

What's New in Appeon for PowerBuilder

Appeon[®] 2.8 for PowerBuilder[®]
FOR WINDOWS

DOCUMENT ID: DC20033-01-0280-01

LAST REVISED: July 16, 2004

Copyright © 2000-2004 by Appeon Corporation. All rights reserved.

This publication pertains to Appeon software and to any subsequent release until otherwise indicated in new editions or technical notes. Information in this document is subject to change without notice. The software described herein is furnished under a license agreement, and it may be used or copied only in accordance with the terms of that agreement.

No part of this publication may be reproduced, transmitted, or translated in any form or by any means, electronic, mechanical, manual, optical, or otherwise, without the prior written permission of Appeon Corporation.

Appeon, the Appeon logo, Appeon Developer, Appeon Enterprise Manager, AEM, Appeon Server and Appeon Server Web Component are trademarks or registered trademarks of Appeon Corporation.

Sybase, Adaptive Server Anywhere, Adaptive Server Enterprise, iAnywhere and PowerBuilder, are trademarks or registered trademarks of Sybase, Inc.

Java, JDBC and JDK are trademarks or registered trademarks of Sun, Inc.

All other company and product names used herein may be trademarks or registered trademarks of their respective companies.

Use, duplication, or disclosure by the government is subject to the restrictions set forth in subparagraph (c)(1)(ii) of DFARS 52.227-7013 for the DOD and as set forth in FAR 52.227-19(a)-(d) for civilian agencies.

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

Contents

1 About This Book	1
1.1 Audience	1
1.2 How to use this book	1
1.3 Related documents	1
1.4 If you need help	2
2 New PowerBuilder Features Supported.....	4
2.1 Command line argument	4
2.2 Dynamic transaction object to connection cache mapping.....	4
2.3 CreateOnDemand property of the Tab control	4
2.4 Multi-column sorting with space as the delimiter	4
2.5 Customize Message global object.....	4
2.6 Call OLE objects, DLLs and run executable programs on the Client.....	5
2.6.1 Client-side DLLs.....	5
2.6.2 OLEObject object.....	5
2.6.3 Run executable programs	6
2.7 Window icon	6
2.8 CheckBoxes property	6
2.9 LongParm property.....	7
2.10 ShareData and ShareDataOff	7
3 Performance Enhancements.....	8
3.1 Runtime benchmarks	8
3.1.1 Changes for Appeon 2.8	8
3.1.2 Expected performance for non-PFC applications.....	8
3.1.3 Expected performance for PFC applications.....	10
3.1.4 Performance impact of the Internet.....	12
3.2 Web deployment benchmarks	13
3.2.1 Expected performance for full deployment.....	13
3.2.2 E	14
3.3 DataWindow improvements	14
3.3.1 Drawing DataWindows.....	14
3.3.2 Improved performance when creating DropDownListBoxes, DropDownDataWindows and DataWindows	14
3.3.3 Synchronization window removed.....	14
3.4 Efficient object creation	15
3.5 Efficient Web file downloading	15
4 New Appeon Developer Features	16
4.1 Overview	16
4.2 Web Application Packaging Wizard.....	16

4.3 Runtime Report	16
4.4 Information Manager	16
4.4.1 Reports and Logs generated during the Web migration process	16
4.5 Web performance runtime tracing report settings.....	17
4.6 Application profiles	18
4.6.1 Database profiles settings.....	18
4.6.2 Performance	20
4.6.3 CmdArg settings	21
4.6.4 Misc	22
4.7 Customize UFA Report	23
4.8 Select and analyze an application	24
4.9 Select and run an application	25
5 New AEM Features.....	27
5.1 Application Properties	27
5.1.1 Transaction Objects.....	27
5.1.2 Display	27
5.1.3 Charset	29
5.2 Transaction Timeout.....	29
5.3 Deployment Sessions.....	30
5.4 Cache.....	30
6 Third-party Product Support.....	31
6.1 Apache Web Server	31
6.2 Netscape Web Server	31
6.3 PowerBuilder 9.0	31
6.4 ASA 9.0	31
6.5 PDFPrinter	31
6.6 Oracle Stored Procedures.....	31
7 Other New Features	32
7.1 Internet Explorer title bar	32
8 Product UI Changes.....	33
8.1 Installation	33
8.1.1 Installation of ASA and EAServer	33
8.1.2 AppeonDB connection cache.....	33
8.2 Appeon Developer.....	33
8.2.1 Appeon Developer Configuration UI changes.....	34
8.2.2 Specify AEM settings for each configured Appeon Server.....	36
8.2.3 Appeon Deployment Wizard	37
8.2.4 Web Report Configuration tool.....	38
8.3 AEM	39
8.3.1 Overview.....	39
8.3.2 Server Properties	39

9 Index.....42

1 About This Book

1.1 Audience

This book introduces the main new features added in Appeon 2.8 for PowerBuilder.

1.2 How to use this book

There are eight chapters in this book.

Chapter 1: About This Book

A general description of this book.

Chapter 2: New PowerBuilder Features Supported

Describe the newly supported PowerBuilder features in Appeon 2.8.

Chapter 3: Performance Enhancements

Describe the performance enhancements for Appeon 2.8.

Chapter 4: New Appeon Developer Features

Describe the new features of Appeon Developer.

Chapter 5: New AEM Features

Describe the new features of AEM.

Chapter 6: Third-party Product Support

Appeon 2.8 supports more third-party product.

Chapter 7: Other New Features

Describe the other new features of Appeon 2.8.

Chapter 8: Product UI Changes

Describe the product UI changes in Appeon 2.8.

1.3 Related documents

Appeon provides the following user documents to assist you in understanding Appeon products and capabilities:

- *Appeon Installation Guide:*
Instructions on how to install *Appeon for PowerBuilder* successfully.
- *Appeon Demo Applications Tutorial:*

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

- *Appeon Enterprise Manager User Guide:*

Introduces the Appeon Enterprise Manager, a Web application to maintain Appeon Web applications and Appeon Server over the Internet, the intranet or over an extranet.

- *Appeon Migration Guide:*

A process-oriented guide that illustrates the complete diagram of the Appeon Web migration procedure, and includes various topics related to steps in the procedure.

- *Appeon Migration Tutorial:*

A tutorial that walks the user through the entire process of deploying a small PowerBuilder application to the Web.

- *Appeon Help:*

Detailed list of what PowerBuilder PFC and non-PFC features are supported and can be converted to the Web with Appeon, and what features are not supported.

Appeon Help is an HTML version of the *Appeon Supported Features Guide* and *Using the PowerBuilder Foundation Class Library with Appeon*.

- *Appeon Supported Features Guide:*

Detailed list of what PowerBuilder non-PFC features are supported and can be converted to the Web with Appeon, and what features are not supported.

- *Using the PowerBuilder Foundation Class Library with Appeon:*

Detailed list of what PowerBuilder PFC features are supported and can be converted to the Web with Appeon, and what features are not supported.

- *Appeon Developer User Guide:*

Instructions on using the Appeon Developer toolbar integrated into the PowerBuilder IDE to perform various tasks related to Web migration.

- *Appeon Performance Tuning Guide:*

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

- *Appeon Troubleshooting Guide:*

Information about troubleshooting issues regarding such things as product installation, Web deployment, AEM, Web application runtime, and so on.

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 the 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 New PowerBuilder Features Supported

2.1 Command line argument

Appeon now supports the command line argument in the Open event of PowerBuilder applications. You can now start Web applications with parameters. On the Web, these parameters are embedded in the URL of the Web application. For example, <http://192.0.0.217:8080/test/index.htm?ArgumentValue>".

2.2 Dynamic transaction object to connection cache mapping

You can now dynamically set transaction object to connection cache mapping in script or statically in AEM. Transaction object to connection cache mapping can now be performed dynamically via PowerScript.

Transaction object to connection cache mapping can be set up dynamically or changed by setting or changing the DBMS and DBParm property of the Transaction object in the application source code, (for example, you can now dynamically change the selected database during application runtime).

To set or change the connection cache dynamically, you can code the DBParm property of the Transaction object in this format:

```
SQLCA.DBParm="CacheName='ASEConnectionCache1'".
```

"ASEConnectionCache1" can be replaced by the name of the connection cache you want to use for the Transaction object.

2.3 CreateOnDemand property of the Tab control

Appeon 2.8 supports the CreateOnDemand property for the Tab control. Similar to PowerBuilder, if CreateOnDemand is set to TRUE, Constructor events for controls on the visible tab page are triggered when the Tab control is created. Constructor events for controls on the hidden tab pages will not be triggered until the user views the tab page. A window will open faster when only Constructor events for controls of the visible tab page are triggered.

2.4 Multi-column sorting with space as the delimiter

Appeon 2.8 now supports both comma-separated and space-separated sorting criteria. For example, the following script is now supported:

```
dw_data.setsort("fitemid D, fcheckcode D, fclasscode A fdesc A flevel A ")  
dw_data.sort()
```

2.5 Customize Message global object

Appeon 2.8 supports customizing the Message object by defining a class user object inherited from the built-in Message object. You can define a Message standard class type and specify it as the default Message global object in the PowerBuilder painter. When the application is run, it

will use the specified standard class user object as the default object instead of the built-in global object.

2.6 Call OLE objects, DLLs and run executable programs on the Client

2.6.1 Client-side DLLs

Appeon can now call DLLs and other external functions on the Client.

DLL support now includes the following:

- Declare and call global external functions
- Call user-defined DLLs
- Call standard Win32 API (such as kernel32.dll, user32.dll, gdi32.dll)
- Data types of external function arguments can be one of unsigned char, char, unsigned short, short, unsigned long, long, float, string, or Boolean.
- Data types of external function return values can be one of unsigned short, short, unsigned long, or long.

2.6.2 OLEObject object

Appeon 2.8 provides limited support to OLE: you can create an OLE object independent of an OLE control, connect to the server application, and set properties for that object. The OLE server application may be either DLLs or separate EXE files. It works the best if it is Microsoft Word or Microsoft Excel program.

It is limited support because OLE control is unsupported yet, and you can only set or get properties, but cannot call functions of the server function. For more supported and unsupported details, refer to Appeon Help | OLEObject object section.

The following is a typical case that applies OLEObject features supported by Appeon. It creates an OLEObject variable, connects to Microsoft Word 's OLE interface, opens a document, displays the user name and document name, inserts some text, saves the edited document, and shuts down the server application:

```
OLEObject o1
string s1
o1 = CREATE oleobject
o1.ConnectToNewObject("word.application")
o1.documents.open("c:\temp\temp.doc")

o1.Application.Visible = True
s1 = o1.UserName
MessageBox("MS Word User Name", s1)
s1 = o1.ActiveDocument.Name
```

```
MessageBox("MS Word Document Name", s1)

o1.Selection.TypeParagraph()
o1.Selection.tyepetext("Insert this text")
o1.Selection.TypeParagraph()

o1.ActiveDocument.Bookmarks[1].Select
o1.Selection.tyepetext("Hail!")

o1.ActiveDocument.Bookmarks.item("End").Select
o1.Selection.tyepetext("Farewell!")

o1.ActiveDocument.Save()
o1.quit()
RETURN
```

2.6.3 Run executable programs

Appeon 2.8 supports running external applications from the Appeon Web application. If an application runs an executable program, you must specify that the application needs to run an executable program by checking the "Call to OLE objects, DLLs, or run executable programs on the client" option in the Appeon Developer Configuration. When the Web application is run, an ActiveX control will be downloaded to the Client from the Web Server to work together with the external calls and run the executable programs.

It is supported to run an executable program with any of the following methods:

```
run("notepad")
run("notepad.exe")
run("C:\winnt\system32\notepad")
run("C:\winnt\system32\notepad.exe")
```

2.7 Window icon

Customized window icons are now supported, instead of always using the Appeon window icon. The customized window icon feature can now apply to Sheet and Child type Windows. You can also disable the icon.

2.8 CheckBoxes property

Appeon 2.8 now supports the CheckBoxes property of the ListView control.

2.9 LongParm property

The LongParm property of the Message object is now supported.

2.10 ShareData and ShareDataOff

The ShareData and ShareDataOff functions are now supported.

Support for this feature in Appeon 2.8 has a wide range of benefits, most notably the ease of converting an existing PowerBuilder application that uses ShareData to the Web with Appeon.

For more information on the ShareData and ShareDataOff functions, refer to Appeon Help.

3 Performance Enhancements

3.1 Runtime benchmarks

3.1.1 Changes for Appeon 2.8

Appeon's runtime performance has been significantly improved in 2.8 over Appeon 2.7.

Windows now open 3 to 5 times faster. A window that took 3 to 8 seconds to be displayed with Appeon 2.7 is now displayed in 1 second. A window that took 17 seconds to be displayed with Appeon 2.7 is now displayed in 4 seconds. Most windows can now be opened in under 3 seconds*.

* Performance is applicable to non-PFC application windows containing 5 DataWindows/DataStores or less and no Tab controls, TreeViews, or UserObjects; DropDownDataWindows count as 0.5 of a DataWindow's weight.

Entering data in a DataWindow is now comparable in performance to a Client Server PowerBuilder application.

3.1.2 Expected performance for non-PFC applications

Appeon has tested a number of applications varying in size and complexity in Appeon 2.7 and Appeon 2.8. What Appeon found is that, Windows containing **fewer than 5 DataWindows*** open in **fewer than 3 seconds** in Appeon 2.8, with some exceptions. In contrast, Windows containing more than 5 DataWindows/DataStores take longer than 3 seconds to open. Refer to Graph 2-1 and 2-2 to view the improvement in performance of Appeon 2.8 over Appeon 2.7.

*DataStores are non-visual DataWindows and should therefore be counted as a DataWindow. 2 DropDownDataWindows are equivalent to 1 DataWindow in terms of its negative impact on runtime performance; therefore, each DropDownDataWindow is equal to 0.5 of a DataWindow's weight.

The following were the test parameters:

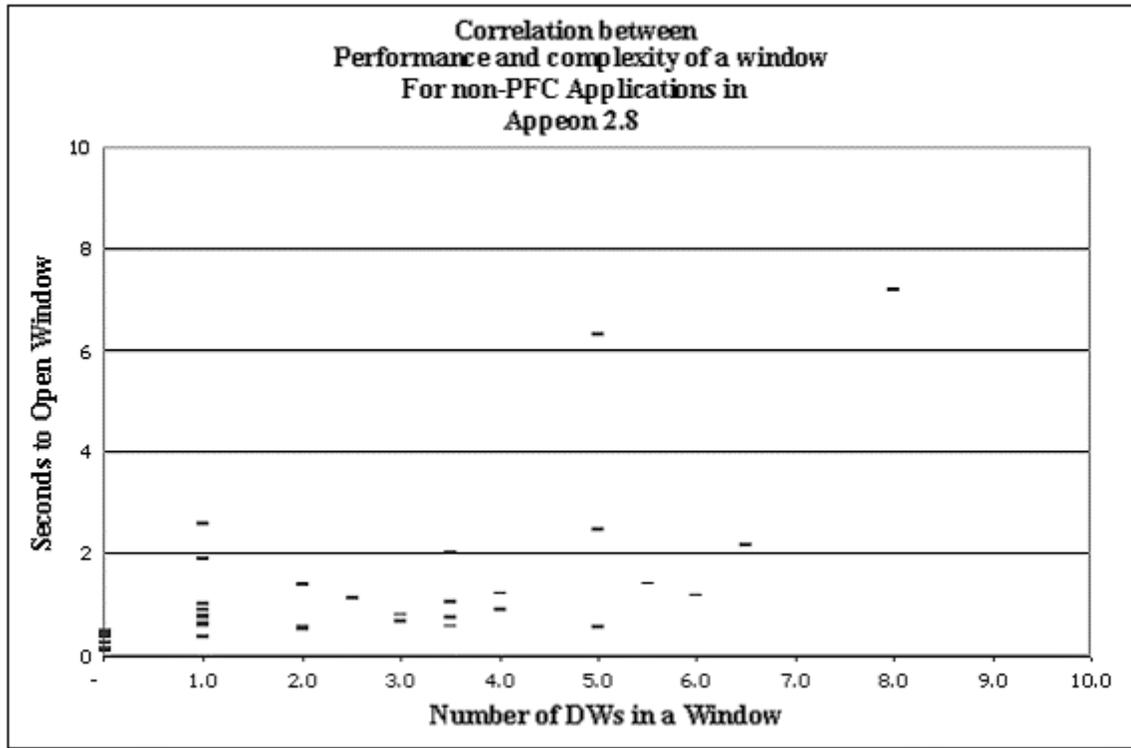
All applications tested were actual customer applications provided to Appeon for testing purposes. Appeon did not optimize the performance of these applications for the tests.

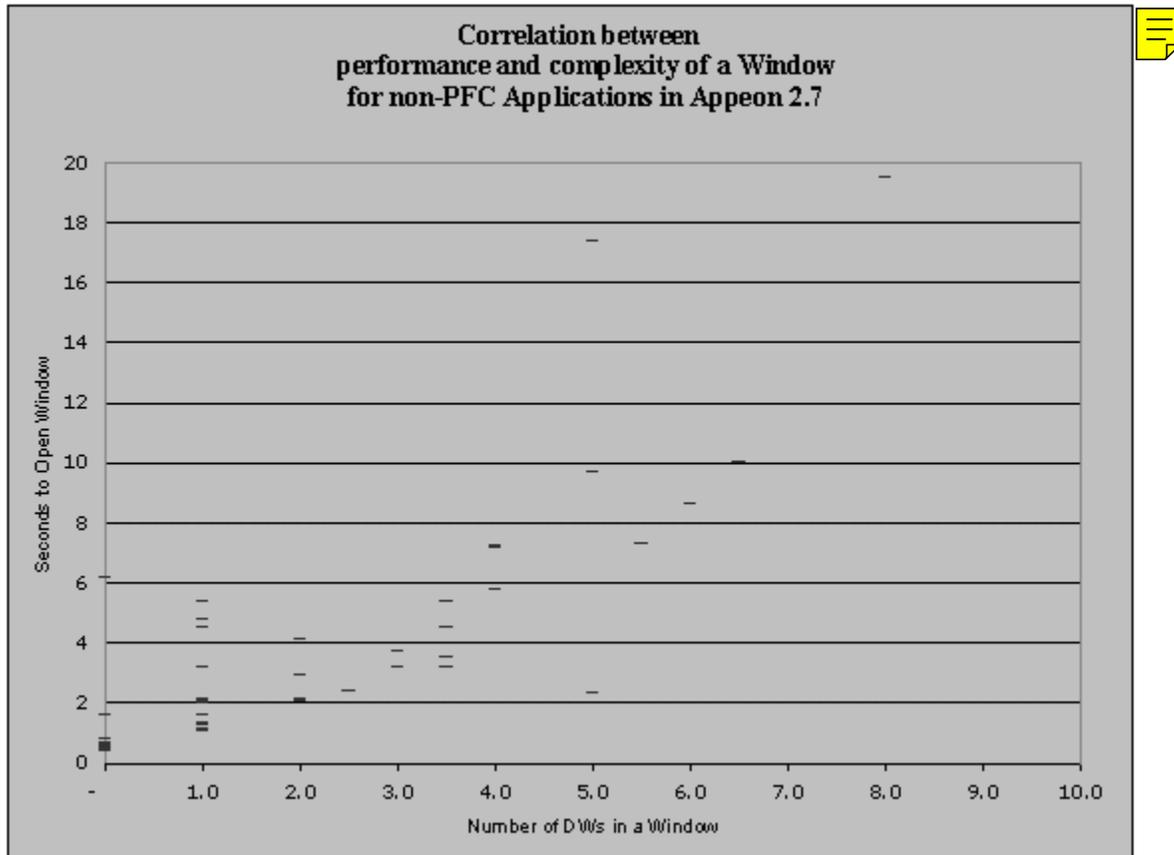
Performance was measured by the number of seconds it took to open a window and display data in the DataWindows.

The test environment was a single CPU Intel P4 1.8 GHz machine with 512 MB RAM configured for local deployment. The machine acted as the Database Server, the Application Server, Web Server, and the Client. (Performance will be slower on a computer with lower CPU performance)

Internet download times are not considered. These results reflect a LAN environment or an Internet environment where the Web files for a given window are cached at the Web browser. Web files are automatically cached at the Web browser after the initial Window opening.

Graph 2-1: Appeon 2.8 runtime performance benchmarks for non-PFC applications



Graph 2-2: Apeon 2.7 runtime performance benchmarks for non-PFC applications

In many situations, this rule holds true for non-PFC applications, and therefore it is the best way to get a sense of what portion of your application will not require much performance tuning.

3.1.3 Expected performance for PFC applications

We have found that the runtime performance of Apeon 2.7 and Apeon 2.8 for PFC applications differs significantly from that of non-PFC applications. In addition, the complex nature of PFC features tends to play a greater role in runtime performance. Therefore, no strong correlation has been observed for PFC applications between the speed at which a window opens and the number of DataWindows the window contains.

Since the performance of Apeon 2.8 for PFC applications is generally 3 to 5 times faster than Apeon 2.7, most of the PFC windows can be opened in a 6 second or less range. Refer to Graphs 2-3 and 2-4 to view the improvement in performance of Apeon 2.8 over Apeon 2.7.

The following were the test parameters:

All applications tested were actual customer applications provided to Apeon for testing. Apeon has not optimized the performance of these applications.

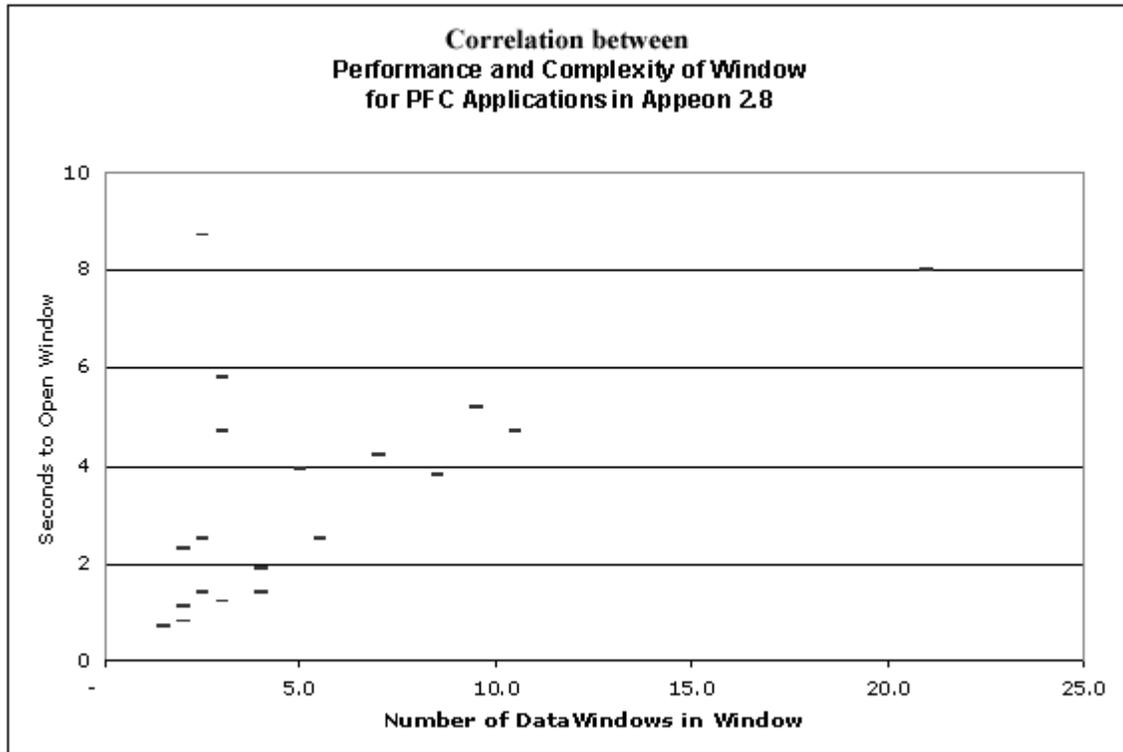
Performance was measured by the number of seconds it took to open a window and display data in the DataWindows.

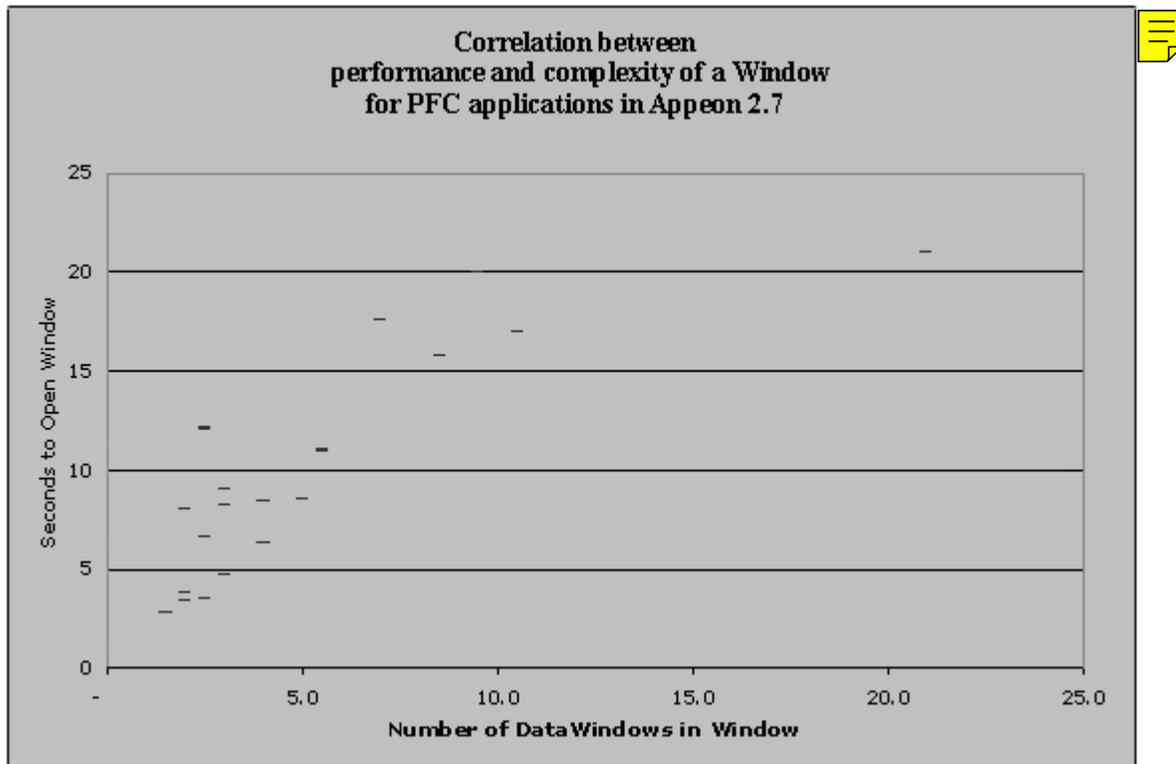
The test environment was a single CPU Intel P4 1.8 GHz machine with 512 MB RAM configured for local deployment. The machine acted as the Database Server, the Application

Server, Web Server, and Client. (Performance will be slower on a computer with a slower CPU speed)

Internet download times are not considered. These results reflect a LAN environment or an Internet environment where the Web files for a given window are cached at the Web browser. Web files are automatically cached at the Web browser after the initial Window opening.

Graph 2-3: Apeon 2.8 runtime Performance Benchmarks for PFC Applications



Graph 2-4: Apeon 2.7 runtime performance benchmarks for PFC applications

Based on Apeon's experience working with customer's PFC applications, it is possible to have many PFC Windows open in a 5 second or less range. However, if the application is deployed without significant performance optimization, it is very likely you will get results similar to those depicted in the above graph. To achieve better performance for PFC applications, it will be necessary for you to spend time and understand what is slowing down the window. Then you can simplify the application by reworking or removing those performance-hungry features.

3.1.4 Performance impact of the Internet

When you move from a LAN environment to the Internet, the time it takes to open a Window will increase. The opening of the Window will be slowed down, due to the time it takes to download the files necessary for the Window to run and display data. There are two types of files downloaded. The first types are HTML, JavaScript and Web files that contain the UI and UI logic of the application Window. The second type is a XML file that contains the result set for a DataWindow retrieve. The time to download these files is affected by two factors: 1) the throughput of the network connection and 2) the size of the files to be downloaded.

The Web files do not impact performance because of their small size and the enhanced ability of the browser to "cache" them. The Web files for a given PowerBuilder window are typically between 25-75KB, which in some cases is slightly larger than the average 50KB Web page. Because these Web files are static in nature, once a given application window has been opened, the Web files will be cached on the Client. Under most circumstances, these Web files are not re-downloaded when the window is reopened. The only exceptions are if 1) the temporary Internet files folder has been emptied or 2) the application has been updated and redeployed to the server.

Only the XML files containing the DataWindow result sets impact performance. A result set of 50 records would result in a 12KB XML file. Every 5 records would add another 1.2 KB to the XML file size. As an example, a 500-record result set would correspond to a 120KB XML file. Because these XML files contain time-sensitive data, they cannot be cached without risking the data becoming stale. Therefore, they must be downloaded from the server each time a DataWindow retrieve is initiated.

Appeon has built-in 10X data compression * for DataWindow result sets to essentially eliminate the time downloading these XML files over the Internet. The same 500-record result set that would normally correspond to a 120KB XML file would only result in the download of a 12KB file from the server. This compression feature makes even the largest of result sets quick to transfer over the Internet.

*Requires Internet Explorer 6.0 and above Web browsers; 10X data compression will be automatically disabled for versions of Internet Explorer that do not meet the requirement.

3.2 Web deployment benchmarks

3.2.1 Expected performance for full deployment

Appeon 2.8 deployment times are approximately 50% lower in comparison to Appeon 2.7, and very close to PowerBuilder deployment to EXE and DLLs.

The table below benchmarks the performance of several applications. The applications vary in size and complexity. The deployment times are recorded for deploying the application to Client and Server with PowerBuilder and to the Web.

The tests were performed on a single CPU Intel P4 1.8 GHz with 512 MB RAM and 60 GB IDE hard drive.

- The simple application had no inheritance.
- The complex application made heavy use of inheritance (up to five levels of inheritance in some places).

The time recorded for Appeon 2.8 is the time used for unsupported feature analysis and Web file generation (the first two tasks in the deployment process). The performance impact of the file deployment is negligible for local deployment mode.

The time recorded for PowerBuilder is the time to complete the entire build process and produce an EXE file with the PBDs and an EXE file with the DLLs.

Table 2-1: Appeon 2.8 Deployment Performance Benchmarks

	PB EXE w/ PBDs	PB EXE w/ DLLs	Appeon 2.8
10 MB simple non-PFC application	3 minutes 0 seconds	15 minutes 50 seconds	5 minutes 0 seconds
12 MB complex non-PFC application	2 minutes 35 seconds	15 minutes 12 seconds	11 minutes 50 seconds
25 MB PFC application	8 minutes 5 seconds	40 minutes 6 seconds	13 minutes 20 seconds

3.2.2 Expected performance for incremental deployment

The incremental deployment function is used when maintaining or upgrading an already-deployed application. This feature only re-deploys to the Web the changes that have been made in the application. Appeon can locate the exact lines of PowerScript code in an object that has been modified and re-deploy only those lines of code. Appeon 2.8 takes just seconds to update a single modified object.

3.3 DataWindow improvements

Appeon have boosted the performance of the following DataWindow features:

- Modify
- Describe
- Dot notation
- DataWindow object assignment
- DataWindow retrieval
- DataWindow UI display
- DropDownDataWindow and DropDownListBox performance

3.3.1 Drawing DataWindows

In Appeon 2.8, a single DataWindow display refresh will be made when an operation has been completed and the new data is ready to be displayed. For example, in a clicked() event where several UI related statements are coded, the DataWindow will not be refreshed until the completion of the event (all code in the event handler has finished executing). DataWindows no longer flash numerous times to display the intermediate data to reflect the execution status, which is usually ignored by the user. The DataWindow will only refresh and the retrieved data will only be displayed when the execution is complete. Now the DataWindow refresh time has been reduced to only once.

3.3.2 Improved performance when creating DropDownListBoxes, DropDownDataWindows and DataWindows

Appeon 2.8 now only creates a DropDownListBox or a DropDownDataWindow when the DropDownListBox or DropDownDataWindow control receives focus (for example, when the user selects the control using the mouse or keyboard).

This has improved DataWindow performance, especially for DataWindows with many of these Edit styles and large result sets.

3.3.3 Synchronization window removed

Appeon no longer displays the synchronization window (flash window) when DataWindows are retrieved. This has shortened the time needed to display a DataWindow and users are not distracted by the flash window.

3.4 Efficient object creation

Appeon has lowered memory usage and reduced the time required to create or instantiate objects, including DataWindows, Menus and Applications. This has improved the runtime performance of Web applications, with a very noticeable improvement when opening application windows.

3.5 Efficient Web file downloading

Appeon 2.8 now uses multiple threads to download Web files; therefore, files will start downloading simultaneously rather than sequentially. When files are requested, they no longer need to wait before downloading commences and the total download time is reduced.

4 New Appeon Developer Features

4.1 Overview

This chapter covers new features in Appeon Developer, with the three most significant changes being the addition of three new buttons and Application Profiles.

Table 3-1: New Appeon Developer toolbar



Table 3-2: New buttons in Appeon 2.8

Button	Name	Function
	Package Wizard	Package your files ready for installation
	Runtime Report	Generate reports during runtime
	Info Manager	Manage your log files

4.2 Web Application Packaging Wizard

The application packaging tool 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 instead of doing several individual Web deployments; this saves time. The PowerBuilder developers do not need to provide the source PowerBuilder application to the user; this protects the author’s intellectual property rights.

4.3 Runtime Report

The Web Performance Runtime Tracing Report provides you with detailed runtime information, to help you identify events and functions that slow down your application. The information supplied by the report details the time at which the event or function is executed and the duration it takes for the event or function to complete its task. The Web Performance Runtime Tracing Report is dynamically generated during application runtime.

For more information on the Web Performance Runtime Tracing Report, refer to chapter 11: *Web Performance Runtime Tracing Report* in the *Appeon Developer User Guide*.

4.4 Information Manager

The Information Manager is a repository to help you organize and easily find your reports and log files generated by Appeon Developer.

Refer to chapter 12: *Information Manager* in the *Appeon Developer User Guide* for more information.

4.4.1 Reports and Logs generated during the Web migration process

Table 3-3 provides a brief description of all the reports generated during the Web migration process.

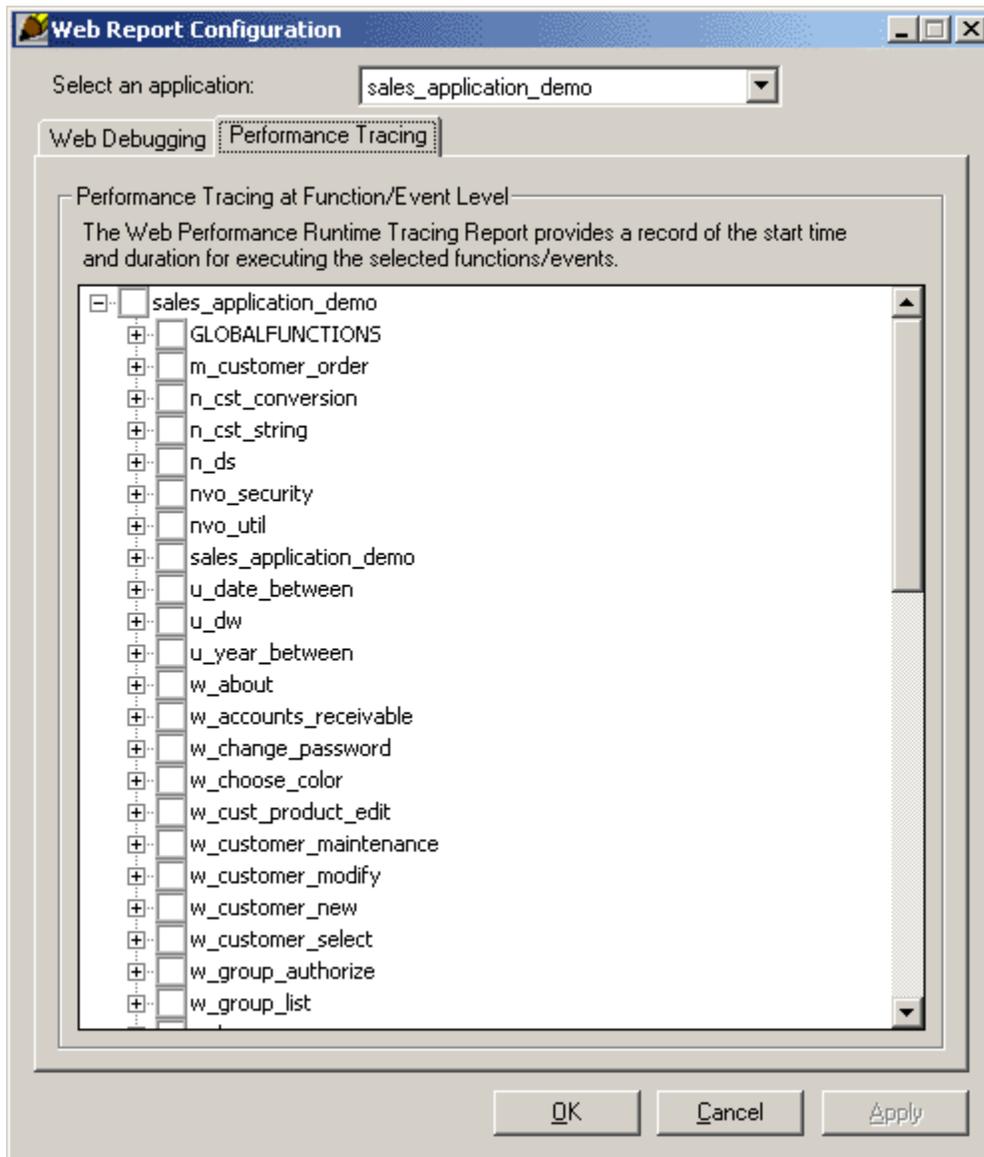
Table 4-3: Report Description

Report Types	Description
Analysis	<p>Analysis Report is generated during features analysis or during the application deployment process, and records the unsupported features in an application.</p> <p>Incremental Report is generated if the application has been modified and deployed incrementally, and displays recently discovered unsupported features in your application.</p>
Debugging	<p>Web Debug Report is generated when the Web application is run, and records the executed objects, controls, and events/functions of the Web application.</p>
Performance	<p>Web Performance Report is generated dynamically during the runtime of the Web application, and records the time details of executing the routines.</p> <p>Heavy Window Report is generated during a feature analysis or during application deployment, and records the complex windows in an application.</p>

4.5 Web performance runtime tracing report settings

The Web Performance Runtime Tracing Report is used for improving the Web application's runtime performance. During the runtime of an application, the following execution information will be recorded: the executed functions/events, the object and root object that contains the functions/events, the time at which functions/events are called, and the duration of each call.

After the application is deployed and before the first time the application is run, configure the Web performance tracing report settings to select the specific execution information you need.

Figure 3-1: Performance Tracing

4.6 Application profiles

4.6.1 Database profiles settings

Database profiles are now required for applications being deployed by Apeon 2.8, these enable Apeon to significantly enhance your Web applications runtime performance.

The DB profile provides information about the database, database type, user ID and password that is used to connect to a database. The deployment process uses this information to connect to the specific database and generates a corresponding set of database syntax. Create a DB profile for every database type used by the application.

This has improved Web application performance.

Refer to section 4.2.1.b: *Database profile settings* in the *Appeon Developer User Guide* for more information.

Figure 4-2: DB Profiles

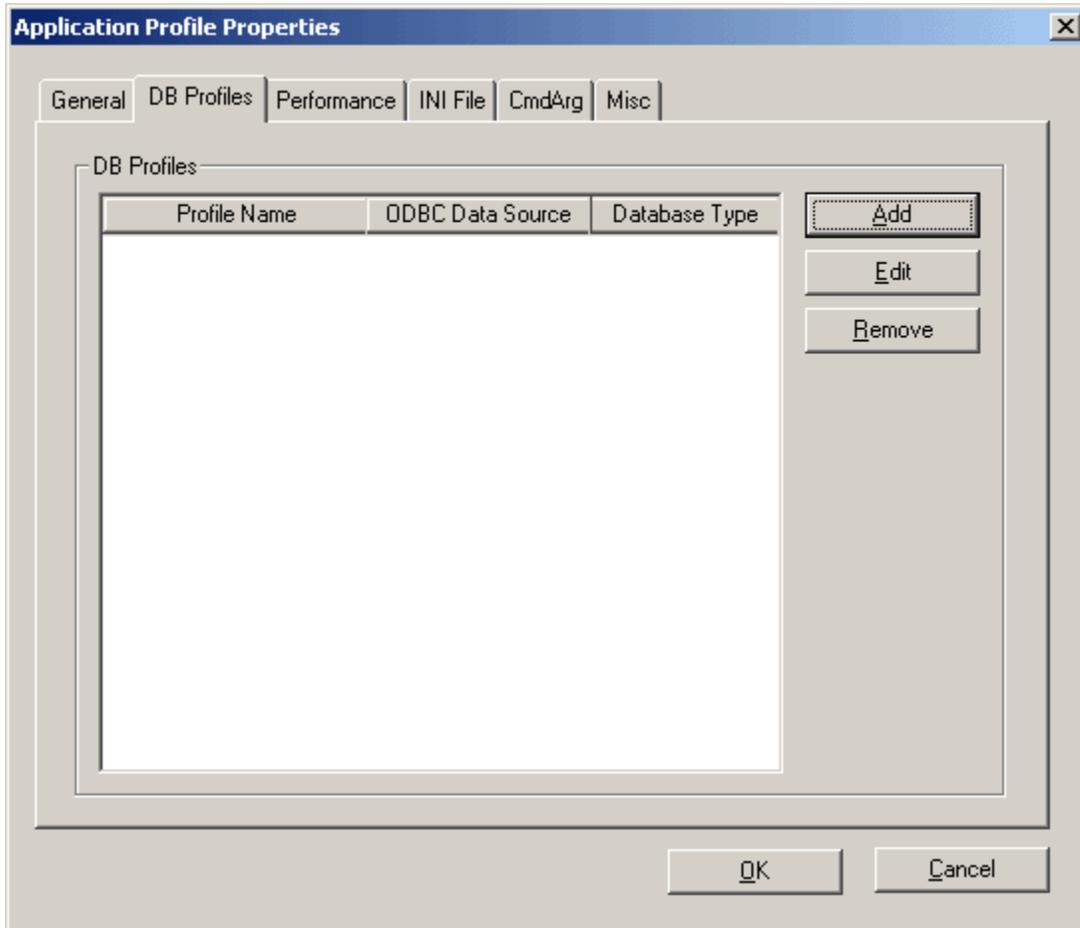
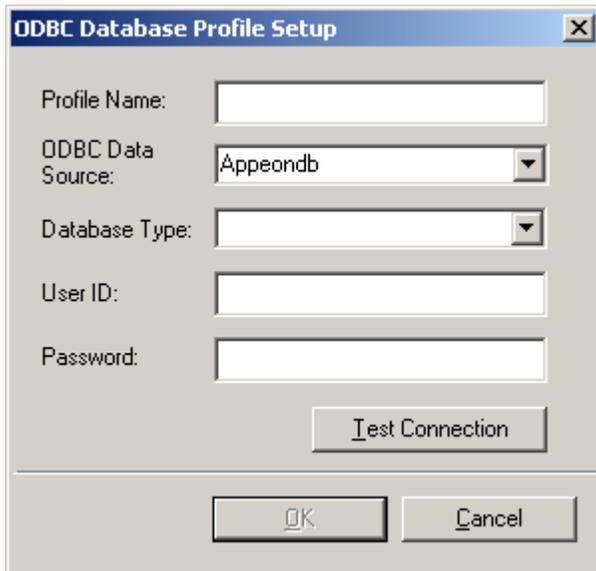


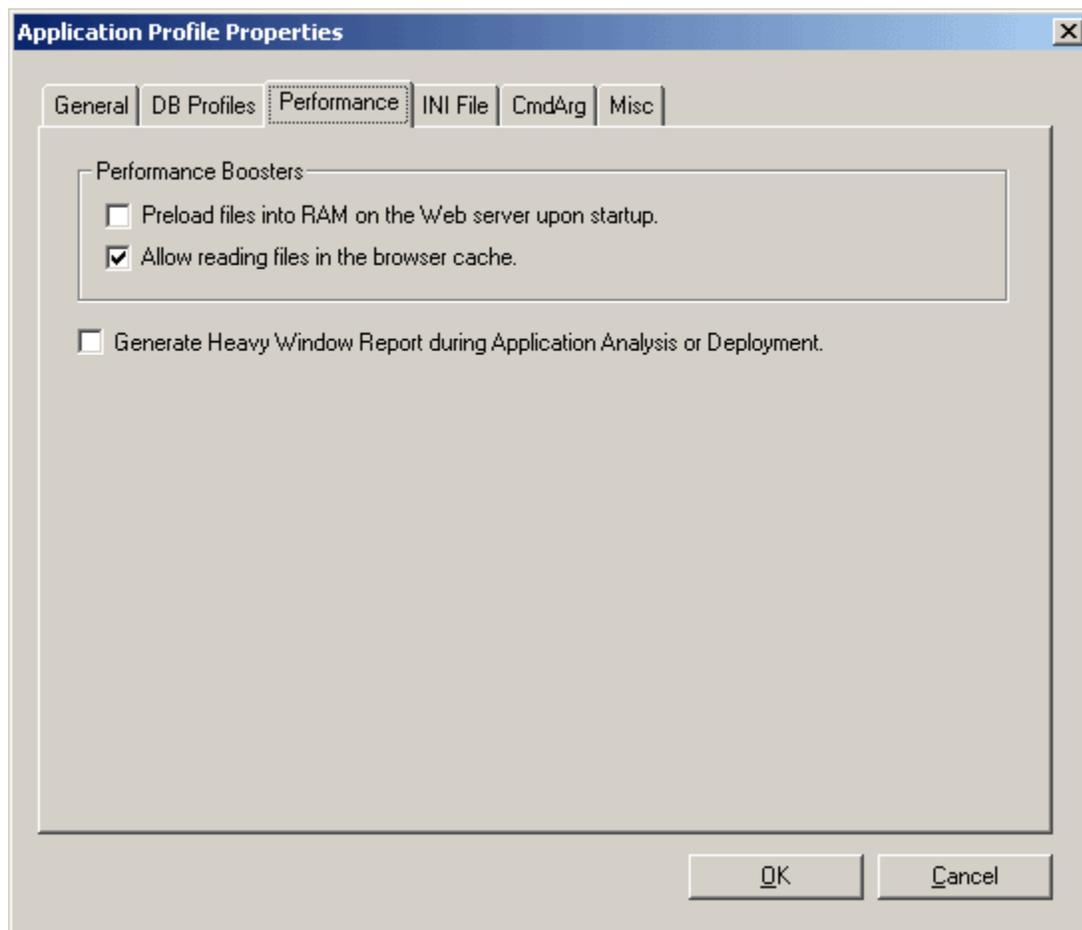
Figure 3-3: ODBC Database Profile Setup



Profile Name	Assign a name to the DB Profile.
ODBC Data Source	Select a proper data source from the DropDownListBox. All the data sources configured in Windows ODBC Administrator will be listed.
Database Type	Select the correct database type from the DropDownListBox. The database type must exactly match the database used in the application. Note: if an incorrect database type is selected, errors will occur when the Web application is running.
User ID	Type a recognized user name to login to the database, if no username has been set, use either the database default or leave this field blank.
Password	Type in the password used to login to the database. If no username has been set, either use the database default or leave this field blank.

4.6.2 Performance

Figure 4-4: Performance settings



4.6.2.a Preload files into RAM on the Web server upon startup

Preload files into RAM on the Web server upon startup is a new feature that improves Appeon Server's performance, all Web files are loaded into RAM on the Web Server when it starts. When a Client visits the application, the Web files in RAM on the Web Server will be used, instead of retrieving Web files from the Web Server's hard disk.

4.6.2.b Allow reading files in the browser cache

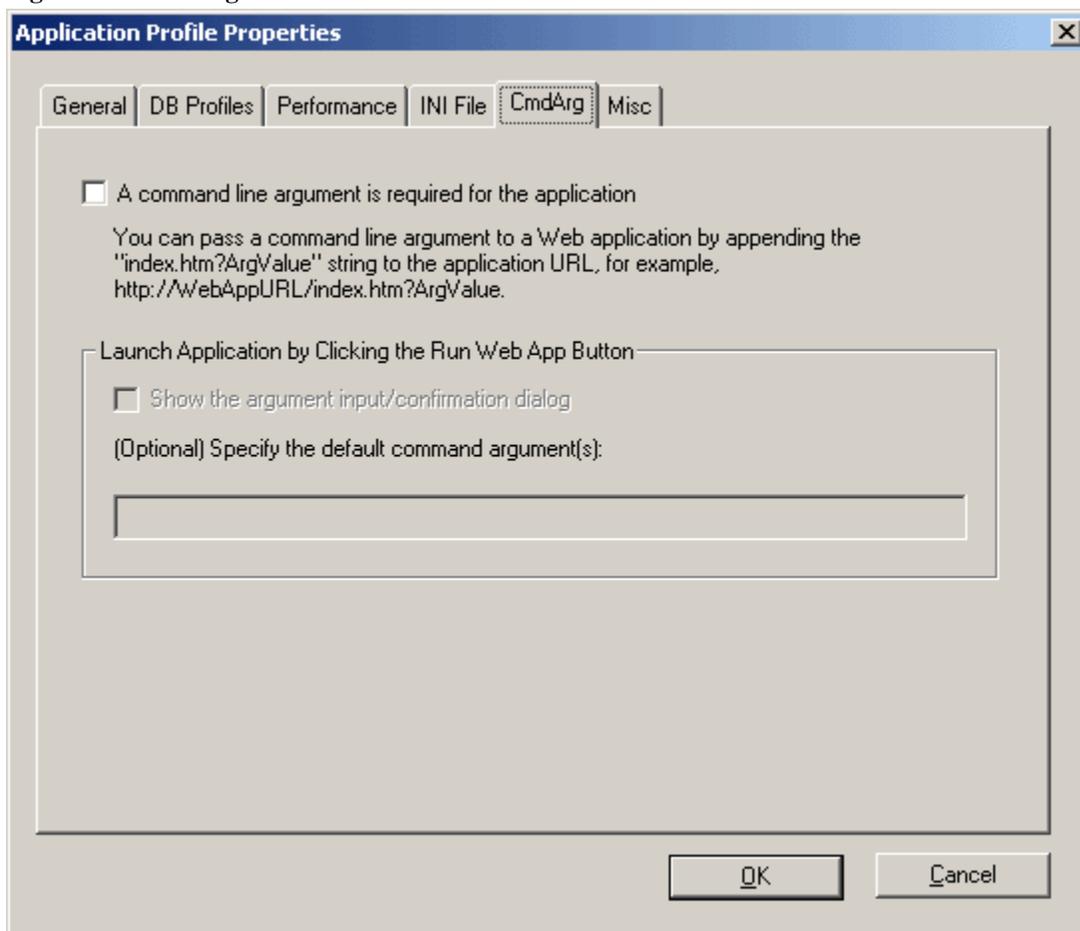
The *Allow reading files in the browser cache* option enables the Client to read Web files directly from Internet Explorer's temporary file cache. After the files have downloaded, the files can be cached in the browser. For subsequent requests to download the files, the browser can read them from the cache directly, unless a newer version of the file is detected on the Web Server. This can greatly speed up subsequent visits by Clients to a Web application.

4.6.3 CmdArg settings

If a PowerBuilder application uses command line arguments, you are required to specify the arguments on the CmdArg tab page, these will then be passed to the corresponding Web application, as shown in figure 3-5.

For more details, refer to section 4.2.1.e: *CmdArg Settings* in the *Appeon Developer User Guide*.

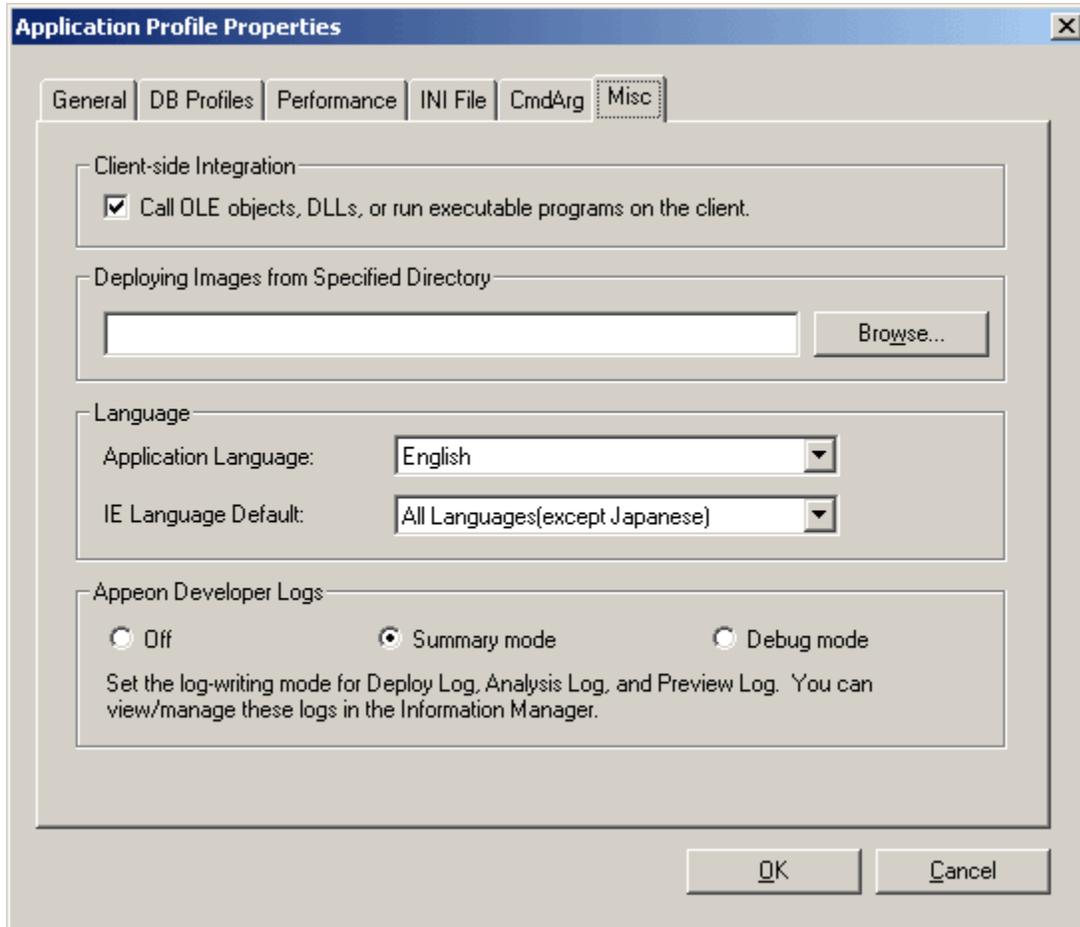
Figure 4-5: CmdArg



4.6.4 Misc

Misc tab page offers settings for client-side integration, image files, language and log-writing mode.

Figure 4-6: Configure Performance and Language options for Web applications



- Call OLE objects, DLLs or run executable programs on the Client.
It is selected by default. Uncheck this option, if your application does not use any external calls or run executable programs. However, it is safe to keep this option selected as default.
If your PowerBuilder application calls OLE objects or DLL files, or runs executable programs, you must check this option so that the calls can also work on the Web application. When the Web application is run, an ActiveX control will be downloaded to the Client from the Web Server.
- Deploying Images from Specified Directory
Specify which folder contains the image files that will be used in the application. The path to the application PBL files will be displayed as the default path of the image files. Click Browse to select a different folder.

The image files can be in any format supported by PowerBuilder. All the image files in the specified folder will be deployed to the Web Server.

- Application Language

The Application Language setting determines the language of the deployed application runtime messages or prompts. There are five languages to choose from for the applications error messages or prompts. English is selected by default.

- IE Language Default

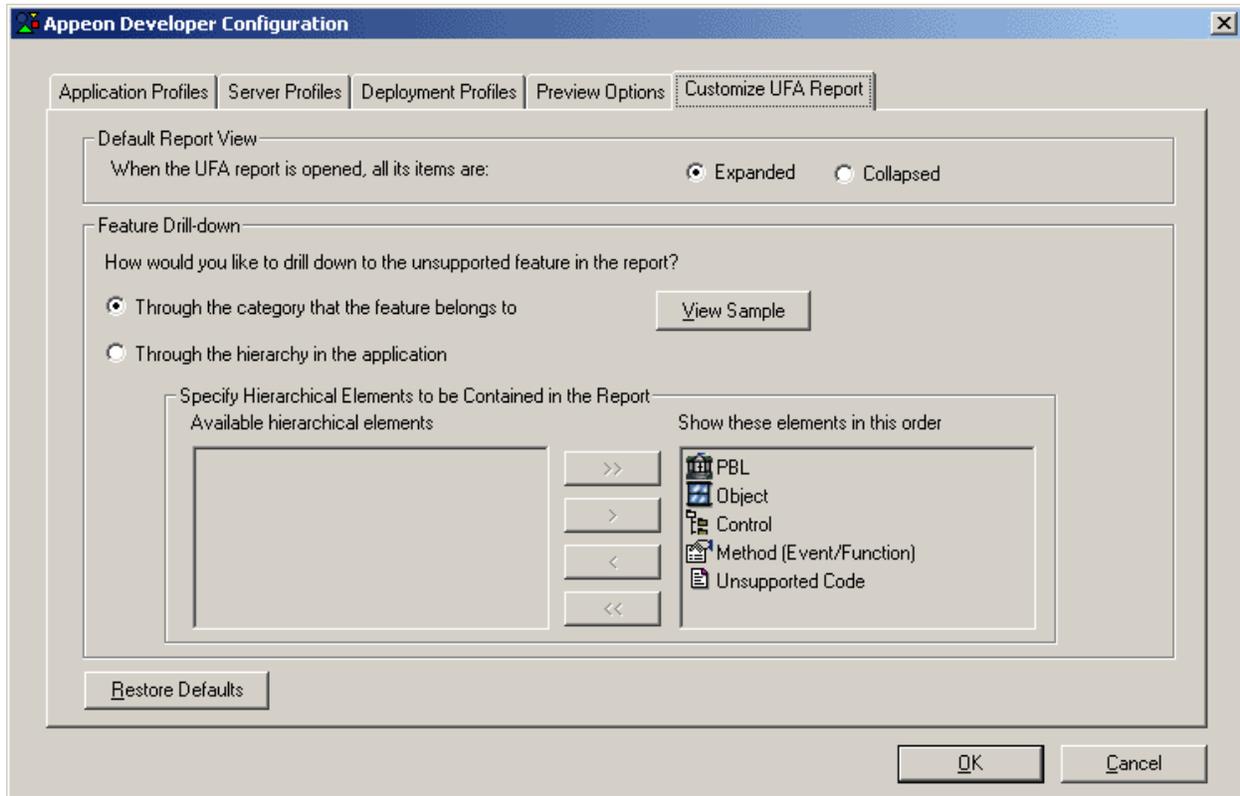
The *IE Language Default* option determines the parsing rules of PowerBuilder units that will be used to deploy the application. Japanese PBUs will be adopted if Japanese is selected, otherwise English PBUs will be adopted.

- Appeon Developer Logs

The logging options enable the user to select the level of information contained in the logs, according to their needs.

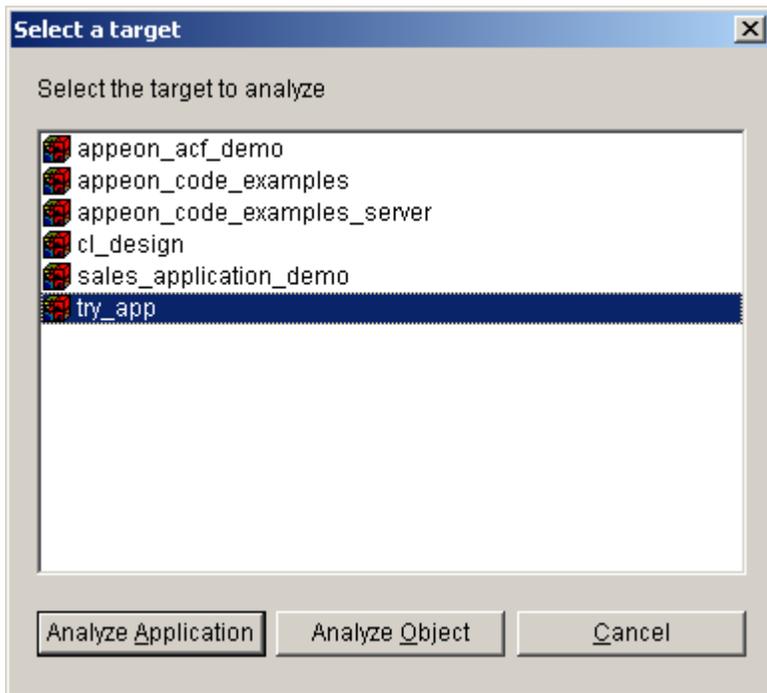
4.7 Customize UFA Report

The UFA (Unsupported Feature Analysis) report is a very important report that is frequently used to detect unsupported features in an application. The UFA report is customizable. Set the display format of the UFA report to best suit the chosen purpose of the report. You can customize the UFA report by configuring the two settings described below. When you run Feature Analysis or application deployment at a later stage, the customized settings will take effect.

Figure 3-7: Customize UFA Report

4.8 Select and analyze an application

The user can select which application to analyze, as shown in Figure 3-7.

Figure 4-8: Select and analyze an application

All applications configured in the *Application Profile* tab page are listed in alphabetical order.

The current application configured in *Application Profile* is selected and highlighted by default.

The *Analyze Application* button is set as the default analysis type.

- If the *Analyze Application* button is clicked, the selected application will be analyzed the same way it is in Appeon 2.7.
- If the *Analyze Object* button is clicked, another window is displayed to specify which objects in the selected application to analyze. The only difference with this window in Appeon 2.8 is that the Analysis Type GroupBox is removed from the *Appeon Unsupported Features Analysis* window.

4.9 Select and run an application

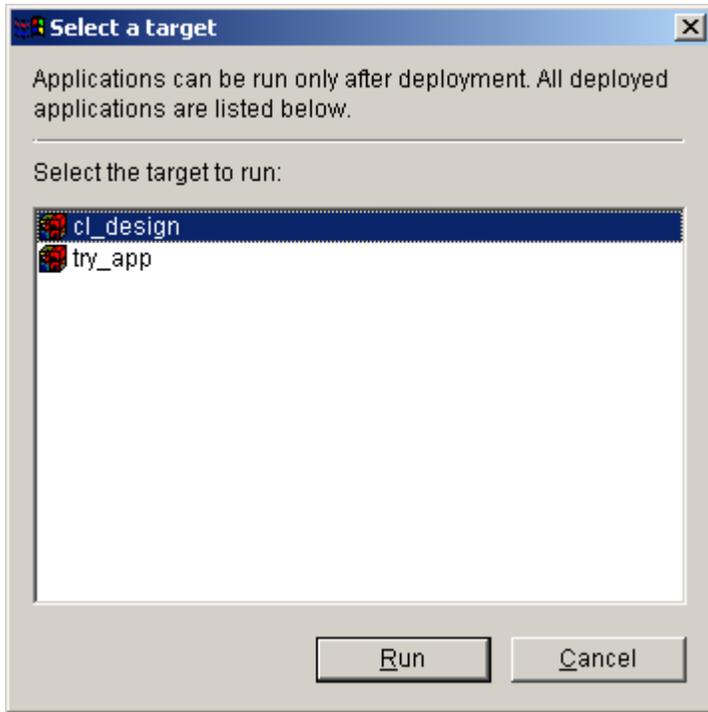
The user can now select which application to run, as shown in Figure 3-7.

Applications included in *Application Profiles* and already deployed with Appeon are listed in alphabetical order.

If the current application configured in *Application Profile* has been deployed, it is selected and highlighted by default. Otherwise, none are selected by default.

The *Run* button is selected by default.

Figure 4-9: select and run Web application



5 New AEM Features

5.1 Application Properties

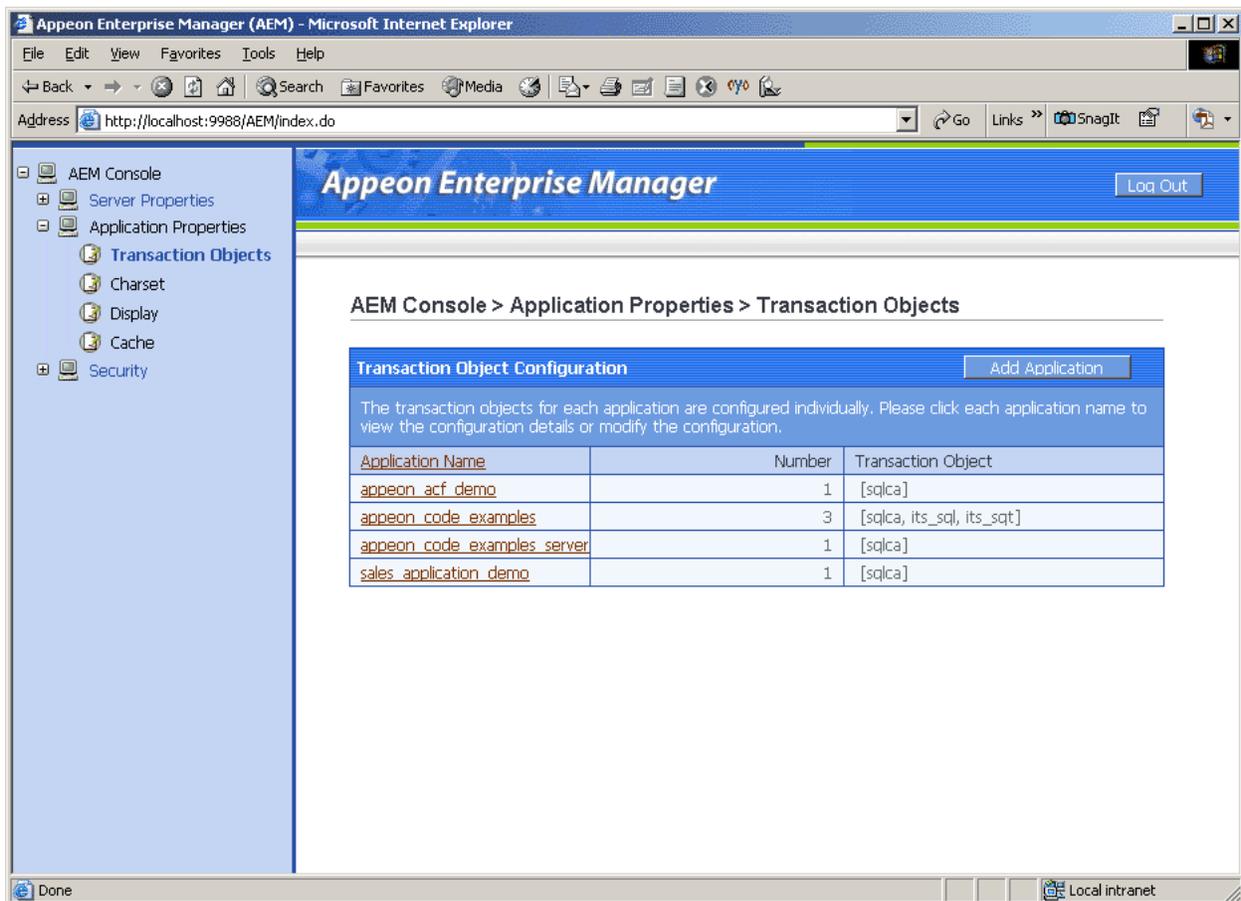
This section describes UI changes to three tools in *Application Properties: Transaction Objects, Display, and Charset*

5.1.1 Transaction Objects

When the user clicks the *Application Name* column header, the table can now be arranged into ascending or descending order, as shown in Figure 5-1.

After you add a new application using the *Add Application* button, the new application will be added to the start of the table.

Figure 5-1: Transaction Objects



5.1.2 Display

To access the *Apeon DataWindow Menu* (Figure 5-3) and the *Configure Date, Time and Currency Display Format* section (Figure 5-4), the user can now select which application to adjust by clicking the applications name, as shown in Figure 5-2.

When the user clicks the *Application Name* column header, the table can now be arranged into ascending or descending order, as shown in Figure 5-2.

Figure 5-2: Display

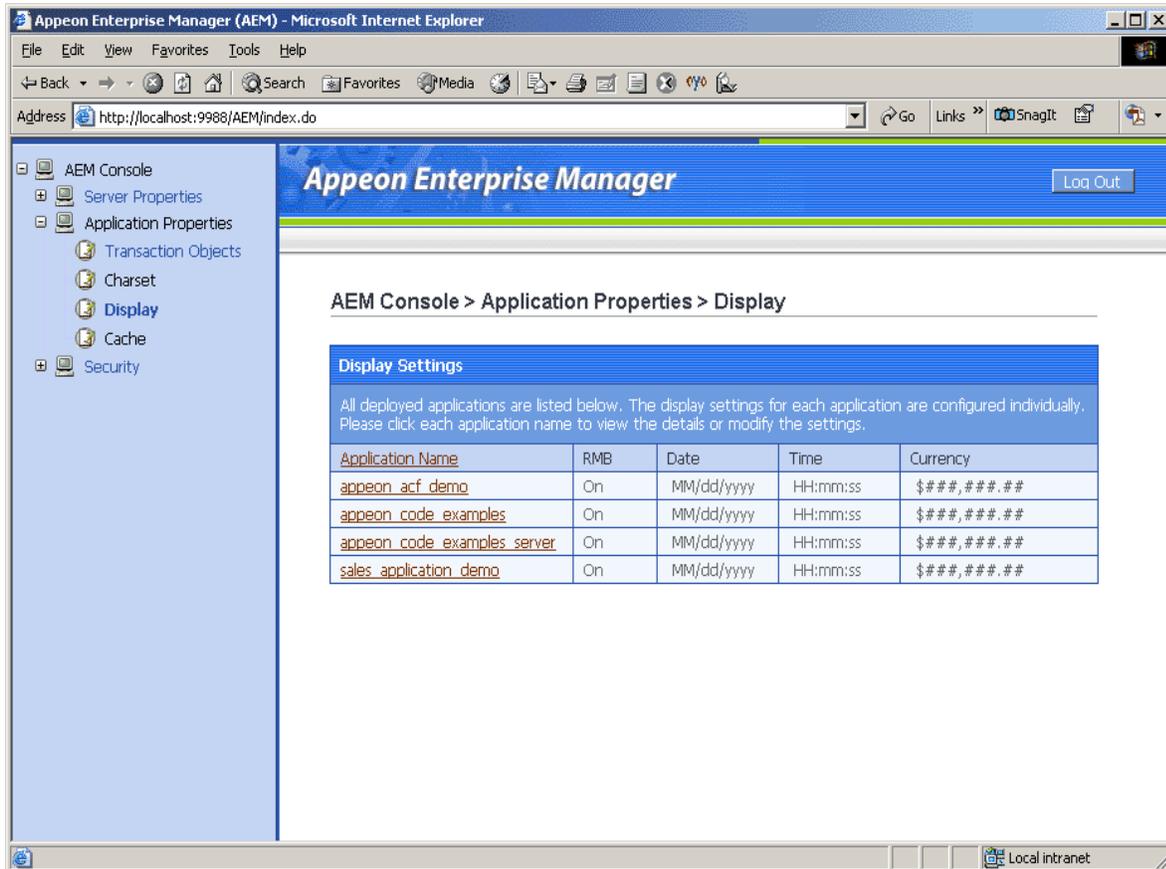
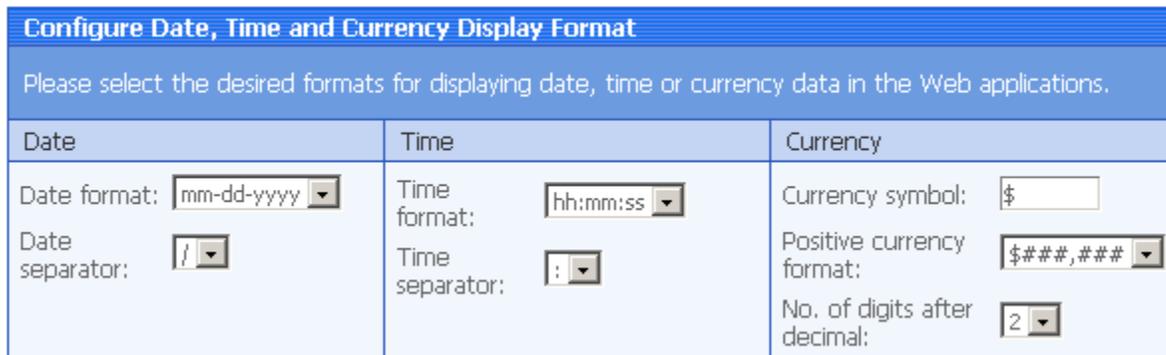


Figure 5-3: Appeon DataWindow Menu



Figure 5-4: Configure Date, Time and Currency Display Format



5.1.3 Charset

It is a connection cache level setting. This option can help you to solve the problem that error code displaying at the client side. To set the database charset and client input charset for a specified connection cache (Figure 5-3), the user can now select which application to adjust by clicking the applications name, as shown in Figure 5-2.

Figure 5-5 Charset

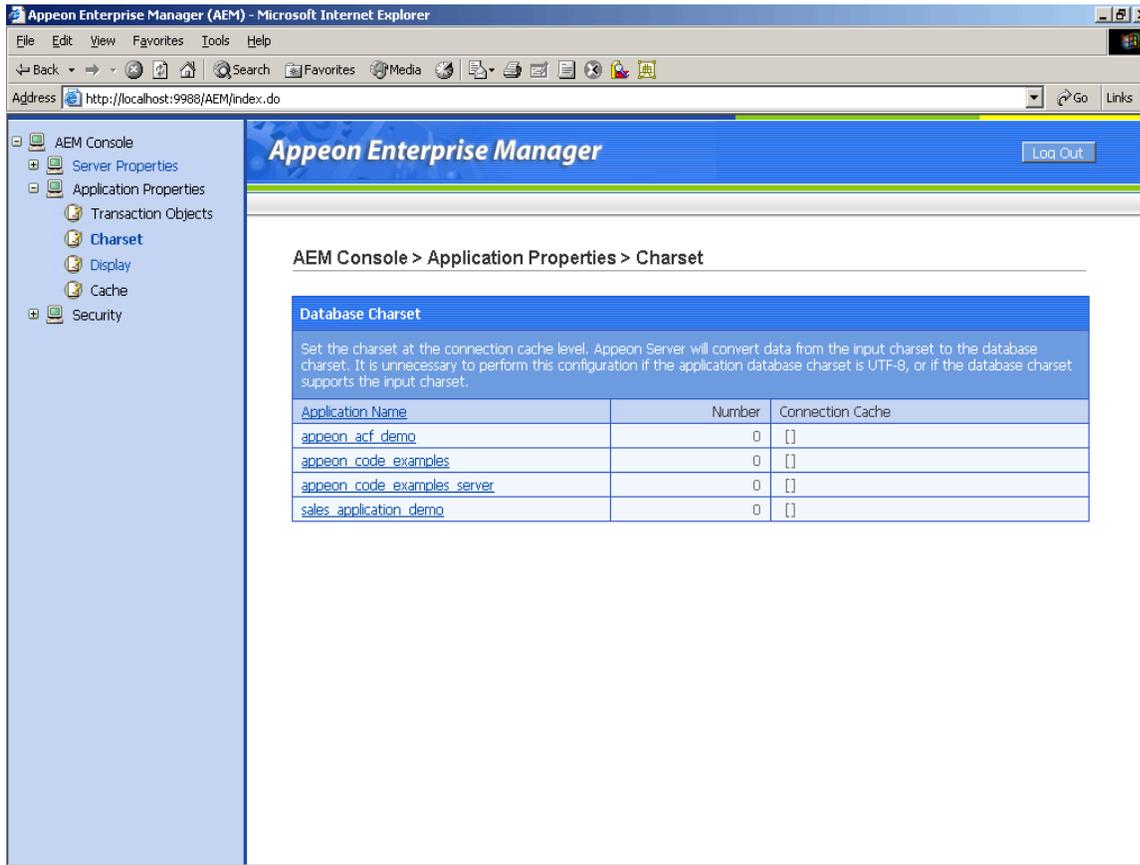


Figure 5-6 charset configuration

AEM Console > Application Properties > Charset > Add Charset > [apeeon_acf_demo]

[Click to return to previous page](#)

Add Charset

Application name:

Connection cache:

Database charset:

Input charset:

5.2 Transaction Timeout

This option sets the timeout period for transactions. This affects all transactions in an application.

5.3 Deployment Sessions

The Deployment Sessions tool can help you manage and monitor all deployment sessions. When the Developer is uploading the embedded SQL statement, the DataWindow, and INI files of an application to Apeon Server, the Deployment Sessions lists the application's name.

Figure 5-5: Deployment sessions

AEM Console > Server Properties > Deployment Sessions

Deployment Sessions		
The deployment sessions in server <i>localhost</i> are listed. Kill the session that is abnormally ended in the Apeon Developer machine.		
	Deployment Session ID	Application Name
<input type="checkbox"/>	645643453	new_feature
<input type="checkbox"/>	12854235	old_feature
<input type="radio"/> Check All <input type="radio"/> Clear All <input type="button" value="Kill Checked Sessions"/>		

5.4 Cache

You can now configure how Apeon Server handles its memory resources, including the ability to devote more resources to chosen applications.

Refer to the *Apeon Enterprise Manager User Guide* for more information.

Figure 5-6: Cache

AEM Console > Application Properties > Cache

Cache Settings	
All deployed applications are listed below. The cache setting for each application is configured individually. Please click each application name to modify its cache setting.	
Application Name	Cache
appeon_acf_demo	3 MB
appeon_code_examples	3 MB
appeon_code_examples_server	3 MB
sales_application_demo	3 MB

Figure 5-7: Modify the cache for an application

AEM Console > Application Properties > Cache > [appeon_acf_demo]

[← Click to return to previous page](#)

Modify Cache Setting.	
Please specify the cache size for storing the DataWindow syntax and SQL statements of the application. The default size is 3 MB. Setting the size to "0" means that no cache is available for the application. Setting the size to less than "0" means that the cache size has no limit.	
Cache size:	<input type="text" value="3"/> MB

6 Third-party Product Support

6.1 Apache Web Server

Appeon now provides Apache Web Server configuration files and instructions for using your Apache Web Server with Appeon 2.8. You must use the supplied files to configure Apache for use with Appeon 2.8. The configuration files reduce the amount of time required to configure an Apache Web Server for use with Appeon 2.8.

6.2 Netscape Web Server

Appeon 2.8 now supports Netscape Web Server for hosting Appeon deployed Web applications.

6.3 PowerBuilder 9.0

Appeon now supports PowerBuilder 9. Appeon Developer installs to the PowerBuilder 9 IDE and Appeon Server works with PBVM (PowerBuilder Virtual Machine) 9.

Appeon can now convert applications built with PowerBuilder 8 or PowerBuilder 9. Appeon can also convert applications upgraded to PowerBuilder 9 from PowerBuilder 8.

6.4 ASA 9.0

Appeon 2.8 supports Adaptive Server Anywhere 9.0.

6.5 PDFPrinter

Appeon 2.8 includes PDFPrinter for PDF printing. It is no longer necessary to purchase Adobe PDFWriter to utilize PDF printing for Appeon Server. Appeon supports Adobe Acrobat Reader 6.0 for viewing PDFs on the Client.

6.6 Oracle Stored Procedures

Appeon 2.8 is now able to support the following features when using the Oracle database stored procedure:

- Calling stored procedures or functions in a package or out of a package.
- Returning the result sets as IN OUT parameters.

Calling Oracle stored procedures using DECLARE Cursor statements or the DECLARE procedure.

Using the Oracle stored procedure as a Data Source in a DataWindow.

7 Other New Features

7.1 Internet Explorer title bar

The Internet Explorer title bar now displays the name of the application: "*Application Name* - Microsoft Internet Explorer". This helps navigation if you have multiple Web applications open at the same time.

Note: *ApplicationName* is the name of the Application object in the PowerBuilder IDE, and is the application name the user will see when configuring Application Properties in AEM.

8 Product UI Changes

8.1 Installation

8.1.1 Installation of ASA and EAServer

Installation now takes place providing EAServer 5.0. During the installation of Appeon Server. The user is given the option of not installing EAServer 5.0 if they already have EAServer 5.0 installed. EAServer 5.0 contains Adaptive Server Anywhere 8.0.2 database server runtime files. If EAServer 5.0 is installed, then ASA 8.0.2 is installed as well.

8.1.2 AppeonDB connection cache

AppeonDB is installed as an ODBC data source, used instead of the ASA service. AppeonDB connection cache now uses the iAnywhere driver instead of using the jConnect driver.

8.2 Appeon Developer

Table 8-1 New UIs added to the existing buttons

Button	Name	New UIs added
	Appeon Config	<ul style="list-style-type: none"> Added the <i>Customize UFA Report</i> tab page Added four tab pages in <i>Application Profile Properties</i>: <i>DB Profiles; Web; CmdArg; Misc.</i>
	Feature Analysis	<i>Select a target</i>
	Run Web App	<i>Select a target</i>

8.2.1 Apppeon Developer Configuration UI changes

Figure 7-1: Web report option

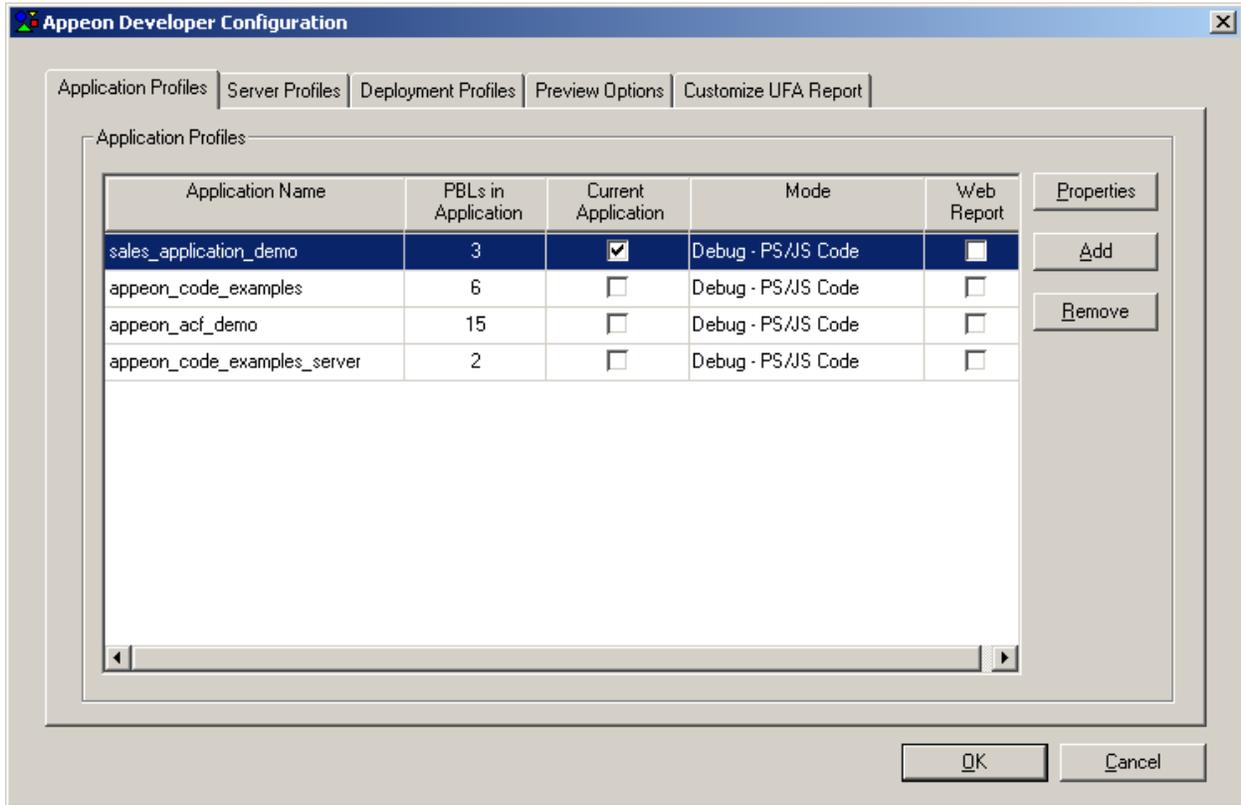


Figure 7-2: General settings

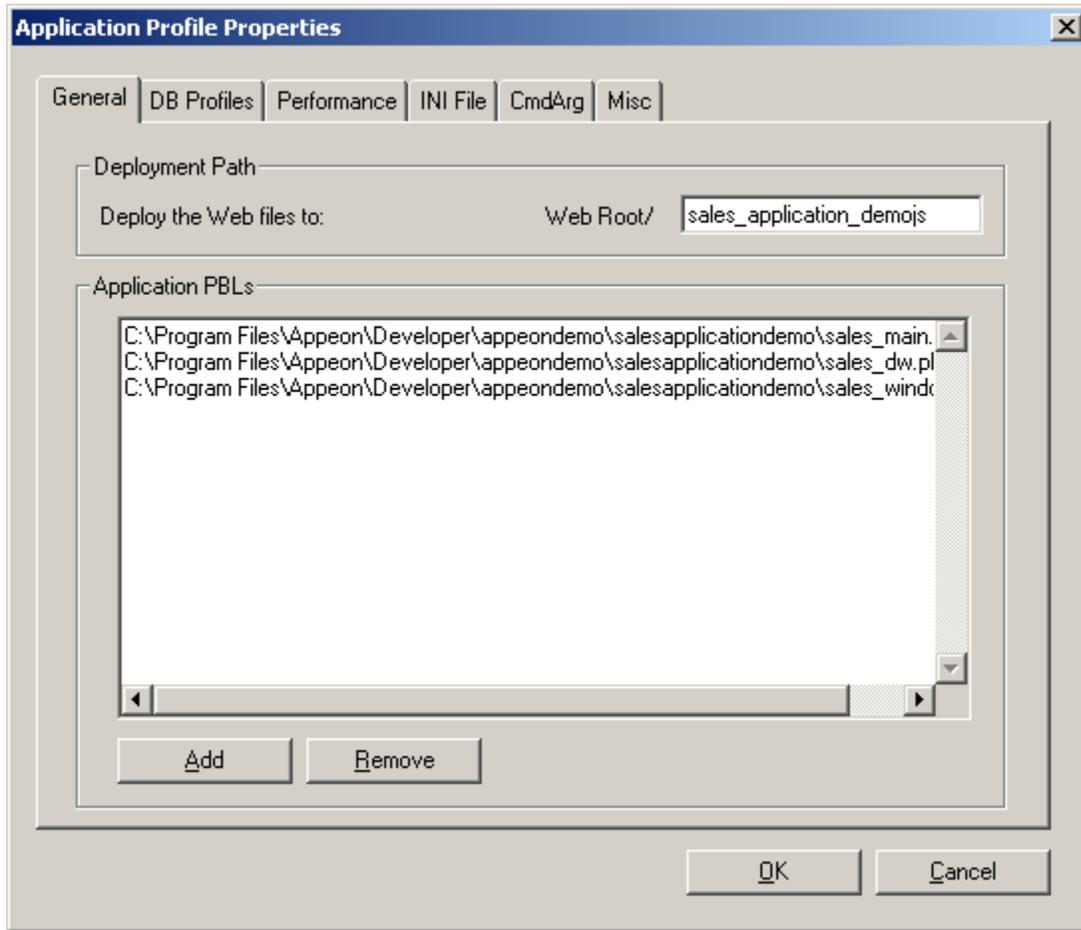
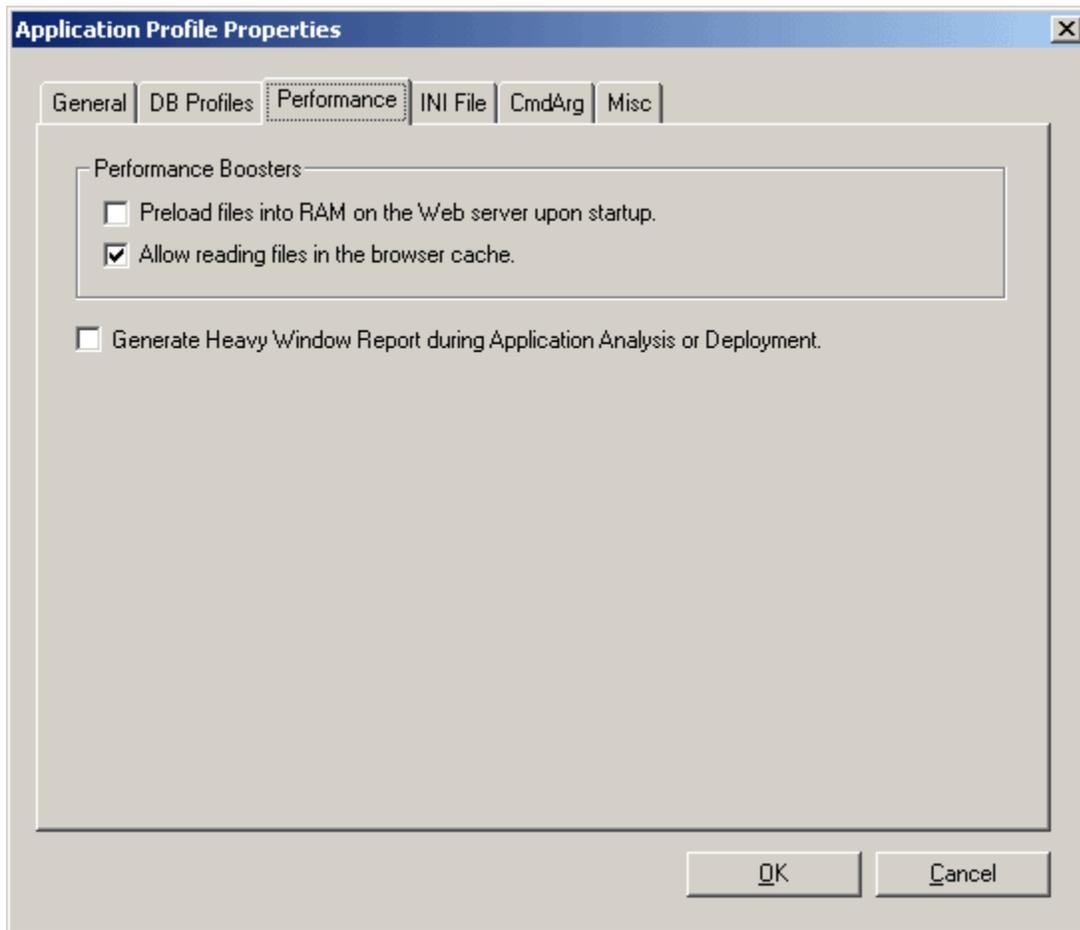


Figure 8-3: Generate Heavy Window Report during application analysis or deployment

8.2.2 Specify AEM settings for each configured Appeon Server

With this feature, the user can configure an individual AEM (as shown in Figure 7-4) for each Appeon Server profile as follows:

- Specify the URL including application server name, port, and AEM name.
- Select which connection method will be used for connecting to AEM, HTTP or HTTPS.

Figure 8-4: AEM setting in each Apeon Server profile

The screenshot shows a dialog box titled "Apeon Server Profile" with a close button (X) in the top right corner. The dialog is divided into four sections:

- Profile Settings:** A text field for "Server Name" containing "Local Apeon Server".
- Apeon Server Settings:** Four text fields: "Server IP" (localhost), "Server Port" (9989), "Username" (jagadmin), and "Password" (empty).
- AEM Settings:** A text field for "AEM URL" containing "http://localhost:9988/AEM". Below it, "Connection Method" has two radio buttons: "HTTP" (selected) and "HTTPS". A note below reads: "When using HTTPS, please ensure you have configured a listener for your SSL connection to this Apeon Server."
- Test Settings:** A text area containing the text: "After entering the Apeon Server settings, we recommend you test the settings by clicking the button below." Below this is a button labeled "Test Apeon Server Settings...".

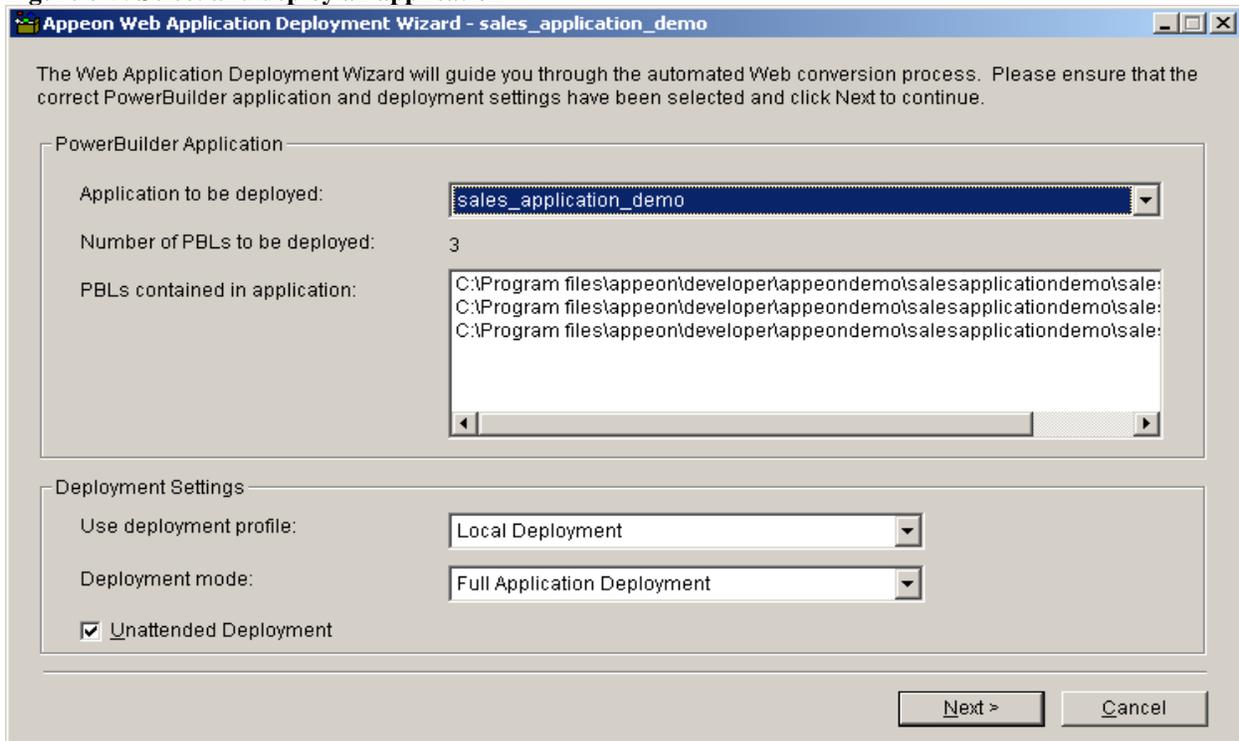
At the bottom of the dialog are "OK" and "Cancel" buttons.

When the user clicks the *AEM* button in the Apeon Developer toolbar, all Apeon Server profiles and corresponding AEM URLs configured will be listed.

- The Apeon Server Profiles are listed in alphabetical order.
- The Apeon Server configured as the Preview Server is selected and highlighted by default.

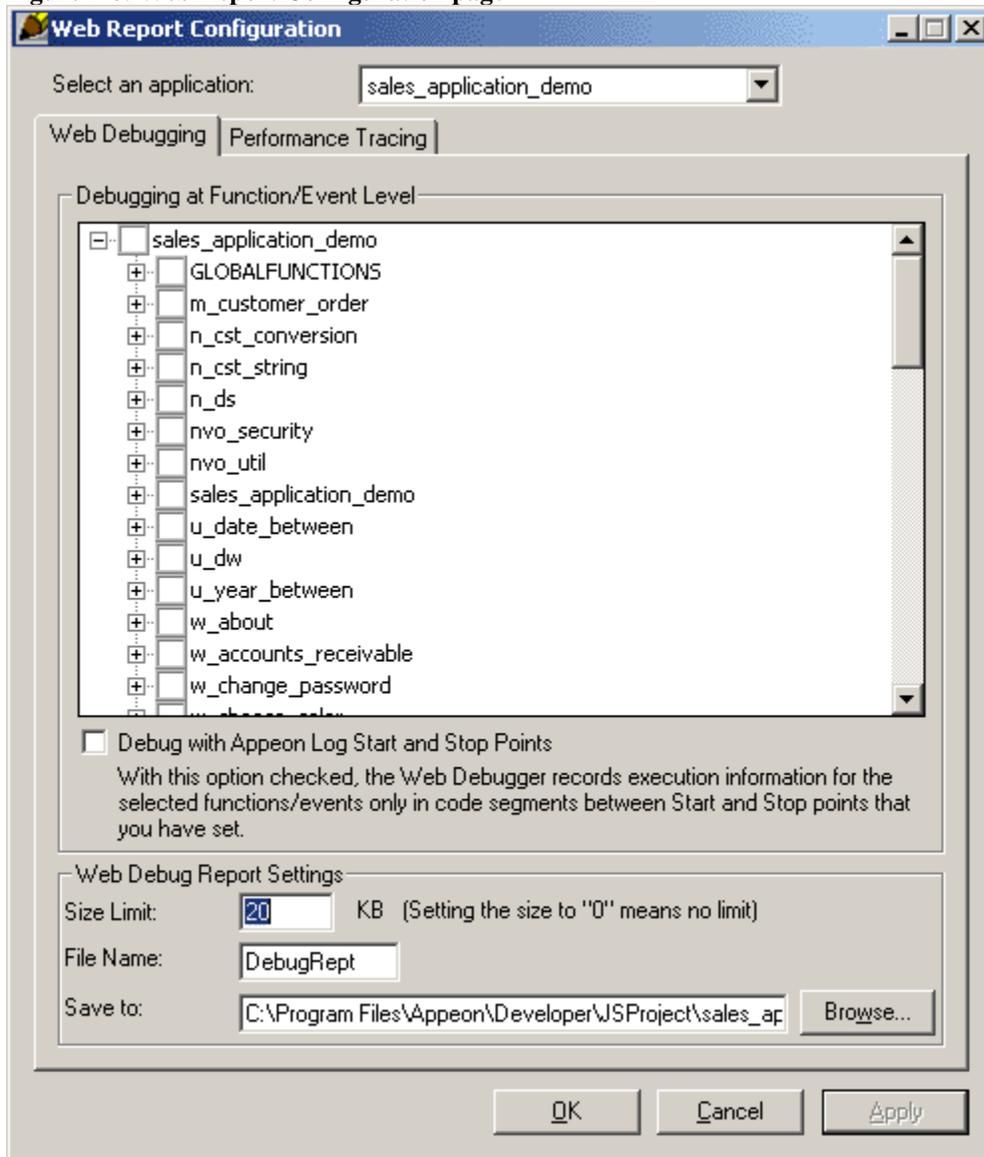
8.2.3 Apeon Deployment Wizard

The title bar of the Apeon Web Application Deployment Wizard is now more informative. The application name is now displayed, as shown in figure 7-5.

Figure 8-5: Select and deploy an application

8.2.4 Web Report Configuration tool

After the application is deployed and before the first time the application is run, configure the Web-debugging settings and performance-tracing settings to select the specific functions/events that will be tracked.

Figure 7-6: Web Report Configuration page

8.3 AEM

8.3.1 Overview

These sections describe UI changes to AEM tools in Apeon 2.8.

8.3.2 Server Properties

This section describes UI changes to two major *Server Property* tools: *Active Sessions* and *Active Transactions*.

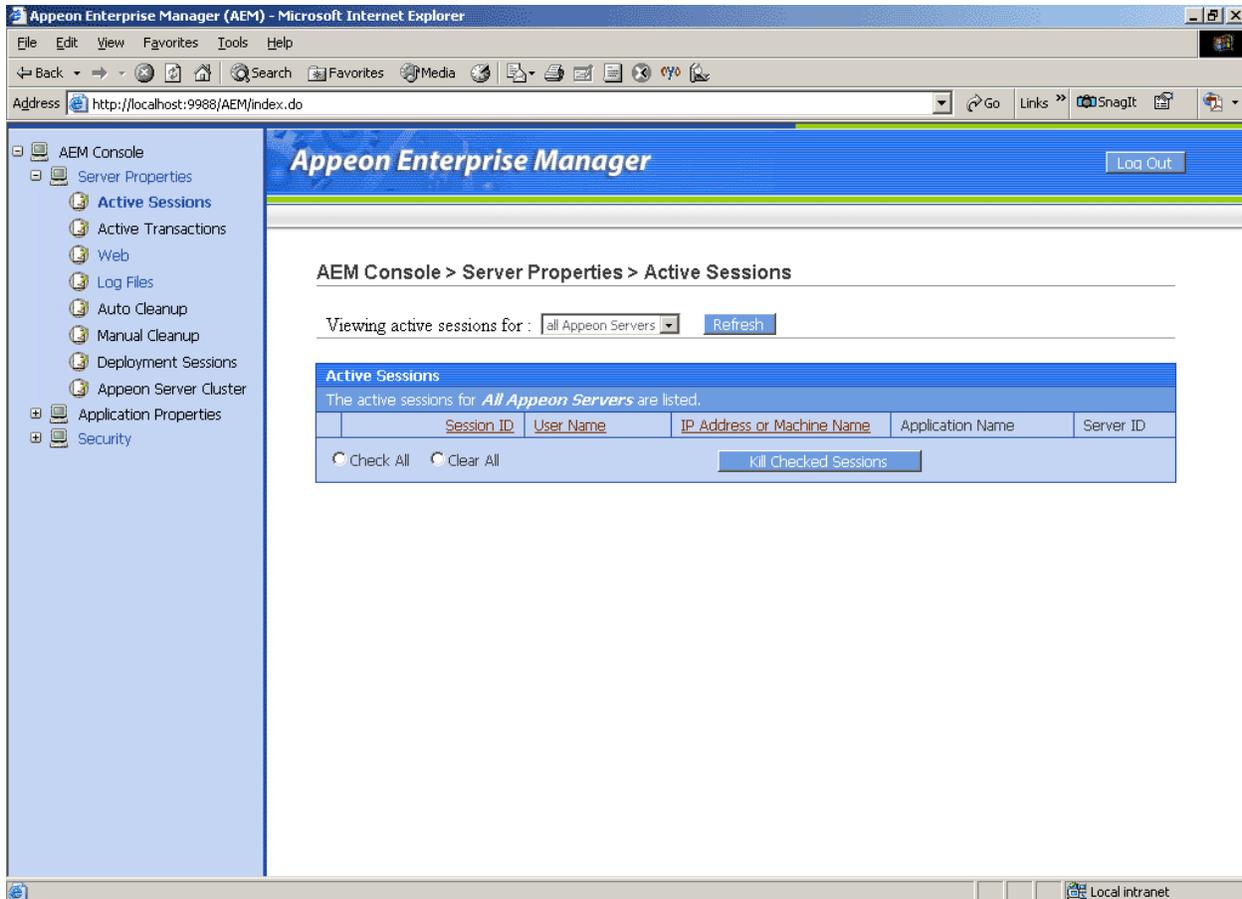
8.3.2.a Active Sessions

The *Refresh* button has been removed.

The table will now automatically refresh after the user selects which Appeon Server to view from the DropDownListBox.

Now the currently selected server name is displayed in bold and italic text.

Figure 8-7: Active Sessions



8.3.2.b Active Transactions

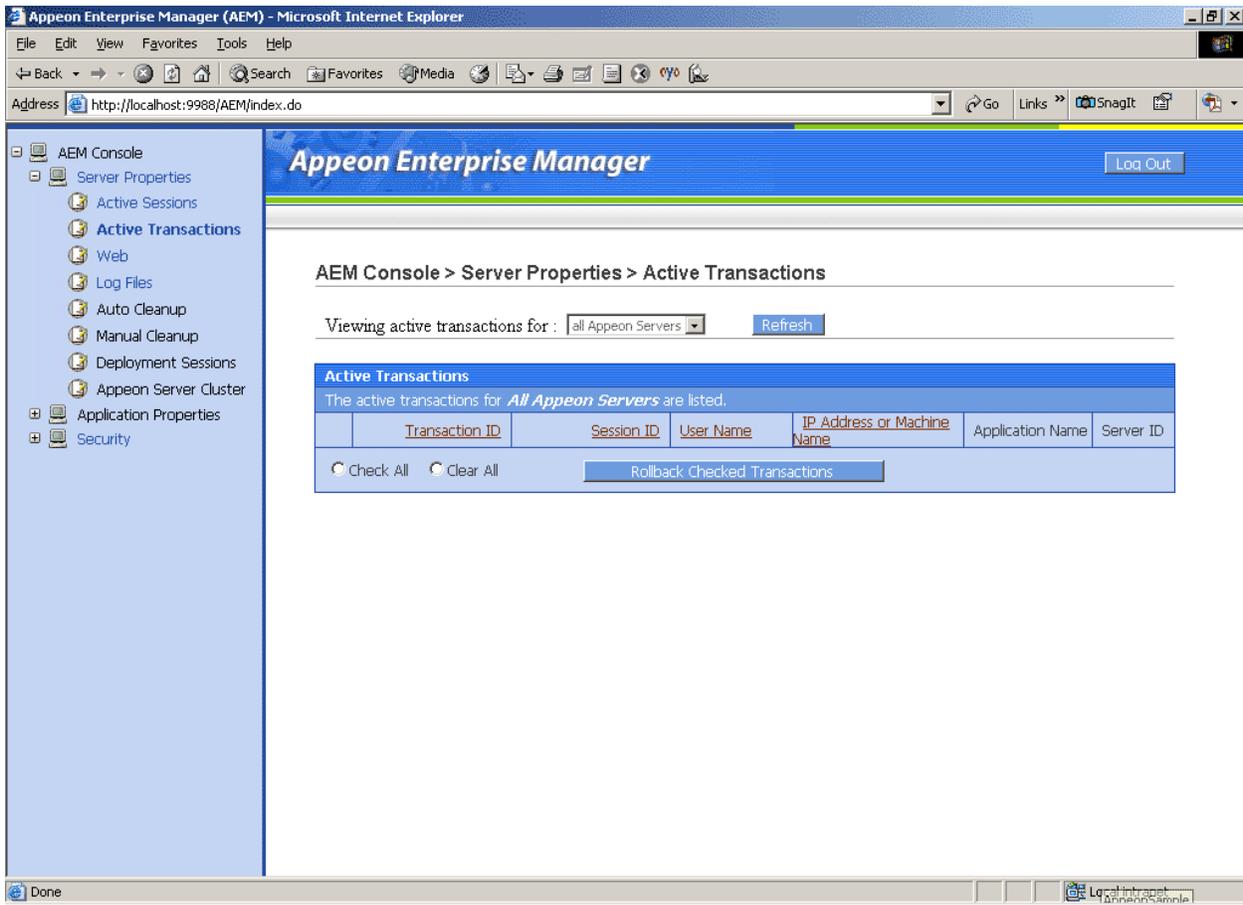
Figure 7-8 shows the Active Transactions display.

The *Refresh* button has been removed.

The table will now automatically refresh after the user selects an Appeon Server from the DropDownListBox.

Now the currently selected server name is displayed in bold and italic text.

Figure 8-8: Active Transactions



Index

1

10X data compression, 13

A

About This Book, 1

active session, 39

active transaction, 40

AEM

server properties, 39

active session, 39

active transaction, 40

AEM, 39

allow reading files in the browser cache, 21

apache web server, 31

apeon developer

apeon developer configuration UI
changes, 34

apeon developer wizard, 37

specify AEM settings for each configured
apeon server, 36

web report configuration tool, 38

apeon developer, 33

apeon developer configuration UI changes,
34

apeon developer wizard, 37

apeondb connection cache, 33

application profiles

CmdArg settings, 21

Misc, 22

performance

allow reading files in the browser cache,
21

preload files into RAM on the web
server upon startup, 21

application profiles, 18

Application profiles

database profiles settings, 18

Performance, 20

application properties

charset, 29

display, 27

transaction objects, 27

application properties, 27

ASA 9.0, 31

audience, 1

C

cache, 30

Call OLE objects, DLLs and run executable
programs on the Client

Client-side DLLs, 5

OLEObject object, 5

Run executable programs, 6

Call OLE objects, DLLs and run executable
programs on the Client, 5

changes for Apeon 2.7, client run-time, 8

charset, 29

CheckBoxes property, 6

Client-side DLLs, 5

CmdArg settings, 21

command line argument, 4

CreateOnDemand property of the Tab
control, 4

Customize Message global object, 4

customize UFA Report, 23

D

database profiles settings, 18

DataWindow improvements

drawing DataWindows, 14

improved performance when creating

DropDownListBoxes,

DropDownDataWindows and

DataWindows, 14

synchronization window removed, 14

DataWindow improvements, 14

deployment session, 30

display, 27

Dynamic transaction object to connection
cache mapping, 4

E

efficient object creation, 15

efficient Web file downloading, 15

expected performance for full deployment,
13

Expected performance for incremental deployment, 14

H

how to use this book, 1

I

if you need help, 2

Improved performance when creating DropDownListBoxes, DropDownDataWindows and DataWindows, 14

information manager reports and logs generated during the web migration process, 16

information manager, 16

installation
 apeondb connection cache, 33
 installation of ASA and EAServer, 33
installation, 33
installation of ASA and EAServer, 33
internet explorer title bar, 32

L

LongParm property, 7

M

Misc, 22

Multi-column sorting with space as the delimiter, 4

N

netscape web server, 31

O

OLEObject object, 5
oracle stored procedures, 31

P

PDFPrinter, 31
Performance, 20
performance impact of Internet, 12
PFC applications

 expected performance, 10
 PowerBuilder 9.0, 31
 preload files into RAM on the web server upon startup, 21

R

readers, 1
related documents, 1, 2
reports and logs generated during the web migration process, 16
Run executable programs, 6
Runtime benchmarks, 8
runtime benchmarks
 changes for Apeon 2.7, 8
 non-PFC applications
 expected performance, 8
 performance impact of Internet 10X data compression, 12
 PFC applications
 expected performance, 10
runtime report, 16

S

select and analyze an application, 24
select and run an application, 25
server properties, 39
ShareData and ShareDataOff, 7
specify AEM settings for each configured apeon server, 36
synchronization window removed, 14

T

transaction objects, 27
transaction timeout, 29

W

web application packaging wizard, 16
web deployment benchmarks, 13
web deployment benchmarks
 expected performance for full deployment, 13
web deployment benchmarks

Expected performance for incremental
deployment
, 14

web performance runtime tracing report
settings, 17
web report configuration tool, 38
Window icon, 6