If an application needs to use command line arguments, enable the command line argument option, as shown in Figure 4-22.

Click *Command Line Settings* on the *Misc Settings* tab. The Command Line Settings dialog box is displayed, as shown in Figure 4-22.

Figure 4-22: Command Line Settings dialog box


Command Line Settings

□ Require a command line argument for the application

You can pass a command line argument to a Web application by appending the string to the application URL, for example, http://WebAppURL/index.htm?ArgValue.

Launch Application from Developer Shortcut

□ Show the argument input/confirmation dialog

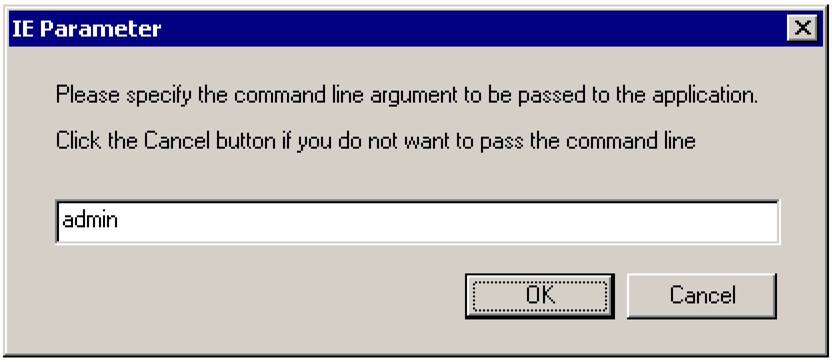
(Optional) Specify the default command argument(s):

- Require a command line argument for the application option – Enables the command line argument option.

  With this check box selected, the following two options will be enabled. Note that they are effective only if you run the application from the Appeon Developer toolbar.

  - *Show argument input/confirmation dialog* option – A dialog box is displayed after you launch an application from the Appeon Developer toolbar but before the application starts. This dialog box enables you to specify the arguments for the application.
  - *Specify the default command arguments* field – The specified default arguments are directly attached to the application URL when the application is run from the Appeon Developer toolbar.
  - *If both options are selected* – A dialog box pops up with the default arguments filled in, as shown in Figure 4-23. You can either change the default arguments or leave them alone.

Figure 4-23: Command line argument dialog box


IE Parameter

Please specify the command line argument to be passed to the application.

Click the Cancel button if you do not want to pass the command line

admin

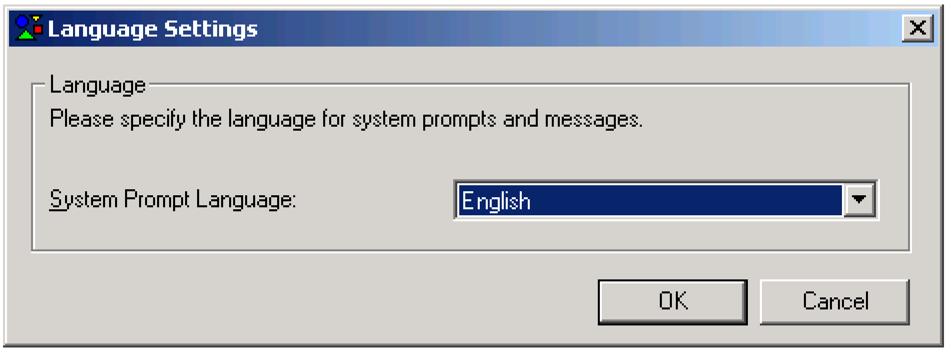
To run a Web application with arguments from Clients, type the application URL with the arguments in the Internet Explorer address bar.

The format of a Web application URL with arguments is: "http://*WebAppURL*/index.htm?*ArgValue*". *ArgValue* indicates the arguments that will be passed to the application. The arguments must be separated by commas if there are more than one argument.

Application language

Step 1 – Click *Language Options* on the *Misc Settings* tab. The Language Settings dialog box is displayed, as shown in Figure 4-24.

Figure 4-24: Application language settings



Language Settings

Language

Please specify the language for system prompts and messages.

System Prompt Language: English

Step 2 – Select a language from the *System Prompt Language* dropdown listbox.

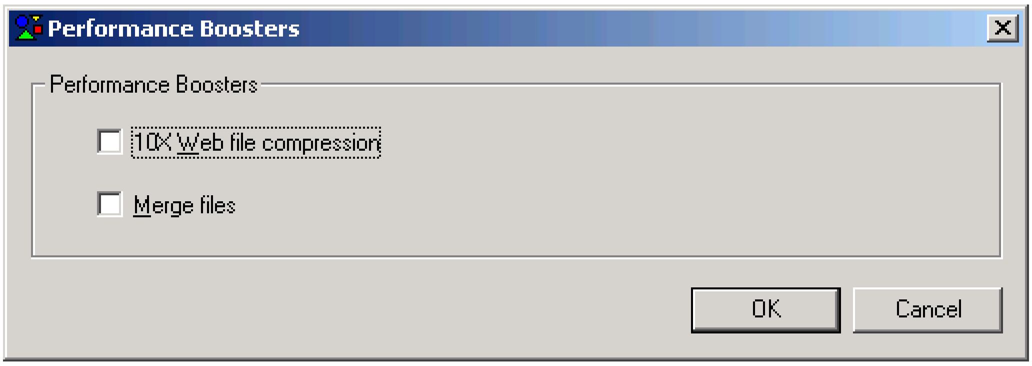
System prompts and messages are error messages, informational messages, warnings, notices, and prompts that are displayed when you run Appeon Web applications.

Performance settings

Appeon recommends that you take advantage of the performance settings to boost the application's runtime performance. To produce the best performance, perform the following steps:

Step 1 – Click the *Performance Settings* on the *Misc Settings* tab. The Performance Boosters dialog box is displayed, as shown in Figure 4-25.

Figure 4-25: Performance settings



Performance Boosters

Performance Boosters

☐ 10X Web file compression

☐ Merge files

Step 2 – Select the performance features in the Performance Boosters box.

Table 4-6 describes how to use the performance features.

Table 4-6: Performance features

Performance Feature	Description	When To Use…
10X Web File Compression	Compresses files when they are transferred over the network.	Always
Merge files	Merges the small files during the application deployment. The small files will be downloaded to the client in one file package at one call, instead of being downloaded one by one at separate calls.	In the production stage.

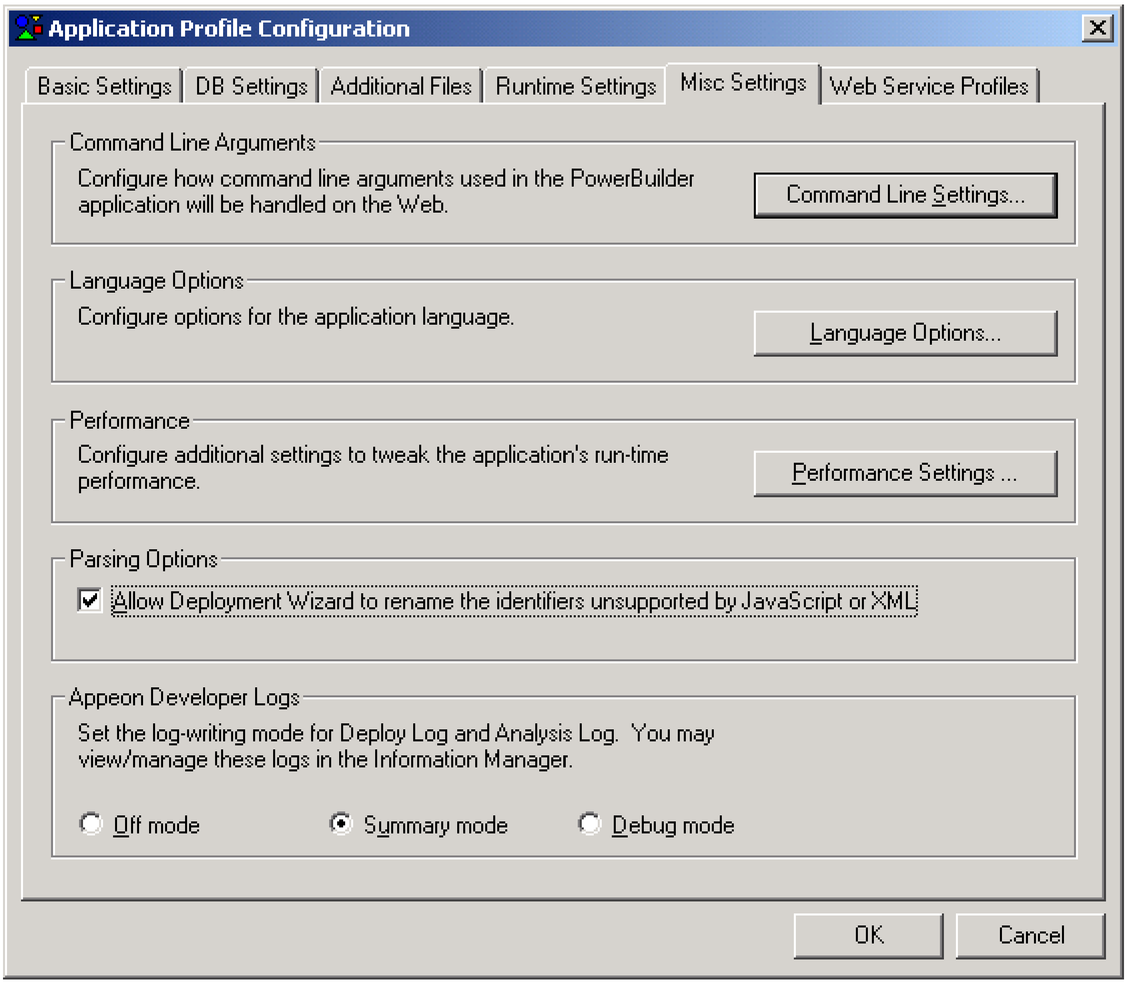
Parsing options

Unsupported identifiers refer to identifiers that are supported in PowerBuilder but unsupported in Appeon, due to the different naming convention between PowerScript and JavaScript/XML. For detailed explanation of such unsupported identifiers, refer to Appeon Features Help.

With this automatic-renaming option enabled, when the Appeon Deployment Wizard deploys an application it automatically detects unsupported identifiers in the application and renames these identifiers to supported ones in the Web files it generates for the application, while keeping the application source code unchanged. After the deployment, you can get the list of all identifiers that are renamed in the deployment log in the Information Manager (Chapter 9: *Using Information Manager*).

The automatic-renaming feature will not work if the option is disabled. In this case, you must rename the unsupported identifiers by yourself, following the guidance of the UFA report and Appeon Features Help.

Figure 4-26: Parsing options



Command Line Arguments: Configure how command line arguments used in the PowerBuilder application will be handled on the Web.

Language Options: Configure options for the application language.

Performance: Configure additional settings to tweak the application's run-time performance.

Parsing Options: ☑ Allow Deployment Wizard to rename the identifiers unsupported by JavaScript or XML

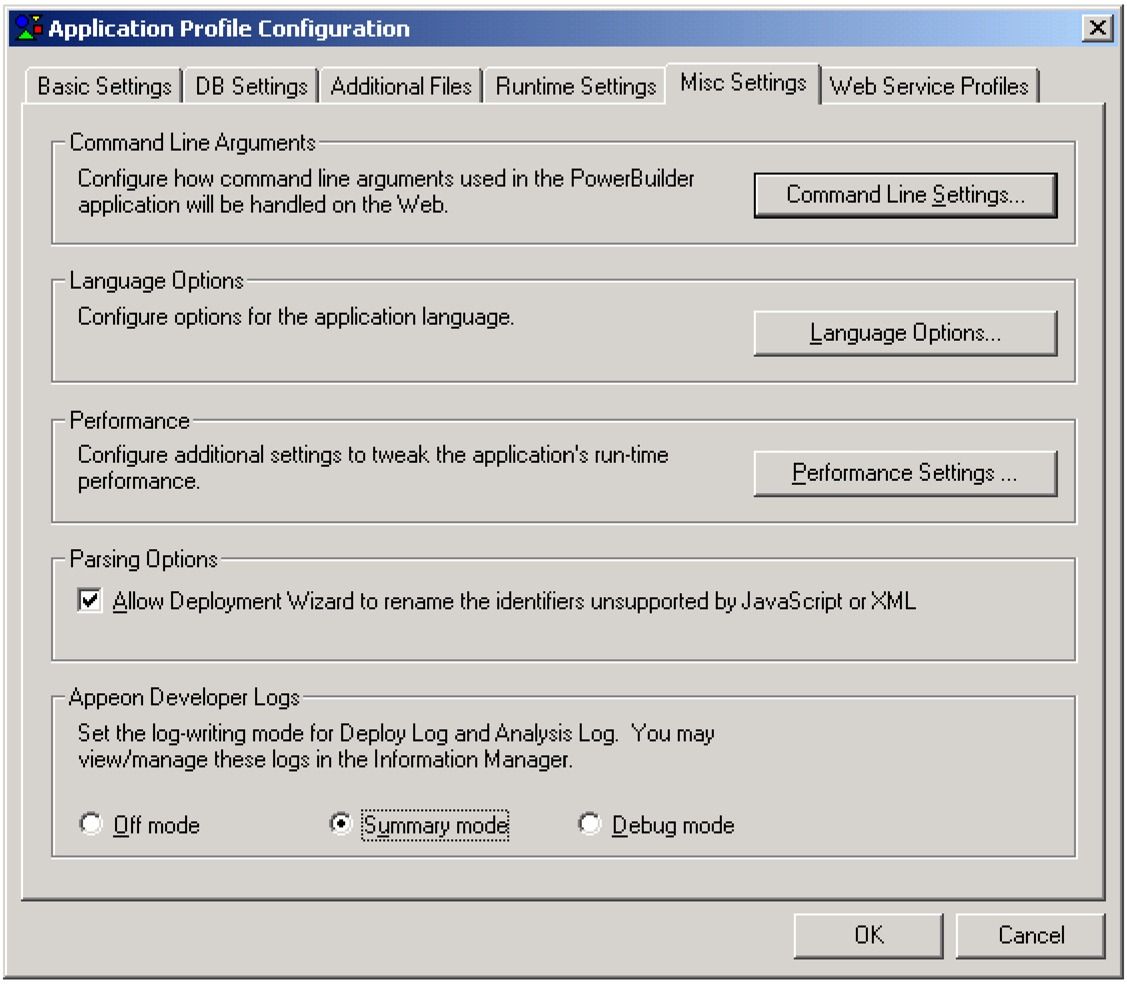
Appeon Developer Logs: Set the log-writing mode for Deploy Log and Analysis Log. You may view/manage these logs in the Information Manager.

○ Off mode ● Summary mode ○ Debug mode

Appeon Developer log-writing mode

The log-writing mode determines the content of the log files generated by Appeon Developer, including the deployment logs and feature analysis logs. Select the desired mode from the Appeon Developer Logs box, as shown in Figure 4-27.

Figure 4-27: Log-writing mode



Command Line Arguments

Configure how command line arguments used in the PowerBuilder application will be handled on the Web.

Language Options

Configure options for the application language.

Performance

Configure additional settings to tweak the application's run-time performance.

Parsing Options

☑ Allow Deployment Wizard to rename the identifiers unsupported by JavaScript or XML

Appeon Developer Logs

Set the log-writing mode for Deploy Log and Analysis Log. You may view/manage these logs in the Information Manager.

○ Off mode ● Summary mode ○ Debug mode

The logging options enable you to select the level of information contained in the logs according to your needs:

Table 4-7: Log writing options

Mode	Description
Off mode	Generates no log files. This mode offers the fastest performance since nothing is written to a log file.
Summary mode (Default)	Generates log files with basic execution information. This is useful for tracking errors that have occurred, but inadequate for detailed troubleshooting. Use this mode once the application is stable.
Debug mode	Generates log files with detailed execution information for troubleshooting obscure and esoteric issues. This is useful for technical support, but performance speed will slow down when using this mode.

You can view or delete these log files. For detailed instructions, refer to Chapter 9: *Using Information Manager*.

Web Service Profiles

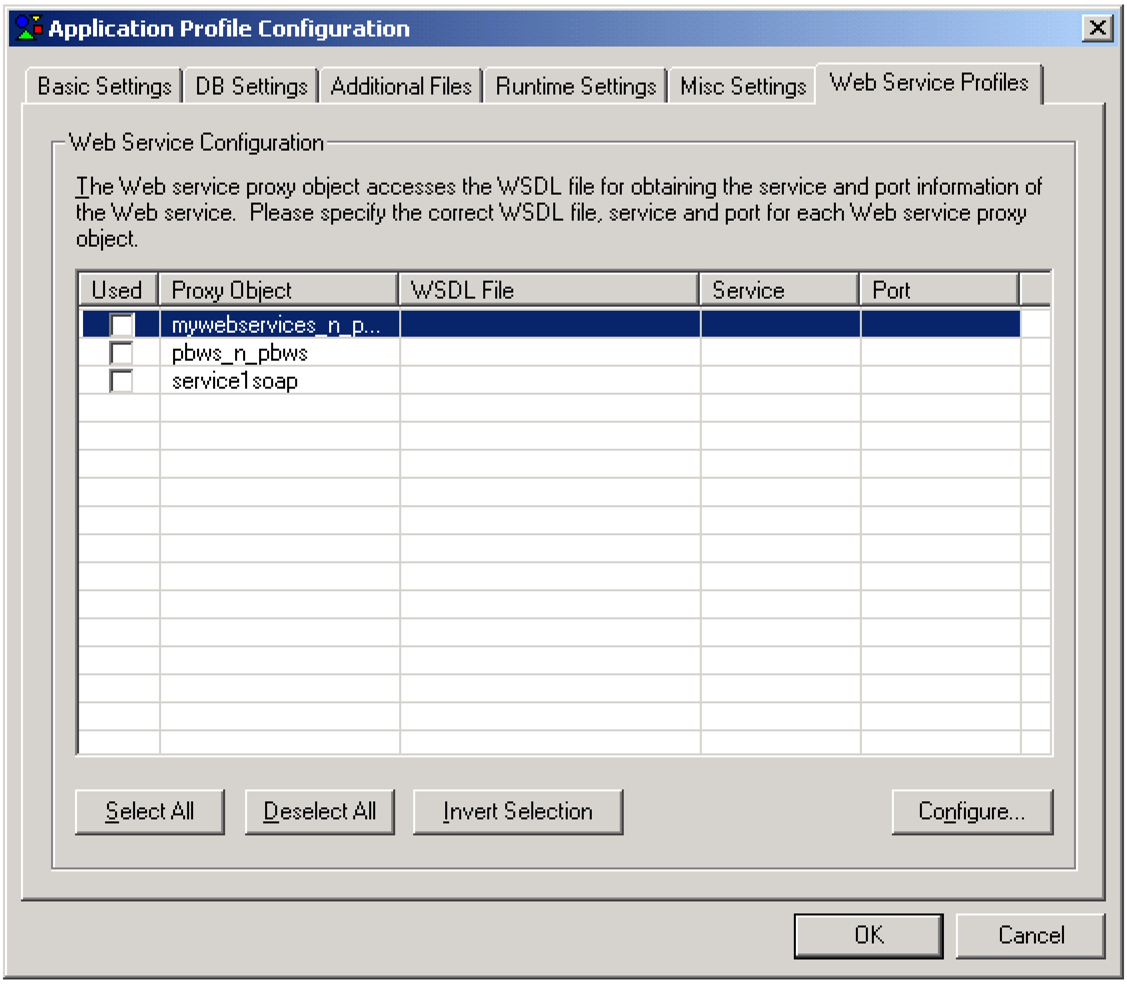
A Web service profile contains the settings for the Web service(s) to be invoked in the Appeon deployed application, including the WSDL file, service name and port to be used by the Web service.

After you specify the PBL files of the application in the Basic Settings tab, the Web service proxy objects, WSDL files, service, and port, which are used in the application, are automatically added to the Web Service Profiles tab, as shown in Figure 4-28.

A valid WSDL file must meet the following requirements:

- The WSDL file must be a .wsdl or .asmx file.
- The WSDL file must reside on the local computer. Appeon cannot access the WSDL file in a remote machine.
- WSDL files used in the same application cannot have the same name, even if they reside in different locations.

Figure 4-28: Web Server Profiles tab



Web Service Configuration

The Web service proxy object accesses the WSDL file for obtaining the service and port information of the Web service. Please specify the correct WSDL file, service and port for each Web service proxy object.

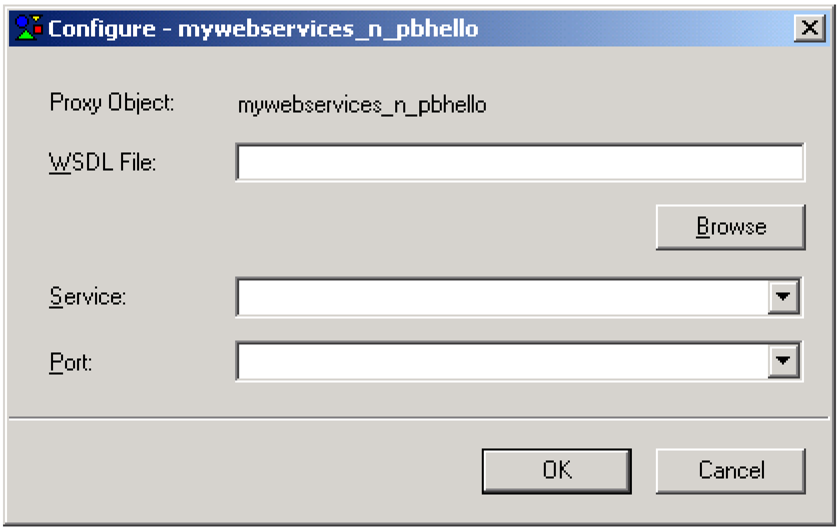
Used	Proxy Object	WSDL File	Service	Port
□	mywebservices_n_p...			
□	pbws_n_pbws			
□	service1soap			

Choose which proxy object(s) will be invoked in the deployed application by selecting the *Used* column of the object. Click *Select All* to select all objects; click *Deselect All* to deselect all objects; or click *Invert Selection* to select all objects which were not selected and deselect the previously selected objects.

If the WSDL file, service and port are not automatically specified for a proxy object, follow the steps below to select them:

Step 1 – Select the proxy object and click *Configure*. The Configure dialog box pops up, as shown in Figure 4-29.

Figure 4-29: Configure dialog box



Configure - mywebservices_n_pbhello

Proxy Object: mywebservices_n_pbhello

WSDL File:

Service:

Port:

Step 2 – Click *Browse* to select a WSDL file on the local computer, and then select the service and port.

A WSDL file may contain multiple services and each service may contain multiple ports.

Step 3 – Click *OK* to save the configuration and return to the *Web Service Profiles* tab page.

4.3.2 Managing database type profiles

Database type profiles configuration is the easiest way to manage the database types that Appeon Deployment Wizard supports for application deployments. You can configure database type profiles in the *DB Type Profiles* tab, and apply a database type profile for an application deployment by selecting the profile in the Appeon Developer Configuration Wizard or in the DB Settings tab of the application profile settings.

Database type profiles in Appeon Developer are different from database profiles in PowerBuilder:

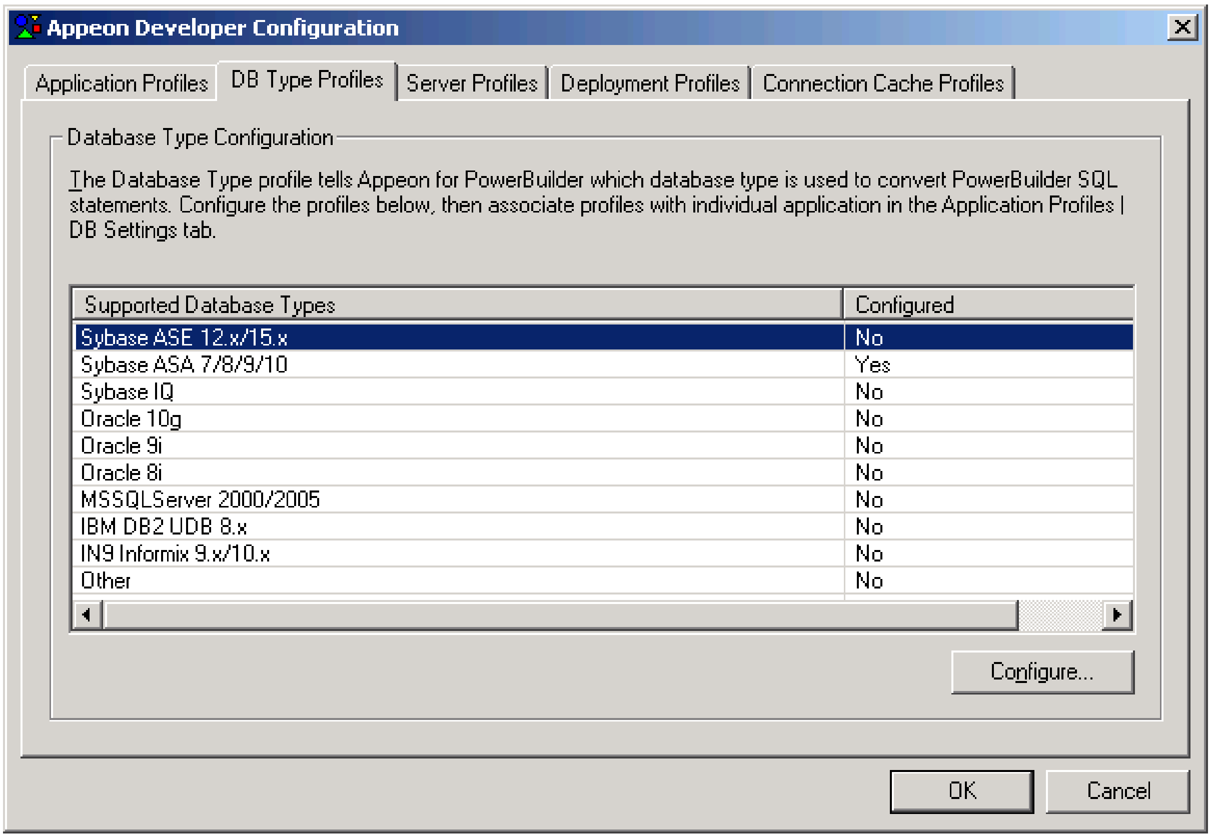
1. You only need to configure one database type profile for a given database type. For example, if application A connects to ASA database d1 and application B connects to ASA database d2, you can use the database type profile “Sybase ASA 7/8/9/10/11” for both applications, although the connection information in the profile “Sybase ASA 7/8/9/10/11” sets up connection to ASA database d1.
2. Database type profiles are not used for setting up connections with application databases. Instead, they are mainly used by Appeon Deployment Wizard to call the relevant database driver for converting PowerBuilder SQL statements.

To create or modify a database type profile, take the following steps:

Step 1 – Click the *Configure* button on the Appeon Developer toolbar, and select the *DB Type Profiles* tab in the Application Developer Configuration page as shown in Figure 4-30.

The *Database Type* column lists all the database types that are supported by Appeon; and the *Configured* column indicates whether a database type profile has been configured.

Figure 4-30: DB Type Profiles



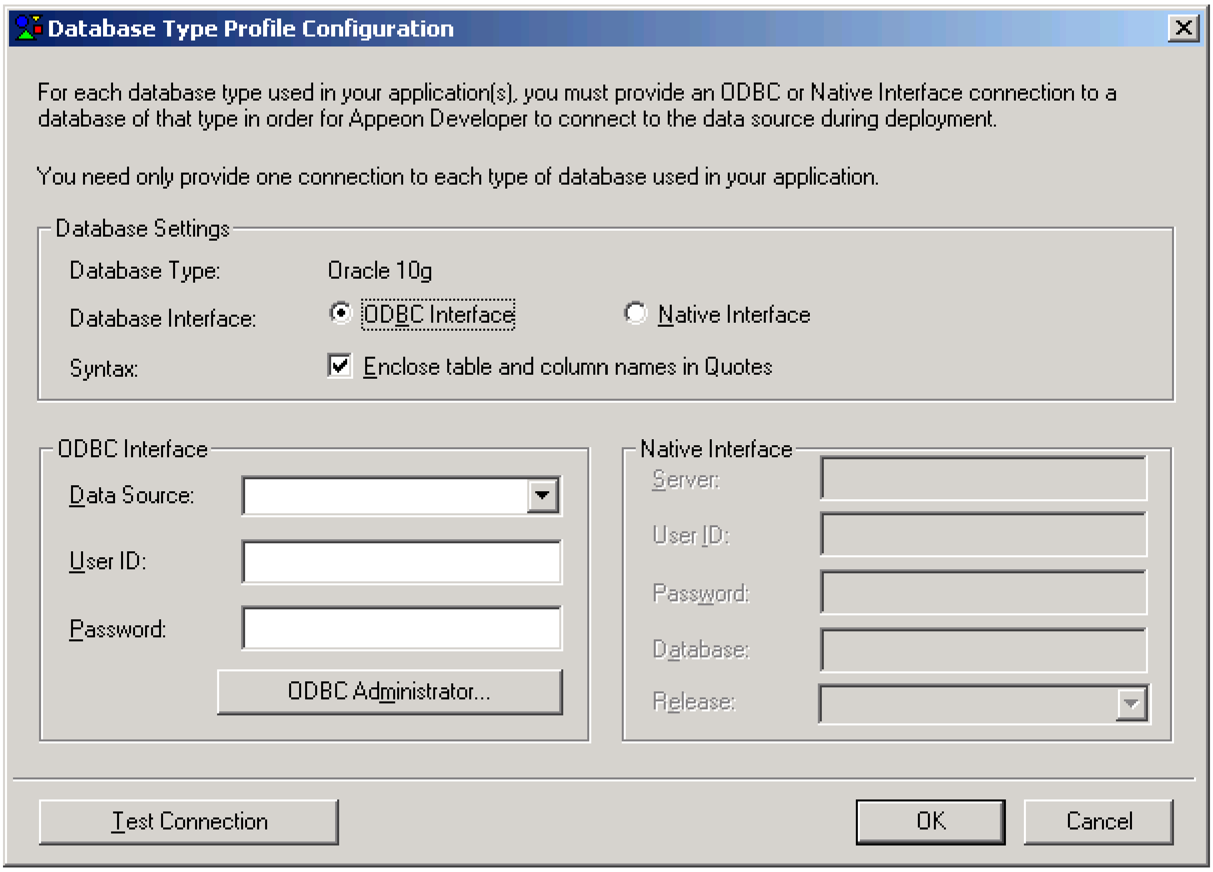
Database Type Configuration

The Database Type profile tells Appeon for PowerBuilder which database type is used to convert PowerBuilder SQL statements. Configure the profiles below, then associate profiles with individual application in the Application Profiles | DB Settings tab.

Supported Database Types	Configured
Sybase ASE 12.x/15.x	No
Sybase ASA 7/8/9/10	Yes
Sybase IQ	No
Oracle 10g	No
Oracle 9i	No
Oracle 8i	No
MSSQLServer 2000/2005	No
IBM DB2 UDB 8.x	No
IN9 Informix 9.x/10.x	No
Other	No

Step 2 – Select a database type (for example, Oracle 10g), and click the *Configure* button to configure the database type profile (as shown in Figure 4-31).

Figure 4-31: Database Type configuration



Database Type Profile Configuration

For each database type used in your application(s), you must provide an ODBC or Native Interface connection to a database of that type in order for Appeon Developer to connect to the data source during deployment.

You need only provide one connection to each type of database used in your application.

Database Settings

Database Type: Oracle 10g

Database Interface: ● ODBC Interface ○ Native Interface

Syntax: ☑ Enclose table and column names in Quotes

ODBC Interface

Data Source:

User ID:

Password:

Step 3 – Specify the information needed for setting up the database type profile.

ODBC Interface – Select this option if using an ODBC interface. Specify the ODBC data source, User ID, and Password.

Table 4-8: ODBC interface settings

Setting	Instructions
Data Source	All the data sources configured in Windows ODBC Administrator are listed. Select a proper data source of the configured database type from the dropdown listbox. It is not necessary to specify the actual databases that the application uses. Click *ODBC Administrator* to configure an ODBC data source if it is not available.
User ID	Type in a recognized (set) user name to login to the database. If no user name has been set, leave this field blank.
Password	Type in the password used to log in to the database. If no password has been set, leave this field blank.

Native Interface – Select this option if using a native database interface. Specify the server name/IP, User ID, Password, and the database name.

Note: Not all the listed database types are supported by Appeon for the native database interface. If Sybase IQ, Sybase ASA 7/8/9/10/11 or IBM DB2 is your database type, the Native Interface options are disabled.

Table 4-9: Native database interface settings

Setting	Instructions
Server	Enter the machine name or IP address of the server where the target database resides.
User ID	Type in a recognized (set) user name to login to the database. If no user name has been set, leave this field blank.
Password	Type in the password used to login to the database. If no password has been set, leave this field blank.
Database	Specify the name of the database that will be used by the application.
Release	(ASE only) Specify the version number of ASE database.

Step 4 – Specify the “Enclose table and column names in double quotes” option. It determines whether Appeon Developer encloses the names of tables, columns, indexes, and constraints in double quotes when it generates the database syntax during deployment.

Notes:

1) When this option is selected, verify that the quoted_identifier setting is set to ON in the database server.

2) Clear the check box of this option when the updated table name of DataWindow contains the owner name.

Step 5 – Click *Test Connection* to test the connection to the database.

4.3.3 Managing server profiles

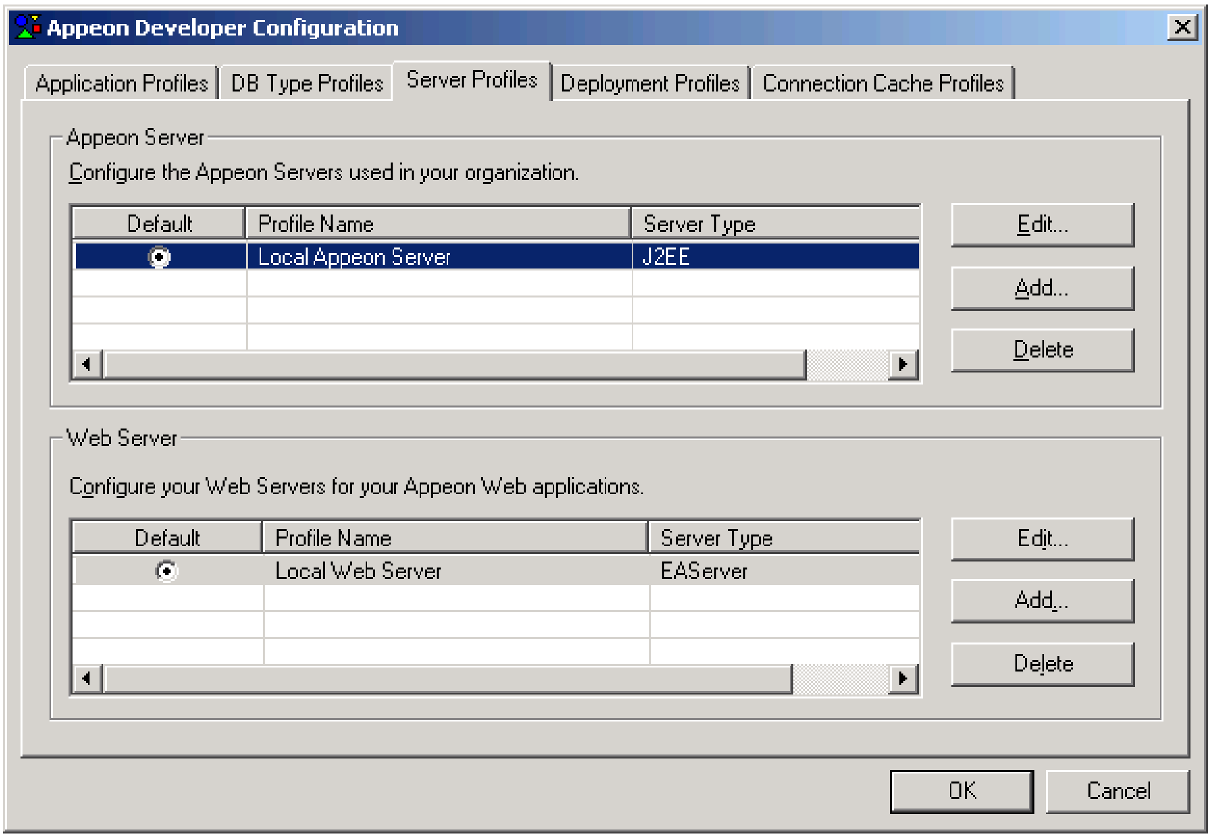
A server profile is a set of parameters for Appeon Developer to connect to, and deploy applications to, a particular Appeon Server or Web server. Creating and using server profiles is the easiest way to manage your application deployments because you can:

- Select one or more Appeon Server profiles and Web Server profiles to set up a deployment profile.
- Easily add, edit, and remove server profiles.
- Test connections in server profiles. If the connection is successful, it means that the application can be deployed to the server configured in the server profile.
- Manage which applications are accessible from the Appeon Developer shortcut *Run* button. The shortcut lists the applications that are deployed to the default server profiles.

4.3.3.a Server Profiles tab page

There are two main parts in the *Server Profiles* tab page, as shown in Figure 4-32. The upper part is for management of Appeon Server profiles, while the lower part is for management of Web server profiles.

Figure 4-32: Server Profiles



Configure the Appeon Servers used in your organization.

Default	Profile Name	Server Type
●	Local Appeon Server	J2EE

Configure your Web Servers for your Appeon Web applications.

Default	Profile Name	Server Type
●	Local Web Server	EAServer

Default server profiles

You can select an Appeon Server profile or a Web Server profile as the DEFAULT server profile in the *Server Profiles* tab page. Appeon Developer's *Run* shortcut reads the default server profiles and lists the applications that are deployed to default server profiles, enabling you to access them quickly. Applications that are deployed to non-default server profiles are not available in the shortcut. For details about the shortcut, refer to Section 8.3.2: *Running Web applications*.

Edit, add or delete server profiles

Two sets of *Edit*, *Add*, and *Delete* buttons are available in the *Server Profiles* tab page for you to modify, create, or remove Appeon Server or Web Server profiles. Sections 4.3.3.b and 4.3.3.c provide instructions on how to configure the properties for an Appeon Server or Web Server profile when you modify or create a new server profile.

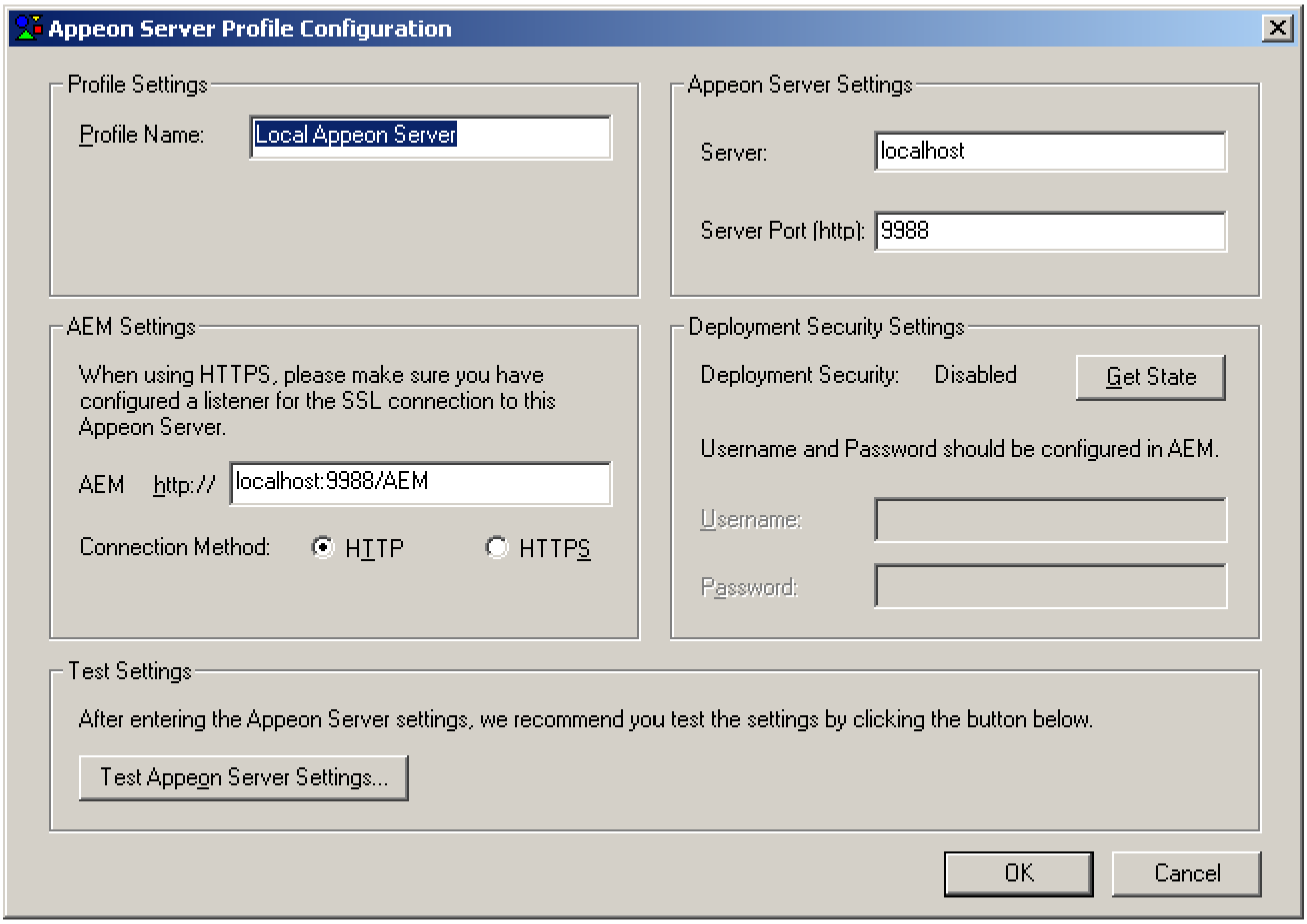
4.3.3.b Appeon Server profile settings

Before you edit or add an Appeon Server profile, make sure the Appeon Server specified in the profile is running, and that the Appeon Developer computer can successfully connect to the Appeon Server computer.

If you want to deploy to an Appeon Server cluster, you must create profiles for each participating Appeon Server and deploy the application to all Appeon Servers in the cluster.

Figure 4-33 shows the Appeon Server Profile Configuration window that displays when you click the *Edit* or *Add* button in the *Appeon Server* group box of the *Server Profiles* tab page.

Figure 4-33: Appeon Server Profile



Appeon Server Profile Configuration

Profile Settings

Profile Name: Local Appeon Server

Appeon Server Settings

Server: localhost

Server Port (http): 9988

AEM Settings

When using HTTPS, please make sure you have configured a listener for the SSL connection to this Appeon Server.

AEM http:// localhost:9988/AEM

Connection Method: ● HTTP ○ HTTPS

Deployment Security Settings

Deployment Security: Disabled

Username and Password should be configured in AEM.

Username:

Password:

Test Settings

After entering the Appeon Server settings, we recommend you test the settings by clicking the button below.

Table 4-10 lists detailed instructions for how to specify the properties for an Appeon Server profile.

Table 4-10: Instructions for creating an Appeon Server profile

	Property	Instructions
Profile Settings	Profile Name	Assign a name to the Appeon Server profile. You should use names that are easy to remember and identify (for example "Appeon Server for Test" or "Remote Appeon Server").
Appeon Server Settings	Server	Enter the IP address or the machine name of Appeon Server.
	Server Port (http)	Enter the HTTP port number used by Appeon Server. Note: Do not enter an HTTPS port here, Appeon Developer must use an HTTP port to connect with Appeon Server, though the Appeon Web application can be connected with an HTTPS port.

AEM Settings	AEM URL	The URL for Appeon Enterprise Manager (AEM) will be automatically generated after Server and Server Port are specified. The URL will be in the following format: http://*server:port*/AEM, (for example, http://localhost:9988/AEM). Note: 1) If you use EAServer as Appeon Server, the Appeon installation program creates an HTTP listener (localhost: 9988) for Appeon Server. If the Appeon Server configured is a local computer, the following URL should always work: http://localhost:9988/AEM. 2) DO NOT use a "localhost" listener in a production environment.
	Connection method	Select HTTP or HTTPS to connect to AEM. If the Web Server is configured as an SSL Web Server, check the HTTPS (secure) option. Otherwise, check the HTTP (insecure) option.
Deployment Security Settings	Deployment Security	The *Get State* button can get the state of the Deployment Security settings in AEM. If Deployment Security is set to ON, the state will be "Enabled", and you must specify the deployment user name and password. By default, the Deployment Security in AEM is OFF, so the state is "Disabled".
	Username	Enter the username used to deploy the application. The username and password for the deployment security feature are configured in the AEM. The username can be left blank if the deployment security feature is turned off in AEM.
	Password	Enter the password used to deploy the application. The username and password for the deployment security feature are configured in the AEM. The password can be left blank if the deployment security feature is turned off in AEM.

After profile configuration, perform the following steps to make sure the Appeon Server profile can be successfully used for application deployments:

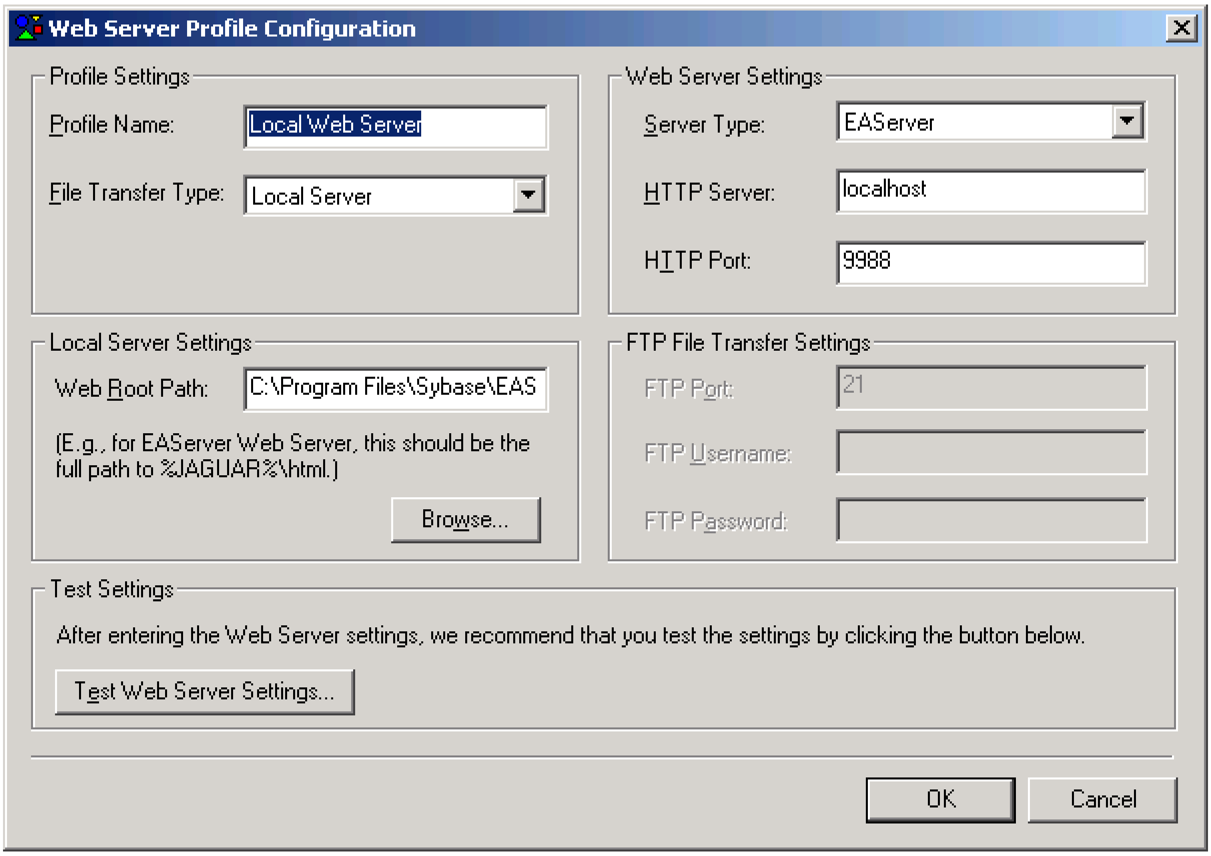
- Check whether the deployment security settings are configured correctly. The username and password in the deployment security settings must be the same as those configured in AEM. Make sure you retrieve the correct deployment username and password from the AEM administrator.
- Test Appeon Server settings by clicking the *Test Appeon Server Settings* button. DO NOT proceed to the next step until the testing succeeds.

4.3.3.c Web Server profile settings

Before you edit or add a Web Server profile, make sure the Web Server specified in the profile is running so the Appeon Developer computer can successfully connect to the Web Server computer.

Figure 4-34 shows the Web Server Profile Configuration window that displays when you click the *Edit* or *Add* button in the *Web Server* group box of the *Server Profiles* tab page.

Figure 4-34: Web Server profile configuration window



Web Server Profile Configuration

Profile Settings

Profile Name: Local Web Server

File Transfer Type: Local Server

Web Server Settings

Server Type: EAServer

HTTP Server: localhost

HTTP Port: 9988

Local Server Settings

Web Root Path: C:\Program Files\Sybase\EAS

(E.g., for EAServer Web Server, this should be the full path to %JAGUAR%\html.)

FTP File Transfer Settings

FTP Port: 21

FTP Username:

FTP Password:

Test Settings

After entering the Web Server settings, we recommend that you test the settings by clicking the button below.

Table 4-11 lists detailed instructions for how to specify the properties for a Web Server profile.

Table 4-11: Instructions for creating a Web Server profile

	Property	Instructions
Profile Settings	Profile Name	Assign a name to the Web Server profile. You should use names that are easy to remember and identify such as "Web Server for Test" or "Production Web Server".
	File Transfer Type	Use **Local Server** if the Web Server is a local computer. Use **Remote Server** if the Web Server locates remotely.
Web Server Settings	Server Type	Select the Web server type. Appeon supports the following Web Server types: Apache, Appeon Server Web Server, Microsoft IIS, WebLogic, and WebSphere. Appeon Server Web Server indicates EAServer HTTP server.
	HTTP Server	Enter the IP address or the machine name of Web Server.
	HTTP Port	(For Local Server file transfer only) Enter the Web Server port number. Note: Do not enter an HTTPS port here, even though the Web server is configured as an SSL Web server, because Appeon Developer must use an HTTP port to connect with Web server. However, you can access the deployed application with the HTTPS port.

Local Server Settings	Web Root Path	Enter the home directory of the Web server (for example: For EAServer HTTP Server, the home directory is C:\Program Files\Sybase\EAServer6\html, For IIS server, the home directory is C:\Inetpub\wwwroot, For Weblogic8, the home directory is E:\bea\user_projects\domains\appeon_domain\applications, For Weblogic9/10, the home directory is E:\bea\user_projects\domains\appeon_domain\autodeploy, For WebSphere, the home directory is E:\IBM\WebSphere6\AppServer\installableApps). Click the Browse button to select the home directory of the Web Server. Click the *Browse* button to select the home directory of the Web server.
FTP File Transfer Settings	FTP Port	Enter the port number of the FTP server. The typical FTP port is port 21.
	FTP Username	Enter the username for FTP login. If the FTP server offers “anonymous” access then the username should be “anonymous”.
	FTP Password	Enter the password for FTP login. If no password is set for the FTP server or the FTP Username is anonymous, leave this field blank.

After the profile configuration, perform the following steps to make sure the Web Server profile can be successfully used for application deployments:

- Test Web Server settings by clicking the *Test Web Server Settings* button. DO NOT proceed to the next step until the testing succeeds.

  If the Web Server is a remote server, refer to the *Two requirements for FTP settings* section to make sure the configuration for the Web Server profile is successful.

  If the Web Server is an SSL Web server, refer to the *If the Web Server is an SSL Web Server* section to make sure the configuration for the Web Server profile is successful.

Two requirements for FTP settings

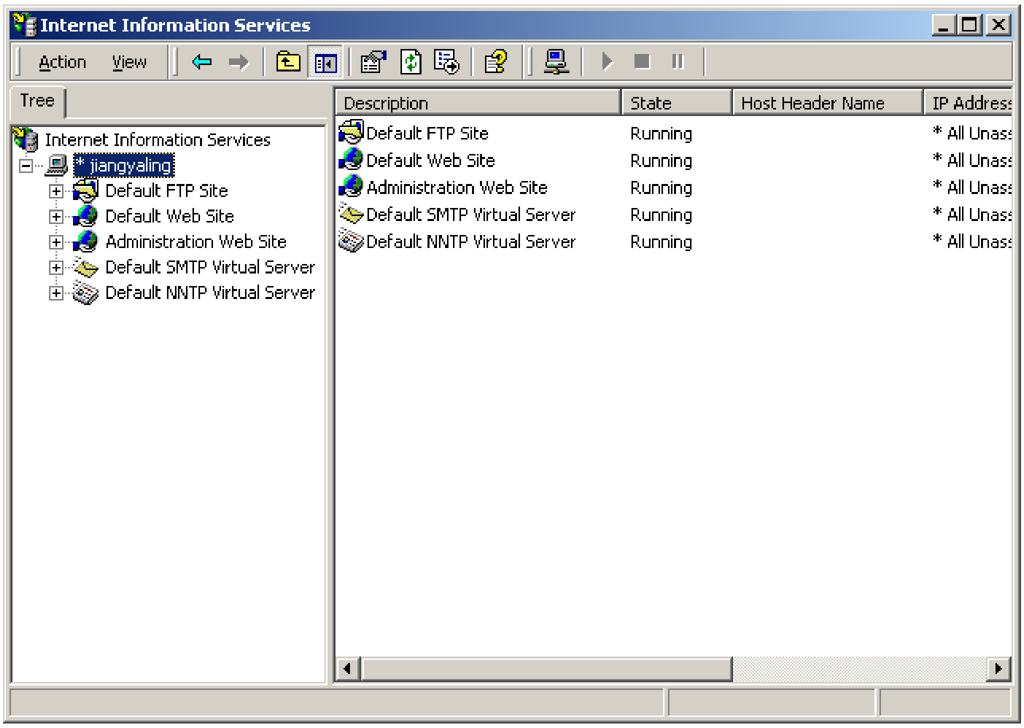
If you configure a Web Server profile for a remote Web Server, make sure the FTP settings in the Web Server profile meet the following two requirements:

1. The user name and password for accessing the FTP server should have permission to read and write files to the FTP server of the Web Server.
2. The FTP home directory should be mapped to the Web Server’s Web root.

The following steps use the Microsoft IIS FTP service as an example to show you how to fulfill the requirements. You should find that the settings for other FTP types are similar to the settings for Microsoft IIS FTP.

Step 1 – On the FTP server (Web Server), open the Internet Services Manager in Administrative Tools, as shown in Figure 4-35.

Figure 4-35: FTP configuration



Internet Information Services
- jiangyaling
  - Default FTP Site
  - Default Web Site
  - Administration Web Site
  - Default SMTP Virtual Server
  - Default NNTP Virtual Server

Description	State	Host Header Name	IP Addres
Default FTP Site	Running		* All Unas
Default Web Site	Running		* All Unas
Administration Web Site	Running		* All Unas
Default SMTP Virtual Server	Running		* All Unas
Default NNTP Virtual Server	Running		* All Unas

Step 2 – Right click on *Default FTP Site* and select *Properties* in the popup menu. The Default FTP Site Properties Window is displayed.

Step 3 – Go to the *Home Directory* tab and verify that:

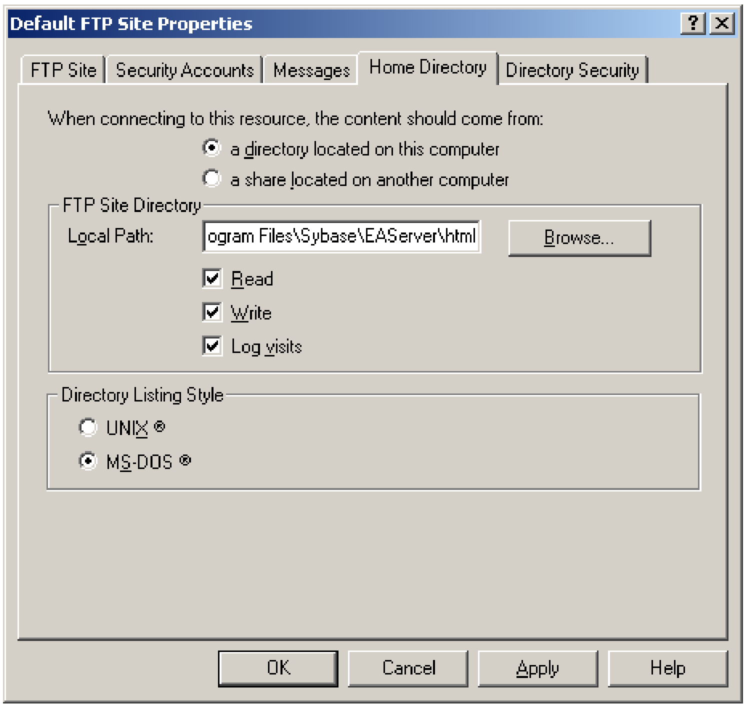
- The Local Path is the full path to the Web Server document root.

  If the Web Server is an Appeon Server Web Server running on Windows, the local path is the full path to the EAServer document root: \%JAGUAR%\html (Figure 4-36), if the Web Server is an IIS server, the local path is C:\Inetpub\wwwroot.

  %JAGUAR% is the EAServer installation directory (for example: C:\Program Files\Sybase\EAServer).

- The Write property of the FTP Site Directory is enabled.

Figure 4-36: FTP site properties



When connecting to this resource, the content should come from:

● a directory located on this computer

○ a share located on another computer

FTP Site Directory

Local Path: ogram Files\Sybase\EAServer\html

☑ Read

☑ Write

☑ Log visits

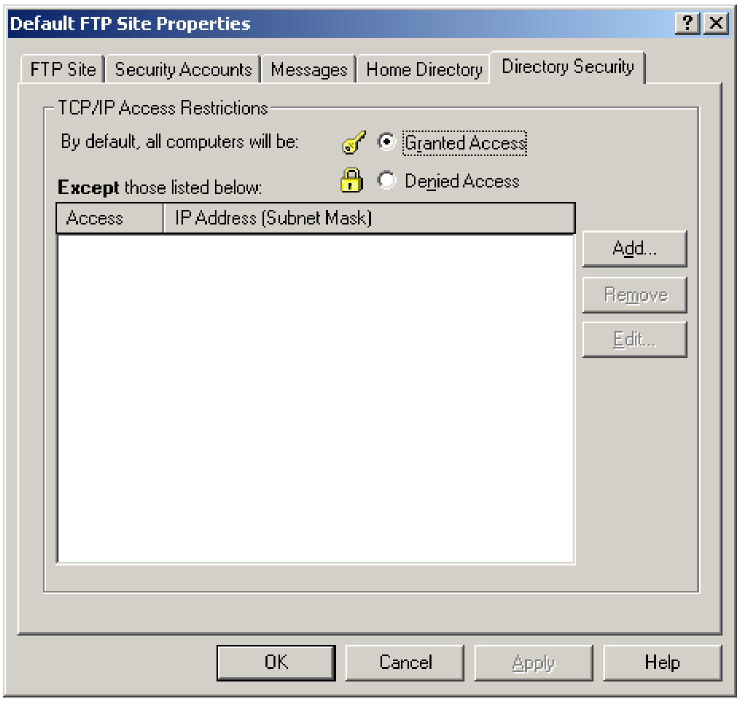
Directory Listing Style

○ UNIX ®

● MS-DOS ®

Step 4 – Go to the *Directory Security* tab and verify that the *Granted Access* option is checked, as shown in Figure 4-37.

Figure 4-37: Directory security



TCP/IP Access Restrictions

By default, all computers will be: ● Granted Access ○ Denied Access

Except those listed below:

Access	IP Address (Subnet Mask)

If the Web Server is an SSL Web Server

If the Web Server is an SSL or secure Web Server, you can configure the Web Server profile following the configuration instructions provided in Table 4-11, except for the following two points:

1. You must configure an HTTPS listener and port number for the Web Server.

2. You must specify an HTTP (not HTTPS) listener and its port number in the HTTP server settings of the Web Server profile, because Appeon Developer must use the HTTP protocol when deploying Web files to the server. After the application is deployed to the Web server, you can input "https" instead of "http" and the HTTPS port in the URL to access the application, (for example, https://192.0.0.80:8181/appeondemo/).

4.3.4 Managing deployment profiles

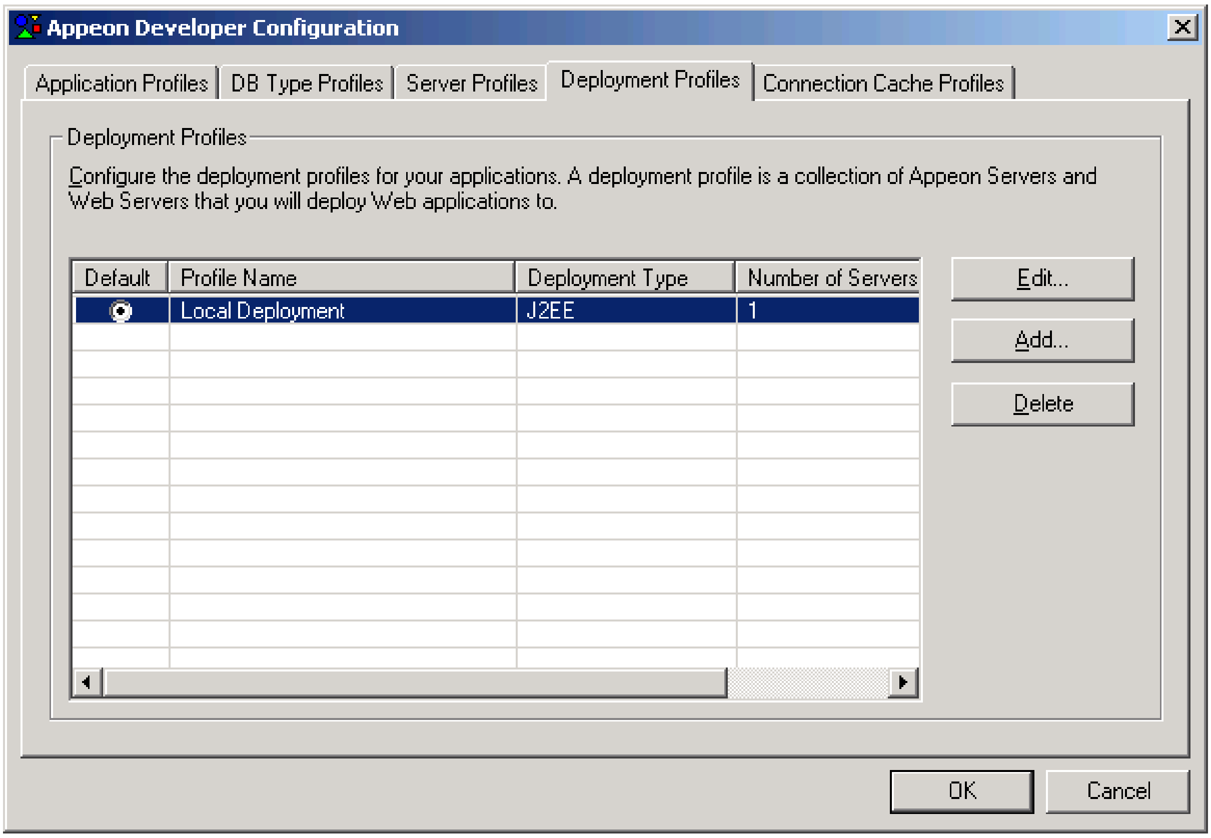
A deployment profile associates specified Web server(s) and Appeon Server(s) as a group used for Web deployment. You may create several deployment profiles (e.g. local deployment, test deployment, production deployment, and so on), and set the most commonly used profile as the default profile.

A deployment profile is based on server profiles. Before configuring the deployment profiles, make sure you have set up one server profile for each of the Appeon Servers and Web servers to be used for Web conversion. You can add as many deployment profiles as you need, but there can be only one default deployment profile.

4.3.4.a Deployment Profiles tab page

On the *Deployment Profiles* tab page, you can edit, add, or delete deployment profiles and specify a default deployment profile.

Figure 4-38: Deployment Profile tab page



Appeon Developer Configuration

Application Profiles | DB Type Profiles | Server Profiles | Deployment Profiles | Connection Cache Profiles

Deployment Profiles

Configure the deployment profiles for your applications. A deployment profile is a collection of Appeon Servers and Web Servers that you will deploy Web applications to.

Default	Profile Name	Deployment Type	Number of Servers
●	Local Deployment	J2EE	1

Table 4-12 provides instructions for how to use the Deployment Profile tab page.

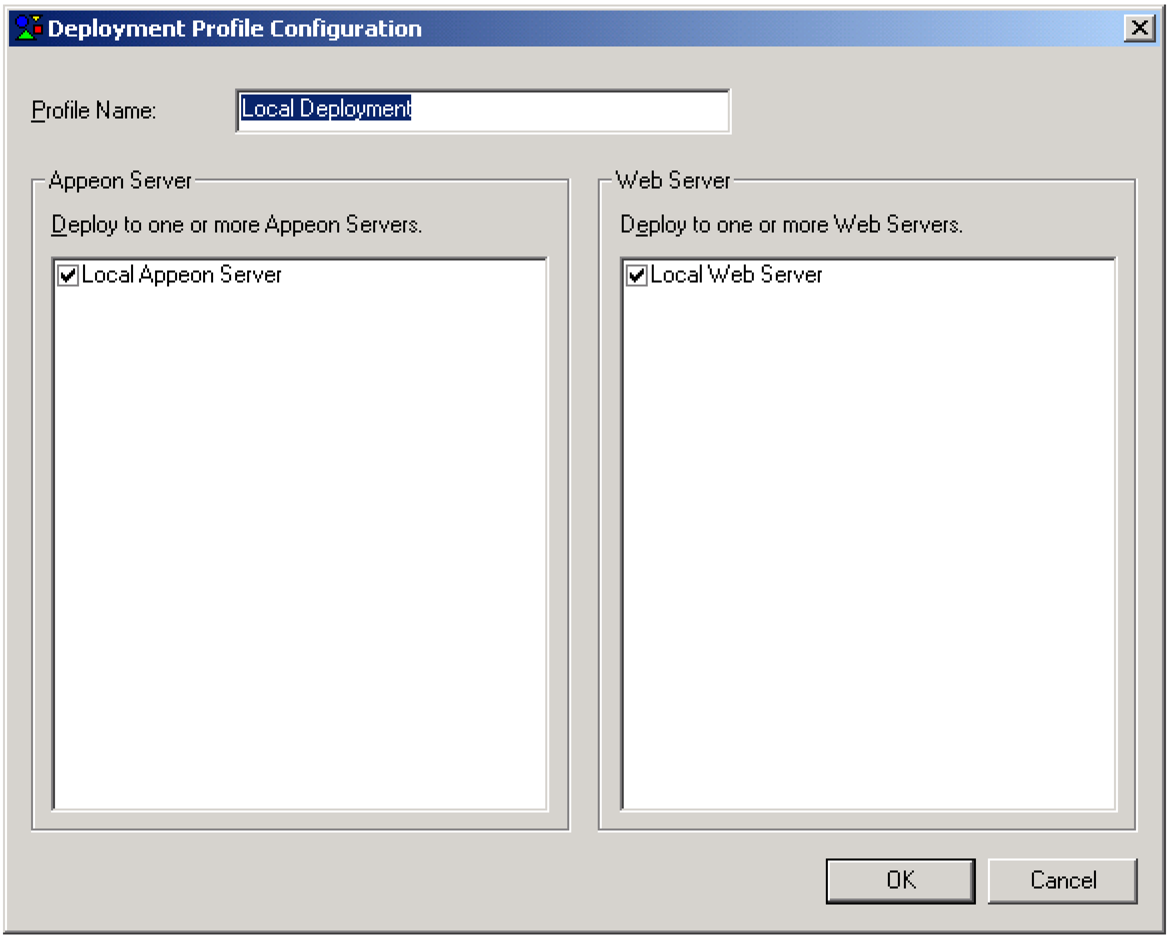
Table 4-12: Instructions to use the Deployment Profile tab

Use this button	To…
Default radio button	Specify the default deployment profile used for Web deployment.
Edit button	Modify an existing deployment profile. This will open the Deployment Profile Configuration dialog box.
Add button	Create a new deployment profile. This will open the Deployment Profile Configuration dialog box.
Delete button	Remove a deployment profile.

4.3.4.b Deployment profile settings

In the Deployment Profile Configuration dialog box, set up the necessary configuration, as shown in Figure 4-39.

Figure 4-39: Deployment Profile



Deployment Profile Configuration

Profile Name: Local Deployment

Appeon Server

Deploy to one or more Appeon Servers.

☑ Local Appeon Server

Web Server

Deploy to one or more Web Servers.

☑ Local Web Server

Table 4-13 provides detailed instructions to specify the deployment profile settings.

Table 4-13: Deployment Profile settings

In this field	You can…
Profile Name	Type the deployment profile name. You can use names that are easy to remember and identify such as “Test Deployment”, “Remote Deployment”, and so on.
Appeon Server	Select a check box to include the Appeon Server in the deployment profile. All the Appeon Server profiles you have created are listed. More than one Appeon Server can be selected.
Web Server	Select a check box to include the Web Server in the deployment profile. All the Web server profiles are listed. More than one Web server can be selected.

Click *OK* to save the new settings and return to the *Deployment Profiles* tab.

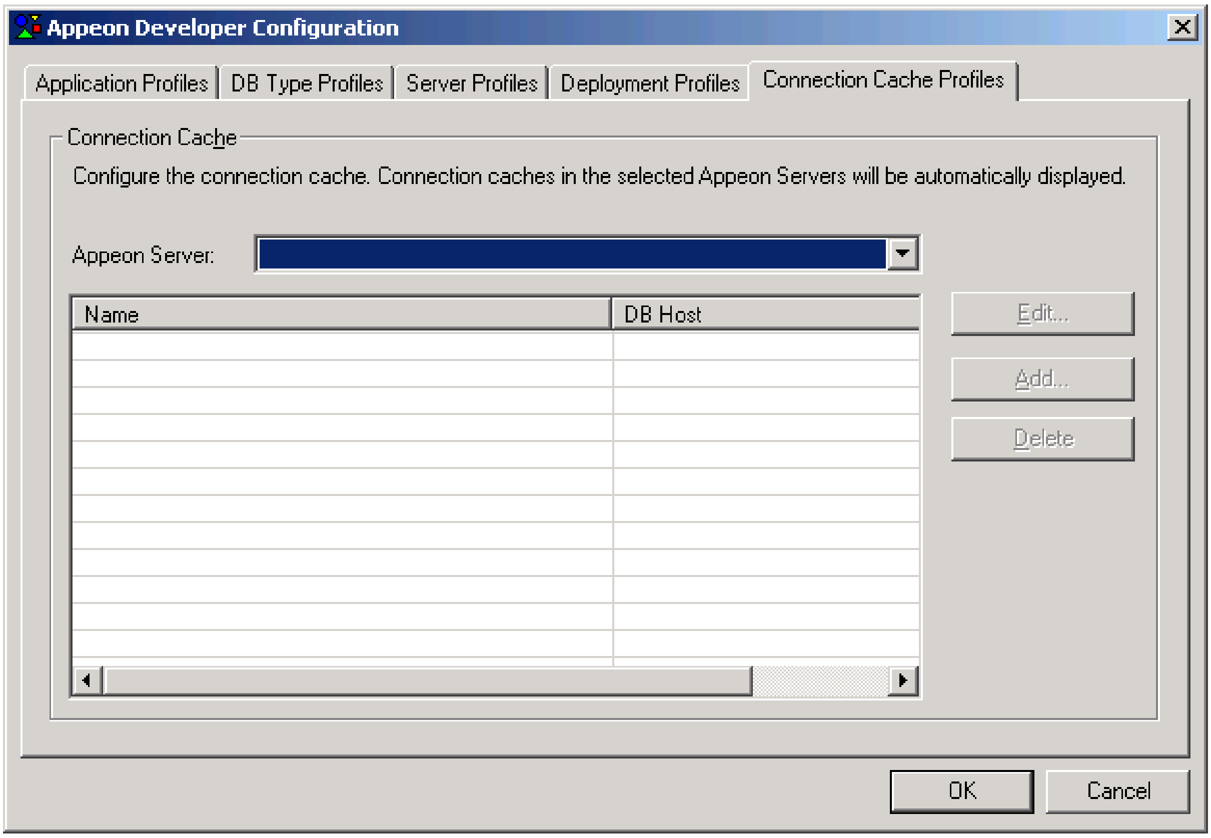
4.3.5 Managing connection cache profiles

Note: This Connection Cache Profiles tool works for EAServer 5.x, but does not work for EAServer 6.x, WebLogic, and WebSphere.

To create connection cache\data source for EAServer 6.x, WebLogic, and WebSphere, refer to the *Database Connection Setup* chapter in the *Appeon Server Configuration Guide*.

Appeon Web applications use JDBC connection caches created in Appeon Server to connect to the database. “Connection Cache Profile” tab (Figure 4-40) allows you to maintain a storage of connection caches which you can specify to use for Appeon applications. Its function is the same as the one provided in AEM, and changes made in the tab or AEM are automatically synchronized.

Figure 4-40: Connection cache profile



Appeon Developer Configuration

Connection Cache

Configure the connection cache. Connection caches in the selected Appeon Servers will be automatically displayed.

Appeon Server:

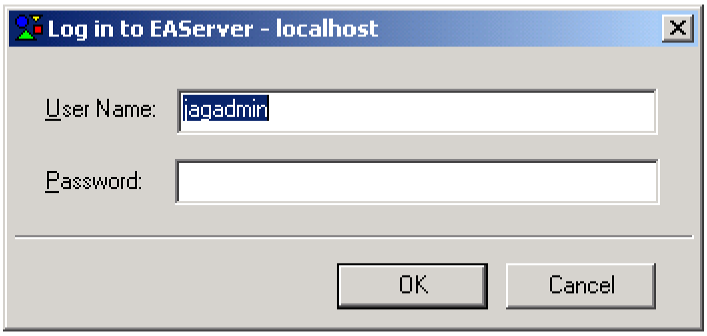
Name	DB Host

Follow the instructions below to create a connection cache on the Connection Cache Profile tab:

Step 1 – Select an Appeon Server from the *Appeon Server* list box. Only the configured Appeon Server profiles are listed.

Step 2 – When it is EAServer 5.x, input the correct user name and password to log in to the EAServer application server, as shown in Figure 4-41.

Figure 4-41: Login dialog box



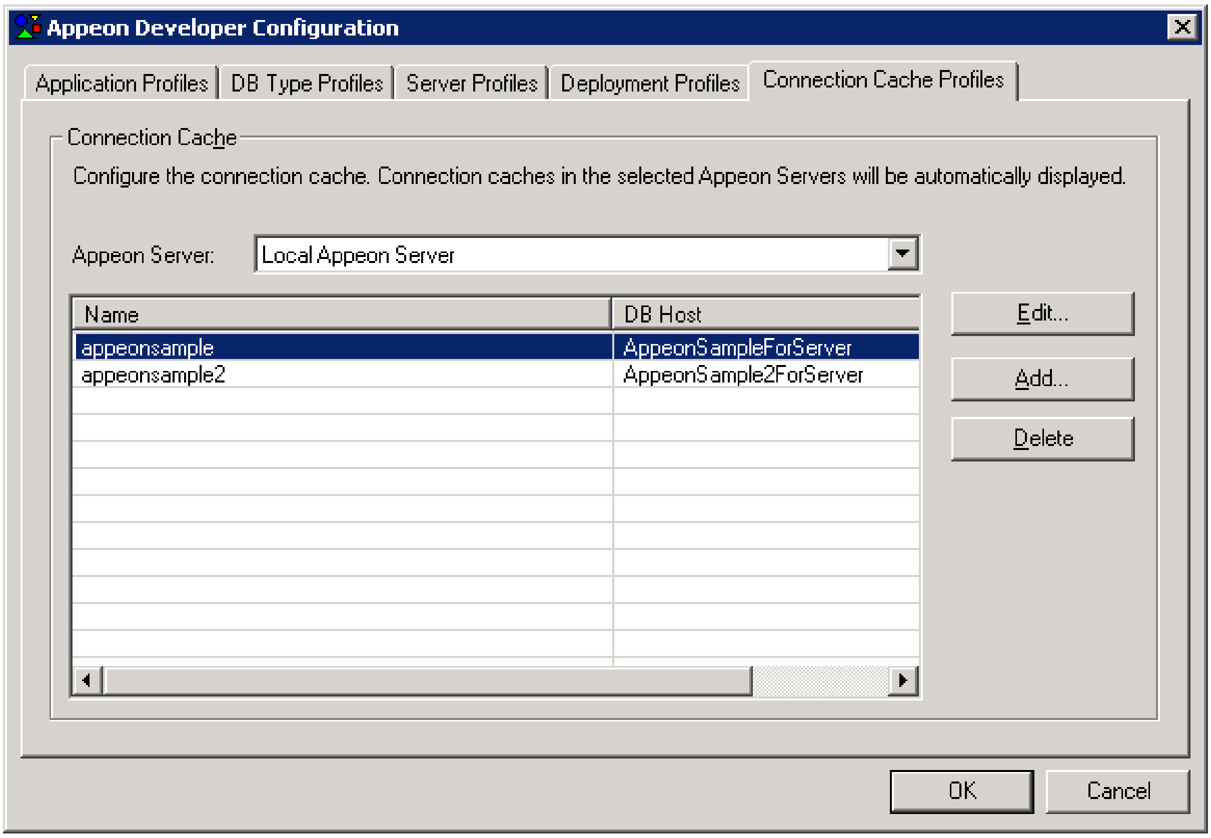
Log in to EAServer - localhost

User Name: jagadmin

Password:

The connection caches existing on the application server will be displayed, as shown in Figure 4-42.

Figure 4-42: Connection caches existing on the application server



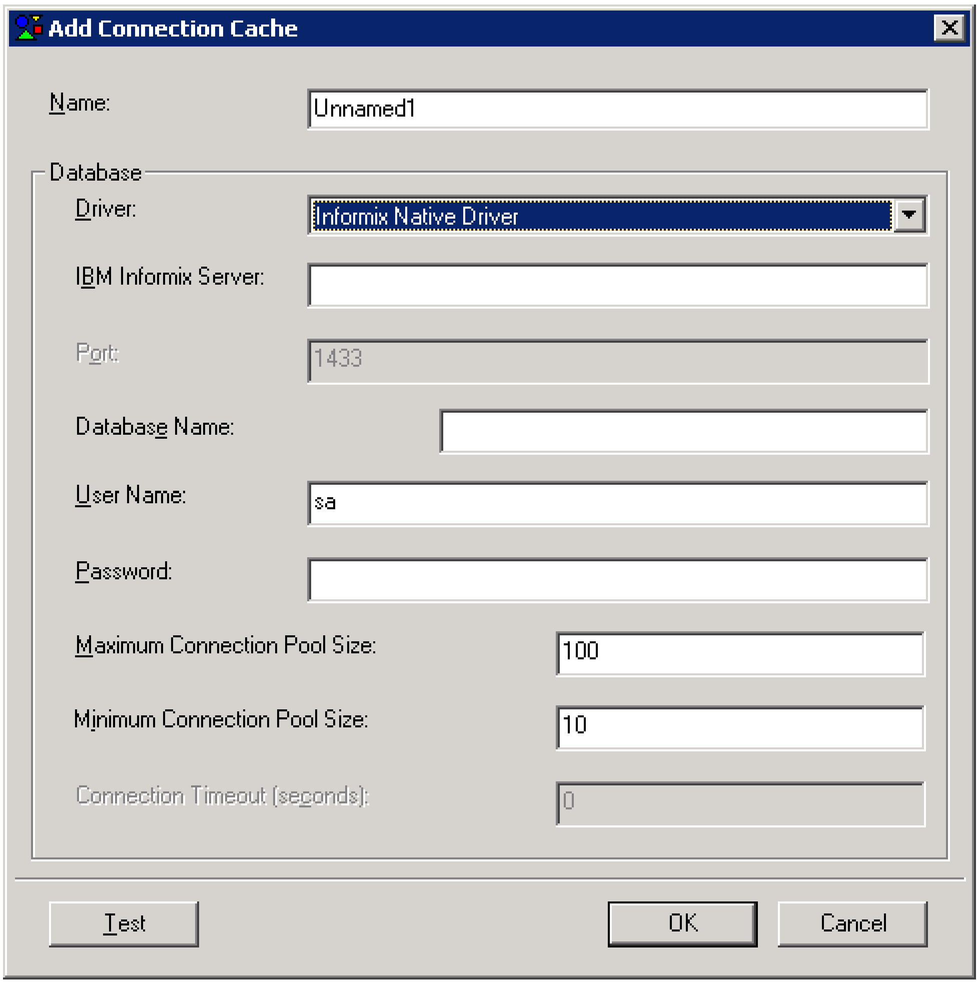
Configure the connection cache. Connection caches in the selected Appeon Servers will be automatically displayed.

Appeon Server: Local Appeon Server

Name	DB Host
appeonsample	AppeonSampleForServer
appeonsample2	AppeonSample2ForServer

Step 3 – Click the *Add* button to create a connection cache in the application server. The Add dialog box is displayed, as shown in Figure 4-43.

Figure 4-43: Add a connection cache



Add Connection Cache

Name: Unnamed1

Database

Driver: Informix Native Driver

IBM Informix Server:

Port: 1433

Database Name:

User Name: sa

Password:

Maximum Connection Pool Size: 100

Minimum Connection Pool Size: 10

Connection Timeout (seconds): 0

Table 4-14 provides instructions for how to specify the connection cache settings.

Table 4-14: Instructions to specify connection cache settings

Settings	Instructions
Name	Type the name of the connection cache. The connection cache name can contain a combination of letters, underscores (“_”), dot (“.”), and numbers. Do not use double-byte characters (such as Chinese, Korean or Japanese characters) or leave a leading or trailing space.
Driver	Select the driver type for the connection cache.
Host or IBM Informix Server	Type the database host.
Port	Type the database port.
Database Name	Type the database name.
ODBC Data Source	Type the data source name only when the driver type is ODBC-JDBC Bridge.
User Name	Type the database login username. The username is set on the database server.
Password	Type the database login password. The password is set on the database server.

Maximum Connection Pool Size	Specify the minimum number of connections Appeon Server opens and pools on startup.
Minimum Connection Pool Size	Specify the maximum number of connections Appeon Server opens and pools on startup.
Connection Timeout	Specify connection timeout.

Refer to the *Appeon Server Configuration Guide* for how to specify the settings for different JDBC drivers and database types.

5 Using UFA Tool

5.1 Overview

Unsupported Features Analysis (UFA) tool helps you analyze an application for unsupported features and heavy windows. Correspondingly, two reports will be generated: one is the Unsupported Features Analysis report (UFA Report) and the other is the Heavy Window report.

Table 5-1 gives brief descriptions of both reports.

Table 5-1: UFA Report and Heavy Window report

	UFA Report	Heavy Window Report
Purpose of the report	Enables you to work around or remove unsupported features in order to make the PowerBuilder application suitable for Appeon Web migration.	Gives you an idea of which windows to tackle for performance tuning and what PowerBuilder features to thin out. The general principle is: the lighter the window is, the better the window will perform in the Web application.
What's reported	Unsupported PowerBuilder coding features.	Complex windows that most affect the performance of the deployed Web application.
How to generate the report	Use the Unsupported Features Analysis tool, or perform the first two tasks in Appeon Deployment Wizard.	Use the Unsupported Features Analysis tool, or perform the first two tasks in Appeon Deployment Wizard. Requirement: the *Generate Heavy Window Report* option must be enabled in the Appeon Deployment Wizard.
Features	These features apply to both reports: Automatic incremental analysis. After the first analysis, the subsequent analysis is automatically incremental to save time, but the report generated still contains all the information as if it were a full analysis. Analysis at application, PBL or object level. You can choose to generate UFA Report or Heavy Window report for an application, PBL(s) in the application, or object(s) in the application.	

This chapter describes:

- How to use the UFA tool to analyze an application and generate the UFA Report.
- How to use the UFA tool to work around unsupported features effectively.
- How to use the UFA tool to manipulate the UFA Report. For example, you can adjust the view of the UFA Report so it suits your preferences.
- Undetected unsupported features. There are a small number of unsupported features that the UFA Report cannot detect.
- The usage of Heavy Window Report

5.2 Analyzing an application

5.2.1 Tasks required before you perform feature analysis

Step 1 – Make sure the computer is powerful enough to support feature analysis.

Unsupported Features Analysis is CPU and memory intensive. A faster CPU can speed up the process. Additional memory is useful for ensuring that RAM will not be exhausted when analyzing a large application. Regarding system requirements for the Developer machine, refer to the *System Requirements* section in the *Appeon Installation Guide*.

Step 2 – Close any unnecessary running programs or Windows services in order to have the maximum amount of memory available for Features Analysis.

Step 3 – Perform an incremental build or object regeneration for the PowerBuilder application. Right-click on the application target and select a build option (for example, "Incremental Build") as shown in Figure 5-1.

Figure 5-1: Incremental Build

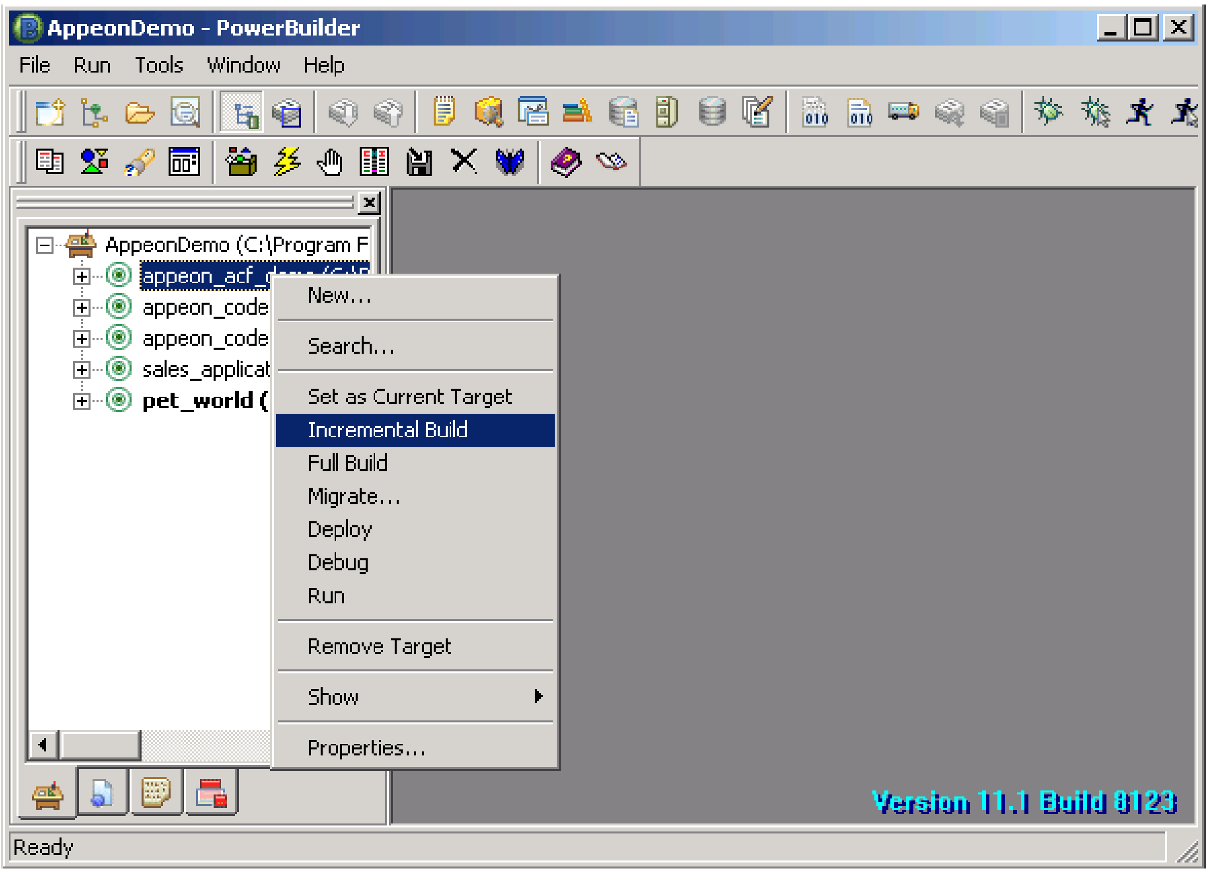


Table 5-2 shows you which type of build options you should select for the application.

Table 5-2: Recommended build options

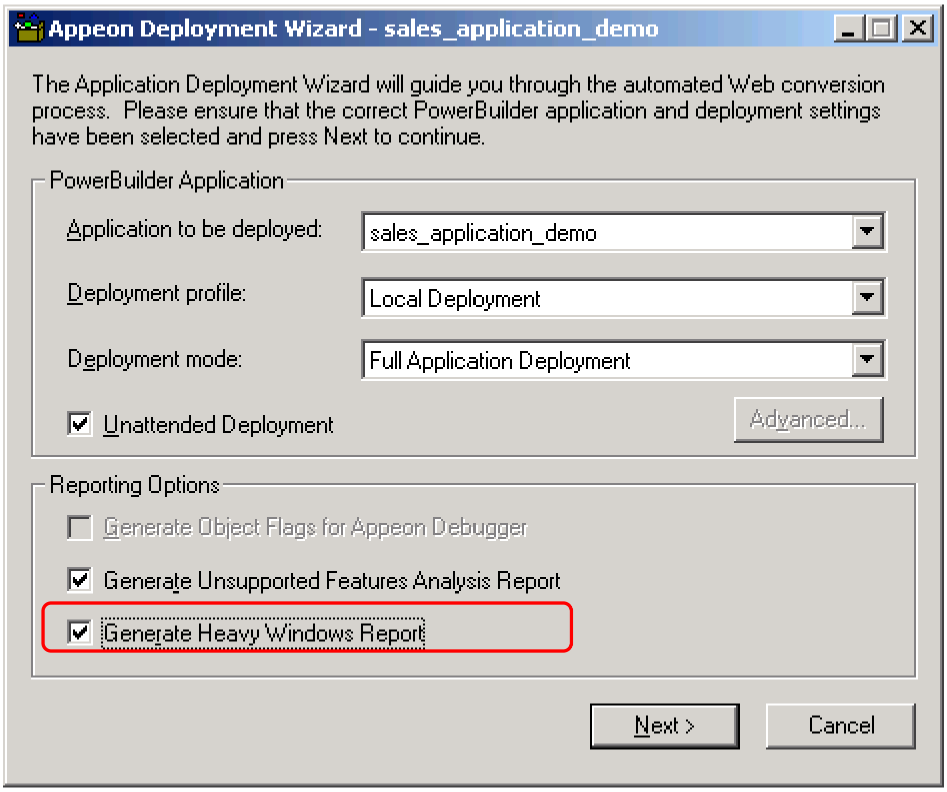
Build Option	What It Does	Recommended For
Regenerate	Refreshes the timestamp for a PowerBuilder object.	Object-level Features Analysis. If the object is regenerated, the object will be analyzed even if no change has made to it.
Incremental	Refreshes the objects that have been changed.	Application-level Features Analysis.
Full	Refreshes all objects.	Not recommended.

Step 4 – Perform PBL optimization for the PowerBuilder application.

The purpose of PBL optimization is to remove gaps from libraries and defrag the storage of objects to get rid of potential problems in PBL files.

Step 5 – (Required only if you want to generate a Heavy Window Report) Check the *Generate Heavy Window Report* option in the Application Deployment Wizard, as shown in Figure 5-2.

Figure 5-2: Enable Heavy Window Report



Appeon Deployment Wizard - sales_application_demo

The Application Deployment Wizard will guide you through the automated Web conversion process. Please ensure that the correct PowerBuilder application and deployment settings have been selected and press Next to continue.

PowerBuilder Application

Application to be deployed: sales_application_demo

Deployment profile: Local Deployment

Deployment mode: Full Application Deployment

☑ Unattended Deployment

Reporting Options

□ Generate Object Flags for Appeon Debugger

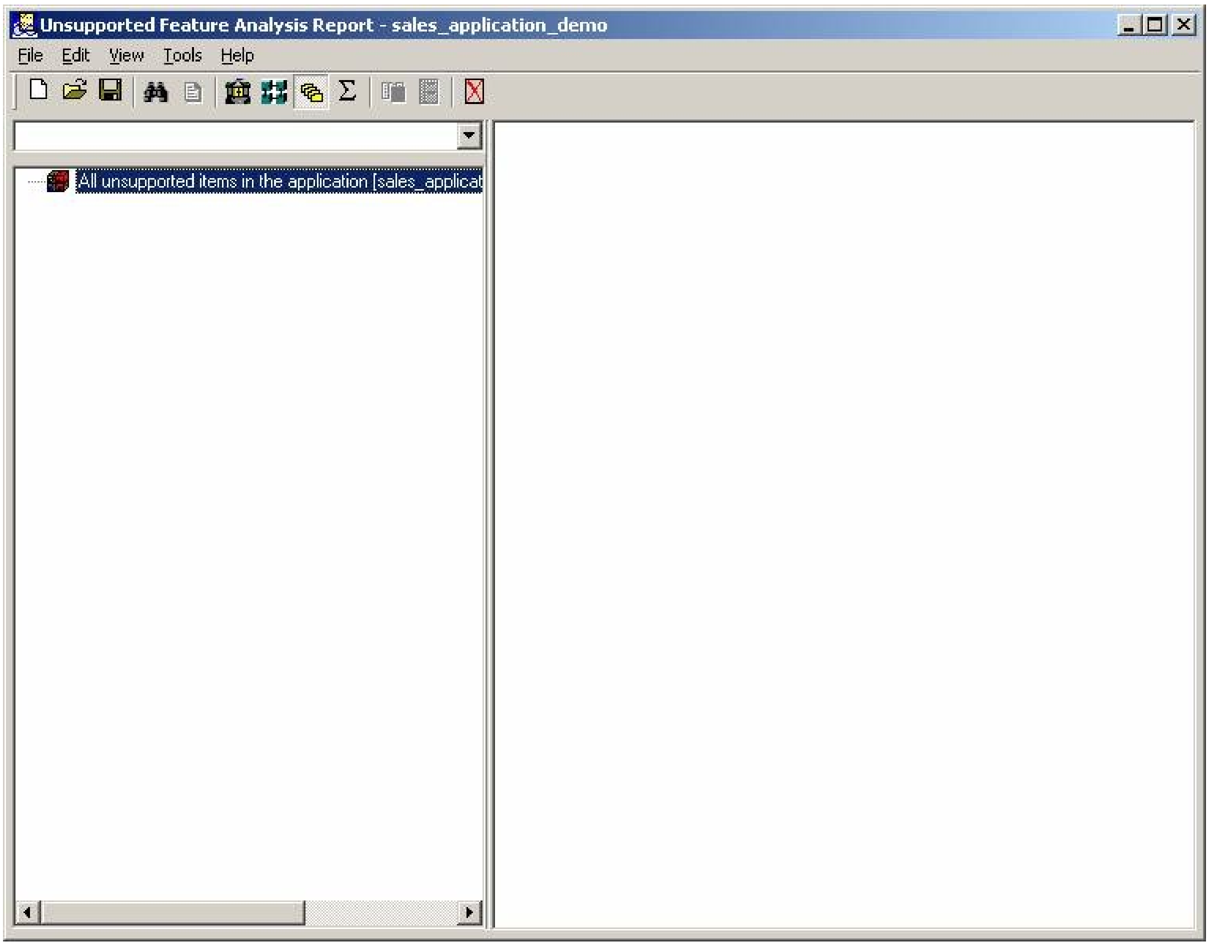
☑ Generate Unsupported Features Analysis Report

☑ Generate Heavy Windows Report

5.2.2 Accessing the UFA tool

Click the *Analyze* button () in the Appeon Developer toolbar to launch the UFA tool. The Unsupported Feature Analysis Report window (UFA Report window) will be displayed.

Figure 5-3: UFA Report Window


Unsupported Feature Analysis Report - sales_application_demo
File
Edit
View
Tools
Help
All unsupported items in the application [sales_applicat

UFA Report window provides a variety of menus to help you manage unsupported features. Table 5-3 gives a brief description of these menus. Some of them are also listed as shortcut menus on the toolbar.

Table 5-3: UFA Report Window

Menu	Description
File	Generates a UFA report. See section 5.2.3. Opens or saves a UFA report. See Section 5.3.2.a.
Edit	Opens and modifies the unsupported source code in the PowerBuilder painter. See Section 5.3.1. Searches for the unsupported features. See Section 5.3.2.c. Filters the unsupported features according to objects, unsupported feature types or priorities. See Section 5.3.2.d. Expands or collapses treeview. Displays the report in the specified level. See Section 5.3.2.e.
View	Selects a display mode of the objects in the Unsupported Feature List treeview. See Section 5.3.2.b.

Tools	Accesses the workaround for the selected unsupported feature. Defines the priority settings of UFA Report items. See Section 5.3.2.f. Customizes the general settings of the UFA Report. See Section 5.3.2.g.
Help	Provides the version number of the UFA tool.

5.2.3 Performing feature analysis

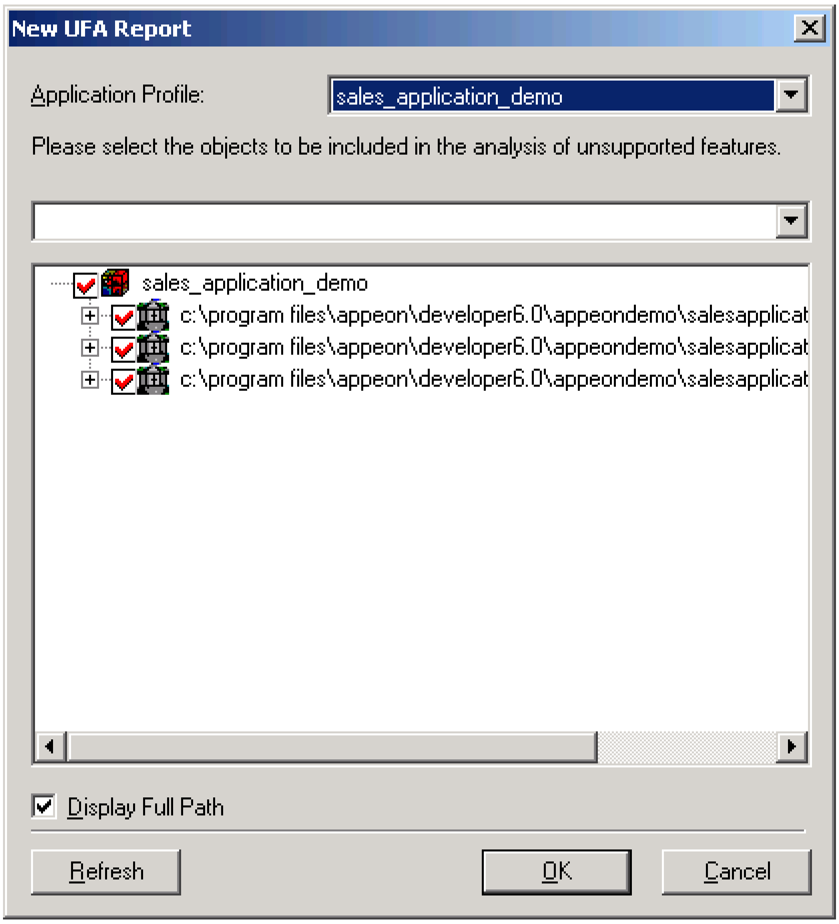
There are two ways you can perform a feature analysis of an application:

- Use the Unsupported Features Analysis tool. This option enables you to do a feature analysis of the whole application, or PBL(s) in the application, or objects in the application.
- Use the first two tasks in Appeon Deployment Wizard. When you use Appeon Deployment Wizard to deploy an application, the first task the wizard performs is a feature analysis of the whole application.

This section mainly gives instructions on the first option (using the Unsupported Features Analysis tool). For more information about the second option, refer to Section 6.3: *Deployment process*.

Step 1 – Select the File | New Report menu in the UFA Report window. The New UFA Report dialog box will appear, as shown in Figure 5-4.

Figure 5-4: New UFA Report dialog box



New UFA Report

Application Profile: sales_application_demo

Please select the objects to be included in the analysis of unsupported features.

☑ sales_application_demo
- ☑ c:\program files\appeon\developer6.0\appeondemo\salesapplicat
- ☑ c:\program files\appeon\developer6.0\appeondemo\salesapplicat
- ☑ c:\program files\appeon\developer6.0\appeondemo\salesapplicat

☑ Display Full Path

Table 5-4 gives a brief description of the elements in the New UFA Report window.

Table 5-4: New UFA Report window

Element	Description
Application profile dropdown list	Provides a list of application profiles for you to select and analyze. Only application profiles configured in the Appeon Developer Configuration window will be listed.
Search field	Searches for PBLs or objects that contain the text you enter.
Treeview of objects	Gives the treeviews of PBLs and objects for the application selected in the *Application* dropdown list and allows you to select the whole application or only some objects to analyze.
Display Full Path option	Gives you options to display or hide the full path of the PBLs and objects in the treeview.
Refresh button	Refreshes the PBL and object list in the treeview.

Step 2 – Select the application that you want to analyze from the *Application profile* dropdown list.

The default target to analyze is the default application profile in the Appeon Developer Configuration window.

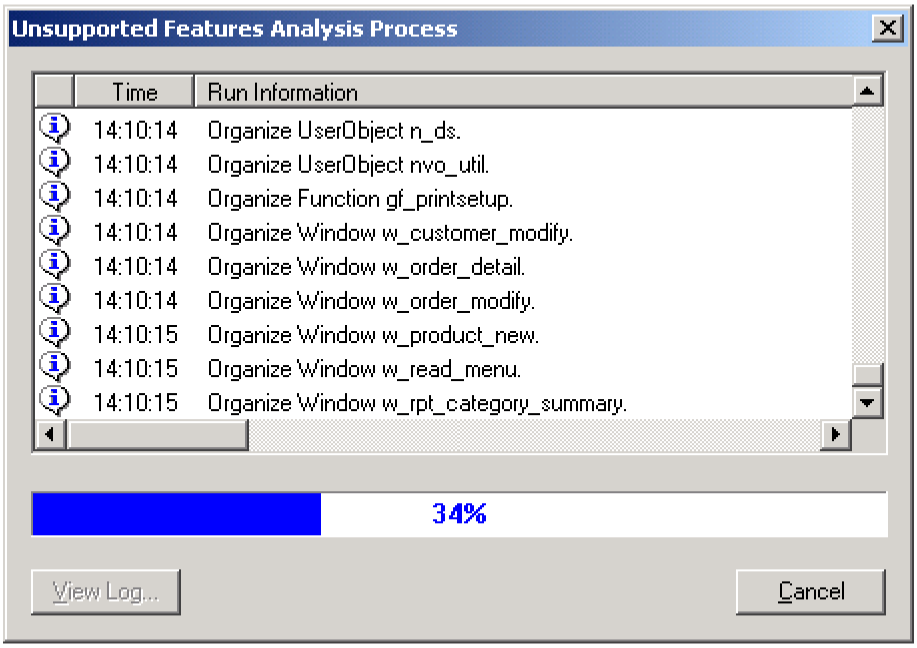
Step 3 – Check the PBLs and/or objects in the treeview for which you want to generate the UFA Report.

The treeview lists all the PBLs and objects contained in the selected target. By default, all objects are selected.

Step 4 – Click *OK* to start the feature analysis.

The feature analysis of the selected objects begins, as shown in Figure 5-5.

Figure 5-5: Unsupported Features Analysis Process dialog



Unsupported Features Analysis Process

	Time	Run Information
ⓘ	14:10:14	Organize UserObject n_ds.
ⓘ	14:10:14	Organize UserObject nvo_util.
ⓘ	14:10:14	Organize Function gf_printsetup.
ⓘ	14:10:14	Organize Window w_customer_modify.
ⓘ	14:10:14	Organize Window w_order_detail.
ⓘ	14:10:14	Organize Window w_order_modify.
ⓘ	14:10:15	Organize Window w_product_new.
ⓘ	14:10:15	Organize Window w_read_menu.
ⓘ	14:10:15	Organize Window w_rpt_category_summary.

Step 5 – Click *Close* when the analysis is completed. The UFA Report is loaded into the UFA Report Window.

5.3 Working with UFA Report

This section describes:

- How to modify unsupported features. You can use the UFA Report to effectively remove unsupported features in the application.
- How to manipulate the UFA Report in the UFA Report Window. You can adjust the view of the UFA Report so it suits your preferences.
- Undetected unsupported features. There are a small number of unsupported features that the UFA Report cannot detect.

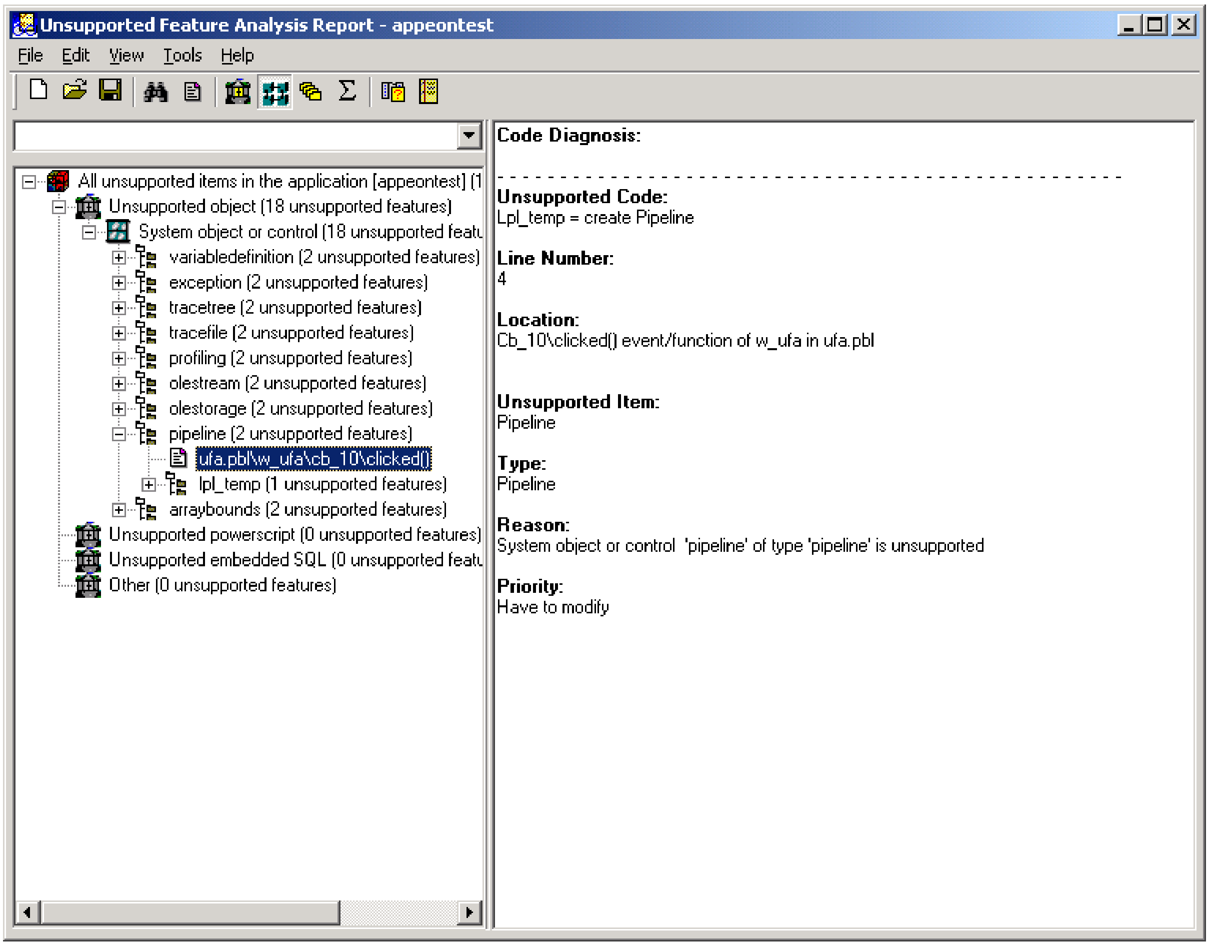
5.3.1 Modifying unsupported features

When the UFA Report is loaded, you can view all the unsupported features in the left treeview and modify them one by one.

Step 1 – Expand the unsupported feature list treeview and select the unsupported item.

The detailed analysis of the selected item will be displayed on the right of the window, as shown in Figure 5-6.

Figure 5-6: Unsupported feature details



All unsupported items in the application [appeontest] (1
Unsupported object (18 unsupported features)
System object or control (18 unsupported featu
variabledefinition (2 unsupported features)
exception (2 unsupported features)
tracetree (2 unsupported features)
tracefile (2 unsupported features)
profiling (2 unsupported features)
olestream (2 unsupported features)
olestorage (2 unsupported features)
pipeline (2 unsupported features)
ufa.pbl\w_ufa\cb_10\clicked()
lpl_temp (1 unsupported features)
arraybounds (2 unsupported features)
Unsupported powerscript (0 unsupported features)
Unsupported embedded SQL (0 unsupported featu
Other (0 unsupported features)

Code Diagnosis:

- -

Unsupported Code:
Lpl_temp = create Pipeline

Line Number:
4

Location:
Cb_10\clicked() event/function of w_ufa in ufa.pbl

Unsupported Item:
Pipeline

Type:
Pipeline

Reason:
System object or control 'pipeline' of type 'pipeline' is unsupported

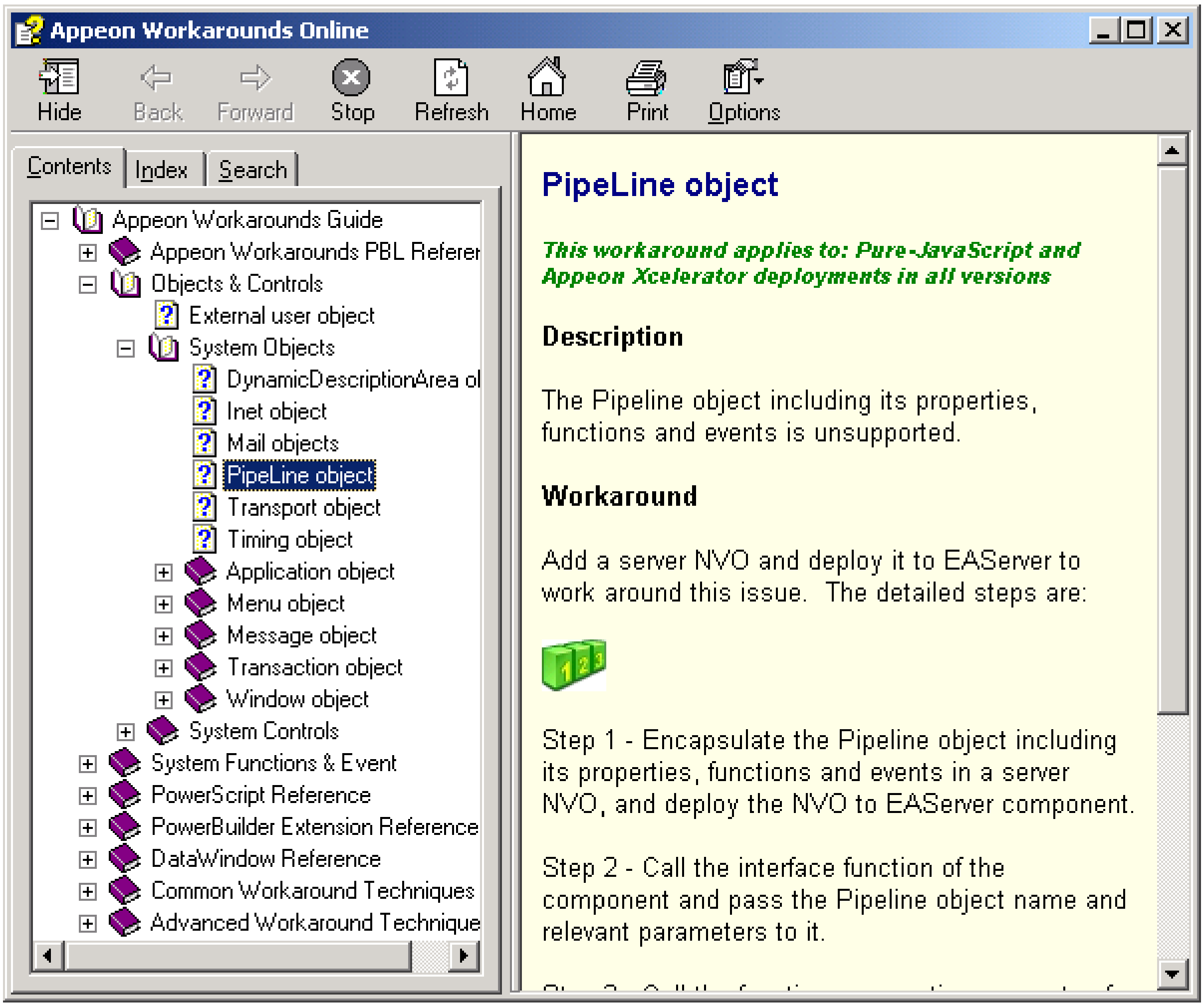
Priority:
Have to modify

The right text box provides a detailed diagnosis on the unsupported code in seven aspects: the Unsupported Code, Line Number, Location, Unsupported Item, Type, Reason, and Priority.

Step 2 – View the workaround for the unsupported item by right-clicking the unsupported item in the unsupported feature list treeview and selecting *Workarounds solutions*.

- Go To Workarounds Online – Opens the online *Appeon Workarounds Guide* at URL: http://www.appeon.com/support/documents/workarounds/6.1/.
- Go To Local Workarounds – Opens the local *Appeon Workarounds Guide* installed with *Appeon Help*.

Figure 5-7: Local Workarounds Guide



PipeLine object

This workaround applies to: Pure-JavaScript and Appeon Xcelerator deployments in all versions

Description

The Pipeline object including its properties, functions and events is unsupported.

Workaround

Add a server NVO and deploy it to EAServer to work around this issue. The detailed steps are:

Step 1 - Encapsulate the Pipeline object including its properties, functions and events in a server NVO, and deploy the NVO to EAServer component.

Step 2 - Call the interface function of the component and pass the Pipeline object name and relevant parameters to it.

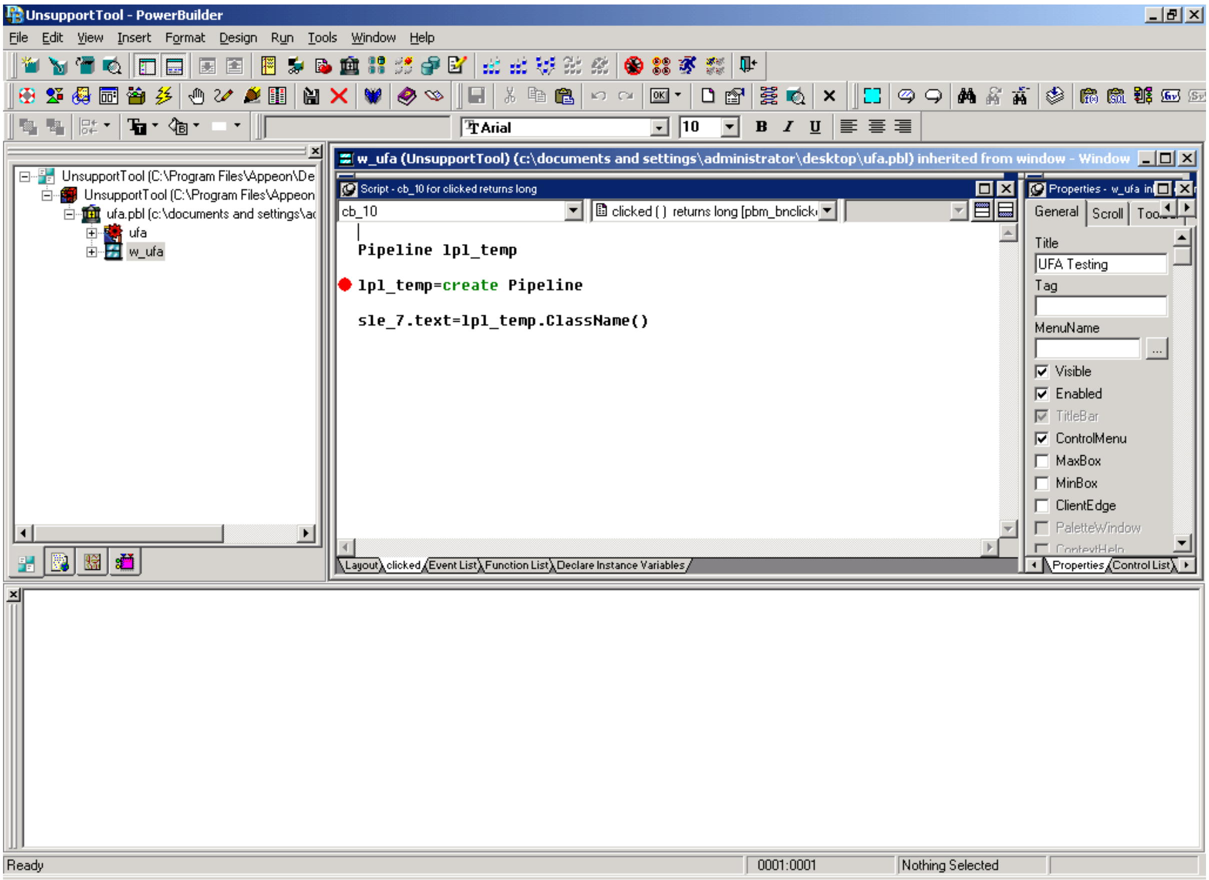
Step 3 – Modify the unsupported source code in the PowerBuilder painter according to the solution provided in the *Appeon Workarounds Guide*.

Right click the unsupported item in the unsupported feature list treeview, and select *Edit* or *Edit Source* to open the source code in the PowerBuilder Script view.

- Edit – Displays the source code of an event or function, which contains the unsupported feature. This button is available only if the unsupported code resides in an event or function and it is not DataWindow-related.
- Edit Source – Displays the source code of the object, which contains the unsupported feature. This button is always available.

The PowerBuilder Script view automatically opens, as shown in Figure 5-8.

Figure 5-8: Unsupported code in the PowerBuilder painter



UnsupportTool - PowerBuilder

w_ufa (UnsupportTool) (c:\documents and settings\administrator\desktop\ufa.pbl) inherited from window - Window

Script - cb_10 for clicked returns long

cb_10 | clicked ( ) returns long [pbm_bnclicked]

```
Pipeline lpl_temp
lpl_temp=create Pipeline
sle_7.text=lpl_temp.ClassName()
```

Properties - w_ufa

Title: UFA Testing

Tag:

MenuName:

☑ Visible
☑ Enabled
☑ TitleBar
☑ ControlMenu
☐ MaxBox
☐ MinBox
☐ ClientEdge
☐ PaletteWindow
☐ ContextHelp

Ready | 0001:0001 | Nothing Selected

File
Edit
View
Insert
Format
Design
Run
Tools
Window
Help
UnsupportTool (C:\Program Files\Appeon\De
UnsupportTool (C:\Program Files\Appeon
ufa.pbl (c:\documents and settings\ac
ufa
w_ufa
Arial
10
Layout
clicked
Event List
Function List
Declare Instance Variables
General
Scroll
Too
Properties
Control List

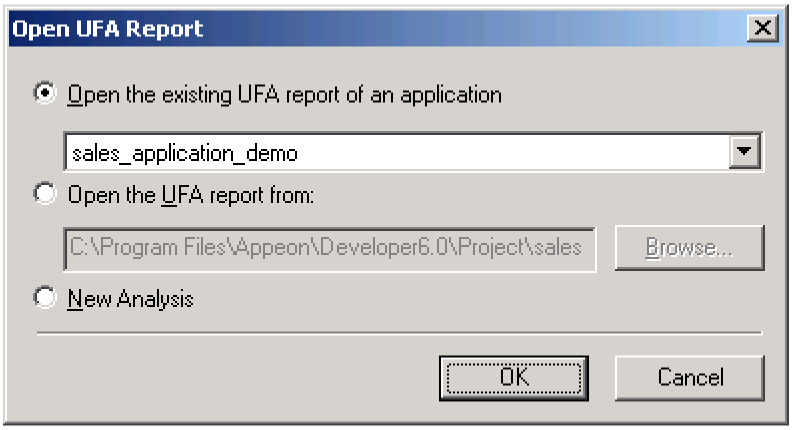
5.3.2 Manipulating the UFA Report

5.3.2.a Opening or saving a UFA Report

You can open or save a UFA Report using the File menu in the UFA Report Window.

- To open an existing UFA Report, select the File | Open menu. The Open UFA Report dialog box is displayed. If there is no UFA report available, select *New Analysis* and click *OK* to analyze an application.

Figure 5-9: Open UFA Report



Open UFA Report

● Open the existing UFA report of an application

sales_application_demo

○ Open the UFA report from:

C:\Program Files\Appeon\Developer6.0\Project\sales

○ New Analysis

- To save a UFA Report, select the File | Save menu.
- To save a UFA Report with a new name, select the File | Save As menu.

5.3.2.b Selecting report view mode

You can view the UFA Report in four different modes. Go to the View menu in the UFA Report Window and select the desired mode:

- Category – Enables you to view the unsupported features in the following categories: Unsupported Objects, Unsupported PowerScript, Unsupported Embedded SQL and Others.
- Hierarchy – Enables you to view the unsupported features in hierarchical order: PBL, Object, Control, Method (Event/Function).
- Priority – Enables you to view the unsupported features in priorities (you can go to the Tools | Define Priority menu to define the priorities of the PowerBuilder features): Have to modify, Suggest to modify, Can be ignored.
- Summary – Enables you to view summary information of each unsupported feature.

You can set the default view mode in the Tools | Options menu. For detailed descriptions, refer to Section 5.3.2.g.

5.3.2.c Searching for UFA Report items

To do a quick search, you can directly enter the text that you want to search in the search field of the UFA Report window. Or follow steps below to use the standard Find Text dialog box:

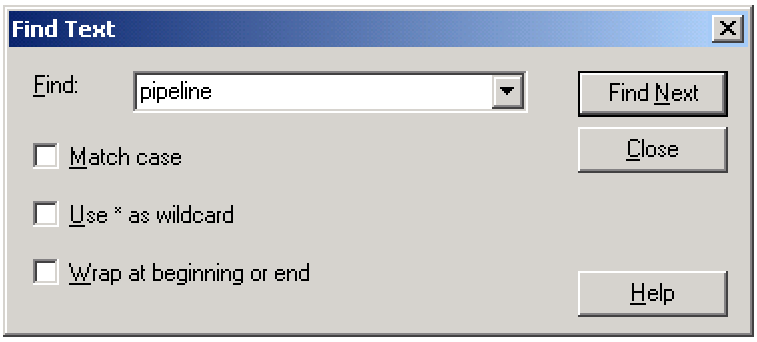
Step 1 – Select the Edit | Find menu. Or right click in the unsupported feature list treeview and select *Find*.

Step 2 – In the Find Text box, enter the text that you want to search.

Step 3 – Select any other options that you want.

Step 4 – Click *Find Next*.

Figure 5-10: Find Text



Find: pipeline

□ Match case

□ Use * as wildcard

□ Wrap at beginning or end

5.3.2.d Filtering UFA Report items

Filtering helps you restrict the number of the unsupported features displayed in the UFA Report and quickly find the target unsupported features, especially when there are a large number of unsupported features in the application.

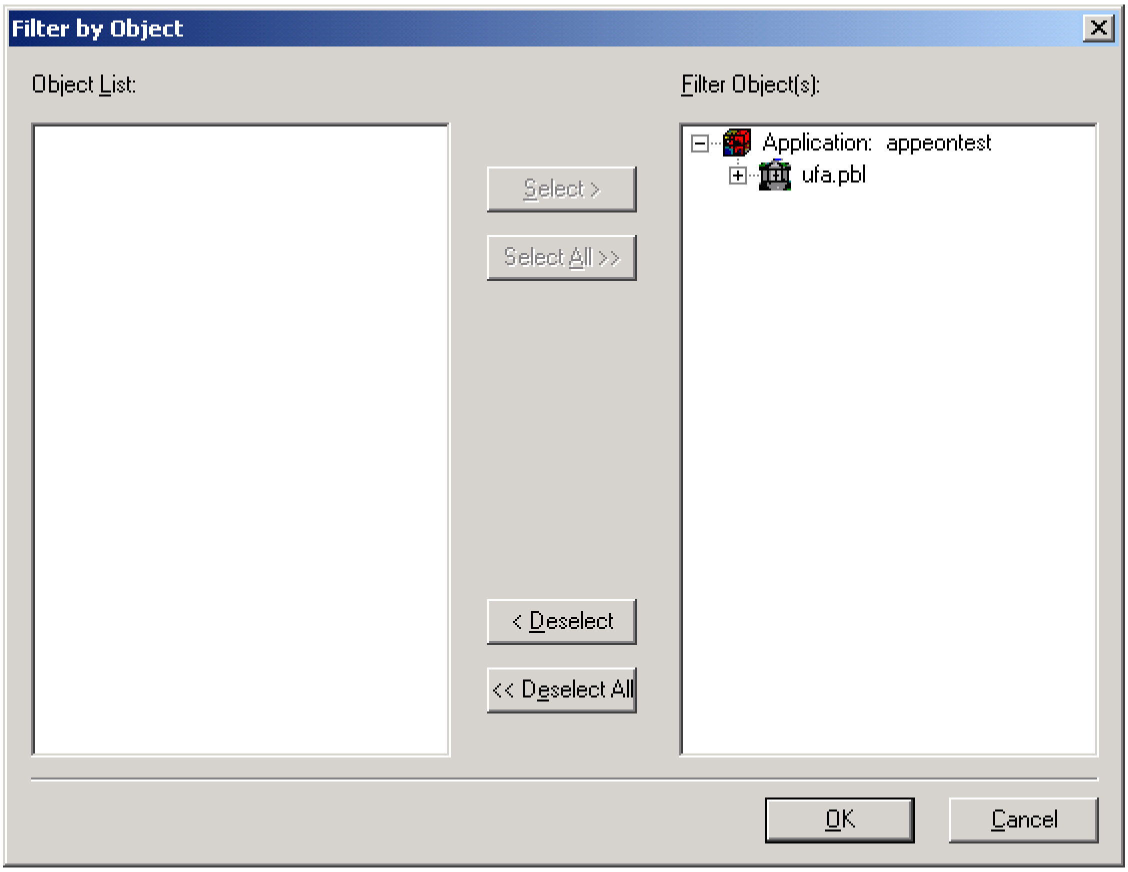
You can filter the unsupported features by the objects they reside in, or by the feature types, or by the priorities they are specified to.

Filtering by objects

Step 1 – Select the Edit | Filter by Object menu. Or right click in the unsupported feature list treeview and select *Filter by Object*.

The Filter by Object dialog box is displayed, as shown in Figure 5-11.

Figure 5-11: Filter by object



Step 2 – Make sure the unsupported objects you want to display in the report are on the right listbox, whereas the unsupported objects you want to hide are on the left listbox. Click *OK*.

To quickly select all objects of the same type, right click the list and select an object type.

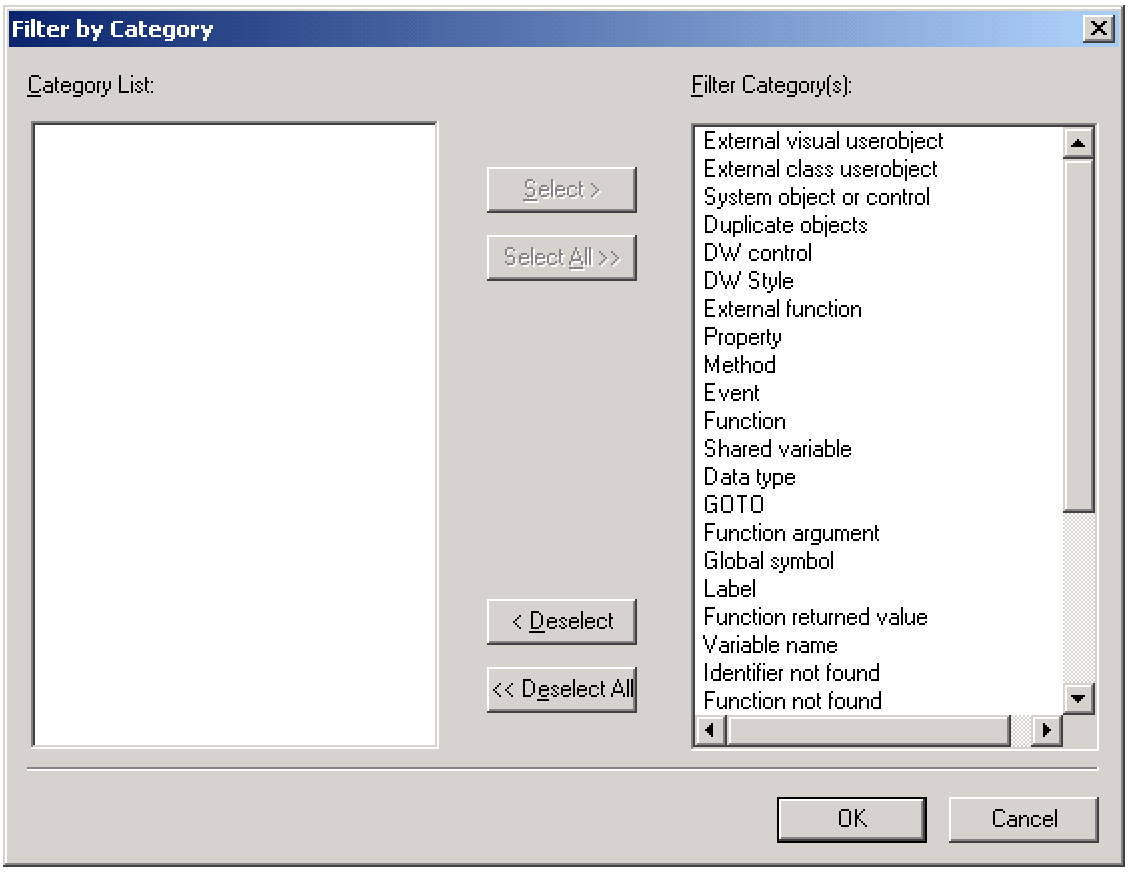
The UFA Report hides the unsupported objects that are on the left listbox in the Filter by Object window.

Filtering by categories

Step 1 – Select the Edit | Filter by Category menu. Or right click in the unsupported feature list treeview and select *Filter by Category*.

The Filter by Category dialog box is displayed, as shown in Figure 5-12.

Figure 5-12: Filter by Category



External visual userobject
External class userobject
System object or control
Duplicate objects
DW control
DW Style
External function
Property
Method
Event
Function
Shared variable
Data type
GOTO
Function argument
Global symbol
Label
Function returned value
Variable name
Identifier not found
Function not found

Step 2 – Make sure the categories of the unsupported features you want to display in the report are on the right listbox, whereas the categories of the unsupported features you want to hide are on the left listbox. Click *OK*.

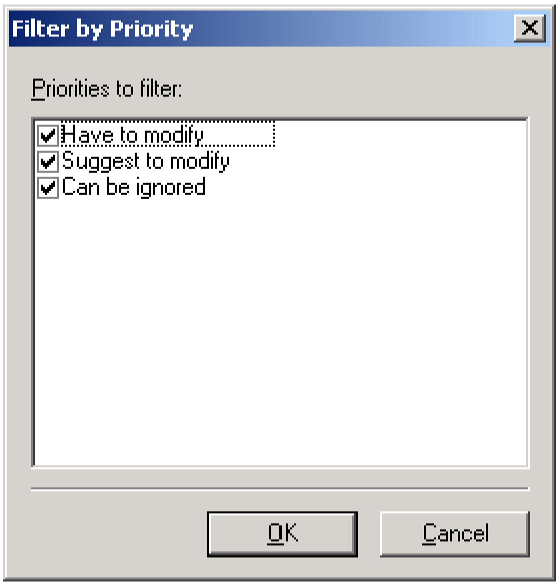
The UFA Report hides unsupported features that belong to the categories that are on the left in the Filter by Category window.

Filtering by priorities

Step 1 – Select the Edit | Filter by Priority menu. Or right click in the unsupported feature list treeview and select *Filter by Priority*.

The Filter by Priority dialog box is displayed, as shown in Figure 5-13.

Figure 5-13: Filter by Priority



Step 2 – Uncheck the priorities of the unsupported features which you want to hide. For example, if you want to hide the features that "can be ignored", uncheck the *Can be ignored* option.

Step 3 – Click *OK*.

The UFA Report only displays unsupported features of the specified priorities.

5.3.2.e Specifying report display level

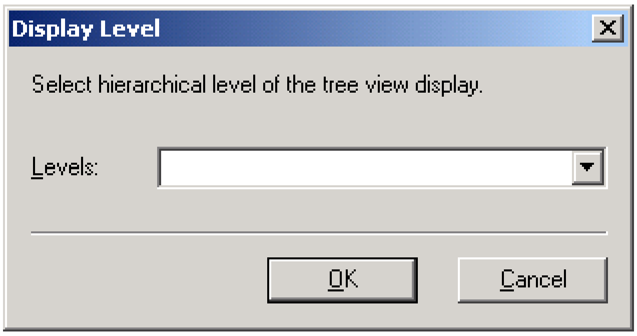
You can quickly expand or collapse a node of the unsupported feature list treeview by selecting Expand Sub-node or Collapse Sub-node from the Edit menu or from the popup menu on any node.

You can also specify the display level using the following steps:

Step 1 – Select the Edit | Display Level menu.

Step 2 – In the Display Level box, enter the number of the level at which the unsupported feature list treeview will display.

Figure 5-14: Display Level



Step 3 – Click *OK*.

The UFA Report only displays unsupported features of the specified level.

5.3.2.f Defining the priority settings of unsupported features

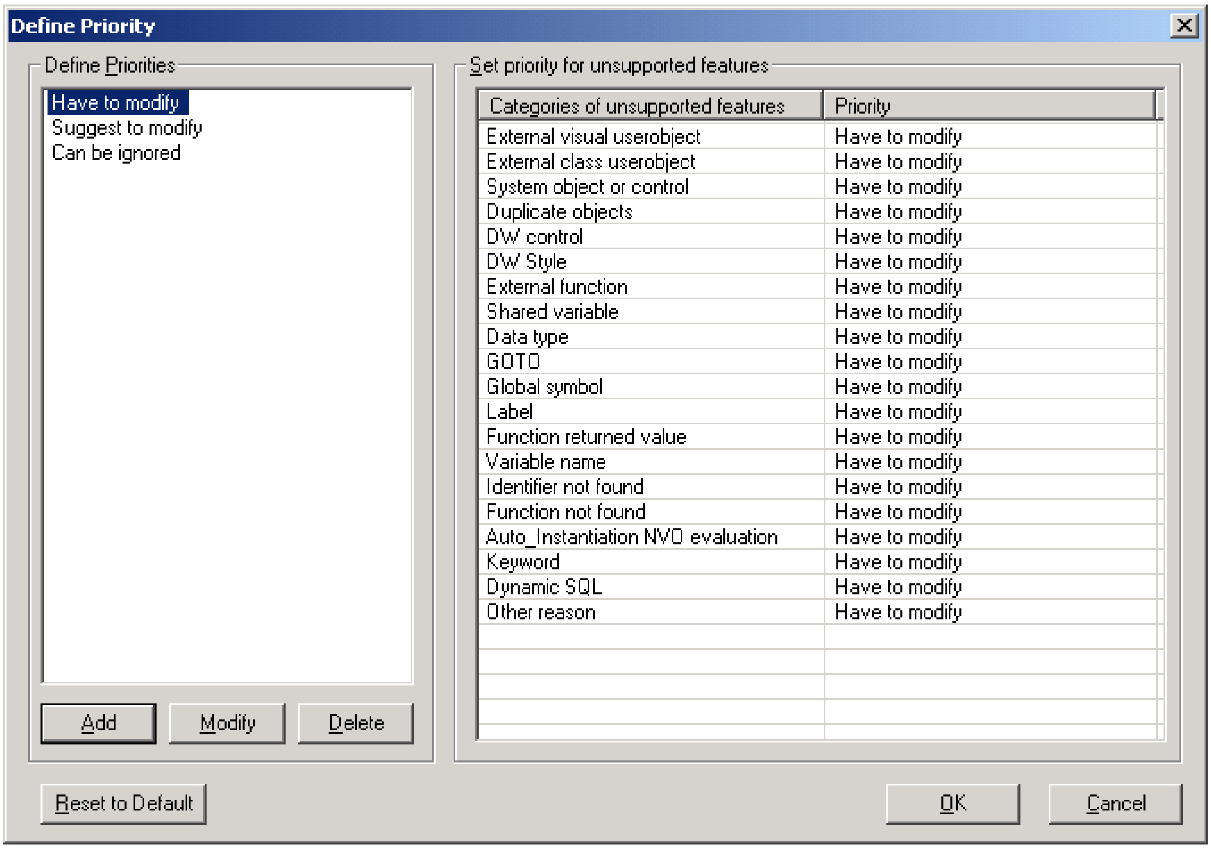
By default, Appeon divides all unsupported features into three priority levels:

- Have to modify – The features flagged with “Have to modify” are important in the application and will cause significant functionality loss if they are not modified or worked around.
- Suggest to modify – The features flagged with “Suggest to modify” will not necessarily cause functionality loss if they are not modified or worked around. You can decide whether to modify them according to their functionalities in the application.
- Can be ignored – The features flagged with “Can be ignored” are small or trivial features which will not compromise application functionality even if they are not modified or worked around.

To customize or change priority levels for unsupported features, take the following steps:

Step 1 – Select the Tools | Define Priority menu. The Define Priority window is displayed, as shown in Figure 5-15.

Figure 5-15: Define Priority dialog box



Categories of unsupported features	Priority
External visual userobject	Have to modify
External class userobject	Have to modify
System object or control	Have to modify
Duplicate objects	Have to modify
DW control	Have to modify
DW Style	Have to modify
External function	Have to modify
Shared variable	Have to modify
Data type	Have to modify
GOTO	Have to modify
Global symbol	Have to modify
Label	Have to modify
Function returned value	Have to modify
Variable name	Have to modify
Identifier not found	Have to modify
Function not found	Have to modify
Auto_Instantiation NVO evaluation	Have to modify
Keyword	Have to modify
Dynamic SQL	Have to modify
Other reason	Have to modify

Step 2 – Add, modify or delete the system default priorities in the left list box.

Step 3 – Change priority levels of unsupported feature types by selecting a priority level in the *Priority* column on the right box.

Step 4 – Click *Save* to save the priority settings.

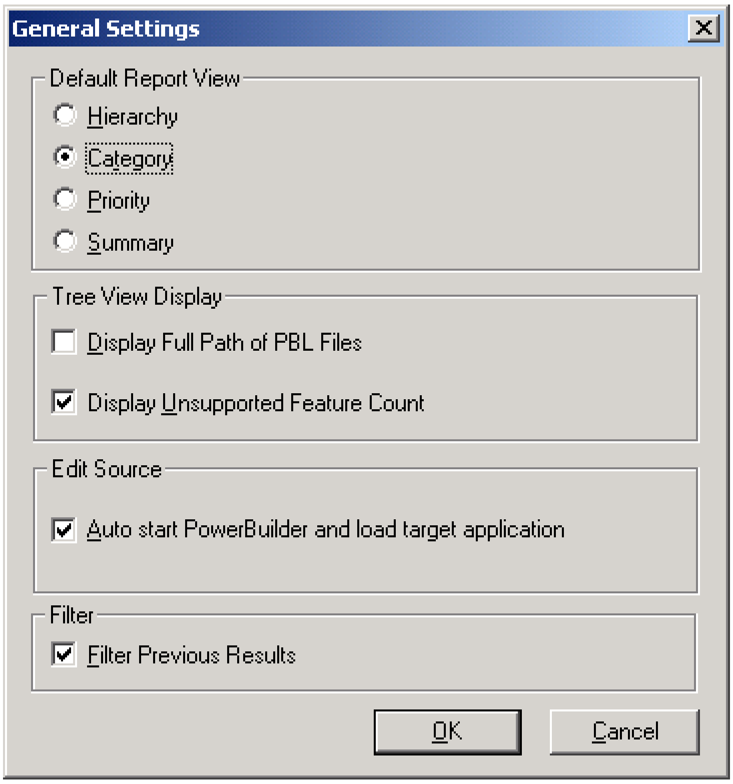
Or click *Reset to Default* to restore the system default priority settings.

5.3.2.g Customizing the general settings of the UFA Report

You can define your preference settings of the UFA Report window. These settings include default report view mode, display formats of PBLs, feature count, unsupported source code edition, and filtering. The new settings will be saved as the default settings.

Select the Tools | Options menu in the UFA Report Window. The General Settings window is displayed, as shown in Figure 5-16.

Figure 5-16: General Settings dialog box



General Settings

Default Report View
○ Hierarchy
● Category
○ Priority
○ Summary

Tree View Display
□ Display Full Path of PBL Files
☑ Display Unsupported Feature Count

Edit Source
☑ Auto start PowerBuilder and load target application

Filter
☑ Filter Previous Results

- *Default Report View* group box – Select the default view mode of the UFA Report. For detailed description of each view mode, refer to Section 5.3.2.b: Selecting report view mode.
- *Display Full Path of PBL Files* – Displays the full path and the file name of the PBLs in the UFA Report.
- *Display Unsupported Feature Count* – Displays the total number of the unsupported features in the UFA Report.
- *Auto start PowerBuilder and load target application* – Enables PowerBuilder to automatically start and load the target application when you click the *Edit* or *Edit Source* button in the UFA Report window.
- *Filter Previous Results* – Allows you to filter only the results of a previous filter. It narrows a filter that resulted in too much data the first time. If you want to search all data of the report, leave this option as unselected.

5.3.3 Undetected unsupported features

Appeon Unsupported Features Analysis is capable of highlighting the majority of unsupported features contained in the PowerBuilder application. There are still a small number of features that Unsupported Features Analysis will not detect, so they will not be listed in the UFA Report.

An application containing unsupported features can still be converted to the Web and may work correctly, depending on the type and number of unsupported features. If the Web application does not work correctly and the UFA Report does not indicate any unsupported features that are likely to cause such problems, the problems may be caused by unsupported features that have not been detected. In this situation, refer to *Undetected Features* in *Appeon Features Help*, which documents the types of unsupported features that may not be detected.

5.4 Working with Heavy Window Report

You can access the Heavy Window Report from the Information Manager if the *Generate Heavy Window Report* option is enabled in the Appeon Deployment Wizard before the feature analysis or the application deployment is performed.

5.4.1 How a heavy window is measured

Two factors determine the heaviness of a window:

- The number of controls in the window.
- The complexity of the controls in the window (for example, a DataWindow with several DropDownDataWindows makes the window significantly heavier than a DataWindow without DropDownDataWindows).

Appeon assigns different weights to different types of controls. (Example: the weight of a DataWindow is 1.0, the weight of a computed field is 0.06, and the weight of a tab page is 0.15.) The weight of a window is determined by summing up the total weight of all the controls contained within it.

The time required to open a window increases proportionally with the weight of the window. As our tests show, it takes 0.5 of a second to open a window that weighs 8.0 using a computer with the following specifications: P4 1.8 GHz 512 K cache CPU and 256 MB RAM. It takes approximately 0.6 second to open a window that weighs 10.0. In different environments the time it takes to open a window can vary. Note that the elapsed time does not include the time used to connect to the database and retrieve data.

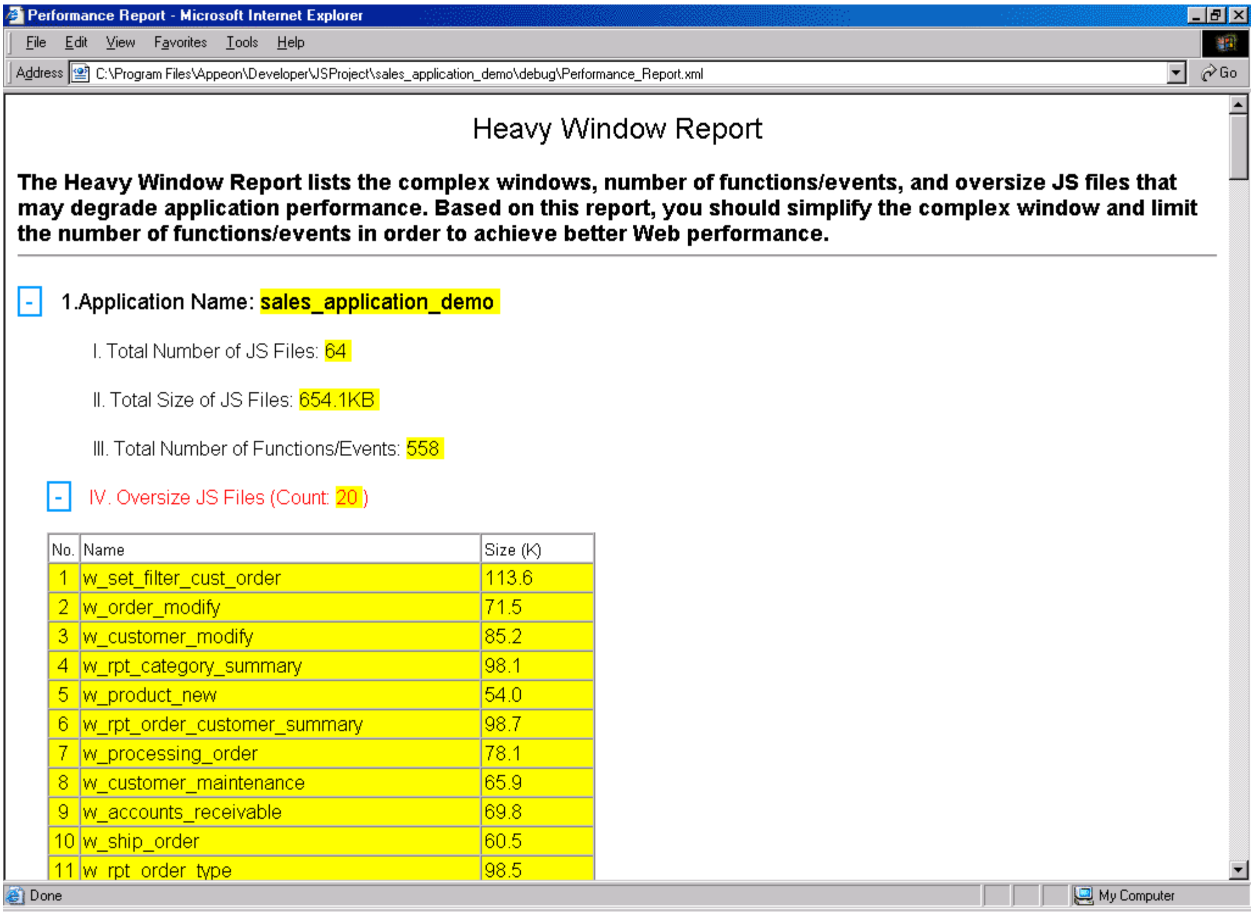
By default, the Heavy Window Report flags all windows that weigh 8.0 (default minimum load value) or more.

The weight of the different control types, as well as the default minimum load value, are all stored in the PerformanceStatConfig.xml file in the %Appeon%\Developer6.1 path. You can change the settings in the XML file based on actual requirements. (Example: change the minimum load value to 6.0 to get a report of all the windows that take about 0.4 second or more to open). However, Appeon strongly recommends that you do not change the default value of each type of control because each default value is a fixed value based on multiple testing of each control's complexity. Most controls will significantly increase the loading time of a window as their number increases. If you change the default value of the controls, the minimum load value may not compute correctly. For example, the minimum load value adds up to 10.0, but the actual loading time may be much more or less than 0.6 of a second.

5.4.2 Reading Heavy Window Report

By using the Heavy Window Report, you can grasp a general idea of how many heavy windows are contained in the application. This report lists the oversize JavaScript files generated for the application and the heavy windows, including the number of complex controls and the minimum load value of each window, as shown in Figure 5-17. This helps you measure each window's performance.

Figure 5-17: Heavy Window Report


Performance Report - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Address C:\Program Files\Appeon\Developer\JSProject\sales_application_demo\debug\Performance_Report.xml
Go
Done
My Computer

Heavy Window Report

The Heavy Window Report lists the complex windows, number of functions/events, and oversize JS files that may degrade application performance. Based on this report, you should simplify the complex window and limit the number of functions/events in order to achieve better Web performance.

- 1.Application Name: sales_application_demo

I. Total Number of JS Files: 64

II. Total Size of JS Files: 654.1KB

III. Total Number of Functions/Events: 558

- IV. Oversize JS Files (Count: 20)

No.	Name	Size (K)
1	w_set_filter_cust_order	113.6
2	w_order_modify	71.5
3	w_customer_modify	85.2
4	w_rpt_category_summary	98.1
5	w_product_new	54.0
6	w_rpt_order_customer_summary	98.7
7	w_processing_order	78.1
8	w_customer_maintenance	65.9
9	w_accounts_receivable	69.8
10	w_ship_order	60.5
11	w_rpt_order_type	98.5

5.4.3 Making heavy windows lighter

For instructions on how to make heavy windows lighter, refer to the *Appeon Performance Tuning Guide*.

6 Deploying PowerBuilder Applications

6.1 Overview

Appeon Deployment Wizard deploys PowerBuilder applications to Web applications. The deployment should take place only after the application has undergone careful analysis and is in compliance with the *Basic and Architectural Requirements* defined in *Appeon Features Help*.

The Appeon Deployment Wizard handles Web deployment in five steps:

1. Exports the PowerBuilder application source code.
2. Analyzes the PowerBuilder application to verify that it does not contain unsupported features.
3. Parses the PowerBuilder source code (PBL files) into Web application files that are stored on the local machine.
4. Uploads the DataWindow syntax to Appeon Server.
5. Transfers local Web files to the Web server by either file copy or FTP.

Notes:

1) Deploying to an SSL Web Server is no different than deploying to a standard Web Server. You only need to make sure that the SSL Web Server is set up correctly and is fully functional. For detailed instructions on configuring a Web server as an SSL Web Server, refer to the corresponding documents provided by the Web server vendor.

2) If you want to re-deploy an application which is now running online:

- You can incrementally deploy the application while keeping the application running online, if non-database related source code and image files are changed.
- You must shut down the running Web application and then full or incrementally deploy the application if the INI files or database-related source code are changed. Otherwise, the application may have runtime problems such as database connection failure or missing tables or fields.

6.2 Deployment performance

Web deployment is a process that can be very CPU and memory intensive, so we recommend that you use a developer computer that conforms to the recommended specifications outlined in the *System Requirements* section of the *Appeon Installation Guide* for a Developer PC to run Appeon Developer.

For more information regarding the deployment performance of different types of applications, refer to the *Appeon Performance Tuning Guide*.

Be sure to close any unnecessary programs and Windows services in order to have an optimal amount of memory available for the deployment process.

6.2.1 Speed of Web Deployment Process

The table below benchmarks the time taken or speed to compile various applications in PowerBuilder and deploy to the Web with Appeon. In general, the results indicate that the speed of the Appeon Web deployment process is similar to the speed of the PowerBuilder compilation/build process.

Table 6-1: Appeon Web Deployment Time Benchmarks

	PB EXE w/ PBDs	PB EXE w/DLLs	Appeon
Sales Application Demo (5.55M)	20seconds	50 seconds	22 seconds
Appeon Code Example Demo (20.5M)	1 minutes 3 seconds	12 minutes 37 seconds	1 minutes 12 seconds
ACF Demo (21.5M)	2 minutes 5 seconds	24 minutes 46 seconds	2 minutes 12 seconds

As a general rule of thumb, each additional megabyte of PBLs will add approximately 8 seconds to the full deployment process. For example, it would take 4 minutes to deploy a 30MB PFC application. This would only apply for the initial or full deployment. All subsequent deployments should take no more than few minutes or even possibly as little as a few seconds.

This rule of thumb applies to hardware configurations that meet the minimum Developer PC requirements stated in the Appeon Installation Guide. Machines with a slower clock-speed will take longer to perform the deployment.

6.2.2 Deployment duration for full deployments

Remember the following main points regarding the deployment time:

- The process is very CPU and memory intensive.
- The size and complexity of the application affect the deployment time.

The following is an example of how long it takes to deploy an application with 3.69 MB in PBLs and images using full deployment:

Table 6-2: Deployment duration

Deployment Task	Elapsed Time
Task 1: Application Source Code Export	8 seconds
Task 2: Web File Generation	27 seconds
Task 3: Web Deployment	15 seconds (for a local deployment)
Test environment: a single CPU Intel P4 1.8 GHz with 256 MB RAM and 60 GB IDE hard drive	

All generated Web files deployed to the server are a combined 1.25 MB in size.

6.2.3 Deployment duration for incremental deployments

Since the initial full deployment can be time-consuming, Appeon provides an Incremental Deployment for maintaining or upgrading an already-deployed application. This feature only re-deploys to the Web the incremental changes that have been made in the application. Appeon can detect whether an object has been modified and re-deploy only the changed objects.

6.3 Deployment process

6.3.1 Preparing the PowerBuilder application

Appeon recommends that you perform a build and PBL optimization on the PowerBuilder application before performing an Appeon Web Application Deployment. Table 6-3 shows you which type of build options you should select for the application.

Table 6-3: Build options

Build Option	What It Does	When To Perform
Incremental build	Refreshes the objects that have been changed.	Before a full or incremental deployment.
Full build	Refreshes all objects.	Not recommended.
Optimize PBLs	Removes unused space in the PBLs as stored on disk, and reduces object fragmentation.	After building the application but prior to deployment.

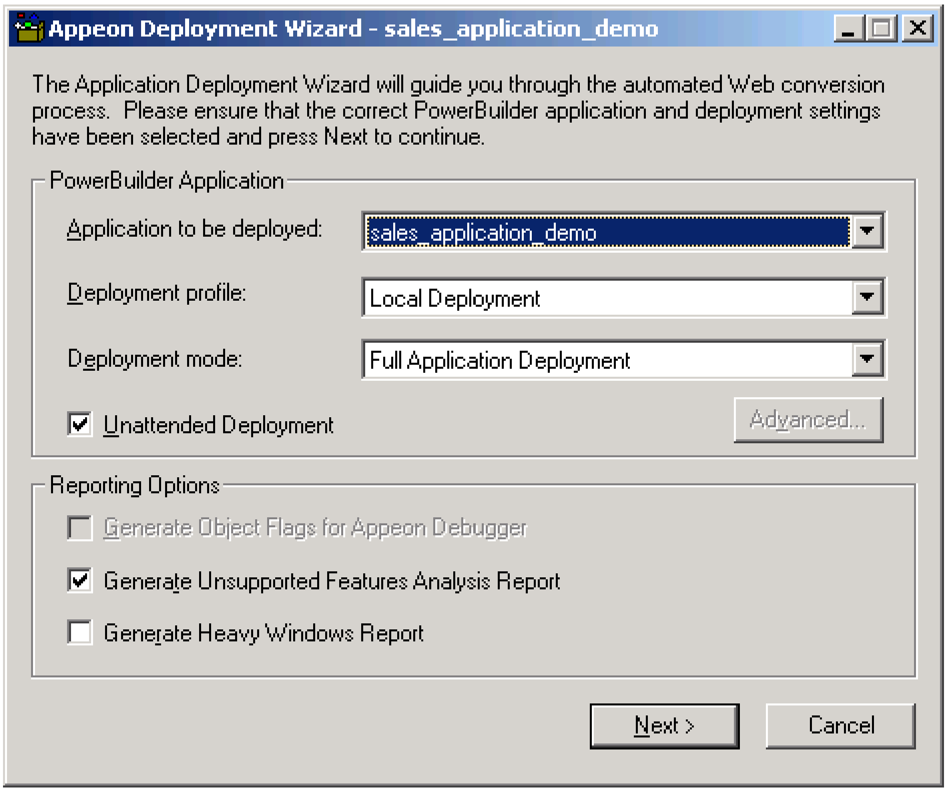
Performing an incremental build in PowerBuilder is necessary because when modifications are made to a parent class, the child class does not reflect the changes unless one of three things occurs:

1. The child class is opened and saved.
2. An incremental build is performed.
3. A full build is performed.

6.3.2 Specifying the deployment settings

Click the *Deploy* button () in the Appeon Developer toolbar. The Appeon Deployment Wizard is displayed.

Figure 6-1: Appeon Deployment Wizard



Appeon Deployment Wizard - sales_application_demo

The Application Deployment Wizard will guide you through the automated Web conversion process. Please ensure that the correct PowerBuilder application and deployment settings have been selected and press Next to continue.

PowerBuilder Application

Application to be deployed: sales_application_demo

Deployment profile: Local Deployment

Deployment mode: Full Application Deployment

☑ Unattended Deployment

Reporting Options

□ Generate Object Flags for Appeon Debugger

☑ Generate Unsupported Features Analysis Report

□ Generate Heavy Windows Report

Table 6-4 describes how to specify the deployment settings for an application:

Table 6-4: Appeon Deployment Wizard settings

Setting	Instruction
Application to be deployed list box	Select an application to be deployed. The default application is listed by default. The application chosen for deployment will become the default application under Application Profiles in Appeon Developer Configuration.
Deployment Profile list box	Select a deployment profile to be used in Web deployment. The selected deployment profile determines which Appeon Server(s) and Web Server(s) the application will be deployed to. You must make sure the Appeon Server(s) and Web Server(s) are started before you perform the deployment.
Deployment Mode list box	Select one of the three available deployment modes: • Full Application Deployment • Incremental Application Deployment • Deploy Already Generated Application
Unattended Deployment check box	Specify whether the whole deployment process will automatically proceed without displaying the *Next* button or waiting for interaction. If you want to view the messages or reports generated at each step, you can set the unattended deployment to false; the wizard will require you to click the *Next* button to start the next task until the whole conversion is complete.

Advanced button	Select the types of objects or files that will be uploaded to servers: DataWindow objects, other objects, INI files, external DLL/OCX files, or image files. By default, all objects and files will be uploaded. This button is effective only when the Deploy Already Generated Application option is selected, therefore, make sure you have performed a full deployment on the application. When using this function, verify that the target servers are the same ones used in the application deployments.
Generate Object Flags for Appeon Debugger	Specify whether to flag the objects for debug with Appeon Debugger. This option is available for the Debug PS/JS and Debug JS modes. De-selecting it will have the following two effects: 1) The deployment process will be faster; 2) The objects will not be able to be debugged in Appeon Debugger, though they can still be debugged in other debugging tools, such as Microsoft Visual InterDev.
Generate Unsupported Features Analysis Report check box	Specify whether to generate an unsupported feature analysis report during deployment. This report lists all unsupported PowerBuilder coding features in the application. You can use this report to remove or work around the unsupported features. For detailed instructions, refer to Section 5.3: *Working with UFA Report*.
Generate Heavy Window Report check box	Specify whether to generate a Heavy Window Report during deployment.

6.3.2.a Selecting the deployment mode

Table 6-5 describes the three different deployment modes and guides you in choosing the proper deployment mode.

Table 6-5: Deployment mode

Deployment Mode	What It Does	Elapsed Time	When To Use It
Full deployment	Exports all PowerBuilder application objects and code, generates the corresponding Web files for the application, and deploys all Web files to the server.	Depends on the size and complexity of the PowerBuilder application.	The first time an application is deployed to the Web.
Incremental deployment	Exports objects, re-generates and re-deploys Web files for the changed objects only. The Deployment Wizard continues even if it finds unsupported features during this process.	Up to 80% less time than a full deployment.	After making changes to an already fully-deployed application’s source code or features in the application profile.

Deploy Already Generated Application	Bypasses the object exporting and file generation tasks and only deploys the Web files for the changed objects or for the entire application.	Much less time than a Full or Incremental Deployment.	When you intend to: 1) deploy the Web files for the changed objects to the server after an Incremental Deployment has been cancelled during Web deployment (Task 3). 2) refresh the set of Web files on the server after a Full Deployment has been cancelled during Web deployment (Task 3). 3) deploy the application to a different server.

Appeon Developer will automatically set the default deployment mode by detecting whether a folder named after the application exists in the %Appeon%\Developer6.1\Projects\ directory on the Developer machine. If the folder is detected, the application will be regarded as an already deployed application and incremental deployment mode is selected; otherwise, full deployment mode is selected. This detection can mislead Developer into selecting incremental deployment mode, because the folder will exist in the following situations:

Table 6-6: Situations when the folder exits

The folder will exist if you have …	In this case, choose …
Performed an unsupported feature analysis on the application before it has undergone a full deployment. The folder is created during analysis.	Incremental deployment mode.
Undeployed an Appeon deployed application from servers using the Application Undeployment Wizard. The folder is preserved.	Full deployment mode.
Deployed the application using a previous version of Appeon Developer. The folder is preserved unless you delete it manually. This legacy folder can be detected by the new Appeon Developer.	Full deployment mode.

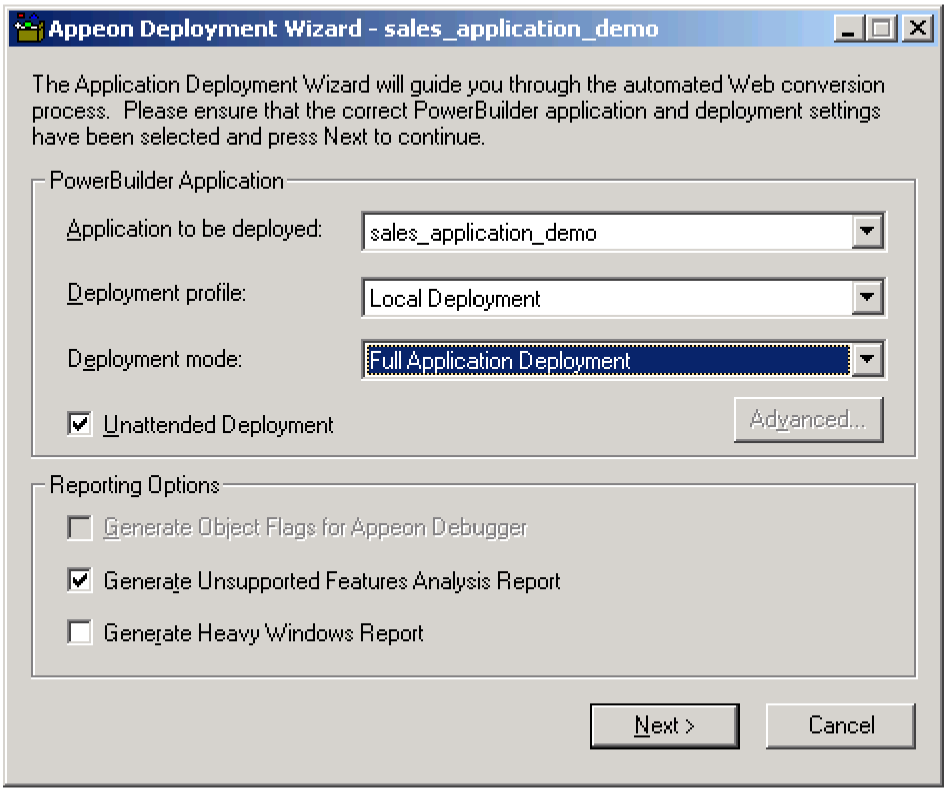
6.3.3 Deploying the PowerBuilder application

The deployment process of an application contains three major deployment tasks. Different deployment modes perform different tasks. Refer to Table 6-7 for the tasks performed during each deployment mode.

Table 6-7: Different tasks performed during each deployment mode

Deployment Task	Full Deployment	Incremental Deployment	Deploy Already Generated Application
Task 1: Application Source Code Export Exports the source code of the original PowerBuilder application.	Y	Y	N
Task 2: Web File Generation Analyzes unsupported features and generates Web files.	Y	Y (Partial)	N
Task 3: Web Deployment Deploys generated Web files to Appeon Server and Web server.	Y	Y (Partial)	Y

Figure 6-2: Full application deployment



Appeon Deployment Wizard - sales_application_demo

The Application Deployment Wizard will guide you through the automated Web conversion process. Please ensure that the correct PowerBuilder application and deployment settings have been selected and press Next to continue.

PowerBuilder Application

Application to be deployed: sales_application_demo

Deployment profile: Local Deployment

Deployment mode: Full Application Deployment

☑ Unattended Deployment

Reporting Options

□ Generate Object Flags for Appeon Debugger

☑ Generate Unsupported Features Analysis Report

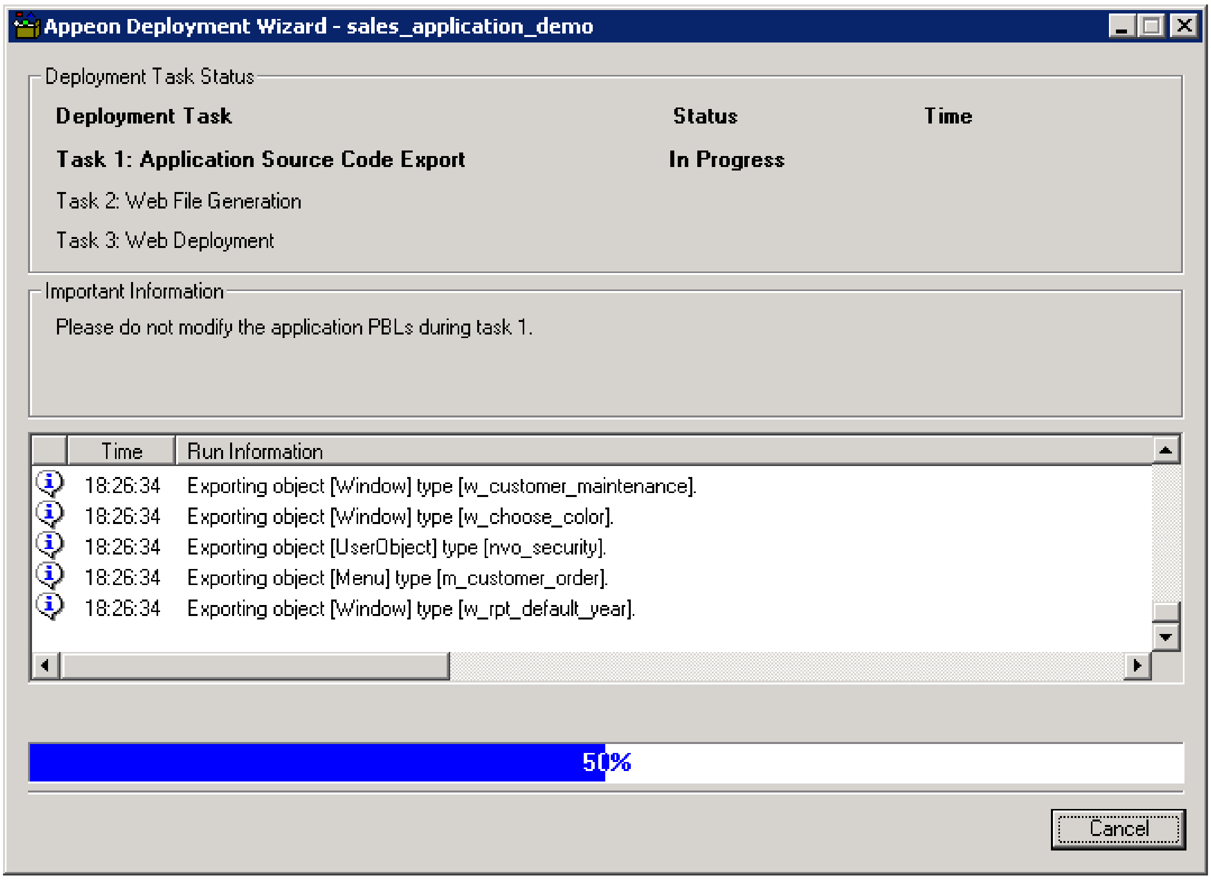
□ Generate Heavy Windows Report

As shown in Figure 6-2, the application to be deployed is the *sales_application_demo_ax* and the deployment mode is Full Application Deployment.

The following are the complete steps in the deployment process:

Step 1 – Click *Next*. The Appeon Deployment Wizard begins Task 1, Application Source Code Export, as shown in Figure 6-3.

Figure 6-3: Task 1: Application Source Code Export



Appeon Deployment Wizard - sales_application_demo

Deployment Task Status

Deployment Task	Status	Time
Task 1: Application Source Code Export	In Progress	
Task 2: Web File Generation		
Task 3: Web Deployment		

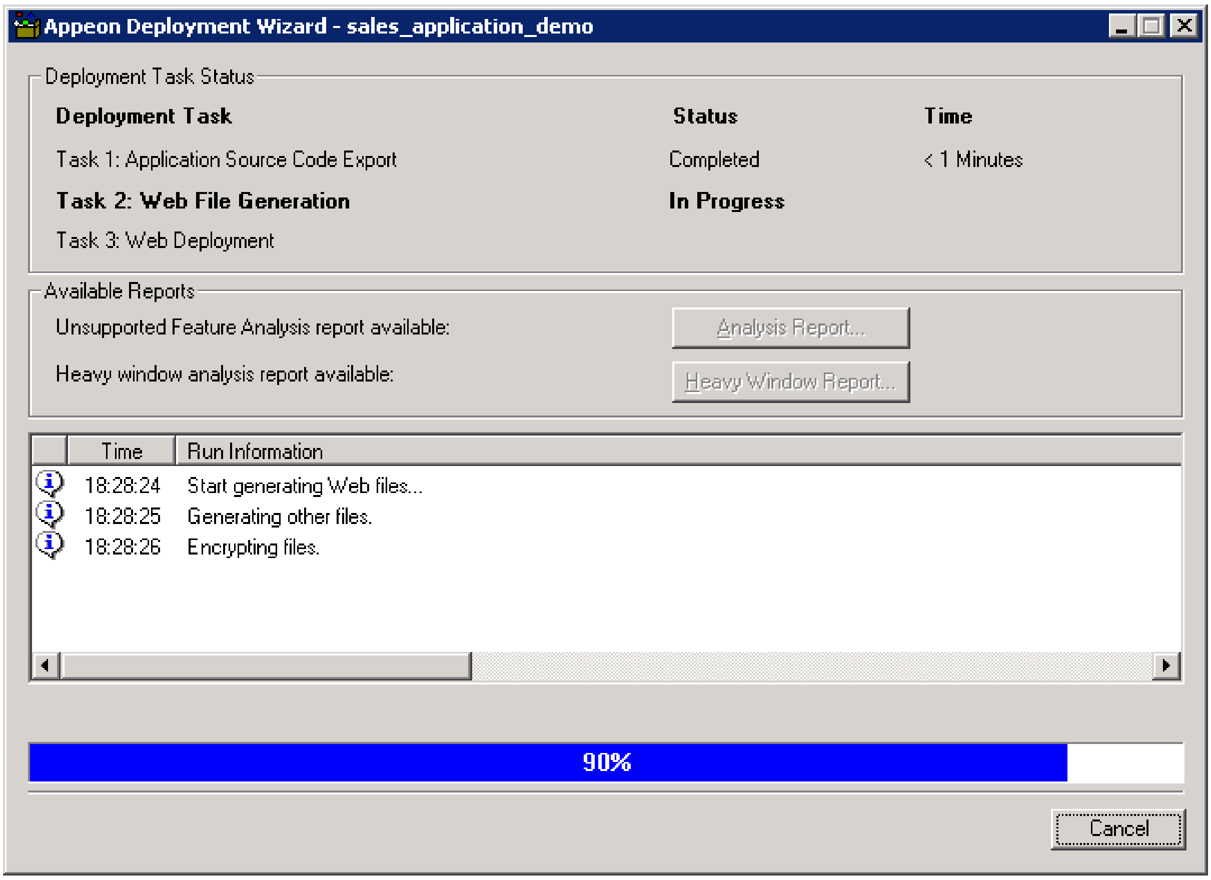
Important Information

Please do not modify the application PBLs during task 1.

	Time	Run Information
ⓘ	18:26:34	Exporting object [Window] type [w_customer_maintenance].
ⓘ	18:26:34	Exporting object [Window] type [w_choose_color].
ⓘ	18:26:34	Exporting object [UserObject] type [nvo_security].
ⓘ	18:26:34	Exporting object [Menu] type [m_customer_order].
ⓘ	18:26:34	Exporting object [Window] type [w_rpt_default_year].

After Task 1 is completed, Task 2, Web File Generation, will automatically start (Figure 6-4) if the Unattended Deployment option is selected, otherwise, you must click *Next* at the bottom of the Appeon Deployment Wizard to begin Task 2.

Figure 6-4: Task 2: Web File Generation



Appeon Deployment Wizard - sales_application_demo

Deployment Task Status

Deployment Task	Status	Time
Task 1: Application Source Code Export	Completed	< 1 Minutes
Task 2: Web File Generation	**In Progress**	
Task 3: Web Deployment		

Available Reports

Unsupported Feature Analysis report available: Analysis Report...

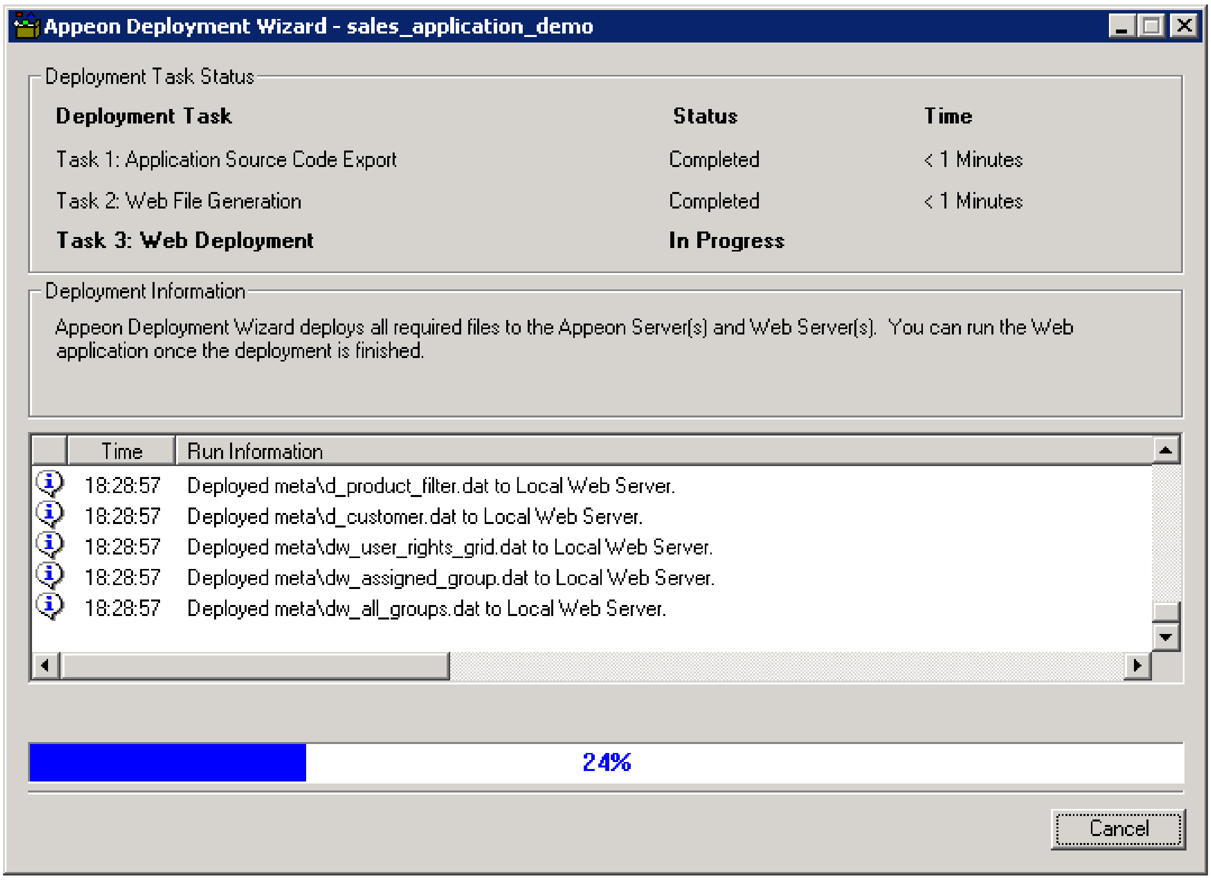
Heavy window analysis report available: Heavy Window Report...

Time	Run Information
18:28:24	Start generating Web files...
18:28:25	Generating other files.
18:28:26	Encrypting files.

After Task 2 is completed, Task 3, Web Deployment, will automatically start (Figure 6-5), or click *Next* at the bottom of the Appeon Deployment Wizard to begin Task 3 if the Unattended Deployment option is not selected.

Confirm that the Web Server and Appeon Server have been started and correctly configured in Appeon Developer Configuration; otherwise, deployment will fail. If deployment fails, a *Retry* button appears so you can retry Task 3 without running through a full deployment again.

Figure 6-5: Task 3 Web Deployment



Appeon Deployment Wizard - sales_application_demo

Deployment Task Status

Deployment Task	Status	Time
Task 1: Application Source Code Export	Completed	< 1 Minutes
Task 2: Web File Generation	Completed	< 1 Minutes
Task 3: Web Deployment	**In Progress**	

Deployment Information

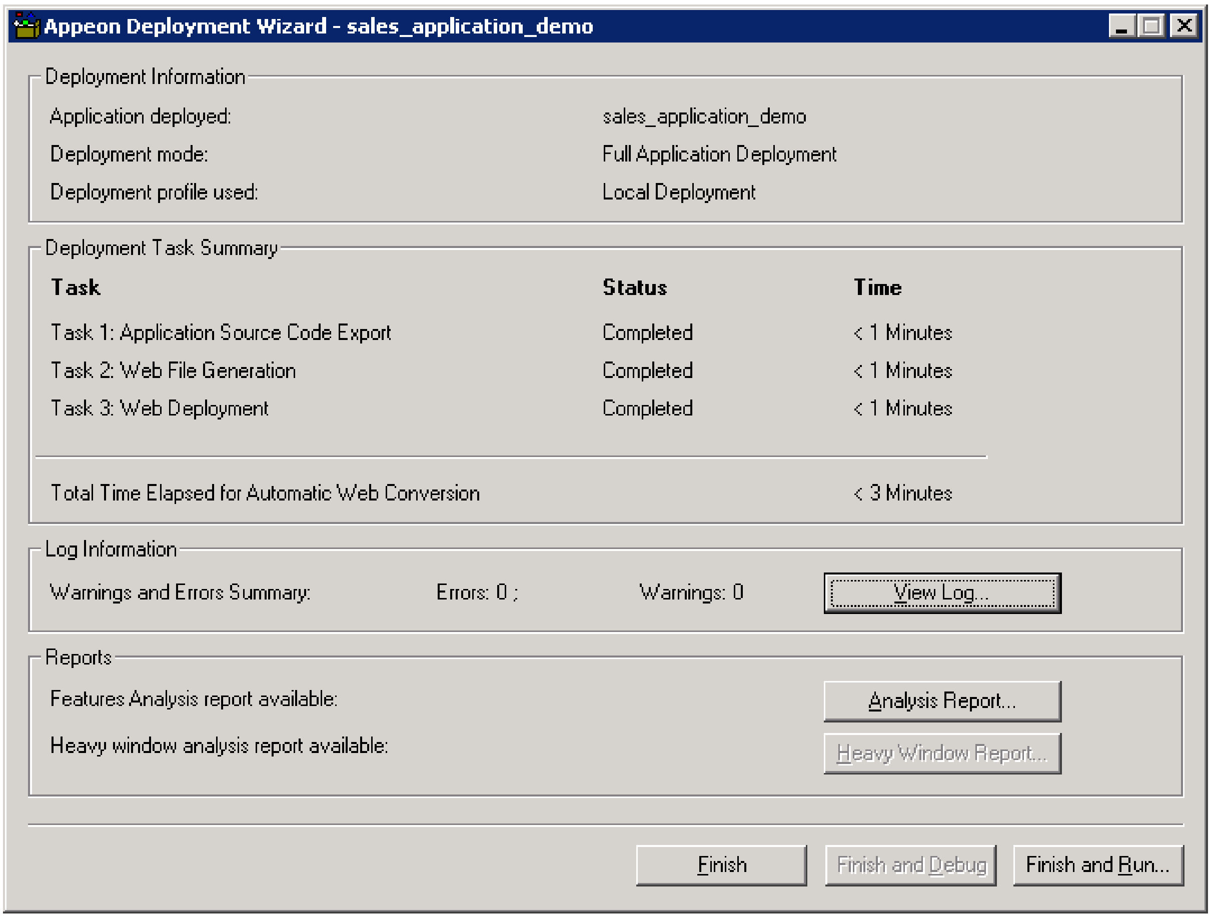
Appeon Deployment Wizard deploys all required files to the Appeon Server(s) and Web Server(s). You can run the Web application once the deployment is finished.

Time	Run Information
18:28:57	Deployed meta\d_product_filter.dat to Local Web Server.
18:28:57	Deployed meta\d_customer.dat to Local Web Server.
18:28:57	Deployed meta\dw_user_rights_grid.dat to Local Web Server.
18:28:57	Deployed meta\dw_assigned_group.dat to Local Web Server.
18:28:57	Deployed meta\dw_all_groups.dat to Local Web Server.

Click the *Cancel* button or press the Esc key to terminate Task 3. If you cancel Task 3 during a Full Deployment, only some of the Web files will have been uploaded to the server and the application will not run correctly. To correct this, select *Incremental Deployment* or *Deploy Already Generated Application* as the deployment mode the next time you attempt to deploy the application.

Step 2 – After Task 3 has been completed, click the *Next* button at the bottom of the Appeon Deployment Wizard page to display the Web deployment report page, as shown in Figure 6-6.

Figure 6-6: Appeon Deployment Wizard



Appeon Deployment Wizard - sales_application_demo

Deployment Information

Application deployed:	sales_application_demo
Deployment mode:	Full Application Deployment
Deployment profile used:	Local Deployment

Deployment Task Summary

Task	Status	Time
Task 1: Application Source Code Export	Completed	< 1 Minutes
Task 2: Web File Generation	Completed	< 1 Minutes
Task 3: Web Deployment	Completed	< 1 Minutes
Total Time Elapsed for Automatic Web Conversion		< 3 Minutes

Log Information

Warnings and Errors Summary: Errors: 0 ; Warnings: 0

Reports

Features Analysis report available:

Heavy window analysis report available:

Table 6-8: Deployment report page

Element	Description
Deployment Information	Displays the name of the application deployed, the deployment mode used, and the deployment profile used.
Deployment Task Summary	Displays the status and time of each of the three deployment tasks, as well as the total time for the entire PowerBuilder-to-Web conversion process.
Log Information	Displays the number of errors and warnings that occurred during the deployment process. Click the *View Log* button to view the errors and warnings generated during the deployment process. The log file can be managed in the Information Manager. For detailed instructions, refer to Chapter 9: *Using Information Manager*.
Reports	Displays the reports (Analysis Report and Heavy Window Report) generated during the deployment process (mainly Task 2): • *Analysis Report* button – This report is available if the *Generate Unsupported Feature Analysis Report* option on the Appeon Deployment Wizard is selected before deployment. After you click this button, the UFA Report Window will be displayed. For detailed instructions, refer to Section 5.3: *Working with UFA Report*. • *Heavy Window Report* button – This report is available if the *Generate Heavy Window Report* option on the Appeon Deployment Wizard is selected before deployment. After you click this button, the Heavy Window Report will be displayed. For detailed instructions, refer to Section 5.4: *Working with Heavy Window Report*.

Step 3 – Click *Finish* to close the Appeon Deployment Wizard, or click *Finish and Debug* to close the Appeon Deployment Wizard and open Appeon Debugger to debug the application, or click *Finish and Run* to close the Appeon Deployment Wizard and automatically launch the newly deployed Web application.

7 Debugging Appeon Web Applications

7.1 Overview

Sometimes a deployed application does not behave well although the original application does. In these situations, you can examine the execution results of the JavaScript code in the deployed application using Appeon Debugger.

The usage of Appeon Debugger is similar to that of PowerBuilder Debugger. You set breakpoints or monitor variables in the PowerBuilder source code, same as in PowerBuilder Debugger. The main difference is that Appeon Debugger runs the generated JavaScript code but not the PowerBuilder source code, and shows the execution results of the JavaScript code.

This chapter assumes that you are familiar with the functions of PowerBuilder Debugger; therefore it does not provide descriptions of common debugging features, but focuses on introducing the specialties of Appeon Debugger. For common debugging features, refer to the *Debugging an application* section in the *PowerBuilder User's Guide*.

7.2 Important Requirements

To start and use Appeon Debugger successfully, be aware of the following important prerequisites:

1. Microsoft Script Debugger must have been correctly installed and registered on the same machine and script debugging is enabled in Internet Explorer. Otherwise, you may encounter problems in starting and using Appeon Debugger. For detailed instructions and solutions, refer to the *Appeon Debugger* section in the *Appeon Troubleshooting Guide*.
2. A script in the deployed application must meet all of the following requirements to be debugged; otherwise Appeon Debugger will simply ignore it:

- The script is deployed in the Debug JS or Debug PS/JS mode, using the local Appeon Developer, and having the Generate Object Flags for Appeon Debugger option enabled in the Appeon Deployment Wizard.

  If you have changed the application PBLs, or the local Web files and the remote Web files are not the same, you must re-deploy the application with the Generate Object Flags for Appeon Debugger option enabled. Or the newly-added code will not be selectable in Appeon Debugger.

- The original PowerScript code of the script is supported by Appeon.

  If the code is unsupported, it is marked with a "»" icon in the Appeon Debugger Source view, and cannot be set as breakpoint, nor be located with the Run To Cursor or Set Next Statement commands in the debugger.

7.3 Introduction to the debugging procedure

Although Appeon Debugger is used for debugging the JavaScript in the deployed application, breakpoints are not set in the JavaScript, but directly in the PowerBuilder code loaded in the Appeon Debugger Source view. The generated JavaScript is invisible throughout the entire process, however the results shown in the Watch view are the execution results of the JavaScript.

The debugger procedure with Appeon Debugger is the typical 5-step procedure as with PowerBuilder Debugger:

Step 1 – Start the debugger;

Step 2 – Set breakpoints;

Step 3 – Run the application in debug mode;

Step 4 – Examine the application at the breakpoints by monitoring variables, expressions, and call stacks. You may also step through the application after a breakpoint to closely examine the effects of each code.

Step 5 – Fix the code.

The actions performed in each step are not very different from those required in PowerBuilder Debugger. However, there are some special requirements related with the actions, and the action results are from the running of JavaScript. Continue reading the following sections for detailed descriptions of these debugging steps.

7.4 Starting Appeon Debugger

To start Appeon Debugger:

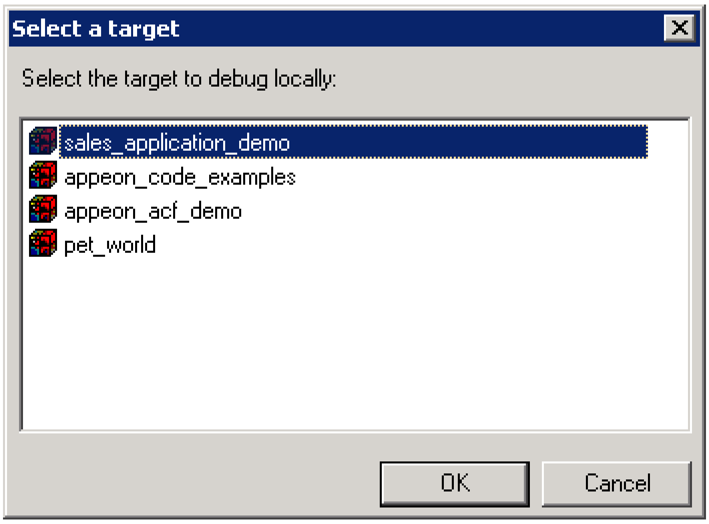
Step 1 – Close the other Appeon Developer tools.

Step 2 – Start Appeon Server and the Web server that host the Web application.

Step 3 – Click *Appeon Debugger* (✋) on the Appeon Developer toolbar.

The "Select a target" window displays, as shown in Figure 7-1.

Figure 7-1: Select an application



Select a target

Select the target to debug locally:

sales_application_demo

appeon_code_examples

appeon_acf_demo

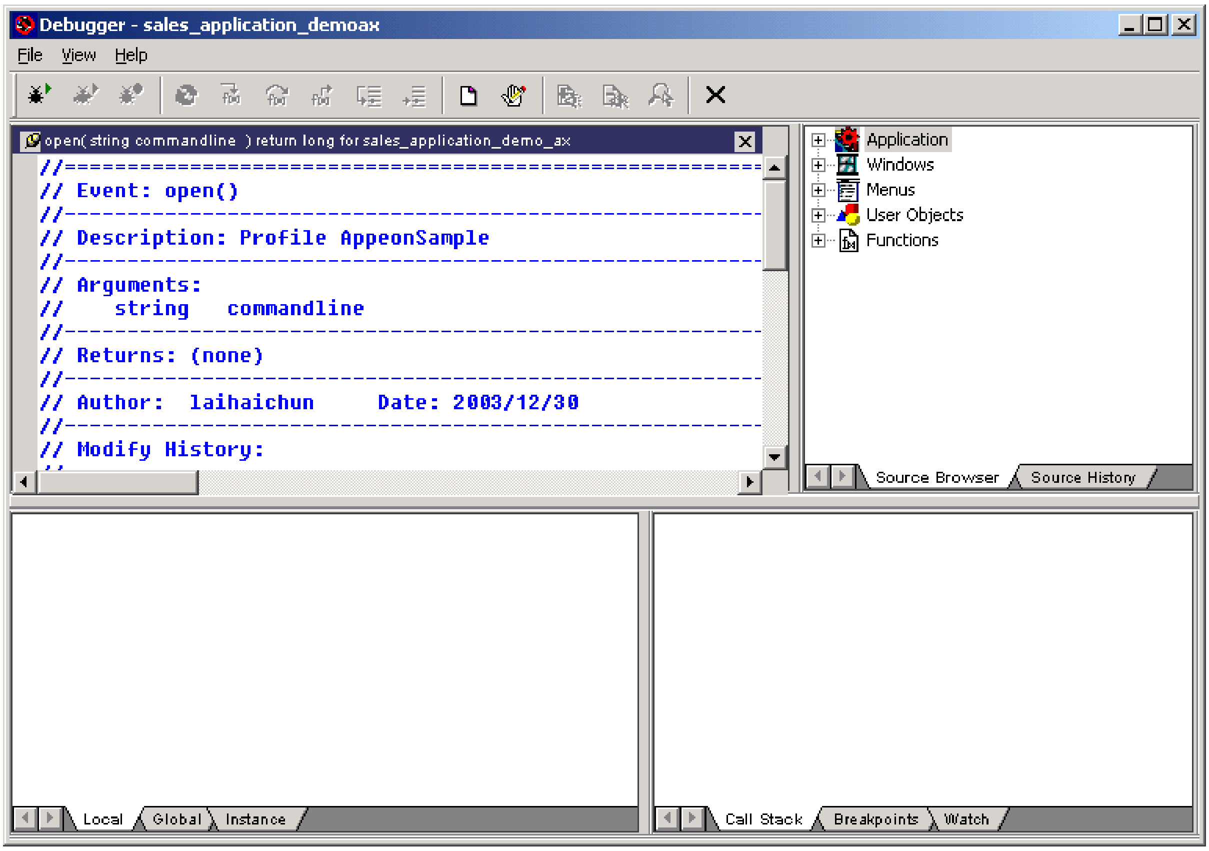
pet_world

Step 4 – Select an application from the list and click *OK*.

Appeon Debugger starts and the selected application is loaded in the Appeon Debugger (Figure 7-2).

The menu bar of the debugger provides three simple menus, File, View and Help. The toolbar is almost the same as that in PowerBuilder Debugger, except that the "Start Remote" button is currently unavailable.

Figure 7-2: Appeon Debugger



```
//========================================================
// Event: open()
//--------------------------------------------------------
// Description: Profile AppeonSample
//--------------------------------------------------------
// Arguments:
//    string   commandline
//--------------------------------------------------------
// Returns: (none)
//--------------------------------------------------------
// Author:  laihaichun      Date: 2003/12/30
//--------------------------------------------------------
// Modify History:
```

7.4.1 Views in Appeon Debugger

Appeon Debugger contains all the views provided by PowerBuilder Debugger except for Instances View and Objects in Memory view. Each Appeon Debugger view shows almost the same information as its counterpart in PowerBuilder Debugger. Table 7-1 summarizes the general differences between the two.

Table 7-1: Comparing the views in Appeon Debugger and PowerBuilder Debugger

Views	Unique to Appeon Debugger (Unavailable in PowerBuilder Debugger)	Unique to PowerBuilder Debugger (Unavailable in Appeon Debugger)
Source	Identifies Appeon unsupported features with the icon (»)	• Ancestor Script popup menu item • Descendant Script popup menu item
Source Browser	None	List of system events, or the events and functions defined in parent classes, or some special events (such as Menu OnCreate event)
Source History	None	None
Variables	The values of variables unsupported by Appeon are marked as “unsupported”. The Global view displays all global variables used in the application, while PowerBuilder Debugger only displays instantiated global variables.	• Parent and Shared views • Parent popup menu item • Shared popup menu item • Break on changes popup menu item

Call Stack	None	None
Breakpoints	None	• Setting special breakpoints • Setting occasional or conditional breakpoints • Setting a breakpoint when a variable changes
Watch	Appeon Debugger handles some expressions in a unique way.	• Break on changes popup menu item

Although the views of Appeon Debugger and PowerBuilder Debugger look similar, they cannot be used together. For example, the breakpoints set in Appeon Debugger cannot be recognized by PowerBuilder Debugger, and vice versa.

7.5 Setting breakpoints

7.5.1 Code lines that can be set as breakpoints

Pay attention to the following **do's** and **don'ts** before setting breakpoints:

1. **Do** set breakpoints to the code lines supported by Appeon.
2. **Do** set simple breakpoints to script, but **don't** set special breakpoints, such as breakpoints triggered when a statement has been executed a specified number of times (an occasional breakpoint), when a specified expression is true (a conditional breakpoint), or when the value of a variable changes.
3. To set a SQL statement that spans multiple lines as a breakpoint, **do** set the breakpoint at the first line, but **don't** set breakpoints at the other lines. If you try to set a breakpoint at the other lines, the breakpoint will be set to the line that follows the SQL statement.
4. To set a PowerScript non-SQL statement, which spans multiple lines with continuation characters (&) as a breakpoint, **do** set the breakpoint at the last line of the statement, and if it is set at the other lines, the breakpoint icon is still added to the last line.

Note: you can set a breakpoint to the FOR statement, but the application will be suspended for only once when this statement is executed.

7.5.2 Methods for setting breakpoints

You can set breakpoints in the Source, Breakpoints, Variables, or Watch view, before or after the application starts to run in debug mode.

Method #1: Setting a breakpoint in the Source view:

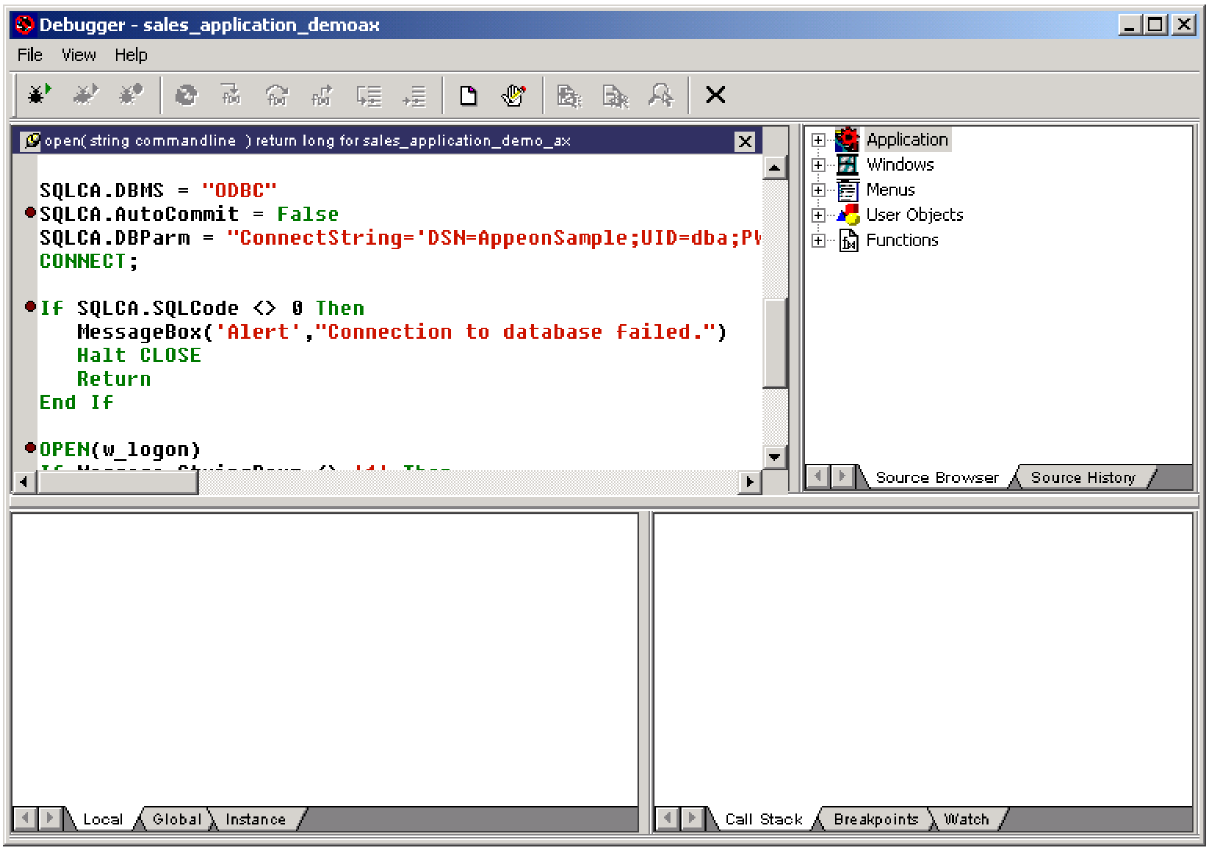
Step 1 – Select the script to display in the Source View.

Select a function or event in the Source Browser view to display its script in the Source View, or right click on the Source view and select *Select Script* from the popup menu to select the script to be displayed.

Step 2 – Right click a line in the Source view and select *Insert Breakpoints* from the popup menu.

Once a breakpoint is set, a red circle displays at the beginning of the line (Figure 7-3).

Figure 7-3: Breakpoints are set



```
 SQLCA.DBMS = "ODBC"
●SQLCA.AutoCommit = False
 SQLCA.DBParm = "ConnectString='DSN=AppeonSample;UID=dba;PW
 CONNECT;

●If SQLCA.SQLCode <> 0 Then
     MessageBox('Alert',"Connection to database failed.")
     Halt CLOSE
     Return
 End If

●OPEN(w_logon)
```

Method #2: Setting a breakpoint in the Breakpoints, Variables, or Watch view:

Step 1 – Right click the Breakpoints, Variables, or Watch view;

Step 2 – Select *Breakpoints* from the popup menu to add a breakpoint using the Edit Breakpoints dialog.

Method #3: Removing or adding a breakpoint at runtime

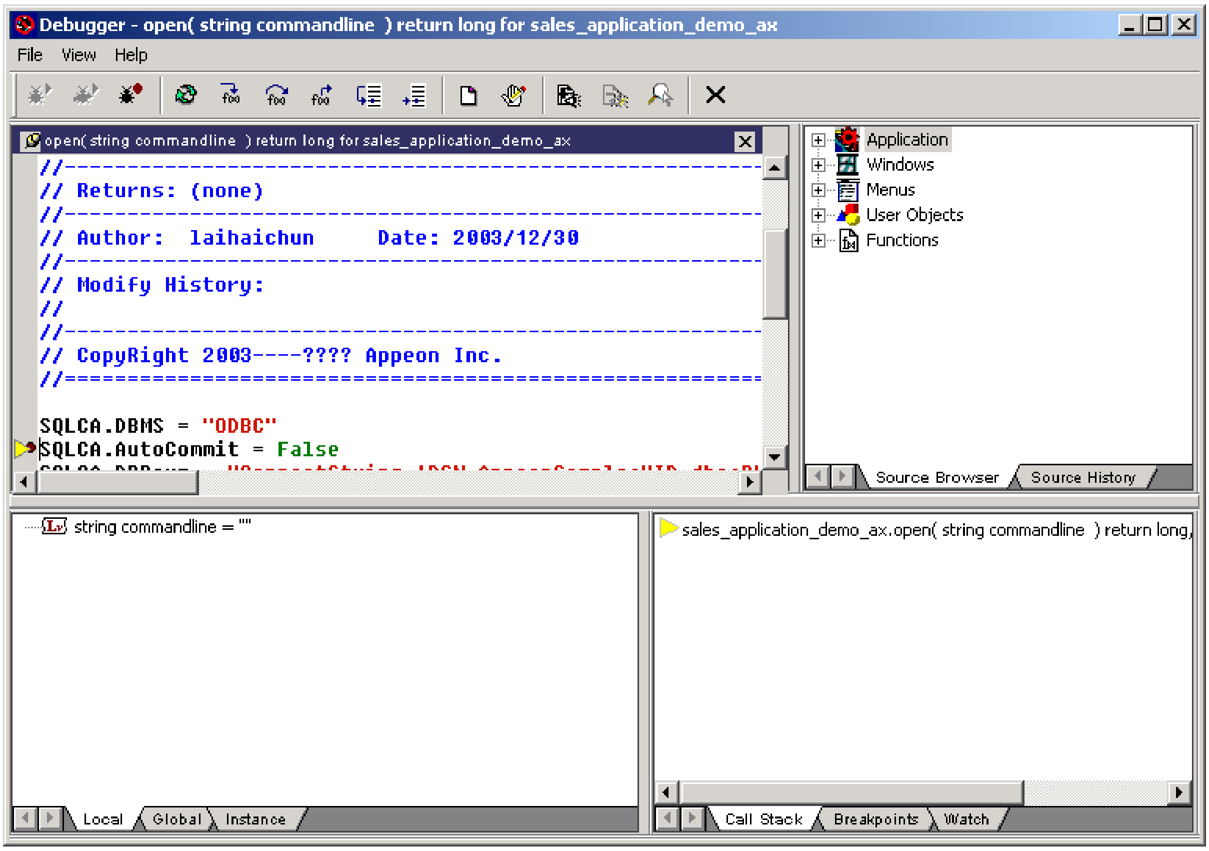
When the application is already running in debug mode, you can still change the way the debug works by removing or adding breakpoints at runtime:

Double-click a line in the Source view, or select *Insert Breakpoint* from the popup menu in the Source view.

7.6 Running the application in debug mode

Click *Start* ( ) on the Appeon Debugger toolbar. The application starts to run in the Internet Explorer. It works normally until it reaches a statement containing a breakpoint. At this point it stops so that you can examine the application in the Appeon Debugger.

Figure 7-4: Application running in debugger mode


Debugger - open(string commandline) return long for sales_application_demo_ax
File
View
Help
open(string commandline) return long for sales_application_demo_ax
Application
Windows
Menus
User Objects
Functions
Source Browser
Source History
Local
Global
Instance
Call Stack
Breakpoints
Watch

```
//-----------------------------------------------------------
// Returns: (none)
//-----------------------------------------------------------
// Author:  laihaichun      Date: 2003/12/30
//-----------------------------------------------------------
// Modify History:
//
//-----------------------------------------------------------
// CopyRight 2003----???? Appeon Inc.
//===========================================================

SQLCA.DBMS = "ODBC"
SQLCA.AutoCommit = False
```

string commandline = ""

sales_application_demo_ax.open(string commandline) return long,

7.7 Examining an application at a breakpoint

Appeon Debugger provides a Quick Watch function, and the Variables, Watch and Call Stack views that work the same as PowerBuilder Debugger for examining the state of the application. It does not provide Break on Changes and TipWatch functions, nor the Object in Memory view.

7.7.1 Special variable and expression handlings

Appeon Debugger watches the variable and expression results out of JavaScript code but not PowerBuilder code. This is the main difference with PowerBuilder Debugger. Despite the difference, the results shall be mostly the same as in PowerBuilder Debugger. If the results are different, there are two possible reasons:

(1) The deployment of the application contains an error. In this case, determine the cause of the error and correct the problem.

(2) Appeon Debugger handles some variables and expressions differently from PowerBuilder Debugger. Such differences are introduced below. Be aware of them, so not be false alarmed and conduct unnecessary complex examination work.

Appeon Debugger does NOT support the following when examining variables:

- Expanding the values of ClassDefinition, PowerObject, and Any variables;
- Expanding the runtime values of System variables. Instead, it just expands the inherent properties of such variables.

For example, with a Window type variable w_1, PowerBuilder Debugger shows all the values of all the controls and variables contained in w_1, while Appeon Debugger only shows the values of the Window properties of for w_1.

Appeon Debugger gives different results when evaluating the following expressions:

- It reports errors for expressions that contain unsupported code;
- If the same expression is added multiple times - in different context - into the Watch, Appeon Debugger regards it as one expression, and evaluates its value according to the context of the breakpoint;
- Appeon Debugger does not validate expressions. Therefore, if an expression does not pass validation in PowerBuilder, PowerBuilder Debugger does not give evaluation results, however Appeon Debugger may give a result, sometimes even an incomprehensible result, such as "[object]+3" for the expression "this + 3".
- If two expressions are interconnected, because the evaluation results of the expressions are dependent on the orders they are executed, and the ordering may follow different rules in PowerBuilder Debugger and Appeon Debugger, Appeon Debugger may produce different results from PowerBuilder Debugger.

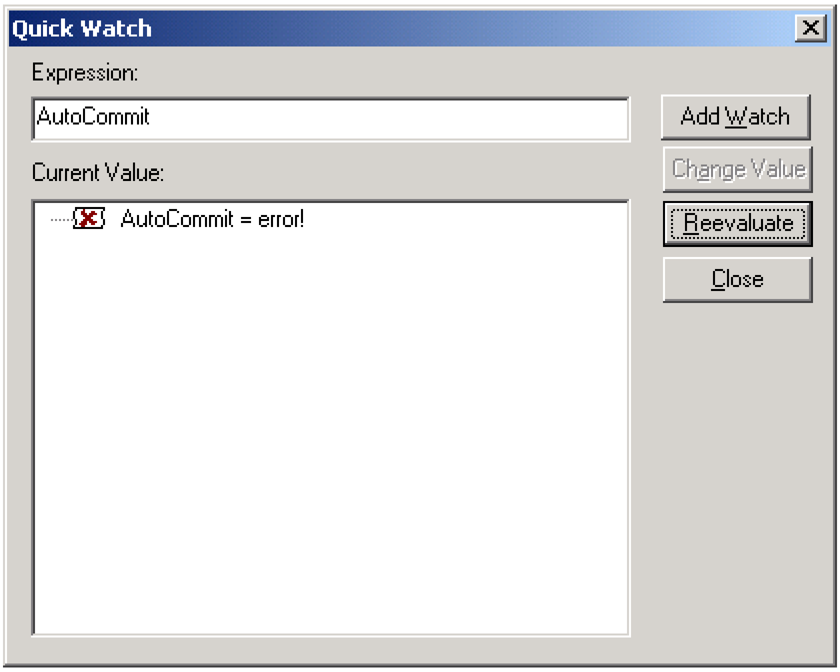
7.7.2 Adding variables or expressions to Watch view

Method #1: Adding variables or expressions to Watch with the Quick Watch

Step 1 – Select a variable or expression or place the cursor to a variable or expression in the Source view, right click and select *Quick Watch* from the popup menu or press Shift+F9.

The variable or the expression will be added to the Expression box on the Quick Watch dialog (Figure 7-5).

Figure 7-5: Quick Watch



AutoCommit

AutoCommit = error!

Step 2 – Click the *Add Watch* button to add the variable or expression to the Watch view.

Method #2: Adding variables or expressions directly in the Watch

Step 1 – Right click the Watch view and select *Insert* from the popup menu.

Step 2 – Type valid PowerScript expression in the New Expression dialog box, and click *OK* to add it to the Watch view.

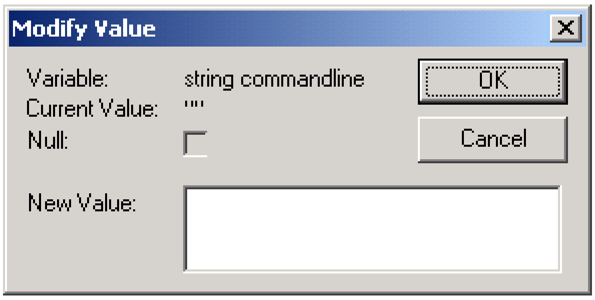
7.7.3 Changing the value of a variable or expression

To change the value of a variable in the Watch view:

Step 1 – Double click a variable or right click a variable and select *Edit Variable*. The Modify Value dialog box will be displayed.

Step 2 – In the Modify Value dialog box (Figure 7-6), type a new value for the variable in the New Value box, click *OK* to close the Modify Value dialog box and continue debugging the application with the variable set to the new value.

Figure 7-6: Modify value



Modify Value

Variable: string commandline

Current Value: ""

Null: □

New Value:

To modify an expression in the Watch view:

Step 1 – Double-click an expression or right click an expression and select *Edit Expression*. The Modify Expression dialog box will be displayed.

Step 2 – In the Modify Expression dialog box, type the new expression in the New Value box and click *OK*.

Notice that it is unsupported to select the Null check box in the Modify Value or Modify Expression dialog to set the value of the variable or expression to null.

7.7.4 Evaluating an expression

To evaluate an expression in the Quick Watch dialog box and add it to the Watch view:

Step 1 – Change the variable or expression in the Expression box.

Step 2 – Click Reevaluate to display the new value in the tree view.

Step 3 – (Optional) Click Add Watch to add the expression to the Watch view.

7.7.5 Examining context in Call Stack view

You can examine the context of the application by clicking the code lines in the call stack.

To show a different context from the Call Stack view:

- Double-click a line or right click a line and select *Set Context*

Once a function is selected in the Call Stack view, the Variable views, the Source view, and the Watch view will all be updated.

7.7.6 Stepping through the application

You can use the Step In, Step Over, Step Out, Run To Cursor, and Set Next Statement commands to step through an application. Using these commands in Appeon Debugger is no different from using them in PowerBuilder Debugger.

7.8 Fixing the code/stopping the debug procedure

Once you locate the cause of the erroneous execution results, you can stop the debugging procedure, return to the PowerBuilder application to fix the code, and re-deploy the application to verify that the error has been corrected.

To stop the debugging process: close Internet Explorer where the application is running, or select *Stop* on the toolbar. The breakpoints you set to the application will be saved when Appeon Debugger is closed, and will be available when Appeon Debugger is re-opened.

When fixing the code, you must return to the PowerBuilder IDE and make code changes in the original PowerBuilder application. Redeploy the application and run the application in Internet Explorer to verify if the problem has been corrected. If the problem still exists and you need to return to Appeon Debugger, make sure you redeploy the application with the Generate Object Flags for Appeon Debugger option enabled. Otherwise, the latest code changes will not be reflected in Appeon Debugger.

8 Running Web Applications

8.1 Overview

Appeon Web applications are accessed via Microsoft Internet Explorer.

Confirm that the Web Server(s) and Appeon Server(s) hosting the Web applications are running, and verify that the Web browser meets the necessary requirements outlined in the *Appeon Installation Guide*.

8.2 Requirements

8.2.1 Windows account with administrator privileges

When you run an Appeon Web application for the first time, the Xcelerator plug-in (for improving the runtime performance of Web applications) must be downloaded to the client. If the login account of the Windows system does not have administrator rights, the Xcelerator plug-in download will be blocked and the application will fail to run.

Ask your administrator to assign the current login account to the Administrators group, so that it can successfully download and register the Xcelerator plug-in. After that, you can use any account to log into Windows and run the application.

8.2.2 Internet Explorer settings

Settings in Internet Explorer can affect the deployed Web application. Sometimes outdated files that are cached in the browser can interfere with how a Web application functions and can cause errors when they are re-deployed many times. Internet Explorer's settings may also block proxy connections or ActiveX control downloads when the Web application is opened. Therefore, you must make sure you configure Internet Explorer's settings accordingly. Note that you may need to have the administrator rights to modify certain settings.

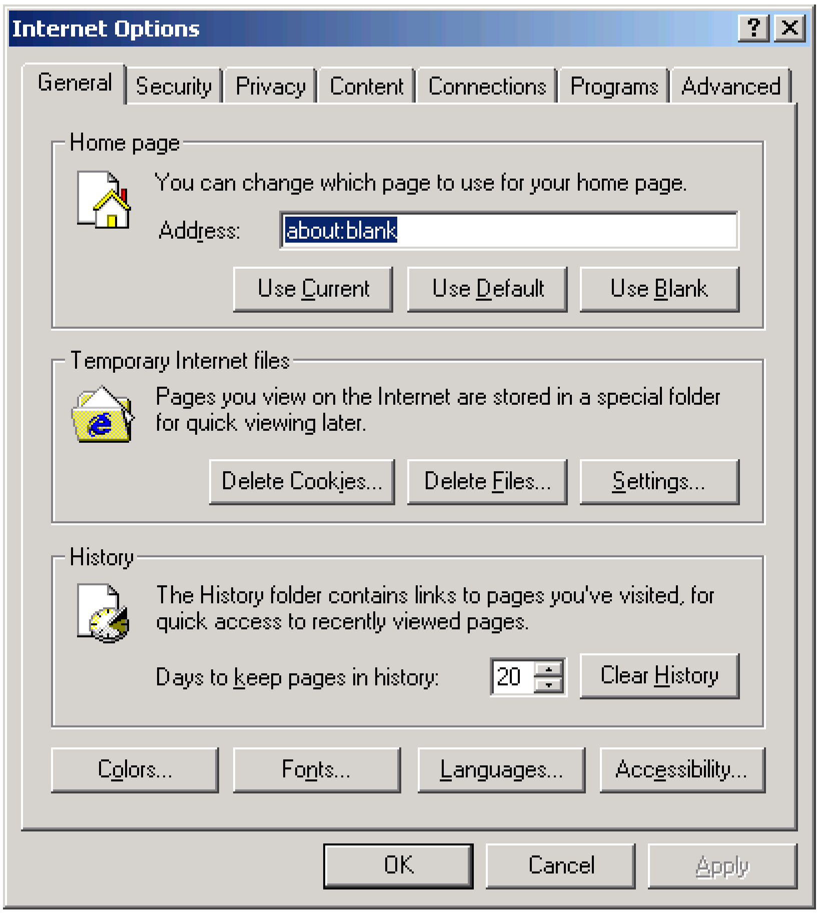
8.2.2.a Settings for temporary Internet files

Sometimes cached files in Internet Explorer can cause problems when accessing the converted Web application after deployment. In order to avoid the risk of providing "stale" data, it is imperative that Internet Explorer has the cache options set to *Automatically* and that there is sufficient disk space on the Client to enable the Web files to be cached.

To set the cache options:

Step 1 – Open Internet Explorer and select Tools | Internet Options, as shown in Figure 8-1.

Figure 8-1: Internet Options in Internet Explorer



Home page

You can change which page to use for your home page.

Address: about:blank

Temporary Internet files

Pages you view on the Internet are stored in a special folder for quick viewing later.

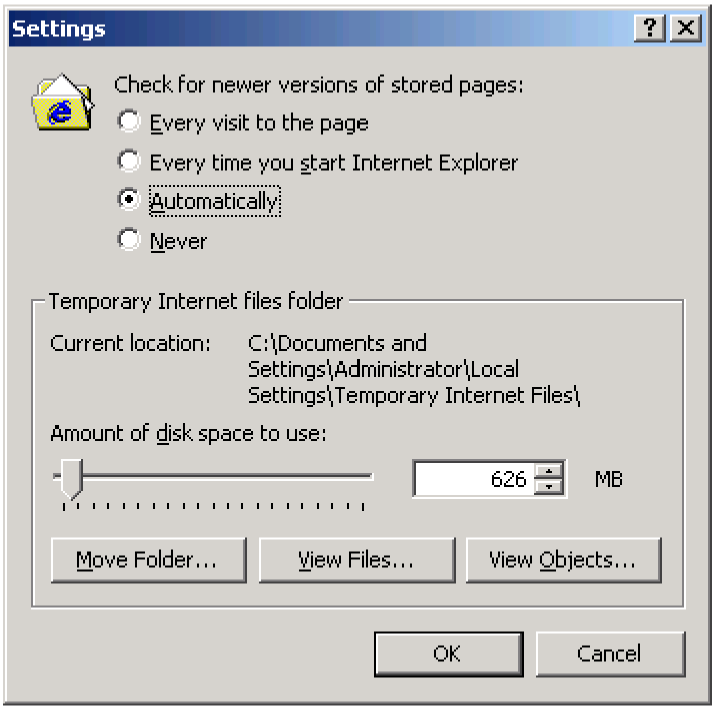
History

The History folder contains links to pages you've visited, for quick access to recently viewed pages.

Days to keep pages in history: 20

Step 2 – Click the *Settings* button in the *Temporary Internet Files* group box, as shown in Figure 8-2.

Figure 8-2: Temporary Internet File Settings



Check for newer versions of stored pages:

○ Every visit to the page

○ Every time you start Internet Explorer

● Automatically

○ Never

Temporary Internet files folder

Current location: C:\Documents and Settings\Administrator\Local Settings\Temporary Internet Files\

Amount of disk space to use: 626 MB

Step 3 – Select the radio button to *Automatically* check for newer versions of stored pages, as shown in Figure 8-2.

Step 4 – Verify that the *Amount of disk space to use* scroll box is set to at least 200 MB. Click *OK* in the Settings window.

8.2.2.b Advanced settings for temporary Internet Explorer files folder

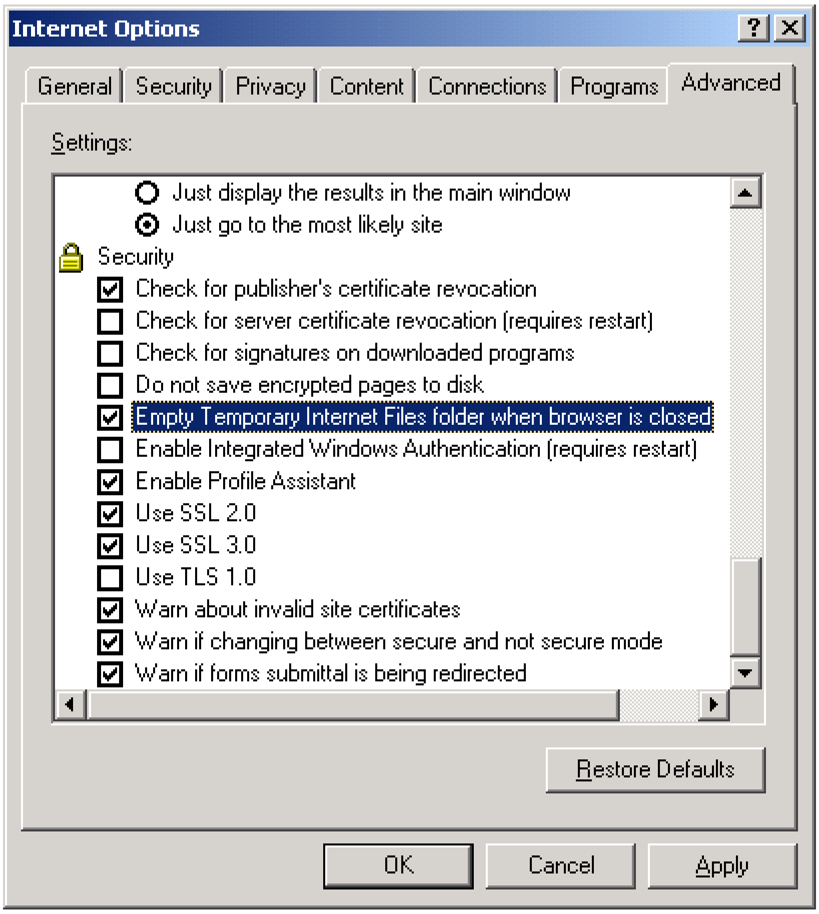
During development, the "Temporary Internet Files folder" should be set to automatically empty each time Internet Explorer is closed. This will force Internet Explorer to re-download the entire application every time it is opened, to ensure the latest version of the application is present, and to prevent interference from outdated cached files.

Here are the necessary steps required to configure Internet Explorer to flush its temporary files each time the browser is closed:

Step 1 – Click the *Advanced* tab in the Internet Options window, as shown in Figure 8-3.

Step 2 – Check the *Empty Temporary Internet files folder when browser is closed* check box under the Security section, as shown in Figure 8-3.

Figure 8-3: Empty Temporary Internet Files folder when browser is closed



Settings:

○ Just display the results in the main window
● Just go to the most likely site
Security
☑ Check for publisher's certificate revocation
□ Check for server certificate revocation (requires restart)
□ Check for signatures on downloaded programs
□ Do not save encrypted pages to disk
☑ Empty Temporary Internet Files folder when browser is closed
□ Enable Integrated Windows Authentication (requires restart)
☑ Enable Profile Assistant
☑ Use SSL 2.0
☑ Use SSL 3.0
□ Use TLS 1.0
☑ Warn about invalid site certificates
☑ Warn if changing between secure and not secure mode
☑ Warn if forms submittal is being redirected

Step 3 – Click *Apply* in the *Advanced* tab and then *OK* in the Internet Options window to allow the settings to take effect.

Allowing the temporary Internet files folder to be emptied each time the browser is closed, is recommended only during the development stage. Once your Web application is ready for production deployment, and frequent changes are not being made to the application, this setting can be disabled once again (unchecked). This setting should be reset once development is complete so the Web application can be cached in each Client PC for better Client-side performance. Using this set of guidelines, caching files should be enabled when they are needed for performance improvement, and disabled during the development and debugging stages.

8.2.2.c Advanced settings for proxy server

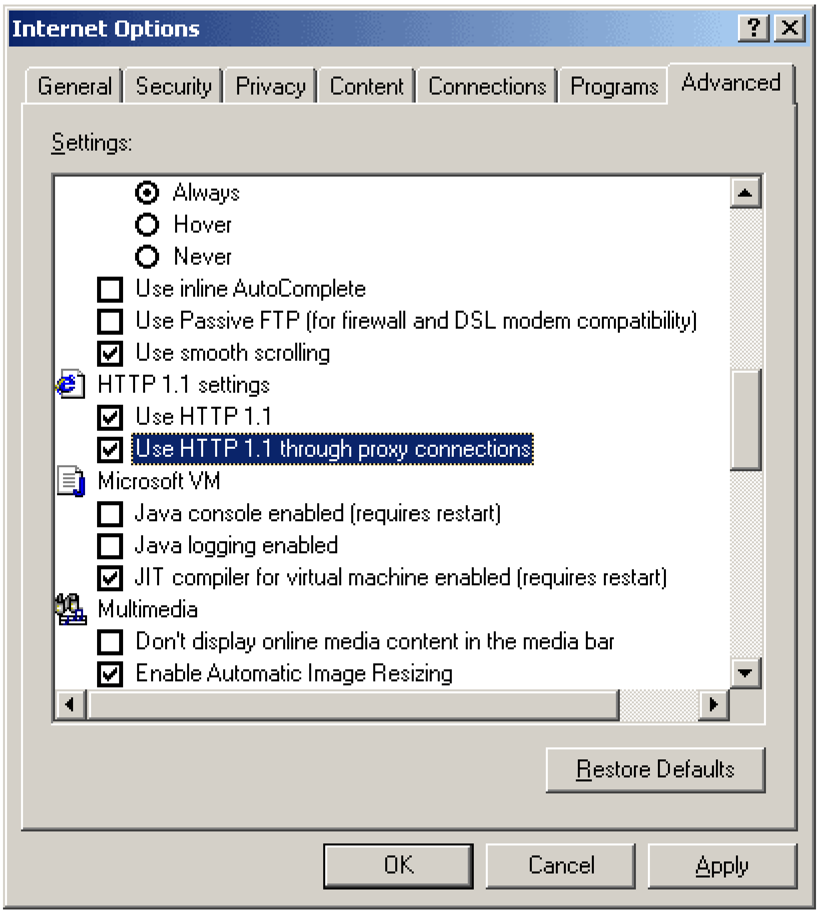
If the Client visits the Web application through a proxy server, you must select the *Use HTTP 1.1 through proxy connections* option under the HTTP 1.1 settings section, as shown in Figure 8-4.

The following steps detail the configuration needed in Internet Explorer:

Step 1 – Click the *Advanced* tab in the Internet Options window, as shown in Figure 8-4.

Step 2 – Check the *Use HTTP 1.1 through proxy connections* check box under the HTTP 1.1 settings section, as shown in Figure 8-4.

Figure 8-4: Proxy connection settings



Settings:

- ● Always
- ○ Hover
- ○ Never
- □ Use inline AutoComplete
- □ Use Passive FTP (for firewall and DSL modem compatibility)
- ☑ Use smooth scrolling

HTTP 1.1 settings

- ☑ Use HTTP 1.1
- ☑ Use HTTP 1.1 through proxy connections

Microsoft VM

- □ Java console enabled (requires restart)
- □ Java logging enabled
- ☑ JIT compiler for virtual machine enabled (requires restart)

Multimedia

- □ Don't display online media content in the media bar
- ☑ Enable Automatic Image Resizing

Step 3 – Click *Apply* in the *Advanced* tab and then *OK* in the Internet Options window to allow the settings to take effect.

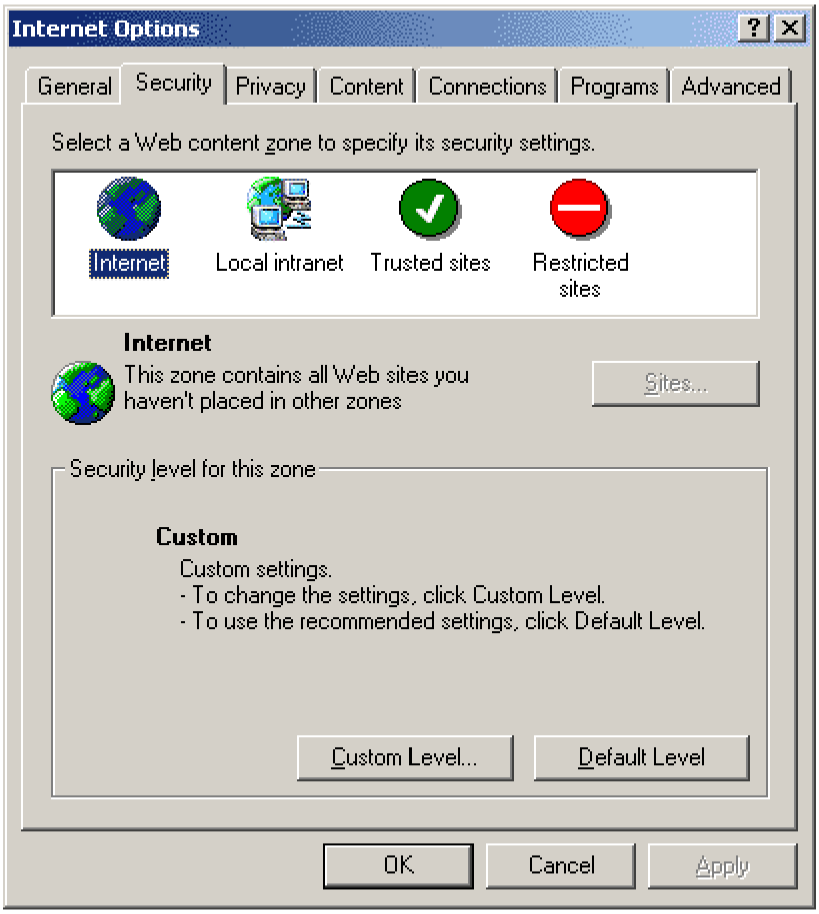
8.2.2.d Security settings for ActiveX controls and Appeon Xcelerator plug-in download

If the Web application calls OLE objects, DLL files, or runs executable programs on the Client, then when the Web application is run for the first time on the Client the user will need to download an ActiveX control (for executing functions that involve client-side integration) and the Appeon Xcelerator plug-in (for improving the runtime performance of Web applications) from the Web Server. To ensure that Internet Explorer will not block the download, configure Internet Explorer's settings, using the following steps:

Step 1 – Open Internet Explorer and select Tools | Internet Options | Security tab.

Step 2 – Verify that you selected an appropriate Web content zone, as shown in Figure 8-5.

Figure 8-5: Select a Web content zone



Select a Web content zone to specify its security settings.

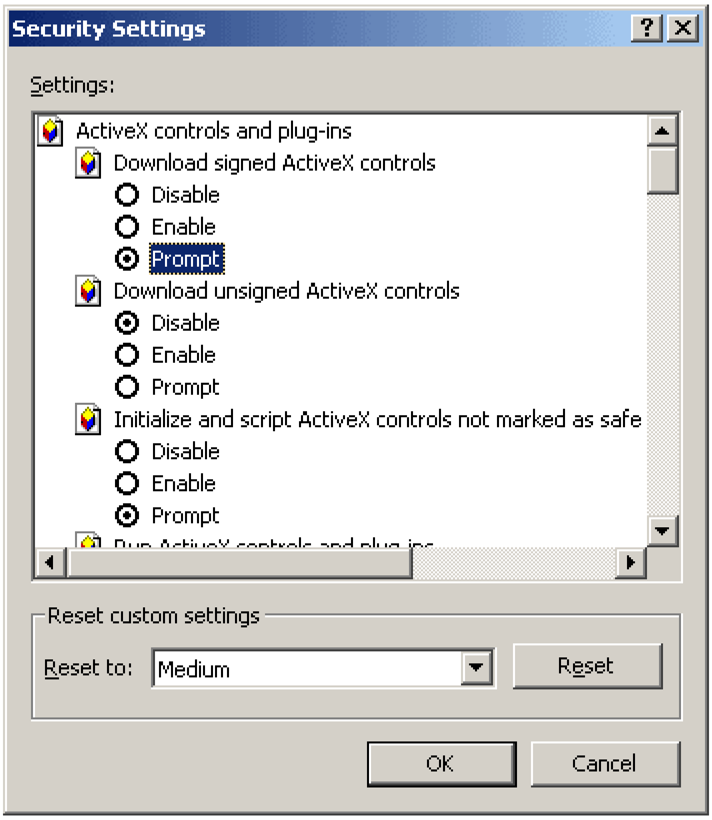
Internet
This zone contains all Web sites you haven't placed in other zones

Security level for this zone

Custom
Custom settings.
- To change the settings, click Custom Level.
- To use the recommended settings, click Default Level.

Step 3 – Click the *Custom Level*…button in the *Security* tab. The Security Settings page is displayed, as shown in Figure 8-6.

Figure 8-6: Security Settings page



Security Settings

Settings:

- ActiveX controls and plug-ins
  - Download signed ActiveX controls
    - ○ Disable
    - ○ Enable
    - ● Prompt
  - Download unsigned ActiveX controls
    - ● Disable
    - ○ Enable
    - ○ Prompt
  - Initialize and script ActiveX controls not marked as safe
    - ○ Disable
    - ○ Enable
    - ● Prompt
  - Run ActiveX controls and plug-ins

Reset custom settings

Reset to: Medium

Step 4 – Check the *Enable* or the *Prompt* (recommended) radio button for the following options:

- Download signed ActiveX controls and plug-ins (or Download unsigned ActiveX controls in Windows XP)
- Initialize and script ActiveX controls not marked as safe
- Run ActiveX controls and plug-ins
- Script ActiveX controls marked safe for scripting
- File download
- Active scripting
- Allow active content to run in files on My Computer (If the version of Internet Explorer is 6.0 SP2)

If *Enable* is selected, the ActiveX control will download automatically without prompting. If *Prompt* is selected, your approval will be required before downloading takes place.

8.2.3 Language setting requirements

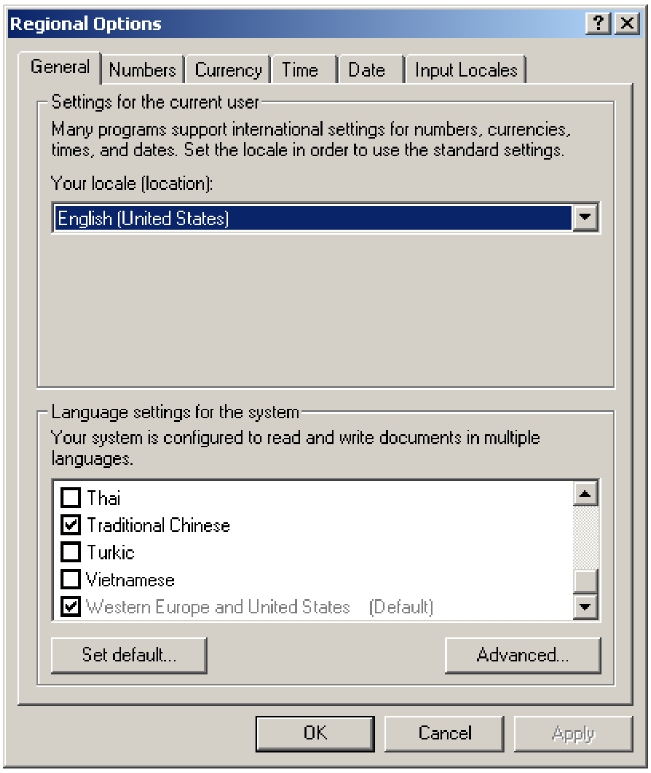
The language of the data in the database used by the application should match the operating system language settings on the machine where Appeon Server is installed. If the language settings conflict, problems will occur when the Web application is run.

Verify that the language of the data in the database used by the Appeon-deployed Web application matches the language settings of the client operating system:

Step 1 – Select Windows Start Menu | Settings | Control Panel.

Step 2 – Click *Regional Options*, as shown in Figure 8-7.

Figure 8-7: Regional Options



Settings for the current user

Many programs support international settings for numbers, currencies, times, and dates. Set the locale in order to use the standard settings.

Your locale (location): English (United States)

Language settings for the system

Your system is configured to read and write documents in multiple languages.

- □ Thai
- ☑ Traditional Chinese
- □ Turkic
- □ Vietnamese
- ☑ Western Europe and United States (Default)

Step 3 – Verify that the language selected in the "Your locale (location)" list box and the language checked as default in "Language settings for the system" settings are the same. The language can be from different regions as long as it is the same language.

This configuration is not required for English language applications.

8.2.4 Disabling anti pop-up software

If anti-popup software (example: Popup Stopper by Panicware) is installed and is active on the Client machine, it will not allow any browser pop-ups to occur. This can cause certain operations to become disabled in the converted Web applications such as the enhanced Web features detailed below, or other operations that automatically load an Internet Explorer window or popup. To prevent any problems, it is recommended that users disable any anti-popup software while using the converted Web applications.

8.3 Accessing Web applications

8.3.1 URLs for Web applications

A typical URL for a Web application is similar to one of the following:

http://192.0.0.80:8080/appeondemo/

http://192.0.0.80:8080/appeondemo/index.html

A complete Web application URL consists of four parts:

Table 8-1: Web URL

Web URL	Description
Protocol	Uses "http://" typically. If your Web Server is configured as an SSL Web Server, use "https://".
IP address or domain name	Uses the IP address or domain name of the Web Server. Using "localhost" listeners in a production environment is not recommended, although the Appeon installation program has created an HTTP listener (localhost:9988) for EAServer\Appeon Server.
HTTP port	(Optional) Specifies the port number that your Web Server accepts for HTTP connections. If it is not specified, the default port is 80. If several Web Servers are running on the same machine, make sure that the port specified is a port of the Appeon Web Server.
Web file path	Specifies the path of the entry page (index.html) of the Web application, relative to the home directory of the Web server. It is the path set for the Web application URL in Application Profiles, for example, "demo/" or "demo/index.html".

8.3.2 Running Web applications

There are two methods to run a Web application:

- **Typing the URL of the Web application in Internet Explorer address bar**: You can always use this method to access the application, on the condition that the computer meets the requirements in Section 8.2: *Requirements*.
- **Launching the Web application from the Appeon Developer *Run* button**: *Run* is a shortcut to Appeon applications that are deployed to the default Appeon Server and Web Server.

This section explains how to launch a Web application from the Appeon Developer *Run* button.

8.3.2.a Launching Web applications from *Run* button

You can launch the Web application by clicking the *Run* button in the Appeon Developer toolbar. Appeon Developer automatically remembers the URLs of the Web applications that are deployed to the default Appeon Server and Web Server.

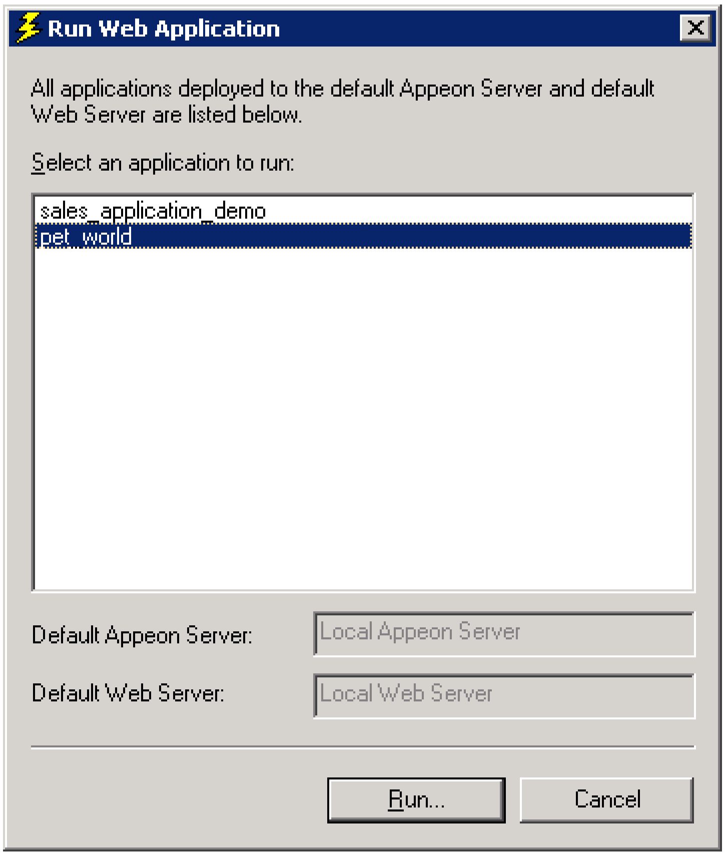
To run a Web application from the Appeon Developer toolbar:

Step 1 – Verify that the default Appeon Server and Web Server in the Server Profiles settings have been started.

Step 2 – Click the *Run* button (⚡) on the Appeon Developer toolbar.

If more than one application is deployed, the Run Web Application window is displayed as shown in Figure 8-8, prompting you to select which application to run.

Figure 8-8: Run Web Application window



Run Web Application

All applications deployed to the default Appeon Server and default Web Server are listed below.

Select an application to run:

sales_application_demo
pet_world

Default Appeon Server: Local Appeon Server

Default Web Server: Local Web Server

Only applications deployed to the Default Appeon Server and Default Web Server are listed in the Run Web Application window.

If you have deployed an application to several different Web file folders on the same Web Server, Appeon Developer will only remember the URL used in the last deployment, although the URLs to access the other Web file folders used in the previous deployments will also be valid.

Step 3 – Select a Web application from the list and click *Run*.

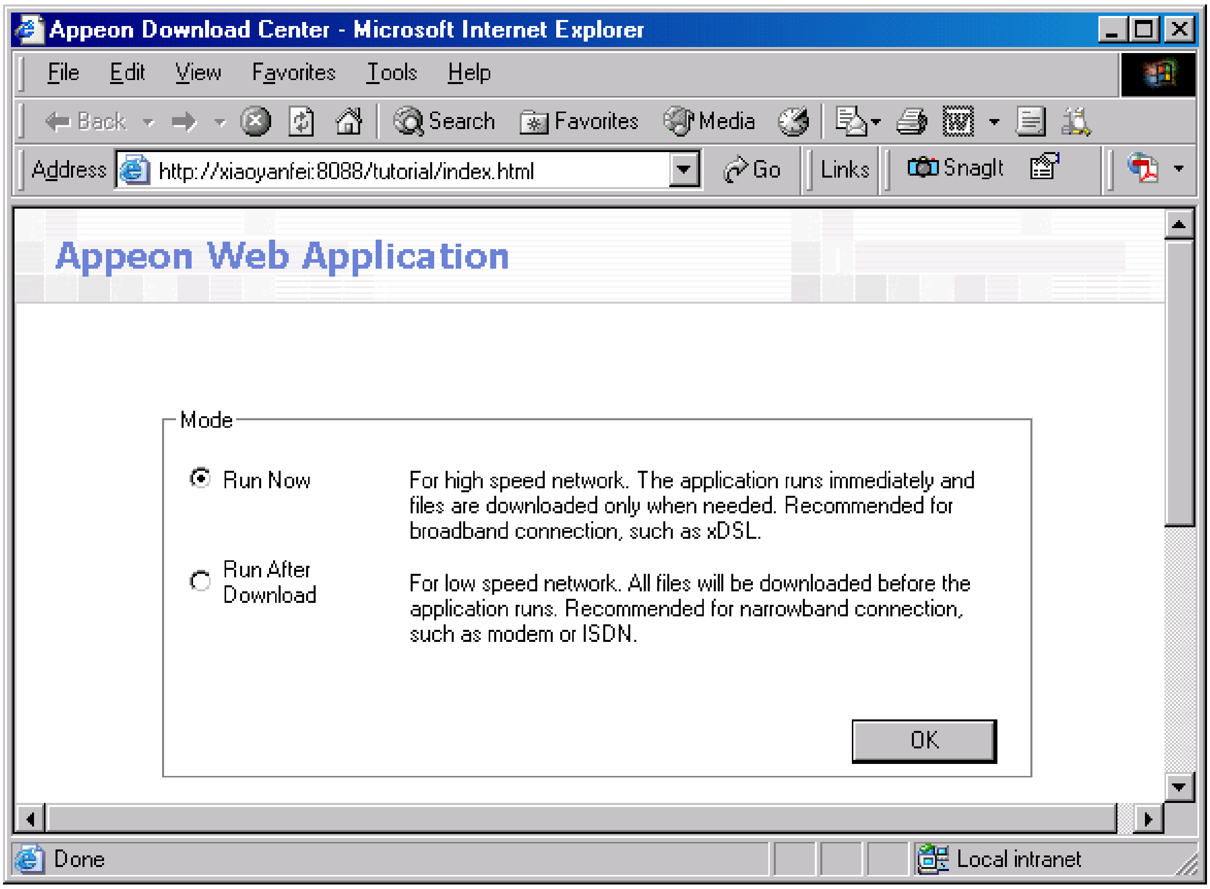
8.3.2.b Selecting the running mode

When you run a Web application for the first time, a run mode page will be displayed, unless you have deselected the "Allows the user to select a run mode" option during the application profile configuration.

This page allows you to select from the following run modes (Figure 8-9):

- Run Now – The application runs immediately and files will be downloaded only when used. This mode is recommended for high speed network.
- Run After Download – Files will be downloaded before the application runs. This mode is recommended for low speed network.

Figure 8-9: Run Mode window



Appeon Download Center - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://xiaoyanfei:8088/tutorial/index.html Go Links SnagIt
Appeon Web Application
Done
Local intranet

Mode

● Run Now — For high speed network. The application runs immediately and files are downloaded only when needed. Recommended for broadband connection, such as xDSL.

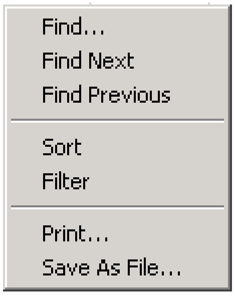
○ Run After Download — For low speed network. All files will be downloaded before the application runs. Recommended for narrowband connection, such as modem or ISDN.

OK

8.4 Appeon DataWindow menu

Appeon provides a Web enhancement, the Appeon DataWindow menu, for all DataWindows deployed with Appeon. When right-clicking on a Web DataWindow, you can access the Appeon DataWindow popup menu: Find, Find Next, Find Previous, Sort, Filter, Print, and Save As File…, as shown in the following figure.

Figure 8-10: Appeon DataWindow popup menu



8.4.1 Enabling Appeon DataWindow menu

To use the Appeon DataWindow menu, you must enable it by using two PowerBuilder global functions: *appeonpopmenu* and *appeonpopmenuon* that are defined in *appeon_workarounds.pbl*. For detailed instructions on how to use *appeonpopmenu* and *appeonpopmenuon* functions in a DataWindow, refer to *Appeon Workarounds Guide* | *Appeon Workarounds PBL Reference* | *Appeon Client Functions*.

8.4.2 Using Appeon DataWindow Menu

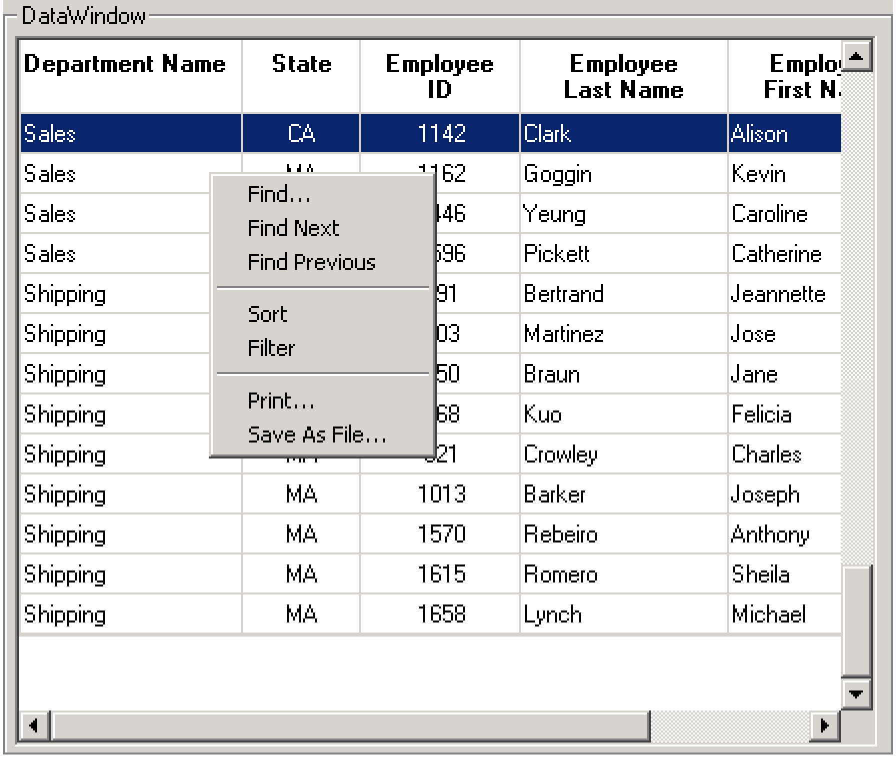
Appeon DataWindows menu have the following features: Find, Find Next, Find Previous, Sort, Filter, Print, and Save As File. Perform the following steps to use the menu in the Web DataWindows.

8.4.2.a Find

The user can search for data using the Find, Find Next and Find Previous functions.

Step 1 – Right-click the Web DataWindow and select *Find* from the popup menu, as shown in Figure 8-11.

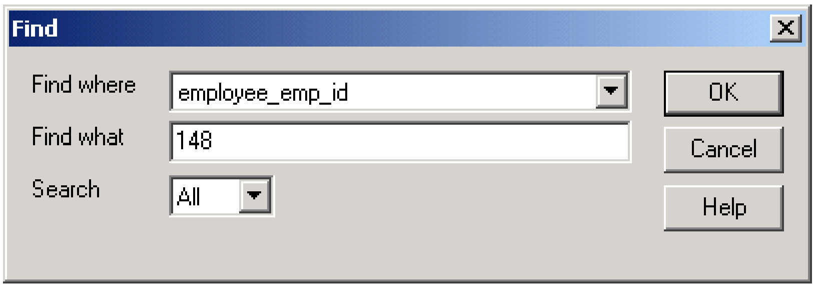
Figure 8-11: Appeon DataWindow Menu



Department Name	State	Employee ID	Employee Last Name	Emplo First N
Sales	CA	1142	Clark	Alison
Sales		162	Goggin	Kevin
Sales		46	Yeung	Caroline
Sales		596	Pickett	Catherine
Shipping		91	Bertrand	Jeannette
Shipping		03	Martinez	Jose
Shipping		50	Braun	Jane
Shipping		68	Kuo	Felicia
Shipping		21	Crowley	Charles
Shipping	MA	1013	Barker	Joseph
Shipping	MA	1570	Rebeiro	Anthony
Shipping	MA	1615	Romero	Sheila
Shipping	MA	1658	Lynch	Michael

Step 2 – Select the column and search scope and input criteria for the search, as shown in Figure 8-12.

Figure 8-12: Find



Find where: employee_emp_id

Find what: 148

Search: All

Step 3 – The first row that matches the search criteria will be highlighted.

Step 4 – Right-click and select *Find Previous* or *Find Next* to highlight the rows (one at a time) that match the criteria.

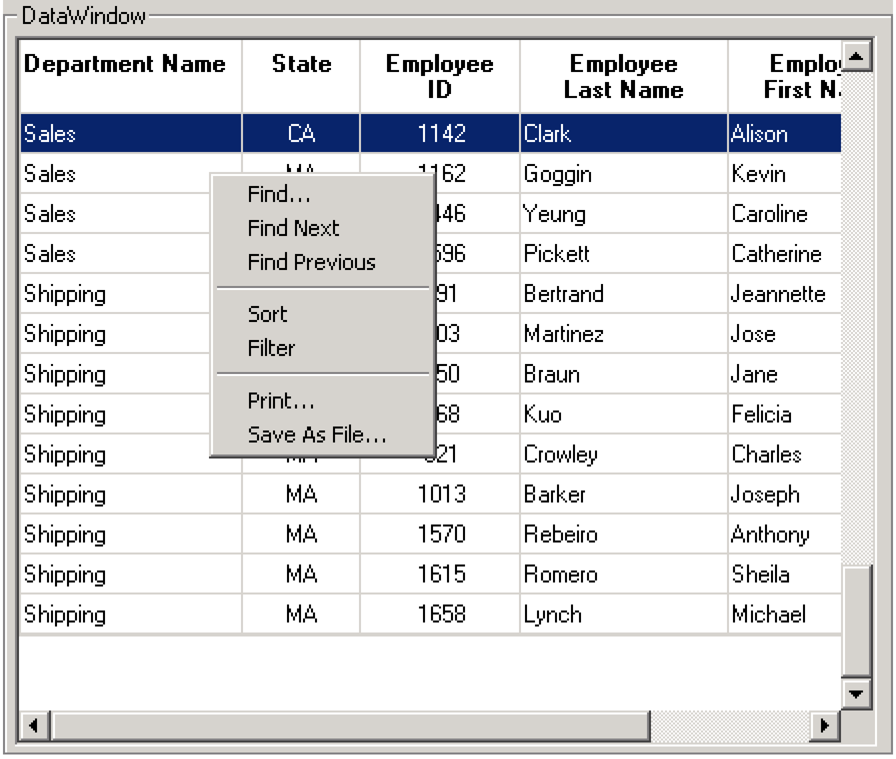
8.4.2.b Sort and filter

The user can sort data by the column in ascending or descending order, or filter data using a number of functions.

To sort data:

Step 1 – Right-click the Web DataWindow and select *Sort* from the popup menu, as shown in Figure 8-13.

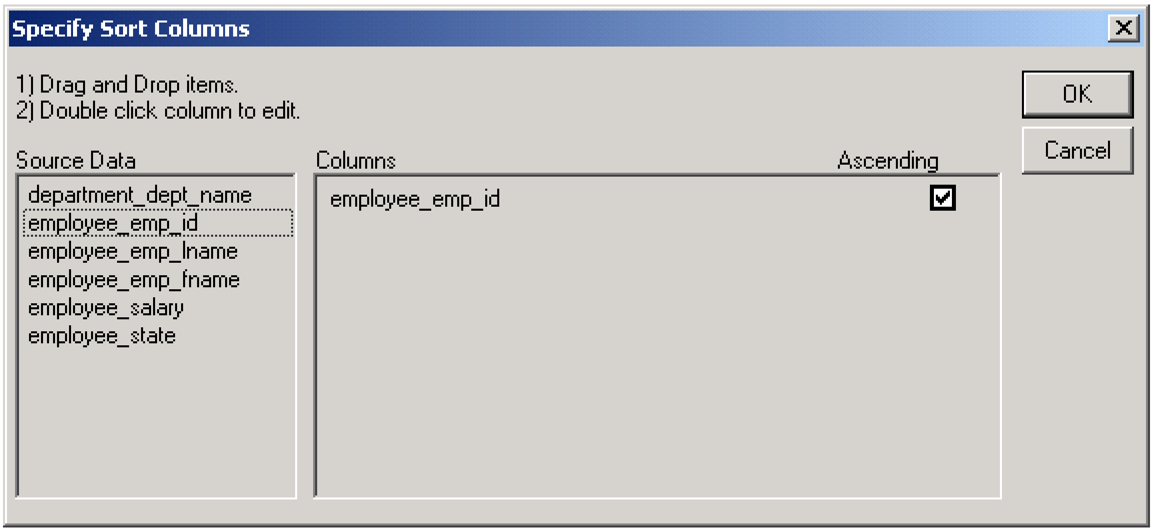
Figure 8-13: Sort



Department Name	State	Employee ID	Employee Last Name	Emplo First N.
Sales	CA	1142	Clark	Alison
Sales	MA	1162	Goggin	Kevin
Sales		46	Yeung	Caroline
Sales		596	Pickett	Catherine
Shipping		91	Bertrand	Jeannette
Shipping		03	Martinez	Jose
Shipping		50	Braun	Jane
Shipping		68	Kuo	Felicia
Shipping	MA	21	Crowley	Charles
Shipping	MA	1013	Barker	Joseph
Shipping	MA	1570	Rebeiro	Anthony
Shipping	MA	1615	Romero	Sheila
Shipping	MA	1658	Lynch	Michael

Step 2 – Specify the column used to sort the data in the DataWindow by double clicking on the column name in the left list box. The column name appears on the right, as shown in Figure 8-14.

Figure 8-14: Specify sort column



1) Drag and Drop items.
2) Double click column to edit.

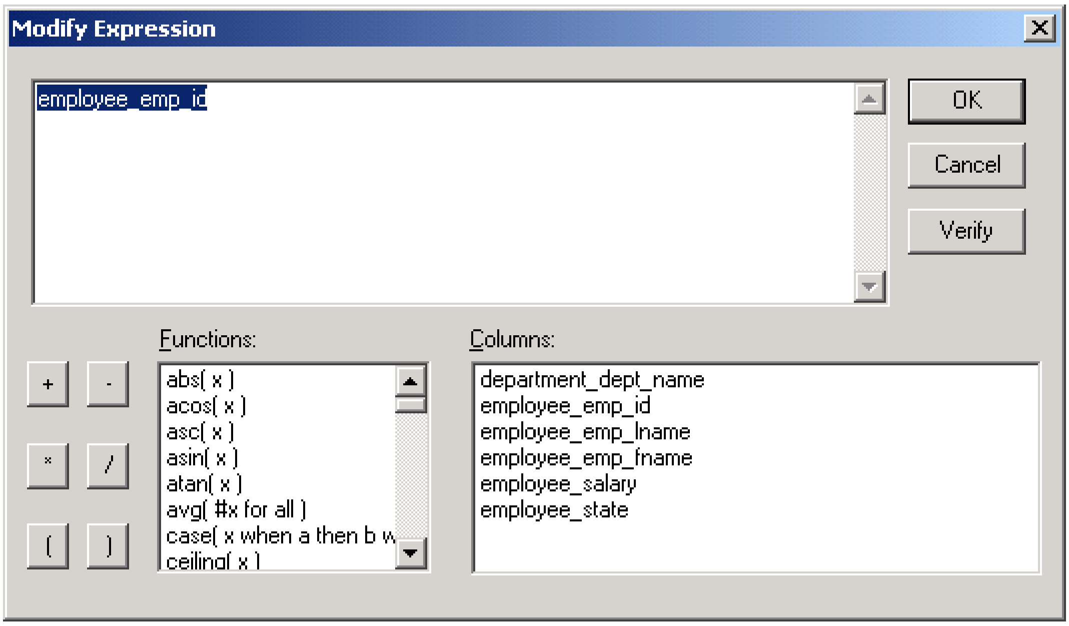
Source Data:
department_dept_name
employee_emp_id
employee_emp_lname
employee_emp_fname
employee_salary
employee_state

Columns: employee_emp_id ☑

Step 3 – Check the *Ascending* check box to sort the data in ascending order or uncheck the *Ascending* check box to sort the data in descending order. Click *OK* to accept the changes. The DataWindow automatically refreshes and the data is sorted.

If you want to edit the selected column, double click it to open the "Modify Expression" dialog, as shown in Figure 8-15.

Figure 8-15: Modify Expression dialog

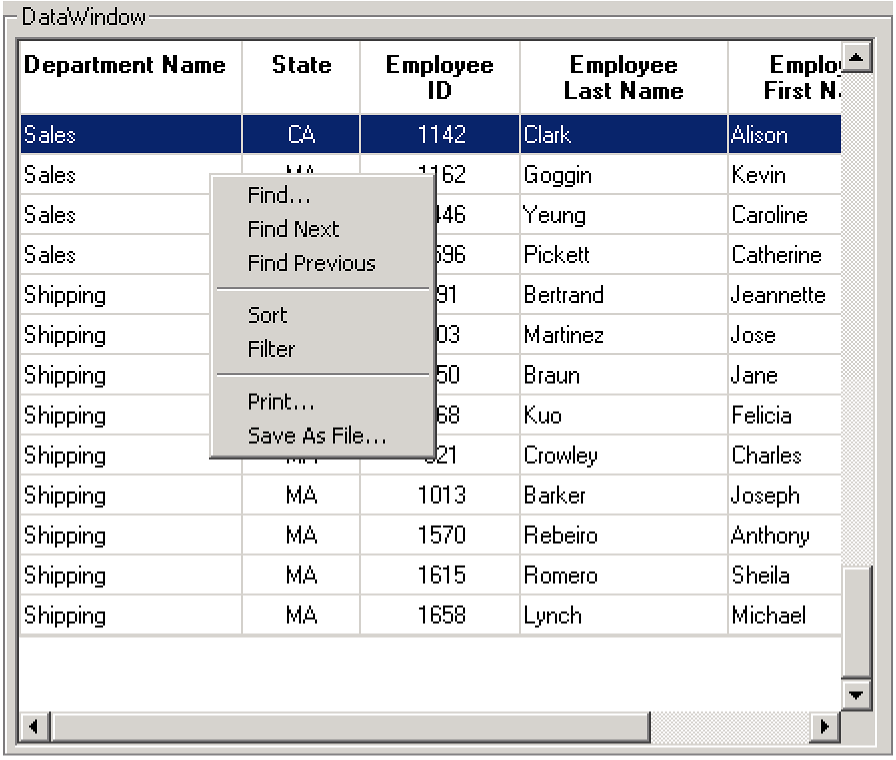


employee_emp_id

To filter data:

Step 1 – Right-click the Web DataWindow and select *Filter* from the popup menu, as shown in Figure 8-16.

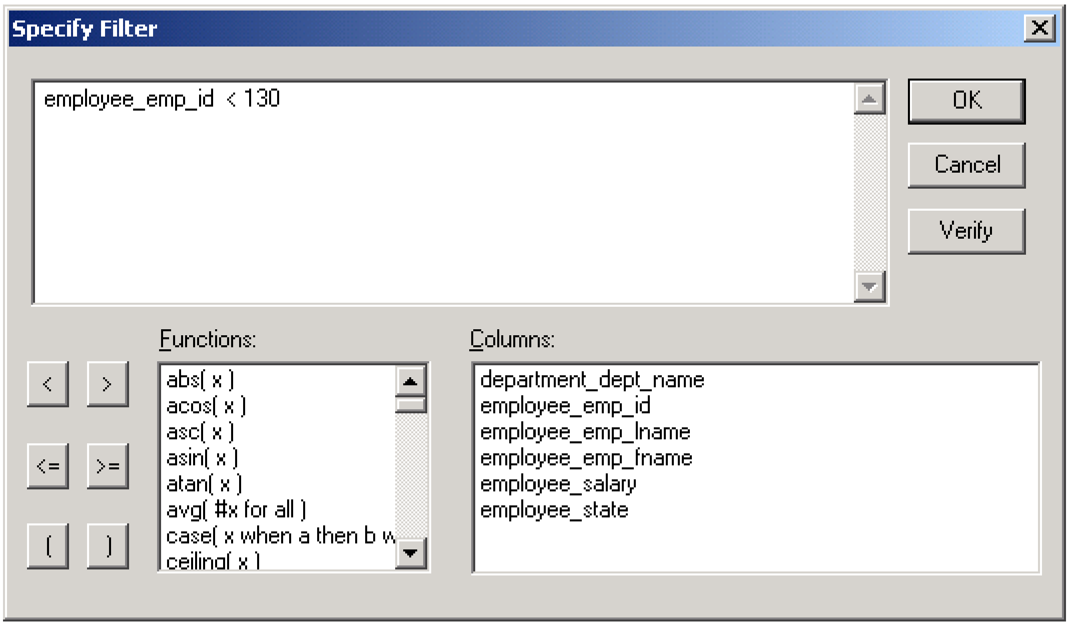
Figure 8-16: Filter



Department Name	State	Employee ID	Employee Last Name	Emplo First N
Sales	CA	1142	Clark	Alison
Sales	MA	1162	Goggin	Kevin
Sales		46	Yeung	Caroline
Sales		596	Pickett	Catherine
Shipping		91	Bertrand	Jeannette
Shipping		03	Martinez	Jose
Shipping		50	Braun	Jane
Shipping		68	Kuo	Felicia
Shipping	MA	21	Crowley	Charles
Shipping	MA	1013	Barker	Joseph
Shipping	MA	1570	Rebeiro	Anthony
Shipping	MA	1615	Romero	Sheila
Shipping	MA	1658	Lynch	Michael

Step 2 – Specify the filter by selecting the functions and columns. Click *Verify* to test whether the filter expression is valid. Click *OK* to accept the filter, as shown in Figure 8-17. The DataWindow automatically refreshes and the data is filtered.

Figure 8-17: Specify Filter



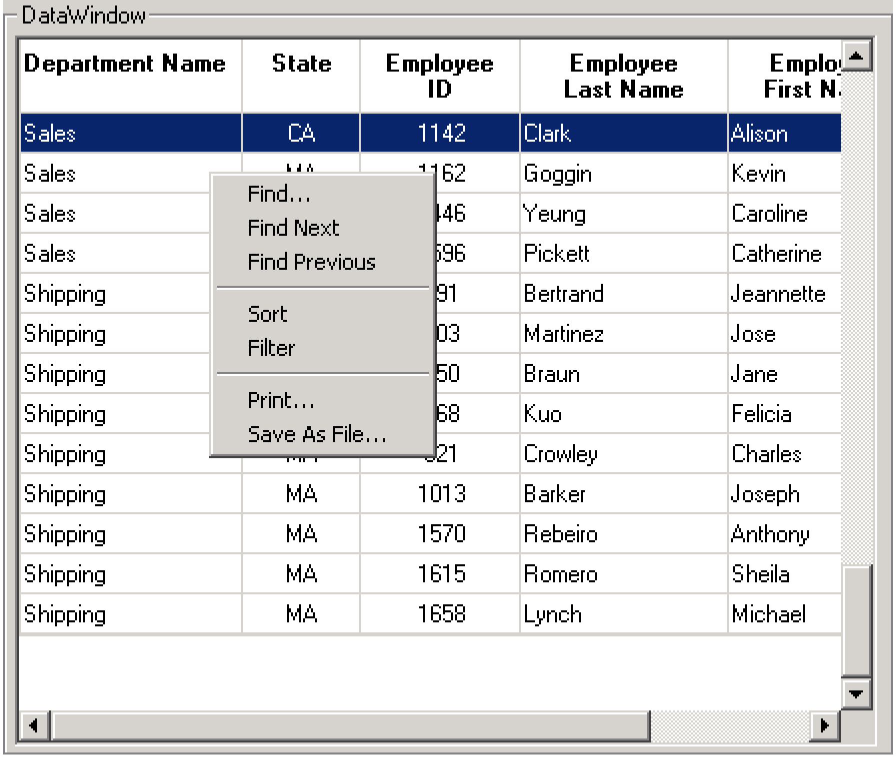
employee_emp_id < 130

8.4.2.c DataWindow printing

The Print menu provides user ability to print DataWindows directly on printers connected to the Client.

Step 1 – Right-click the DataWindow and select "Print" from the popup menu, as shown in Figure 8-18.

Figure 8-18: Print file with physical printers

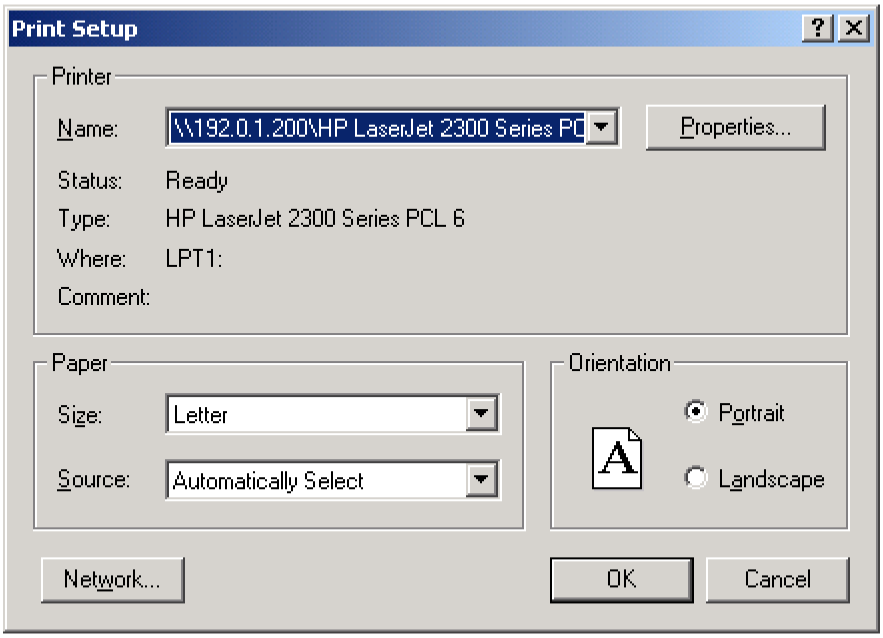


Department Name	State	Employee ID	Employee Last Name	Employ First N.
Sales	CA	1142	Clark	Alison
Sales	MA	1162	Goggin	Kevin
Sales		46	Yeung	Caroline
Sales		96	Pickett	Catherine
Shipping		91	Bertrand	Jeannette
Shipping		03	Martinez	Jose
Shipping		50	Braun	Jane
Shipping		68	Kuo	Felicia
Shipping	MA	21	Crowley	Charles
Shipping	MA	1013	Barker	Joseph
Shipping	MA	1570	Rebeiro	Anthony
Shipping	MA	1615	Romero	Sheila
Shipping	MA	1658	Lynch	Michael

Step 2 – Specify the printer settings and paper settings in the Print Setup window, as shown in Figure 8-19. Click *OK*.

The DataWindow will be printed directly to the local printer.

Figure 8-19: Print setup window



Print Setup

Printer

Name: \\192.0.1.200\HP LaserJet 2300 Series PC

Status: Ready

Type: HP LaserJet 2300 Series PCL 6

Where: LPT1:

Comment:

Paper

Size: Letter

Source: Automatically Select

Orientation

● Portrait

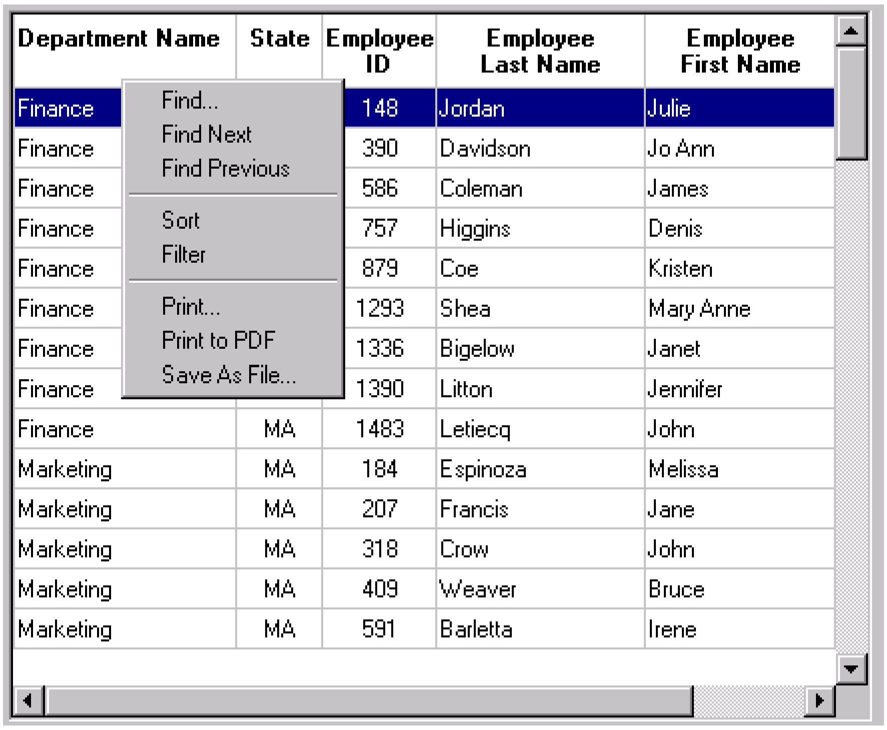
○ Landscape

8.4.2.d DataWindow SaveAs

The user can save the contents of a DataWindow in different file formats.

Step 1 – Right-click on a DataWindow and select “Save As File…” from the popup menu, as shown in Figure 8-20.

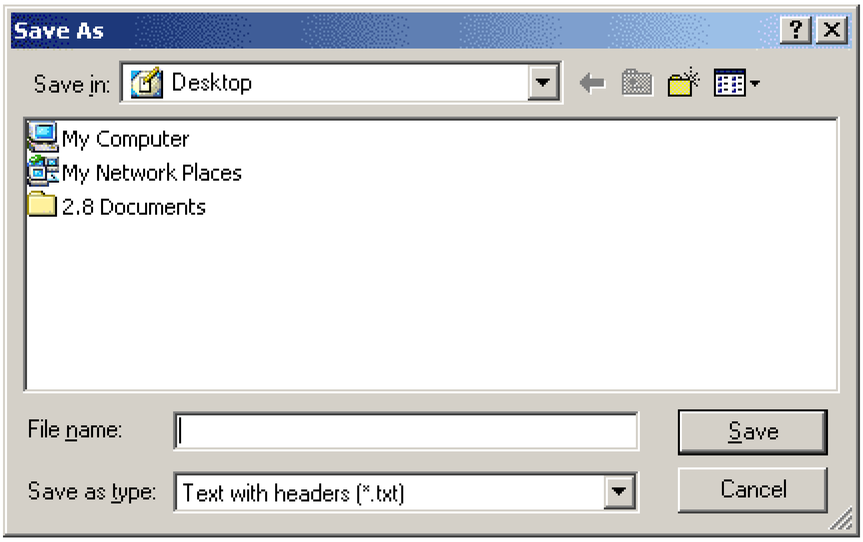
Figure 8-20: Save As File…



Department Name	State	Employee ID	Employee Last Name	Employee First Name
Finance		148	Jordan	Julie
Finance		390	Davidson	Jo Ann
Finance		586	Coleman	James
Finance		757	Higgins	Denis
Finance		879	Coe	Kristen
Finance		1293	Shea	Mary Anne
Finance		1336	Bigelow	Janet
Finance		1390	Litton	Jennifer
Finance	MA	1483	Letiecq	John
Marketing	MA	184	Espinoza	Melissa
Marketing	MA	207	Francis	Jane
Marketing	MA	318	Crow	John
Marketing	MA	409	Weaver	Bruce
Marketing	MA	591	Barletta	Irene

Step 2 – Specify the file location, file name and file type in the Save As window, as shown in Figure 8-21.

Figure 8-21: Save As window



Save As

Save in: Desktop

My Computer

My Network Places

2.8 Documents

File name:

Save as type: Text with headers (*.txt)

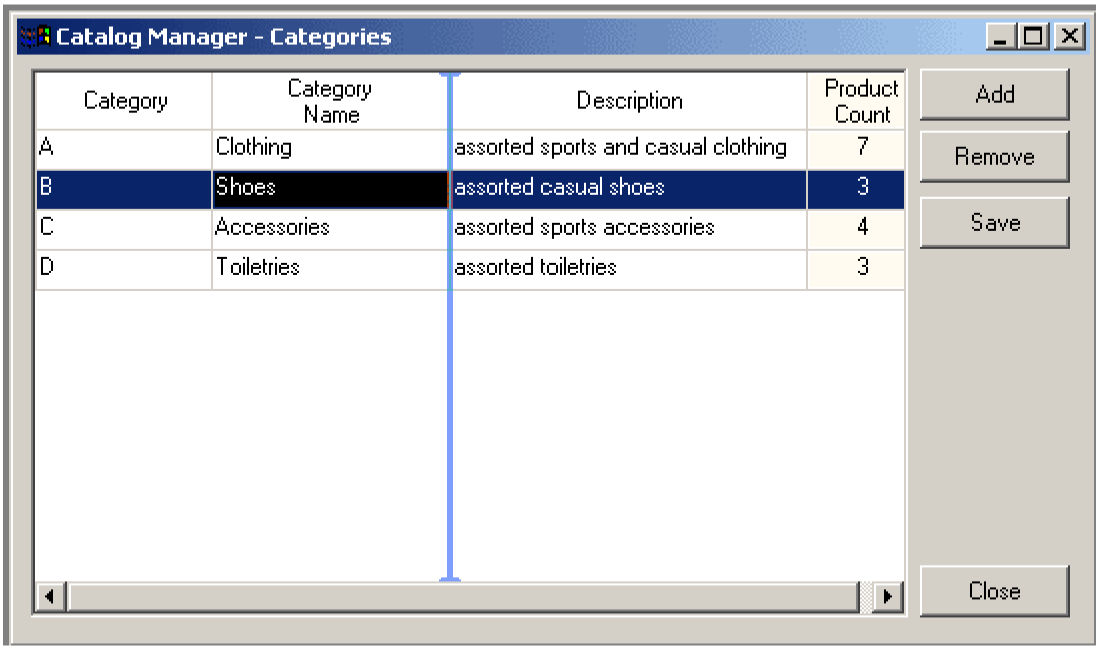
8.4.2.e Additional Enhanced Features

In addition to the popup menu (Find, Find Next, Find Previous, Sort, Filter, Print, Save As File), Appeon offers more features for users to manage DataWindows on the Web. Users can place DataWindow columns in a different order by dragging and dropping them to new positions. When navigating an editable DataWindow, users can use the popup menu to quickly edit the contents. These features are always available and require no extra coding to be enabled. The following steps show you how to access them:

Step 1 – Select a column header in the Web DataWindow and the selected location is highlighted, as shown in Figure 8-22.

Step 2 – Drag and drop it to the desired location, and the location to be dropped is highlighted as well.

Figure 8-22: Select a column header



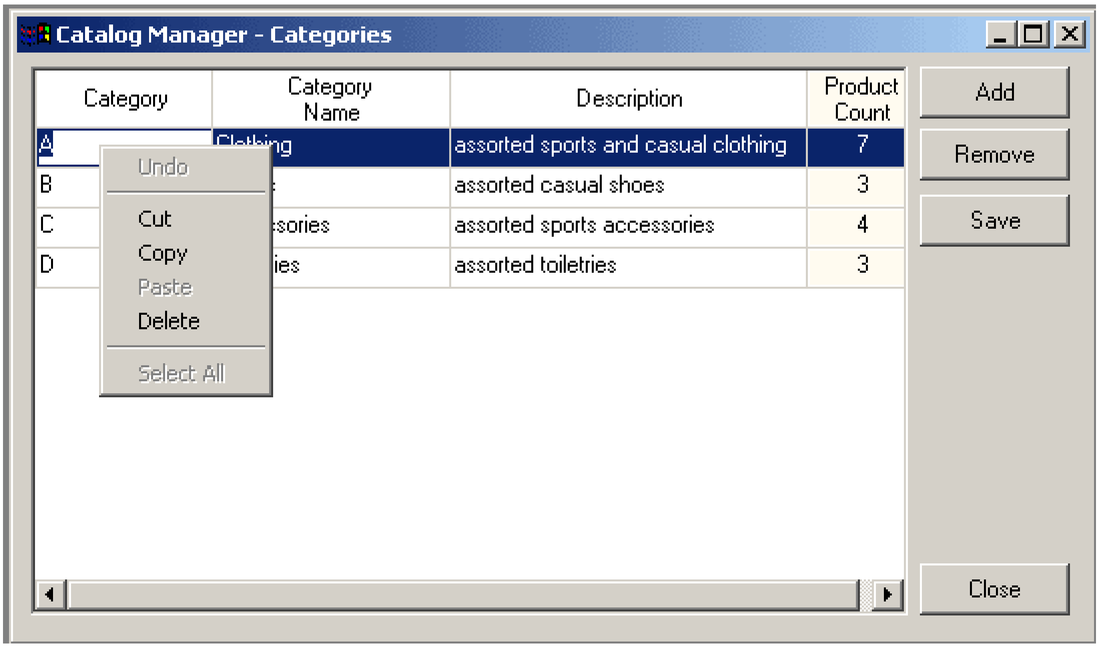
Catalog Manager - Categories

Category	Category Name	Description	Product Count
A	Clothing	assorted sports and casual clothing	7
B	Shoes	assorted casual shoes	3
C	Accessories	assorted sports accessories	4
D	Toiletries	assorted toiletries	3

Step 3 – Right click on an item in an editable DataWindow and a menu pops up, as shown in Figure 8-23.

The user can select the following menu items from the popup menu to perform quick editing: Undo, Cut, Copy, Paste, Delete, Select All.

Figure 8-23: Popup menu for quick edition



Category	Category Name	Description	Product Count
A	Clothing	assorted sports and casual clothing	7
B		assorted casual shoes	3
C	sories	assorted sports accessories	4
D	ies	assorted toiletries	3

9 Using Information Manager

9.1 Overview

Information Manager provides easy access to all the available reports or log files generated during the application migration process, including debug, development, and deployment. The reports are categorized as follows: Analysis and Performance. The log files are categorized as follows: Deploy Log and Analysis Log.

Table 9-1 provides a brief description of all the reports generated during the Web migration process and the report types they are associated with.

Table 9-1: Report Description

Report Type	Description
Analysis	**Analysis Report** is generated during a feature analysis or application deployment, and records the unsupported features in the application. After you click this button, the UFA Report Window will be displayed. For detailed instructions, refer to Section 5.3: *Working with UFA Report*.
Performance	**Heavy Window Report** is generated during a feature analysis or application deployment, and records the complex windows in the application. For detailed instructions, refer to Section 5.4: *Working with Heavy Window Report.*

Table 9-2 provides a brief description of all the log files generated during the Web migration process, and the log types they are associated with.

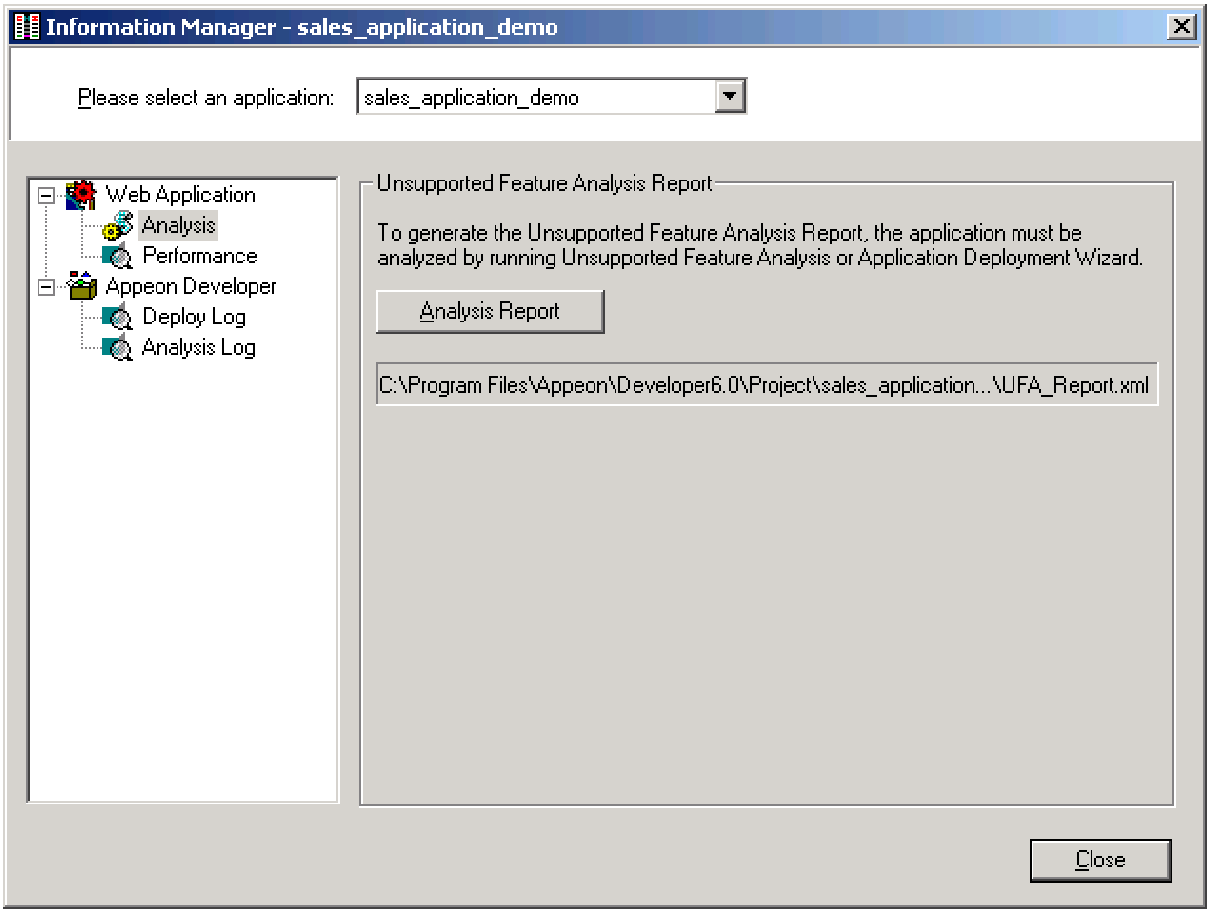
Table 9-2: Log Type Description

Log Type	Description
Deploy Log	**Deploy Log** is generated during application deployment, and records the deployment process status, such as errors and warnings. All log files are named according to the following format: Log Type (i.e. “DeployWizardLog”, or “FeatureAnalysisLog”) + time when the log file is generated (yyyy/mm/dd/hh/mm/ss) + “xml”. For example, DeployWizardLog_20050220101918.xml.
Analysis Log	**Analysis Log** is generated during the feature analysis process, and records the analyzing status.

9.2 Viewing reports and logs

Step 1 – Click the *Information* button (▦) on the Appeon Developer toolbar and the Information Manager window is displayed, as shown in Figure 9-1.

Figure 9-1: Information Manager



Information Manager - sales_application_demo
Please select an application: sales_application_demo
Web Application
Analysis
Performance
Appeon Developer
Deploy Log
Analysis Log
Unsupported Feature Analysis Report
Analysis Report
Close

To generate the Unsupported Feature Analysis Report, the application must be analyzed by running Unsupported Feature Analysis or Application Deployment Wizard.

C:\Program Files\Appeon\Developer6.0\Project\sales_application...\UFA_Report.xml

Step 2 – Select the application in the top dropdown listbox.

Step 3 – Select a report type or log type in the left box and the corresponding report button or log files are displayed in the right box.

If there is more than one log file generated, the log files are listed in order according to the date and time of generation; the latest log will be listed at the top.

You can open one log at a time, or open several logs at one time: select a continuous range of log files by holding down Shift and click log files, or select a disconnected range of log files by holding down Ctrl and clicking log files. Then click *View* to open the selected log files.

The *View* and *Delete* buttons will be disabled if no log files were generated.

Note: For Deploy Log, you can click View All to view all information such as errors, warnings, and process description, or click View Warning to view warnings only, or click View Errors to view errors only.

Step 4 – Click the report button or log file to view the report or log.

You can continue with other operations while the Information Manager window opens. For example, you can perform feature analysis, deploy the application, or run the Web application. If any new reports or logs are generated when the Information Manager window is still open, the new reports or logs are not reflected in the Information Manager immediately. You must refresh Information Manager in either of the following ways:

- Close the open Information Manager window and open it again, or

- Select a different application and then select the original application again in the open Information Manager window.

10 Packaging and Installing Web Applications

10.1 Overview

The Appeon Web Application Package Wizard helps PowerBuilder developers to generate portable installation "packages" for their Appeon deployed Web applications. With the installation package, the user can directly install the Web application to servers without installing Appeon Developer and using it to do Web deployments; this will save time. The PowerBuilder developers do not need to provide the source PowerBuilder application to the user; this protects the author's intellectual property rights.

DO NOT try to copy the generated Web files from one Appeon Server or Web server to another Appeon Server or Web server simply by copying and pasting the files. This copy and paste method cannot guarantee that the Web application will work. If you need to distribute an Appeon Web application to multiple Appeon Servers and Web servers located in different network areas, instead of installing Appeon Developer in each area and performing Web deployments, Appeon recommends that you:

1. Deploy the target PowerBuilder application using the Appeon Deployment Wizard (See Chapter 6: *Deploying PowerBuilder Applications*) and
2. Pack all the application files using the Appeon Web Application Package Wizard (See Section 10.2: *Packaging a Web application*), then
3. Copy the installation package to a machine running Windows and run the installation package to install the application to any number of target servers located in the same network area. (See Section 10.3: *Installing and uninstalling a Web application*). Note that the generated installation packages run on the Windows platform only.

   You can customize the installation/uninstallation wizard provided by Appeon to suit your needs. (See the *Web_App_Install_Help.chm* help file under the *help* folder of the generated package.)

10.2 Packaging a Web application

10.2.1 What can be packaged?

The package wizard allows you to pack the following files and settings:

- **Mandatory** Web application files

  Include all Web files, DataWindow syntax, INI files, image files, DLL/OCX files etc.

- **Mandatory** application profile, Appeon Server profile, and Web server profile settings

  Include the profile name, Web URL, PowerBuilder version, file encryption, performance settings, server type, Appeon Server version, Appeon Server and Web server connection settings, AEM, connection caches etc. All these settings will be saved to the configuration files in the INI folder of the generated package. Of these configuration files, you can modify the config.xml file using the config.exe tool. For detailed information, refer to the *Modifying the deploy-config file* section.

- Optional EAServer components

When the Appeon Server is of type EAServer, you can package the local or remote EAServer components used by the application. Note that EAServer components work with EAServer application server only.

10.2.2 Important requirements

Before using the Appeon Web Application Package Wizard, verify that

1. The target PowerBuilder application has been deployed as required:

   You must have performed a **full deployment** on the target PowerBuilder application on the **same Developer machine** where you will use the package wizard to pack this application later.

2. The deployed application can run correctly.

10.2.3 Packaging instructions

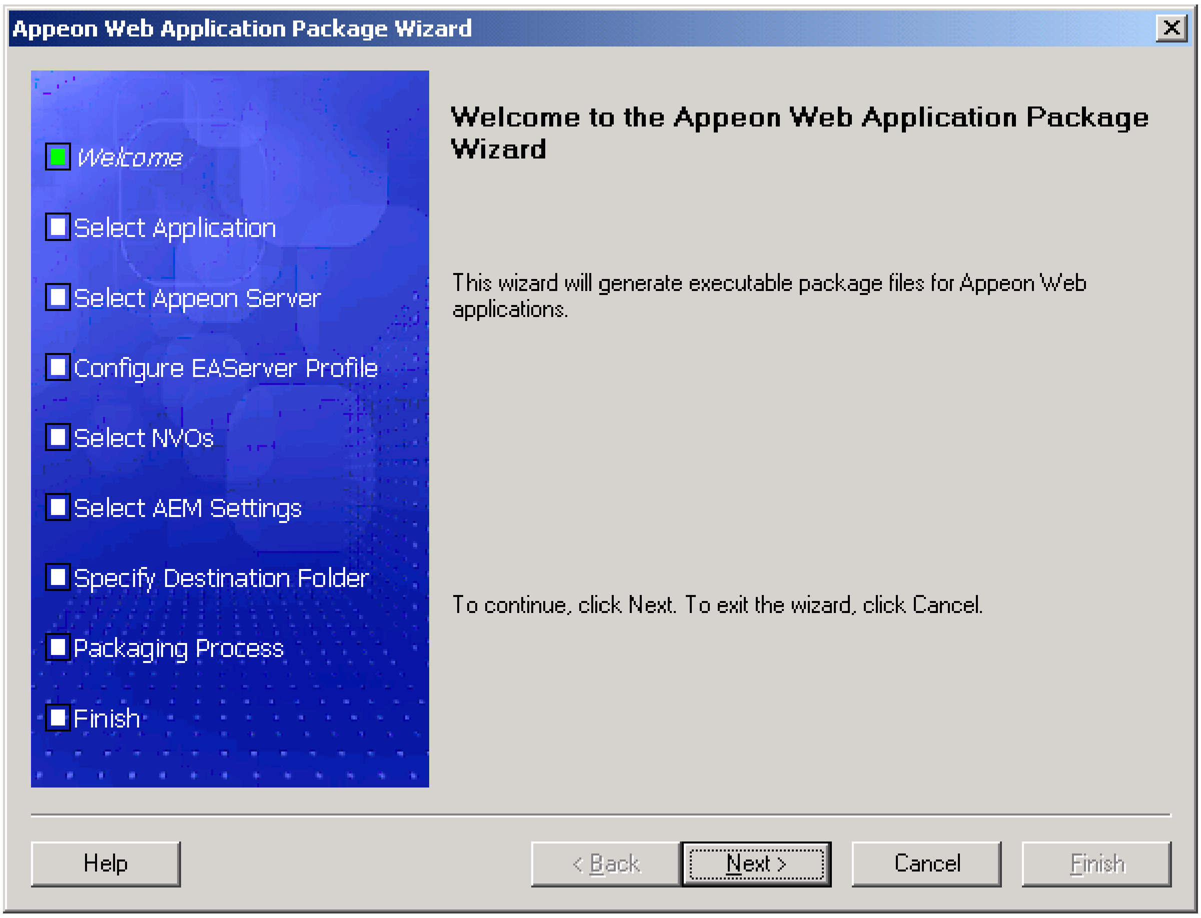
To package a Web application and the related settings, take the following steps:

Step 1 – Start the servers, as the package wizard will pack settings of Appeon Server, AEM and NVOs.

Step 2 – Click the *Package* (💾) button on the Appeon Developer toolbar.

The Appeon Web Application Package Wizard is displayed, as shown in Figure 10-1. Click *Next* to proceed.

Figure 10-1: Welcome page



Welcome to the Appeon Web Application Package Wizard

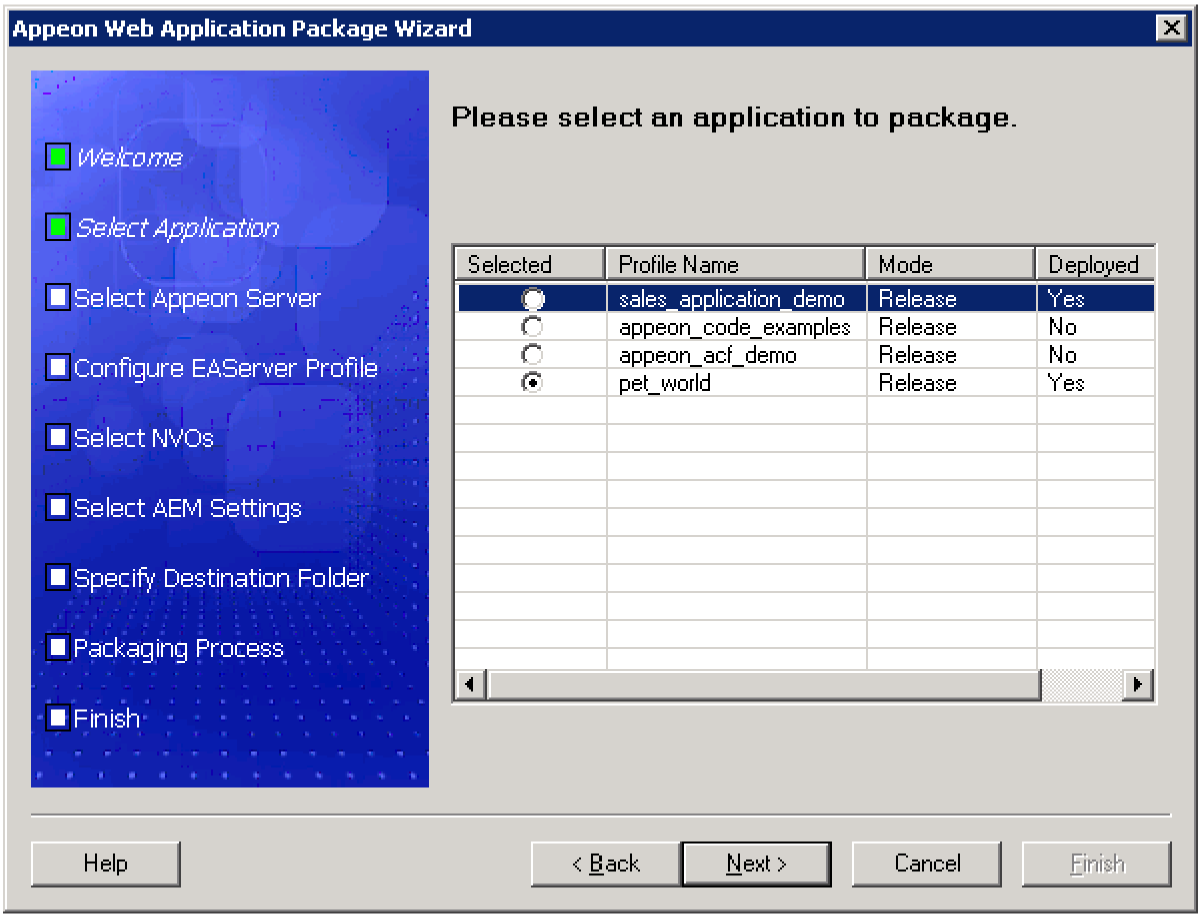
This wizard will generate executable package files for Appeon Web applications.

To continue, click Next. To exit the wizard, click Cancel.

Step 3 – Select the profile of the application that you want to package (Figure 10-2).

Select the profile that is identified with "Yes" in the Deployed column and click *Next*.

Figure 10-2: Select an application



Please select an application to package.

Selected	Profile Name	Mode	Deployed
○	sales_application_demo	Release	Yes
○	appeon_code_examples	Release	No
○	appeon_acf_demo	Release	No
●	pet_world	Release	Yes

Table 10-1 gives a brief introduction of the columns:

Table 10-1: Web Application Package Wizard

Column	Description
Selected	Identical to the default application profile selected in the Appeon Developer Configuration window.
Profile Name	All the application profiles configured in the Appeon Developer Configuration window are listed in the order that the application profiles are created.
Mode	Identical to the current mode the application profile is set to in the Appeon Developer Configuration window. Note that as the package wizard indicates, packaged applications should be available in **Release** mode. This mode prevents most forms of reverse engineering, and like application packaging, it protects the author's intellectual property.
Deployed	Indicates whether the application has been deployed using the local Appeon Developer. Only locally deployed applications can be packaged.

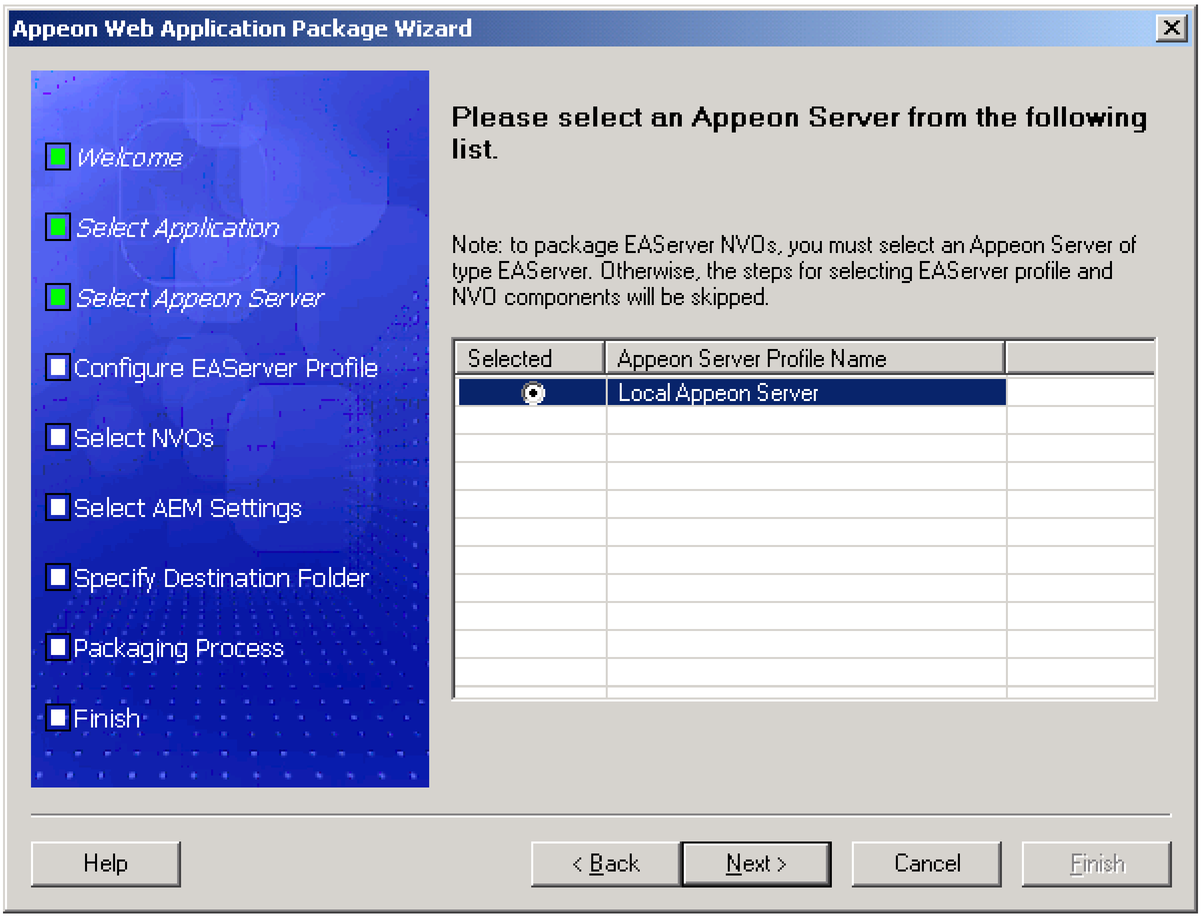
Step 4 – Select the Appeon Server whose settings will be packaged and click *Next*.

The Appeon Server settings, such as the server type and the Appeon Server version, will be saved to the config.xml file in the INI folder of the generated package. If necessary, modify this file using the config.exe tool after the packaging is complete (refer to *Modifying the deploy-config file*).

All the configured Appeon Servers selected into the default deployment profile in the Appeon Developer Configuration window are listed, as shown in Figure 10-3.

The selected Appeon Server determines that its AEM settings will be packaged and that the packaged application can only be installed to Appeon Servers of the same type and version.

Figure 10-3: Select an Appeon Server



Appeon Web Application Package Wizard

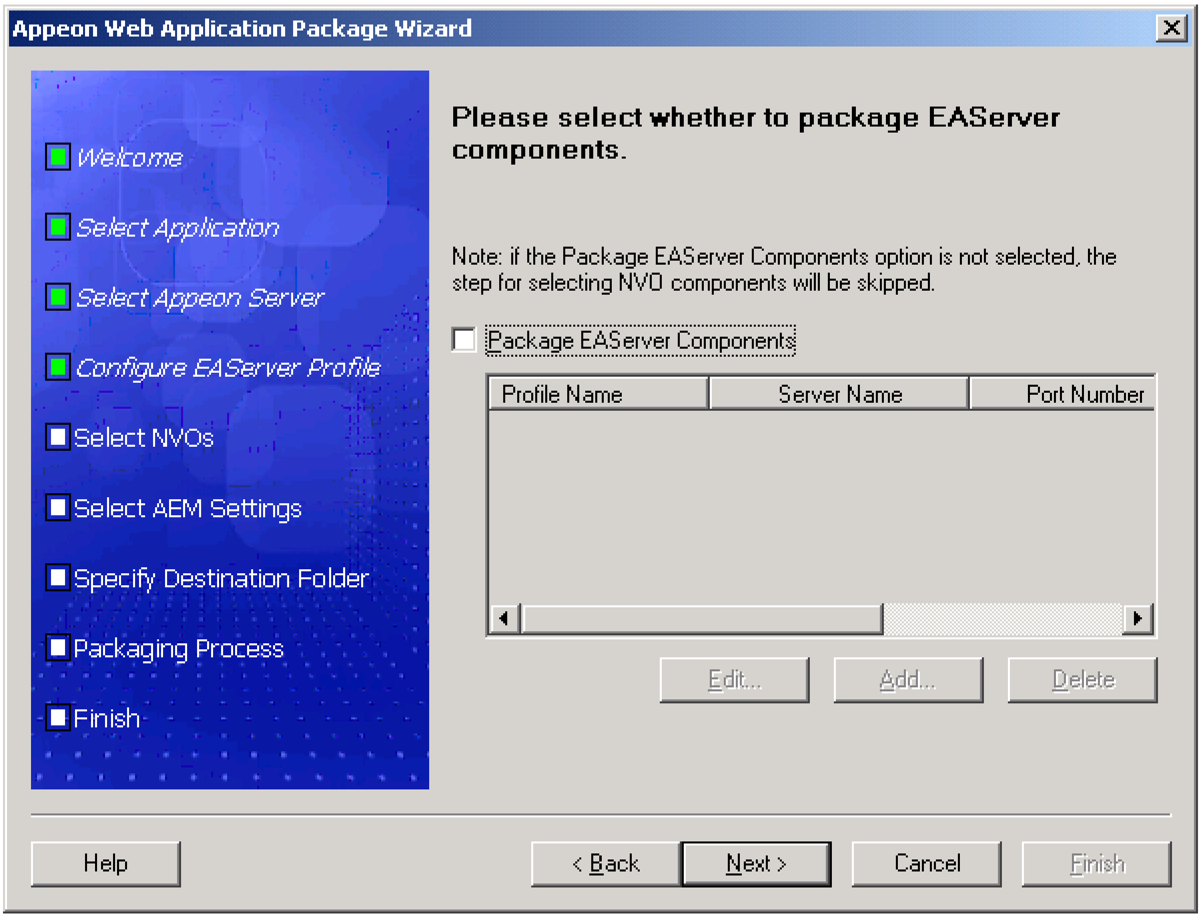
Please select an Appeon Server from the following list.

Note: to package EAServer NVOs, you must select an Appeon Server of type EAServer. Otherwise, the steps for selecting EAServer profile and NVO components will be skipped.

Selected	Appeon Server Profile Name	
●	Local Appeon Server	

Step 5 – If the selected Appeon Server is of type EAServer, select whether to package EAServer components, as shown in Figure 10-4.

Figure 10-4: Configure EAServer profiles



Appeon Web Application Package Wizard

Please select whether to package EAServer components.

Note: if the Package EAServer Components option is not selected, the step for selecting NVO components will be skipped.

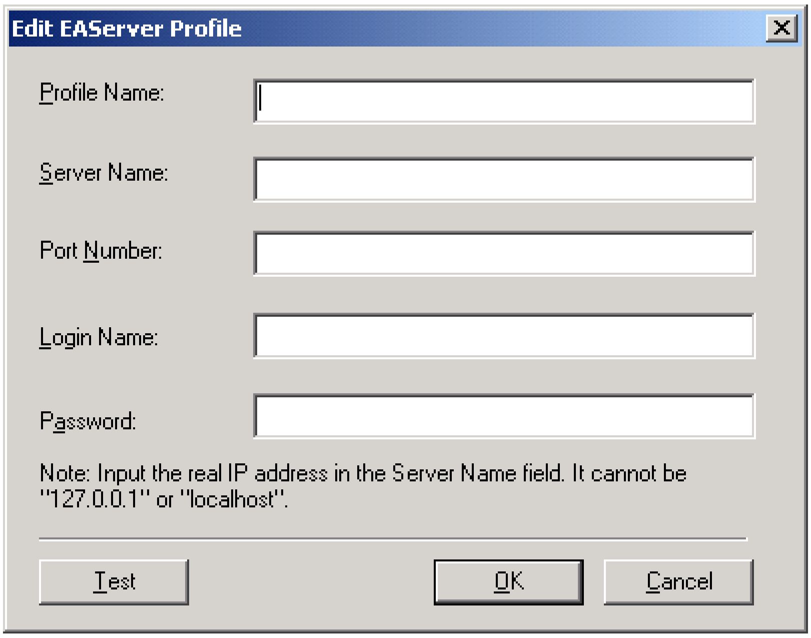
☐ Package EAServer Components

Profile Name	Server Name	Port Number

If EAServer components are used in the application, select the *Package EAServer Components* option and then create connection profiles for EAServers which host the components. Follow instructions below to create EAServer profiles:

- Click *Add* to create an EAServer profile.
- In the Edit EAServer Profile dialog box (Figure 10-5), input the following connection parameters: profile name, host name or IP address, port number, user name and password.
- Click *Test* to ensure that the connection is successful.
- Click *OK* to save the new EAServer profile.

Figure 10-5: Edit EAServer profile



Edit EAServer Profile

Profile Name:

Server Name:

Port Number:

Login Name:

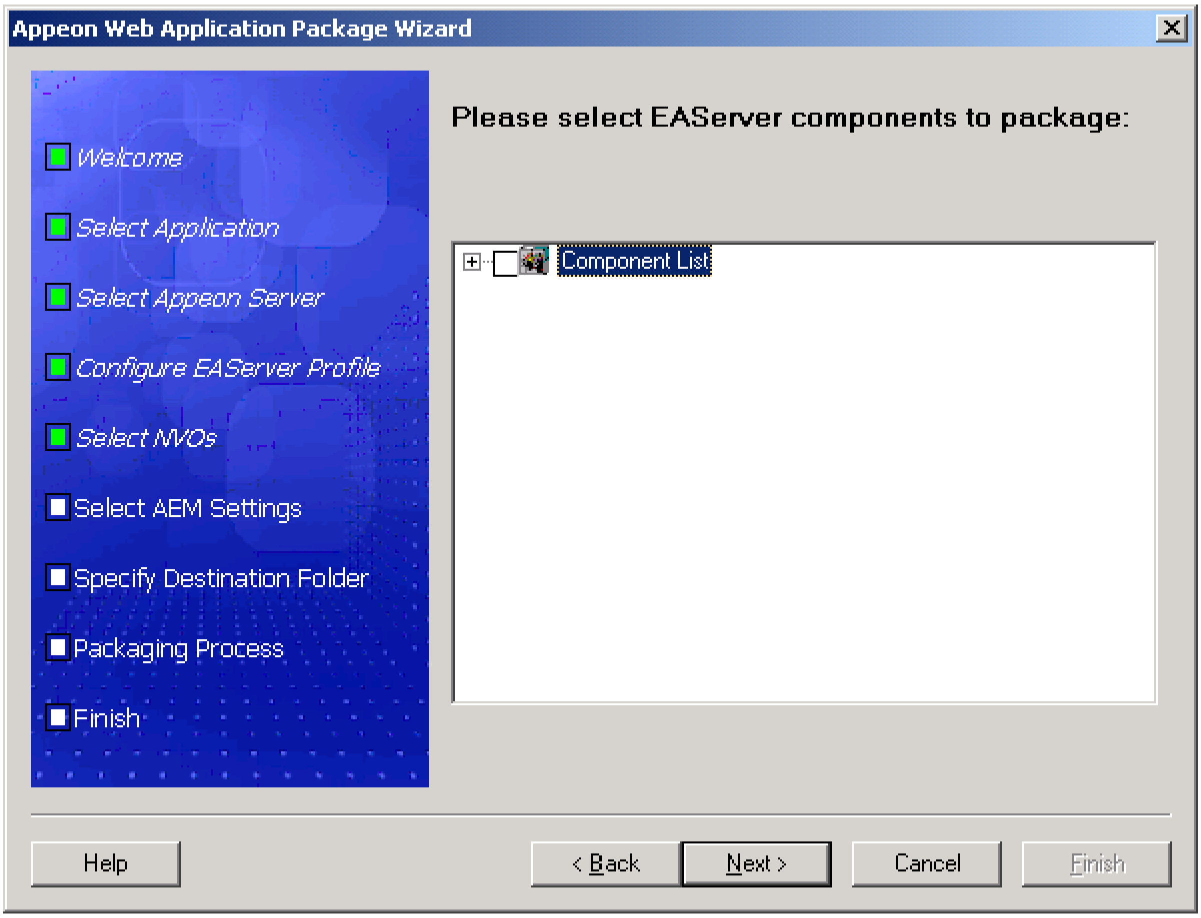
Password:

Note: Input the real IP address in the Server Name field. It cannot be "127.0.0.1" or "localhost".

Step 6 – If the *Package EAServer Components* option is selected in the previous step, select which components to package and click *Next*, as shown in Figure 10-6.

The selected NVOs will be packaged as *ComponentName*.jar and saved under the /NVO/*server profile name/package name* folder.

Figure 10-6: Select NVOs



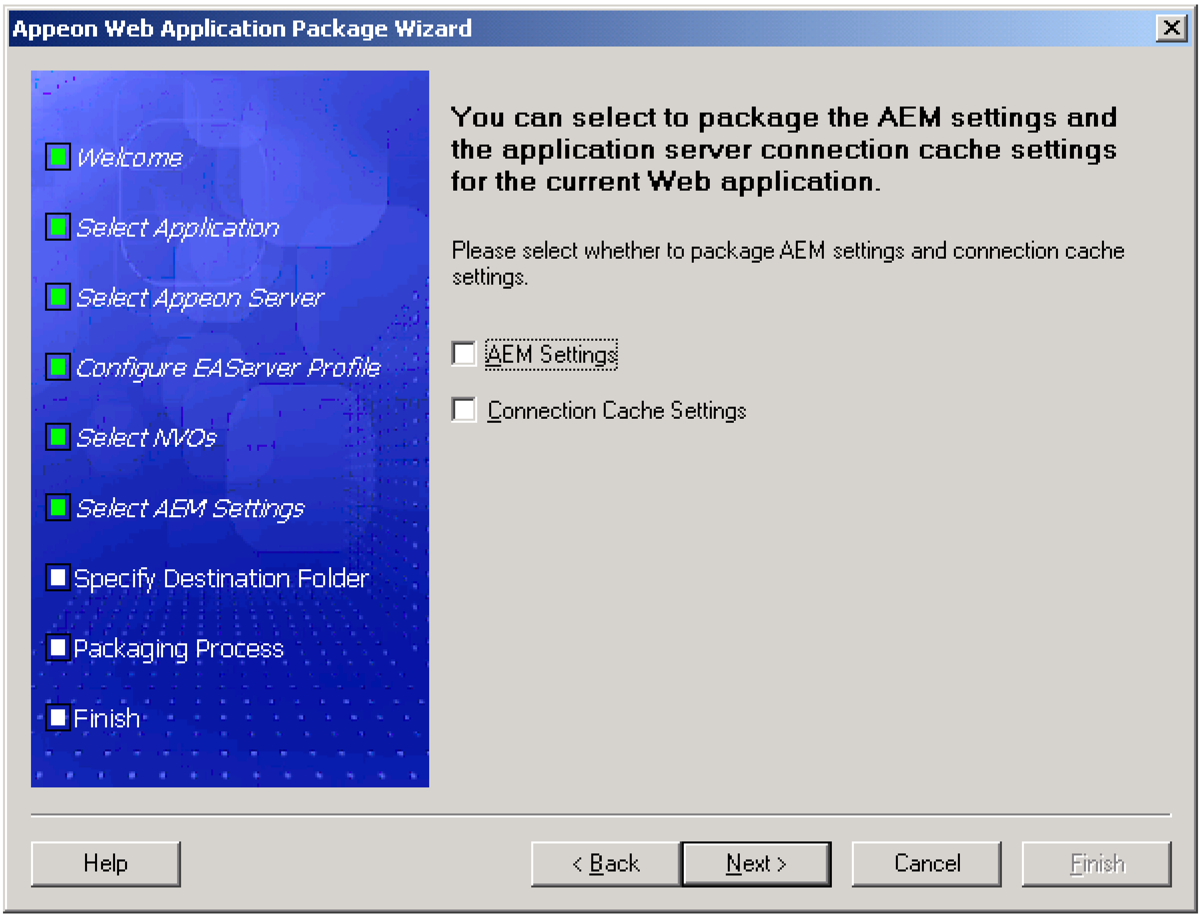
Appeon Web Application Package Wizard

Please select EASeerver components to package:

Step 7 – Select whether AEM settings and connection caches for the current application will be packaged and click *Next*, as shown in Figure 10-7.

The AEM settings will be saved in the AEM.xml file and connection caches will be saved in the config.xml file.

Figure 10-7: Select AEM settings and connection caches



Appeon Web Application Package Wizard

You can select to package the AEM settings and the application server connection cache settings for the current Web application.

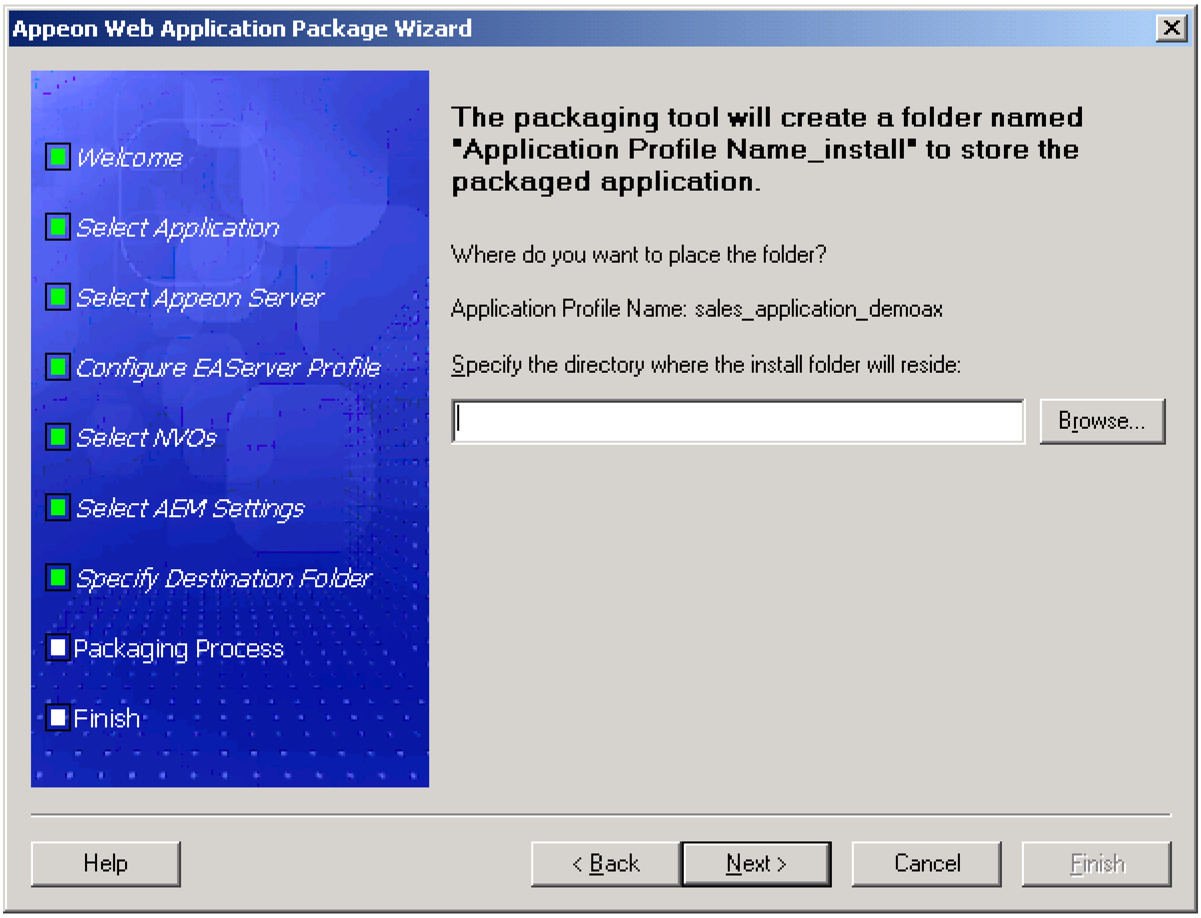
Please select whether to package AEM settings and connection cache settings.

☐ AEM Settings

☐ Connection Cache Settings

Step 8 – Specify the destination folder for the installation package and click *Next*, as shown in Figure 10-8.

Figure 10-8: Specify destination location for the package



The packaging tool will create a folder named "Application Profile Name_install" to store the packaged application.

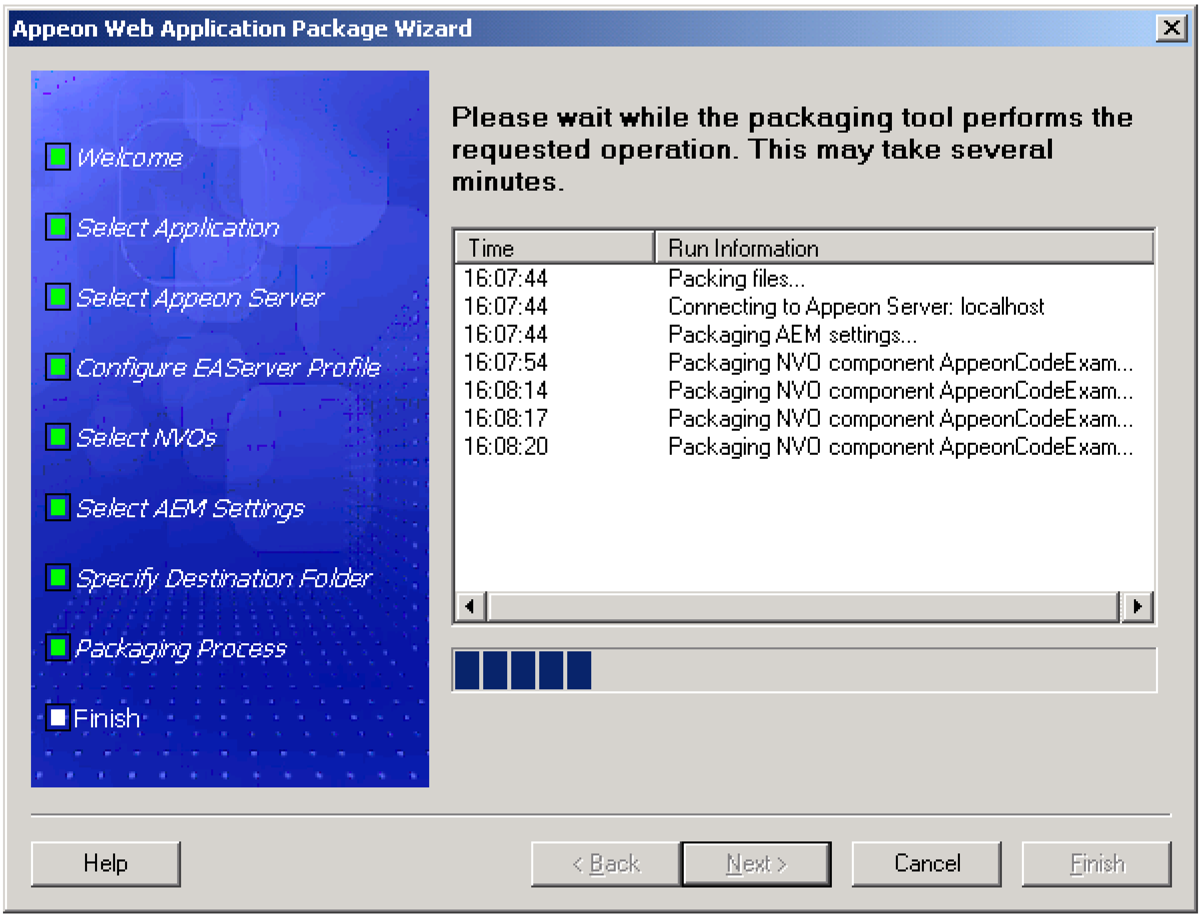
Where do you want to place the folder?

Application Profile Name: sales_application_demoax

Specify the directory where the install folder will reside:

Step 9 – Wait while the wizard is packing files and settings, as shown in Figure 10-9.

Figure 10-9: Web application packaging status window



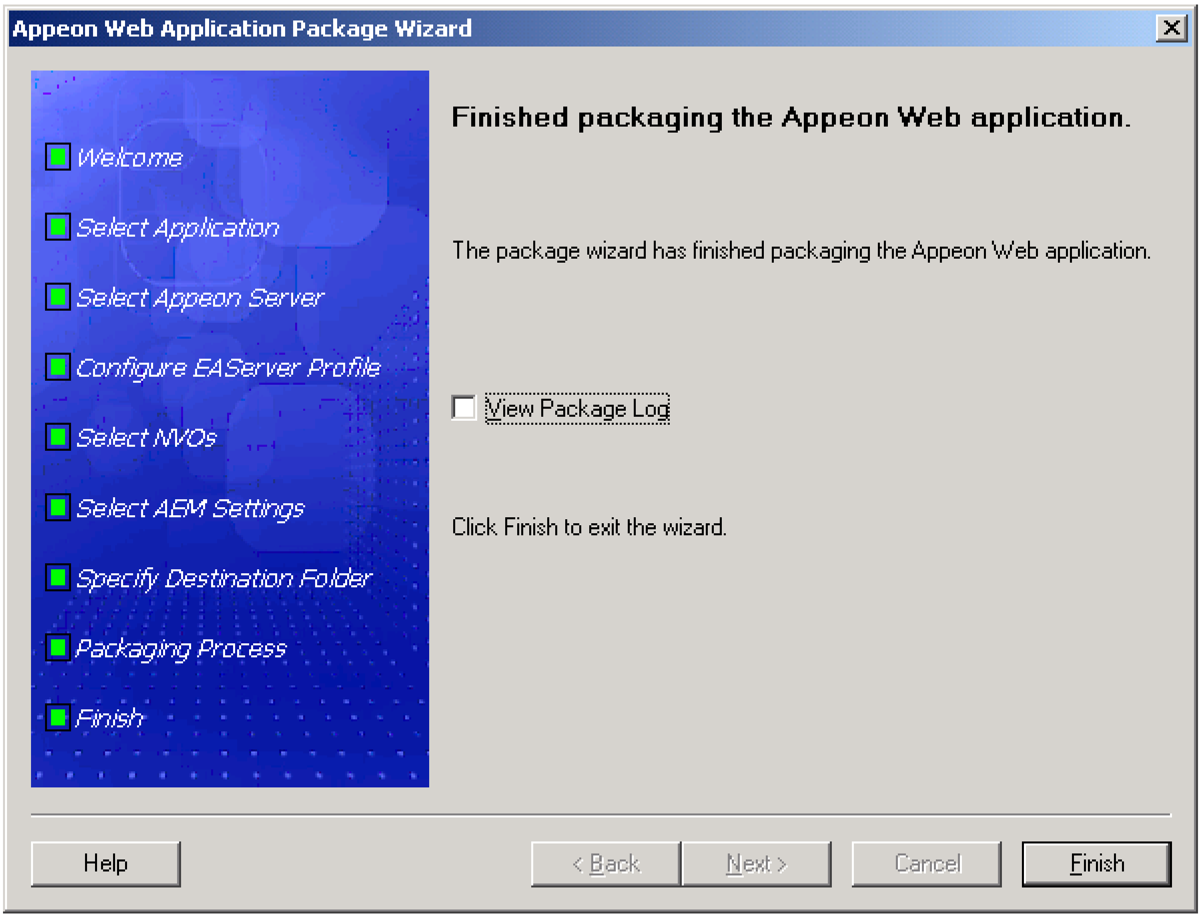
Please wait while the packaging tool performs the requested operation. This may take several minutes.

Time	Run Information
16:07:44	Packing files...
16:07:44	Connecting to Appeon Server: localhost
16:07:44	Packaging AEM settings...
16:07:54	Packaging NVO component AppeonCodeExam...
16:08:14	Packaging NVO component AppeonCodeExam...
16:08:17	Packaging NVO component AppeonCodeExam...
16:08:20	Packaging NVO component AppeonCodeExam...

Step 10 – Once the operation is complete, click *Finish* to exit the package wizard (Figure 10-10).

To view the log information generated during the packaging process, select the *View Package Log* box and then click *Finish*. The log file will be displayed.

Figure 10-10: Web application packaging finished



Finished packaging the Appeon Web application.

The package wizard has finished packaging the Appeon Web application.

☐ View Package Log

Click Finish to exit the wizard.

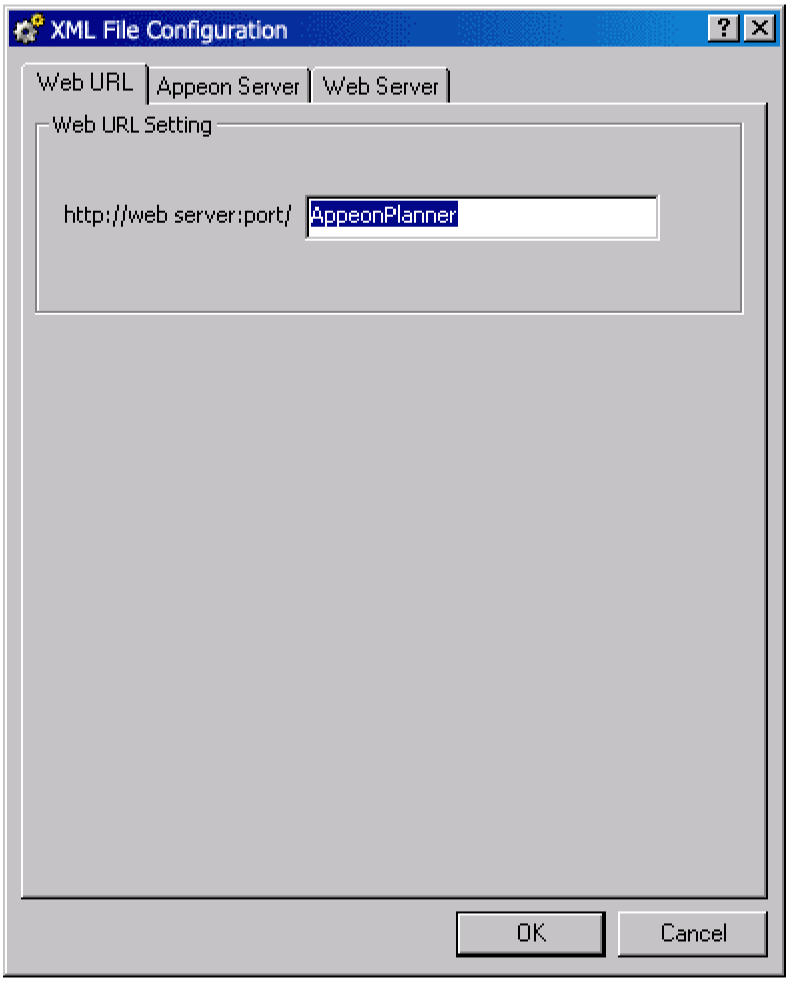
After the application is successfully packaged, all the necessary application files will be packaged and generated in the specified destination folder. You can install the application to any Appeon Servers and Web Servers.

10.2.4 Modifying the deploy-config file

The deploy-config file (config.xml) is created during the packaging process and includes the default settings of the application: application profile, Appeon Server profile, Web server profile, EAServer profile, EAServer components, and connection caches. You can modify the config.xml file using the config.exe tool to have identical settings of the target environment, so that the user can run the installation very conveniently without needing to specify them.

When you run the config.exe file in the generated package, an XML File Configuration dialog is displayed and loads the Web URL setting, Appeon Server settings, and Web server settings from the config.xml file. You can modify these settings on the corresponding tab pages and then click OK to save them to the Config.xml file, as shown in Figure 10-11.

Figure 10-11: Configuring the XML file



http://web server:port/ AppeonPlanner

10.3 Installing and uninstalling a Web application

10.3.1 Points to note before installation

Be aware of the following points before installing an Appeon application:

- The Setup program of the Web application runs on the Windows platform only. To install applications to servers running on Unix\Linux, you must run the Setup program on the Windows platform and then install the application to remote servers running on Unix\Linux.
- The target Web servers must have the Appeon Server Web Component correctly installed.
- The packaged EAServer components, if any, can only be installed to EAServer application server.

10.3.2 Installing an application

You can copy the generated package to any computer running Windows and install the application to any number of Appeon Servers and Web servers. Installing an application using the generated package is the same as deploying an application using the Appeon Deployment Wizard.

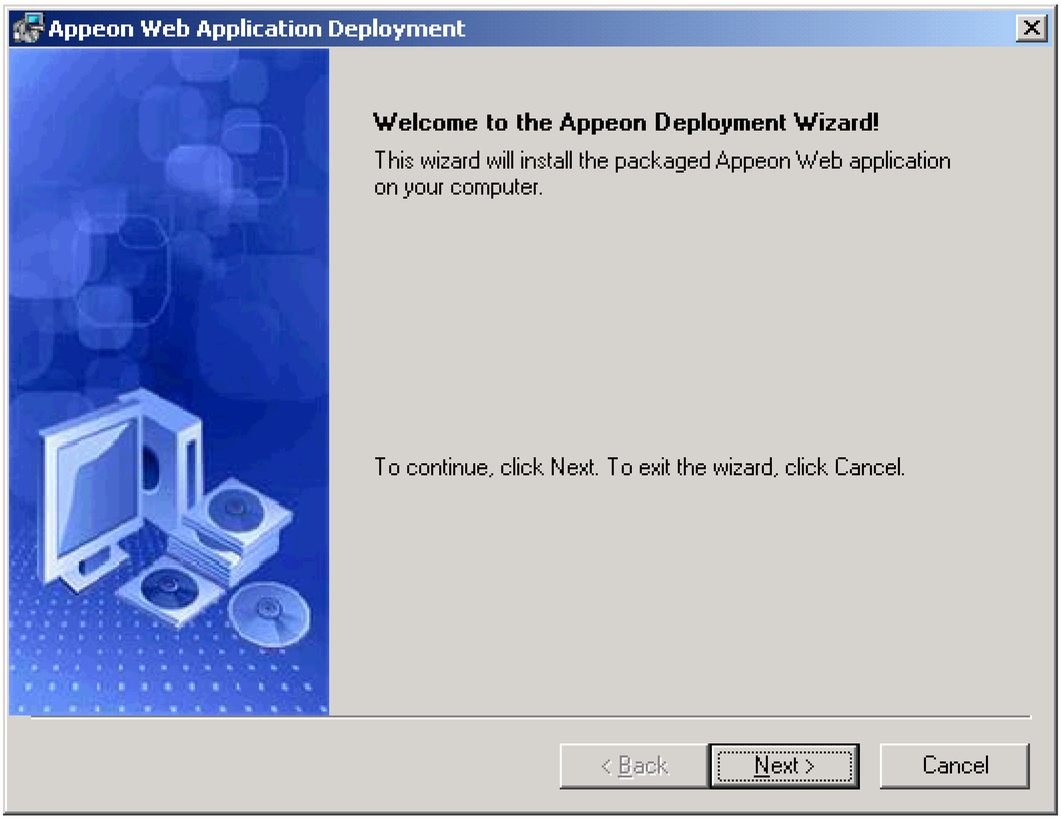
Step 1 – Start the target Appeon Server(s) and Web server(s) where you want to install the application.

Verify that the target Web servers already have Appeon Server Web Component installed.

Step 2 – Run the Setup.exe file in the generated package.

The Appeon Web Application Deployment wizard is displayed, as shown in Figure 10-12. Click *Next* to proceed.

Figure 10-12: Welcome page



Welcome to the Appeon Deployment Wizard!

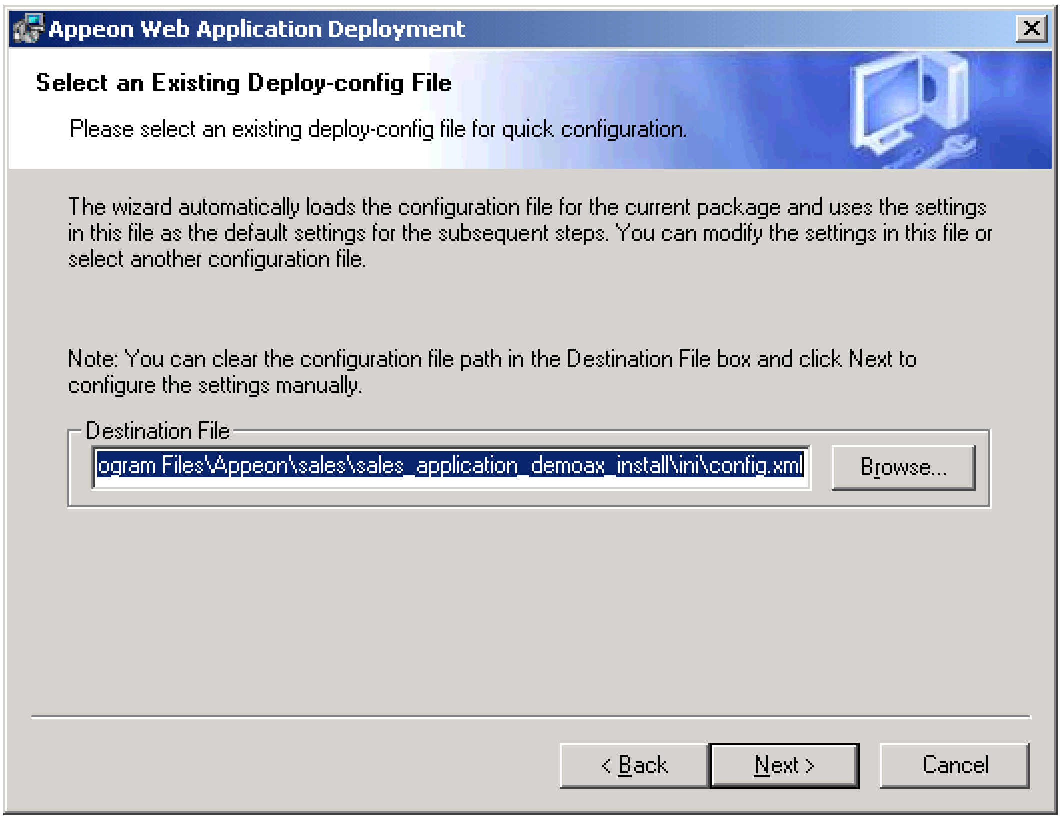
This wizard will install the packaged Appeon Web application on your computer.

To continue, click Next. To exit the wizard, click Cancel.

Step 3 – Select the deploy-config file (config.xml) to ease your configuration of the installation wizard. The installation wizard will use the settings in the config.xml file.

If you do not want to use the deploy-config file, simply skip this option and click *Next* to configure the settings step by step.

Figure 10-13: Select the configuration file



Appeon Web Application Deployment

Select an Existing Deploy-config File

Please select an existing deploy-config file for quick configuration.

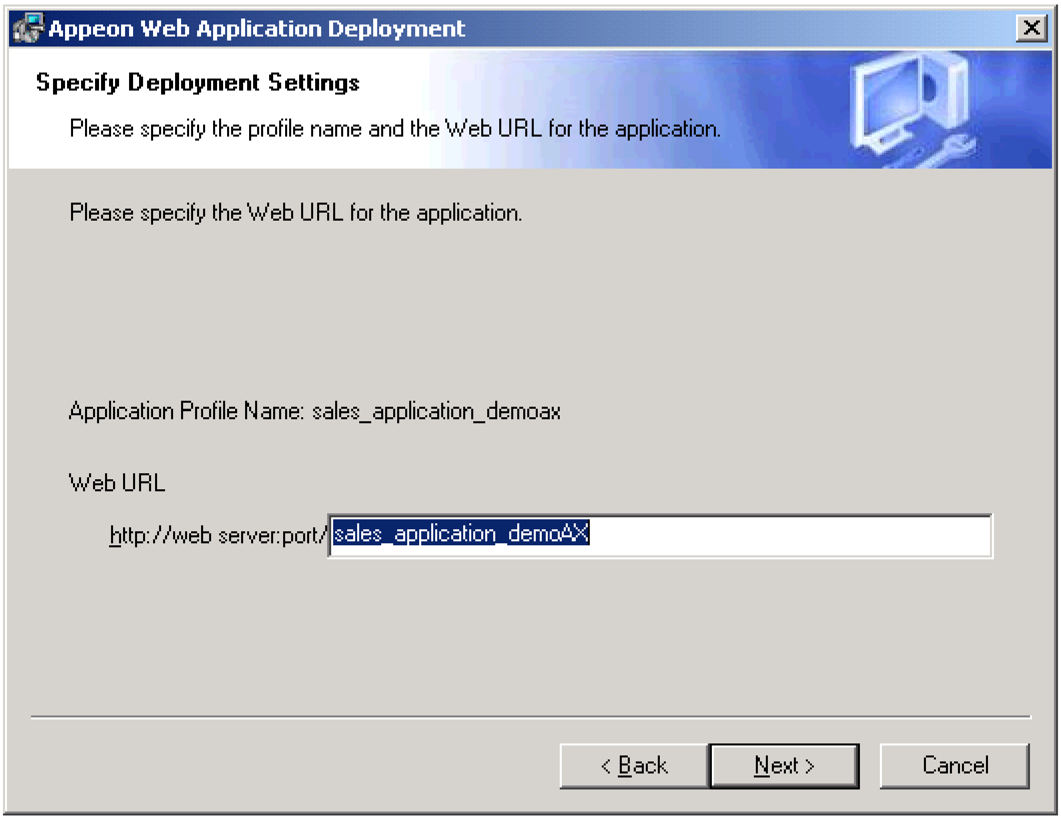
The wizard automatically loads the configuration file for the current package and uses the settings in this file as the default settings for the subsequent steps. You can modify the settings in this file or select another configuration file.

Note: You can clear the configuration file path in the Destination File box and click Next to configure the settings manually.

Destination File: ogram Files\Appeon\sales\sales_application_demoax_install\ini\config.xml

Step 4 – Input the Web URL for the application and click *Next*.

Figure 10-14: Specify Web URL



Appeon Web Application Deployment

Specify Deployment Settings

Please specify the profile name and the Web URL for the application.

Please specify the Web URL for the application.

Application Profile Name: sales_application_demoax

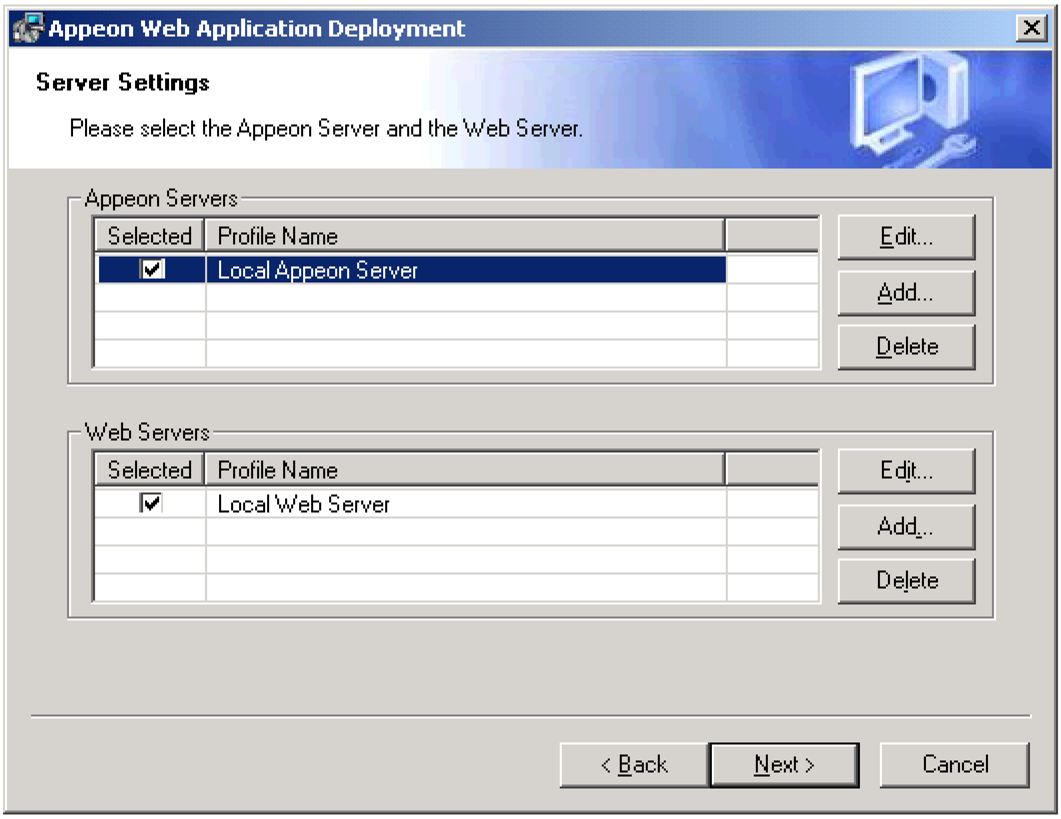
Web URL

http://web server:port/ sales_application_demoAX

Step 5 – Configure and select profiles for the Appeon Server(s) and the Web server(s) where the application will be installed.

The application can only be installed to Appeon Servers of the same type and version as the selected Appeon Server during packaging.

Figure 10-15: Specify server settings



Appeon Web Application Deployment

Server Settings

Please select the Appeon Server and the Web Server.

Appeon Servers

Selected	Profile Name
☑	Local Appeon Server

Web Servers

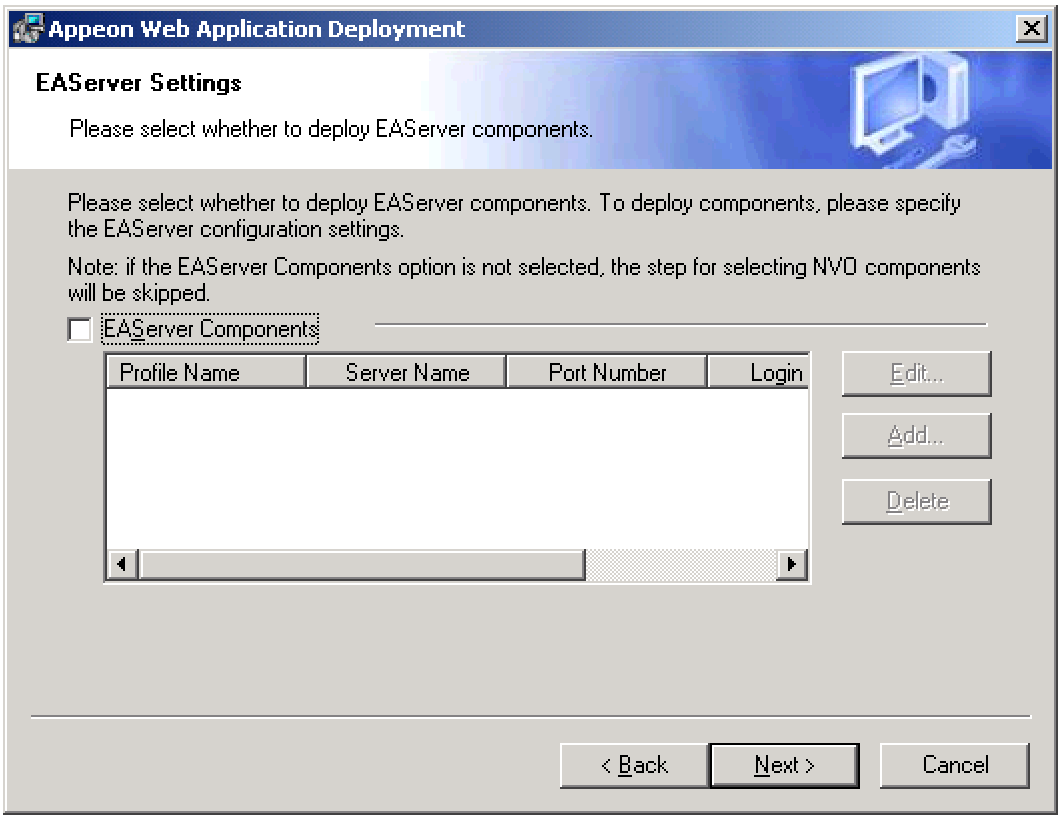
Selected	Profile Name
☑	Local Web Server

For detailed instructions on how to create the Appeon Server profile and Web server profile, refer to Section 4.3.3: *Managing server profiles*.

Step 6 – Select whether to deploy EAServer components.

After NVO components are deployed, you must generate stub/skeleton in EAServer. Detailed instructions are provided in the *Appeon Migration Guide*.

Figure 10-16: Specify EAServer settings



Appeon Web Application Deployment

EAServer Settings

Please select whether to deploy EAServer components.

Please select whether to deploy EAServer components. To deploy components, please specify the EAServer configuration settings.

Note: if the EAServer Components option is not selected, the step for selecting NVO components will be skipped.

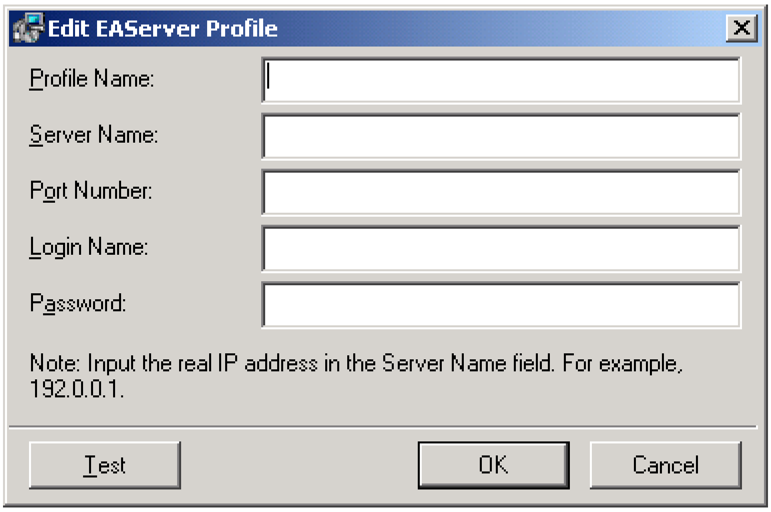
□ EAServer Components

Profile Name	Server Name	Port Number	Login

If EAServer components are used in the application, select the *EAServer Components* option and then create connection profiles for EAServers where you want to deploy the components. Follow instructions below to create EAServer profiles:

- Click *Add* to open the Edit EAServer Profile dialog box (Figure 10-30).
- Input the connection parameters: profile name, host name or IP address, port number, user name and password.
- Click *Test* to verify the connection and then click *OK* to finish.

Figure 10-17: Edit EAServer Profile



Edit EAServer Profile

Profile Name:

Server Name:

Port Number:

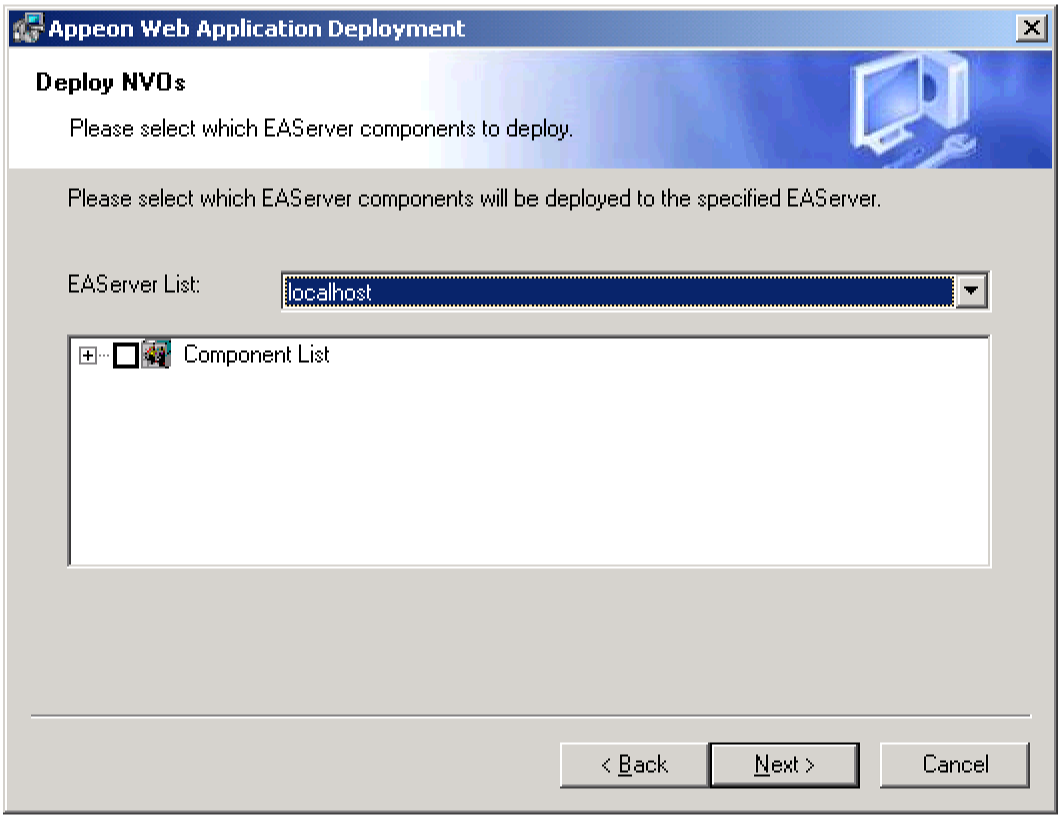
Login Name:

Password:

Note: Input the real IP address in the Server Name field. For example, 192.0.0.1.

Step 7 – If the *EAServer Components* option is selected in the previous step, select the EAServer profile from the list and the components to deploy. Click *Next* to proceed.

Figure 10-18: Deploy NVOs



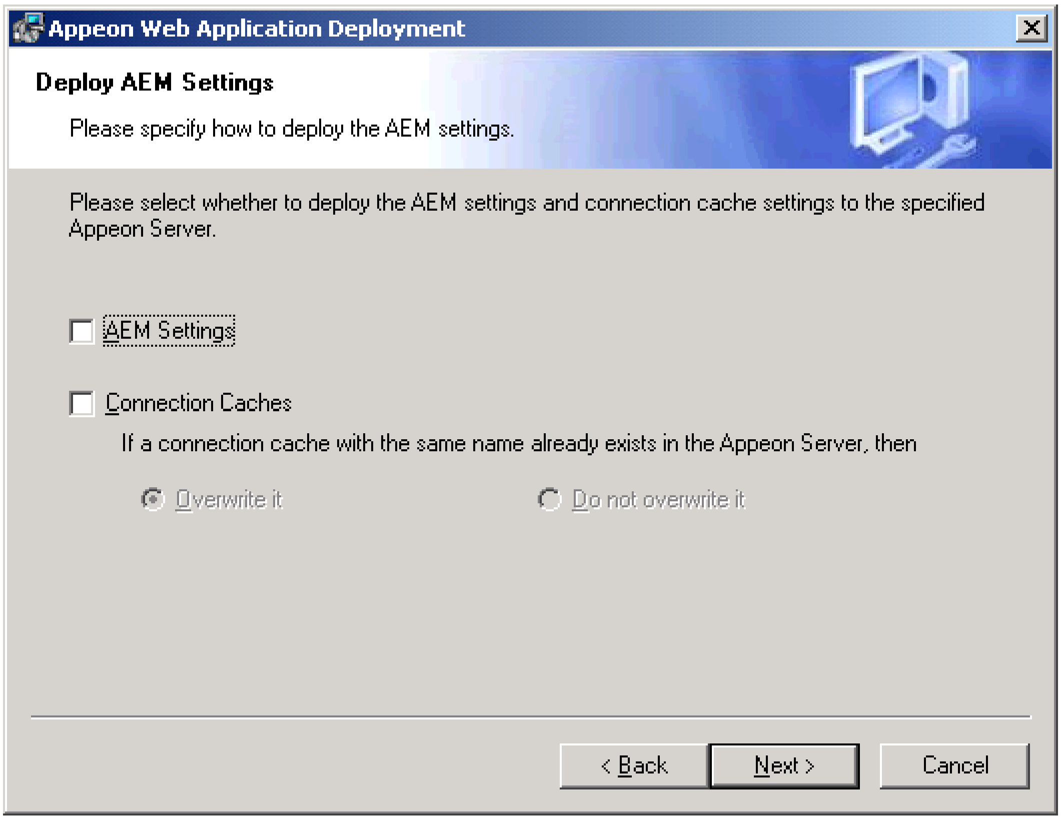
Please select which EAServer components to deploy.

Please select which EAServer components will be deployed to the specified EAServer.

EAServer List: localhost

Step 8 – Select whether to install AEM settings and connection caches for the current application and click *Next*.

Figure 10-19: Deploy AEM settings



Appeon Web Application Deployment

Deploy AEM Settings

Please specify how to deploy the AEM settings.

Please select whether to deploy the AEM settings and connection cache settings to the specified Appeon Server.

☐ AEM Settings

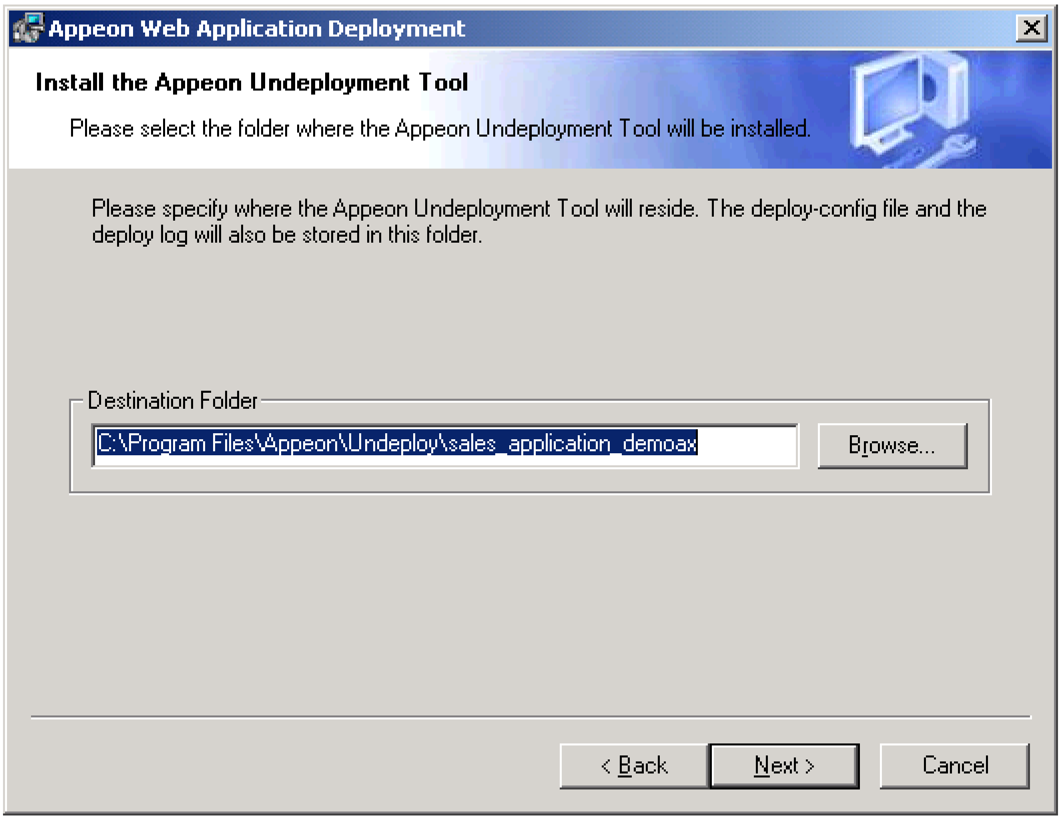
☐ Connection Caches

If a connection cache with the same name already exists in the Appeon Server, then

● Overwrite it ○ Do not overwrite it

Step 9 – Specify the destination folder for the undeployment tool and the log file and click *Next*, as shown in Figure 10-20.

Figure 10-20: Specify location for the Appeon Undeployment Tool



Appeon Web Application Deployment

Install the Appeon Undeployment Tool

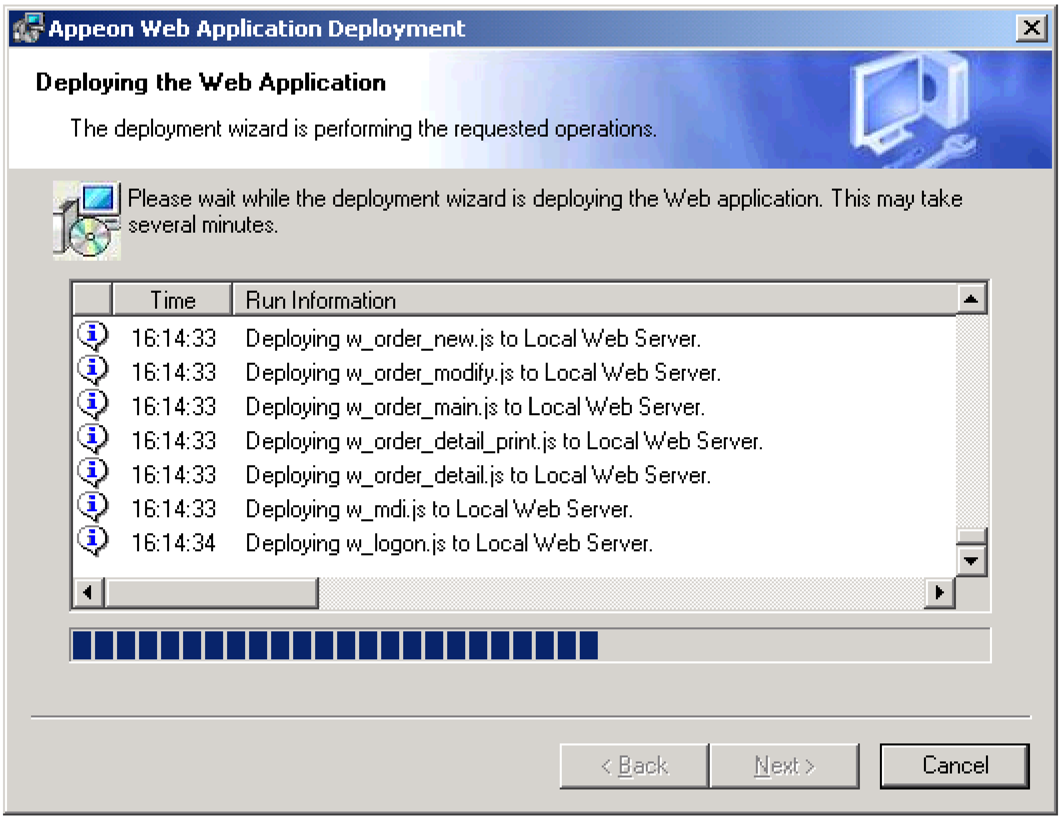
Please select the folder where the Appeon Undeployment Tool will be installed.

Please specify where the Appeon Undeployment Tool will reside. The deploy-config file and the deploy log will also be stored in this folder.

Destination Folder: C:\Program Files\Appeon\Undeploy\sales_application_demoax

Step 10 – Wait while the wizard is installing files and settings, as shown in Figure 10-21.

Figure 10-21: Deploying the application



Deploying the Web Application

The deployment wizard is performing the requested operations.

Please wait while the deployment wizard is deploying the Web application. This may take several minutes.

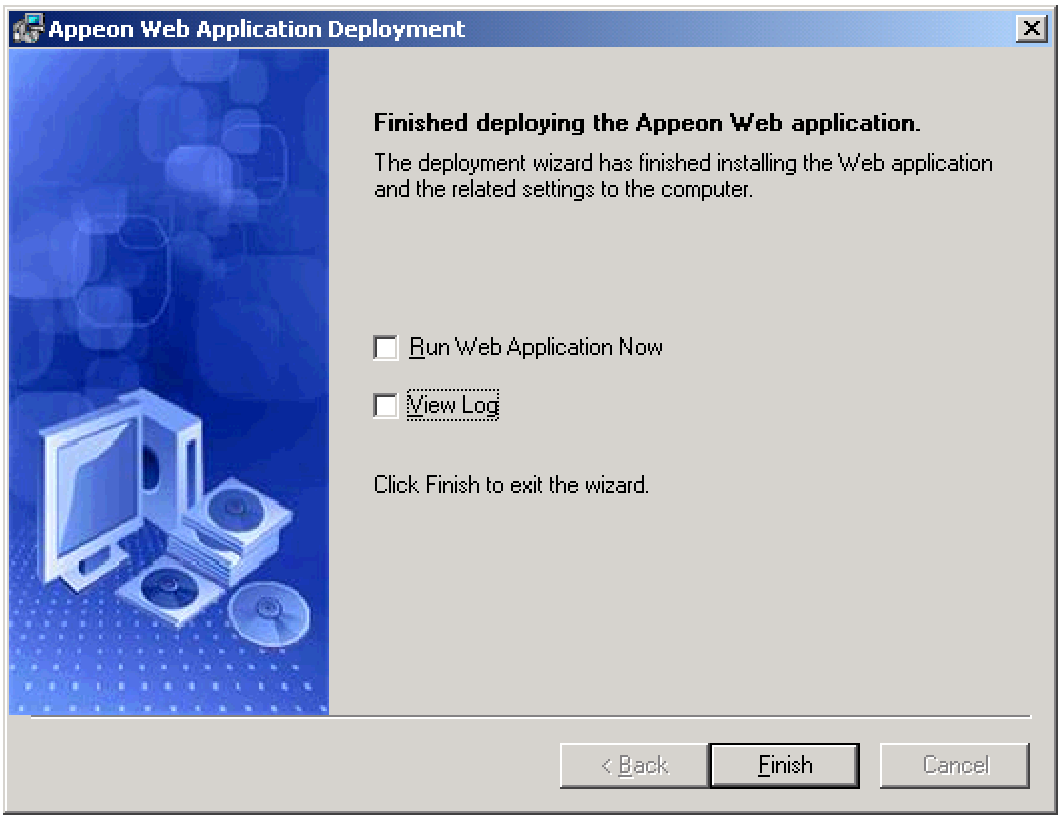
Time	Run Information
16:14:33	Deploying w_order_new.js to Local Web Server.
16:14:33	Deploying w_order_modify.js to Local Web Server.
16:14:33	Deploying w_order_main.js to Local Web Server.
16:14:33	Deploying w_order_detail_print.js to Local Web Server.
16:14:33	Deploying w_order_detail.js to Local Web Server.
16:14:33	Deploying w_mdi.js to Local Web Server.
16:14:34	Deploying w_logon.js to Local Web Server.

Step 11 – Once the operation is complete, click *Finish* to exit the installation wizard.

To run the Web application immediately, select the *Run Web Application Now* option and click *Finish*. Or you can run the application later from the Windows Start | Programs | Appeon Web Application | *ApplicationName*.

To view the log information generated during the installation process, select the *View Log* box and then click *Finish*. The log file will be displayed.

Figure 10-22: Deployment complete



Finished deploying the Appeon Web application.

The deployment wizard has finished installing the Web application and the related settings to the computer.

□ Run Web Application Now

□ View Log

Click Finish to exit the wizard.

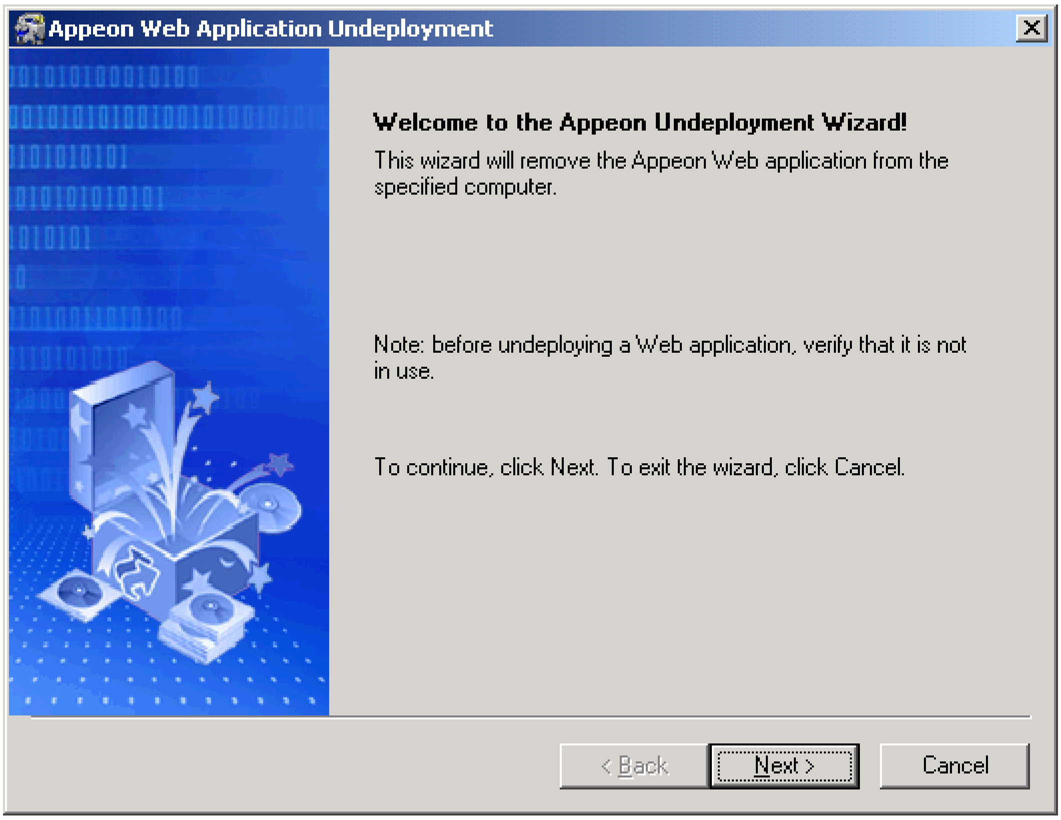
10.3.3 Uninstalling an application

Step 1 – Start the target Appeon Server(s) and/or Web server(s) where you intend to uninstall the application.

Step 2 – Select Windows Start | Programs | Appeon Web Application | Undeploy *ApplicationName*.

The Appeon Web Application Undeployment wizard is displayed, as shown in Figure 10-23. Click *Next* to proceed.

Figure 10-23: Appeon Web Application Undeployment wizard



Appeon Web Application Undeployment

Welcome to the Appeon Undeployment Wizard!

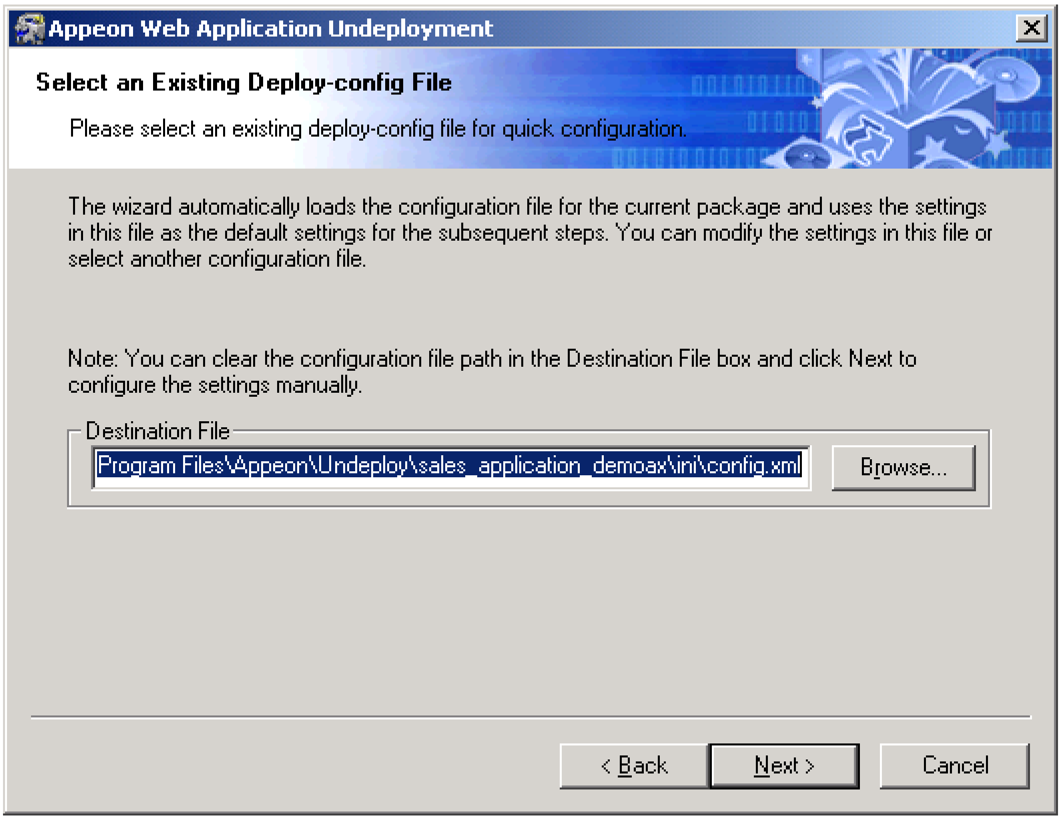
This wizard will remove the Appeon Web application from the specified computer.

Note: before undeploying a Web application, verify that it is not in use.

To continue, click Next. To exit the wizard, click Cancel.

Step 3 – Select an existing deploy-config file to automatically use the configuration previously used for the deployment wizard.

Figure 10-24: Select the configuration file



Appeon Web Application Undeployment

Select an Existing Deploy-config File

Please select an existing deploy-config file for quick configuration.

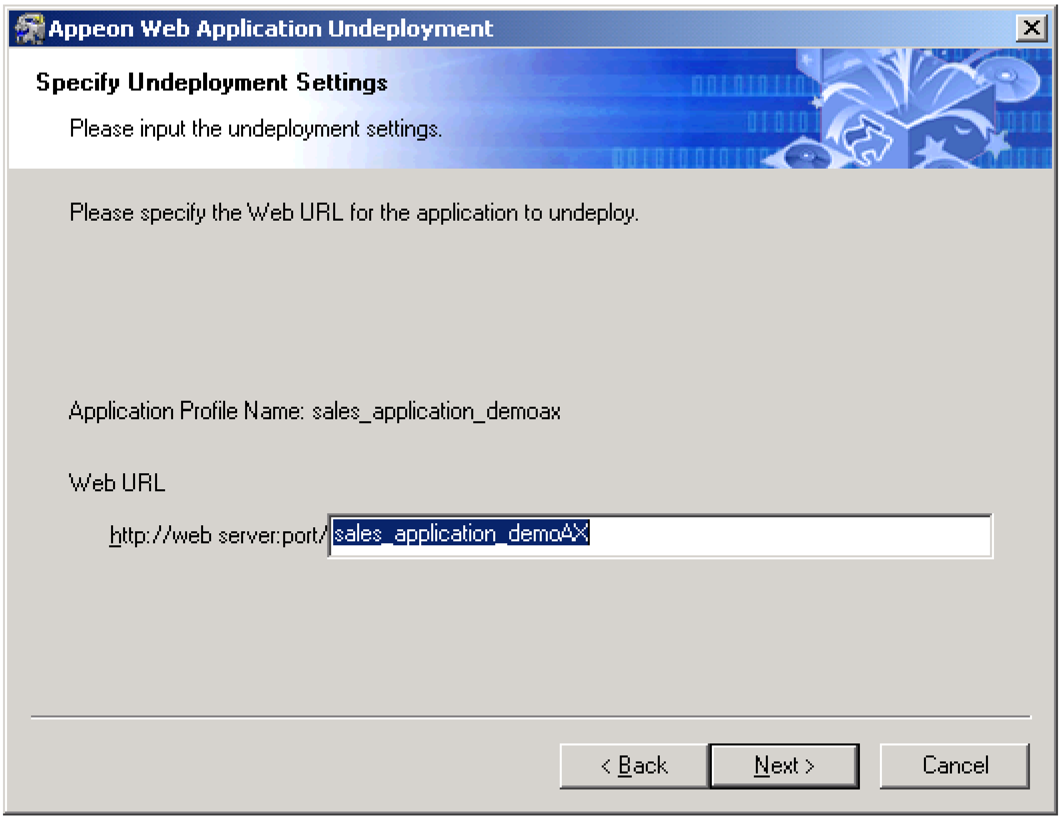
The wizard automatically loads the configuration file for the current package and uses the settings in this file as the default settings for the subsequent steps. You can modify the settings in this file or select another configuration file.

Note: You can clear the configuration file path in the Destination File box and click Next to configure the settings manually.

Destination File: Program Files\Appeon\Undeploy\sales_application_demoax\ini\config.xml

Step 4 – Specify the Web URL.

Figure 10-25: Specify Web URL



Appeon Web Application Undeployment

Specify Undeployment Settings

Please input the undeployment settings.

Please specify the Web URL for the application to undeploy.

Application Profile Name: sales_application_demoax

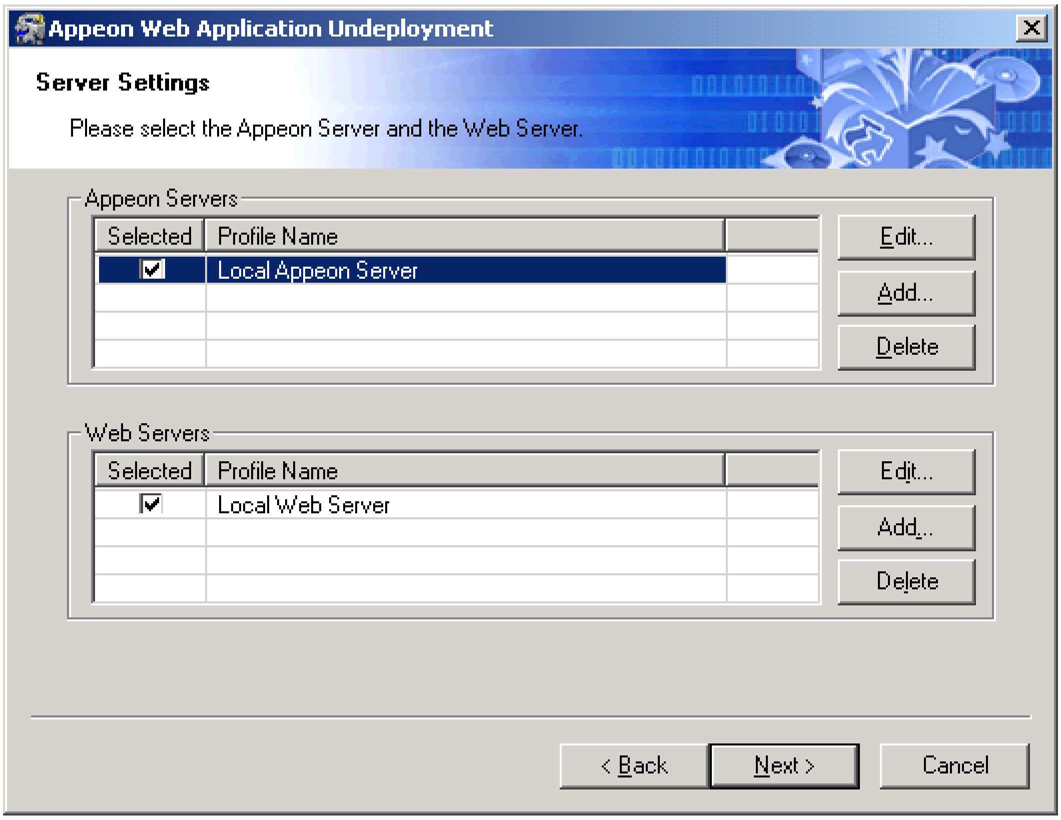
Web URL

http://web server:port/ sales_application_demoAX

Step 5 – Select the Appeon Server(s) and Web server(s) where application will be undeployed.

To create the Appeon Server profile and Web server profile, refer to Section 4.3.3: *Managing server profiles*.

Figure 10-26: Specify server settings



Server Settings

Please select the Appeon Server and the Web Server.

Appeon Servers

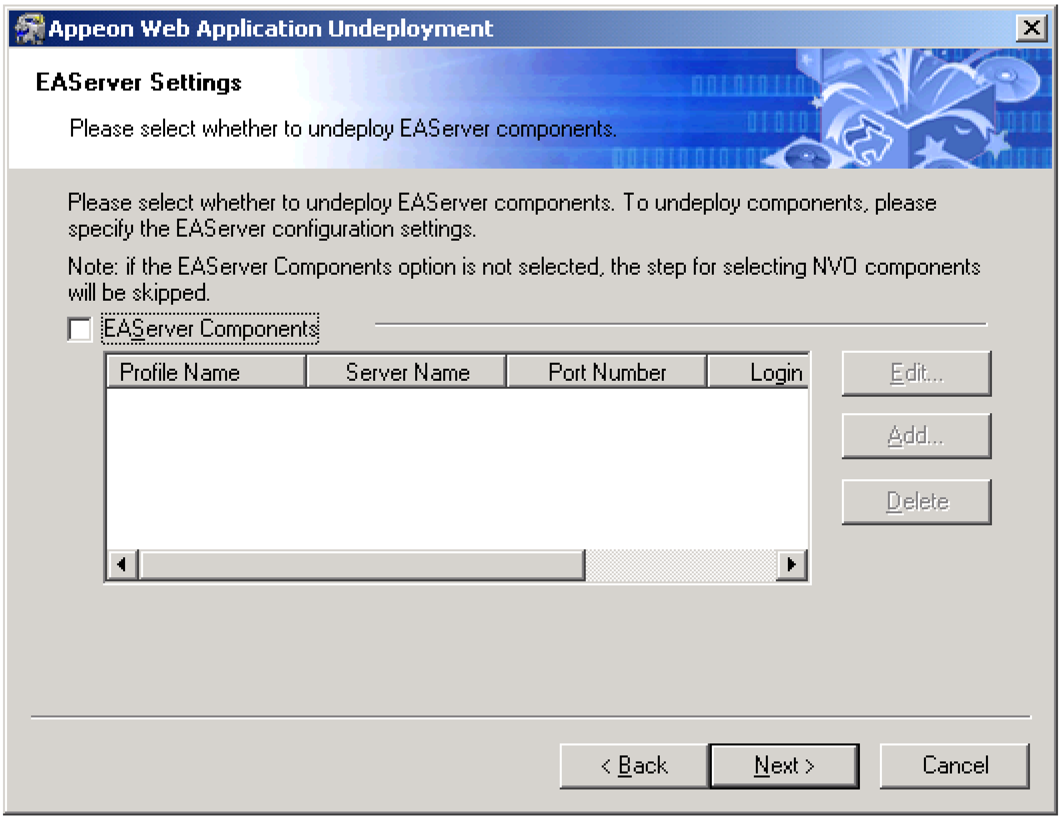
Selected	Profile Name
☑	Local Appeon Server

Web Servers

Selected	Profile Name
☑	Local Web Server

Step 6 – Select whether to undeploy EAServer components.

Figure 10-27: Undeploy EAServer components



EAServer Settings

Please select whether to undeploy EAServer components.

Please select whether to undeploy EAServer components. To undeploy components, please specify the EAServer configuration settings.

Note: if the EAServer Components option is not selected, the step for selecting NVO components will be skipped.

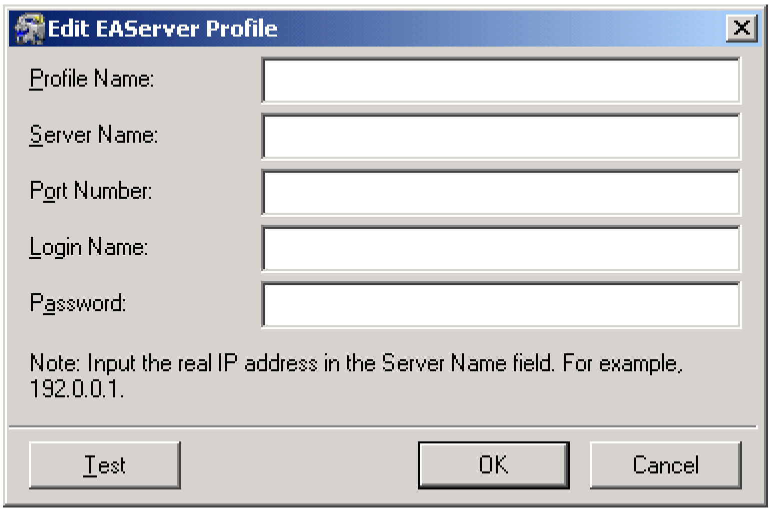
☐ EAServer Components

Profile Name	Server Name	Port Number	Login

Select the *EAServer Components* option and then create connection profiles for EAServers where you want to undeploy the components. Follow instructions below to create EAServer profiles:

- Click *Add* to open the Edit EAServer Profile dialog box (Figure 10-28).
- Input the connection parameters: profile name, host name or IP address, port number, user name and password.
- Click *Test* to verify the connection and then click *OK* to finish.

Figure 10-28: Edit EAServer profile



Profile Name:

Server Name:

Port Number:

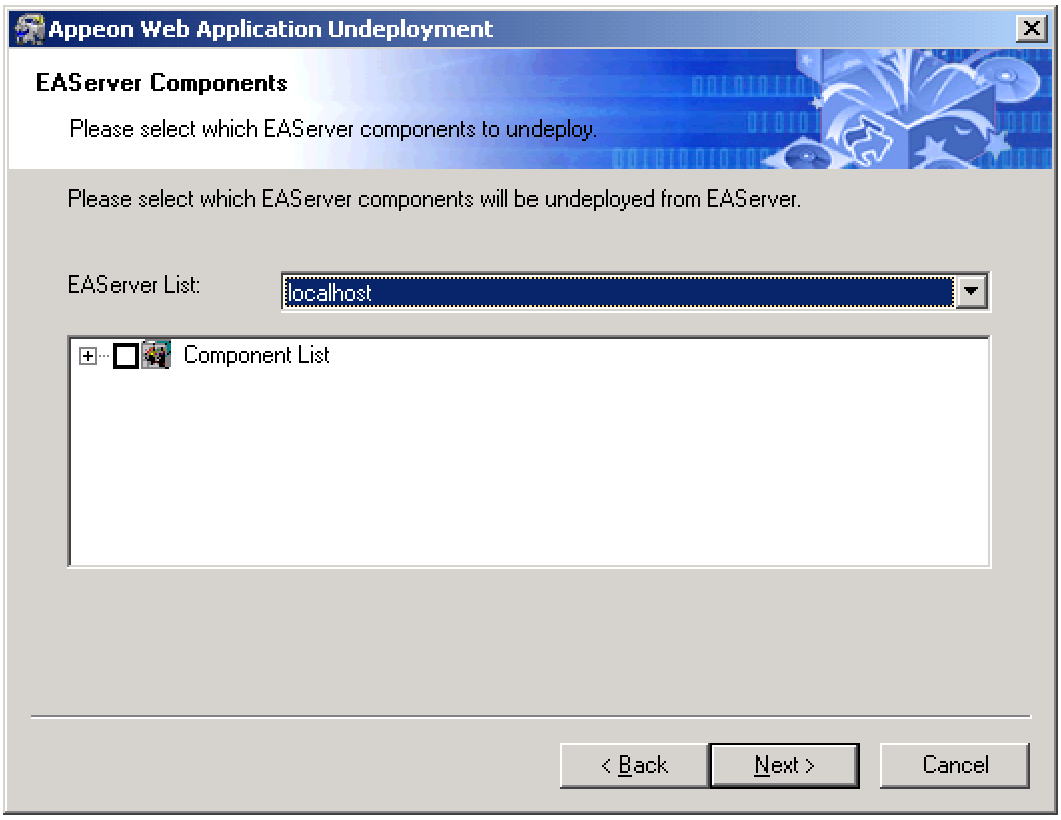
Login Name:

Password:

Note: Input the real IP address in the Server Name field. For example, 192.0.0.1.

Step 7 – If the *EAServer Components* option is selected in the previous step, select the EAServer profile from the list and the components to undeploy. Click *Next* to proceed.

Figure 10-29: Select EAServer components



EAServer Components

Please select which EAServer components to undeploy.

Please select which EAServer components will be undeployed from EAServer.

EAServer List: localhost

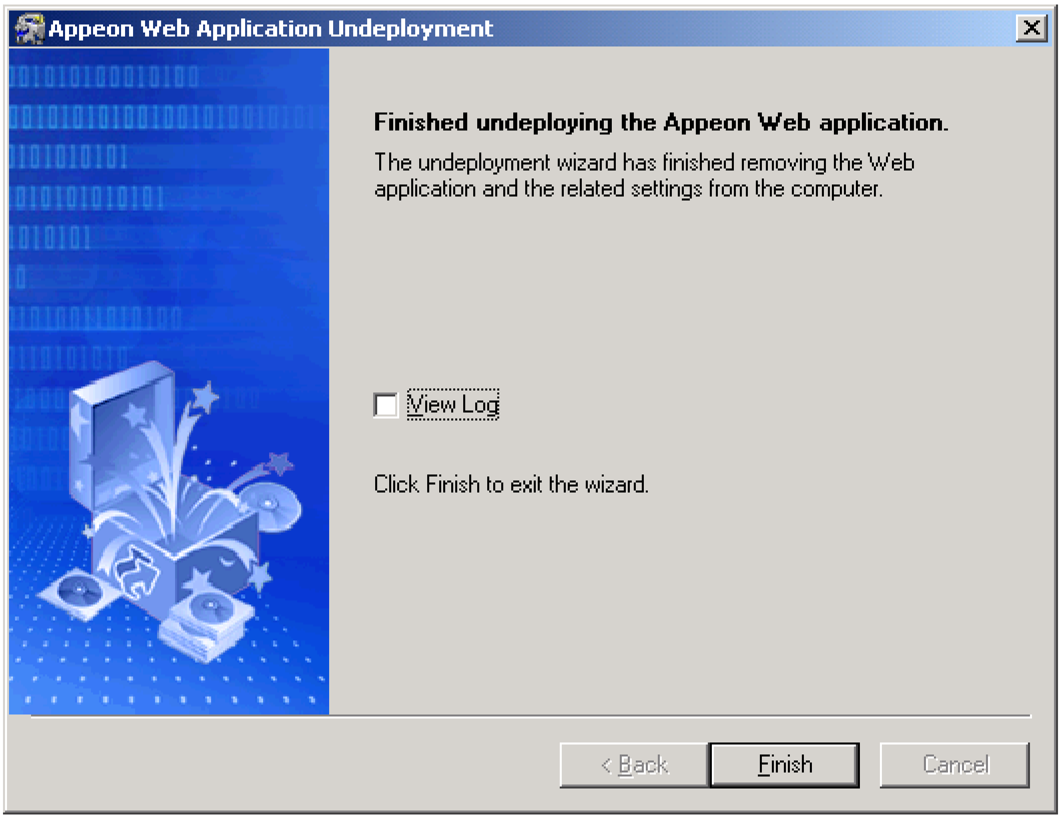
☐ Component List

Step 8 – Wait while the wizard is removing files and settings.

Step 9 – Once the operation is complete, click *Finish* to exit the undeployment wizard.

To view the log information generated during the uninstall process, select the *View Log* box and then click *Finish*. The log file will be automatically opened.

Figure 10-30: Undeployment complete



Appeon Web Application Undeployment

Finished undeploying the Appeon Web application.

The undeployment wizard has finished removing the Web application and the related settings from the computer.

□ View Log

Click Finish to exit the wizard.

11 Undeploying Web Applications

11.1 Overview

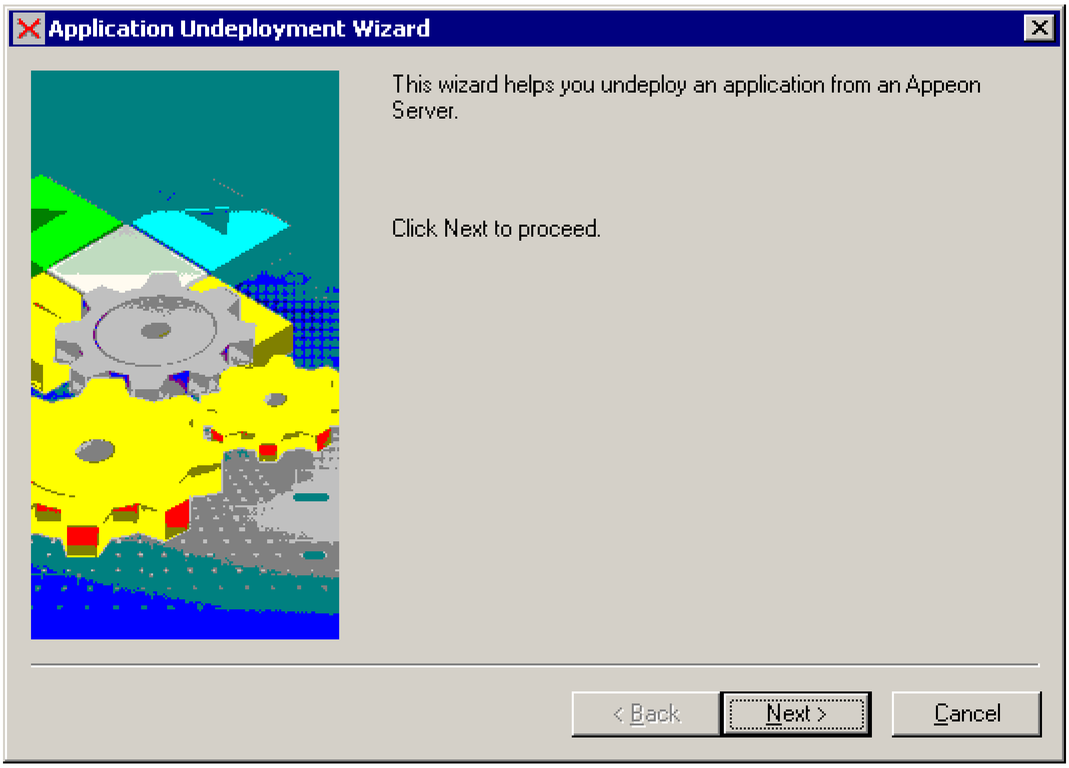
The Application Undeployment Wizard removes a deployed application from the associated Appeon Server(s) and Web Server(s), including all DataWindows from the Appeon Server(s), all Web files from the Web Server(s), and all Transaction Object mappings from AEM. This is the only way that an Appeon deployed application should be removed. Any other method may not fully remove the application, and may cause errors.

11.2 Undeploying instructions

Step 1 – Verify that the Appeon Server and the Web Server(s) hosting the Web application to be undeployed are running before you proceed with undeployment.

Step 2 – Click the *Undeploy* button (✕) in the Appeon Developer toolbar. The Application Undeployment Wizard is displayed, as shown in Figure 11-1.

Figure 11-1: Application Undeployment Wizard

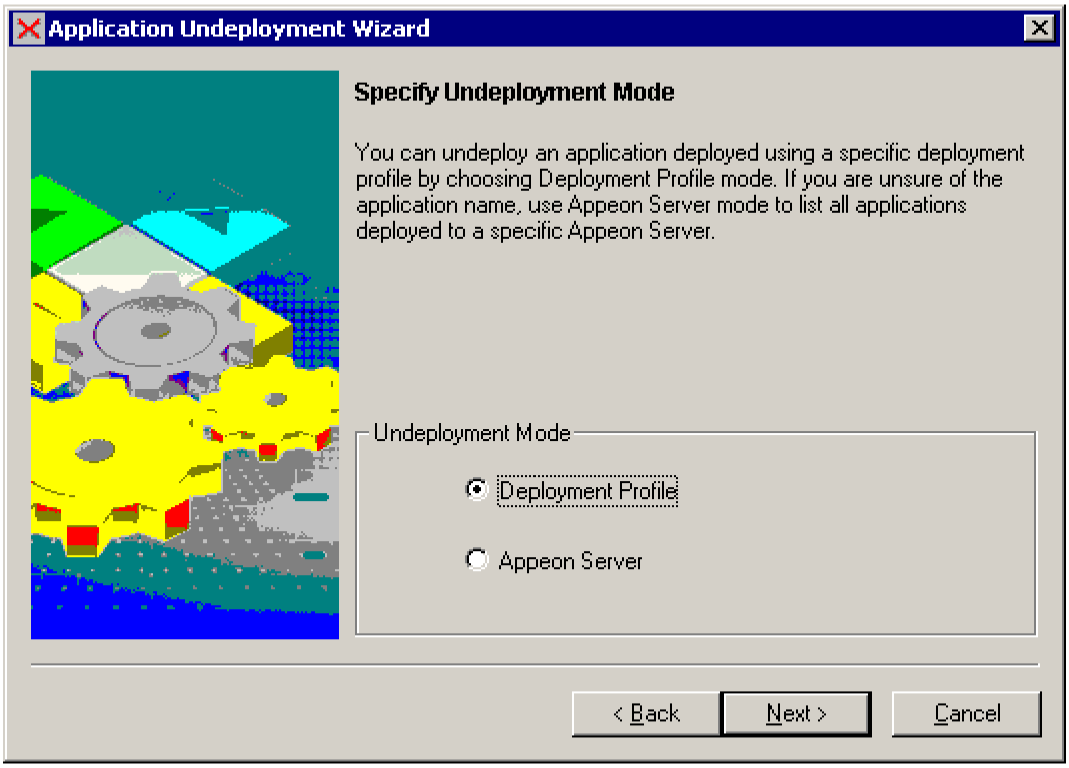


This wizard helps you undeploy an application from an Appeon Server.

Click Next to proceed.

Step 3 – Click *Next* to continue. The Specify Undeployment Mode window is displayed, as shown in Figure 11-2, prompting you to select an undeployment mode.

Figure 11-2: Specify Undeployment Mode window



Specify Undeployment Mode

You can undeploy an application deployed using a specific deployment profile by choosing Deployment Profile mode. If you are unsure of the application name, use Appeon Server mode to list all applications deployed to a specific Appeon Server.

Undeployment Mode

● Deployment Profile

○ Appeon Server

Table 11-1 shows when and how to use each mode.

Table 11-1: Undeployment modes

	When To Use It	How To Use It
Deployment Profile mode	If you clearly know which application you want to delete from which server.	Select an application to be undeployed and its associated deployment profile.
Appeon Server mode	When you are uncertain of the name of the application you want to undeploy, or an application profile does not exist for the application you want to undeploy, but you clearly know the Appeon Server hosting the Web application intended for undeployment.	Specify an Appeon Server by selecting an Appeon Server profile. The selected Appeon Server will refer to all the Web applications that are deployed. Then you can choose one Web application for undeployment.

11.2.1 Undeploying with the Deployment Profile mode

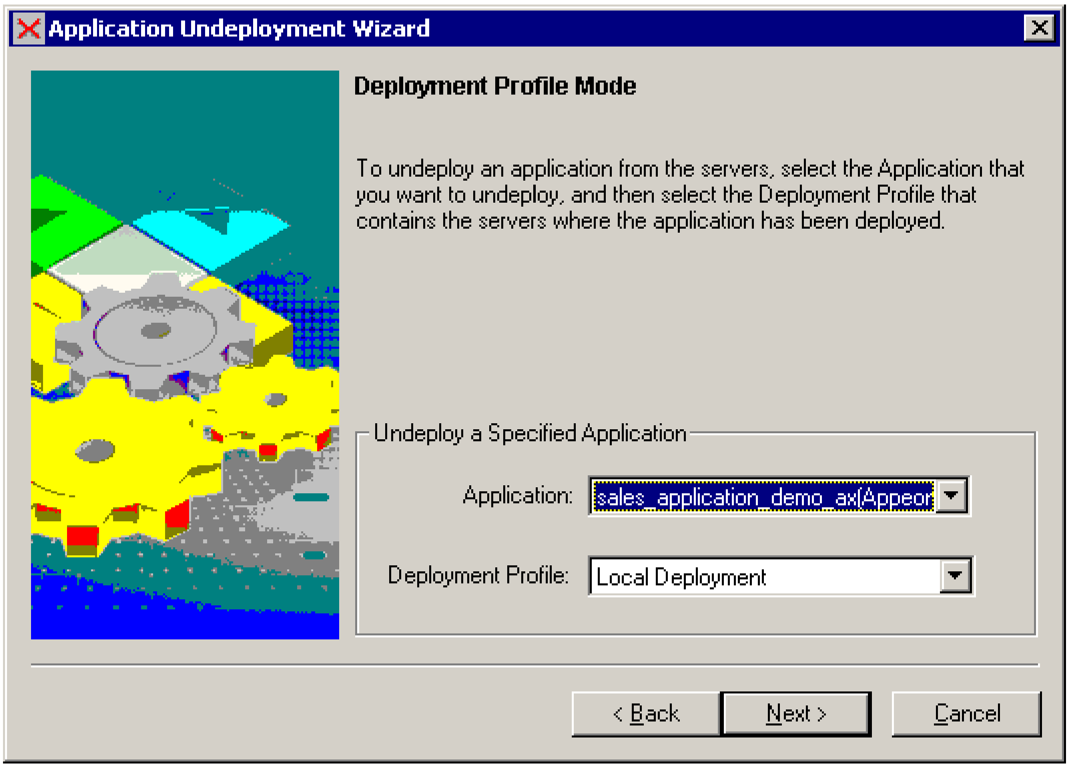
Step 1 – Select the *Deployment Profile* radio button in the Specify Undeployment Mode window as shown in Figure 11-2, and click *Next* to continue.

Step 2 – Select a Web application from the Application list box, as shown in Figure 11-3.

The Application list box lists all the application profile names. Be sure to choose one that has been deployed and is intended for undeployment. The application and the deployment profile used in the last deployment will be selected by default.

Step 3 – Select the deployment profile that is used to deploy the Web application from the Deployment Profile list box, as shown in Figure 11-3. Click *Next* to continue.

Figure 11-3: Deployment Profile Mode window



Application Undeployment Wizard

Deployment Profile Mode

To undeploy an application from the servers, select the Application that you want to undeploy, and then select the Deployment Profile that contains the servers where the application has been deployed.

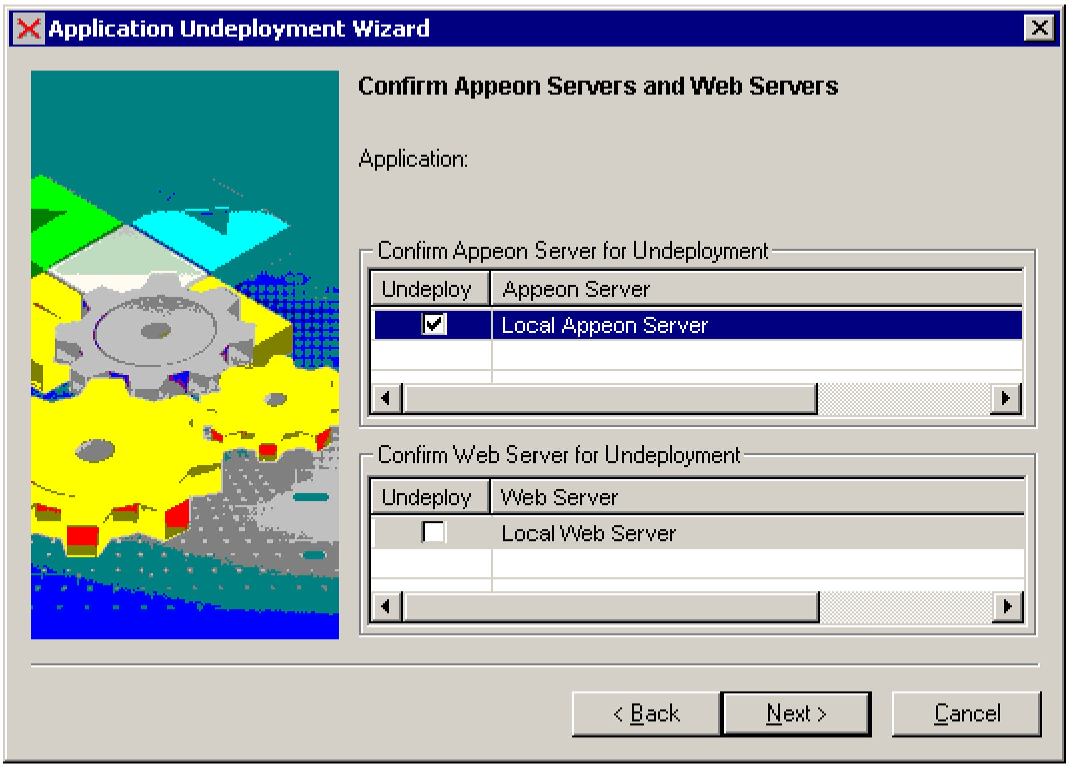
Undeploy a Specified Application

Application: sales_application_demo_ax(Appeon

Deployment Profile: Local Deployment

Step 4 – Select the Appeon Server(s) or Web Server(s) that are intended for undeployment, as shown in Figure 11-4. The Appeon Server(s) and Web Server(s) defined in the deployment profile selected in the previous step are listed. Click *Next* to continue.

Figure 11-4: Confirm Appeon Servers and Web Servers



Application Undeployment Wizard

Confirm Appeon Servers and Web Servers

Application:

Confirm Appeon Server for Undeployment

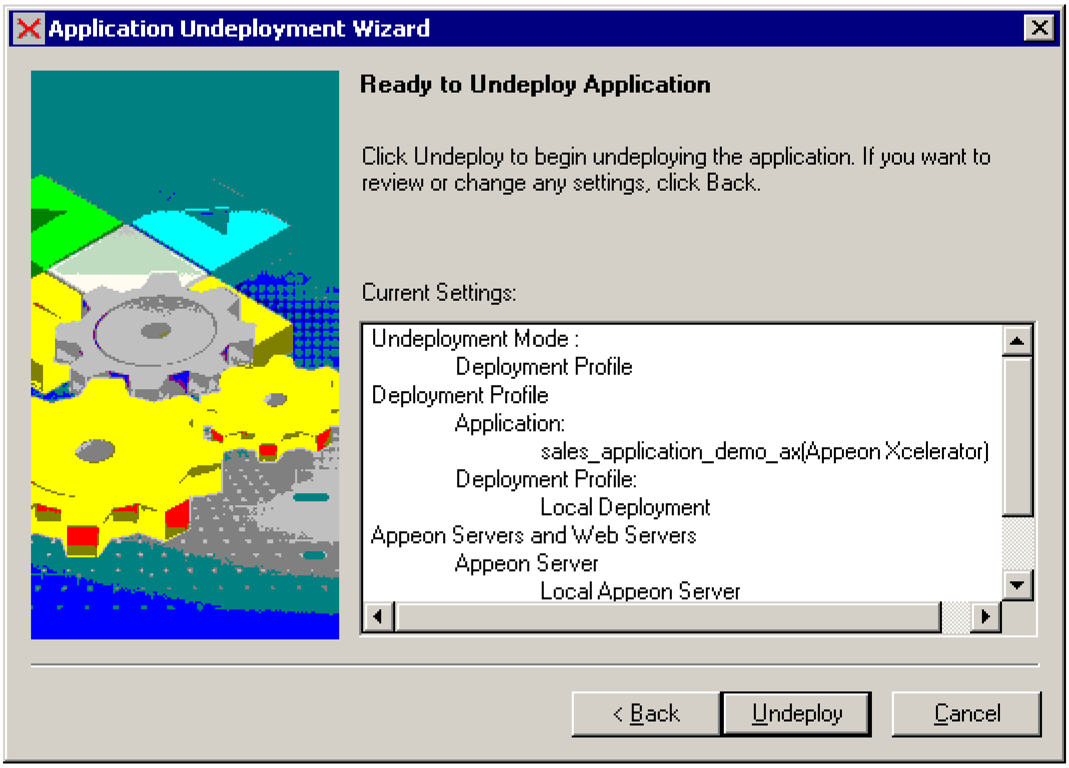
Undeploy	Appeon Server
☑	Local Appeon Server

Confirm Web Server for Undeployment

Undeploy	Web Server
☐	Local Web Server

Step 5 – Click *Undeploy* to confirm the undeployment settings and start the undeployment process, as shown in Figure 11-5.

Figure 11-5: Confirm undeployment settings



Application Undeployment Wizard

Ready to Undeploy Application

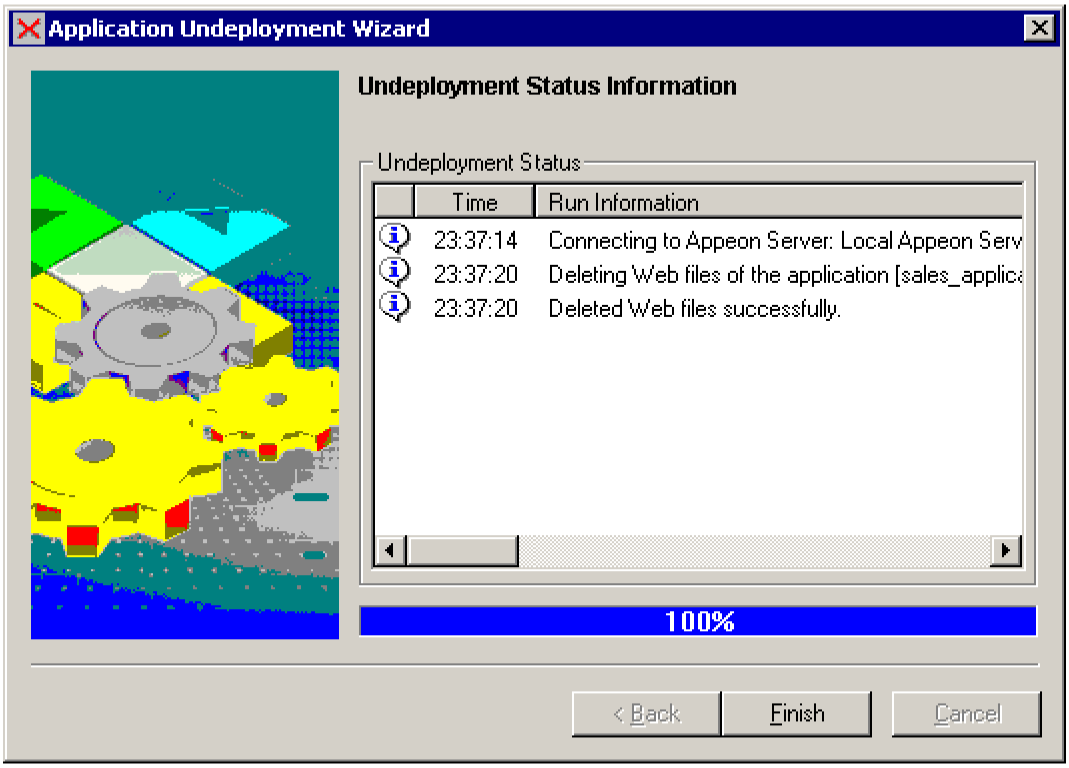
Click Undeploy to begin undeploying the application. If you want to review or change any settings, click Back.

Current Settings:

```
Undeployment Mode :
        Deployment Profile
Deployment Profile
        Application:
                sales_application_demo_ax(Appeon Xcelerator)
        Deployment Profile:
                Local Deployment
Appeon Servers and Web Servers
        Appeon Server
                Local Appeon Server
```

The undeployment process begins, as shown in Figure 11-6.

Figure 11-6: Undeployment in Progress



Application Undeployment Wizard

Undeployment Status Information

Undeployment Status

	Time	Run Information
ℹ	23:37:14	Connecting to Appeon Server: Local Appeon Serv
ℹ	23:37:20	Deleting Web files of the application [sales_applic
ℹ	23:37:20	Deleted Web files successfully.

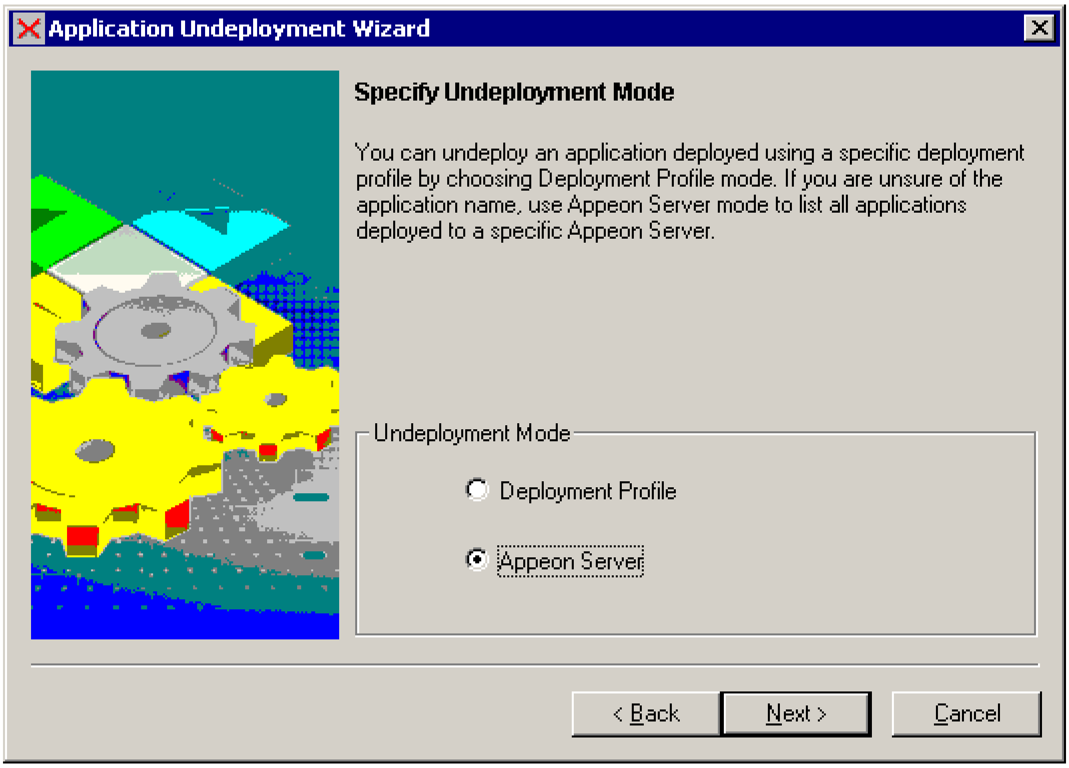
100%

Step 6 – Click *Finish* to close the dialog box, as shown in Figure 11-6.

11.2.2 Undeploying with the Appeon Server mode

Step 1 – Select the *Appeon Server* radio button in the Specify Undeployment Mode window as shown in Figure 11-7, and click *Next* to continue.

Figure 11-7: Specify Undeployment Mode window



Application Undeployment Wizard

Specify Undeployment Mode

You can undeploy an application deployed using a specific deployment profile by choosing Deployment Profile mode. If you are unsure of the application name, use Appeon Server mode to list all applications deployed to a specific Appeon Server.

Undeployment Mode

○ Deployment Profile

● Appeon Server

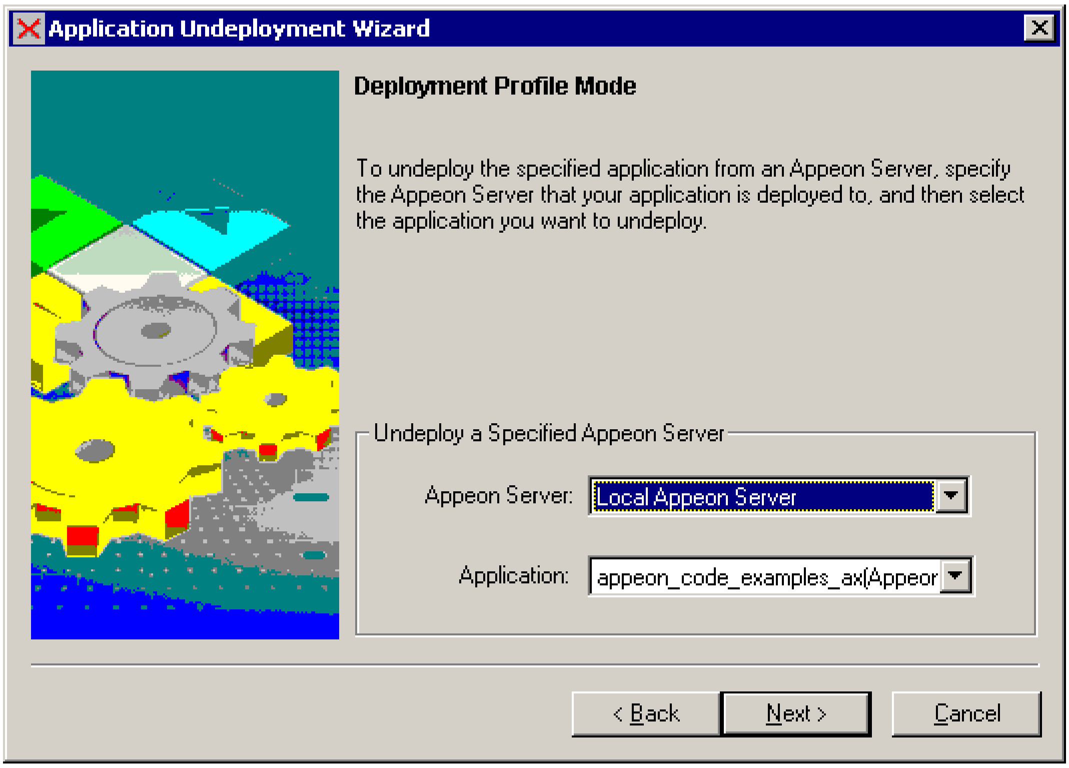
Step 2 – Select the Appeon Server that contains the Web application to be undeployed from the Appeon Server list box, as shown in Figure 11-8.

The Appeon Server list box lists all the Appeon Server profile names. Make sure the selected Appeon Server is running.

Step 3 – Select the application to be undeployed from the Application list box and click *Next*, as shown in Figure 11-8.

The Application list box lists all the Web applications deployed to the selected Appeon Server.

Figure 11-8: Appeon Server Mode window



Deployment Profile Mode

To undeploy the specified application from an Appeon Server, specify the Appeon Server that your application is deployed to, and then select the application you want to undeploy.

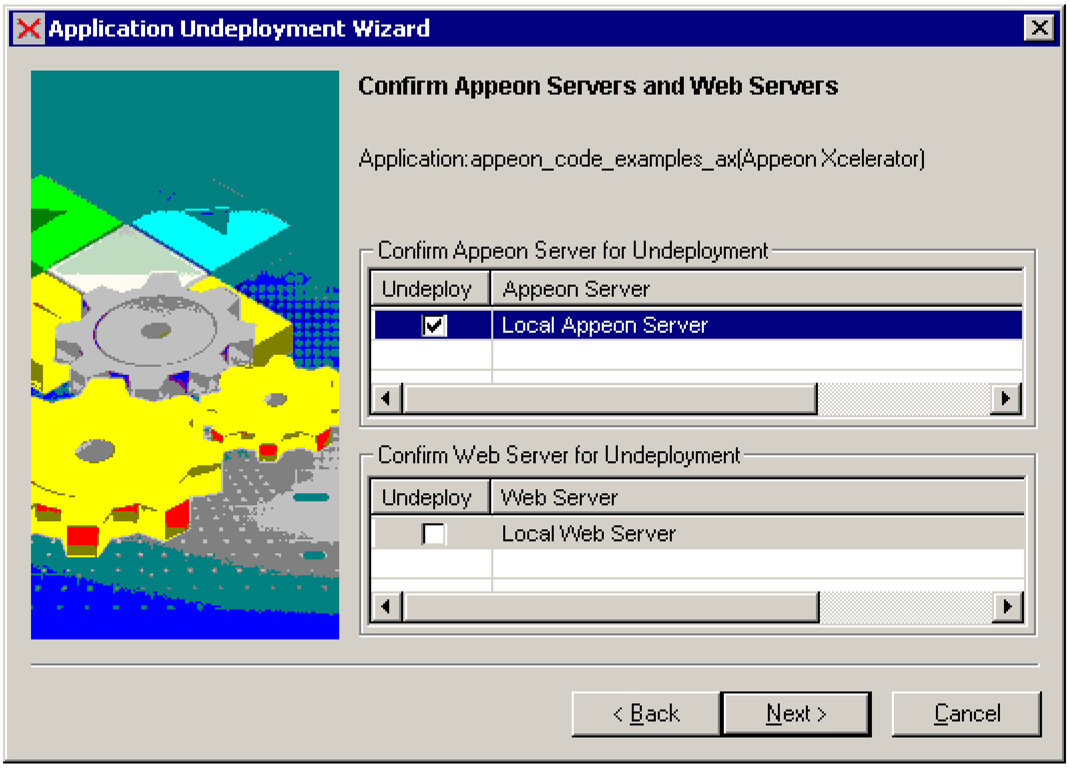
Undeploy a Specified Appeon Server

Appeon Server: Local Appeon Server

Application: appeon_code_examples_ax(Appeor

Step 4 – Select the Appeon Server(s) or Web Server(s) that are intended for undeployment, as shown in Figure 11-9. The Appeon Server(s) and Web Server(s) defined in the default deployment profile are listed. Click *Next* to continue.

Figure 11-9: Ready to Undeploy Application window



Confirm Appeon Servers and Web Servers

Application:appeon_code_examples_ax(Appeon Xcelerator)

Confirm Appeon Server for Undeployment

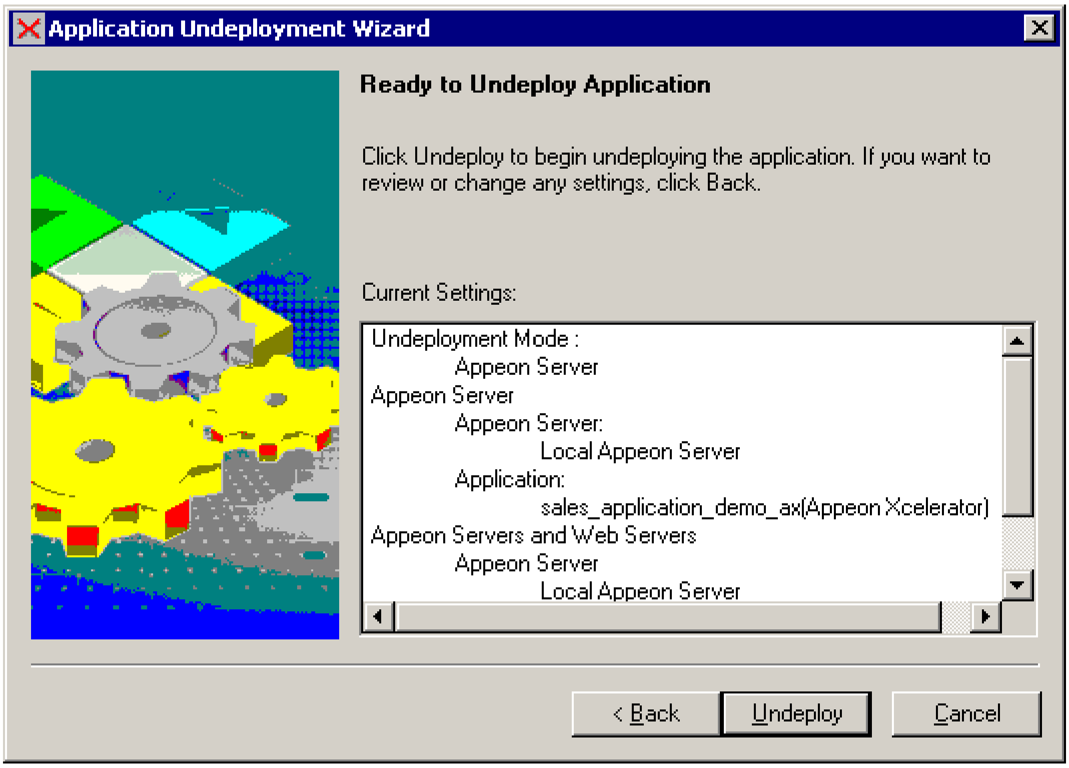
Undeploy	Appeon Server
☑	Local Appeon Server

Confirm Web Server for Undeployment

Undeploy	Web Server
□	Local Web Server

Step 5 – Click *Undeploy* to confirm the undeployment settings and start the undeployment process, as shown in Figure 11-10.

Figure 11-10: Undeployment settings



Ready to Undeploy Application

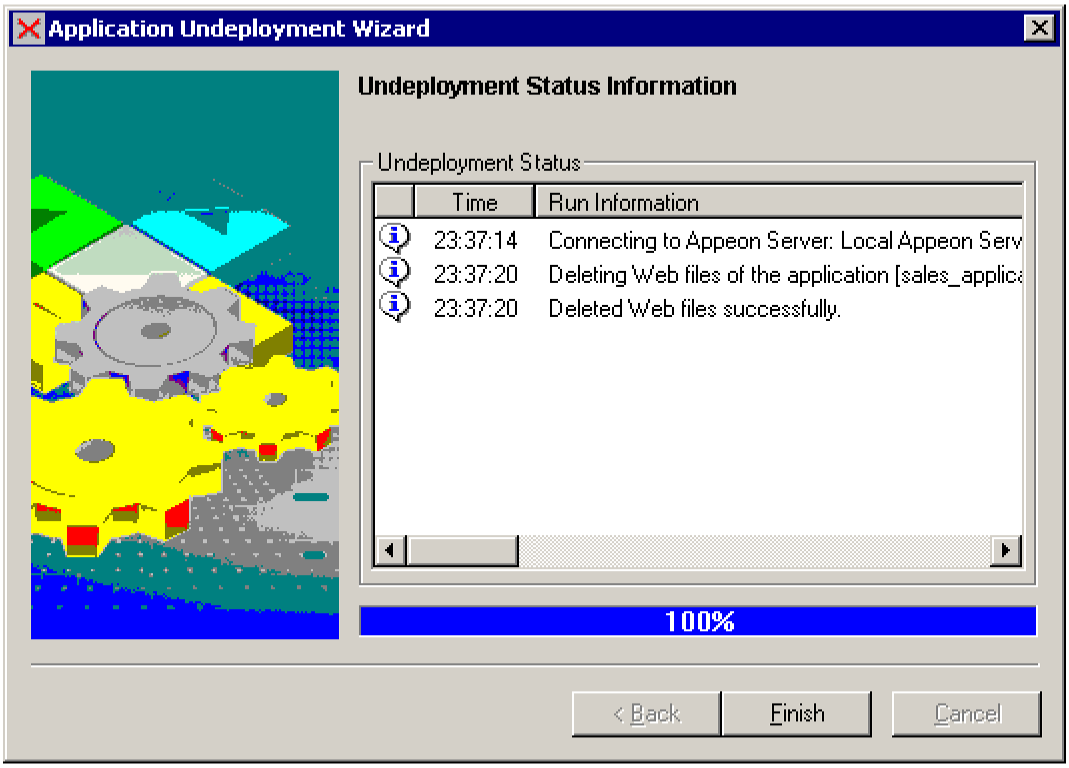
Click Undeploy to begin undeploying the application. If you want to review or change any settings, click Back.

Current Settings:

```
Undeployment Mode :
        Appeon Server
Appeon Server
        Appeon Server:
                Local Appeon Server
        Application:
                sales_application_demo_ax(Appeon Xcelerator)
Appeon Servers and Web Servers
        Appeon Server
                Local Appeon Server
```

The undeployment process begins, as shown in Figure 11-11.

Figure 11-11: Appeon Web Application Undeployment



Undeployment Status Information

Undeployment Status

	Time	Run Information
ⓘ	23:37:14	Connecting to Appeon Server: Local Appeon Serv
ⓘ	23:37:20	Deleting Web files of the application [sales_applica
ⓘ	23:37:20	Deleted Web files successfully.

100%

Step 6 – Click *Finish* to close the dialog box, as shown in Figure 11-11.

12 Developing with Code Insight

12.1 Overview

Code Insight is a tool designed to help you write Appeon-supported PowerScript code more quickly by providing a lookup and paste service inside the PowerBuilder Script view. You can use it to develop PowerBuilder applications that are free of Appeon unsupported features and tailored to Web migration.

The usage of Code Insight is similar to that of PowerBuilder AutoScript. Table 12-1 describes the differences between the PowerBuilder AutoScript and Appeon Code Insight.

Table 12-1: Comparison between AutoScript and Code Insight

	PowerBuilder AutoScript	Appeon Code Insight
Effective for	Functions, events, variables, properties, and templates for PowerBuilder DO, FOR, IF, and CHOOSE statements.	Items referred in dot notations
What pops up	A list of properties, variables, methods, or statements.	1) A list of properties, variables, or methods. 2) The Appeon unsupported properties, variables or methods that are marked with a red icon left to them.
Ways to use it	Two ways to use AutoScript: • Turn automatic popup on to pop up a list automatically when you pause while typing, or • Invoke AutoScript only once when you need it by selecting the menu items.	Only one way to use Code Insight: Step 1: Activate Code Insight; Step 2: Configure the default PBT file; Step 3: Enable Code Insight. Refer to the following sections for detailed instructions.
Priority	When Code Insight is activated, AutoScript will be automatically turned off.	

12.2 Activating Code Insight

Click the *Code Insight* button (▦) in the Appeon Developer toolbar to launch Code Insight.

At the click of the *Code Insight* button, an icon will appear in the status area of the task bar. Code Insight has five status icons:

Table 12-2: Status of Code Insight

Icon	Status	Description
🚫	Disabled	Code Insight is disabled.
🔄	Loading	Code Insight is analyzing the default PBT and generating the supported and unsupported feature list based on the analysis result.
Off	Off	Code Insight is enabled but ineffective.

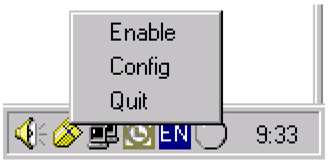
On	On	Code Insight is enabled and effective in the current application.
●	Gray	This icon appears anytime the PowerBuilder IDE loses focus, even if Code Insight is disabled.

12.2.1 Configuring Code Insight

Before you can enable Code Insight for a PowerBuilder application, you need to configure the PBT file of the desired application as the default PBT file. You can also specify the Appeon feature sets and the feature types for which feature list will be displayed.

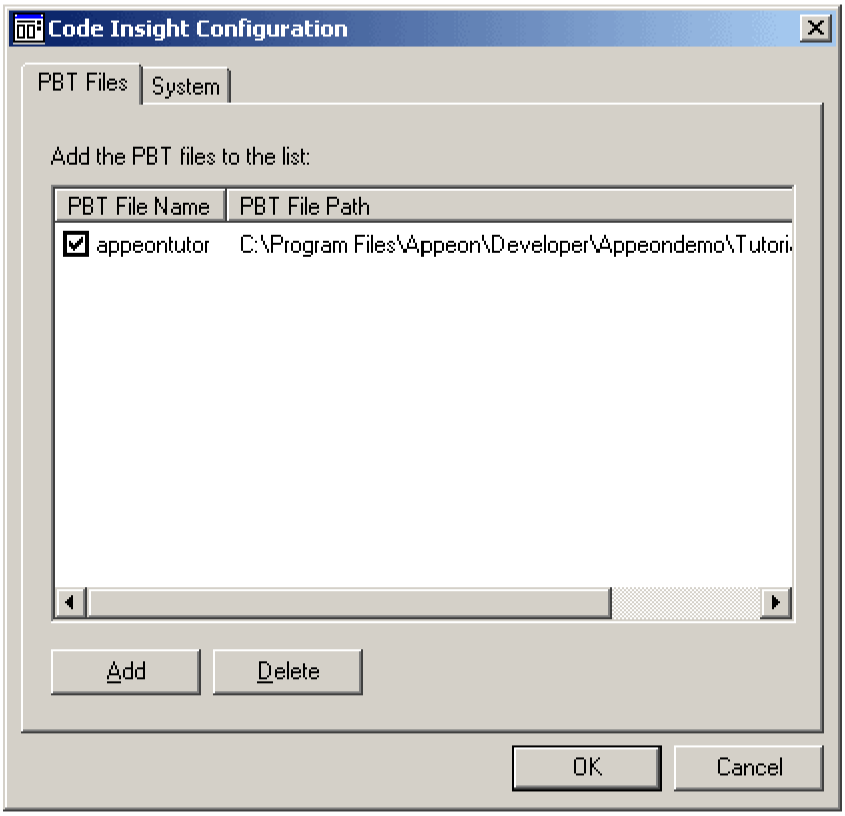
Step 1 – Click on the icon of Code Insight, and the status-area-icon menu pops up with three items, Enable (or Disable), Config, and Quit, as shown in Figure 12-1.

Figure 12-1: The status-area-icon menu



Step 2 – Click *Config* to open the Code Insight Configuration window, as shown in Figure 12-2.

Figure 12-2: Code Insight Configuration window



Add the PBT files to the list:

PBT File Name	PBT File Path
☑ appeontutor	C:\Program Files\Appeon\Developer\Appeondemo\Tutori

The Code Insight Configuration window contains two tab pages where you can manage the PBT files and specify what to display in the list.

Table 12-3: Code Insight Configuration window

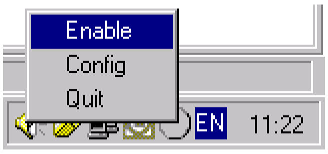
Select this tab	To do this	Detailed Steps
PBT File tab page	Add the PBT files.	1) Click *Add*. 2) Search and select the PBT file in the common dialog box. 3) Click *Open* in the common dialog box. 4) Click *OK*.
	Delete the PBT files.	1) Select the PBT file you want to delete. 2) Click *Delete* to delete the selected PBT file. 3) Click *OK*. NOTE: The default PBT file cannot be deleted.
	Set the default PBT file.	1) Select the check box in front of a PBT file. 2) Click *OK*. NOTE: 1) only one PBT file can be set to the default PBT. 2) Code Insight can only be effective to the default PBT.
System tab page	Specify what to display in the list.	1) In the *Auto List* group box, opt to automatically display functions, events, properties, controls and/or variables in the list. 2) In the *Feature Type* group box, opt to display only the unsupported features and/or the supported features in the list. 3) Click *OK*.

12.2.2 Enabling Code Insight

After you complete the configuration, you can enable Code Insight and use it to develop the PowerBuilder application.

Click on the icon of Code Insight and select *Enable* from the status-area-icon menu, as shown in Figure 12-3.

Figure 12-3: Enable Code Insight



The Code Insight icon appears in one of the following statuses:

Table 12-4: Enable Code Insight

Status	Detailed Steps
⊘	Code Insight remains **disabled**. This happens when the PBT File list in the Code Insight Configuration window is empty. Make sure you have added the PBT file to the list.
Off	Code Insight is **enabled but ineffective** in the current application. To make it effective, make sure you have

	1) Configured the PBT file of the current application as the default PBT file in the Code Insight Configuration window. 2) Opened an object of the current application in a painter (not in the Source editor) by double-clicking it in the Workspace or selecting *Edit* from the popup menu. After taking the above steps, Code Insight becomes effective (On).
On	Code Insight is **enabled and effective**. You can develop an application with the assistance of Code Insight.

12.3 Coding with Code Insight

When Code Insight is enabled and effective (On) for an application, you can start editing the objects of the application using Code Insight. For example, take the following steps to add a new control and use Code Insight to edit it:

Step 1 – Add a button to the current object in the painter.

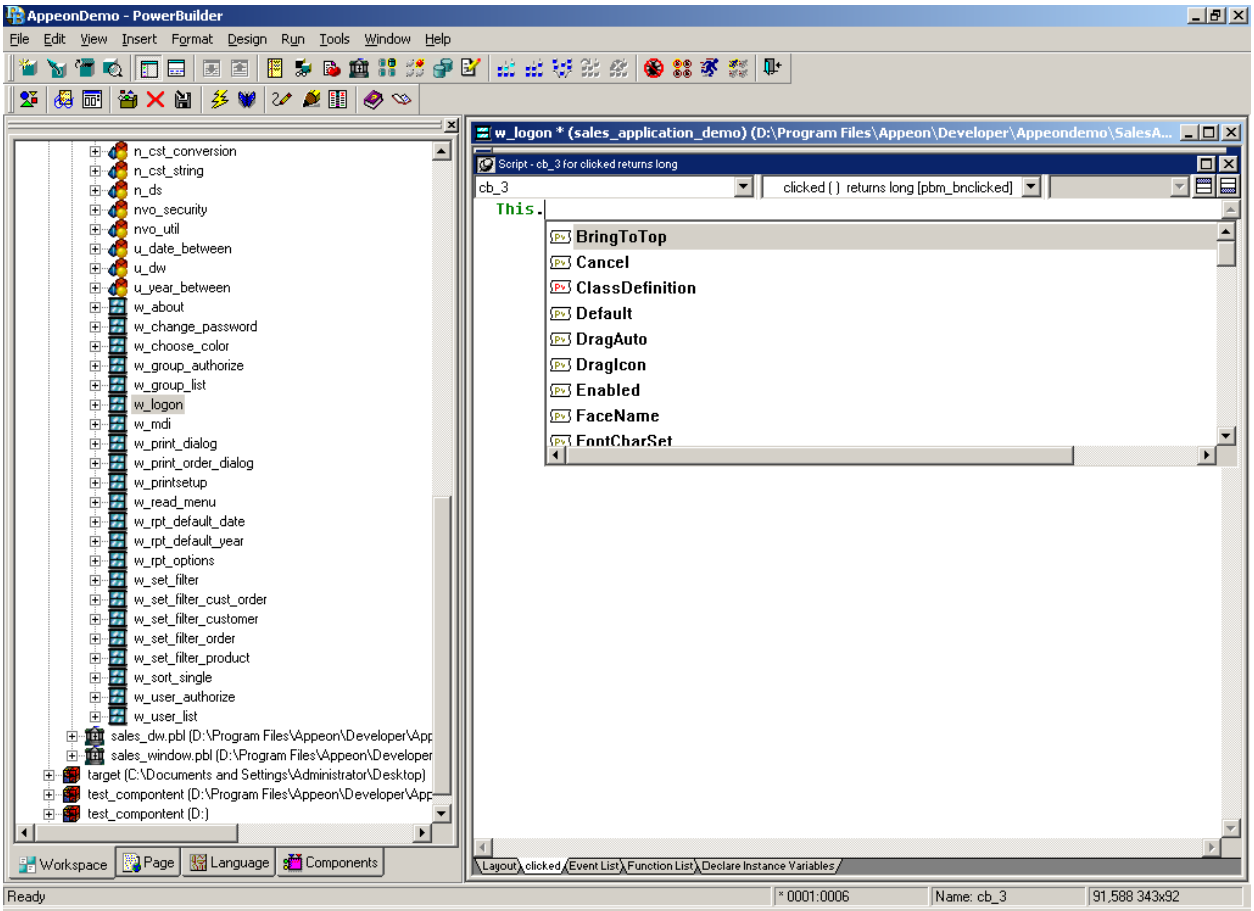
Step 2 – Save the object by clicking the *Save* button on the PowerBar.

Note: Saving the object enables Code Insight to generate the supported and unsupported feature list for the new content. Code Insight cannot detect the new content unless the object is saved.

Step 3 – Double click the button to open the Script view and pause for a few seconds after typing an identifier followed by a dot (for example, pause after "This.").

If there is more than one property, variable, method, or statement that could be inserted, the Code Insight feature list pops up for you to select a supported property, variable or method, as shown in Figure 12-4.

Figure 12-4: Develop an application with Code Insight

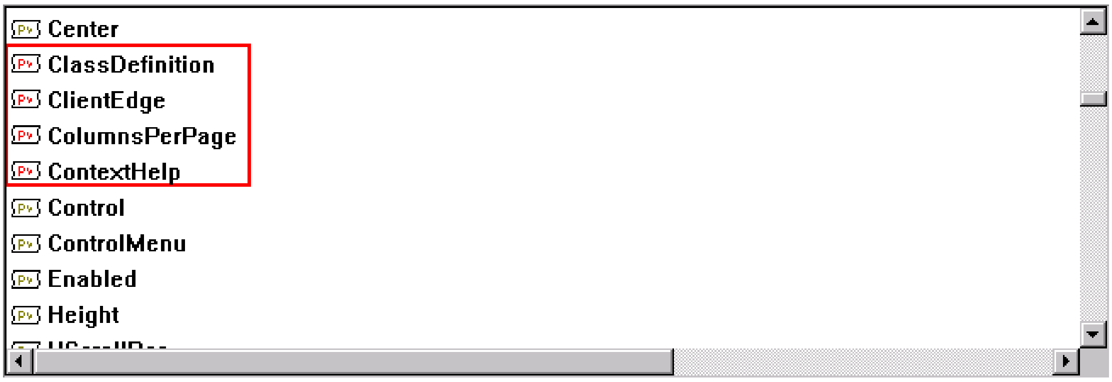


```
This.
```

BringToTop
Cancel
ClassDefinition
Default
DragAuto
DragIcon
Enabled
FaceName
FontCharSet

The unsupported properties, variables or methods in the popup list will have a red icon in front, as shown in Figure 12-5. Avoid using the unsupported features.

Figure 12-5: Appeon unsupported features displayed in popup window



Center
ClassDefinition
ClientEdge
ColumnsPerPage
ContextHelp
Control
ControlMenu
Enabled
Height

Step 4 – Select a supported property, variable or method from the popup list.

You can use Code Insight to help develop an application free of unsupported features and suitable for migration.

13 Launching Appeon Enterprise Manager

Appeon Enterprise Manager (AEM) is a Web-based application that manages the converted Web applications and Appeon Server over the Internet, an intranet or an extranet. It includes a comprehensive set of easy-to-use tools for system configuration, system maintenance, performance optimization, and application security.

AEM is included when Appeon Server is installed.

AEM is accessible from any Client's Web browser or it is possible to access AEM through the *AEM* button that resides on the Appeon Developer toolbar. Clicking the *AEM* button will automatically open a Web browser window and load the AEM entry page.

13.1 Requirements

Before launching AEM, verify that:

1. The Appeon Server hosting AEM has been started.
2. You have correctly specified the host name and port of the Appeon Server in the Appeon Server Profile Configuration page.

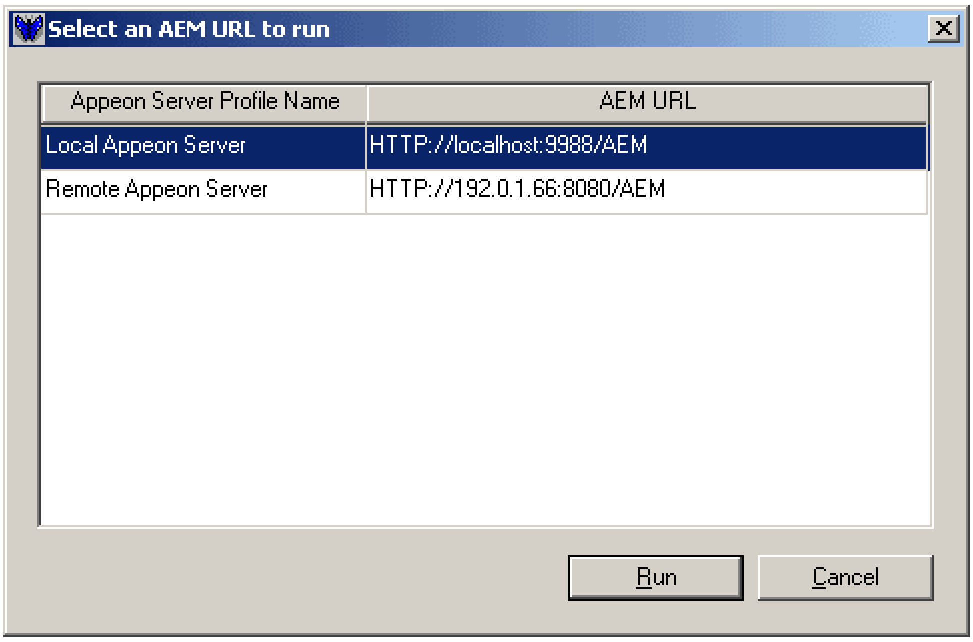
13.2 Launching Appeon Enterprise Manager

Step 1 – Click the *AEM* button () on the Appeon Developer toolbar.

If more than one Appeon Server profile is configured, the "Select an AEM URL to run" page is displayed, as shown in Figure 13-1. All Appeon Server profiles and corresponding AEM URLs are listed. The Appeon Server configured as the default server is highlighted and selected.

If only one Appeon Server profile is configured, the entry page of AEM is displayed, as shown in Figure 13-2.

Figure 13-1: Select an AEM URL to run



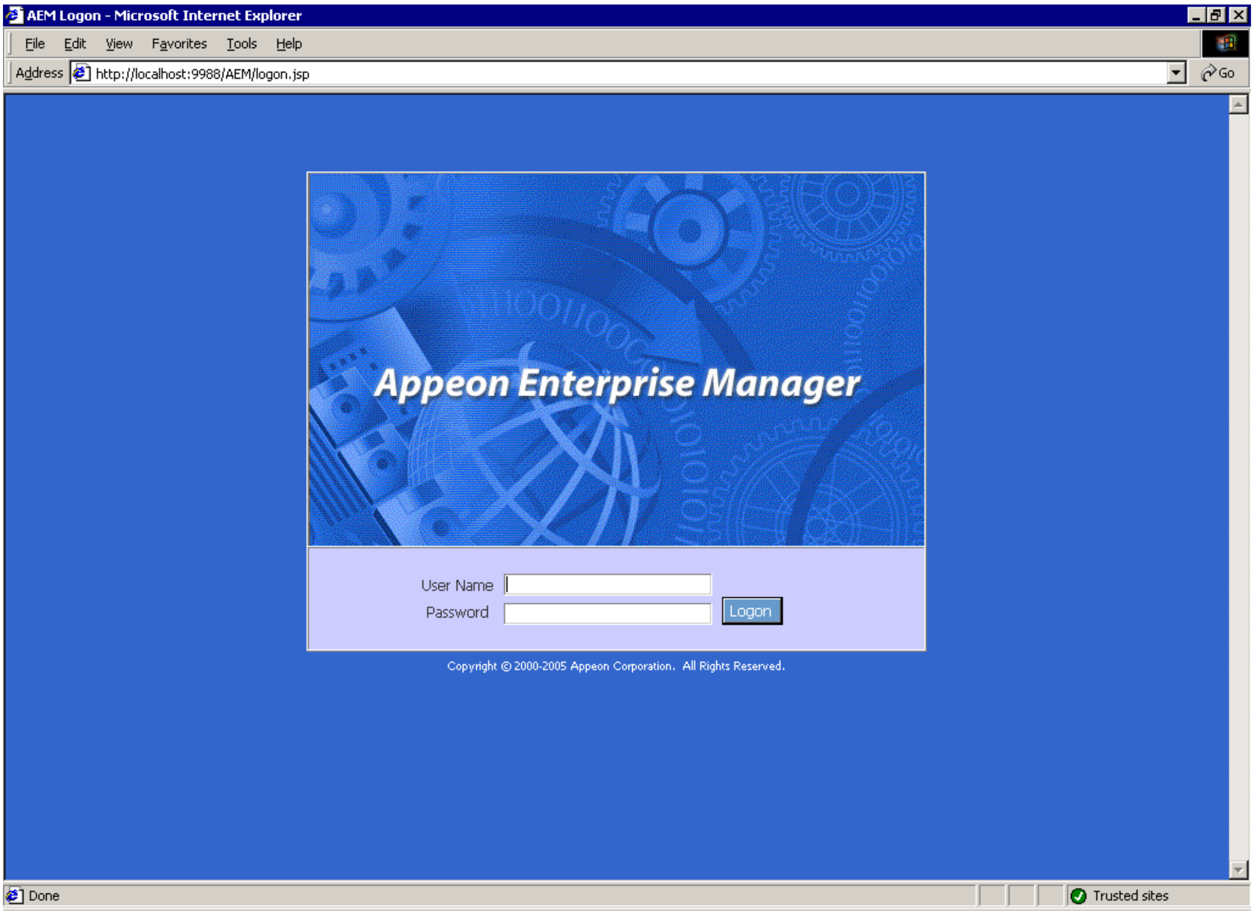
Select an AEM URL to run

Appeon Server Profile Name	AEM URL
Local Appeon Server	HTTP://localhost:9988/AEM
Remote Appeon Server	HTTP://192.0.1.66:8080/AEM

Step 2 – Select an AEM URL and click *Run*, as shown in Figure 13-1.

The entry page of AEM is loaded in an Internet Explorer page, as shown in Figure 13-2.

Figure 13-2: Appeon Enterprise Manager entry page



AEM Logon - Microsoft Internet Explorer
File
Edit
View
Favorites
Tools
Help
Address
http://localhost:9988/AEM/logon.jsp
Go
Appeon Enterprise Manager
User Name
Password
Logon
Copyright © 2000-2005 Appeon Corporation. All Rights Reserved.
Done
Trusted sites

The AEM URL is automatically entered into the Internet Explorer address bar when you click the *Run* button. You can configure AEM's URL and connection method (https or http) in the Appeon Server Profile Configuration page.

The default username is "admin" and the default password is "admin". The username and password settings for AEM can be modified in Security settings in AEM. Refer to the *Appeon Server Configuration Guide* for more information.

14 Appeon for PowerBuilder Help

Appeon provides a comprehensive HTML help system that contains:

- Introduction to Appeon
- What's New in Appeon, the HTML version of the *New Features Guide*.
- Appeon Installation Guide
- Appeon Migration Guide
- Appeon Features Help, the HTML version of the *Supported Features Guide*
- Working with Appeon Developer, the HTML version of the *Appeon Developer User Guide*
- Appeon Server Configuration Guide
- Web Server Configuration Guide
- Appeon Troubleshooting Guide & Error Messages
- Appeon Workarounds Guide & Workarounds PBL Reference
- Appeon Performance Tuning Guide

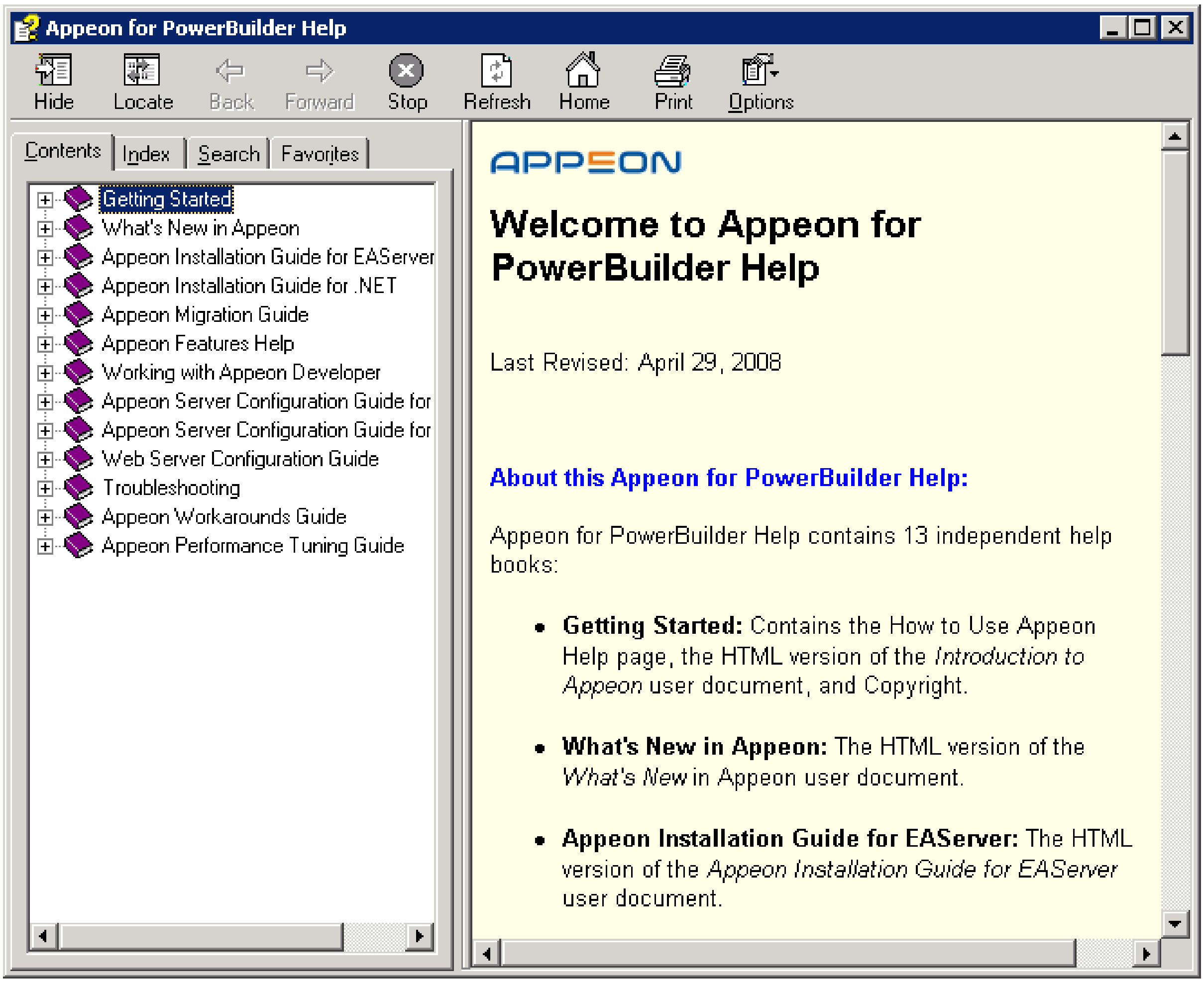
Appeon for PowerBuilder Help is in compiled HTML format, so that you can search by index or by typing keywords, navigate back and forth, or print a specified page.

To access Appeon HTML Help:

Step 1 – Click the *Help* button (📖) in the Appeon Developer toolbar.

Step 2 – Click the corresponding help book to access the detailed help information, as shown in Figure 14-1.

Figure 14-1: Appeon for PowerBuilder Help



Getting Started
What's New in Appeon
Appeon Installation Guide for EAServer
Appeon Installation Guide for .NET
Appeon Migration Guide
Appeon Features Help
Working with Appeon Developer
Appeon Server Configuration Guide for
Appeon Server Configuration Guide for
Web Server Configuration Guide
Troubleshooting
Appeon Workarounds Guide
Appeon Performance Tuning Guide

Welcome to Appeon for PowerBuilder Help

Last Revised: April 29, 2008

About this Appeon for PowerBuilder Help:

Appeon for PowerBuilder Help contains 13 independent help books:

- **Getting Started:** Contains the How to Use Appeon Help page, the HTML version of the *Introduction to Appeon* user document, and Copyright.
- **What's New in Appeon:** The HTML version of the *What's New* in Appeon user document.
- **Appeon Installation Guide for EAServer:** The HTML version of the *Appeon Installation Guide for EAServer* user document.

15 Extended Toolkit

15.1 Overview

Appeon provides extended tools to facilitate the Web migration process. These tools include:

- DLL/OCX Files Package Tool: It is designed as a solution for packaging and installing large-sized DLL/OCX files.
- Appeon Silent Installer (EAServer Windows only): Installs EAServer, Appeon Server, Appeon Server EBF (Emergency Bug Fix), and Appeon Web applications in the silent mode.
- Appeon Developer Toolbar Register Tool: Enables you to register (or unregister) one Appeon Developer version to more than one PowerBuilder version or different Appeon Developer versions to different PowerBuilder versions.

15.2 DLL/OCX Files Package Tool

15.2.1 Overview

DLL and OCX files are frequently used in PowerBuilder client/server application. Appeon offers you two ways to distribute DLL/OCX files.

Table 15-1: Two ways to distribute DLL/OCX files

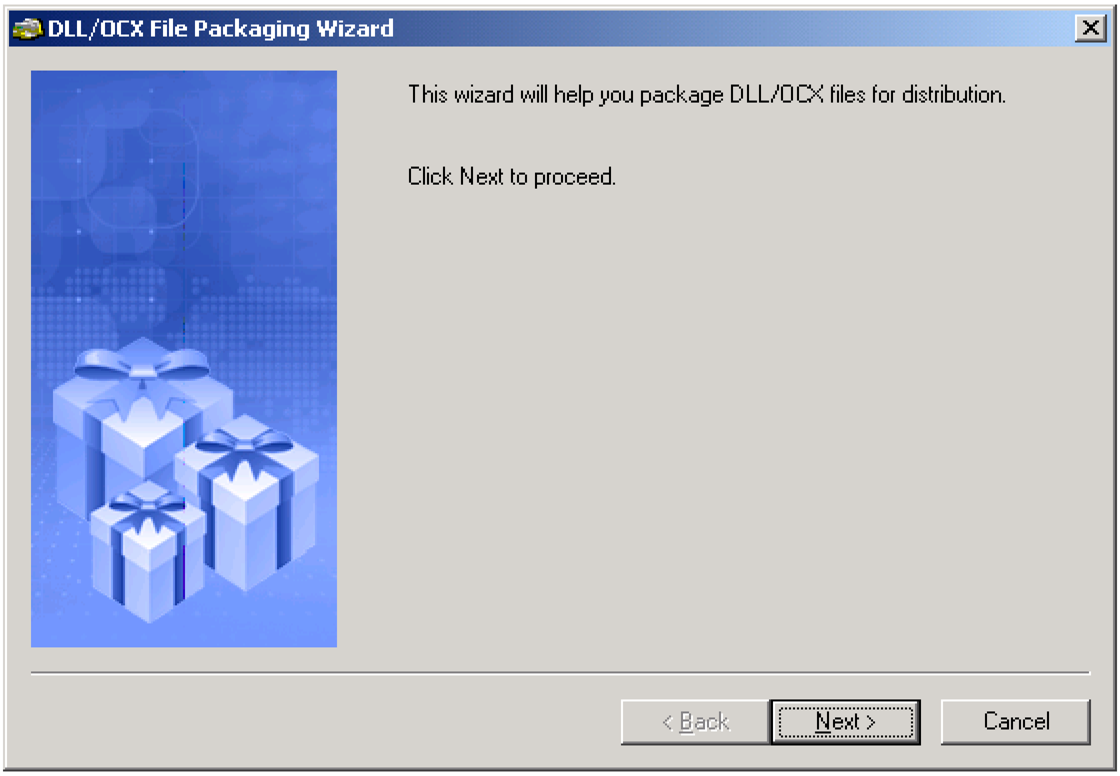
	When to use…	Which tool to use…
Deploy DLL/OCX files to the server and then download them to the client at runtime.	There are only a few DLL/OCX files or the files are small.	In the Additional Files tab of the Application Profiles Configuration window. Refer to the *Deploy External Non-System DLL/OCX Files* section.
Package DLL/OCX files and install them to the client before the application is run for the first time.	There are many DLL/OCX files or the files are large; and the client machine is fixed.	Use the DLL/OCX file packaging wizard. Refer to the following sections for detailed information.

15.2.2 Packaging DLL/OCX files

Step 1 – Run the EonDLLOCX.exe file under the \Toolkit\DLL&OCX Package folder in the Appeon Developer installation directory.

The DLL/OCX File Packaging Wizard starts, as shown in Figure 15-1. Click *Next* to proceed.

Figure 15-1: DLL/OCX File Packaging Wizard



This wizard will help you package DLL/OCX files for distribution.

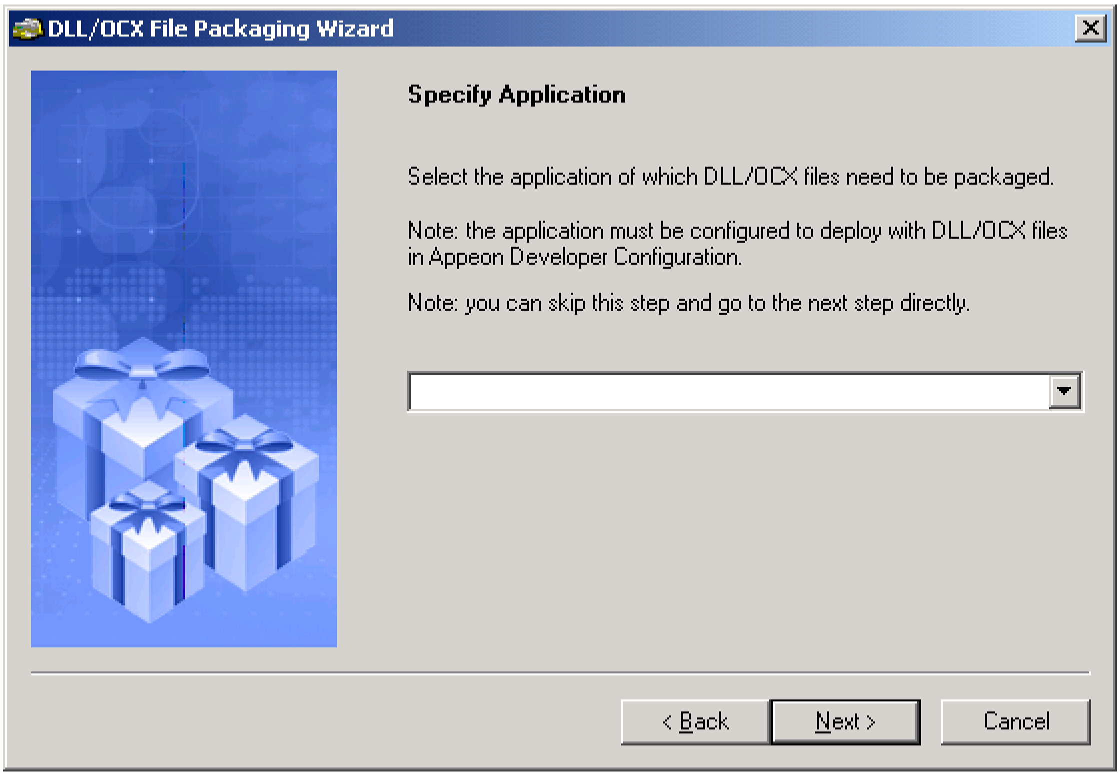
Click Next to proceed.

Step 2 – Select an application profile and click *Next* (Figure 15-2).

Only application profiles that are already configured with DLL/OCX files will be listed. For configuring DLL/OCX files for an application profile, refer to the *Deploy External Non-System DLL/OCX Files* section. When *Next* is clicked, the pre-configured DLL/OCX files will be automatically added for you.

Or, skip the selection and click *Next* to manually add the DLL/OCX files.

Figure 15-2: Select an application



DLL/OCX File Packaging Wizard

Specify Application

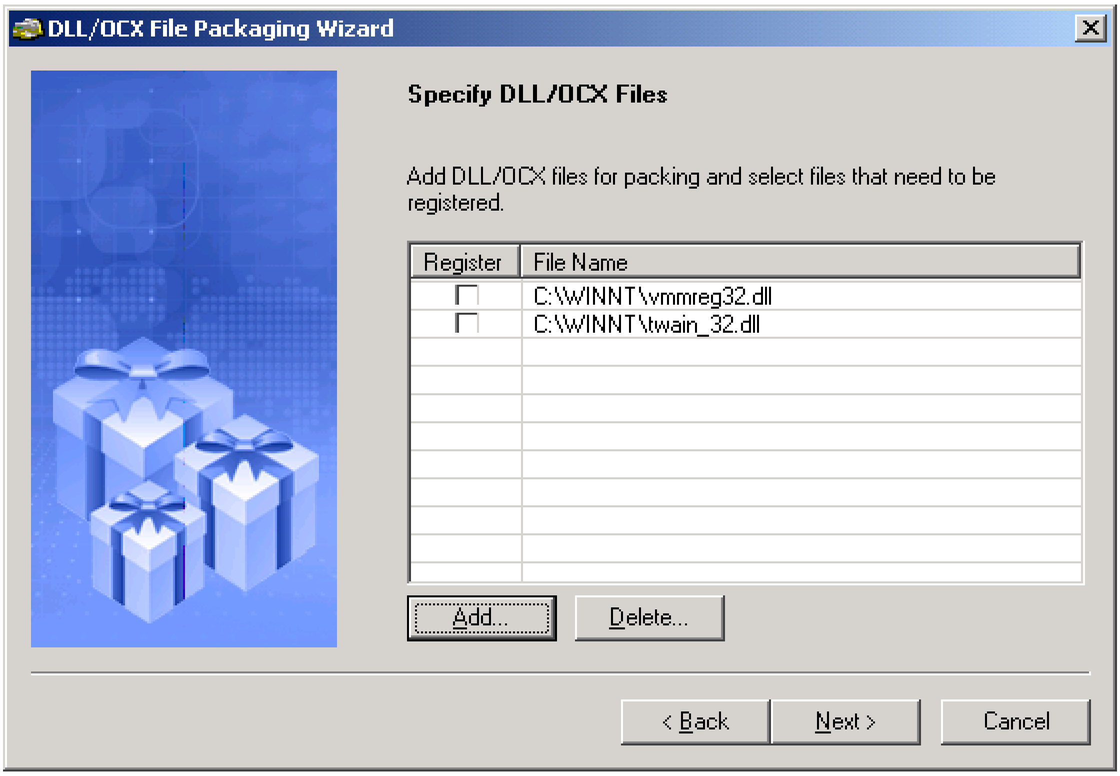
Select the application of which DLL/OCX files need to be packaged.

Note: the application must be configured to deploy with DLL/OCX files in Appeon Developer Configuration.

Note: you can skip this step and go to the next step directly.

Step 3 – Click *Add* to add the DLL/OCX files, select the *Register* box to automatically register the file when installed, and click *Next* to proceed (Figure 15-3).

Figure 15-3: Add DLL/OCX files



DLL/OCX File Packaging Wizard

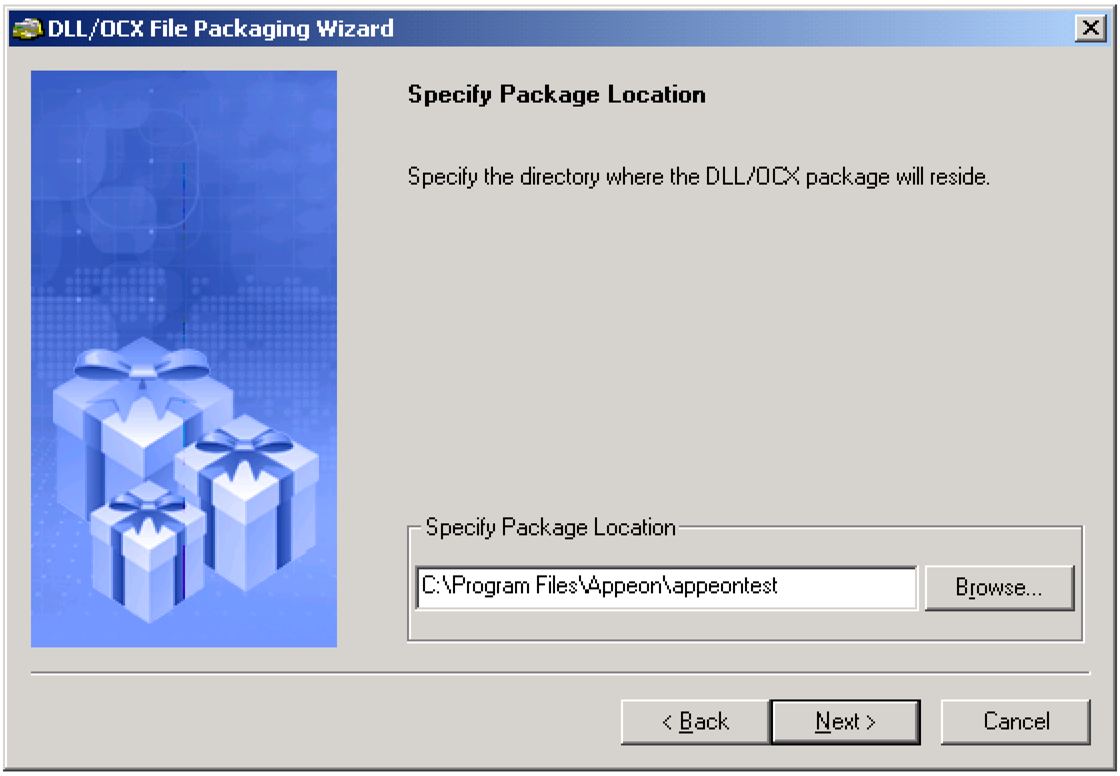
Specify DLL/OCX Files

Add DLL/OCX files for packing and select files that need to be registered.

Register	File Name
□	C:\WINNT\vmmreg32.dll
□	C:\WINNT\twain_32.dll

Step 4 – Specify the location where the package will be created (Figure 15-4). A folder named Package will be created under this location.

Figure 15-4: Specify the package location



DLL/OCX File Packaging Wizard

Specify Package Location

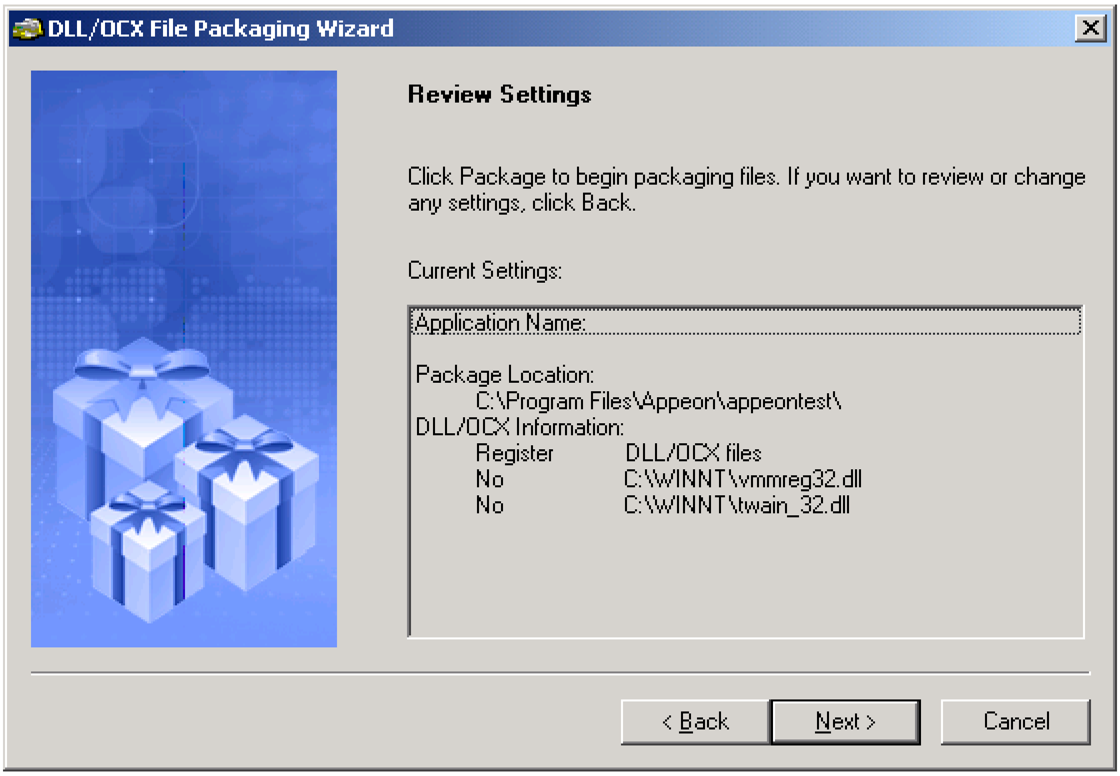
Specify the directory where the DLL/OCX package will reside.

Specify Package Location

C:\Program Files\Appeon\appeontest

Step 5 – Review the settings and click *Next* to generate the package (Figure 15-5).

Figure 15-5: Review settings



DLL/OCX File Packaging Wizard

Review Settings

Click Package to begin packaging files. If you want to review or change any settings, click Back.

Current Settings:

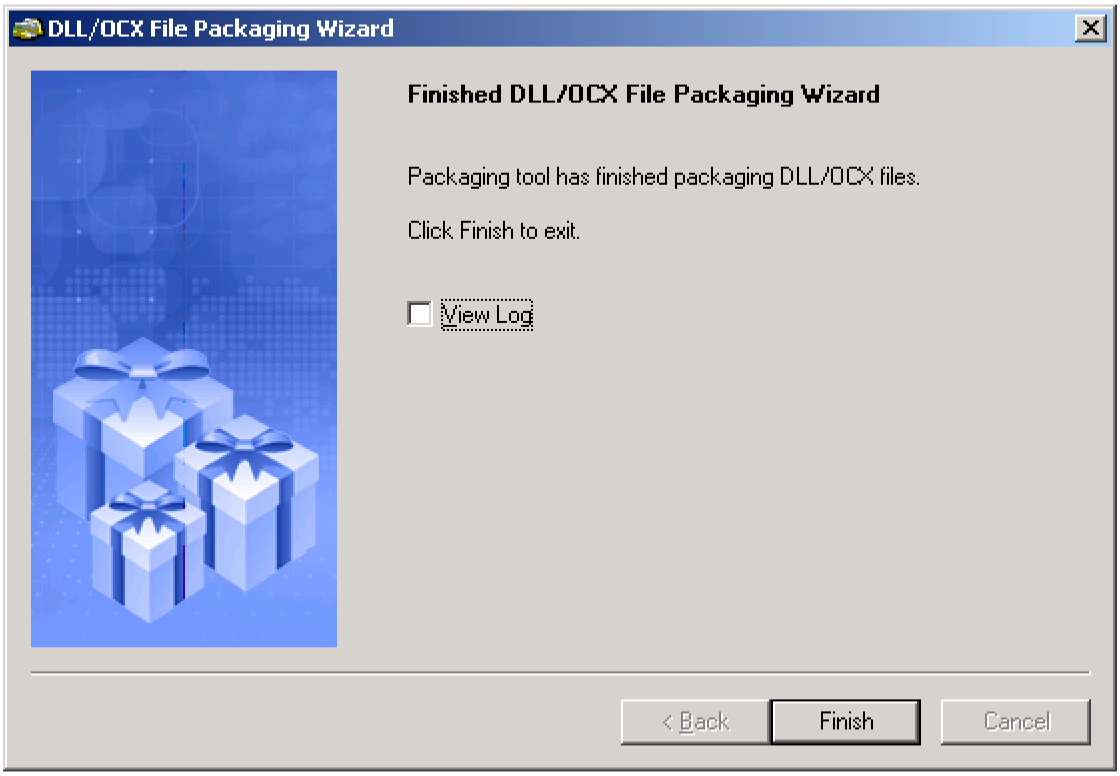
```
Application Name:

Package Location:
        C:\Program Files\Appeon\appeontest\
DLL/OCX Information:
        Register        DLL/OCX files
        No              C:\WINNT\vmmreg32.dll
        No              C:\WINNT\twain_32.dll
```

Step 6 – Wait until the file packaging process is complete.

Step 7 – When packaging is complete, click *Finish* to close the wizard (Figure 15-6).

To view the log information generated during the process, select the *View Log* box and then click *Finish*. The log file will be automatically opened. You can also manually open the DLLOCX.log file in the generated package.

Figure 15-6: Packaging complete



DLL/OCX File Packaging Wizard

Finished DLL/OCX File Packaging Wizard

Packaging tool has finished packaging DLL/OCX files.

Click Finish to exit.

□ View Log

15.2.3 Installing and Uninstalling DLL/OCX files

You will use the same setup program (Setup.exe file) in the generated package for installing and uninstalling DLL/OCX files:

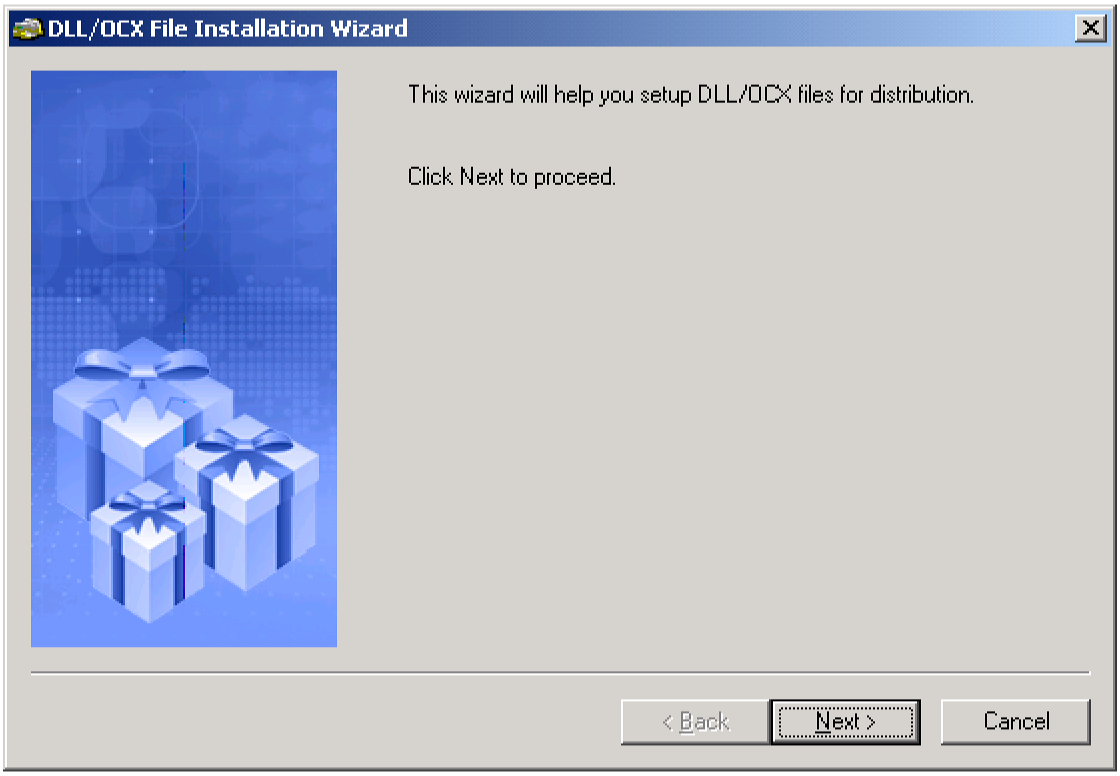
- If the DLL/OCX files are not installed on the client, when you run the setup program, you will be guided to install the files. Refer to Section 15.2.3.a: *Installing DLL/OCX files* for detailed instructions.
- If the DLL/OCX files are already installed on the client, when you run the setup program, you will be guided to reinstall or uninstall the files. If you select to reinstall the files, you will run through the same process as a new installation, as described in Section 15.2.3.a: *Installing DLL/OCX files*. If you select to uninstall the files, refer to Section 15.2.3.b: *Uninstalling DLL/OCX files* for detailed instruction.

15.2.3.a Installing DLL/OCX files

Step 1 – On a client machine, run the Setup.exe file in the generated package.

The DLL/OCX File Installation Wizard starts, as shown in Figure 15-7. Click *Next* to proceed.

Figure 15-7: DLL/OCX File Installation Wizard



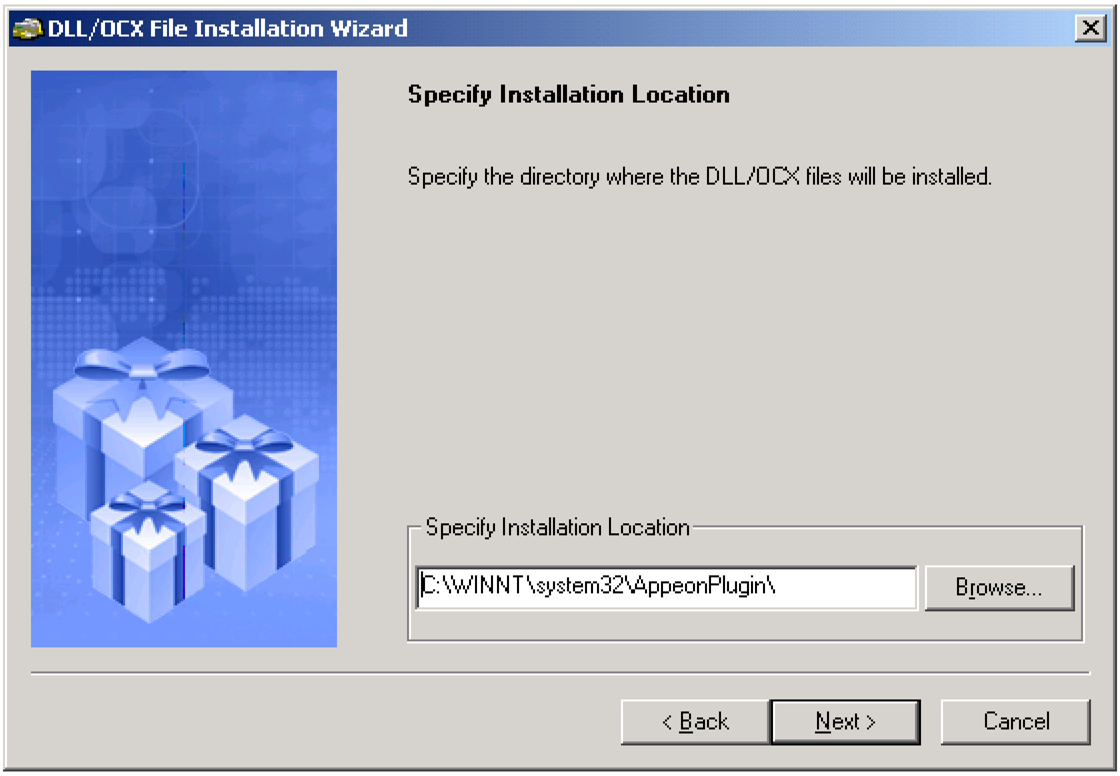
DLL/OCX File Installation Wizard

This wizard will help you setup DLL/OCX files for distribution.

Click Next to proceed.

Step 2 – Specify the location where the DLL/OCX files will be installed. Click *Next* to start the installation process (Figure 15-8).

Figure 15-8: Specify the installation location



DLL/OCX File Installation Wizard

Specify Installation Location

Specify the directory where the DLL/OCX files will be installed.

Specify Installation Location: C:\WINNT\system32\AppeonPlugin\

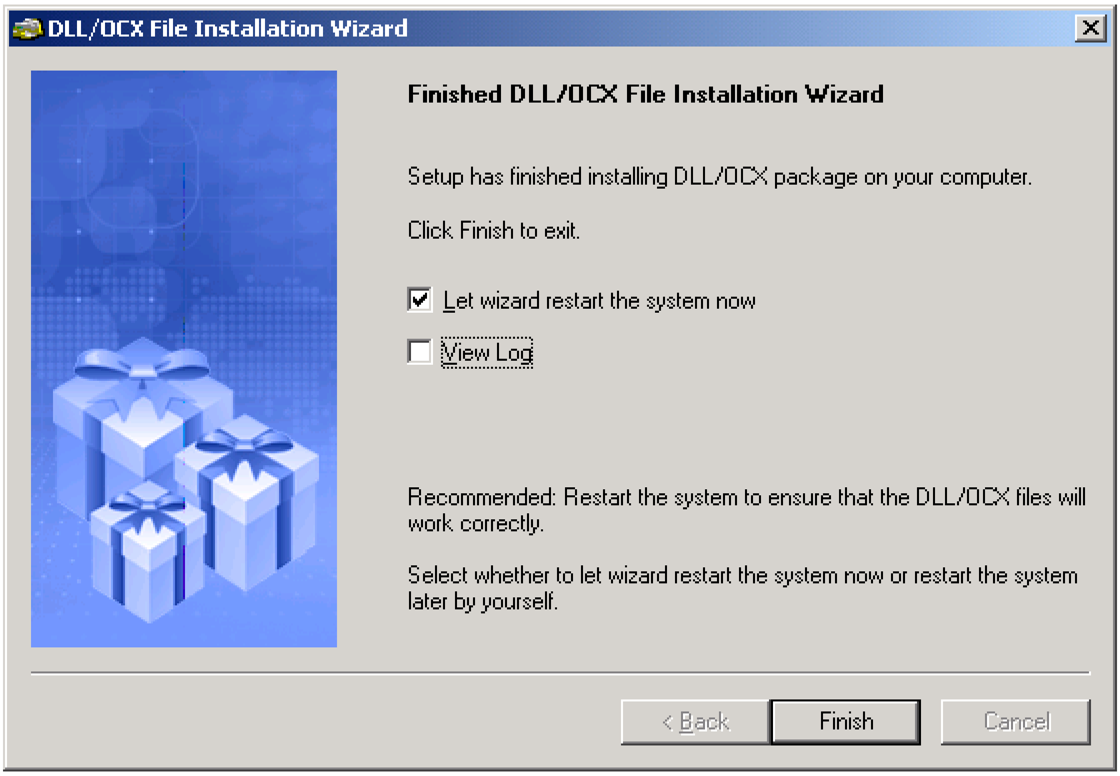
Step 3 – Wait until the installation is complete.

During the installation process, the DLL/OCX files will first be copied to the specified location and then registered if they have been selected for auto-registration when packaged.

Step 4 – When the installation is complete, select the *Let wizard restart the system now* option and click *Finish* to close the installation wizard and restart your machine (Figure 15-9).

To view the log information generated during the process, select the *View Log* box and click *Finish*. The log file will be automatically opened. You can also manually open the DLLOCX.log file in the generated package.

Figure 15-9: Installation complete



DLL/OCX File Installation Wizard

Finished DLL/OCX File Installation Wizard

Setup has finished installing DLL/OCX package on your computer.

Click Finish to exit.

☑ Let wizard restart the system now

□ View Log

Recommended: Restart the system to ensure that the DLL/OCX files will work correctly.

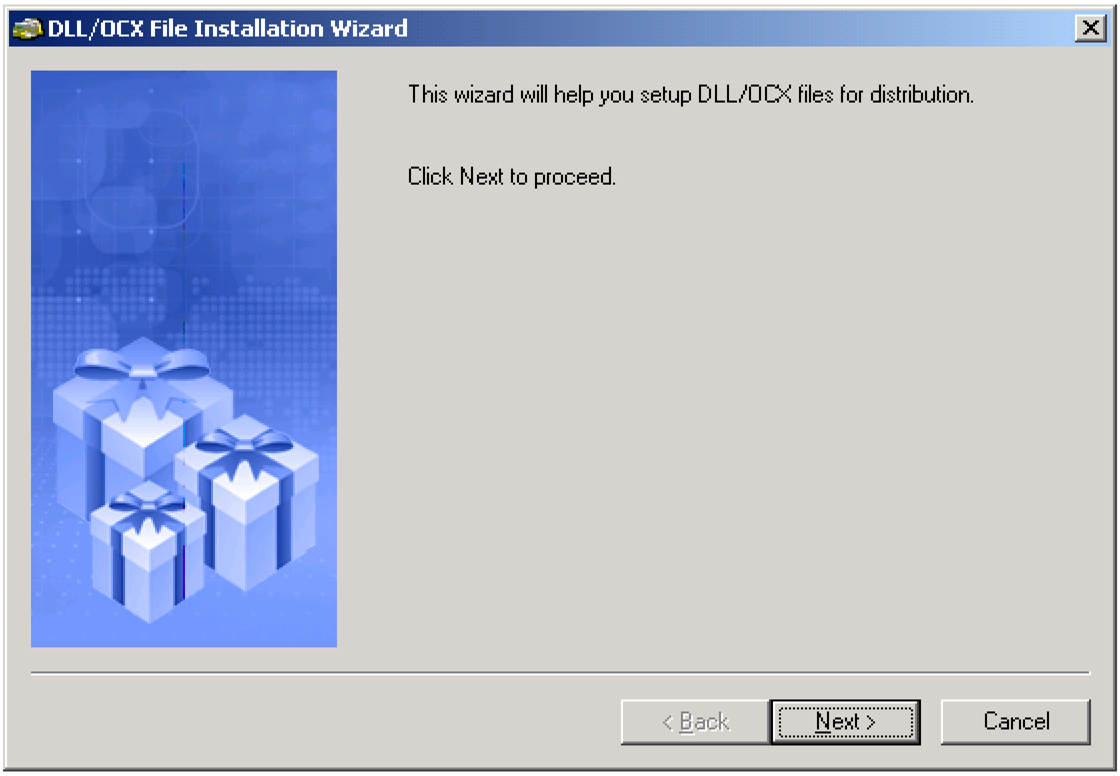
Select whether to let wizard restart the system now or restart the system later by yourself.

15.2.3.b Uninstalling DLL/OCX files

Step 1 – On a client machine, run the Setup.exe file in the generated package.

The DLL/OCX File Installation Wizard starts, as shown in Figure 15-10. Click *Next* to proceed.

Figure 15-10: DLL/OCX File Installation Wizard



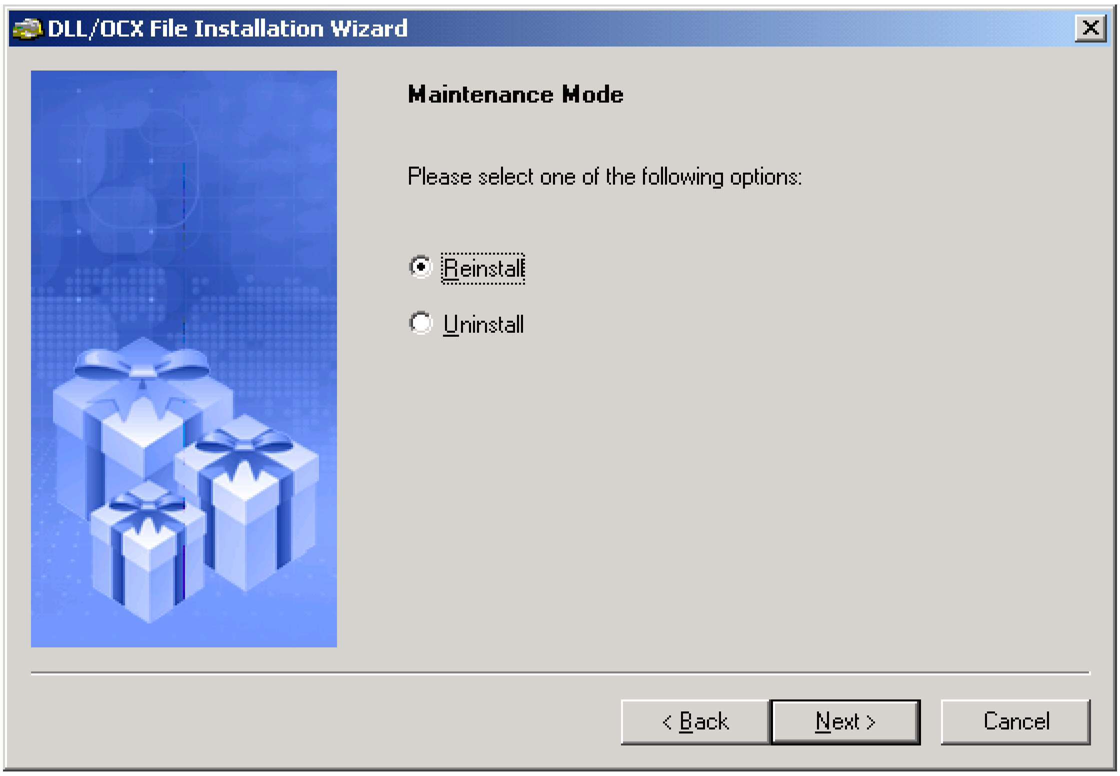
This wizard will help you setup DLL/OCX files for distribution.

Click Next to proceed.

Step 2 – Select *Uninstall* and click *Next* to proceed.

If you select to reinstall the files, you will run through the same process as a new installation, as described in Section 15.2.3.a: *Installing DLL/OCX files*.

Figure 15-11: Reinstall or uninstall



Maintenance Mode

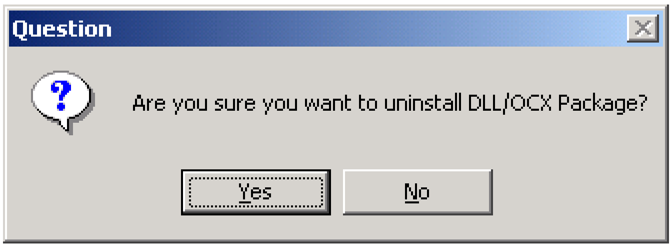
Please select one of the following options:

● Reinstall

○ Uninstall

Step 3 – Click *Yes* to confirm the uninstall (Figure 15-12).

Figure 15-12: Uninstall confirmation



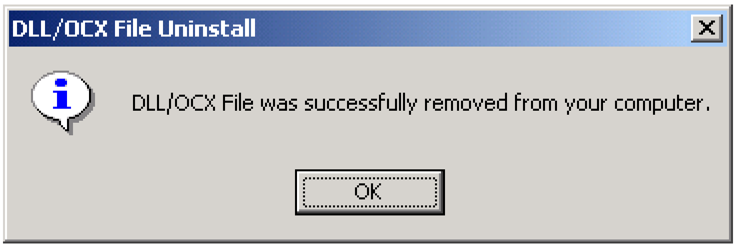
Question

Are you sure you want to uninstall DLL/OCX Package?

Step 4 – Wait until the uninstall is complete.

Step 5 – When uninstall is complete, click *OK* to exit (Figure 15-13).

Figure 15-13: Uninstall complete



DLL/OCX File Uninstall

DLL/OCX File was successfully removed from your computer.

15.3 Appeon Silent Installer (Windows EAServer only)

15.3.1 Overview

Note: this tool does not work in WebLogic and WebSphere versions.

To help ISVs create an easy-to-use Appeon installation package for the end users, Appeon specially provides a silent installer which can be customized to install EAServer, Appeon Server, Appeon Server EBF (Emergency Bug Fix), and Appeon Web applications in the silent mode.

This guide provides some useful instructions for **ISVs** on:

- Creating folder structure of Silent Installer
- Copying installation program of EAServer, Appeon Server, Appeon Server EBF or Web applications
- Customizing configuration file
- Burning installation CDs

Additional instructions are also provided for **end users** on:

- Running Appeon Silent Installer

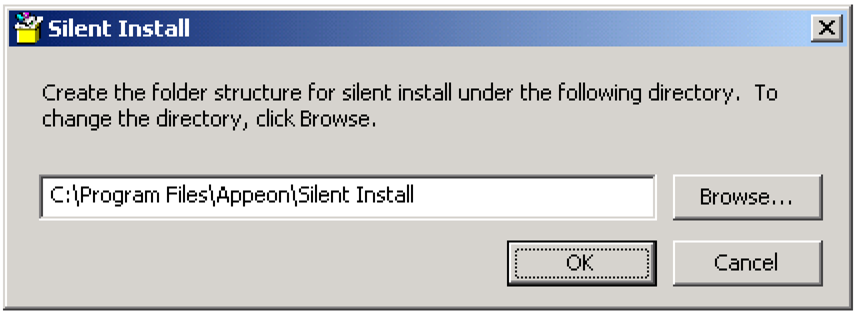
15.3.2 Creating folder structure of Silent Installer

The installation program files of EAServer, Appeon Server, Appeon Server EBF and Web application must be placed in the required folder structure, so that Appeon Silent Installer can run them. For your convenience, Appeon provides a compressed folder structure which can be decompressed under any directory. To decompress and create the folder structure and the required files, take the following steps:

Step 1 – Run SilentInstall.exe in the "\Toolkit\Silent Install" under the Appeon Developer installation directory, for example, C:\Program Files\Appeon\Developer6.1\Toolkit\Silent Install.

Step 2 – Select where to create the folder structure. Click OK.

Figure 15-14: Specify the location



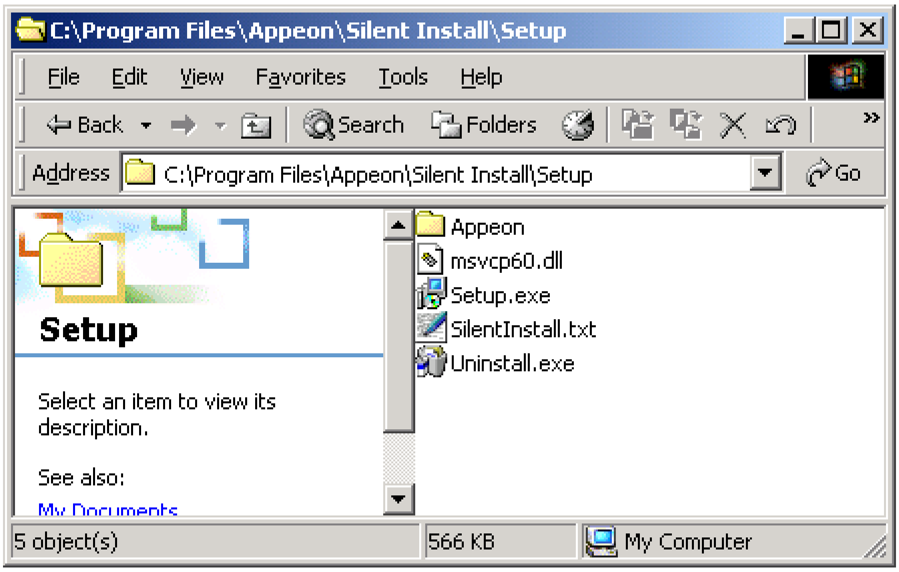
Silent Install

Create the folder structure for silent install under the following directory. To change the directory, click Browse.

C:\Program Files\Appeon\Silent Install

After the creation, a folder named "Setup" is created with the following sub-folders and files:

Figure 15-15: Folder structure



C:\Program Files\Appeon\Silent Install\Setup

Address: C:\Program Files\Appeon\Silent Install\Setup

Setup

Select an item to view its description.

See also:
My Documents

Appeon
msvcp60.dll
Setup.exe
SilentInstall.txt
Uninstall.exe

5 object(s) | 566 KB | My Computer

The following is a brief introduction to each folder in the Appeon Silent Installer:

- Setup: The root directory of the Silent Installer. It contains the "Appeon" folder, the Setup.exe, Uninstall.exe, SilentInstall.txt, and msvcp60.dll files.
  - Appeon: It contains the "AS", "AS EBF", "EAServer" and "WebApp" directory.
    - AS: It stores the installation program of Appeon Server.
    - AS EBF: It stores the installation program of Appeon Server EBF.
    - EAServer: It stores the installation program of EAServer.
    - WebApp: It stores the Web application files packaged by the Appeon application packaging tool.
  - Setup.exe: The setup program for launching the silent installation.
  - SilentInstall.txt: The configuration file which can be modified to customize the installation.

- Uninstall.exe: The uninstall program for silently removing what has been installed with Appeon Silent Installer.
- msvcp60.dll: The DLL file for Appeon Silent Installer.

15.3.3 Copying installation program of EAServer, Appeon Server, Appeon Server EBF or Web applications

After the required folders and files are generated, you can start copying the installation program files of Appeon Server, EAServer and Web applications to the corresponding folder.

To copy the EAServer installation program to Appeon Silent Installer:

Copy all files and folders from the directory where setup.exe file resides to the “EAServer” folder of Appeon Silent Installer.

To copy the Appeon Server installation program to Appeon Silent Installer:

Copy all files from the “ASP” folder on the Appeon installation CD to the “AS” folder of Appeon Silent Installer.

To copy the Appeon Server EBF installation program to Appeon Silent Installer:

Copy all files from the "asebf" folder on the Appeon installation CD to the "AS EBF" folder of Appeon Silent Installer.

To copy Web application packages to Appeon Silent Installer:

Step 1 – Verify that the Web application packages have already been generated using the Appeon application packaging tool. If no package is generated, follow steps in *Packaging a Web application*.

Step 2 – Copy the generated packages to the “WebApp” folder of Appeon Silent Installer.

15.3.4 Customizing configuration file

With Appeon Silent Installer, you can customize the installation program using the configuration file (SilentInstall.txt). The following is a detailed description of each parameter in the SilentInstall.txt file. Each parameter contains no default value.

15.3.4.a Installation Types

Specify whether to install EAServer, Appeon Server and Web applications.

Parameter	EAServer
Description	Specifies whether to install EAServer. If you do not want to install EAServer using this setup program, make sure that a supported EAServer was installed already.
Accepted Values	Yes – Install EAServer. No – Not to install EAServer, instead use an existing EAServer.
Default Value	None
Example	EAServer ="yes"

Parameter	Appeon Server
Description	Specifies whether to install Appeon Server.

Accepted Values	Yes No
Default Value	None
Example	Appeon Server="yes"

Parameter	Appeon Server EBF
Description	Specifies whether to apply Appeon Server EBF.
Accepted Values	Yes No
Default Value	None
Example	Appeon Server EBF="yes"

Parameter	WebApplication
Description	Specifies whether to install Web applications.
Accepted Values	Yes No
Default Value	None
Example	WebApplication="yes"

15.3.4.b EAServer and ASA Path

Specify the path of EAServer and ASA. ASA is required because Appeon Server has an ASA database for storing the sensitive system data.

Parameter	EAServerPath
Description	If the *EAServer* parameter is set to “No”, this specifies the path of an existing EAServer. The specified path will be validated. If the *EAServer* parameter is set to “Yes”, this indicates where EAServer will be installed.
Accepted Values	A valid path
Default Value	None
Example	EAServerPath="C:\Program Files\Sybase\EAServer"

Parameter	ASAPath
Description	Specifies the ASA installation path. The specified path will be validated.
Accepted Values	A valid path
Default Value	None
Example	ASAPath="C:\Program Files\Sybase\Shared\ASA802"

15.3.4.c EAServer Host Name, Port, Username and Password

Specify the EAServer host name, port number, user name and password.

Parameter	EAServerName
Description	Specifies the IP address of EAServer.
Accepted Values	A valid IP address
Default Value	None
Example	EAServerName="localhost"

Parameter	EAServerPort
Description	Specifies the port number of EAServer.
Accepted Values	A valid port
Default Value	None
Example	EAServerPort="9000"

Parameter	EAServerUserName
Description	Specifies the user name to connect to EAServer.
Accepted Values	A valid user name on EAServer.
Default Value	None
Example	EAServerUserName="jagadmin"

Parameter	EAServerPassWord
Description	Specifies the password to connect to EAServer.
Accepted Values	A valid password for the current user name.
Default Value	None
Example	EAServerPassWord=""

15.3.4.d Appeon Server Port, Username and Password

Specify the Appeon Server port number, user name and password.

Parameter	AppeonServerPort
Description	Specifies the HTTP port number of Appeon Server.
Accepted Values	A valid port If both *EAServer* and *AppeonServer* parameters are set to “Yes”, port “9988” is the only available port that can be used.
Default Value	None
Example	AppeonServerPort="9988"

Parameter	AppeonServerUserName
Description	Specifies the user name for deploying an application to Appeon Server. This deployment user name can be created using the Security tool in AEM.
Accepted Values	A valid deployment user name on Appeon Server.
Default Value	None
Example	AppeonServerUserName=""

Parameter	AppeonServerPassWord
Description	Specifies the password for deploying an application to Appeon Server.
Accepted Values	A valid deployment password for the current user name.
Default Value	None
Example	AppeonServerPassWord=""

15.3.4.e Web Server Path

Specify the Web root of the Web server.

Parameter	WebServerRootPath
Description	Specifies the document root of the Web server. If the *WebApplication* parameter is set to “No”, this parameter will not take effect. If the *WebApplication* parameter is set to “Yes”, the specified document root will be validated.
Accepted Values	A valid path
Default Value	None
Example	WebServerRootPath="C:\Program Files\Sybase\EAServer\html"

15.3.4.f Setup Log Path

Specify the log file path.

Parameter	LogPath
Description	Specifies the path for the setup log file (APBLog.txt).
Accepted Values	A valid path
Default Value	None
Example	LogPath="c:\log"

15.3.4.g Web Applications

Specify the number of Web applications, application name, connection cache, transaction object and database type.

Parameter	WebApplications

Description	Specifies the number of Web applications that you want to install.
Accepted Values	Any number greater than zero
Default Value	None
Example	WebApplications="2"

If more than one application will be installed, create the following parameters through incrementing by “1”, for example, “WebApplication2”, “EAServerNVO2”, “ConnectCache2” and “AEM2” for the second application, “WebApplication3”, “EAServerNVO3”, “ConnectCache3” and “AEM3” for the third application etc.

Parameter	WebApplication1
Description	Specifies the actual name of the first Web application.
Accepted Values	An alphanumeric string
Default Value	None

Example	Examples for the first application:
	WebApplication1 = "appeon_sales_application_demo"
	Examples for the second application:
	WebApplication2 = "appeon_code_examples"

Parameter	EAServerNVO1
Description	Specifies whether to install NVO components.
Accepted Values	Yes No
Default Value	None
Example	EAServerNVO1=”Yes”

To make connection caches work on the target Appeon Server, be sure to install the required database drivers to the target Appeon Server. For more information on database drivers, refer to the *Database Connection Setup* section in the *Appeon Server Configuration Guide*.

Parameter	ConnectCache1
Description	Specifies whether to install connection caches.
Accepted Values	0 – Not to install connection caches. 1 – Install and overwrite existing connection caches. 2 – Install but not to overwrite existing connection caches.
Default Value	None
Example	ConnectCache1=”1”

Parameter	AEM1
Description	Specifies whether to install AEM settings.

Accepted Values	Yes No
Default Value	None
Example	AEM1=”Yes”

15.3.4.h Web Application Uninstall Program Path

Specify the directory for the Web application uninstall program. A shortcut for the uninstall program will be created in Windows Start menu.

Parameter	UninstallWebAppPath
Description	Specifies where the uninstall program for Web applications will reside.
Accepted Values	A valid path.
Default Value	None
Example	UninstallWebAppPath=”C:\uninstallwebapp”

15.3.4.i Resource Language

Specify the language of the messages displayed during the installation or uninstallation process.

Parameter	Language
Description	Specifies the language of UI messages.
Accepted Values	EN -- English JP -- Japanese
Default Value	EN
Example	Language=”EN”

15.3.4.j SilentInstall.txt example

The following example shows a sample SilentInstall.txt file.

```
####################################################################
# Configure installation type: Yes or No
####################################################################
EAServer ="Yes"
Appeon Server="Yes"
Appeon Server EBF="Yes"
WebApplication="Yes"

####################################################################
#  Enter the path of the EAServer installed or you want to install.
####################################################################
EAServerPath="C:\Program Files\Sybase\EAServer"

####################################################################
```

```
#  Enter the ASA path
####################################################################
ASAPath=""C:\Program Files\Sybase\Shared\ASA802"

####################################################################
# Enter the Port, Username, and Password of EAServer
####################################################################
EAServerName=”localhost”
EAServerPort="9000"
EAServerUserName="jagadmin"
EAServerPassWord=""

####################################################################
# Enter the Port, Username, and Password of Appeon Server
####################################################################
AppeonServerPort="9988"
AppeonServerUserName=""
AppeonServerPassWord=""

####################################################################
# Enter the Web root of Web server
####################################################################
WebServerRootPath="C:\Program Files\Sybase\EAServer\html"

####################################################################
# Enter the path of Setup Log: APBLog.txt
####################################################################
LogPath="c:\log"

####################################################################
# Enter the Web application number
####################################################################
WebApplications="1"

####################################################################
# Web Application Configuration
# ConnectCache:
#              0: not install
#              1: install, overwrite if exist
#              2: install, not overwirte if exist
# EAServerNVO: Yes or No
# AEM: Yes or No
####################################################################
WebApplication1="Web Application Name1"
EAServerNVO1=”YES”
AEM1=”YES”
ConnectCache1=”1”
####################################################################
```

```
# Uninstall Path
####################################################################
UninstallWebAppPath = "C:\uninstallwebapp"

####################################################################
# Language
#[default] = "EN"
#  "EN"  English
#  "JP"  Japanse
####################################################################
Language="EN"
```

15.3.5 Burning installation CDs

After configuration is completed, you can burn installation CDs with the customized silent installer (the entire "setup" folder) and distribute CDs to your end users.

The Appeon Silent Installer can be provided in one, two or three CDs:

- In one CD: Installing the Appeon Server ("AS" folder), Appeon Server EBF ("AS EBF" folder), EAServer ("EAServer" folder) and the Web application ("WebApp" folder).
- In two CDs: One for installing the Appeon Server ("AS" folder), Appeon Server EBF ("AS EBF" folder), and EAServer ("EAServer" folder), and the other for installing the Web application ("WebApp" folder).
- In three CDs: One for installing the Appeon Server ("AS" folder) and Appeon Server EBF ("AS EBF" folder), one for installing EAServer ("EAServer" folder), and one for installing the Web application ("WebApp" folder).

Each CD must contain the following four files right under the Setup folder (that is, place the following files in the same directory level of the "AS", "EAServer" and "WebApp" folder):

- Setup.exe
- SilentInstall.txt
- Uninstall.exe
- msvcp60.dll

15.3.6 Running Appeon Silent Installer (for End Users)

This section provides some additional instructions for **end users** on using the Appeon Silent Installer to

- Install EAServer, Appeon Server, Appeon Server EBF and/or Web applications
- Uninstall EAServer, Appeon Server, Appeon Server EBF and/or Web applications

15.3.6.a Installing EAServer, Appeon Server, Appeon Server EBF and/or Web applications

Before running the Appeon Silent Installer, be aware that

1) The Setup.exe program can only be run on the Windows platform.

2) The Appeon Server and the Web server must reside on the same machine.

Start the silent installation by running the Setup.exe file on the installation CD.

By default, the configuration file in the same location of the Setup.exe file will be used during the installation process. To use a configuration file residing in a different location, run the Setup.exe file from the command line and append the configuration file location, for example, *Setup.exe d:/setuptmp/SilentInstall.txt*.

15.3.6.b Uninstalling EAServer, Appeon Server, Appeon Server EBF and/or Web applications

To silently uninstall EAServer, Appeon Server, Appeon Server EBF, and/or Web applications:

Run Uninstall.exe on the installation CD. The uninstall program will automatically load the SilentInstall.txt file on the same CD and remove what has been installed with Appeon Silent Installer.

To uninstall Web applications from Windows Start menu:

Select Windows Start | Programs | Appeon Web Application | Undeploy Appeon Web Application. Or run the uninstall program under the location specified by the "UninstallWebAppPath" parameter.

You will go through the same process as those in the Uninstalling an application section in the "Packaging and Installing Web Applications" chapter.

15.3.6.c FAQs

1. What happens if EAServer is started during the process of silent installation?

Appeon Silent Installer will shut down the EAServer and continue the installation.

2. What happens if EAServer or Appeon Server was already installed during the process of silent installation?

Appeon Silent Installer will overwrite the existing EAServer or Appeon Server.

3. What happens if the names of NVO packages for a Web application are duplicated?

Appeon Silent Installer will still deploy the first NVO package, but will not deploy the second NVO package. This activity will be recorded in the log file.

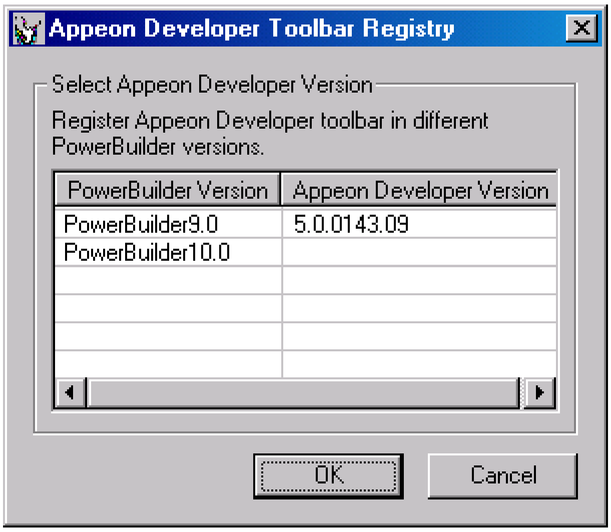
15.4 Appeon Developer Toolbar Register Tool

Appeon Developer can run with PowerBuilder 8, 9, 10, 10.5, 11 and 11.5 at the same time and multiple Appeon Developer versions can run on the same machine. The Appeon Developer Toolbar register tool enables you to register (or unregister) one Appeon Developer version to more than one PowerBuilder version or different Appeon Developer versions to different PowerBuilder versions, so that you can easily and flexibly select which Appeon Developer version and PowerBuilder version to use.

Following are steps to use the Appeon Developer Toolbar register tool:

Step 1 – Select Start | Programs | Appeon 6.1 for PowerBuilder | Appeon Developer | Toolbar Register Tool.

Figure 15-16: Registry tool



Appeon Developer Toolbar Registry

Select Appeon Developer Version

Register Appeon Developer toolbar in different PowerBuilder versions.

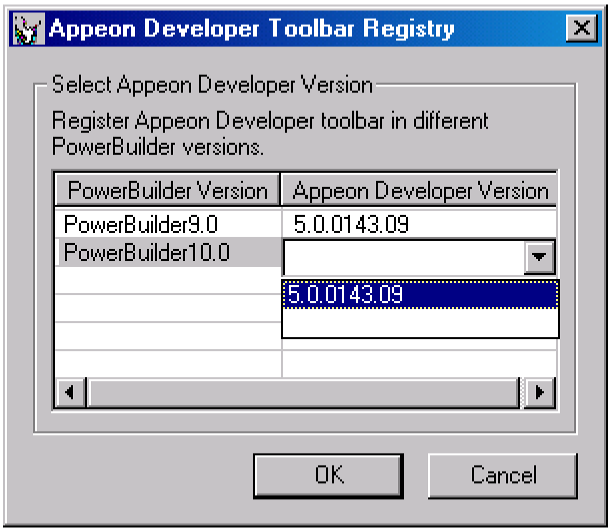
PowerBuilder Version	Appeon Developer Version
PowerBuilder9.0	5.0.0143.09
PowerBuilder10.0	

The PowerBuilder versions existing on this computer will be listed.

Step 2 – Select the Appeon Developer version to be registered in the PowerBuilder version.

If you want to unregister an Appeon Developer version from a PowerBuilder version, clear the selection in the Appeon Developer Version column.

Figure 15-17: Register toolbar to PowerBuilder



Appeon Developer Toolbar Registry

Select Appeon Developer Version

Register Appeon Developer toolbar in different PowerBuilder versions.

PowerBuilder Version	Appeon Developer Version
PowerBuilder9.0	5.0.0143.09
PowerBuilder10.0	

Step 3 – Click OK to register (or unregister) the selected Appeon Developer version with the PowerBuilder version.

16 Technical Support

Technical support information is available at http://www.sybase.com/support/. Please click on the *Contact Support* link for help.

When logging a case, be prepared to provide the following files to Sybase Technical Support in order to aid in debugging:

Table 16-1: Provide the following files for technical support

File Type	File Name
Appeon Server configuration file	All files in the %appeonserver%\repository\<instancename>\config folder. %appeonserver% indicates the Appeon Server installation path.
Appeon Server log file	All files in the %appeonserver%\repository\<instancename>\log folder.
Appeon Server Repository DB	For EAServer: %appeonserver%\db\AppeonServer.db For WebLogic\WebSphere: %appeonserver%\db\appeondb.script
Appeon license file	%appeonserver%\license.appeon
Web Server configuration file	httpd.conf
Web Page file	*Web Server path\application name\window name*.html *Web Server path\application name\window name*.js *Web Server path\application name\all DataWindows on window*.xml Note: If the window named includes inheritance, then also supply all ancestors.html and ancestors.js files.
PowerBuilder exported source code file	*PowerBuilder application path\exported window name*.srw *PowerBuilder application path\exported DataWindow names*.srd Note: Exported windows should include exported ancestor windows and exported ancestor user objects.

Index