

Appeon Supported Features Guide

Appeon® 6.1 for PowerBuilder®

DOCUMENT ID: DC20239-01-0610-01

LAST REVISED: October 2008

Copyright© 2008 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 of Appeon Corporation.

Sybase, PowerBuilder and Sybase Central are trademarks or registered trademarks of Sybase, Inc.

Java and JDBC are trademarks or registered trademarks of Sun Microsystems, 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 Objective	1
1.2 How to use this book.....	1
1.3 Related documents	2
1.4 If you need help	3
2 What's New in Appeon 6.0 for PowerBuilder	4
3 What's New in Appeon 6.1 for PowerBuilder	5
4 Basic Requirements and Recommendations	6
4.1 Overview.....	6
4.2 Basic Requirements	6
4.3 Application PBL requirement	6
4.4 Application language requirement	7
5 Web Browser Limitations	8
6 Application Techniques	9
6.1 Distributed application support (EAServer +Windows only).....	9
6.1.1 N-Tier PowerBuilder NVOs (EAServer + Windows only)	9
6.1.2 N-Tier DataWindows (EAServer + Windows only).....	10
6.2 Program access techniques	10
6.2.1 Using OLE in an application.....	10
6.2.2 Using external functions	12
6.2.3 Using Run PowerScript function	14
6.2.4 Building a mail-enabled application	14
6.2.5 Using Drag and Drop	15
6.3 General techniques.....	15
6.3.1 Using Unicode	15
7 Web Enhancements and Differences.....	17
7.1 Application enhancements and differences	17
7.1.1 Appeon security enhancement	17
7.1.2 Installing Appeon Active X control.....	17
7.1.3 Appeon Server open interfaces	18
7.1.4 Calling Appeon Server open interfaces in EAServer 5.x	21
7.1.5 Calling Appeon Server open interfaces in EAServer 6.1, WebLogic, Websphere.....	22
7.1.6 Application differences	23
7.2 DataWindow enhancements and differences	24
7.2.1 Appeon DataWindow menu.....	24
7.2.2 DataWindow printing.....	26
7.2.3 DataWindow user operation differences.....	27
7.3 Object/control enhancements and differences	27
7.3.1 Object/control enhancements.....	27

7.3.2 Object/control user operation differences	28
8 System Objects and Controls	29
8.1 Controls	29
8.1.1 Supported controls	29
8.2 System objects	136
8.2.1 Supported objects	139
8.2.2 Unsupported objects	201
9 PowerScript Reference	203
9.1 PowerScript Topics	203
9.1.1 Overview	203
9.1.2 Object-Oriented programming	203
9.1.3 Language basics	203
9.1.4 Data types	208
9.1.5 Declarations	212
9.1.6 Operators & expressions	216
9.1.7 Structures	217
9.1.8 User objects	219
9.1.9 Calling functions and events	223
9.1.10 Document Interface	226
9.2 PowerScript statements	227
9.3 Using PowerBuilder Source Editor	231
10 Embedded SQL	232
10.1 Database server and data types	232
10.2 Operators	234
10.3 Transaction management statements	235
10.4 Non-cursor statements	235
10.5 Cursor statements	239
10.6 Database stored procedures	239
10.7 Dynamic SQL	241
11 Functions	243
11.1 System functions	243
11.1.1 Supported types	243
11.1.2 Unsupported types	268
12 Events	270
12.1 Event types	270
12.2 System messages	270
12.3 System message (non-standard EventID)	277
13 DataWindow	279
13.1 DataWindow data sources	279
13.2 Using SQL statements in DataWindows	279
13.3 DataWindow presentation styles	279

13.3.1 Composite DataWindow	282
13.3.2 CrossTab DataWindow	283
13.3.3 Grouping in DataWindow	285
13.3.4 Graph DataWindow	287
13.3.5 RichText DataWindow	288
13.3.6 TreeView DataWindow	291
13.4 Displaying and validating data.....	293
13.5 Dynamic DataWindow.....	295
13.6 DataWindow operators and expressions.....	298
13.6.1 Specific Requirements for Using Expressions.....	300
13.7 DataWindow object and the properties.....	301
13.7.1 DataWindow object	301
13.7.2 DataWindow object properties	302
13.7.3 Controls in a DataWindow and their properties	307
13.8 DataWindow data and property expressions.....	329
13.9 DataWindow constants	330
13.10 DataWindow control.....	333
13.10.1 Properties of DataWindow control	333
13.10.2 Events of DataWindow control.....	334
13.10.3 Functions of DataWindow control.....	335
13.11 DataStore object.....	340
13.11.1 Properties of DataStore object	340
13.11.2 Events of DataStore object.....	340
13.11.3 Functions of DataStore object.....	341
13.12 DataWindowChild object.....	343
13.12.1 Properties of DataWindowChild object.....	343
13.12.2 Functions for DataWindowChild.....	343
13.13 DataWindow performance considerations.....	346
14 DBParm parameters in Database.....	347
15 Web Services Client.....	348
15.1 SoapConnection object	348
15.2 SoapException object.....	348
16 Undetected features	350
16.1 Overview.....	350
16.2 Undetected Features Listing	350

1 About This Book

1.1 Objective

This guide gives a detailed list of what PowerBuilder features are supported and can be converted to the Web and what features are unsupported.

Note: Unless specified, the supported features in this Help document work with Appeon Server for EAServer, WebLogic, and WebSphere; a feature marked "**EAServer only**" only works with Appeon Server for EAServer.

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: What's New in Appeon 6.0 for PowerBuilder

Lists the major new features added in Appeon 6.0 since Appeon 5.1 for PowerBuilder.

Chapter 3: What's New in Appeon 6.1 for PowerBuilder

Lists the major new features added in Appeon 6.1 since Appeon 6.0 for PowerBuilder.

Chapter 4: Basic Requirements and Recommendations

Please make sure your application meets the requirements outlined in this section.

Chapter 5: Web Browser Limitations

Lists the limitations for using Internet Explorer 7.

Chapter 6: Application Techniques

Instructions on how to apply Appeon-supported application techniques including distributed application support and program access techniques.

Chapter 7: Web Enhancements and Differences

Lists the Web enhancements that an Appeon Web application can take, and the possible differences it will have with the original PowerBuilder application.

Chapter 8: System Objects and Controls

Provides the supported features list for PowerBuilder system objects and controls.

Chapter 9: PowerScript Reference

Provides the supported features list for PowerBuilder programming language, PowerScript.

Chapter 10: Embedded SQL

Provides the supported features list for database servers, database operators, database statements.

Chapter 11: Functions

Lists the supported and unsupported system functions, and describes what type of user functions are supported or unsupported.

Chapter 12: Events

Lists the supported event types and system messages.

Chapter 13: DataWindow

Gives a detailed list on the supported features for DataWindow.

Chapter 14: DBParm parameters in Database

Lists the supported DBParm parameters.

Chapter 15: Web Services Client

Provides the supported interfaces that enable developers to extend the functionality of PowerBuilder.

Chapter 16: Undetected features

Lists unsupported features that are not detected during Unsupported Features Analysis.

1.3 Related documents

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

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 What's New in Appeon 6.0 for PowerBuilder

These are major new features added to Appeon® 6.0 for PowerBuilder®.

PowerBuilder version support	Supported
	<p>Supports PowerBuilder 11 applications.</p> <p>Note: DatePicker, MonthCalendar, TreeView DataWindow, are the only new features that we supported since PowerBuilder 10.</p>
DatePicker control	Supported
	<p>Most properties, events and functions for DatePicker control are supported.</p>
MonthCalendar control	Supported
	<p>Most properties, events and functions for MonthCalendar control are supported.</p>
DataWindow	Supported
	<p>TreeView DataWindow is supported.</p> <p>More DataWindow object properties are supported: CollapsedTreeNodeIconName, DefaultExpandToLevel, ExpandedTreeNodeIconName, Indent, Level, SelectNodeByMouse, ShowNodeConnectLines, ShowLeafNodeConnectLines, ShowTreeNodeIcon, StateIconAlignMode, and TreeNodeIconName.</p> <p>More Functions of DataWindow control are supported: Collapse, CollapseAll, CollapseAllChildren, CollapseLevel, Expand, ExpandAll, ExpandAllChildren, ExpandLevel, IsExpanded, and SelectTreeNode.</p>
PowerScript features	Supported
	<p>Extend Decimal precision up to 28 digits in DataWindow. Details refer to Standard data types</p>

3 What's New in Appeon 6.1 for PowerBuilder

The following new feature is supported by Appeon® 6.1 for PowerBuilder® since version 6.1.

PowerBuilder version	Supported
	Supports PowerBuilder 11.5 applications.
Database version	Supported
	Supports Sybase ASA 11 with iAnywhere JDBC driver and jConnect JDBC driver.
EAServer version	Supported
	Extends the support to EAServer 6.2.
Application Object	Supported
	Supports the ToolbarText property of the Application object.
Others	Supported
	Appeon Active X can be silently installed to the client machine.

4 Basic Requirements and Recommendations

4.1 Overview

Appeon® for PowerBuilder® is able to deploy most PowerBuilder features to Web successfully. However there are still some unsupported features to remove, general guideline to follow and common pitfalls to avoid during the Web migration process. An application that contains unsupported features and does not adhere to some basic requirements and architectural requirements, may fail to migrate to the Web and suffer from performance problems.

This book will help you:

1. Figure out what PowerBuilder features are supported in Appeon and what are not;
2. Understand Appeon Web features.

4.2 Basic Requirements

There are some basic requirements for the PowerBuilder application which is going to move to the Web with Appeon.

1. The PBLs size is suggested less than 500MB. Appeon has successfully tested a PFC application that was artificially inflated to over 500MB.
2. The code is upgraded to be 100% compatible with Appeon-supported environment. Details refer to the Installation Requirements section in Appeon Installation Guide.
3. The application can generate .exe files (with no errors) in the PowerBuilder version that you install Appeon Developer to.
4. The application has been tested thoroughly to be bug-free?

4.3 Application PBL requirement

The application code must be 100% PowerBuilder 9 or above code.

Non-PowerBuilder 9 and above code, including code that is defined as obsolete in PowerBuilder 9 or above help, must be replaced/upgraded before you can begin to use Appeon for PowerBuilder. In addition, supported PowerBuilder 9 or above code is demonstrated in the PowerBuilder Help. Undocumented syntax is unsupported.

All objects that are referenced by PBLs need to be available - PBLs cannot point to objects that do not exist. If a PBL does reference an unavailable object, the Appeon Web Application Deployment process will crash.

Upgrading DataWindow objects to PowerBuilder 9 or above

- In PowerBuilder 9 or above, the DataWindow column name cannot be null (in previous versions of PowerBuilder, this was allowed). Ensure that the column names of DataWindows are not null.
- In the earlier versions of PowerBuilder, the edit style of a DataWindow could be null. The user must make sure that after upgrading to PowerBuilder 9 or above, the edit styles of the DataWindows are not null.
- In the earlier versions of PowerBuilder, the user could use the GetChild function to provide a reference to a child DataWindow in the DataWindow itself.

- If an application is updated to PowerBuilder 9 or above, some DataWindow columns may have duplicate names. Duplicate names for columns in a DataWindow are unsupported.
- If a DataWindow is not correctly upgraded to a PowerBuilder 9 or above format, the problematic DataWindow(s) will be reported with the following information: PBL > DATAWINDOW NAME.

Upgrading Menu objects to PowerBuilder 9 or above

- With menu inheritance, the source code in earlier versions of PowerBuilder is different from the source code in PowerBuilder 9 or above. The user must open the menu objects with inheritance in PowerBuilder 9 or above and upgrade the objects to PowerBuilder 9 or above objects.

Upgrading color specification to PowerBuilder 9 or above

A color value in an earlier PowerBuilder version may map to a different color of the same value in PowerBuilder 9 or above. If you prefer the color of the earlier version, be sure to verify that the correct colors are selected when upgrading the application to a PowerBuilder 9 or above application.

4.4 Application language requirement

Text in the user interface (UI) of the application and application data (from the database) must be in one of the following five languages:

- English
- Simplified Chinese
- Traditional Chinese
- Korean
- Japanese

Other languages can potentially be supported, but they have not been tested.

5 Web Browser Limitations

Appeon Web applications run well in the Internet Explorer with two known issues:

- Multiple Appeon applications cannot be run in the same Internet Explorer process.
- When the Internet Explorer is refreshed, the Web application will restart.

If you are an Internet Explorer 7 user, there are more known issues:

- IE 7 supports tabbed browsing, but an Appeon application cannot be opened in more than one tab, and applications deployed in the same Appeon version cannot be opened in tabs in a single browser window, because Appeon applications are single-threaded.
- IE 7 supports page zoom, enlarging individual Web pages, including both text and graphics, to either focus on specific content or to make content more accessible to those with vision limitations, however, this feature has no effect to Appeon application Web pages and places the UI elements in disorder.
- When a response window of the Appeon Web application is opened in IE 7, if you click the area out of the response window, the focus will be moved away from the window and you will not be able to set focus to the response window again.
- When an Appeon Web application is opened in IE 7, if you move the focus which is currently on the treeview of an MDIClient window to a different tab which has opened a non-Appeon application, and then switch back to the Appeon Web application, the toolbar of IE 7 will not be able to get the focus.
- When IE browser is displayed in full screen, the popup menu on the Appeon Web application will be closed automatically, because IE 7 automatically switches the focus.
- When an Appeon Web application is opened in IE 7, thumbnail images are unsupported.
- If the shortcut defined in PowerBuilder application is the same as the shortcut of IE 7, shortcuts in IE 7 have a higher priority when it converts to the Web.

6 Application Techniques

6.1 Distributed application support (EAServer +Windows only)

Work around Unsupported Features

Apeon supports N-Tier NVOs as well as the distributed DataWindow technique, which synchronizes a DataWindow control on a client with a DataStore on a server using GetFullState, SetFullState, GetChanges and SetChanges. These features can be applied to work around most unsupported features as well as connectivity to Web Services, Java Corba, EJB, C/C++ DLLs (Windows only), and ActiveX components, etc.

The basic requirement for migrating distributed PowerBuilder applications is as follows: all components directly called by the PowerBuilder client must be hosted in one Apeon Server that is used for Web deployment of the PowerBuilder client, and the component type must be a PowerBuilder NVO or a Java Corba component.

The N-Tier NVO that is hosted in one Apeon Server can serve as a "bridge" to call different kinds of middle-tier components that are hosted in different servers.

For instance, if your application calls both .NET components in Web Services and some Java components in EAServer, add an intermediate layer of PowerBuilder NVO components to EAServer: Encapsulate the .NET components and deploy them to EAServer as N-tier NVOs. Modify the application so that it can call the deployed NVO in EAServer instead of directly invoking the .NET components.

Distributed applications involve the use of the PowerBuilder Connection object. Refer to *Connection Object* for information on supported and unsupported features.

6.1.1 N-Tier PowerBuilder NVOs (EAServer + Windows only)

Supported

- The variables used in n-Tier PowerBuilder NVO can be the following standard PowerScript data types:

Boolean	Blob	Char	Date
DateTime	Decimal	Double	Int
Integer	Long	Real	String
Time	UInt	ULong*	UnsignedInt
UnsignedInteger*	UnsignedLong*		

- Functions in NVO components can use UnsignedLong or UnsignedInteger as arguments, but not as return values.
- Functions in NVO components can use array, structure and reference as arguments.
- The parameter and return value of an NVO public interface can be arrays, complex data types, or structures.
- Declaring external functions (such as API functions, DLLs) in the NVO is supported if Apeon Server runs on Windows. External Function DLLs in Apeon Server/EAServer PowerBuilder components have no issues from PowerBuilder's perspective, but if the external function is not thread-safe, that can certainly cause a problem. It's dependent on the DLL's implementation, which has nothing to do with

Appeon Server/EAServer. If it's not threadsafe you can set the `com.sybase.jaguar.component.thread.safe` property of the PowerBuilder component to false and that will prevent multiple threads from trying to access the external function simultaneously.

Differences

- The precision of the decimal number, which is the return value of NVO components, is different from that in PowerBuilder and on the Web. This is caused by the different precision between PowerScript and JavaScript.

Unsupported

The n-Tier PowerBuilder NVOs cannot use the following features:

- PowerScript MessageBox function in the NVO
- Global variables
- Visual controls or objects
- Any visual control/object data type as the parameters for NVO functions and events

For calling and using remote EAServer PowerBuilder NVO components in the PowerBuilder client, some limitations are listed below:

- The proxy object for an NVO cannot use the same name as the NVO.
- It is impossible to use Dot Notation to refer to the instance variables of a remote NVO. Appeon suggests that you add functions to the remote NVO to get/set its instance variable values.
- You cannot trigger or post the events of a remote NVO in the PowerBuilder client.
- The script for an NVO or the script referred to in an NVO cannot contain UI-related scripts or objects.
- Web applications are unsupported to call NVOs that contain inheritance hierarchy.

6.1.2 N-Tier DataWindows (EAServer + Windows only)

Appeon Server installed to EAServer on Sun Solaris can only support n-Tier DataWindows that do not call the `GetChanges`, `GetFullState`, `SetChanges`, and `SetFullState` functions.

Appeon Server installed to EAServer on Windows can support n-Tier DataWindows that call the `GetChanges`, `GetFullState`, `SetChanges`, and `SetFullState` functions, as long as the Appeon Workaround PBL is implemented.

For more instructions on how to use the Appeon Workaround PBL, please refer to the Distributed DataWindows section in Appeon Workarounds Guide.

6.2 Program access techniques

6.2.1 Using OLE in an application

OLE, originally an acronym for Object Linking and Embedding, is a facility that allows Windows programs to share data and program functionality. Appeon supports the use of OLE in three ways:

- OLE control

- OLECustomControl
- OLE object

OLE controls	Supported
	OLEControl control OLECustomControl control
	Unsupported
	OLE Control of DataWindow
OLE objects	Supported
	OLEObject object
	Unsupported
	OLEStorage OLEStream OLERuntimeError OLERuntimeOLETxnObject Large Binary/Text database OLE Object of DataWindow
Calling OLE object functions	Requirements
	Using parentheses when calling an OLEObject method is required, even if the OLEObject method does not require any parameter. For example, when executing the Save function for a Microsoft Word document: o1.ActiveDocument.Save //Unsupported o1.ActiveDocument.Save() //Supported
	Unsupported
	1) OLEObject object does not support cascaded calling. The following example is unsupported: OleObject.function1.function2() 2) The OLE object method cannot contain reference parameters.
Shortcut	Unsupported
	Using shortcuts in the OLECustomControl control are unsupported.
Requirements for	Requirements

client environment	<p>When an OLE object is called in a deployed application, the location used to store the OLE object on the client should match the location specified in the original PowerBuilder application.</p> <p>ActiveX control(s), if used in the Web application, should be installed and registered on the client. There are two ways to install and register ActiveX controls:</p> <p>Manually install and register ActiveX controls that are associated with the Web application.</p> <p>Deploy ActiveX controls that are associated with the Web application to Apeon Web Server, so ActiveX controls can be automatically downloaded and registered on the client. For detail information on deploying ActiveX controls to the Web Server, refer to the <i>Apeon Developer User Guide</i>.</p>
---------------------------	---

6.2.2 Using external functions

Data types of external function arguments	Supported				
	Array	Boolean	Blob	Char	Date
	DateTime	Double	Decimal*	Integer	Long
	Real	String*	Structure	Void	
	String The length of an argument or return value as String data type cannot be larger than 512 characters.				
	Decimal The precision of Decimal argument is the same as the precision of Double data type.				
	Unsupported				
Data types of external function return values	LongLong	Object	UInt	UnsignedInt	UnsignedInteger
	UnsignedLong				
	Supported				
	Array	Boolean	Blob	Char	Date
	DateTime	Double	Decimal*	Integer	Long
	Real	String*	Unsigned Integer	Unsigned Long	Void
	String The length of an argument or return value as String data type can not be larger than 512 characters.				
Decimal The precision of Decimal argument is the same as the precision of Double data type.					
Unsupported					
DLLs or	LongLong	Structure	Object		
	Supported				

<p>shared libraries Location</p>	<p>When your application calls an external function, it must be able to find the DLL (Dynamic link library) or the shared library in which that function resides. To ensure this, you must update the library path environment variable on the client machine to include the directory where that DLL or shared library is stored. The directory is searched in the following order of precedence:</p> <p>Windows system directory</p> <ul style="list-style-type: none"> • Windows 2000/2003: The 32-bit Windows system directory. Use the GetSystemDirectory function to get the path of this directory. The name of this directory is SYSTEM32. or • Windows 2000/2003: The 16-bit Windows system directory. There is no Win32 function that retrieves the path of this directory, but it is searched. The name of this directory is SYSTEM. <p>The Windows directory</p> <ul style="list-style-type: none"> • Use the GetWindowsDirectory function to get the path of this directory. The directories that are listed in the PATH environment variable. When you make changes to the environment variables, the environment variables take effect only after you restart the machine.
<p>Ensuring the resource file in an external function when running Web application</p>	<p>Supported</p> <ol style="list-style-type: none"> 1. If the file is an Image, directly specify the file name. When you deploy the application with Apeon Developer, you need to deploy the Image, DLL or OCX file with the application Web files to the Web server. Refer to the <i>Application profile configuration</i> section in the <i>Apeon Developer Guide</i> for instructions. 2. If the file type is not Image, DLL or OCX, specify the file name and absolute file path. The client that runs the application must have the file ready in the same path before executing the external function. 3. If the file is a DLL or OCX, there are two ways to ensure the resource file in an external function: <ul style="list-style-type: none"> • Directly specify the file name. When you deploy the application with Apeon Developer, you need to deploy the Image, DLL or OCX file with the application Web files to the Web server. Refer to the <i>Application profiles configuration</i> section in the <i>Apeon Developer Guide</i> for instructions. • Package the file using the DLL/OCX file packaging wizard. Installed them to the client before the application is run for the first time. Refer to the <i>Packaging DLL OCX files</i> section in the <i>Apeon Developer User Guide</i>.
<p>Others</p>	<p>Supported</p> <p>Calling external functions in Win32-based DLLs (including both the system DLLs and the user-defined DLLs) is supported. API for file and directory access operations is supported.</p> <p>Unsupported</p> <p>Apeon does not support calling polymorphism external functions. Graphs created by Window API function may not be displayed on the Web.</p>

6.2.3 Using Run PowerScript function

Syntax	<p>Supported</p> <p>Run (string {, windowstate })</p> <p>The value of a string argument can be a filename without a path or extension. The following examples are supported:</p> <pre>run ("notepad") run ("notepad.exe") run ("C:\winnt\system32\notepad") run ("C:\winnt\system32\notepad") run ("C:\winnt\system32\notepad.exe")</pre>
Application programs location	<p>Supported</p> <p>When you call the Run function and do not specify the path of the application program to the string argument, make sure that the application program is stored in one of the following directories. If it is not, an execution error will occur. The directory is searched in the following order:</p> <ol style="list-style-type: none"> 1. Windows system directory <p>Windows 2000/2003: The 32-bit Windows system directory. Use the GetSystemDirectory function to get the path of this directory. The name of this directory is SYSTEM32. Or</p> <p>Windows NT/2000: The 16-bit Windows system directory. There is no Win32 function that retrieves the path of this directory, but it is searched. The name of this directory is SYSTEM.</p> <ol style="list-style-type: none"> 2. The Windows directory <p>Use the GetWindowsDirectory function to get the path of this directory.</p> <ol style="list-style-type: none"> 3. The directories that are listed in the PATH environment variable. <p>When you make changes to the environment variables, the environment variables take effect only after you restart the machine.</p>

6.2.4 Building a mail-enabled application

Apeon supports MAPI (messaging application program interface), so you can enable your Web applications to send and receive messages using any MAPI-compliant electronic mail system.

Apeon supports the following mail related objects and structures to enable the mail service.

A mail-related system object	<p>Supported</p> <p>mailSession object</p>
Mail-related	Supported

structures	mailMessage object	mailFileDescription object	mailRecipient object
	Unsupported		
	Properties of the three structures cannot be NULL.		
Object-level functions for the MailSession object	Supported		
	mailAddress	mailDeleteMessage	mailGetMessages
	mailHandle	mailLogoff	mailLogon
	mailReadMessage	mailRecipientDetails	mailResolveRecipient
	mailSaveMessage	mailSend	
Enumerated data types	Supported		
	MailFileType	MailLogonOption	MailReadOption
	MailRecipientType	MailReturnCode	

6.2.5 Using Drag and Drop

Drag and drop allows users to initiate activities by dragging a control and dropping it on another control. Except OLE, drag and drop is supported for other draggable controls and objects with features listed in the table below:

Drag and Drop properties	Supported			
	DragAuto	DragIcon		
Drag and Drop Events	Supported			
	BeginDrag	BeginRightDrag	DragDrop	DragEnter
	DragLeave	DragWithin		
	Unsupported			
	If the DragIcon is not specified for a control except the ListView or TreeView control, pressing the left mouse button without moving will trigger both DragEnter and DragWithin events in PowerBuilder, however, DragEnter event only on the Web.			
Drag and Drop Functions	Supported			
	Drag	DraggedObject		
Drag mode	Unsupported			
	Key event is unsupported if controls are in drag mode			
UI differences during the drag and drop	Difference			
	(For all supported draggable controls except the TreeView and ListView controls) The icon of the mouse will not change, if it is not on the client area of a control. (For TreeView and ListView) When you drag a control, you cannot see the mouse icon during the drag. Apeon does not support dragging and dropping multiple controls simultaneously. You can only drag and drop one control at a time.			

6.3 General techniques

6.3.1 Using Unicode

Apeon supports the Unicode character encoding in PowerBuilder 10. You can display characters from multiple languages on the same page of a Web application. Since 3.1,

Apeon provides support to other Unicode related functions for backward compatibility with earlier versions of PowerBuilder.

Supported

- **Encoding enumeration**
Apeon supports the optional encoding parameter in the Blob, BlobEdit, SaveAs and String functions.
- **String related functions**
Apeon supports the string related functions in both PowerBuilder 9 and PowerBuilder 10. The following changes, made in PowerBuilder 10, are also supported.
 1. Modified processing of string-manipulation functions.
 2. Asc and Char functions
 3. Blob and String conversion functions.
- **Support ANSI and Unicode database**
Apeon supports both ANSI and Unicode database. For Unicode database it is recommended to use iAnywhere and jConnect driver because using JDBC-ODBC driver with Unicode databases may cause problems.
- **Calling external functions**
Apeon supports both Unicode encoding and ANSI encoding when the Web application calls external functions.

Unsupported

Apeon does not support the optional encoding parameter in File-related functions.

7 Web Enhancements and Differences

7.1 Application enhancements and differences

7.1.1 Appeon security enhancement

If your PowerBuilder application has no coded username/password verification during application startup, Appeon's built-in user group management can assist you. Refer to the *Configuration during security management* section in the *Appeon Server Configuration Guide* for more information. When using Appeon's built-in security, you will be prompted to enter a user name and password in the Appeon Login Web dialog box.

7.1.2 Installing Appeon Active X control

Originally, there are two ways to install Appeon Active X. One way is Appeon Active X will be automatically downloaded to the client when you access the Web application with correct IE settings and user privilege. The other way to install Appeon Active X is to manually download the Active X control to the client and double click the ax_install.msi file to install it.

Starting from Appeon 6.1, Appeon provides the following command-line option to enable you manually install the MSI file. You can also use those command-line options together with the MSI default options.

Option	Description
-u	Uninstalls Appeon Active X.
-s	Silently installs Appeon Active X.
-x	Prompts the failure message.
-r	Reverses the old files.
-l	Prints log file.

Following are some examples on how to use Appeon command-line options. You should execute these command lines in the Command Prompt window.

1. Use Appeon command-line options to silently install the Appeon Active X control and prompts message if there is any failure during the installation. With this command you cannot see any user interface related to the installation of the Active X, however, MSI installer related user interface will be displayed. Note the MSI related windows do not require user interaction.

```
A:\ax_install.msi OPTION="-s -x"
```

2. Use Appeon command-line option together with MSI command-line option to silently install the Active X control and also the MSI installer. With this command, the whole Active X package will be installed silently.

```
A:\ax_install.msi /quiet /qn OPTION="-s"
```

7.1.3 Appeon Server open interfaces

7.1.3.1 Overview

Appeon Server open interfaces give users the opportunity to manage services provided by Appeon Server through PowerBuilder code. In Appeon 2.8 or earlier versions, you can only use the Appeon Server AEM system to manage Appeon-deployed Web applications. Beginning with Appeon 3.0, you can easily manage Appeon-deployed Web applications in PowerBuilder code.

7.1.3.2 Appeon Server open interfaces

There are five open interfaces provided by Appeon Server. You can write code for calling the interface in the PowerBuilder application, but the code only takes effect in the Appeon-deployed application.

7.1.3.3 getSessionCount

Description

With getSessionCount method, you can get the following three types of information.

The total number of active sessions opened for the specified application in the specified Appeon Server.

The total number of active sessions in a specified Appeon Server.

The total number of sessions opened for the specified application in an Appeon Server cluster. To get the number of sessions in an Appeon Server cluster, you need to first configure the cluster in AEM.

Syntax

proxyobject.getSessionCount (String *serverName*, String *appName*)

Argument	Description
<i>proxyobject</i>	The instance of the proxy object generated for the component OpenInterface.
<i>serverName</i>	The name of the Appeon Server that the sessions are created in.
<i>appName</i>	The name of the application that is deployed to the Appeon Server for which you want to know how many sessions are opened.

Return value

Integer. Returns the number of active sessions opened for the specified application in the specified Appeon Server if it succeeds and one of the following negative values if an error occurs.

- 1: function fails to execute
- 2: the specified Appeon Server is not found
- 3: the specified application is not found

If *serverName* is NULL or empty string (""), it returns the number of the active sessions in an Appeon Server cluster. If *AppName* is NULL or empty string (""), it returns the total number of active sessions in the specified Appeon Server.

Usage

By using the `getSessionCount` method, you can easily get the total number of active sessions in a specified Appeon Server using PowerBuilder code and apply the information in other open interfaces, such as `KillSession`, to manage the sessions. For example, you can first call `getSessionCount` and then call `KillSession` in the PowerBuilder application to make the deployed application kill all sessions in Appeon Server when there are up to 100 active sessions in the server .

You can use the `AppeonGetSessionCount` function to achieve the same purpose without deploying `EAServer` components. For details on how to use the Appeon client function, refer to the *Appeon client functions* section in *Appeon Workarounds Guide*.

7.1.3.4 killAllSessions

Description

`KillAllSessions` kills all active sessions in an Appeon Server or an Appeon Server cluster and rolls back all associated transactions. To kill all sessions in an Appeon Server cluster, you need to first configure the cluster in AEM.

Syntax

proxyobject.killAllSessions (String *serverName*)

Argument	Description
<i>proxyobject</i>	The instance of the proxy object generated for the component <code>OpenInterface</code> .
<i>serverName</i>	The name of Appeon Server for which you want to kill all sessions.

Return value

Boolean. It returns true if it succeeds and false if it fails. If *serverName* is NULL or empty string (""), it kills all sessions and rolls back all associated transactions in an Appeon Server cluster.

7.1.3.5 rollbackAllTransactions

Description

`RollbackAllTransactions` rolls back all transactions in an Appeon Server or an Appeon Server cluster. To roll back all transactions in an Appeon Server cluster, you need to first configure the cluster in AEM.

Syntax

proxyobject.rollbackAllTransactions (String *serverName*)

Argument	Description
<i>proxyobject</i>	The instance of the proxy object generated for the component <code>OpenInterface</code> .
<i>serverName</i>	The name of the Appeon Server that you want to kill all sessions on.

Return value

Boolean. It returns true if it succeeds and false if it fails. If *serverName* is NULL or empty string (""), it rolls back all transactions in an Appeon Server cluster.

7.1.3.6 getAllClients

Description

getAllClients gets the IP addresses of all client machines which corresponds to the active sessions that are opened for the specified application in the specified Apeon Server.

Syntax

proxyobject.getAllClients (String serverName,String appName)

Argument	Description
<i>proxyobject</i>	The instance of the proxy object generated for the component OpenInterface.
<i>appName</i>	The name of the application that is deployed to the specified Apeon Server.
<i>serverName</i>	The name of the Apeon Server in which the active sessions are created. The value should be IP address or IP address with port number, for example, 192.0.0.123, or 192.0.0.123:8080.

Return value

String. Returns the IP addresses that correspond to the active sessions opened for the specified application in the specified Apeon Server if it succeeds and one of the following negative values if an error occurs.

- 1: function fails to execute
- 2: the specified Apeon Server is not found
- 3: the specified application is not found

If appName is NULL or empty string (""), the methods will apply to all applications in the specified Apeon Server. If serverName is NULL or an empty string (""), GetAllClients returns all the corresponding IP addresses for the specified application in all servers in an Apeon Server luster. Note you need to configure the cluster in AEM first.

7.1.3.7 getAllSessions

Description

getAllSessions returns the detail information of active sessions with XML format, which are opened for the specific application in the specific Apeon Server.

Syntax

proxyobject.getAllSessions (String serverName,String appName)

Argument	Description
<i>proxyobject</i>	The instance of the proxy object generated for the component OpenInterface.
<i>appName</i>	The name of the application that is deployed to the Apeon Server.
<i>serverName</i>	The name of the Apeon Server that the sessions are created in.

Return value

XML string. Method execution results are recorded in the *error* node. Two parameters, *code* and *text*, are contained in the *error* node. The *code* parameter shows the execution result and the *text* parameter records the detail error information. If the execution result is not 0, only the *error* node is listed in the XML.

Values for the *code* parameter:

- 0: function succeeds in execution.
- 1: function fails to execute
- 2: the specified Apeon Server is not found
- 3: the specified application is not found

If *appName* is NULL or an empty string (""), *GetAllSessions* returns the detail information of active sessions opened for all applications in the specified Apeon Server. If *serverName* is NULL or an empty string (""), *GetAllSessions* returns the detail information of active sessions in an Apeon Server cluster. Note you need to configure the cluster in AEM first.

7.1.3.8 Applying Apeon Server open interfaces in Apeon-deployed applications

There are two steps that should be performed when applying Apeon Server open interfaces in Apeon-deployed Web applications:

1. Call the Apeon Server open interface in the PowerBuilder application.
2. Deploy the PowerBuilder application to Apeon Server the same way you would deploy a normal PowerBuilder application.

7.1.4 Calling Apeon Server open interfaces in EAServer 5.x

Apeon Server open interfaces (methods) are encapsulated in a standard EAServer component named *OpenInterface* that is included in the *ASInterface* package. You can call the open interfaces in PowerBuilder scripts the same way you would call any EAServer component interface.

To invoke an open interface (method) of the *OpenInterface* component, perform the following steps in the PowerBuilder application:

Step 1 - Connect to the EAServer where Apeon Server is installed.

```
Long ll_rc
myconnect = create connection
myconnect.driver = "jaguar"
myconnect.location = "192.0.2.39:9100"
myconnect.application = "ASInterface"
myconnect.userID = "jagadmin"
myconnect.password = ""
ll_rc = myconnect.ConnectToServer()
IF ll_rc <> 0 THEN
//MessageBox("Connection failed", ll_rc)
END IF
```

Step 2 - Generate a proxy object for the component *OpenInterface*.

Please refer to the *Generating EAServer proxy objects | building an EAServer client* section in PowerBuilder Help for details.

Step 3 - Create an instance of the component *OpenInterface* by using the *CreateInstance* method.

```
int rc
rc = myconnect.CreateInstance(mycomp,
```

```
"ASInterface/OpenInterface")
  IF IsValid(mycomp) = FALSE THEN
    MessageBox('','create instance failed')
  End if
```

Step 4 - Invoke one or more interfaces in the component.

```
mycomp.getSessionCount( "192.0.2.39", "testApbInterface")
```

7.1.5 Calling Apeon Server open interfaces in EAServer 6.1, WebLogic, Websphere

Apeon Server open interfaces (methods) in EAServer 6.1, EAServer 6.1, WebLogic and WebSphere are encapsulated in a standard EJB component named OpenInterface that is included in the ASInterface package. To invoke the open interfaces in PowerBuilder, you need to follow the step instructions of how to call Apeon EJBObject. Details refer to *Apeon EJB Solution* in Apeon Workarounds Guide.

Following is general step guide to invoke the open interface in PowerBuilder.

Step 1 - Load Apeon Workaround PBL to the application.

Step 2 - Deploy Apeon Bridge

Step 3 - Connect to the EAServer where Apeon Server and Apeon Bridge are installed through Apeon EJBObject.

Example

```
string ls_prop[5]
string ls_serurl,ls_msg
ls_serurl = "http://192.0.3.35:8000/appeonbridge/Dispatch"
ls_prop[1]= "applicationA"
ls_prop[2]=
"javax.naming.Context.INITIAL_CONTEXT_FACTORY='com.sybase.ejb.
InitialContextFactory'"
ls_prop[3]=
"javax.naming.Context.PROVIDER_URL='iiop://192.0.1.249:2000'"
ls_prop[4]= "username=admin@system"
ls_prop[5]= "password=appeon249"
ls_msg = io_ejb.connectserver(ls_serurl, ls_prop)
if ls_msg <> "" then
MessageBox("Connect Application Server error!", ls_msg)
return
end if
```

Step 4- Create an instance of the EJB component OpenInterface.

Example

```
String ls_jndi, ls_home, ls_method,ls_msg
ls_jndi =
"ejb.components.appeonserver.appeonserver.OpenInterfaceRemoteH
ome"
```

```
ls_msg = io_ejb.createremoteinstance(ls_jndi, ls_home,
"create", ref il_bean1)
if ls_msg <> "" then
MessageBox("create remote instance failed!", ls_msg)
return
end if
```

Step 5 - Invoke one or more open interfaces in the component.

Example

```
string ls_msg
int c_val
string retval01
c_val = -1
io_ejb.regstring("")
io_ejb.regstring("")
ls_msg = io_ejb.invokeretstring (il_bean1,
"getAllSessions",true, ref retval01)
if ls_msg = "" then
messagebox('',"retval=" + string(retval01))
else
messagebox("","retval=" + ls_msg)
end If
```

7.1.6 Application differences

User interface and operation difference

- There may be slight differences in color and size between the Web UI and the PowerBuilder UI due to minor mismatching of the conversion algorithm for converting PowerBuilder measurements to Web measurements.
- When the Internet Explorer is refreshed, the Web application will restart.
- Using the middle mouse key to operate the Web application is unsupported.

Design-time property settings difference

- Appeon creates the resulting Web application based on design-time property settings of the application obtained from PowerBuilder. Therefore, even if the PowerBuilder application does not work properly at run-time, the Appeon-deployed Web application may still run properly. For instance, in a particular build, the PowerBuilder Edit Mask DataWindow column style did not run properly in the PBVM, but the Appeon for PowerBuilder Edit Mask DataWindow column style ran successfully because the Appeon framework for Web applications works properly.

Web applications and Internet Explorer

- While a Web application is running and the TextSize in the Internet Explorer is reset, the Web application will not change its font size accordingly.
- If an application is an SDI application, the Internet Explorer browser will not close at the same time a user closes the SDI window.
- All Apeon system error messages appear in the Internet Explorer status bar rather than in a dialog box.

Printing

- Keyboard Navigation: tab order and arrow keys are partially supported.

Section 508 compliance

All standard controls supported by Web applications are section 508 compliant but with the following noted issue on keyboard focus:

- When an Apeon Web application runs in Internet Explorer, the keyboard focus, by default, is on the Web application. You can press Alt+Tab to shift the focus to Internet Explorer, and then press Tab to shift the focus back to the application.
- You can press Alt to access the menu of the application (or Internet Explorer) when the keyboard focus is on the application (or Internet Explorer).

7.2 DataWindow enhancements and differences

7.2.1 Apeon DataWindow menu

Apeon DataWindow menu client functions

Since Apeon 3.0 on Windows, Apeon DataWindow Menu cannot be enabled through AEM (Apeon Enterprise Manager). You can enable the Apeon DataWindow Menu by calling two Apeon client functions that are defined in *apeon_workarounds_ax.pbl*:

ApeonPopupMenu function

Argument	Description
<i>adw_dw</i>	The DataWindow control on which you want to pop up the Apeon DataWindow menu.
<i>nx</i>	The instance from the left edge of the DataWindow control
<i>ny</i>	The instance from the up edge of the DataWindow control.

Return value Usage User customized RMB menus should have a higher priority than Apeon customized menus.

The ApeonPopupMenu function has a higher priority than the ApeonPopupMenuOn function.

Defining your RMB menu in RButtonDown event is not recommended because this will cause the confusion in the system. To work around this, define your RMB menu in the RButtonUp event.

ApeonPopupMenuOn function

Argument	Description
<i>adw_dw</i>	The DataWindow control on which you want to show the Appeon DataWindow menu.
<i>ab_show</i>	The Boolean value that decides whether to show Appeon DataWindow menu. True - Enables the display of the Appeon DataWindow menu. False - Disables the display of the Appeon DataWindow menu.

Description

Pops up Appeon DataWindow menu in a specified window when you right click the mouse button.

Syntax

AppeonPopupMenuOn (datawindow *adw_dw*, Boolean *ab_show*)

Return value

None.

Usage

User customized RMB menus should have a higher priority than Appeon customized menus.

The AppeonPopupMenu function has a higher priority than the AppeonPopupMenuOn function.

Defining your RMB menu in RButtonDown event is not recommended because this will cause the confusion of the system. To work around this, define your RMB menu in the RButtonUp event.

How to invoke an Appeon DataWindow Menu

There are two ways to invoke an Appeon DataWindow menu:

1. Execute the AppeonPopupMenu function for a specified DataWindow.
2. After the AppeonPopupMenuOn function is executed for a specified DataWindow, right click on a DataWindow (the area clicked should not be a DataWindow field that has focus).

Note: If the area clicked is a DataWindow field that has focus, the DataWindow menu that pops up is the same as it would be in PowerBuilder, and contains the following items: Undo, Cut, Copy, Past, Delete and Select All. This is different from Pure-JavaScript deployment, where even if the area clicked is a DataWindow field that has focus, the Appeon DataWindow Menu pops up.

Appeon DataWindow Menu Items for different types of DataWindow

Appeon DataWindow Menu Items for HTML DataWindows

Appeon DataWindow Menu Items	Functionality
Find	Finds a record with user-specified text.
Find next	Finds the next record with the specified text.
Find previous	Finds the previous record with the specified text.

Sort	Displays a sort dialog for the user to specify sorting criteria and sorts the DataWindow.
Filter	Displays a filter dialog for the user to specify filtering criteria and filters the DataWindow records. This is done on the client side.
Print...	<p>Prints the content in the DataWindow with a Client printer.</p> <p>At the click of the Print button, a Print Setup dialog pops up with the names of all the Client printers listed for the printing job.</p> <p>Using PDFPrinter on the Client for printing is not recommended. If the default printer on the Client is PDFPrinter, the printed result is saved to the "\\My Documents\Appeon Printer" folder on the Client. By default, the PDF file that is generated is only named with the .pdf extension and will have no name. If a previously printed .pdf document resides in the "Appeon Printer" folder then it will be overwritten the next time a document is printed to the folder. If the previously printed .pdf file is open, printing of the new document will fail.</p>
Save As File...	Saves the DataWindow data to a file in the specified format (TEXT, HTMLTable, WMF, EXCEL, EXCEL5 or XML).

7.2.2 DataWindow printing

Appeon provides two types of DataWindow printing to print DataWindows.

Client printing & server printing

Client printing means printing with a printer at the Client. A printing command is directly sent to a printer at the Client, and has nothing to do with Appeon Server. The default printer can be or not be the PDFPrinter provided in the Appeon installation program.

Server printing means printing with a PDFPrinter at Appeon Server. It only works with Appeon Server that is installed to EAServer on Windows. Server printing must be executed with the PDFPrinter provided in the Appeon installation program. The PDFPrinter must be of the same version as the Appeon Server that it is installed to. Server printing requires a transaction object for getting the DataWindow data from the database, and it always prints the data that is specified in the Where clause of the current Table.Select property.

DataWindow printing

To print the DataWindows at the Client, you can choose any of the following methods. There are slight differences between the two methods.

1. Using the Print item in the Appeon DataWindow Menu.
At the click of the Print... button in the Appeon DataWindow, a Print Setup dialog pops up with the names of all the Client printers listed. The user can select one of the printers for printing the DataWindow. For more information, refer to [Appeon DataWindow menu](#).
2. Executing the DataWindow Print function.
This is the same as in PowerBuilder applications: when the DataWindow Print function is executed in the Web application, the DataWindow is printed on the Client's default printer.

Server printing(EAServer only) can be performed using any of the following three methods.

- Call the `AppeonPrint2PDF` function
For more information on the usage of this function, refer to the *Appeon client functions* section in *Appeon Workarounds Guide*.
- Using the distributed `DataWindow` technique.
 - 1) Copy the `DataWindow` into a `DataStore` on EAServer using `GetFullState/SetFullState`
 - 2) On the server, call `SaveAs` with the relevant file type. Save the file to a Web-accessible location and return the URL of this location to the client
 - 3) Open the URL from the client with `HyperlinkToURL`

7.2.3 DataWindow user operation differences

User interface differences

- In some cases, text in the `DataWindow` will appear truncated or UI elements may not be fully visible in the Window. This is because the PowerBuilder units used to size the UI of the application, such as the `DataWindow` rows and columns, the Window object, etc, cannot be converted to the Web perfectly.
- The number of data records displayed per page in deployed `DataWindows` may be different from that in PowerBuilder.

DataWindow modify/update

- When you click a date field in a `DataWindow`, the format of the date is changed to format `yyyy-mm-dd`, regardless of what the original display format is.
- If a cell in a `DataWindow` contains a value that is composed of only spaces and no other characters, the update to the database will fail (the value is updated to the database as an empty string rather than blank spaces). The retrieving functionality still works correctly and is not affected. Workaround: If a cell must be empty, make sure its value is an empty string, not a string composed of only one or more spaces. This issue affects Sybase ASE, Oracle, and Microsoft SQL Server. It does not affect Sybase ASA.
- In PowerBuilder, if the user drags the scroll bar while a `DataWindow` field is being edited, `AcceptText` will not be executed for the field. In a Web `DataWindow`, `AcceptText` is executed for the field.

More behavioral differences

- Motion effect of GIF files is unsupported.

7.3 Object/control enhancements and differences

7.3.1 Object/control enhancements

MultiLineEdit

- In the Web `MultiLineEdit` control, you can drag and drop the selected text.
- In the Web `MultiLineEdit` control, you can select characters by mouse dragging and right-click on the selected characters to display the popup menu to cut/copy/paste them.

SingleLineEdit

- In the Web SingleLineEdit control, you can select characters by mouse dragging and right-click on the selected characters to display the popup menu to cut/copy/paste them.

7.3.2 Object/control user operation differences

Controls

- In a container such as a UserObject, DataWindow, or Window, if the Enable property of a control in the container is set to FALSE, a corresponding event of the container will be triggered in PowerBuilder when the user clicks the control. However, no event is triggered for the container.
- Controls in the Web application get focus when SetFocus is triggered, while controls in the PowerBuilder application get focus when the LoseFocus event of the control (which is to lose focus) is triggered. For example, when focus is moved from Control A to Control B due to the user operation, on Web, Control B gets focus when the SetFocus event of Control B is triggered, while in PowerBuilder, Control B gets focus when the LoseFocus event of Control A is triggered. This does not affect the user operation, but may affect the execution result of script dependent on the focus.

DropDownListBox, DropdownPictureListBox, ListBox or PictureListBox

- The size of the pictures displayed in a PictureListBox will always be 16px * 16px. It will not adjust according to the actual height and width available as it does in PowerBuilder.

Motion effect

- Motion effect of GIF files is unsupported.

8 System Objects and Controls

8.1 Controls

8.1.1 Supported controls

Important Requirements

- Supported controls can be dynamically created (e.g. using the CREATE statement).
- Creating a new control in a child object by copying and pasting an existing control in the PowerBuilder painter is unsupported if the existing control is inherited from the parent object.

Supported

The following list includes the controls Apeon supports:

CheckBox	CommandButton	DatePicker	DataWindow
DropDownListBox	DropDownPictureList Box	EditMask	GroupBox
HProgressBar	HScrollBar	HTrackBar	Line
ListBox	ListView	MonthCalendar	MultiLineEdit
OLEControl	OLECustomControl	Oval	Picture
PictureButton	PictureHyperLink	PictureListBox	RadioButton
Rectangle	RichTextEdit	RoundRectangle	SingleLineEdit
StaticHyperLink	StaticText	Tab	TreeView
VProgressBar	VScrollBar	VTrackBar	

For information on properties, events and functions of supported controls, refer to the remaining sections in this chapter.

For DataWindow controls, please refer to the [DataWindow](#) section.

Unsupported

It is unsupported to change the skin of system standard controls on the Web.

8.1.1.1 CheckBox control

8.1.1.1.1 Properties for CheckBox control

Property	Support Level	Example Code
Automatic	Supported	cbx_1.Automatic = TRUE
BackColor	Supported	cbx_1.BackColor = ll_value cbx_1.BackColor = RGB(192,192,192)
BorderStyle	Supported	cbx_1.BorderStyle = StyleLowered! Note: StyleRaised! is unsupported.
BringToTop	Supported	cbx_1.BringToTop = TRUE
Checked	Supported	cbx_1.Checked = TRUE cbx_1.Checked = lb_value
DragAuto	Supported	cbx_1.DragAuto = TRUE

DragIcon	Supported	cbx_1.DragIcon = 'c:\archive\arrow.ico'
Enabled	Supported	cbx_1.Enabled=TRUE cbx_1.Enabled = lb_value
FaceName	Supported	cbx_1.FaceName = "Arial" cbx_1.FaceName = ls_value
FontCharSet	Supported	cbx_1.FontCharSet = ANSI!
FontFamily	Supported	cbx_1.FontFamily = Roman!
FontPitch	Supported	cbx_1.FontPitch = Fixed!
Height	Supported	cbx_1.Height=889 cbx_1.Height = li_value
Italic	Supported	cbx_1.Italic=TRUE cbx_1.Italic = lb_value
LeftText	Supported	cbx_1.LeftText = TRUE
Pointer	Supported	cbx_1.Pointer='Cross!' cbx_1.Pointer='d:\archive\IBEAM.CUR'
RightToLeft	Supported	cbx_1.RightToLeft = TRUE
TabOrder	Supported	cbx_1.TabOrder = 10
Tag	Supported	cbx_1.Tag = ls_value
Text	Supported	cbx_1.Text = ls_value
TextColor	Supported	cbx_1.TextColor = long(88995) cbx_1.TextColor = RGB(0,0,255)
TextSize	Supported	cbx_1.TextSize = 15 cbx_1.TextSize = li_value
ThirdState	Supported	cbx_1.ThreeState = TRUE cbx_1.ThirdState = TRUE
ThreeState	Supported	cbx_1.ThreeState = TRUE cbx_1.ThirdState = TRUE
Underline	Supported	cbx_1.Underline = TRUE cbx_1.Underline = lb_value
Visible	Supported	cbx_1.Visible =TRUE cbx_1.Visible = lb_value
Weight	Supported	cbx_1.Weight = 700 cbx_1.Weight = li_value In both PowerBuilder and Apeon Web applications, a weight value smaller or equal to 550 indicates a normal weight, and a weight value larger than 550 indicates a bold weight.
Width	Supported	cbx_1.Width=899 cbx_1.Width = li_value
X	Supported	cbx_1.X = 100 cbx_1.X = li_value
Y	Supported	cbx_1.Y = 500 cbx_1.Y = li_value
ClassDefinition	Unsupported	None

8.1.1.1.2 Events for CheckBox control

Event	Support Level	Example Code
Clicked	Supported	Clicked()
Constructor	Supported	Constructor()
Destructor	Supported	Destructor()
DragDrop	Supported	DragDrop(source)
DragEnter	Supported	DragEnter(source)
DragLeave	Supported	DragLeave(source)
DragWithin	Supported	DragWithin(source)
GetFocus	Supported	GetFocus()
LoseFocus	Supported	LoseFocus()
RButtonDown	Supported	RbuttonDown(flags, xpos, ypos)
Help	Unsupported	None
Other	Unsupported	None

8.1.1.1.3 Functions for CheckBox control

Function	Support Level	Example Code
ClassName	Supported	ls_return = cbx_1.ClassName()
Drag	Supported	cbx_1.Drag(Begin!)
GetParent	Supported	lobj_parent = cbx_1.GetParent()
Hide	Supported	li_return = cbx_1.Hide()
Move	Supported	cbx_1.Move(100,100] li_return = cbx_1.Move(li_x,li_y)
PointerX	Supported	li_return = cbx_1.PointerX()
PointerY	Supported	li_return = cbx_1.PointerY()
PostEvent	Supported	cbx_1.PostEvent(RButtonDown!) lb_return = cbx_1.PostEvent(event)
Resize	Supported	cbx_1.Resize(100,20) li_return = cbx_1.Resize(li_x, li_y)
SetFocus	Supported	cbx_1.SetFocus() li_return = cbx_1.SetFocus() Unsupported: SetFocus(cbx_1)
SetPosition	Supported	li_return = cbx_1.SetPosition(position)
SetRedraw	Supported	li_return = cbx_1.SetRedraw(TRUE)
Show	Supported	li_return = cbx_1.Show()
TriggerEvent	Supported	cbx_1.TriggerEvent(RButtonDown!)
TypeOf	Supported	if cbx_1.TypeOf() = Checkbox! Then ls_return = "Checkbox!" else ls_return = "Not a Checkbox" end if
GetContextService	Unsupported	None
Print	Unsupported	None

8.1.1.2 CommandButton control

8.1.1.2.1 Properties for CommandButton control

Property	Support Level	Example Code
BringToTop	Supported	cb_1.BringToTop = TRUE
Cancel	Supported	cb_1.Cancel = TRUE
Default	Supported	cb_1.Default = TRUE
DragAuto	Supported	cb_1.DragAuto = TRUE
DragIcon	Supported	cb_1.DragIcon = 'c:\archive\arrow.ico'
Enabled	Supported	cb_1.Enabled = lb_value
FaceName	Supported	cb_1.FaceName = "Arial" cb_1.FaceName = "Courier"
Height	Supported	cb_1.Height = li_value
FontCharSet	Supported	cb_1.FontCharSet = ANSI!
FontFamily	Supported	cb_1.FontFamily = Roman!
FontPitch	Supported	cb_1.FontPitch = Fixed!
Italic	Supported	cb_1.Italic = lb_value
Pointer	Supported	cb_1.Pointer = 'Size!' cb_1.Pointer = 'd:\archive\IBeam.BMP.cur'
TabOrder	Supported	cb_1.TabOrder = 20
Tag	Supported	cb_1.Tag = ls_value
Text	Supported	cb_1.Text = ls_value
TextSize	Supported	cb_1.TextSize = li_value
Underline	Supported	cb_1.Underline = TRUE cb_1.Underline = lb_value
Visible	Supported	cb_1.Visible = TRUE cb_1.Visible = lb_value
Weight	Supported	cb_1.Weight = 700 cb_1.Weight = li_value
Width	Supported	cb_1.Width = 750 cb_1.Width = li_value
X	Supported	cb_1.X = 280 cb_1.X = li_value
Y	Supported	cb_1.Y = 1280 cb_1.Y = li_value
ClassDefinition	Unsupported	None

8.1.1.2.2 Events for CommandButton control

Event	Support Level	Example Code
Clicked	Supported	Clicked()
Constructor	Supported	Constructor()
Destructor	Supported	Destructor()
DragDrop	Supported	DragDrop(source)
DragEnter	Supported	DragEnter(source)
DragLeave	Supported	DragLeave(source)

DragWithin	Supported	DragWithin(source)
GetFocus	Supported	GetFocus()
LoseFocus	Supported	LoseFocus()
RButtonDown	Supported	RButtonDown(flags, xpos, ypos)
Help	Unsupported	None
Other	Unsupported	None

8.1.1.2.3 Functions for CommandButton control

Function	Support Level	Example Code
ClassName	Supported	ls_return = cb_1.ClassName()
Drag	Supported	cb_1.Drag(Begin!)
GetParent	Supported	PowerObject lpo_return lpo_return = cb_1.GetParent()
Hide	Supported	li_return = cb_1.Hide()
Move	Supported	cb_1.Move(200,800) li_return = cb_1.Move(li_x,li_y)
PointerX	Supported	li_return = cb_1.PointerX()
PointerY	Supported	li_return = cb_1.PointerY()
PostEvent	Supported	cb_1.PostEvent(Clicked!)
Resize	Supported	cb_1.Resize(200,800) li_return = cb_1.Resize(li_x, li_y)
SetFocus	Supported	li_return = cb_1.SetFocus() Unsupported: SetFocus(cb_1)
SetPosition	Supported	ll_return = cb_1.SetPosition(ToTop!)
SetRedraw	Supported	li_return = cb_1.SetRedraw(TRUE)
Show	Supported	li_return = cb_1.Show()
TriggerEvent	Supported	li_return = cb_1.TriggerEvent(Clicked!)
TypeOf	Supported	if cb_1.TypeOf() = commandbutton! Then ls_return = "commandbutton!" end if
GetContextService	Unsupported	None
Print	Unsupported	None

8.1.1.3 DropDownListBox control

Important Requirements

In PowerBuilder, if a DropDownListBox has no item, an empty row will display in the ListBox portion when the user clicks the down arrow. However, on the Web application, no empty row will display.

8.1.1.3.1 Properties for DropDownListBox control

Property	Support Level	Example Code
Accelerator	Supported	ddlb_1.Accelerator=67

AllowEdit	Supported	ddlb_1.AllowEdit = TRUE ddlb_1.AllowEdit = lb_value Note: The pbm_keydown event is unsupported if the AllowEdit property is set to true.
AutoHScroll	Supported	ddlb_1.AutoHScroll = TRUE
BringToTop	Supported	ddlb_1.BringToTop = TRUE
BackColor	Supported	ddlb_1.BackColor = RGB(0, 0, 255) ddlb_1.BackColor = ll_value
Border	Supported	ddlb_1.Border = TRUE ddlb_1.Border = lb_value
BorderStyle	Supported	ddlb_1.BorderStyle = StyleBox! ddlb_1.BorderStyle = StyleLowered! Note: StyleRaised! is a unsupported value.
DragAuto	Supported	ddlb_1.DragAuto = TRUE
DragIcon	Supported	ddlb_1.DragIcon = 'c:\archive\arrow.ico'
Enabled	Supported	ddlb_1.Enabled=TRUE ddlb_1.Enabled = lb_value
FaceName	Supported	ddlb_1.FaceName = "Arial" ddlb_1.FaceName = ls_value
FontCharSet	Supported	ddlb_1.FontCharSet = ANSI!
FontFamily	Supported	ddlb_1.FontFamily = Roman!
FontPitch	Supported	ddlb_1.FontPitch = Fixed!
Height	Supported	ddlb_1.Height = 750 ddlb_1.Height = li_value
HScrollBar	Supported	ddlb_1.HScrollBar = TRUE ddlb_1.HScrollBar = lb_value
Italic	Supported	ddlb_1.Italic=TRUE ddlb_1.Italic = lb_value
Item[]	Supported	ls_value = ddbl_1.Item[1]
Limit	Supported	ddlb_1.Limit = 256 ddlb_1.Limit = li_value
Pointer	Supported	ddlb_1.Pointer = 'Size!' ddlb_1.Pointer ='d:\archive\IBEAM.BMP.cur'
RightToLeft	Supported	ddlb_1.RightToLeft = TRUE
ShowList	Supported	ddlb_1.ShowList = TRUE
Sorted	Supported	ddlb_1.Sorted = TRUE ddlb_1.Sorted = lb_value
TabOrder	Supported	ddlb_1.TabOrder = 30
Tag	Supported	ddlb_1.Tag = ls_value
Text	Supported	ddlb_1.Text = ls_value
TextColor	Supported	ddlb_1.TextColor = ll_value ddlb_1.TextColor = RGB(192,192,192)
TextSize	Supported	ddlb_1.TextSize = li_value
Underline	Supported	ddlb_1.Underline= TRUE ddlb_1.Underline = lb_value
Visible	Supported	ddlb_1.Visible = TRUE

		ddlb_1.Visible = lb_value
VScrollBar	Supported	ddlb_1.VScrollBar = TRUE ddlb_1.VScrollBar = lb_value
Weight	Supported	ddlb_1.Weight = 700 ddlb_1.Weight = li_value
Width	Supported	ddlb_1.Width = li_value
X	Supported	ddlb_1.X = li_value
Y	Supported	ddlb_1.Y = li_value
ClassDefinition	Unsupported	None

8.1.1.3.2 Events for DropDownListBox control

Event	Support Level	Example Code
Constructor	Supported	Constructor()
Destructor	Supported	Destructor()
DoubleClicked	Supported	DoubleClicked()
DragDrop	Supported	DragDrop(source)
DragEnter	Supported	DragEnter(source)
DragLeave	Supported	DragLeave(source)
DragWithin	Supported	DragWithin(source)
GetFocus	Supported	GetFocus()
LoseFocus	Supported	LoseFocus()
Modified	Supported	Modified() Note: it is unsupported to trigger the event by clicking the Enter key.
RButtonDown	Supported	RButtonDown(flags, xpos, ypos)
SelectionChanged	Supported	SelectionChanged(index) If the return value of the event is greater than 0, the Modified event will be triggered. Otherwise, continue processing.
Help	Unsupported	None.
Other	Unsupported	None

8.1.1.3.3 Functions for DropDownListBox control

Function	Support Level	Example Code
AddItem	Supported	li_return = ddlb_1.AddItem(item)
ClassName	Supported	ls_return = ddlb_1.ClassName()
Clear	Supported	li_return = ddlb_1.Clear()
Copy	Supported	li_return = ddlb_1.Copy()
Cut	Supported	li_return = ddlb_1.Cut()
DeleteItem	Supported	li_return = ddlb_1.DeleteItem(index)
DirList	Supported	ddlb_1.DirList("C:\EMPLOYEE*.TXT", 0)
DirSelect	Supported	lb_dir = ddlb_1.DirSelect(ls_dirname)
Drag	Supported	ddlb_1.Drag(Cancel!)
FindItem	Supported	li_return = ddlb_1.FindItem('hello', 4)

		li_return = ddlb_1.FindItem(text, index)
GetParent	Supported	PowerObject lpo_return lpo_return = ddlb_1.GetParent()
Hide	Supported	li_return = ddlb_1.Hide()
InsertItem	Supported	ddlb_1.InsertItem('hello',3) li_return = ddlb_1.InsertItem(item, index)
Move	Supported	ddlb_1.Move(200,400) li_return = ddlb_1.Move(li_x,li_y)
Paste	Supported	li_return = ddlb_1.Paste()
PointerX	Supported	li_return = ddlb_1.PointerX()
PointerY	Supported	li_return = ddlb_1.PointerY()
PostEvent	Supported	ddlb_1.PostEvent(Clicked!)
ReplaceText	Supported	ddlb_1.ReplaceText("60 Days")
Reset	Supported	li_return = ddlb_1.Reset()
Resize	Supported	ddlb_1.Resize(200,800) li_return = ddlb_1.Resize(li_x, li_y)
SelectItem	Supported	ddlb_1.SelectItem ('hello',3) li_return = ddlb_1.SelectItem(item, index)
SelectedLength	Supported	li_returnvalue = ddlb_1.SelectedLength()
SelectedStart	Supported	li_returnvalue = ddlb_1.SelectedStart()
SelectedText	Supported	li_returnvalue = ddlb_1.SelectedText()
SelectText	Supported	li_returnvalue = ddlb_1.SelectText(start, length)
SetFocus	Supported	li_return = ddlb_1.SetFocus() Unsupported: SetFocus(ddlb_1)
SetPosition	Supported	ddlb_1.SetPosition(ToTop!) li_return = ddlb_1.SetPosition(position)
SetRedraw	Supported	li_return = ddlb_1.SetRedraw(FALSE)
Show	Supported	li_return = ddlb_1.Show()
Text	Supported	ls_return = ddlb_1.Text(index)
TotalItems	Supported	li_return = ddlb_1.TotalItems()
TriggerEvent	Supported	ddlb_1.TriggerEvent(Constructor!) li_return = ddlb_1.TriggerEvent(event)
TypeOf	Supported	if ddlb_1.typeof() = DropDownListBox! messagebox("show", " DropDownListBox!") end if
GetContextService	Unsupported	None
Position	Unsupported	None
Print	Unsupported	None

8.1.1.4 DatePicker control

Difference

Left mouse clicking on the DatePicker control will trigger the Clicked, GetFocused events in sequence.

8.1.1.4.1 Properties for DatePicker control

Property	Support Level	Example Code
----------	---------------	--------------

Accelerator	Supported	dp_1.Accelerator=67
AllowEdit	Supported	dp_1.AllowEdit = TRUE
Border	Supported	dp_1.Border = TRUE
BorderStyle	Supported	dp_1.BorderStyle = StyleBox! dp_1.BorderStyle = StyleLowered! dp_1.BorderStyle = StyleRaised! Note: StyleShadowBox! is unsupported.
BringToTop	Supported	
CalendarBackColor	Supported	dp_1.CalendarBackColor = RGB(255, 255, 0)
CalendarFontCharset	Supported	dp_1.CalendarFontCharset = ANSI!
CalendarFontFamily	Supported	dp_1.CalendarFontFamily = Roman!
CalendarFontName	Supported	dp_1.CalendarFontName = ls_value
CalendarFontPitch	Supported	dp_1.CalendarFontPitch = Fixed!
CalendarFontWeight	Supported	dp_1.CalendarFontWeight = li_value
CalendarItalic	Supported	dp_1.CalendarItalic = lb_value
CalendarTextColor	Supported	dp_1.CalendarTextColor = RGB(0,0,255)
CalendarTextSize	Supported	dp_1.CalendarTextSize = li_value
CalendarTitleBackColor	Supported	dp_1.CalendarTitleBackColor = RGB(128, 255, 128)
CalendarTrailingTextColor	Supported	dp_1.CalendarTrailingTextColor = RGB(128, 255, 128)
CalendarUnderline	Supported	dp_1.CalendarUnderline = lb_value
CustomFormat	Supported	dp_1.CustomFormat = ls_value
DateValue	Supported	dp_1.DateValue = ld_value
DragAuto	Supported	dp_1.DragAuto = TRUE Notes: 1. On the Web, dragging a DatePicker control will not trigger the DropDown event if the DragAuto property is set to True, which means there is no Month Calendar displayed on the Web when you Drag the DatePicker with setting the DragAuto property to True. 2. On the Web, if the DragAuto is set to True, clicking, dragging, and releasing the DatePicker control will trigger the following event in sequence: DragEnter, DragWithin, DragLeave.
DragIcon	Supported	dp_1.DragIcon = 'arrow.ico'
DropDownRight	Supported	dp_1.DropDownRight = true
Enabled	Supported	dp_1.Enabled = TRUE
FaceName	Supported	dp_1.FaceName = "Arial"
FirstDayOfWeek	Supported	dp_1.FirstDayOfWeek = Monday!
FontCharSet	Supported	dp_1.FontCharSet = ANSI!
FontFamily	Supported	dp_1.FontFamily = Roman!
FontPitch	Supported	dp_1.FontPitch = Fixed!
FontWeight	Supported	dp_1.FontWeight = 700
Format	Supported	dp_1.Format = dtfShortDate!
Height	Supported	dp_1.Height = 750 dp_1.Height = li_value

Italic	Supported	dp_1.Italic=TRUE dp_1.Italic = lb_value
MaxDate	Supported	dp_1.MaxDate = Date("2007/06/30")
MinDate	Supported	dp_1.MinDate = Date("2006/09/01")
Pointer	Supported	dp_1.Pointer = 'Size!'
ShowUpDown	Supported	dp_1.ShowUpDown = true
TabOrder	Supported	dp_1.TabOrder = 30
Tag	Supported	dp_1.Tag = ls_value
Text	Supported	dp_1.Text = ls_value
TextSize	Supported	dp_1.TextSize = li_value
TimeValue	Supported	lt_value = dp_1.TimeValue
TodayCircle	Supported	dp_1.TodayCircle = false
TodaySection	Supported	dp_1.TodaySection = false
Underline	Supported	dp_1.Underline = TRUE
Value	Supported	dp_1.Value = DateTime(Date("2005/07/01"),Time("12:00:00"))
Visible	Supported	dp_1.Visible = True
WeekNumbers	Supported	dp_1.WeekNumbers = true
Width	Supported	dp_1.Width = li_value Note: 1. If the Width value is set to -32768, obtaining the width property returns -32768 on the Web, however it returns 0 in PowerBuilder. 2. If the Width value is set to 32768, obtaining the width property returns 0 on the Web, however it returns -32768 in PowerBuilder.
X	Supported	dp_1.X = li_value Note: 1. If the X value is set to -32768, obtaining the X property returns -32768 on the Web, however it returns 0 in PowerBuilder. 2. If the Width value is set to 32768, obtaining the width property returns 0 on the Web, however it returns -32768 in PowerBuilder.
Y	Supported	dp_1.Y = li_value Note: 1. If the Y value is set to -32768, obtaining the Y property returns -32768 on the Web, however it returns 0 in PowerBuilder. 2. If the Width value is set to 32768, obtaining the width property returns 0 on the Web, however it returns -32768 in PowerBuilder.
AccessibleDescription	Unsupported	
AccessibleName	Unsupported	
AccessibleRole	Unsupported	
ClassDefinition	Unsupported	
RightToLeft	Unsupported	

8.1.1.4.2 Events for DatePicker control

Event	Support Level	Example Code
Clicked	Supported	Clicked()
CloseUp	Supported	CloseUp()
Constructor	Supported	Constructor()
Destructor	Supported	Destructor()
DoubleClicked	Supported	Doubleclicked(flags,xpos,ypos)
DragDrop	Supported	DragDrop()
DragEnter	Supported	DragEnter(source)
DragLeave	Supported	DragLeave(source)
DragWithin	Supported	DragWithin(source)
DropDown	Supported	DropDown()
LoseFocus	Supported	GetFocus()
GetFocus	Supported	GetFocus()
PreCreateWindow	Supported	Precreatewindow(style,styleex,other)
RButtonDown	Supported	RButtonDown(flags, xpos, ypos)
ValueChanged	Supported	Valuechanged(flag,dtm)
Help	Unsupported	
UserString	Unsupported	
Other	Unsupported	

8.1.1.4.3 Functions for DatePicker control

Function	Support Level	Example Code
ClassName	Supported	ls_value = dp_1.ClassName()
Drag	Supported	dp_1.Drag (dragmode) Note: If you use Drag(Begin!), single mouse clicking will trigger DragEnter and DragDrop event in sequence on the Web. This is different from the event sequences in

		PowerBuilder.
GetCalendar	Supported	dp_1.getcalendar()
GetParent	Supported	PowerObject lpo_returnvalue lpo_returnvalue = dp_1.GetParent()
GetText	Supported	ls_text = dp_1.GetText()
GetToday	Supported	currentdate = dp_1.GetToday()
GetValue	Supported	li_value = dp_1.GetValue(dt)
Hide	Supported	li_return = dp_1.Hide()
Move	Supported	li_return = dp_1.Move(li_x,li_y)
PointerX	Supported	li_value = dp_1.PointerX()
PointerY	Supported	li_value = dp_1.PointerY()
PostEvent	Supported	dp_1.PostEvent(Constructor!)
Resize	Supported	li_return = dp_1.Resize(li_width,li_height)
SetFocus	Supported	_return = dp_1.SetFocus()
SetPosition	Supported	li_return = dp_1.SetPosition(position)
SetRedraw	Supported	li_return= dp_1.SetRedraw(FALSE)
SetToday	Supported	li_return = dp_1.SetToday(currentdate)
SetValue	Supported	dp_1.SetValue(dt)
Show	Supported	li_return = dp_1.Show()
TriggerEvent	Supported	dp_1.TriggerEvent(Constructor!)
TypeOf	Supported	if dp_1.TypeOf() = DatePicker! Then ls_return = DatePicker! else ls_return = 'invalid' end if
GetContextService	Unsupported	
Print	Unsupported	

8.1.1.5 DropDownPictureListBox control

8.1.1.5.1 Properties for DropDownPictureListBox control

Property	Support Level	Example Code
Accelerator	Supported	ddplb_1.Accelerator = 67
AllowEdit	Supported	ddplb_1.AllowEdit = TRUE ddplb_1.AllowEdit = lb_value Note: The pbm_keydown event is unsupported if the AllowEdit property is set to true.
AutoHScroll	Supported	ddplb_1.AutoHScroll = TRUE
BackColor	Supported	ddplb_1.BackColor = ll_value ddplb_1.BackColor = RGB (0,0,255)

BringToTop	Supported	ddplb_1.BringToTop = TRUE
Border	Supported	ddplb_1.Border = TRUE
		ddplb_1.Border = lb_value
BorderStyle	Supported	ddplb_1.BorderStyle = StyleBox!
		ddplb_1.BorderStyle = StyleLowered!
		ddplb_1.BorderStyle = StyleRaised!
		Note: StyleShadowBox! is unsupported.
DragAuto	Supported	ddplb_1.DragAuto = TRUE
DragIcon	Supported	ddplb_1.DragIcon = 'c:\archive\arrow.ico'
Enabled	Supported	ddplb_1.Enabled = TRUE
		ddplb_1.Enabled = lb_value
FaceName	Supported	ddplb_1.Facename = ls_value
FontCharSet	Supported	ddplb_1.FontCharSet = ANSI!
FontFamily	Supported	ddplb_1.FontFamily = Roman!
FontPitch	Supported	ddplb_1.FontPitch = Fixed!
Height	Supported	ddplb_1.Height = li_value
HScrollBar	Supported	ddplb_1.Hscrollbar = lb_value
Italic	Supported	ddplb_1.Italic = lb_value
Item[]	Supported	String a[] = ddplb_1.Item[]
ItemPictureIndex[]	Supported	ddplb_1.ItemPictureIndex[] = li_value[]
Limit	Supported	ddplb_1.Limit = 256
		ddplb_1.Limit = li_value
PictureHeight	Supported	ddplb_1.PictureHeight = 16
PictureWidth	Supported	ddplb_1.PictureWidth = 16
PictureName[]	Supported	ls_value = ddplb_1.PictureName[1]
		<p>Assigning values to part of an array and using the default values for the other part is not supported. For example:</p> <pre>picturename[1] = "aaa.bmp" picturename[3] = "bbb.bmp" var pic1 = picutrename[1]; var pic2 = picturename[3];</pre> <p>A supported example is given as follows:</p> <pre>picturename[1] = "aaa.bmp" picturename[2] = "bbb.bmp" picturename[3] = "ccc.bmp" var pic1 = picutrename[1]; var pic2 = picturename[2]; var pic3 = picturename[3];</pre>
Pointer	Supported	ddplb_1.Pointer = "Beam!"
		dbplb_1.Pointer = 'd:\archive\IBEAM.BMP.cur'
RightToLeft	Supported	dbplb_1.RightToLeft = TRUE
ShowList	Supported	ddplb_1.ShowList = TRUE
Sorted	Supported	ddplb_1.Sorted = lb_value
TabOrder	Supported	ddplb_1.TabOrder = 40

Tag	Supported	ddplb_1.Tag = ls_value
Text	Supported	ddplb_1.Text = ls_value
TextColor	Supported	ddplb_1.Textcolor = ll_value ddplb_1.Textcolor = RGB(192,192,192)
TextSize	Supported	ddplb_1.TextSize = li_value
Underline	Supported	ddplb_1.Underline = lb_value
Visible	Supported	ddplb_1.Visible = lb_value
VScrollBar	Supported	ddplb_1.VscrollBar = lb_value
Weight	Supported	ddplb_1.Weight = li_value
Width	Supported	ddplb_1.Width = li_value
X	Supported	ddplb_1.X = li_value
Y	Supported	ddplb_1.Y = li_value
ClassDefinition	Unsupported	None
PictureMaskColor	Unsupported	None

8.1.1.5.2 Events for DropDownPictureListBox control

Event	Support Level	Example Code
Constructor	Supported	Constructor()
Destructor	Supported	Destructor()
DoubleClicked	Supported	DoubleClicked()
DragDrop	Supported	DragDrop(source)
DragEnter	Supported	DragEnter(source)
DragLeave	Supported	DragLeave(source)
DragWithin	Supported	DragWithin(source)
GetFocus	Supported	GetFocus()
LoseFocus	Supported	LoseFocus()
Modified	Supported	Modified()
RButtonDown	Supported	RButtonDown(flags, xpos, ypos)
SelectionChanged	Supported	SelectionChanged(index) //If the return value is greater than 0, the Modified event will be triggered. Otherwise, continue processing.
Help	Unsupported	None
Other	Unsupported	None

8.1.1.5.3 Functions for DropDownPictureListBox control

Function	Support Level	Example Code
AddItem	Supported	li_return = ddplb_1.AddItem(ls_name, li_pic)
AddPicture	Supported	li_return = ddplb_1.AddPicture(picturename)
ClassName	Supported	ls_return = ddplb_1.ClassName()
Clear	Supported	ls_return = ddplb_1.Clear()
Copy	Supported	ls_return = ddplb_1.Copy()

Cut	Supported	ls_return = ddplb_1.Cut()
DeleteItem	Supported	li_return = ddplb_1.DeleteItem(index)
DeletePicture	Supported	li_return = ddplb_1.DeletePicture(index)
DeletePictures	Supported	li_return = ddplb_1.DeletePictures()
DirList	Supported	ddplb_1.DirList("C:\EMPLOYEE*.TXT", 0) ddplb_1.DirList("C:\EMPLOYEE*.DOC", 1, st_path)
DirSelect	Supported	ddplb_1.DirSelect(ls_filename)
Drag	Supported	ddplb_1.Drag(End!)
FindItem	Supported	li_return = ddplb_1.FindItem(text, index)
GetParent	Supported	PowerObject lpo_return lpo_return = ddplb_1.GetParent()
Hide	Supported	li_return = ddplb_1.Hide()
InsertItem	Supported	ddplb_1.InsertItem("Run Application", 5)
Move	Supported	li_return = ddplb_1.Move(li_x, li_y)
Paste	Supported	li_return = ddplb_1.Paste()
PointerX	Supported	li_return = ddplb_1.PointerX()
PointerY	Supported	li_return = ddplb_1.PointerY()
PostEvent	Supported	ddplb_1.PostEvent(Clicked!) lb_return = ddplb_1.PostEvent(event)
Reset	Supported	li_return = ddplb_1.Reset()
Resize	Supported	li_return = ddplb_1.Resize(li_width, li_height)
SelectItem	Supported	li_return = ddplb_1.SelectItem(item, index)
SelectedLength	Supported	li_returnvalue = ddplb_1.SelectedLength()
SelectedStart	Supported	li_returnvalue = ddplb_1.SelectedStart()
SelectedText	Supported	li_returnvalue = ddplb_1.SelectedText()
SelectText	Supported	li_returnvalue = ddplb_1.SelectText(start, length)
SetFocus	Supported	li_return = ddplb_1.SetFocus() Unsupported: SetFocus(ddplb_1)
SetPosition	Supported	li_return = ddplb_1.SetPosition(position)
SetRedraw	Supported	li_return = ddplb_1.SetRedraw(FALSE)
Show	Supported	li_return = ddplb_1.Show()
Text	Supported	ls_return = ddplb_1.Text(index)
TotalItems	Supported	li_return = ddplb_1.TotalItems()
TriggerEvent	Supported	li_return = ddplb_1.TriggerEvent(event)
TypeOf	Supported	if ddplb_1.TypeOf() = DropDownPictureListBox! Then ls_return = 'DropDownPictureListBox!' else ls_return = 'isvalid' end if
GetContextService	Unsupported	None
Position	Unsupported	None
Print	Unsupported	None

8.1.1.6 EditMask control

8.1.1.6.1 Properties for EditMask control

Property	Support Level	Example Code
----------	---------------	--------------

Accelerator	Supported	em_1.Accelerator = 67
Alignment	Supported	em_1.Alignment = Center!
AutoSkip	Supported	em_1.AutoSkip = TRUE
BackColor	Supported	em_1.BackColor = ll_value em_1.BackColor = RGB(192,192,192)
Border	Supported	em_1.Border = TRUE em_1.Border = lb_value
BorderStyle	Supported	em_1.BorderStyle = StyleLowered! em_1.BorderStyle = StyleRaised! em_1.BorderStyle = StyleBox! Note: StyleShadowBox! is unsupported.
BringToTop	Supported	em_1.BringToTop = TRUE
DisplayData	Supported	None
DisplayOnly	Supported	em_1.DisplayOnly = TRUE em_1.DisplayOnly = lb_value
DragAuto	Supported	em_1.DragAuto = TRUE
DragIcon	Supported	em_1.DragIcon = 'c:\archive\arrow.ico'
Enabled	Supported	em_1.Enabled = TRUE em_1.Enabled = lb_value
FaceName	Supported	em_1.FaceName = ls_value
FontCharSet	Supported	em_1.FontCharSet = ANSI!
FontFamily	Supported	em_1.FontFamily = Roman!
FontPitch	Supported	em_1.FontPitch = Fixed!
Height	Supported	em_1.Height = li_value
HideSelection	Supported	em_1.HideSelection = FALSE
Increment	Supported	em_1.Increment = 5.0 em_1.Increment = ld_value
Italic	Supported	em_1.Italic = lb_value
Limit	Supported	em_1.Limit = 50 em_1.Limit = li_value Note: the Limit property cannot be dynamically changed.
Mask	Supported	em_1.MaskDataType =StringMask em_1.Mask = "^^^-!!!" Note: The Mask property for EditMask can only be set in the PowerBuilder Painter.
MaskDataType	Supported	em_1.MaskDataType = StringMask! em_1.MaskDataType = DateTimeMask! em_1.MaskDataType = NumericMask! Note: this property can be set only in painter.
MinMax	Supported	em_1.MinMax = ("100 ~ 10000")
Pointer	Supported	em_1.Pointer = 'AppStarting!' em_1.Pointer = 'd:\archive\IBeam.BMP.cur'
RightToLeft	Supported	em_1.RightToLeft = TRUE
Spin	Supported	em_1.Spin = TRUE em_1.Spin = FALSE
TabOrder	Supported	em_1.TabOrder = 50

Tag	Supported	em_1.Tag = ls_value
Text	Supported	em_1.Text = ls_value
TextColor	Supported	em_1.TextColor = ll_value
		em_1.TextColor = RGB(192,192,192)
TextCase	Supported	em_1.TextCase = AnyCase!
		em_1.TextCase = Lower!
		em_1.TextCase = Upper!
TextSize	Supported	em_1.TextSize = li_value
Underline	Supported	em_1.Underline = TRUE
		em_1.Underline = lb_value
UseCodeTable	Supported	em_1.UseCodeTable = TRUE
Visible	Supported	em_1.Visible = lb_value
Weight	Supported	em_1.Weight = li_value
Width	Supported	em_1.Width = li_value
X	Supported	em_1.X = li_value
Y	Supported	em_1.Y = li_value
AutoHScroll	Unsupported	None
AutoVScroll	Unsupported	None
ClassDefinition	Unsupported	None
HScrollBar	Unsupported	None
IgnoreDefaultButton	Unsupported	None
TabStop[]	Unsupported	None
VScrollBar	Unsupported	None

8.1.1.6.2 Events for EditMask control

Event	Support Level	Example Code
Constructor	Supported	Constructor()
Destructor	Supported	Destructor()
DragDrop	Supported	DragDrop(source)
DragEnter	Supported	DragEnter(source)
DragLeave	Supported	DragLeave(source)
DragWithin	Supported	DragWithin(source)
GetFocus	Supported	GetFocus()
LoseFocus	Supported	LoseFocus()
Modified	Supported	Modified()
RButtonDown	Supported	RButtonDown(flags, xpos, ypos)
Help	Unsupported	None
Other	Unsupported	None

8.1.1.6.3 Functions for EditMask control

Function	Support Level	Example Code
ClassName	Supported	ls_returnvalue = em_1.ClassName()
Clear	Supported	li_returnvalue = em_1.Clear()

Copy	Supported	li_returnvalue = em_1.Copy()
Cut	Supported	li_returnvalue = em_1.Cut()
Drag	Supported	em_1.Drag(Begin!)
GetData	Supported	li_returnvalue = em_1.GetData(datavariablename)
GetParent	Supported	PowerObject lpo_returnvalue lpo_returnvalue = em_1.GetParent()
Hide	Supported	li_returnvalue = em_1.Hide()
LineCount	Supported	li_returnvalue = em_1.LineCount()
LineLength	Supported	li_returnvalue = em_1.LineLength()
Move	Supported	li_returnvalue = em_1.Move(li_xpos, li_ypos)
Paste	Supported	li_returnvalue = em_1.Paste()
PointerX	Supported	li_returnvalue = em_1.PointerX()
PointerY	Supported	li_returnvalue = em_1.PointerY()
PostEvent	Supported	lb_returnvalue = em_1.PostEvent(event)
ReplaceText	Supported	li_returnvalue = em_1.ReplaceText(text)
Resize	Supported	li_returnvalue = em_1.Resize(li_width, li_height)
SelectedLength	Supported	li_returnvalue = em_1.SelectedLength() If the control is off screen, the execution of the function may have an incorrect return result.
SelectedLine	Supported	integer li_SL li_SL = em_1.SelectedLine()
SelectedStart	Supported	li_returnvalue = em_1.SelectedStart() If the control is off screen, the execution of the function may have an incorrect return result.
SelectedText	Supported	ls_returnvalue = em_1.SelectedText() If the control is off screen, the execution of the function may have an incorrect return result.
SelectText	Supported	ls_returnvalue = em_1.SelectText(start, length) If the control is off screen, the execution of the function may have an incorrect return result.
SetFocus	Supported	li_returnvalue = em_1.SetFocus() Unsupported: SetFocus(em_test)
SetMask	Supported	li_returnvalue = em_1.SetMask(maskdatatype, mask)
SetPosition	Supported	li_returnvalue = em_1.SetPosition(position)
SetRedraw	Supported	li_returnvalue = em_1.SetRedraw(FALSE)
Show	Supported	li_returnvalue = em_1.Show()
TextLine	Supported	li_returnvalue = em_1.TextLine()
TriggerEvent	Supported	li_returnvalue = em_1.TriggerEvent()

TypeOf	Supported	If em_1.TypeOf() = editmask! Then ls_returnvalue = 'editmask!' else ls_returnvalue = 'invalid!' End if
CanUndo	Unsupported	None
GetContextService	Unsupported	None
Position	Unsupported	None
Print	Unsupported	None
Scroll	Unsupported	None
Undo	Unsupported	None

8.1.1.7 GroupBox control

8.1.1.7.1 Properties for GroupBox control

Property	Support Level	Example Code
BackColor	Supported	gb_1.BackColor = ll_value
BorderStyle	Supported	gb_1.Border = TRUE gb_1.BorderStyle = StyleLowered!
BringToTop	Supported	gb_1.BringToTop = TRUE
DragAuto	Supported	gb_1.DragAuto = TRUE
DragIcon	Supported	gb_1.DragIcon = 'c:\archive\arrow.ico'
Enabled	Supported	gb_1.Enabled = lb_value
FaceName	Supported	gb_1.FaceName = ls_value
FontCharSet	Supported	gb_1.FontCharSet = ANSI!
FontFamily	Supported	gb_1.FontFamily = Roman!
FontPitch	Supported	gb_1.FontPitch = Fixed!
Height	Supported	gb_1.Height = li_value
Italic	Supported	gb_1.Italic = lb_value
Pointer	Supported	gb_1.Pointer = 'Beam!' gb_1.Pointer = 'd:\archive\IBEAM.BMP.cur'
RightToLeft	Supported	gb_1.RightToLeft = TRUE
TabOrder	Supported	gb_1.TabOrder = 50
Tag	Supported	gb_1.Tag = ls_value
Text	Supported	gb_1.Text = ls_value
TextColor	Supported	gb_1.TextColor = ll_value gb_1.TextColor = RGB(192,192,192)
TextSize	Supported	gb_1.TextSize = li_value
Underline	Supported	gb_1.Underline = lb_value
Visible	Supported	gb_1.Visible = lb_value
Weight	Supported	gb_1.Weight = li_value
Width	Supported	gb_1.Width = li_value
X	Supported	gb_1.X = li_value
Y	Supported	gb_1.Y = li_value
ClassDefinition	Unsupported	None

8.1.1.7.2 Events for GroupBox control

Event	Support Level	Example Code
Constructor	Supported	Constructor()
Destructor	Supported	Destructor()
Help	Unsupported	None
Other	Unsupported	None

8.1.1.7.3 Functions for GroupBox control

Function	Support Level	Example Code
ClassName	Supported	ls_returnvalue = gb_1.ClassName()
Drag	Supported	gb_1.drap(Cancel!)
GetParent	Supported	PowerObject lpo_returnvalue lpo_returnvalue = gb_1.GetParent()
Hide	Supported	li_returnvalue = gb_1.Hide()
Move	Supported	gb_1.Move(200,500) li_returnvalue = gb_1.Move(li_xpos, li_ypos)
PointerX	Supported	li_returnvalue = gb_1.PointerX()
PointerY	Supported	li_returnvalue = gb_1.PointerY()
PostEvent	Supported	lb_returnvalue = gb_1.PostEvent(event)
Resize	Supported	li_returnvalue = gb_1.Resize(li_width,li_height)
SetFocus	Supported	li_returnvalue = gb_1.SetFocus() Unsupported: SetFocus(gb_text)
SetPosition	Supported	li_returnvalue = gb_1.SetPosition(position)
SetRedraw	Supported	li_value = gb_1.SetRedraw(TRUE)
Show	Supported	li_returnvalue = gb_1.Show()
TriggerEvent	Supported	li_returnvalue = gb_1.TriggerEvent(event)
TypeOf	Supported	if gb_1.TypeOf() = GroupBox! Then ls_returnvalue = "GroupBox!" else ls_returnvalue = "invalid" end if
GetContextService	Unsupported	None
Print	Unsupported	None

8.1.1.8 HProgressBar control**8.1.1.8.1 Properties for HProgressBar control**

Property	Support Level	Example Code
BringToTop	Supported	hpb_1.BringToTop = TRUE
DragAuto	Supported	hpb_1.DragAuto = TRUE
DragIcon	Supported	hpb_1.DragIcon = 'Question!'

Height	Supported	hpb_1.Height = li_value
MaxPosition	Supported	hpb_1.MaxPosition = 120
MinPosition	Supported	hpb_1.MinPosition = 20
Pointer	Supported	hpb_1.Pointer = 'Beam!' hpb_1.Pointer = 'd:\archive\IBEAM.BMP.cur'
Position	Supported	hpb_1.Position = 50
SetStep	Supported	hpb_1.setstep = 20
SmoothScroll	Supported	hpb_1.SmoothScroll = True
TabOrder	Supported	hpb_1.TabOrder = 20
Tag	Supported	hpb_1.Tag = "HProgressBar control"
Visible	Supported	hpb_1.Visible = TRUE
Width	Supported	hpb_1.Width = 700
X	Supported	hpb_1.X = 280
Y	Supported	hpb_1.Y = 1280
ClassDefinition	Unsupported	None

8.1.1.8.2 Events for HProgressBar control

Event	Support Level	Example Code
Clicked	Supported	Clicked()
Constructor	Supported	Constructor()
Destructor	Supported	Destructor()
DoubleClicked	Supported	DoubleClicked()
DragDrop	Supported	DragDrop(source)
DragEnter	Supported	DragEnter(source)
DragLeave	Supported	DragLeave(source)
DragWithin	Supported	DragWithin(source)
GetFocus	Supported	GetFocus()
LoseFocus	Supported	LoseFocus()
RightClicked	Supported	RightClicked()
Help	Unsupported	None
Other	Unsupported	None

8.1.1.8.3 Functions for HProgressBar control

Function	Support Level	Example Code
ClassName	Supported	ls_classnm = hpb_1.classname()
Drag	Supported	li_return = hpb_1.drag(Begin!)
GetParent	Supported	lpo_return = hpb_1.GetParent()
Hide	Supported	li_return = hpb_1.Hide()
Move	Supported	li_return = hpb_1.Move(li_x,li_y)
OffsetPos	Supported	li_return = hpb_1.OffsetPos(10)
PointerX	Supported	li_return = hpb_1.PointerX()
PointerY	Supported	li_return = hpb_1.PointerY()
PostEvent	Supported	hpb_1.PostEvent(Clicked!)

Resize	Supported	li_return = hpb_1.Resize(li_x, li_y)
SetFocus	Supported	li_return = hpb_1.SetFocus()
SetPosition	Supported	li_return = hpb_1.SetPosition(ToTop!)
SetRange	Supported	li_return = hpb_1.SetRange (1, 10)
SetRedraw	Supported	li_return = hpb_1.SetRedraw(FALSE)
Show	Supported	li_return = hpb_1.Show()
StepIt	Supported	li_return = hpb_1.StepIt()
TriggerEvent	Supported	li_return = hpb_1.TriggerEvent(Clicked!)
TypeOf	Supported	type_obj = hpb_1.typeof()
GetContextService	Unsupported	None
Print	Unsupported	None

8.1.1.9 HScrollBar control

8.1.1.9.1 Properties for HScrollBar control

Property	Support Level	Example Code
BringToTop	Supported	hsb_1.BringToTop = TRUE
DragAuto	Supported	hsb_1.DragAuto = TRUE
DragIcon	Supported	hsb_1.DragIcon = 'Question!'
Height	Supported	hsb_1.Height = li_value
MaxPosition	Supported	hsb_1.MaxPosition = 120
MinPosition	Supported	hsb_1.MinPosition = 20
Pointer	Supported	hsb_1.Pointer = 'Beam!' hsb_1.Pointer = 'd:\archive\IBEAM.BMP.cur'
Position	Supported	hsb_1.Position = 50
StdHeight	Supported	hsb_1.StdHeight = True
TabOrder	Supported	hsb_1.TabOrder = 20
Tag	Supported	hsb_1.Tag = "HScrollBar control"
Visible	Supported	hsb_1.Visible = TRUE
Width	Supported	hsb_1.Width = 700
X	Supported	hsb_1.X = 280
Y	Supported	hsb_1.Y = 1280
ClassDefinition	Unsupported	None

8.1.1.9.2 Events for HScrollBar control

Event	Support Level	Example Code
Constructor	Supported	Constructor()
Destructor	Supported	Destructor()
DragDrop	Supported	DragDrop(source)
DragEnter	Supported	DragEnter(source)
DragLeave	Supported	DragLeave(source)
DragWithin	Supported	DragWithin(source)
GetFocus	Supported	GetFocus()
LineLeft	Supported	LineLeft()

LineRight	Supported	LineRight() Note: User defined events in the LineLeft or LineRight events for the HScrollbar control may be triggered more than once.
LoseFocus	Supported	LoseFocus()
Moved	Supported	Moved(scrollpos)
PageLeft	Supported	PageLeft()
PageRight	Supported	PageRight()
RButtonDown	Supported	RButtonDown()
Help	Unsupported	None
Other	Unsupported	None

8.1.1.9.3 Functions for HScrollbar control

Function	Support Level	Example Code
ClassName	Supported	ls_classnm = hsb_1.classname()
Drag	Supported	li_return = hsb_1.drag(Begin!)
GetParent	Supported	lpo_return = hsb_1.GetParent()
Hide	Supported	li_return = hsb_1.Hide()
Move	Supported	li_return = hsb_1.Move(li_x, li_y)
PointerX	Supported	li_return = hsb_1.PointerX()
PointerY	Supported	li_return = hsb_1.PointerY()
PostEvent	Supported	hsb_1.PostEvent(Clicked!)
Resize	Supported	li_return = hsb_1.Resize(li_x, li_y)
SetFocus	Supported	li_return = hsb_1.SetFocus()
SetPosition	Supported	li_return = hsb_1.SetPosition(ToTop!)
SetRedraw	Supported	li_return = hsb_1.SetRedraw(FALSE)
Show	Supported	li_return = hsb_1.Show()
TriggerEvent	Supported	li_return = hsb_1.TriggerEvent(Clicked!)
TypeOf	Supported	type_obj = hsb_1.typeof()
GetContextService	Unsupported	None
Print	Unsupported	None

8.1.1.10 HTrackBar control

8.1.1.10.1 Properties for HTrackBar control

Property	Support Level	Example Code
BringToTop	Supported	htb_1.BringToTop = TRUE
DragAuto	Supported	htb_1.DragAuto = TRUE
DragIcon	Supported	htb_1.DragIcon = 'Question!'
Height	Supported	htb_1.Height = li_value
LineSize	Supported	htb_1.LineSize = li_value
MaxPosition	Supported	htb_1.MaxPosition = 120
MinPosition	Supported	htb_1.MinPosition = 20
PageSize	Supported	htb_1.PageSize = li_value

Pointer	Supported	htb_1.Pointer = 'Beam!'
Position	Supported	htb_1.Position = 50
Slider	Supported	htb_1.Slider = TRUE
SliderSize	Supported	htb_1.SliderSize = li_value
TabOrder	Supported	htb_1.TabOrder = 20
Tag	Supported	htb_1.Tag = ls_tag
TickFrequency	Supported	htb_1.TickFrequency = li_value
TickMarks	Supported	htb_1.TickMarks = HTicksOnRight!
Visible	Supported	htb_1.Visible = TRUE
Width	Supported	htb_1.Width = li_value
X	Supported	htb_1.X = li_x
Y	Supported	htb_1.Y = li_y
ClassDefinition	Unsupported	None

8.1.1.10.2 Events for HTrackBar control

Event	Support Level	Example Code
Constructor	Supported	Constructor()
Destructor	Supported	Destructor()
DragDrop	Supported	DragDrop(source)
DragEnter	Supported	DragEnter(source)
DragLeave	Supported	DragLeave(source)
DragWithin	Supported	DragWithin(source)
GetFocus	Supported	GetFocus()
LineDown	Supported	LineDown()
LineUp	Supported	LineUp()
LoseFocus	Supported	LoseFocus()
Moved	Supported	Moved(scrollpos)
PageDown	Supported	PageDown()
PageUp	Supported	PageUp()
RButtonDown	Supported	RButtonDown()
Help	Unsupported	None
Other	Unsupported	None

8.1.1.10.3 Functions for HTrackBar control

Function	Support Level	Example Code
ClassName	Supported	ls_classnm = htb_1.classname()
Drag	Supported	li_return = htb_1.drag(Begin!)
GetParent	Supported	lpo_return = htb_1.GetParent()
Hide	Supported	li_return = htb_1.Hide()
Move	Supported	li_return = htb_1.Move(li_x,li_y)
PointerX	Supported	li_return = htb_1.PointerX()
PointerY	Supported	li_return = htb_1.PointerY()
PostEvent	Supported	htb_1.PostEvent(Clicked!)
Resize	Supported	li_return = htb_1.Resize(li_x, li_y)
SelectionRange	Supported	li_return = htb_1.SelectionRange(30, 70)
SetFocus	Supported	li_return = htb_1.SetFocus()
SetPosition	Supported	li_return = htb_1.SetPosition(ToTop!)

SetRedraw	Supported	li_return = htb_1.SetRedraw(FALSE)
Show	Supported	li_return = htb_1.Show()
TriggerEvent	Supported	li_return = htb_1.TriggerEvent(Clicked!)
TypeOf	Supported	type_obj = htb_1.typeof()
GetContextService	Unsupported	None
Print	Unsupported	None

8.1.1.11 Line control

8.1.1.11.1 Properties for Line control

Property	Support Level	Example Code
BeginX	Supported	ln_1.BeginX = li_value
BeginY	Supported	ln_1.BeginY = li_value
EndX	Supported	ln_1.EndX = li_value
EndY	Supported	ln_1.EndY = li_value
LineColor	Supported	ln_1.LineColor = ll_value ln_1.LineColor = RGB(192,192,192)
LineStyle	Supported	ln_1.LineStyle = Continuous!
		ln_1.LineStyle = Dash!
		ln_1.LineStyle = DashDot!
		ln_1.LineStyle = DashDotDot!
		ln_1.LineStyle = Dot!
ln_1.LineStyle = Transparent!		
LineThickness	Supported	ln_1.LineThickness = li_value
Tag	Supported	ln_1.Tag = ls_value
Visible	Supported	ln_1.Visible = TRUE
		ln_1.Visible = lb_value
ClassDefinition	Unsupported	None

8.1.1.11.2 Events for Line control

Event	Support Level	Example Code
Constructor	Supported	Constructor()
Destructor	Supported	Destructor()

8.1.1.11.3 Functions for Line control

Function	Support Level	Example Code
ClassName	Supported	ls_returnvalue = ln_1.ClassName()
GetParent	Supported	li_returnvalue = ln_1.GetParent()
Hide	Supported	li_returnvalue = ln_1.Hide()
Move	Supported	li_returnvalue = ln_1.Move(li_xpos, li_ypos)
Resize	Supported	li_returnvalue = ln_1.Resize(100, 150)
Show	Supported	li_returnvalue = ln_1.Show()
TypeOf	Supported	if ln_1.typeof() = Line! Then

		<pre> ls_returnvalue = 'Line!' else ls_returnvalue = 'Invalid' end if </pre>
GetContextService	Unsupported	None

8.1.1.12 ListBox control

8.1.1.12.1 Properties for ListBox control

Property	Support Level	Example Code
Accelerator	Supported	lb_1.Accelerator=67
BackColor	Supported	lb_1.BackColor = ll_value
		lb_1.BackColor = RGB(192,192,192)
Border	Supported	lb_1.Border = lb_value
BringToTop	Supported	lb_1.BringToTop = TRUE
BorderStyle	Supported	lb_1.BorderStyle = StyleLowered!
		lb_1.BorderStyle = StyleRaised!
		lb_1.BorderStyle = StyleBox!
		Note: StyleShadowBox! is unsupported.
DisableNoScroll	Supported	lb_1.DisableNoScroll = FALSE
		lb_1.DisableNoScroll = lb_value
DragAuto	Supported	lb_1.DragAuto = TRUE
DragIcon	Supported	lb_1.DragIcon = 'c:\examples\arrow.ico'
		lb_1.DragIcon = 'Question!'
Enabled	Supported	lb_1.Enabled = lb_value
ExtendedSelect	Supported	lb_1.ExtendedSelect = lb_value
FaceName	Supported	lb_1.FaceName = ls_value
FontCharSet	Supported	lb_1.FontCharSet = ANSI!
FontFamily	Supported	lb_1.FontFamily = Roman!
FontPitch	Supported	lb_1.FontPitch = Fixed!
Height	Supported	lb_1.Height = li_value
HScrollBar	Supported	lb_1.HScrollBar = TRUE
		lb_1.HScrollBar = lb_value
Italic	Supported	lb_1.Italic = lb_value
Item[]	Supported	ls_value = lb_1.Item[1]
MultiSelect	Supported	lb_1.MultiSelect = lb_value
Pointer	Supported	lb_1.Pointer = 'SizeNS!'
		lb_1.Pointer='d:\archive\IBEAM.BMP.cur'
RightToLeft	Supported	lb_1.RightToLeft = TRUE
Sorted	Supported	lb_1.Sorted = TRUE
TabOrder	Supported	lb_1.TabOrder = 60
Tag	Supported	lb_1.Tag = ls_value
TextColor	Supported	lb_1.TextColor = ll_value
		lb_1.TextColor = RGB(192,192,192)
TextSize	Supported	lb_1.TextSize = li_value
Underline	Supported	lb_1.Underline = lb_value
Visible	Supported	lb_1.Visible = lb_value
VScrollBar	Supported	lb_1.VScrollBar = lb_value
Weight	Supported	lb_1.Weight = li_value

Width	Supported	lb_1.Width = li_value
X	Supported	lb_1.X = li_value
Y	Supported	lb_1.Y = li_value
ClassDefinition	Unsupported	None
TabStop[]	Unsupported	None

8.1.1.12.2 Events for ListBox control

Event	Support Level	Example Code
Constructor	Supported	Constructor()
Destructor	Supported	Destructor()
DoubleClicked	Supported	DoubleClicked()
DragDrop	Supported	DragDrop(source)
DragEnter	Supported	DragEnter(source)
DragLeave	Supported	DragLeave(source)
DragWithin	Supported	DragWithin(source)
GetFocus	Supported	GetFocus()
LoseFocus	Supported	LoseFocus()
RButtonDown	Supported	RButtonDown(flags, xpos, ypos)
SelectionChanged	Supported	SelectionChanged(index)
Help	Unsupported	None
Other	Unsupported	None

8.1.1.12.3 Functions for ListBox control

Function	Support Level	Example Code
AddItem	Supported	li_returnvalue = lb_1.AddItem(item)
ClassName	Supported	ls_returnvalue = lb_1.ClassName()
DeleteItem	Supported	li_returnvalue = lb_1.DeleteItem(index)
DirList	Supported	lb_1.DirList("C:\EMPLOYEE*.TXT", 0)
DirSelect	Supported	lb_1.DirSelect(ls_filename)
Drag	Supported	lb_1.Drag(End!)
Find Item	Supported	li_returnvalue = lb_1.FindItem(text, index)
GetParent	Supported	PowerObject lpo_returnvalue lpo_returnvalue = lb_1.GetParent()
Hide	Supported	li_returnvalue = lb_1.Hide()
InsertItem	Supported	li_returnvalue = lb_1.InsertItem(ls_x, li_y)
Move	Supported	li_returnvalue = string(lb_1.Move(li_x, li_y))
PointerX	Supported	li_returnvalue = lb_1.PointerX()
PointerY	Supported	li_returnvalue = lb_1.PointerY()
PostEvent	Supported	lb_returnvalue = string(lb_1.PostEvent(Clicked!))
Reset	Supported	li_returnvalue = lb_1.Reset()
Resize	Supported	li_returnvalue = lb_1.Resize(li_x, li_y)
SelectedIndex	Supported	li_returnvalue = lb_1.SelectedIndex()
SelectedItem	Supported	ls_returnvalue = lb_1.SelectedItem()
SelectItem	Supported	li_returnvalue = lb_1.SelectItem (item, index)
SetFocus	Supported	li_returnvalue = lb_1.SetFocus() Unsupported: SetFocus(lb_test)

SetPosition	Supported	li_returnvalue = lb_1.SetPosition(position)
SetRedraw	Supported	li_value = lb_1.SetRedraw(TRUE)
SetState	Supported	li_returnvalue = lb_1.SetState(index, lb_y)
SetTop	Supported	li_returnvalue = lb_1.SetTop(index)
Show	Supported	li_returnvalue = lb_1.Show()
State	Supported	li_returnvalue = lb_1.State(index)
Text	Supported	ls_returnvalue = lb_1.Text (index)
Top	Supported	li_returnvalue = lb_1.Top()
TotalItems	Supported	li_returnvalue = lb_1.TotalItems()
TotalSelected	Supported	li_returnvalue = lb_1.TotalSelected()
TriggerEvent	Supported	li_returnvalue = lb_1.TriggerEvent(event)
TypeOf	Supported	if lb_1.typeof() = listbox! Then ls_returnvalue ='listbox!' else ls_returnvalue ='invalid' end if
GetContextService	Unsupported	None
Print	Unsupported	None

8.1.1.13 ListView control

Important Requirements

In the ListView control, selecting multiple items at one time is unsupported.

8.1.1.13.1 Properties for ListView control

Property	Support Level	Example Code
Accelerator	Supported	lv_1.Accelerator=67
AutoArrange	Supported	lv_1.AutoArrange = TRUE
BackColor	Supported	lv_1.BackColor = ll_value lv_1.BackColor = RGB(192,192,192)
Border	Supported	lv_1.Border = lb_value
BorderStyle	Supported	lv_1.BorderStyle = StyleBox! lv_1.BorderStyle = StyleLowered! lv_1.BorderStyle = StyleRaised! Note: StyleShadowBox! is not supported.
BringToTop	Supported	lv_1.BringToTop = TRUE
ButtonHeader	Supported	lv_1.ButtonHeader = TRUE
CheckBoxes	Supported	lv_1.CheckBoxes = TRUE
DeleteItems	Supported	lv_1.DeleteItems = lb_value
DragAuto	Supported	lv_1.DragAuto = TRUE
DragIcon	Supported	lv_1.DragIcon = 'c:\archive\arrow.ico'
EditLabels	Supported	lv_1.EditLabels = lb_value
Enabled	Supported	lv_1.Enabled = lb_value
ExtendedSelect	Supported	lv_1.ExtendedSelect = TRUE
FaceName	Supported	lv_1.FaceName =ls_value
FixedLocations	Supported	lv_1.FixedLocations = FALSE
FontCharSet	Supported	lv_1.FontCharSet = ANSI!
FontFamily	Supported	lv_1.FontFamily = Roman!
FontPitch	Supported	lv_1.FontPitch = Fixed!

FullRowSelect	Supported	lv_1.FullRowSelect = TRUE
GridLines	Supported	lv_1.GridLines = TRUE
HeaderDragDrop	Supported	lv_1.HeaderDragDrop = TRUE
Height	Supported	lv_1.Height = li_value
HideSelection	Supported	lv_1.HideSelection = FALSE
Italic	Supported	lv_1.Italic = lb_value
Item[]	Supported	lv_1.item[intx] = ls_value
ItemPictureIndex[]	Supported	lv_1.ItemPictureIndex [li_x] = ls_value
LabelWrap	Supported	lv_1.LabelWrap = TRUE
LargePictureHeight	Supported	lv_1.LargePictureHeight = li_value
		Note: This value cannot be set to 0 or negative.
LargePictureMaskColor	Supported	lv_1.LargePictureMaskColor = RGB(255, 255, 0)
LargePictureName[]	Supported	lv_1.LargePictureName [li_x] = ls_value
LargePictureWidth	Supported	lv_1.LargePictureWidth = li_value
		Note: This value cannot be set to 0 or negative.
OneClickActivate	Supported	lv_1.OneClickActivate = TRUE
Pointer	Supported	lv_1.Pointer = 'SizeNS!'
		lv_test.Pointer = 'd:\archive\IBEAM.BMP.cur'
Scrolling	Supported	lv_1.Scrolling = TRUE
ShowHeader	Supported	lv_1.ShowHeader = TRUE
SmallPictureHeight	Supported	lv_1.SmallPictureHeight = li_value
		Note: This value cannot be set to 0 or negative.
SmallPictureMaskColor	Supported	lv_1.SmallPictureMaskColor = RGB(255, 255, 0)
SmallPictureName[]	Supported	lv_1.SmallPictureName [li_x] = ls_value
SmallPictureWidth	Supported	lv_1.SmallPictureWidth = li_value
		Note: This value cannot be set to 0 or negative.
SortType	Supported	lv_1.SortType = Unsorted!
StatePictureMaskColor	Supported	lv_1.StatePictureMaskColor = RGB(255, 255, 0)
StatePictureName[]	Supported	lv_1.StatePictureName [li_x] = ls_value
TabOrder	Supported	lv_1.TabOrder = 70
Tag	Supported	lv_1.Tag = ls_value
TextColor	Supported	lv_1.TextColor = ll_value
		lv_1.TextColor = RGB(192,192,192)
TextSize	Supported	lv_1.TextSize = li_value
TrackSelect	Supported	lv_1.TrackSelect = TRUE
TwoClickActivate	Supported	lv_1.TwoClickActivate = TRUE
Underline	Supported	lv_1.Underline = lb_value

UnderlineCold	Supported	lv_1.UnderlineCold = TRUE
UnderlineHot	Supported	lv_1.UnderlineHot = TRUE
View	Supported	lv_1.View = ListViewLargeIcon!
Visible	Supported	lv_1.Visible = lb_value
Weight	Supported	lv_1.Weight = li_value
Width	Supported	lv_1.Width = li_value
X	Supported	lv_1.X = li_value
Y	Supported	lv_1.Y = li_value
ClassDefinition	Unsupported	None
LayoutRTL	Unsupported	None
StatePictureHeight	Unsupported	None
StatePictureWidth	Unsupported	None

8.1.1.13.2 Events for ListView control

Event	Support Level	Example Code
BeginDrag	Supported	BeginDrag(index)
BeginLabelEdit	Supported	BeginLabelEdit(index)
BeginRightDrag	Supported	BeginRightDrag(index)
Clicked	Supported	Clicked(index)
ColumnClick	Supported	ColumnClick(column)
Constructor	Supported	Constructor()
DeleteAllItems	Supported	DeleteAllItems()
DeleteItem	Supported	DeleteItem(index)
Destructor	Supported	Destructor()
DoubleClicked	Supported	DoubleClicked(index)
DragDrop	Supported	DragDrop(source,index)
DragEnter	Supported	DragEnter(source)
DragLeave	Supported	DragLeave(source)
DragWithin	Supported	DragWithin(source,index)
EndLabelEdit	Supported	EndLabelEdit(index,newlabel)
GetFocus	Supported	GetFocus()
InsertItem	Supported	InsertItem(index)
ItemActivate	Supported	ItemActivate(index, focuschanged, hasfocus, selectionchange, selected, otherchange)
ItemChanged	Supported	ItemChanged(index, focuschanged, hasfocus, selectionchange, selected, otherchange)
ItemChanging	Supported	ItemChanging(index, focuschange, hasfocus, selectionchange, selected, otherchange)
Key	Supported	Key(key,keyflags)
LoseFocus	Supported	LoseFocus()
RightClicked	Supported	RightClicked(index)
RightDoubleClicked	Supported	RightDoubleClicked(index)
Help	Unsupported	None
Other	Unsupported	None
Sort	Unsupported	None

8.1.1.13.3 Functions for ListView control

Function	Support Level	Example Code
AddColumn	Supported	li_value = lv_1.AddColumn(ls_x,Center!,li_y)
AddItem	Supported	li_value = lv_1.AddItem(ls_x,li_y)
AddLargePicture	Supported	li_value = lv_1.AddLargePicture(ls_prop)
AddSmallPicture	Supported	li_value = lv_1.AddSmallPicture(ls_prop)
AddStatePicture	Supported	li_value = lv_1.AddstatePicture(ls_prop)
Arrange	Supported	li_value = lv_1.Arrange()
ClassName	Supported	ls_value = lv_1.ClassName()
DeleteColumn	Supported	li_value = lv_1.DeleteColumn(1)
DeleteColumns	Supported	li_value = lv_1.DeleteColumns()
DeleteItem	Supported	li_value = lv_1.DeleteItem(li_x)
DeleteItems	Supported	li_value = lv_1.DeleteItems()
DeleteLargePicture	Supported	li_value = lv_1.DeleteLargePicture(li_x)
DeleteLargePictures	Supported	li_value = lv_1.DeleteLargePictures()
DeleteSmallPicture	Supported	li_value = lv_1.DeleteSmallPicture(li_x)
DeleteSmallPictures	Supported	li_value = lv_1.DeleteSmallPictures()
DeleteStatePicture	Supported	li_value = lv_1.DeleteStatePicture(li_x)
DeleteStatePictures	Supported	li_value = lv_1.DeleteStatePictures()
Drag	Supported	lv_1.Drag(Begin!)
EditLabel	Supported	li_value= lv_1.EditLabel()
FindItem	Supported	li_value = lv_1.FindItem (startindex, label, partial, wrap) li_value = lv_1.FindItem(startindex, direction, focused, selected, cuthighlighted, drophighlighted)
GetColumn	Supported	li_strValue = lv_1.GetColumn (index, label, alignment, width)
GetItem	Supported	li_value = lv_1.GetItem(li_x,lv1)
GetParent	Supported	PowerObject lpo_returnvalue lpo_returnvalue = lv_1.GetParent()
Hide	Supported	li_value = lv_1.Hide()
InsertColumn	Supported	li_value = lv_1.InsertColumn(li_x,ls_x,Center!,li_y)
InsertItem	Supported	li_value = lv_1.InsertItem(2, "Insert String")
Move	Supported	li_value = lv_1.Move(li_x,li_y)
PointerX	Supported	li_value = lv_1.PointerX()
PointerY	Supported	li_value = lv_1.PointerY()
PostEvent	Supported	lv_1.PostEvent(Clicked!)
Resize	Supported	li_value = lv_1.Resize(li_x,li_y)
SelectedIndex	Supported	li_value = lv_1.SelectedIndex()
SetColumn	Supported	li_value = lv_1.SetColumn(li_x,ls_x,Center!,li_y)
SetFocus	Supported	li_value = lv_1.SetFocus() Unsupported: SetFocus(lv_test)
SetItem	Supported	li_value = lv_1.SetItem(li_x[],li_y[],lv1)
SetPosition	Supported	li_value = lv_1.SetPosition(Behind!, lv_three)
SetRedraw	Supported	li_value = lv_1.SetRedraw(TRUE)

Show	Supported	li_value = lv_1.Show()
Sort	Supported	li_value = lv_1.Sort(Ascending!)
		li_value = lv_1.Sort(Descending!)
		li_value = lv_1.Sort(Unsorted!)
		Note: three types, ascending!, descending! and unsorted!, are supported. The sort type userdefinesort! is not supported.
TotalColumns	Supported	li_cols = lv_1.TotalColumns()
TotalItems	Supported	li_value = lv_1.TotalItems()
TriggerEvent	Supported	li_value = lv_1.TriggerEvent("Clicked")
TypeOf	Supported	if lv_1.TypeOf() = ListView! Then ls_value = "ListView!" else ls_value = "invalid" end if
SetOverlayPicture	Supported	lv_1.SetOverlayPicture(1, index)
TotalSelected	Supported	None
GetContextService	Unsupported	None
GetOrigin	Unsupported	None
Print	Unsupported	None

8.1.1.14 MultiLineEdit control

8.1.1.14.1 Properties for MultiLineEdit control

Property	Support Level	Example Code
Accelerator	Supported	mle_1.Accelerator = 67
Alignment	Supported	mle_1.Alignment = Center!
AutoHScroll	Supported	mle_1.AutoHScroll = TRUE
AutoVScroll	Supported	mle_1.AutoVscroll = TRUE
BackColor	Supported	mle_1.BackColor = ll_value
		mle_1.BackColor = RGB(192,192,192)
Border	Supported	mle_1.Border = lb_value
BorderStyle	Supported	mle_1.BorderStyle = StyleBox!
		mle_1.BorderStyle = StyleLowered!
		mle_1.BorderStyle = StyleRaised!
		Note: StyleShadowBox! is unsupported.
BringToTop	Supported	mle_1.BringToTop = TRUE
DisplayOnly	Supported	mle_1.DisplayOnly = lb_value
DragAuto	Supported	mle_1.DragAuto = TRUE
DragIcon	Supported	mle_1.DragIcon = 'c:\archive\arrow.ico'
Enabled	Supported	mle_1.Enabled = lb_value
		Note: when the user moves the pointer over a MultiLineEdit control that has set the Enabled property to false, the pointer on the Web will be different from that in the PowerBuilder application.
FaceName	Supported	mle_1.FaceName = ls_value

FontCharSet	Supported	mle_1.FontCharSet = ANSI!
FontFamily	Supported	mle_1.FontFamily = Roman!
FontPitch	Supported	mle_1.FontPitch = Fixed!
Height	Supported	mle_1.Height = li_value
HideSelection	Supported	mle_1.HideSelection = FALSE
HScrollBar	Supported	mle_1.HScrollBar = TRUE
IgnoreDefaultButton	Supported	mle_1.IgnoreDefaultButton = TRUE
Italic	Supported	mle_1.Italic = lb_value
Limit	Supported	mle_1.Limit = li_value
Pointer	Supported	mle_1.pointer= 'beam!' mle_1.pointer ='d:\archive\IBEAM.BMP.cur'
RightToLeft	Supported	mle_1.RightToLeft = TRUE
TabOrder	Supported	mle_1.TabOrder = 80
Tag	Supported	mle_1.Tag = ls_value
Text	Supported	mle_1.Text = ls_value It is unsupported to set the value of this property to null.
TextCase	Supported	mle_1.TextCase = AnyCase! mle_1.TextCase = Lower! mle_1.TextCase = Upper!
TextColor	Supported	mle_1.TextColor = ll_value
TextSize	Supported	mle_1.TextSize = li_value
Underline	Supported	mle_1.Underline = lb_value
Visible	Supported	mle_1.Visible = lb_value
VScrollBar	Supported	mle_1.VScrollBar = TRUE
Weight	Supported	mle_1.Weight = li_value
Width	Supported	mle_1.Width = li_value
X	Supported	mle_1.X = li_value
Y	Supported	mle_1.Y = li_value
ClassDefinition	Unsupported	None
TabStop[]	Unsupported	None

8.1.1.14.2 Events for MultiLineEdit control

Event	Support Level	Example Code
Constructor	Supported	Constructor()
Destructor	Supported	Destructor()
DragDrop	Supported	DragDrop(source)
DragEnter	Supported	DragEnter(source)
DragLeave	Supported	DragLeave(source)
DragWithin	Supported	DragWithin(source)
GetFocus	Supported	GetFocus()
LoseFocus	Supported	LoseFocus()
Modified	Supported	Modified() A Modified event is followed by a LoseFocus event.
RButtonDown	Supported	RButtonDown(flags, xpos, ypos)

Help	Unsupported	None
Other	Unsupported	None

8.1.1.14.3 Functions for MultiLineEdit control

Function	Support Level	Example Code
ClassName	Supported	ls_value = mle_1.ClassName()
Clear	Supported	ls_value = mle_1.Clear() Note: the function executes successfully only when the control has focus.
Copy	Supported	ls_value = mle_1.Copy()
Cut	Supported	ls_value = mle_1.Cut()
Drag	Supported	mle_1.Drag(Begin!)
GetParent	Supported	PowerObject lpo_returnvalue lpo_returnvalue = mle_1.GetParent()
Hide	Supported	ls_value = mle_1.Hide()
LineCount	Supported	li_value = mle_1.LineCount()
LineLength	Supported	li_value = mle_1.LineLength()
Move	Supported	ls_value = mle_1.Move(li_x,li_y)
Paste	Supported	ls_value = mle_1.Paste()
PointerX	Supported	ls_value = mle_1.PointerX()
PointerY	Supported	ls_value = mle_1.PointerY()
Position	Supported	mle_1.Position()
PostEvent	Supported	ls_value = mle_1.PostEvent(Constructor!)
ReplaceText	Supported	ls_strexevalue = mle_1.ReplaceText(ls_x)
Resize	Supported	ls_value = mle_1.Resize(li_x,li_y)
Scroll	Supported	mle_1.Scroll(4)
SelectedLength	Supported	ls_value = mle_1.SelectedLength()
SelectedLine	Supported	integer li_SL li_SL = mle_1.SelectedLine()
SelectedStart	Supported	ls_value = mle_1.SelectedStart()
SelectedText	Supported	ls_value = mle_1.SelectedText()
SelectText	Supported	li_value = mle_1.SelectText(li_x,li_y)
SetFocus	Supported	ls_value = mle_1.SetFocus() Unsupported: SetFocus(mle_1)
SetPosition	Supported	mle_1.SetPosition(Behind!, mle_2) ls_value = mle_1.SetPosition(ToTop!) ls_value = mle_1.SetPosition(ToBottom!)
SetRedraw	Supported	li_return= mle_1.SetRedraw(FALSE)
Show	Supported	ls_value = mle_1.Show()
TextLine	Supported	ls_value = mle_1.TextLine()
TriggerEvent	Supported	ls_value = mle_1.TriggerEvent(Constructor!)
TypeOf	Supported	if mle_1.typeof() = MultiLineEdit! Then ls_value = 'MultiLineEdit!' else ls_value = 'Invalid' end if
Undo	Supported	li_value = mle_1.Undo()
CanUndo	Unsupported	None

GetContextService	Unsupported	None
Print	Unsupported	None

8.1.1.15 MonthCalendar control

8.1.1.15.1 Properties for MonthCalendar control

Property	Support Level	Example Code
Accelerator	Supported	mc_1.Accelerator=67
AutoSize	Supported	mc_1.AutoSize = false
BackColor	Supported	mc_1.BackColor = RGB(0, 0, 255) mc_1.BackColor = ll_value
Border	Supported	mc_1.Border = TRUE mc_1.Border = lb_value
BorderStyle	Supported	mc_1.BorderStyle = StyleBox! mc_1.BorderStyle = StyleLowered! mc_1.BorderStyle = StyleRaised! Note: StyleShadowBox! is unsupported.
BringToTop	Supported	mc_1.BringToTop = TRUE
DragAuto	Supported	mc_1.DragAuto = TRUE Note: On the Web Dragging a DatePicker control will not trigger the DropDown event if the DragAuto property is set to True, which means there is no Month Calendar displayed on the Web when you Drag the DatePicker with setting the DragAuto property to True.
DragIcon	Supported	mc_1.DragIcon = 'arrow.ico'
Enabled	Supported	mc_1.Enabled=TRUE mc_1.Enabled = lb_value
FaceName	Supported	mc_1.FaceName = "Arial" mc_1.FaceName = ls_value
FirstDayOfWeek	Supported	mc_1.FirstDayOfWeek = Monday!
FontCharSet	Supported	mc_1.FontCharSet = ANSI!
FontFamily	Supported	mc_1.FontFamily = Roman!
FontPitch	Supported	mc_1.FontPitch = Fixed!
Height	Supported	mc_1.Height = 750 mc_1.Height = li_value
Italic	Supported	mc_1.Italic=TRUE mc_1.Italic = lb_value
MaxSelectCount	Supported	mc_1.MaxSelectCount = 7
MonthBackColor	Supported	mc_1.MonthBackColor = RGB(255, 255, 0)
Pointer	Supported	mc_1.Pointer = 'Size!'
ScrollRate	Supported	mc_1.ScrollRate = 3
TabOrder	Supported	mc_1.TabOrder = 30
Tag	Supported	mc_1.Tag = ls_value
TextColor	Supported	mc_1.TextColor = ll_value
TextSize	Supported	mc_1.TextSize = li_value

TitleBackColor	Supported	mc_1.TitleBackColor = RGB(128, 255, 128)
TitleTextColor	Supported	mc_1.TitleTextColor = RGB(0,0,255)
TodayCircle	Supported	mc_1.TodaySection = false
TodaySection	Supported	mc_1.TodaySection = false
TrailingTextColor	Supported	mc_1.TrailingTextColor = 134217731
Underline	Supported	mc_1.Underline = lb_value
Visible	Supported	mc_1.Visible = True
WeekNumbers	Supported	mc_1.WeekNumbers = true
Weight	Supported	mc_1.Weight = 700 mc_1.Weight = li_value
Width	Supported	mc_1.Width = li_value Note: 1. If the Width value is set to -32768, obtaining the width property returns -32768 on the Web, however it returns 0 in PowerBuilder. 2. If the Width value is set to 32768, obtaining the width property returns 0 on the Web, however it returns -32768 in PowerBuilder.
X	Supported	mc_1.X = li_value Note: 1. If the X value is set to -32768, obtaining the X property returns -32768 on the Web, however it returns 0 in PowerBuilder. 2. If the Width value is set to 32768, obtaining the width property returns 0 on the Web, however it returns -32768 in PowerBuilder.
Y	Supported	mc_1.Y = li_value Note: 1. If the Y value is set to -32768, obtaining the Y property returns -32768 on the Web, however it returns 0 in PowerBuilder. 2. If the Width value is set to 32768, obtaining the width property returns 0 on the Web, however it returns -32768 in PowerBuilder.
AccessibleDescription	Unsupported	
AccessibleName	Unsupported	
AccessibleRole	Unsupported	
ClassDefinition	Unsupported	
RightToLeft	Unsupported	

8.1.1.15.2 Events for MonthCalendar control

Event	Support Level	Example Code
Clicked	Supported	Clicked()
Constructor	Supported	Constructor()
DateChanged	Supported	DateChanged() Note: SetSelectedDate and SetSelectedRange trigger a

		DateChanged event twice on the Web.
Destructor	Supported	Destructor()
DoubleClicked	Supported	Doubleclicked(flags,xpos,ypos)
DragDrop	Supported	DragDrop(source)
DragEnter	Supported	DragEnter(source)
DragLeave	Supported	DragLeave(source)
DragWithin	Supported	DragWithin(source)
RButtonDown	Supported	RButtonDown(flags, xpos, ypos)
GetFocus	Unsupported	
LoseFocus	Unsupported	
Help	Unsupported	
UserString	Unsupported	
Other	Unsupported	

8.1.1.15.3 Functions for MonthCalendar control

Function	Support Level	Example Code
ClassName	Supported	ls_value = mc_1.ClassName()
ClearBoldDates	Supported	li_return = mc_1.ClearBoldDates()
Drag	Supported	mc_1.Drag(Begin!) Note: If you use Drag(Begin!), single mouse clicking will trigger DragEnter and DragDrop event in sequence on the Web. This is different from the event sequences in PowerBuilder.
GetDateLimits	Supported	li_return = mc_1.GetDateLimits(mindate, maxdate)
GetDisplayRange	Supported	li_return = mc_1.GetDisplayRange(startdate, enddate,PartlyDisplayed!)
GetParent	Supported	PowerObject lpo_returnvalue lpo_returnvalue = mc_1.GetParent()
GetSelectedDate	Supported	li_return = mc_1.GetSelectedDate(seldate) Notes: 1. The function returns 0 on the Web if there are several dates selected or the specified date is an invalid value. 2. The function will trigger a DateChanged event twice on the Web.

GetSelectedRange	Supported	li_return = mc_1.GetSelectedRange(startdate, enddate) Note: The function will trigger a DateChanged event twice on the Web.
GetToday	Supported	currentdate = mc_1.GetToday()
Hide	Supported	li_return = mc_1.Hide()
Move	Supported	li_return = mc_1.Move(li_x,li_y)
PointerX	Supported	li_value = mc_1.PointerX()
PointerY	Supported	li_value = mc_1.PointerY()
PostEvent	Supported	mc_1.PostEvent(Constructor!)
Resize	Supported	li_return = mc_1.Resize(li_width,li_height)
SetBoldDate	Supported	li_return = mc_1.SetBoldDate(d, true)
SetDateLimits	Supported	li_return = mc_1.SetDateLimits(mindate, maxdate)
SetFocus	Supported	li_return = mc_1.SetFocus()
SetPosition	Supported	li_return = mc_1.SetPosition(position)
SetRedraw	Supported	li_return= mc_1.SetRedraw(FALSE)
SetSelectedDate	Supported	li_return = mc_1.SetSelectedDate(seldate) Notes:If input an invalid date, the function returns 0 on the Web but -1 in PowerBuilder.
SetSelectedRange	Supported	li_return = mc_1.SetSelectedRange(startdate, enddate)
SetToday	Supported	li_return = mc_1.SetToday(currentdate)
Show	Supported	li_return = mc_1.Show()
TriggerEvent	Supported	mc_1.TriggerEvent(Constructor!)
TypeOf	Supported	if mc_1.TypeOf() = MonthCalendar! Then ls_return = 'MonthCalendar!' else ls_return = 'isvalid' end if
GetContextService	Unsupported	
Print	Unsupported	

8.1.1.16 OLEControl control

Important Requirements

The shortcut key for Ocx is unsupported.

Using parentheses when calling an OLEObject method is required.

8.1.1.16.1 Properties for OLEControl control

Properties	Support Level	Example Code
Activation	Supported	ole_1.Activation = ActivateOnDoubleClick! Note: activateOnGetfocus! is unsupported.
BackColor	Supported	ole_1.BackColor = ll_value ole_1.BackColor = RGB(192,192,192)
BinaryIndex	Supported	None
BinaryKey	Supported	None

Border	Supported	ole_1.Border = TRUE
		ole_1.Border = lb_value
BorderStyle	Supported	ole_1.BorderStyle = StyleBox!
		ole_1.BorderStyle = StyleLowered!
		Note: StyleShadowBox! is unsupported.
BringToTop	Supported	ole_1.BringToTop = TRUE
DisplayName	Supported	ole_1.DisplayName = "My project"
DisplayType	Supported	ole_1.DisplayType = DisplayAsIcon!
		Note: displayasactivexdocument! is unsupported
DocFileName	Supported	ls_ret = ole_1.docfilename
DragAuto	Supported	ole_1.DragAuto = TRUE
DragIcon	Supported	ole_1.DragIcon = 'Question!'
Enabled	Supported	ole_1.Enabled=TRUE
		ole_1.Enabled = lb_value
FocusRectangle	Supported	ole_1.FocusRectangle = TRUE
		ole_1.FocusRectangle = lb_value
Height	Supported	ole_1.Height = 750
		ole_1.Height = li_value
LinkItem	Supported	ls_ret = iole.LinkItem
Object	Supported	OLEObject obj = ole_1.object.activesheet.cells
		Int x = ole_1.object.month
ObjectData	Supported	blob lblob_data = ole_1.objectdata
Pointer	Supported	ole_1.Pointer = 'SizeWE!'
		ole_1.Pointer='d:\archive\IBeam.BMP.cur'
SizeMode	Supported	ole_1.SizeMode = Clip!
TabOrder	Supported	ole_1.TabOrder = 40
		ole_1.TabOrder = li_order
Tag	Supported	ole_1.Tag = ls_value
Visible	Supported	ole_1.Visible = lb_value
Width	Supported	ole_1.Width = li_value
X	Supported	ole_1.X = li_value
Y	Supported	ole_1.Y = li_value
ClassDefinition	Unsupported	None
ClassLongName	Unsupported	None
ClassShortName	Unsupported	None
ContentsAllowed	Unsupported	None
IsDragTarget	Unsupported	None
LinkUpdateOptions	Unsupported	None
ParentStorage	Unsupported	None

8.1.1.16.2 Events for OLEControl control

Event	Support Level	Example Code
Clicked	Supported	Clicked()
Constructor	Supported	Constructor()
DataChange	Supported	DataChange()
Destructor	Supported	Destructor()
DoubleClicked	Supported	DoubleClicked()

GetFocus	Supported	GetFocus()
LoseFocus	Supported	LoseFocus()
PropertyChanged	Supported	PropertyChanged(propertyname)
PropertyRequestEdit	Supported	PropertyRequestEdit(propertyname,cancelchange)
RButtonDown	Supported	RButtonDown(flags, xpos, ypos)
Close	Unsupported	None
DragDrop	Unsupported	None
DragEnter	Unsupported	None
DragLeave	Unsupported	None
DragWithin	Unsupported	None
Error	Unsupported	None
ExternalException	Unsupported	None
Help	Unsupported	None
Other	Unsupported	None
Rename	Unsupported	None
Save	Unsupported	None
SaveObject	Unsupported	None
ViewChange	Unsupported	None

8.1.1.16.3 Functions for OLEControl control

Function	Support Level	Coding Examples
Activate	Supported	integer li_return li_return = ole_1.Activate(OffSite!) Note: the argument InPlace! is unsupported
ClassName	Supported	ls_return = ole_1.ClassName()
Clear	Supported	li_return = ole_1.Clear()
DoVerb	Supported	li_return = ole_1.DoVerb(7)
GetData	Supported	li_return = ole_1.GetData(ClipFormatText!, ls_oledata)
GetNativePointer	Supported	UnsignedLong lul_oleptr li_return = ole_1.GetNativePointer(lul_oleptr)
GetParent	Supported	PowerObject lpo_parent lpo_parent = ole_1.GetParent()
Hide	Supported	ole_1.Hide()
InsertClass	Supported	li_return = ole_1.InsertClass("excel.sheet")
InsertFile	Supported	li_return = ole_1.InsertFile("c:\xls\expense.xls")
InsertObject	Supported	li_return = ole_1.InsertObject() Items listed in the Web Insert Object dialog box contain and only contain all the registered OLE applications in the client. Items listed in PowerBuilder Insert object dialog box are defined in PowerBuilder.
LinkTo	Supported	li_return = ole_1.LinkTo("c:\xls\expense.xls", "R1C1:R5C5")
Move	Supported	li_return = ole_1.Move(P_Train1.X + P_Train1.Width, P_Train1.Y)
Open	Supported	li_return = ole_1.Open("c:\ole2\mystuff.ole")

PointerX	Supported	li_return = ole_1.PointerX()
PointerY	Supported	li_return = ole_1.PointerY()
PostEvent	Supported	ole_1.PostEvent("DataChange")
ReleaseNativePointer	Supported	li_return = ole_1.ReleaseNativePointer(lul_oleptr)
Resize	Supported	li_return = ole_1.Resize(500, 1000)
Save	Supported	li_return = ole_1.Save()
SaveAs	Supported	li_return = ole_1.SaveAs("c:\ole\expense.ole")
SetData	Supported	li_return = ole_1.SetData(ClipFormatText!, ls_oledata)
SetFocus	Supported	li_return = ole_1.SetFocus()
SetPosition	Supported	ole_1.SetPosition(ToTop!)
		ole_1.SetPosition(Behind!, cb_exec)
SetRedraw	Supported	ole_1.SetRedraw(true)
Show	Supported	ole_1.Show()
TriggerEvent	Supported	li_return = ole_1.TriggerEvent("DataChange")
TypeOf	Supported	if ole_1.typeof()=olecontrol! then ls_return="olecontrol!"
Copy	Unsupported	None
Cut	Unsupported	None
Drag	Unsupported	None
GetContextService	Unsupported	None
Paste	Unsupported	None
PasteLink	Unsupported	None
PasteSpecial	Unsupported	None
Print	Unsupported	None
SelectObject	Unsupported	None
UpdateLinksDialog	Unsupported	None

8.1.1.17 OLECustomControl control

Important Requirements

The shortcut key for Ocx is unsupported.

Using parentheses when calling an OLEObject method is required.

8.1.1.17.1 Properties for OLECustomControl control

Properties	Support Level	Example Code
BackColor	Supported	ole_1.BackColor = ll_value
		ole_1.BackColor = RGB(192,192,192)
BinaryIndex	Supported	None
BinaryKey	Supported	None
Border	Supported	ole_1.Border = TRUE
		ole_1.Border = lb_value
BorderStyle	Supported	ole_1.BorderStyle = StyleBox!
		ole_1.BorderStyle = StyleLowered!
		Note: StyleShadowBox! is unsupported.
BringToTop	Supported	ole_1.BringToTop = TRUE
DisplayName	Supported	ole_1.DisplayName = "My project"

DragAuto	Supported	ole_1.DragAuto = TRUE
DragIcon	Supported	ole_1.DragIcon = 'Question!'
Enabled	Supported	ole_1.Enabled = TRUE
		ole_1.Enabled = lb_value
FocusRectangle	Supported	ole_1.FocusRectangle = TRUE
		ole_1.FocusRectangle = lb_value
Object	Supported	OLEObject obj = ole_1.object.activesheet.cells
		Int x = ole_1.object.month
Pointer	Supported	ole_1.Pointer = 'SizeNWSE!'
		ole_1.Pointer='d:\archive\IBeam.BMP.cur'
TabOrder	Supported	ole_1.TabOrder = 40
		ole_1.TabOrder = li_order
Tag	Supported	ole_1.Tag = ls_value
Visible	Supported	ole_1.Visible = lb_value
Width	Supported	ole_1.Width = li_value
X	Supported	ole_1.X = li_value
Y	Supported	ole_1.Y = li_value
Alignment	Unsupported	None
Cancel	Unsupported	None
ClassDefinition	Unsupported	None
ClassLongName	Unsupported	None
ClassShortName	Unsupported	None
Default	Unsupported	None
FaceName	Unsupported	None
FontCharSet	Unsupported	None
FontFamily	Unsupported	None
FontPitch	Unsupported	None
IsDragTarget	Unsupported	None
Italic	Unsupported	None
TextColor	Unsupported	None
TextSize	Unsupported	None
Underline	Unsupported	None
Weight	Unsupported	None

8.1.1.17.2 Events for OLECustomControl control

Event	Support Level	Example Code
Clicked	Supported	Clicked()
Constructor	Supported	Constructor()
DataChange	Supported	DataChange()
Destructor	Supported	Destructor()
DoubleClicked	Supported	DoubleClicked()
GetFocus	Supported	GetFocus()
LoseFocus	Supported	LoseFocus()
PropertyChanged	Supported	PropertyChanged(propertyname)
PropertyRequestEdit	Supported	PropertyRequestEdit(propertyname,cancelchange)
RButtonDown	Supported	RButtonDown(flags, xpos, ypos)
DragDrop	Unsupported	None

DragEnter	Unsupported	None
DragLeave	Unsupported	None
DragWithin	Unsupported	None
Error	Unsupported	None
ExternalException	Unsupported	None
Help	Unsupported	None
Other	Unsupported	None

8.1.1.17.3 Functions for OLECustomControl control

Function	Support Level	Coding Examples
ClassName	Supported	ls_return = ole_1.ClassName()
GetData	Supported	li_return = ole_1.GetData(ClipFormatText!, ls_oledata)
GetNativePointer	Supported	UnsignedLong lul_oleptr li_return = ole_1.GetNativePointer(lul_oleptr)
GetParent	Supported	PowerObject lpo_parent lpo_parent = ole_1.GetParent()
Hide	Supported	li_return = ole_1.Hide()
Move	Supported	li_return = ole_1.Move(500 , 1000)
PointerX	Supported	li_return = ole_1.PointX()
PointerY	Supported	li_return = ole_1.PointY()
PostEvent	Supported	ole_1.PostEvent("DoubleClicked")
ReleaseNativePointer	Supported	li_return = ole_1.ReleaseNativePointer(lul_oleptr)
Resize	Supported	li_return = ole_1.Resize(500, 1000)
SetData	Supported	li_return = ole_1.SetData(ClipFormatText!, ls_oledata)
SetFocus	Supported	li_return = ole_1.SetFocus()
SetPosition	Supported	ole_1.SetPosition(ToTop!) ole_1.SetPosition(Behind!, cb_exec)
SetRedraw	Supported	ole_1.SetRedraw(true)
Show	Supported	li_return = ole_1.Show()
TriggerEvent	Supported	li_return = ole_1.TriggerEvent("DoubleClicked")
TypeOf	Supported	if ole_1.typeof()=olecustomcontrol! then ls_return="olecustomcontrol!" end if
Drag	Unsupported	None
GetContextService	Unsupported	None
Print	Unsupported	None

8.1.1.18 Oval control

8.1.1.18.1 Properties for Oval control

Property	Support Level	Example Code
FillColor	Supported	o_1.FillColor=RGB(255,255,0)
FillPattern	Supported	o_1.FillPattern = Diamond!

Height	Supported	o_1.Height = 750
LineColor	Supported	o_1.LineColor = RGB(255,0,0)
LineStyle	Supported	o_1.LineStyle = Dash!
LineThickness	Supported	None
Tag	Supported	o_1.SetMicroHelp(This.Tag)
Visible	Supported	o_1.Visible = TRUE
Width	Supported	o_1.Width = 750
X	Supported	o_1.X = 215
Y	Supported	o_1.Y = 215
ClassDefinition	Unsupported	None

8.1.1.18.2 Events for Oval control

Event	Support Level	Example Code
Constructor	Supported	Constructor()
Destructor	Supported	Destructor()

8.1.1.18.3 Functions for Oval control

Function	Support Level	Example Code
ClassName	Supported	ls_value = o_1.ClassName()
GetParent	Supported	lo_value = o_1.GetParent()
Hide	Supported	o_1.Hide()
Move	Supported	o_1.Move(150, 200)
PostEvent	Supported	o_1.PostEvent(Clicked!) Note: the PostEvent function returns 1 if it is successful.
Resize	Supported	o_1.Resize(100, 150)
Show	Supported	o_1.Show()
TriggerEvent	Supported	o_1.TriggerEvent(Clicked!) Note: the TriggerEvent function returns 1 if it is successful.
TypeOf	Supported	if o_1.typeof()=Oval! then ls_return="Oval!" end if
GetContextService	Unsupported	None

8.1.1.19 Picture control

Important Requirements

Picture controls can contain images in the following formats:

- Bitmaps, with .BMP extensions
- GIF or animated GIF files, with the .GIF extension
- JPEG files, with .JPEG or .JPG extensions
- Ico files with .ICO extensions
- Cursor files with .CUR extensions

Pictures in an application

- The name of a picture file cannot be a string that ends with '!'.
 - The picture directory can be an absolute path or relative path. For example:
 - this.** picturename = "c:\app\about.bmp"
 - or this.** picturename = " image\about.bmp"
- If a picture fails to display in the Web application, copy the picture directly to the *Web Root Path /images* directory. *Web Root Path* stands for the folder in the Web server that is configured in Apeon Developer Config for storing the Web application files.

8.1.1.19.1 Properties for Picture control

Property	Support Level	Example Code
Border	Supported	p_1.Border = lb_value
BringToTop	Supported	p_1.BringToTop = TRUE
BorderStyle	Supported	p_1.BorderStyle = StyleBox!
		p_1.BorderStyle = StyleLowered!
		p_1.BorderStyle = StyleRaised!
		Note: StyleShadowBox! is unsupported.
DragAuto	Supported	p_1.DragAuto = TRUE
DragIcon	Supported	p_1.DragIcon = 'c:\examples\arrow.ico'
Enabled	Supported	p_1.Enabled = lb_value
FocusRectangle	Supported	p_1.FocusRectangle = TRUE
Height	Supported	p_1.Height = li_value
Invert	Supported	p_1.Invert = FALSE
OriginalSize	Supported	p_1.OriginalSize = lb_value
PictureName	Supported	p_1.PictureName = ls_value
		Note: It is unsupported to dynamically change this property if the initial value is a GIF file.
Pointer	Supported	p_1.pointer= 'size!'
		p_1.pointer= "d:\archive\IBeam.BMP.cur"
PowerTipText	Supported	p_1.PowerTipText = "Cancel the operation"
TabOrder	Supported	p_1.TabOrder = 100
Tag	Supported	p_1.Tag = ls_value
Visible	Supported	p_1.Visible = lb_value
Width	Supported	p_1.Width = li_value
X	Supported	p_1.X = li_value
Y	Supported	p_1.Y = li_value
ClassDefinition	Unsupported	None
Map3DColors	Unsupported	None

8.1.1.19.2 Events for Picture control

Event	Support Level	Example Code
Clicked	Supported	Clicked()
Constructor	Supported	Constructor()
Destructor	Supported	Destructor()
DoubleClicked	Supported	DoubleClicked()
DragDrop	Supported	DragDrop(source)
DragEnter	Supported	DragEnter(source)

DragLeave	Supported	DragLeave(source)
DragWithin	Supported	DragWithin(source)
GetFocus	Supported	GetFocus()
LoseFocus	Supported	LoseFocus()
RButtonDown	Supported	RButtonDown(flags, xpos, ypos)
Help	Unsupported	None
Other	Unsupported	None

8.1.1.19.3 Functions for Picture control

Function	Support Level	Example Code
ClassName	Supported	ls_value = p_1.ClassName()
Drag	Supported	p_1.Drag(Begin!)
Draw	Supported	li_value = p_1.Draw(li_x,li_y) Note: run the Draw function once the picture has been resized. After conversion, the size of the modified picture will be the same as the most recent state. In PowerBuilder, the size of the drawn picture will be the same as the original one.
GetParent	Supported	PowerObject lpo_value lpo_value = p_1.GetParent()
Hide	Supported	li_value = p_1.Hide()
Move	Supported	li_value = p_1.Move(li_x,li_y)
PointerX	Supported	li_value = p_1.PointerX()
PointerY	Supported	li_value = p_1.PointerY()
PostEvent	Supported	lb_value = p_1.PostEvent(Clicked!)
Resize	Supported	li_value = p_1.Resize(li_x,li_y)
SetFocus	Supported	li_value = p_1.SetFocus() Unsupported: SetFocus(p_1)
SetPicture	Supported	p_1.SetPicture(Emp_pic)
SetPosition	Supported	li_value = p_1.SetPosition()
SetRedraw	Supported	li_value = p_1.SetRedraw(FALSE)
Show	Supported	li_value = p_1.Show()
TriggerEvent	Supported	li_value = p_1.TriggerEvent(Clicked!)
TypeOf	Supported	if p_1.TypeOf() = Picture! Then ls_value = "Picture!" else ls_value = "invalid" end if
GetContextService	Unsupported	None
Print	Unsupported	

8.1.1.20 PictureBox control

Important Requirements

It is unsupported to use GIF files in PictureBox controls.

Properties for PictureButton control

Property	Support Level	Example Code
BringToTop	Supported	pb_1.BringToTop = TRUE
Cancel	Supported	pb_1.Cancel = TRUE
Default	Supported	pb_1.Default = TRUE
DisabledName	Supported	pb_1.DisabledName = "d:\pbhelp\controls.bmp"
DragAuto	Supported	pb_1.DragAuto = TRUE
DragIcon	Supported	pb_1.DragIcon = 'c:\archive\arrow.ico'
Enabled	Supported	pb_1.Enabled = TRUE
FaceName	Supported	pb_1.FaceName = ls_value
FontCharSet	Supported	pb_1.FontCharSet = ANSI!
FontFamily	Supported	pb_1.FontFamily = Roman!
FontPitch	Supported	pb_1.FontPitch = Fixed!
Height	Supported	pb_1.Height = li_value
HTextAlign	Supported	pb_1.HtextAlign = "Left!" //Specifies how the text in the control is aligned. Values are: Center! Justify! Left! Right!
Italic	Supported	pb_1.Italic = lb_value
OriginalSize	Supported	pb_1.OriginalSize = FALSE //Specifies whether the width and height properties of a bitmap image (picture) are set to their original values.
PictureName	Supported	pb_1.PictureName = ls_value
Pointer	Supported	pb_1.pointer = 'Arrow!' pb_1.pointer = 'd:\archive\IBEAM.BMP.cur'
PowerTipText	Supported	pb_1.PowerTipText = "This button opens a new form"
TabOrder	Supported	pb_1.TabOrder = 10
Tag	Supported	pb_1.Tag = ls_value
Text	Supported	pb_1.Text = ls_value
TextSize	Supported	pb_1.TextSize = li_value
Underline	Supported	pb_1.Underline = lb_value
Visible	Supported	pb_1.Visible = lb_value
VTextAlign	Supported	pb_1.VTextAlign= Top! //Specifies how the text in the control is aligned. Values are: Bottom!, MultiLine!, Top!, and VCenter! All these values, except for MultiLine!, assume that there is only one line of text.
Weight	Supported	pb_1.Weight = li_value
Width	Supported	pb_1.Width = li_value
X	Supported	pb_1.X = li_value
Y	Supported	pb_1.Y = li_value
ClassDefinition	Unsupported	None
Map3DColors	Unsupported	None

8.1.1.20.1 Events for PictureButton control

Event	Support Level	Example Code
Clicked	Supported	Clicked()
Constructor	Supported	Constructor()
Destructor	Supported	Destructor()
DragDrop	Supported	DragDrop(source)
DragEnter	Supported	DragEnter(source)
DragLeave	Supported	DragLeave(source)
DragWithin	Supported	DragWithin(source)
GetFocus	Supported	GetFocus()
LoseFocus	Supported	LoseFocus()
RButtonDown	Supported	RButtonDown(flags, xpos, ypos)
Help	Unsupported	None
Other	Unsupported	None

8.1.1.20.2 Functions for PictureButton control

Function	Support Level	Example Code
ClassName	Supported	ls_returnvalue = pb_1.ClassName()
Drag	Supported	pb_1.Drag (Cancel!)
GetParent	Supported	PowerObject lpo_returnvalue lpo_returnvalue = pb_1.GetParent()
Hide	Supported	li_returnvalue = pb_1.Hide()
Move	Supported	li_returnvalue = pb_1.Move(li_xpos, li_ypos)
PointerX	Supported	lb_returnvalue = pb_1.PointerX()
PointerY	Supported	lb_returnvalue = pb_1.PointerY()
PostEvent	Supported	lb_returnvalue = pb_1.PostEvent(event)
Resize	Supported	li_returnvalue = pb_1.Resize(li_width, li_height)
SetFocus	Supported	li_returnvalue = pb_1.SetFocus() Unsupported: SetFocus(pb_1)
SetPosition	Supported	li_returnvalue = pb_1.SetPosition(position)
SetRedraw	Supported	li_returnvalue = pb_1.SetRedraw (TRUE)
Show	Supported	li_returnvalue = pb_1.Show()
TriggerEvent	Supported	li_returnvalue = pb_1.TriggerEvent(event)
TypeOf	Supported	if pb_1.TypeOf() = PictureButton! Then ls_returnvalue = 'PictureButton!' end if
GetContextService	Unsupported	None
Print	Unsupported	None

8.1.1.21 PictureHyperLink control**Important Requirements**

The PictureHyperLink controls can contain images in the following formats:

- Bitmaps, with .BMP extensions

- GIF or animated GIF files, with the .GIF extensions
- JPEG files, with .JPEG or .JPG extensions
- Ico files with .ICO extensions
- Cursor files with .CUR extensions

8.1.1.21.1 Properties for PictureHyperLink control

Property	Support Level	Example Code
Border	Supported	phl_1.Border = lb_value
BorderStyle	Supported	phl_1.BorderStyle = StyleBox!
		phl_1.BorderStyle = StyleLowered!
		phl_1.BorderStyle = StyleRaised!
		Note: StyleShadowBox! is unsupported.
BringToTop	Supported	phl_1.BringToTop = TRUE
DragAuto	Supported	None
DragIcon	Supported	None
Enabled	Supported	phl_1.Enabled = lb_value
FocusRectangle	Supported	phl_1.FocusRectangle = TRUE
Height	Supported	phl_1.Height = li_value
OriginalSize	Supported	phl_1.OriginalSize = lb_value
PictureName	Supported	phl_1.PictureName = ls_value
Pointer	Supported	phl_1='Size!'
		phl_1='d:\archive\IBEAM.BMP.cur'
PowerTipText	Supported	phl_1.PowerTipText = "This button opens a new form"
TabOrder	Supported	phl_1.TabOrder = 100
Tag	Supported	phl_1.Tag = ls_value
URL	Supported	phl_1.URL = " http://www.appeon.com/ "
Visible	Supported	phl_1.Visible = lb_value
Width	Supported	phl_1.Width = li_value
X	Supported	phl_1.X = li_value
Y	Supported	phl_1.Y = li_value
ClassDefinition	Unsupported	None
Invert	Unsupported	None
Map3DColors	Unsupported	None

8.1.1.21.2 Events for PictureHyperLink control

Event	Support Level	Example Code
Clicked	Supported	Clicked()
Constructor	Supported	Constructor()
Destructor	Supported	Destructor()
DoubleClicked	Supported	DoubleClicked()
DragDrop	Supported	DragDrop(source)
DragEnter	Supported	DragEnter(source)
DragLeave	Supported	DragLeave(source)
DragWithin	Supported	DragWithin(source)
GetFocus	Supported	GetFocus()
LoseFocus	Supported	LoseFocus()

RButtonDown	Supported	RButtonDown(flags, xpos, ypos)
Help	Unsupported	None
Other	Unsupported	None

8.1.1.21.3 Functions for PictureHyperLink control

Function	Support Level	Example Code
ClassName	Supported	ls_value = phl_1.ClassName()
Draw	Supported	li_value = phl_1.Draw(li_x,li_y)
Drag	Supported	phl_1.Drag(Begin!)
GetParent	Supported	PowerObject lpo_value lpo_value = phl_1.GetParent()
Hide	Supported	li_value = phl_1.Hide()
Move	Supported	li_value = phl_1.Move(li_x,li_y)
PointerX	Supported	li_value = phl_1.PointerX()
PointerY	Supported	li_value = phl_1.PointerY()
PostEvent	Supported	lb_value = phl_1.PostEvent(Clicked!)
Resize	Supported	li_value = phl_1.Resize(li_x,li_y)
SetFocus	Supported	li_value = phl_1.SetFocus() Unsupported: SetFocus(phl_1)
SetPicture	Supported	phl_1.SetPicture(Emp_pic)
SetPosition	Supported	li_value = phl_1.SetPosition()
SetRedraw	Supported	li_returnvalue = phl_1.SetRedraw (FALSE)
Show	Supported	li_value = phl_1.Show()
TriggerEvent	Supported	li_value = phl_1.TriggerEvent(Clicked!)
TypeOf	Supported	if phl_1.TypeOf() = PictureHyperLink! Then ls_value = "PictureHyperLink!" else ls_value = "invalid" end if
GetContextService	Unsupported	None
Print	Unsupported	None

8.1.1.22 PictureListBox control

Important Requirements

Setting a PictureListBox as a Dropdown PictureListBox is not supported.

8.1.1.22.1 Properties for PictureListBox control

Property	Support Level	Example Code
Accelerator	Supported	plb_1.Accelerator=67
BackColor	Supported	plb_1.BackColor = ll_value

		plb_1.BackColor = RGB(192,192,192)
Border	Supported	plb_1.Border = lb_value
BorderStyle	Supported	plb_1.BorderStyle = StyleLowered!
		plb_1.BorderStyle = StyleRaised!
		plb_1.BorderStyle = StyleBox!
		Note: StyleShadowBox! is not supported.
BringToTop	Supported	plb_1.BringToTop = TRUE
DisableNoScroll	Supported	plb_1.DisableNoScroll = lb_value
DragAuto	Supported	plb_1.DragAuto = TRUE
DragIcon	Supported	plb_1.DragIcon ='c:\archive\arrow.ico'
Enabled	Supported	plb_1.Enabled = lb_value
ExtendedSelect	Supported	plb_1.ExtendedSelect = lb_value
FaceName	Supported	plb_1.FaceName = ls_value
FontCharSet	Supported	plb_1.FontCharSet = ANSI!
FontFamily	Supported	plb_1.FontFamily = Roman!
FontPitch	Supported	plb_1.FontPitch = Fixed!
Height	Supported	plb_1.Height = li_value
HScrollBar	Supported	plb_1.Hscrollbar = lb_value
Italic	Supported	plb_1.Italic = lb_value
Item[]	Supported	plb_1.Item[] = ls_value
ItemPictureIndex[]	Supported	plb_1.ItemPictureIndex[] = li_value[]
MultiSelect	Supported	plb_1.MultiSelect = lb_value
PictureHeight	Supported	plb_1.PictureHeight = 16

PictureWidth	Supported	plb_1.PictureWidth = 16
PictureName[]	Supported	ls_result[] = plb_1.PictureName[li_x[]]
		Assigning values to part of an array and using the default values for the other part is not supported. For example: <pre>picturename[1] = "aaa.bmp" picturename[3] = "bbb.bmp" var pic1 = picutrename[1]; var pic2 = picturename[3];</pre>
		A supported example is given as follows: <pre>picturename[1] = "aaa.bmp" picturename[2] = "bbb.bmp" picturename[3] = "ccc.bmp" var pic1 = picutrename[1]; var pic2 = picturename[2]; var pic3 = picturename[3];</pre>
Pointer	Supported	plb_1.pointer = 'Icon!' plb_1.pointer = 'd:\archive\IBEAM.BMP'
RightToLeft	Supported	plb_1.RightToLeft = TRUE
Sorted	Supported	plb_1.Sorted = lb_value
TabOrder	Supported	plb_1.TabOrder = 110
Tag	Supported	plb_1.Tag = ls_value
TextColor	Supported	plb_1.Textcolor = ll_value
TextSize	Supported	plb_1.TextSize = li_value
Underline	Supported	plb_1.Underline = lb_value
Visible	Supported	plb_1.Visible = lb_value
VScrollBar	Supported	plb_1.VScrollBar = lb_value
Weight	Supported	plb_1.Weight = li_value

Width	Supported	plb_1.Width = li_value
X	Supported	plb_1.X = li_value
Y	Supported	plb_1.Y = li_value
ClassDefinition	Unsupported	None
PictureMaskColor	Unsupported	None
ShowList	Unsupported	None
TabStop[]	Unsupported	None

8.1.1.22.2 Events for PictureListBox control

Event	Support Level	Example Code
Constructor	Supported	Constructor()
Destructor	Supported	Destructor()
DoubleClicked	Supported	DoubleClicked()
DragDrop	Supported	DragDrop(source)
DragEnter	Supported	DragEnter(source)
DragLeave	Supported	DragLeave(source)
DragWithin	Supported	DragWithin(source)
GetFocus	Supported	GetFocus()
LoseFocus	Supported	LoseFocus()
RButtonDown	Supported	RButtonDown(flags, xpos, ypos)
SelectionChanged	Supported	SelectionChanged(index)

Help	Unsupported	None
Other	Unsupported	None

8.1.1.22.3 Functions for PictureListBox control

Function	Support Level	Example Code
AddItem	Supported	li_position = plb_1.AddItem(ls_name, li_pic)
AddPicture	Supported	li_pic = plb_1.AddPicture("c:\pics\cardinal.bmp")
ClassName	Supported	ls_value = plb_1.ClassName()
DeleteItem	Supported	plb_1.DeleteItem(li_value)
DeletePicture	Supported	plb_1.DeletePicture (6)
DeletePictures	Supported	plb_1.DeletePictures ()
DirList	Supported	plb_1.DirList("C:\EMPLOYEE*.TXT", 0)
DirSelect	Supported	plb_1.DirSelect(ls_filename)
Drag	Supported	plb_1.Drag(End!)
FindItem	Supported	li_index = plb_1.FindItem ("Delete", 1)
GetParent	Supported	PowerObject lpo_value lpo_value = plb_1.GetParent()
Hide	Supported	li_value = plb_1.Hide()
InsertItem	Supported	plb_1.InsertItem("Run Application", 5)
		plb_1.InsertItem("Run Application", 5, 4)
Move	Supported	li_value = plb_1.Move(li_x[],li_y[])
PointerX	Supported	li_value = plb_1.PointerX()
PointerY	Supported	li_value = plb_1.PointerY()

PostEvent	Supported	lb_value = plb_1.PostEvent(Clicked!)
Reset	Supported	li_value = plb_1.Reset()
Resize	Supported	li_value = plb_1.Resize(li_x[],li_y[])
SelectedIndex	Supported	li_index = plb_1.SelectedIndex()
SelectedItem	Supported	li_item = plb_1.SelectedItem()
SelectItem	Supported	li_value = plb_1.SelectItem (li_x[],li_y[])
SetFocus	Supported	li_value = plb_1.SetFocus()
		Unsupported: SetFocus(plb_1)
SetPosition	Supported	li_returnvalue = plb_1.SetPosition(position)
SetRedraw	Supported	li_returnvalue = plb_1.SetRedraw (FALSE)
SetState	Supported	plb_1.SetState(6, TRUE)
SetTop	Supported	plb_1.SetTop(6)
Show	Supported	li_value = plb_1.Show()
State	Supported	li_item = plb_1.State(3)
Text	Supported	ls_return = plb_1.Text(2)
Top	Supported	li_num = plb_1.Top()
TotalItems	Supported	li_value= plb_1.TotalItems()
TotalSelected	Supported	integer SelectedTotal SelectedTotal = plb_1.TotalSelected()
TriggerEvent	Supported	li_value = plb_1.TriggerEvent(Constructor!)

TypeOf	Supported	if plb_1.typeof()=picturelistbox! Then ls_value='picturelistbox!' else ls_value='isvalid' end if
Clear	Unsupported	None
Copy	Unsupported	None
Cut	Unsupported	None
GetContextService	Unsupported	None
Paste	Unsupported	None
Position	Unsupported	None
Print	Unsupported	None

8.1.1.23 RadioButton control

8.1.1.23.1 Properties for RadioButton control

Property	Support Level	Example Code
Automatic	Supported	rb_1.Automatic = TRUE
BackColor	Supported	rb_1.BackColor = ll_value
		rb_1.BackColor = RGB(192,192,192)
BorderStyle	Supported	rb_1.BorderStyle = StyleLowered!
		Note: StyleRaised! is unsupported.
BringToTop	Supported	rb_1.BringToTop = TRUE
Checked	Supported	rb_1.Checked = lb_value

DragAuto	Supported	rb_1.DragAuto = TRUE
DragIcon	Supported	rb_1.DragIcon ='c:\archive\arrow.ico'
Enabled	Supported	rb_1.Enabled = lb_value
FaceName	Supported	rb_1.FaceName = ls_value
FontCharSet	Supported	rb_1.FontCharSet = ANSI!
FontFamily	Supported	rb_1.FontFamily = Roman!
FontPitch	Supported	rb_1.FontPitch = Fixed!
Height	Supported	rb_1.Height = li_value
LeftText	Supported	rb_1.LeftText = TRUE
Italic	Supported	rb_1.Italic = lb_value
Pointer	Supported	rb_1.Pointer ='AppStarting!'
		rb_1.Pointer='d:\archive\IBEAM.BMP'
RightToLeft	Supported	rb_1.RightToLeft = TRUE
TabOrder	Supported	rb_1.TabOrder = 120
Tag	Supported	rb_1.Tag = ls_value
Text	Supported	rb_1.Text = ls_value
TextColor	Supported	rb_1.TextColor = ll_value
TextSize	Supported	rb_1.TextSize = li_value
Underline	Supported	rb_1.Underline = lb_value
Visible	Supported	rb_1.Visible = lb_value
Weight	Supported	rb_1.Weight = li_value
Width	Supported	rb_1.Width = li_value
X	Supported	rb_1.X = li_value

Y	Supported	rb_1.Y = li_value
ClassDefinition	Unsupported	None

8.1.1.23.2 Events for RadioButton control

Event	Support Level	Example Code
Clicked	Supported	Clicked()
Constructor	Supported	Constructor()
Destructor	Supported	Destructor()
DragDrop	Supported	DragDrop(source)
DragEnter	Supported	DragEnter(source)
DragLeave	Supported	DragLeave(source)
DragWithin	Supported	DragWithin(source)
GetFocus	Supported	GetFocus()
LoseFocus	Supported	LoseFocus()
RButtonDown	Supported	RButtonDown(flags, xpos, ypos)
Help	Unsupported	None
Other	Unsupported	None

8.1.1.23.3 Functions for RadioButton control

Function	Support Level	Example Code
ClassName	Supported	ls_value = rb_1.ClassName()
Drag	Supported	rb_1.Drag(Begin!)

GetParent	Supported	PowerObject lpo_value lpo_value = rb_1.GetParent()
Hide	Supported	li_value = rb_1.Hide()
Move	Supported	li_value = rb_1.Move(li_x,li_y)
PointerX	Supported	li_vaule = rb_1.PointerX()
PointerY	Supported	li_value = rb_1.PointerY()
PostEvent	Supported	lb_value = rb_1.PostEvent(Clicked!)
Resize	Supported	li_value = rb_1.Resize(li_x,li_y)
SetFocus	Supported	li_value = rb_1.SetFocus()
		Unsupported: SetFocus(rb_1)
SetPosition	Supported	li_value = rb_1.SetPosition(ToTop!)
		li_value = rb_1.SetPosition(ToBottom!)
SetRedraw	Supported	li_returnvalue = rb_1.SetRedraw (FALSE)
Show	Supported	li_value = rb_1.Show()
TriggerEvent	Supported	li_value = rb_1.TriggerEvent(Clicked!)
TypeOf	Supported	if rb_1.TypeOf() = RadioButton! Then ls_value = "RadioButton!" end if
GetContextService	Unsupported	None
Print	Unsupported	None

8.1.1.24 Rectangle control

8.1.1.24.1 Properties for Rectangle control

Property	Support Level	Example Code
FillColor	Supported	<code>r_1.FillColor = ll_value</code>
FillPattern	Supported	<code>r_1.FillPattern = Diamond!</code>
Height	Supported	<code>r_1.Height = li_value</code>
LineColor	Supported	<code>r_1.LineColor = ll_value</code>
LineStyle	Supported	<code>r_1.LineStyle = Continuous!</code>
		<code>r_1.LineStyle = Dash!</code>
		<code>r_1.LineStyle = DashDot!</code>
		<code>r_1.LineStyle = DashDotDot!</code>
		<code>r_1.LineStyle = Dot!</code>
		<code>r_1.LineStyle = Transparent!</code>
LineThickness	Supported	<code>r_1.LineThickness = li_value</code>
		Note: If LineThickness is greater than one pixel (about four PowerBuilder units), the LineStyle property is forced to Continuous!
Tag	Supported	<code>r_1.Tag = ls_value</code>
Visible	Supported	<code>r_1.Visible = lb_value</code>
Width	Supported	<code>r_1.Width = li_value</code>
X	Supported	<code>r_1.X = li_value</code>
Y	Supported	<code>r_1.Y = li_value</code>
ClassDefinition	Unsupported	None

8.1.1.24.2 Events for Rectangle control

Event	Support Level	Example Code
Constructor	Supported	Constructor()
Destructor	Supported	Destructor()

8.1.1.24.3 Functions for Rectangle control

Function	Support Level	Example Code
ClassName	Supported	ls_value = r_1.ClassName()
GetParent	Supported	PowerObject lpo_value lpo_value = r_1.GetParent()
Hide	Supported	li_value = r_1.Hide()
Move	Supported	li_value = r_1.Move(li_x,li_y)
PostEvent	Supported	lb_value = r_1.PostEvent(Clicked!) The PostEvent function returns 1 if it is successful.
Resize	Supported	lb_value = r_1.Resize(Clicked!)
Show	Supported	li_value = r_1.Show()
TriggerEvent	Supported	li_value = r_1.TriggerEvent(Clicked!) The TriggerEvent function returns 1 if it is successful.
TypeOf	Supported	if r_1.TypeOf() = Rectangle! Then li_value = "Rectangle! " end if
GetContextService	Unsupported	None

8.1.1.25 RichTextEdit control

Differences

1. On the Web, the printed result of the RichText may be different from what you see in the RichText control. This is because the printed result will be reformatted according to the size of the paper.
2. On the Web, an input field is selected when you click on it, you can then delete or replace it by using the keyboard. This is different from that in PowerBuilder.
3. Because of the following differences, the return value on the Web and in PowerBuilder is different when using the functions that relates to the calculation of length. For example, SelectText, SelectedText, SelectAllText, CopyRTF, PasteRTF, Position, Clear, LineCount, TextLine, Copy, Cut, Position, Paste and SelectLength.
 - A picture will be counted as one character on the Web.
 - A new line character will be counted as one character on the Web.
 - A line ending character will be counted as one character.
 - An input field will be counted as separated characters, therefore the return value will be the length of the content in the input fields or the content of the input field.
4. Apeon adopts Windows RTF specification, which differs from PowerBuilder RTF specification, so the displaying result on the Web may differ from that in PowerBuilder. For example: Copying and pasting the content to the RichText control, the displaying result on the Web is different from that in PowerBuilder; The content of RichTextEdit control saved on the Web differs from that in PowerBuilder. However the pure text displayed is almost the same.
5. A menu does not pop up when right clicking a picture in the RichTextEdit control on the Web. This is different from that in PowerBuilder.
6. When a part of text in an InputField is selected, pressing Tab control replaces the selected text with a white space on the Web while a white space is inserted before the selected text in PowerBuilder.

Unsupported

1. After you double click the ruler bar in the RichTextEdit control, Paragraph window will be displayed. On the Web, the Standard tab item in the Paragraph window is unsupported.
2. End identifier is unsupported.

8.1.1.25.1 Properties for RichTextEdit control

Property	Support Level	Example Code
Accelerator	Supported	rte_1.Accelerator=67
		rte_1.Accelerator = ASC("M")
BackColor	Supported	rte_1.BackColor = RGB(255, 255, 0)
Border	Supported	rte_1.Border = TRUE

BorderStyle	Supported	rte_1.BorderStyle = StyleLowered!
BottomMargin	Supported	rte_1.BottomMargin = 1
BringToTop	Supported	rte_1.BringToTop = TRUE
DisplayOnly	Supported	rte_1.DisplayOnly = TRUE
DocumentName	Supported	rte_1.DocumentName = "Report 1"
DragAuto	Supported	rte_1.DragAuto = TRUE
DragIcon	Supported	rte_1.DragIcon = 'Question!'
Enabled	Supported	rte_1.Enabled = TRUE
HeaderFooter	Supported	None
Height	Supported	rte_1.Height = 750
HScrollBar	Supported	rte_1.HScrollBar = TRUE Note: On the Web, the horizontal scrollbar is always displayed even if the property is set to FALSE, However, the scroll box of the scrollbar is displayed only when all of the data cannot be displayed at one time
InputFieldNamesVisible	Supported	rte_1.InputFieldNamesVisible = FALSE
InputFieldsVisible	Supported	rte_1.InputFieldsVisible = TRUE
LeftMargin	Supported	rte_1.LeftMargin = 1
Modified	Supported	None
PopMenu	Supported	rte_1.PopMenu = TRUE
Resizable	Supported	rte_1.Resizable = TRUE Note: After setting this property to True, setting Border properties will not be effective.
RightMargin	Supported	rte_1.RightMargin = 1

RulerBar	Supported	rte_1.RulerBar = TRUE
TabBar	Supported	rte_1.TabBar = TRUE
TabOrder	Supported	rte_1.TabOrder = 10
Tag	Supported	rte_1.SetMicroHelp(This.Tag)
ToolBar	Supported	rte_1.ToolBar = TRUE
TopMargin	Supported	rte_1.TopMargin = 1
Visible	Supported	rte_1.Visible = TRUE
VScrollBar	Supported	rte_1.VScrollBar = TRUE
		Note: On the Web, the vertical scrollbar is always displayed even if the property is set to FALSE. However, the scroll box of the scrollbar is displayed only when the contents of the object are outside the borders.
Width	Supported	rte_1.Width = 750
WordWrap	Supported	rte_1.WordWrap = TRUE
		Note: 1. When this property is changed from TRUE to FALSE, the content in the RichTextEdit control will be rearranged on the Web. This is different from that in PowerBuilder. 2. If this property is set to False, you cannot execute SetAlignment to set the alignment for the selected paragraphs on the Web and the text will not automatically wrap to the next line even if the line reaches the margin. These are different from that in PowerBuilder. 3. Alignment buttons on the Web are only effective if the WordWrap property is set to True. This is different from that in PowerBuilder.
X	Supported	rte_1.X = 215
Y	Supported	rte_1.Y = 215

ClassDefinition	Unsupported	None
ImeMode	Unsupported	None
InputFieldBackColor	Unsupported	None
Pointer	Unsupported	None
PicturesAsFrame	Unsupported	None
ReturnsVisible	Unsupported	None
SpacesVisible	Unsupported	None
TabsVisible	Unsupported	None
UndoDepth	Unsupported	None

8.1.1.25.2 Events for RichTextEdit control

Event	Support Level	Example Code
Constructor	Supported	Constructor
Destructor	Supported	Destruvtor
DoubleClicked	Supported	DoubleClicked
GetFocus	Supported	rte_1 = GetFocus()
Key	Supported	IF key = KeyF1! THEN Beep(1)
		Note: Some keystrokes take no effect in formatting the document in RichText DataWindow. For example, pressing Esc after updating data in an inputfield restores the data in PowerBuilder but takes no effect on the Web.
LoseFocus	Supported	LoseFocus
Modified	Supported	Modified
MouseDown	Supported	rte_1.text = "Mouse Down"

MouseMove	Supported	rte_1.Text = "Mouse Move"
MouseUp	Supported	rte_1.Text = "Mouse Up"
RButtonDown	Supported	RButtonDown
RButtonUp	Supported	RButtonUp
DragDrop	Unsupported	None
DragEnter	Unsupported	None
DragLeave	Unsupported	None
DragWithin	Unsupported	None
FileExists	Unsupported	None
Help	Unsupported	None
InputFieldSelected	Unsupported	None
Other	Unsupported	None
PictureSelected	Unsupported	None
PrintFooter	Unsupported	None
PrintHeader	Unsupported	None

8.1.1.25.3 Functions for RichTextEdit control

Function	Support Level	Example Code
ClassName	Supported	ls_value = rte_1.ClassName(gd_double)
Clear	Supported	rte_1.Clear()
		Note: Executing this function on the Web clears the selected text as well as the format from the control. However, in PowerBuilder, it only clears the selected text.
Copy	Supported	li_value = rte_1.Copy()

CopyRTF	Supported	ls_value = rte_1.CopyRTF()
Cut	Supported	rte_1.Cut()
DataSource	Supported	rte_1.DataSource(dw_1)
Drag	Supported	rte_1.Drag(Begin!)
Find	Supported	li_value = rte_1.Find(sle_search.Text, TRUE, & TRUE, FALSE, TRUE)
FindNext	Supported	rte_1.FindNext()
GetAlignment	Supported	l_align = rte_1.GetAlignment()
GetParagraphSetting	Supported	ll_value = rte_1.GetParagraphSetting(Indent!)
GetParent	Supported	luo_value = rte_1.GetParent()
GetSpacing	Supported	l_spacing = rte_1.GetSpacing()
GetTextColor	Supported	ll_color = rte_1.GetTextColor()
GetTextStyle	Supported	lb_value = rte_1.GetTextStyle(Bold!)
Hide	Supported	rte_1.Hide()
InputFieldChangeData	Supported	li_value = rte_1.InputFieldChangeData ("instrument", & lb_instruments.SelectedItem())
InputFieldCurrentName	Supported	ls_value = rte_1.InputFieldCurrentName()
InputFieldDeleteCurrent	Supported	li_value = rte_1.InputFieldDeleteCurrent() The input filed that is being edited can be deleted by executing the InputFieldDeleteCurrent function on the Web. But it cannot be deleted in PowerBuilder in that way.
InputFieldGetData	Supported	ls_value = rte_1.InputFieldGetData(empname)
InputFieldInsert	Supported	ls_value = String(rte_1.InputFieldInsert("lastname"))

t		Note: On the Web, when you click on an input field and then execute InputFieldInsert, the input field is replaced by the new one. In PowerBuilder, the new input field is inserted and the old one will still exist.
InputFieldLocate	Supported	ls_value = rte_1.InputFieldLocate(Last!, "address")
InsertDocument	Supported	li_value = rte_1.InsertDocument("c:\pb\test.rtf", & TRUE, FileTypeRichText!)
InsertPicture	Supported	li_value = rte_1.InsertPicture("c:\windows\earth.bmp")
IsPreview	Supported	lb_value = rte_1.IsPreview()
LineCount	Supported	li_value = rte_1.LineCount()
LineLength	Supported	li_value = rte_1.LineLength()
Move	Supported	rte_1.Move(150, 200)
PageCount	Supported	ls_value = String(rte_1.PageCount())
Paste	Supported	rte_1.Paste() Note: Double-byte characters inserted by executing this function can not be displayed properly.
PasteRTF	Supported	string ls_richtext rte_1.PasteRTF(ls_richtext, Header!)
PointerX	Supported	li_value = rte_1.PointerX()
PointerY	Supported	li_value = rte_1.PointerY()
Position	Supported	l_band = rte_1.Position(ll_startline, ll_startchar, & ll_endline, ll_endchar)
PostEvent	Supported	rte_1.PostEvent(Clicked!)
Preview	Supported	rte_1.Preview(TRUE)

		Note: When executing this function to display the RichTextEdit control, the content will be rearranged according to the size of the page on the Web. This is different from that in PowerBuilder.
Print	Supported	rte_1.Print(1, "1-5", FALSE, TRUE)
ReplaceText	Supported	rte_1.ReplaceText("Draft")
		Note: If the text contains double-byte characters, the return value of this function is different from in PowerBuilder.
Resize	Supported	rte_1.Resize(100, 150)
SaveDocument	Supported	rte_1.SaveDocument("c:\test.rtf", FileTypeRichText!) Note: encoding argument is unsupported.
Scroll	Supported	rte_1.Scroll(4)
ScrollNextPage	Supported	rte_1.ScrollNextPage()
ScrollNextRow	Supported	rte_1.ScrollNextRow()
ScrollPriorPage	Supported	rte_1.ScrollPriorPage()
ScrollPriorRow	Supported	rte_1.ScrollPriorRow()
ScrollToRow	Supported	rte_1.ScrollToRow(25)
SelectedLength	Supported	ll_length = rte_1.SelectedLength()
SelectedLine	Supported	li_value = rte_1.SelectedLine()
SelectedPage	Supported	li_value = rte_1.SelectedPage()
SelectedStart	Supported	li_value = rte_1.SelectedStart()
SelectedText	Supported	ls_value = rte_1.SelectedText()

		<p>Notes:</p> <ol style="list-style-type: none"> 1. Executing the this function on the Web will select text and pictures , however the pictures will not appear to be selected. 2. It returns a string containing the selected text when selecting text in the header or footer bands on the Web. This is different from in PowerBuilder.
SelectText	Supported	<p>rte_1.SelectText(1,1, 3,4)</p> <p>Note:</p> <ol style="list-style-type: none"> 1. When the focus is on an input field, executing the SelectText function will select the input field however the input field will not appear to be selected on the Web. 2. After executing the SelectText function on the Web, if the focus is not at the area where the arguments specify, the focus will be automatically placed at that area. This is different from that in PowerBuilder. 3. If the ReadOnly is set to True, this function still takes effect on the Web.
SelectTextAll	Supported	<p>rte_1.SelectTextAll()</p> <p>Note: If the focus is in the header or footer band, executing “SelectTextAll (detail!)” on the Web returns -1 and no text will be selected. However, in PowerBuilder, the text in the band, which the focus is in, will be selected.</p>
SelectTextLine	Supported	rte_1.SelectTextLine()
SelectTextWord	Supported	rte_1.SelectTextWord()
SetAlignment	Supported	li_value = rte_1.SetAlignment(Right!)
SetFocus	Supported	rte_1.SetFocus()
SetParagraphSetting	Supported	ll_indent = rte_1.SetParagraphSetting(Indent!, 250)
SetPosition	Supported	rte_1.SetPosition(ToTop!)
SetRedraw	Supported	rte_1.SetRedraw(FALSE)
SetSpacing	Supported	rte_1.SetSpacing(Spacing2!)

SetTextColor	Supported	rte_1.SetTextColor(RGB(100, 0, 0))
SetTextStyle	Supported	rte_1.SetTextStyle(TRUE, FALSE, FALSE, & FALSE, TRUE, FALSE) Note: If the RichTextEdit control retrieves the focus by executing the SetFocus or SetText, executing the SetTextStyle function will affect the formatting of the following inputted text on the Web. This is different from that in PowerBuilder.
Show	Supported	rte_1.Show()
ShowHeadFoot	Supported	rte_1.ShowHeadFoot(TRUE)
TextLine	Supported	ls_value = rte_1.TextLine()
TriggerEvent	Supported	rte_1.TriggerEvent(Clicked!)
TypeOf	Supported	rte_1.Typeof()
CanUndo	Unsupported	None
GetContextService	Unsupported	None
SelectedColumn	Unsupported	None
Undo	Unsupported	None

8.1.1.26 RoundedRectangle control

8.1.1.26.1 Properties for RoundedRectangle control

Property	Support Level	Example Code
CornerHeight	Supported	rr_1.CornerHeight = 10
CornerWidth	Supported	rr_1.CornerWidth = 10

FillColor	Supported	rr_1.FillColor=RGB(255,255,0)
FillPattern	Supported	rr_1.FillPattern = Diamond!
Height	Supported	rr_1.Height = 750
LineColor	Supported	rr_1.LineColor = RGB(255,0,0)
LineStyle	Supported	rr_1.LineStyle = Dash!
LineThickness	Supported	rr_1.LineThickness =li_value
Tag	Supported	rr_1.Tag = ls_value
Visible	Supported	rr_1.Visible = True
Width	Supported	rr_1.Width = 750
X	Supported	rr_1.X = 215
Y	Supported	rr_1.Y = 215
ClassDefinition	Unsupported	None

8.1.1.26.2 Events for RoundRectangle control

Event	Support Level	Example Code
Constructor	Supported	Constructor()
Destructor	Supported	Destructor()

8.1.1.26.3 Functions for RoundRectangle control

Function	Support Level	Example Code
ClassName	Supported	ls_value = rr_1.ClassName()
GetParent	Supported	PowerObject lpo_value lpo_value = rr_1.GetParent()

Hide	Supported	li_value = rr_1.Hide()
Move	Supported	li_value = rr_1.Move(li_x,li_y)
PostEvent	Supported	lb_value = rr_1.PostEvent(Clicked!)
		The PostEvent function returns 1 if it is successful.
Resize	Supported	lb_value = rr_1.Resize(Clicked!)
Show	Supported	li_value = rr_1.Show()
TriggerEvent	Supported	li_value = rr_1.TriggerEvent(Clicked!)
		The TriggerEvent function returns 1 if it is successful.
TypeOf	Supported	if rr_1.TypeOf() = RoundedRectangle! Then li_value = "RoundRectangle!" end if
GetContextService	Unsupported	None

8.1.1.27 SingleLineEdit control

8.1.1.27.1 Properties for SingleLineEdit control

Property	Support Level	Example Code
Accelerator	Supported	sle_1.Accelerator = 67
AutoHScroll	Supported	sle_1.AutoHScroll = TRUE
BackColor	Supported	sle_1.BackColor = ll_value
Border	Supported	sle_1.Border = lb_value
BorderStyle	Supported	sle_1.BorderStyle = StyleBox!
		sle_1.BorderStyle = StyleLowered!

		sle_1.BorderStyle = StyleRaised!
		Note: StyleShadowBox! is unsupported.
BringToTop	Supported	sle_1.BringToTop = TRUE
DisplayOnly	Supported	sle_1.DisplayOnly = lb_value
DragAuto	Supported	sle_1.DragAuto = TRUE
DragIcon	Supported	sle_1.DragIcon ='c:\archive\arrow.ico'
Enabled	Supported	sle_1.Enabled = lb_value
FaceName	Supported	sle_1.FaceName = ls_value
FontCharSet	Supported	sle_1.FontCharSet = ANSI!
FontFamily	Supported	sle_1.FontFamily = Roman!
FontPitch	Supported	sle_1.FontPitch = Fixed!
Height	Supported	sle_1.Height = li_value
HideSelection	Supported	sle_1.HideSelection = FALSE
Italic	Supported	sle_1.Italic = lb_value
Limit	Supported	sle_1.Limit = li_value
Password	Supported	sle_1.Password = lb_value
Pointer	Supported	sle_1.pointer= 'Arrow!'
		sle_1.pointer= 'd:\archive\IBEAM.BMP.cur'
RightToLeft	Supported	sle_1.RightToLeft = TRUE
TabOrder	Supported	sle_1.TabOrder = 130
Tag	Supported	sle_1.Tag = ls_value
Text	Supported	sle_1.Text = ls_value
TextCase	Supported	sle_1.TextCase = AnyCase!

		sle_1.TextCase = Lower!
		sle_1.TextCase = Upper!
TextColor	Supported	sle_1.TextColor = ll_value
TextSize	Supported	sle_1.TextSize = li_value
Underline	Supported	sle_1.Underline = lb_value
Visible	Supported	sle_1.Visible = lb_value
Weight	Supported	sle_1.Weight = li_value
Width	Supported	sle_1.Width = li_value
X	Supported	sle_1.X = li_value
Y	Supported	sle_1.Y = li_value
ClassDefinition	Unsupported	None

8.1.1.27.2 Events for SingleLineEdit control

Event	Support Level	Example Code
Constructor	Supported	Constructor()
Destructor	Supported	Destructor()
DragDrop	Supported	DragDrop(source)
DragEnter	Supported	DragEnter(source)
DragLeave	Supported	DragLeave(source)
DragWithin	Supported	DragWithin(source)
GetFocus	Supported	GetFocus()
LoseFocus	Supported	LoseFocus()
Modified	Supported	Modified()

RButtonDown	Supported	RButtonDown(flags, xpos, ypos)
Help	Unsupported	None
Other	Unsupported	None

8.1.1.27.3 Functions for SingleLineEdit control

Function	Support Level	Example Code
ClassName	Supported	ls_returnvalue = sle_1.ClassName()
Clear	Supported	li_returnvalue = sle_1.Clear()
Copy	Supported	li_returnvalue = sle_1.Copy()
Cut	Supported	li_returnvalue = sle_1.Cut()
Drag	Supported	sle_1Drag(Cancel!)
GetParent	Supported	PowerObject lpo_returnvalue lpo_returnvalue = sle_1.GetParent()
Hide	Supported	li_returnvalue = sle_1.Hide()
Move	Supported	li_returnvalue = sle_1.Move(li_x,li_y)
Paste	Supported	li_returnvalue = sle_1.Paste()
PointerX	Supported	li_returnvalue = sle_1.PointerX()
PointerY	Supported	li_returnvalue = sle_1.PointerY()
Position	Supported	sle_1.Position()
PostEvent	Supported	lb_returnvalue = sle_1.PostEvent(event)
ReplaceText	Supported	li_returnvalue = sle_1.ReplaceText(text)
Resize	Supported	li_returnvalue = sle_1.Resize(li_x,li_y)
SelectedLength	Supported	li_returnvalue = sle_1.SelectedLength()

		If the control is off screen, the execution of the function may have an incorrect return result.
SelectedStart	Supported	li_returnvalue = sle_1.SelectedStart()
		If the control is off screen, the execution of the function may have an incorrect return result.
SelectedText	Supported	ls_returnvalue = sle_1.SelectedText()
		If the control is off screen, the execution of the function may have an incorrect return result.
SelectText	Supported	li_returnvalue = sle_1.SelectText(start, length)
		If the control is off screen, the execution of the function may have an incorrect return result.
SetFocus	Supported	li_returnvalue = sle_1.SetFocus()
		Unsupported: SetFocus(sle_1)
SetPosition	Supported	li_returnvalue = sle_1.SetPosition(position)
SetRedraw	Supported	li_return= sle_1.SetRedraw(FALSE)
Show	Supported	li_returnvalue = sle_1.Show()
TriggerEvent	Supported	li_returnvalue = sle_1.TriggerEvent(event)
TypeOf	Supported	if sle_1.TypeOf() = SingleLineEdit! Then ls_returnvalue = "SingleLineEdit!" end if
Undo	Supported	li_returnvalue = sle_1.Undo()
CanUndo	Unsupported	None
GetContextService	Unsupported	None
Print	Unsupported	None

8.1.1.28 StaticHyperLink control

8.1.1.28.1 Properties for StaticHyperLink control

Property	Support Level	Example Code
Alignment	Supported	shl_1.Alignment = ls_value
BackColor	Supported	shl_1.BackColor = ll_value
		shl_1.BackColor = RGB(192,192,192)
Border	Supported	shl_1.Border = lb_value
BorderColor	Supported	shl_1.BorderColor = ll_value
BorderStyle	Supported	shl_1.BorderStyle = StyleBox!
		shl_1.BorderStyle = StyleLowered!
		shl_1.BorderStyle = StyleRaised!
		Note: StyleShadowBox! is unsupported.
BringToTop	Supported	shl_1.BringToTop = TRUE
DisabledLook	Supported	None
DragAuto	Supported	shl_1.DragAuto = TRUE
DragIcon	Supported	shl_1.DragIcon = 'c:\archive\arrow.ico'
Enabled	Supported	shl_1.Enabled = lb_value
		Note: in PowerBuilder, setting the Enabled property of StaticHyperLink to TRUE or FALSE has no effect. The text always displays as it is.
FaceName	Supported	shl_1.FaceName = ls_value
FillPattern	Supported	shl_1.FillPattern = Diamond!
FocusRectangle	Supported	shl_1.FocusRectangle = TRUE
FontCharSet	Supported	shl_1.FontCharSet = ANSI!
FontFamily	Supported	shl_1.FontFamily = Roman!

FontPitch	Supported	shl_1.FontPitch = Fixed!
Height	Supported	shl_1.Height = li_value
Italic	Supported	shl_1.Italic = lb_value
Pointer	Supported	shl_1.Pointer = 'HourGlass!'
		shl_1.Pointer='d:\archive\IBEAM.BMP.cur'
RightToLeft	Supported	shl_1.RightToLeft = TRUE
TabOrder	Supported	shl_1.TabOrder = 140
Tag	Supported	shl_1.Tag = ls_value
Text	Supported	shl_1.Text = ls_value
TextColor	Supported	shl_1.TextColor = ll_value
TextSize	Supported	shl_1.TextSize = li_value
Underline	Supported	shl_1.Underline = lb_value
URL	Supported	shl_1.URL = " http://www.appeon.com "
Visible	Supported	shl_1.Visible = lb_value
Weight	Supported	shl_1.Weight = li_value
Width	Supported	shl_1.Width = li_value
X	Supported	shl_1.X = li_value
Y	Supported	shl_1.Y = li_value
ClassDefinition	Unsupported	None

8.1.1.28.2 Events for StaticHyperLink control

Event	Support Level	Example Code
-------	---------------	--------------

Clicked	Supported	Clicked()
Constructor	Supported	Constructor()
Destructor	Supported	Destructor()
DoubleClicked	Supported	DoubleClicked()
DragDrop	Supported	DragDrop(source)
DragEnter	Supported	DragEnter(source)
DragLeave	Supported	DragLeave(source)
DragWithin	Supported	DragWithin(source)
GetFocus	Supported	GetFocus()
LoseFocus	Supported	LoseFocus()
RButtonDown	Supported	RButtonDown(flags, xpos, ypos)
Help	Unsupported	None
Other	Unsupported	None

8.1.1.28.3 Functions for StaticHyperLink control

Function	Support Level	Example Code
ClassName	Supported	ls_returnvalue = shl_1.ClassName()
Drag	Supported	shl_1.Drag(End!)
GetParent	Supported	PowerObject lpo_returnvalue lpo_returnvalue = shl_1.GetParent()
Hide	Supported	li_returnvalue = shl_1.Hide()
Move	Supported	li_returnvalue = shl_1.Move(li_x,li_y)

PointerX	Supported	li_returnvalue = shl_1.PointerX()
PointerY	Supported	li_returnvalue = shl_1.PointerY()
PostEvent	Supported	lb_returnvalue = shl_1.PostEvent(event)
Resize	Supported	li_returnvalue = shl_1.Resize(li_x,li_y)
SetFocus	Supported	li_returnvalue = shl_1.SetFocus()
		Unsupported: SetFocus(shl_1)
SetPosition	Supported	li_returnvalue = shl_1.SetPosition(position)
SetRedraw	Supported	shl_1.SetRedraw(FALSE)
Show	Supported	li_returnvalue = shl_1.Show()
TriggerEvent	Supported	li_returnvalue = shl_1.TriggerEvent(event)
TypeOf	Supported	if shl_1.typeof() = StaticHyperLink! Then ls_returnvalue = 'StaticHyperLink!' end if
GetContextService	Unsupported	None
Print	Unsupported	None

8.1.1.29 StaticText control

8.1.1.29.1 Properties for StaticText control

Property	Support Level	Example Code
Alignment	Supported	st_1.Alignment = ls_value
BackColor	Supported	st_1.BackColor = ll_value
		st_1.BackColor = RGB(192,192,192)

Border	Supported	st_1.Border = lb_value
BorderColor	Supported	st_1.BorderColor = ll_value
BorderStyle	Supported	st_1.BorderStyle = StyleBox!
		st_1.BorderStyle = StyleLowered!
		st_1.BorderStyle = StyleRaised!
		Note: The border style cannot be StyleShadowBox!.
BringToTop	Supported	st_1.BringToTop = TRUE
DisabledLook	Supported	None
DragAuto	Supported	st_1.DragAuto = TRUE
DragIcon	Supported	st_1.DragIcon = 'c:\archive\arrow.ico'
Enabled	Supported	st_1.Enabled = lb_value
FaceName	Supported	st_1.FaceName = ls_value
FillPattern	Supported	st_1.FillPattern = Diamond!
FocusRectangle	Supported	st_1.FocusRectangle = TRUE
FontCharSet	Supported	st_1.FontCharSet = ANSI!
FontFamily	Supported	st_1.FontFamily = Roman!
FontPitch	Supported	st_1.FontPitch = Fixed!
Height	Supported	st_1.Height = li_value
Italic	Supported	st_1.Italic = lb_value
Pointer	Supported	st_1.Pointer = 'AppStarting!'
		st_1.Pointer = 'd:\archive\IBeam.BMP.cur'
RightToLeft	Supported	st_1.RightToLeft = TRUE
TabOrder	Supported	st_1.TabOrder = 140

Tag	Supported	st_1.Tag = ls_value
Text	Supported	st_1.Text = ls_value
TextColor	Supported	st_1.TextColor = ll_value
TextSize	Supported	st_1.TextSize = li_value
Underline	Supported	st_1.Underline = lb_value
Visible	Supported	st_1.Visible = lb_value
Weight	Supported	st_1.Weight = li_value
Width	Supported	st_1.Width = li_value
X	Supported	st_1.X = li_value
Y	Supported	st_1.Y = li_value
ClassDefinition	Unsupported	None

8.1.1.29.2 Events for StaticText control

Event	Support Level	Example Code
Clicked	Supported	Clicked()
Constructor	Supported	Constructor()
Destructor	Supported	Destructor()
DoubleClicked	Supported	DoubleClicked()
DragDrop	Supported	DragDrop(source)
DragEnter	Supported	DragEnter(source)
DragLeave	Supported	DragLeave(source)
DragWithin	Supported	DragWithin(source)
GetFocus	Supported	GetFocus()

LoseFocus	Supported	LoseFocus()
RButtonDown	Supported	RButtonDown(flags, xpos, ypos)
Help	Unsupported	None
Other	Unsupported	None

8.1.1.29.3 Functions for StaticText control

Function	Support Level	Example Code
ClassName	Supported	ls_returnvalue = st_1.ClassName()
Drag	Supported	st_1.Drag(Cancel!)
GetParent	Supported	PowerObject lpo_returnvalue lpo_returnvalue = st_1.GetParent()
Hide	Supported	li_returnvalue = st_1.Hide()
Move	Supported	li_returnvalue = st_1.Move(li_x,li_y)
PointerX	Supported	li_returnvalue = st_1.PointerX()
PointerY	Supported	li_returnvalue = st_1.PointerY()
PostEvent	Supported	lb_returnvalue = st_1.PostEvent(event)
Resize	Supported	li_returnvalue = st_1.Resize(li_x,li_y)
SetFocus	Supported	li_returnvalue = st_1.SetFocus()
		Unsupported: SetFocus(st_1)
SetPosition	Supported	li_returnvalue = st_1.SetPosition(position)
Show	Supported	li_returnvalue = st_1.Show()
SetRedraw	Supported	li_returnvalue = st_1.SetRedraw (FALSE)
TriggerEvent	Supported	li_returnvalue = st_1.TriggerEvent(event)

TypeOf	Supported	if st_1.typeof() = StaticText! Then ls_returnvalue = 'StaticText!' end if
GetContextService	Unsupported	None
Print	Unsupported	None

8.1.1.30 Tab control

Differences

- If the MultiLine property of a Tab control is set to True, and the width of the control is exactly the same as the width of all the tabs in it, the tabs may be displayed in two lines on the Web although one line in PowerBuilder. To avoid the difference, specify a width for the Tab control that is bigger than the width of all the tabs, and do more testing till the tabs on the Web can be displayed in the same way as in PowerBuilder.
- In the following scenario, setting the RaggedRight to TRUE will not take effect:
 1. The MultiLine property is disabled.
 2. The MultiLine property is enabled but all tabs can be located in a row.
- In PowerBuilder, if the BoldSelectText property is enabled, only the label of the selected tab page will be displayed in bold type. However, on the Web, the labels of all tabs appear in bold type. Remember to adjust the width of each tab page so that the text can be displayed within the label when enabling this property for the Web.
- In the following scenario, the width of tabs on the Web may be different from in PowerBuilder. When the MultiLine property is enabled and the RaggedRight property is disabled, the largest width of a tab is over W/N (W stands for the total width of all tabs, N stands for the total number of all tabs.)

8.1.1.30.1 Properties for Tab control

Property	Support Level	Example Code
Alignment	Supported	tab_1.Alignment = 'Center!'
BackColor	Supported	tab_1.BackColor = ll_value
		tab_1.BackColor = RGB(192,192,192)
BoldSelectedText	Supported	tab_1.BoldSelectedText = TRUE

BringToTop	Supported	tab_1.BringToTop = TRUE
Control[]	Supported	Userobject luo_1 luo_1 = tab_1.Control[1]
		Note: this property cannot be dynamically changed.
CreateOnDemand	Supported	tab_1.CreateOnDemand = TRUE
DragAuto	Supported	tab_1.DragAuto = True
DragIcon	Supported	tab_1.DragIcon ='c:\archive\arrow.ico'
Enabled	Supported	tab_1.Enabled = FALSE
FaceName	Supported	tab_1.FaceName = ls_value
FixedWidth	Supported	tab_1.FixedWidth = lb_value
FocusOnButtonDown	Supported	tab_1.FocusOnButtonDown = TRUE
FontCharSet	Supported	tab_1.FontCharSet = ANSI!
FontFamily	Supported	tab_1.FontFamily = Roman!
FontPitch	Supported	tab_1.FontPitch = Fixed!
Height	Supported	tab_1.Height = li_value
Italic	Supported	tab_1.Italic = lb_value
MultiLine	Supported	tab_1.Multiline = TRUE
PictureOnRight	Supported	tab_1.PictureOnRight = TRUE
Pointer	Supported	tab_1.Pointer='SizeNWSE!'
		tab_1.Pointer='d:\archive\IBEAM.BMP.cur'
PowerTips	Supported	tab_1.PowerTips = TRUE
RaggedRight	Supported	tab_1.RaggedRight=TRUE
SelectedTab	Supported	tab_1.SelectedTab = li_value

ShowPicture	Supported	tab_1.ShowPicture = TRUE
ShowText	Supported	tab_1.ShowText = lb_value
TabOrder	Supported	tab_1.TabOrder = 150
TabPosition	Supported	tab_1.TabPosition = TabsOnTop!
		<p>The TabPosition property is partially supported: Supported values of the TabPosition property: TabsOnBottom!, TabsOnTop!</p> <p>Unsupported values of the TabPosition property: TabsOnBottomAndTop!, TabsOnLeftAndRight!, TabsOnRightAndLeft!, TabsOnTopAndBottom!, TabsOnLeft!, TabsOnRight!</p>
Tag	Supported	tab_1.Tag = ls_value
TextSize	Supported	tab_1.TextSize = li_value
Underline	Supported	tab_1.Underline = lb_value
Visible	Supported	tab_1.Visible = lb_value
Weight	Supported	tab_1.Weight = li_value
Width	Supported	tab_1.Width = li_value
X	Supported	tab_1.X = li_value
Y	Supported	tab_1.Y = li_value
ClassDefinition	Unsupported	None
PerpendicularText	Unsupported	None

8.1.1.30.2 Events for Tab control

Event	Support Level	Example Code
Clicked	Supported	Clicked(index)

Constructor	Supported	Constructor()
Destructor	Supported	Destructor()
DoubleClicked	Supported	DoubleClicked()
DragDrop	Supported	DragDrop(source,index)
DragEnter	Supported	DragEnter(source)
DragLeave	Supported	DragLeave(source)
DragWithin	Supported	DragWithin(source,index)
GetFocus	Supported	GetFocus()
Key	Supported	Key(key,keyflags)
LoseFocus	Supported	LoseFocus()
RightClicked	Supported	RightClicked(index)
RightDoubleClicked	Supported	RightDoubleClicked(index)
SelectionChanged	Supported	SelectionChanged(oldindex,newindex)
SelectionChanging	Supported	SelectionChanging(oldindex,newindex)
Help	Unsupported	None
Other	Unsupported	None

8.1.1.30.3 Functions for Tab control

Function	Support Level	Example Code
ClassName	Supported	ls_returnvalue = tab_1.ClassName()
CloseTab	Supported	li_returnvalue = tab_1.CloseTab()
Drag	Supported	li_returnvalue = tab_1.Drag()

GetParent	Supported	PowerObject lpo_returnvalue lpo_returnvalue = tab_1.GetParent()
Hide	Supported	li_returnvalue = tab_1.Hide()
Move	Supported	li_returnvalue = tab_1.Move(li_x,li_y)
MoveTab	Supported	li_returnvalue = tab_1.MoveTab(li_x,li_y)
OpenTab	Supported	li_returnvalue = tab_1.OpenTab()
OpenTabWithParm	Supported	li_returnvalue = tab_1.OpenTabWithParm()
PointerX	Supported	li_returnvalue = tab_1.PointerX()
PointerY	Supported	li_returnvalue = tab_1.PointerY()
PostEvent	Supported	li_returnvalue = tab_1.PostEvent(Clicked!)
Resize	Supported	li_returnvalue = tab_1.Resize(li_x,li_y)
SelectTab	Supported	li_returnvalue = tab_1.SelectTab(tabidentifier)
SetFocus	Supported	li_returnvalue = tab_1.SetFocus()
		Unsupported: SetFocus(tab_1)
SetPosition	Supported	li_returnvalue = tab_1.SetPosition(position)
SetRedraw	Supported	li_return= tab_1.SetRedraw(FALSE)
Show	Supported	li_returnvalue = tab_1.Show()
TabTriggerEvent	Supported	li_returnvalue = tab_address.TabTriggerEvent("ue_display")
TriggerEvent	Supported	li_returnvalue = tab_1.TriggerEvent(event)
TypeOf	Supported	if tab_1.typeof() = Tab! Then ls_returnvalue = 'Tab!' End if
GetContextService	Unsupported	None

Print	Unsupported	None
TabPostEvent	Unsupported	None

8.1.1.30.4 Properties for TabPage object

Property	Support Level	Example Code
BackColor	Supported	tab_1.tabpage_1.BackColor = ll_value
Enabled	Supported	tab_1.tabpage_1.BackColor = TRUE
Pointer	Supported	tab_1.tabpage_1.Pointer='No!'
		tab_1.tabpage_1.Pointer ='d:\archive\IBEAM.BMP.cur'
PictureName	Supported	tab_1.tabpage_1.picturename = "Blob!"
PowerTipText	Supported	tab_1.tabpage_1.PowerTipText = ls_value
TabBackColor	Supported	tab_1.tabpage_1.TabBackColor = ll_value
TabTextColor	Supported	tab_1.tabpage_1.TabTextColor = ll_value
Tag	Supported	ls_value = tab_1.tabpage_1.Tag
Text	Supported	ls_value = tab_1.tabpage_1.Text
		On the Web, ASCII characters (~n & ~t) take no effect in the text of the tabpage.
Visible	Supported	tab_1.tabpage_1.Visible = lb_value
		Note: If the Visible property of a TabPage is set to False, the TabPage that gets the first focus in the Tab control would fail to display its content correctly. When it occurs, switching between the tab pages may get the content displayed well.
PictureMaskColor	Unsupported	None

8.1.1.30.5 Events for TabPage object

Event	Support Level	Example Code
Constructor	Supported	Constructor
Destructor	Supported	Destructor
DragDrop	Supported	DragDrop
DragEnter	Supported	DragEnter
DragLeave	Supported	DragLeave
DragWithin	Supported	DragWithin
RButtondown	Supported	RButtonDown
Help	Unsupported	None
Other	Unsupported	None

8.1.1.30.6 Functions for TabPage object

Function	Support Level	Example Code
ClassName	Supported	ls_value = tabpage_1.ClassName()
Drag	Supported	tabpage_1.Drag(Cancel!)
GetParent	Supported	li_returnvalue = tabpage_1.GetParent()
Hide	Supported	li_returnvalue = tabpage_1.Hide()
PostEvent	Supported	lb_returnvalue = tabpage_1.PostEvent(event)
SetRedraw	Supported	li_return= tabpage_1.SetRedraw(True)
Show	Supported	li_returnvalue = tabpage_1.Show()
TriggerEvent	Supported	li_returnvalue = tabpage_1.TriggerEvent(event)

TypeOf	Supported	if tabpage_1.TypeOf() = tabpage! Then ls_returnvalue='tabpage!' end if
CreatePage	Unsupported	None
GetContextService	Unsupported	None
PageCreated	Unsupported	None
PointerX	Unsupported	None
PointerY	Unsupported	None

8.1.1.31 TreeView control

Differences

The length of the converted TreeView labels has minor differences to those in PowerBuilder.

8.1.1.31.1 Properties for TreeView control

Property	Support Level	Example Code
Accelerator	Supported	tv_1.Accelerator = 67
BackColor	Supported	tv_1.BackColor = ll_value
		tv_1.BackColor = RGB(192,192,192)
Border	Supported	tv_1.Border = lb_value
BorderStyle	Supported	tv_1.BorderStyle=StyleRaised!
		tv_1.BorderStyle=StyleBox!
		tv_1.BorderStyle=StyleLowered!
		Note: StyleShadowBox! is not supported.
BringToTop	Supported	tv_1.BringToTop = TRUE

CheckBoxes	Supported	tv_1.CheckBoxes = lb_value
DeleteItems	Supported	tv_1.DeleteItems = lb_value
DisableDragDrop	Supported	tv_1.DisableDragDrop = TRUE
DragAuto	Supported	tv_1.DragAuto = TRUE
DragIcon	Supported	tv_1.DragIcon='c:\archive\arrow.ico'
EditLabels	Supported	tv_1.EditLabels = lb_value
Enabled	Supported	tv_1.Enabled = lb_value
FontCharSet	Supported	tv_1.FontCharSet = ANSI!
FontFamily	Supported	tv_1.FontFamily = Roman!
FontPitch	Supported	tv_1.FontPitch = Fixed!
FaceName	Supported	tv_1.FaceName = ls_value
FullRowSelect	Supported	tv_1.FullRowSelect = TRUE
HasButtons	Supported	tv_1.HasButtons = lb_value
HasLines	Supported	tv_1.HasLines = lb_value
Height	Supported	tv_1.Height = li_value
HideSelection	Supported	tv_1.HideSelection = FALSE
Indent	Supported	tv_1.Indent = 100
Italic	Supported	tv_1.Italic = lb_value
LinesAtRoot	Supported	tv_1.LinesAtRoot = lb_value
PictureHeight	Supported	tv_1.PictureHeight = 16
PictureMaskColor	Supported	tv_1.PictureMaskColor = RGB(255, 255, 0)
PictureName	Supported	tv_1.PictureName = ls_value

		<p>Assigning values to part of an array and using the default values for the other part is not supported. For example:</p> <pre>picturename[1] = "aaa.bmp" picturename[3] = "bbb.bmp" var pic1 = picutrename[1]; var pic2 = picturename[3];</pre>
		<p>A supported example is given as follows:</p> <pre>picturename[1] = "aaa.bmp" picturename[2] = "bbb.bmp" picturename[3] = "ccc.bmp" var pic1 = picutrename[1]; var pic2 = picturename[2]; var pic3 = picturename[3];</pre>
PictureWidth	Supported	tv_1.PictureWidth = 16
Pointer	Supported	tv_1.Pointer ='UPArrow!'
		tv_1.Pointer ='d:\archive\IBEAM.BMP.cur'
SingleExpand	Supported	tv_1.SingleExpand = TRUE
StatePictureHeight	Supported	tv_1.StatePictureHeight = 16
StatePictureMaskColor	Supported	tv_1.StatePictureMaskColor = RGB(255, 255, 0)
StatePictureName[]	Supported	ls_value[] = tv_1.StatePictureName[]
		<p>Assigning values to part of an array and using the default values for the other part is not supported. For example:</p> <pre>statepicturename[1] = "aaa.bmp" statepicturename[3] = "bbb.bmp" var pic1 = statepicutrename[1]; var pic2 = statepicturename[3];</pre>

		A supported example is given as follows: statepicturename[1] = "aaa.bmp" statepicturename[2] = "bbb.bmp" statepicturename[3] = "ccc.bmp" var pic1 = statepicutrename[1]; var pic2 = statepicturename[2]; var pic3 = statepicturename[3];
StatePictureWidth	Supported	tv_1.StatePictureWidth = 16
TabOrder	Supported	tv_1.TabOrder = 160
Tag	Supported	tv_1.Tag = ls_value
TextColor	Supported	tv_1.TextColor = ll_value
TextSize	Supported	tv_1.TextSize = li_value
ToolTips	Supported	tv_1.ToolTips = TRUE
TrackSelect	Supported	tv_1.TrackSelect = TRUE
Underline	Supported	tv_1.Underline = lb_value
Visible	Supported	tv_1.Visible = lb_value
Weight	Supported	tv_1.Weight = li_value
Width	Supported	tv_1.Width = li_value
X	Supported	tv_1.X = li_value
Y	Supported	tv_1.Y = li_value
ClassDefinition	Unsupported	None
SortType	Unsupported	None

8.1.1.31.2 Events for TreeView control

Event	Support Level	Example Code
BeginDrag	Supported	BeginDrag(handle)
BeginLabelEdit	Supported	BeginLabelEdit(handle)
BeginRightDrag	Supported	BeginRightDrag(handle)
Clicked	Supported	Clicked(handle)
Constructor	Supported	Constructor()
DeleteItem	Supported	DeleteItem(handle)
Destructor	Supported	Destructor()
DoubleClicked	Supported	DoubleClicked(handle)
DragDrop	Supported	DragDrop(source,handle)
DragEnter	Supported	DragEnter(source)
DragLeave	Supported	DragLeave(source)
DragWithin	Supported	DragWithin(source,handle)
EndLabelEdit	Supported	EndLabelEdit(handle,newtext)
GetFocus	Supported	GetFocus()
Key	Supported	Key(key,keyflags)
ItemCollapsed	Supported	ItemCollapsed(handle)
ItemCollapsing	Supported	ItemCollapsing(handle)
ItemExpanded	Supported	ItemExpanded(handle)
ItemExpanding	Supported	ItemExpanding(handle)
ItemPopulate	Supported	ItemPopulate(handle)
LoseFocus	Supported	LoseFocus()

RightClicked	Supported	RightClicked(handle)
RightDoubleClicked	Supported	RightDoubleClicked(handle)
SelectionChanged	Supported	SelectionChanged(oldhandle,newhandle)
SelectionChanging	Supported	SelectionChanging(oldhandle,newhandle)
Help	Unsupported	None
Other	Unsupported	None
Sort	Unsupported	None

8.1.1.31.3 Functions for TreeView control

Function	Support Level	Example Code
AddPicture	Supported	li_returnvalue = tv_1.AddPicture(picturename)
AddStatePicture	Supported	li_returnvalue = tv_1.AddStatePicture(picturename)
ClassName	Supported	ls_returnvalue = tv_1.ClassName()
CollapseItem	Supported	li_returnvalue = tv_1.CollapseItem(itemhandle)
DeleteItem	Supported	li_returnvalue = tv_1.DeleteItem(itemhandle)
DeletePicture	Supported	li_returnvalue = tv_1.DeletePicture(index)
DeletePictures	Supported	li_returnvalue = tv_1.DeletePictures()
DeleteStatePicture	Supported	li_returnvalue = tv_1.DeleteStatePicture(index)
DeleteStatePictures	Supported	li_returnvalue = tv_1.DeleteStatePictures()
Drag	Supported	tv_1.Drag(End!)
EditLabel	Supported	li_returnvalue = tv_1.EditLabel(itemhandle)

ExpandAll	Supported	li_returnvalue = tv_1.ExpandAll(itemhandle)
ExpandItem	Supported	li_returnvalue = tv_1.ExpandItem(itemhandle)
FindItem	Supported	ll_returnvalue = tv_1.FindItem(navigationcode, itemhandle)
		Note: the argument <i>navigationcode</i> can only use the following values: RootTreeItem!, NextTreeItem!, PreviousTreeItem!, ParentTreeItem!, ChildTreeItem!, FirstVisibleTreeItem!, NextVisibleTreeItem!, PreviousVisibleTreeItem!, CurrentTreeItem!. The <i>navigationcode</i> value cannot be DropHighlightTreeItem!.
GetItem	Supported	li_returnvalue = tv_1.GetItem(itemhandle,tvi_ret)
GetItemAtPointer	Supported	li_index = tv_1.GetItematPointer ()
GetParent	Supported	lobj_returnvalue = tv_1.GetParent()
Hide	Supported	li_returnvalue = tv_1.Hide()
InsertItem	Supported	ll_returnvalue = tv_1.InsertItem(handleparent,handleafter,item)
		ll_returnvalue = tv_1.InsertItem(handleparent,handleafter,label,pictureindex)
InsertItemFirst	Supported	ll_returnvalue = tv_1.InsertItemFirst(handleparent,item)
		ll_returnvalue = tv_1.InsertItemFirst(handleparent,label,pictureindex)
InsertItemLast	Supported	ll_returnvalue = tv_1.InsertItemLast(handleparent,item)
		ll_returnvalue = tv_1.InsertItemLast(handleparent, label, pictureindex)
InsertItemSort	Supported	tv_1.InsertItemSort(ll_roothandle, l_tvi)
Move	Supported	li_returnvalue = tv_1.Move(li_xpos,li_ypos)
PointerX	Supported	li_returnvalue = tv_1.PointerX()
PointerY	Supported	li_returnvalue = tv_1.PointerY()
PostEvent	Supported	lb_returnvalue = tv_1.PostEvent(event)

Resize	Supported	li_returnvalue = tv_1.Resize(li_width,li_height)
SelectItem	Supported	ll_returnvalue = tv_1.SelectItem(itemhandle)
SetFirstVisible	Supported	li_tvret = tv_1.SetFirstVisible(ll_tvi)
SetFocus	Supported	li_returnvalue = tv_1.SetFocus()
		Unsupported: SetFocus(tv_1)
SetItem	Supported	li_returnvalue = tv_1.SetItem(itemhandle,item)
SetLevelPictures	Supported	tv_1.SetLevelPictures(3, li_level, li_level, & li_level, li_level)
SetOverlayPicture	Supported	tv_1.SetOverlayPicture(1, index)
SetPosition	Supported	li_returnvalue = tv_1.SetPosition(position)
SetRedraw	Supported	li_return= tv_1.SetRedraw(True)
Show	Supported	li_returnvalue = tv_1.Show()
Sort	Supported	li_returnvalue = tv_1.Sort()
SortAll	Supported	tv_1.SortAll(ll_tvi , Ascending!)
TriggerEvent	Supported	li_returnvalue = tv_1.TriggerEvent(event)
TypeOf	Supported	if tv_1.TypeOf() = TreeView! Then ls_returnvalue = 'TreeView!' end if
GetContextService	Unsupported	None
Print	Unsupported	None
SetDropHighlight	Unsupported	None

8.1.1.32 VProgressBar control

8.1.1.32.1 Properties for VProgressBar control

Property	Support Level	Example Code
BringToTop	Supported	<code>vpb_1.BringToTop = TRUE</code>
DragAuto	Supported	<code>vpb_1.DragAuto = TRUE</code>
DragIcon	Supported	<code>vpb_1.DragIcon = 'Question!'</code>
Height	Supported	<code>vpb_1.Height = li_value</code>
MaxPosition	Supported	<code>vpb_1.MaxPosition = 120</code>
MinPosition	Supported	<code>vpb_1.MinPosition = 20</code>
Pointer	Supported	<code>vpb_1.Pointer = 'Beam!'</code>
		<code>vpb_1.Pointer = 'd:\archive\IBEAM.BMP.cur'</code>
Position	Supported	<code>vpb_1.Position = 50</code>
StdStep	Supported	<code>vpb_1.setstep = 20</code>
SmoothScroll	Supported	<code>vpb_1.SmoothScroll = True</code>
TabOrder	Supported	<code>vpb_1.TabOrder = 20</code>
Tag	Supported	<code>vpb_1.Tag = "VProgressBar control"</code>
Visible	Supported	<code>vpb_1.Visible = TRUE</code>
Width	Supported	<code>vpb_1.Width = 700</code>
X	Supported	<code>vpb_1.X = 280</code>
Y	Supported	<code>vpb_1.Y = 1280</code>
ClassDefinition	Unsupported	None

8.1.1.32.2 Events for VProgressBar control

Event	Support Level	Example Code
Clicked	Supported	Clicked(flags,xpos,ypos)
Constructor	Supported	Constructor()
Destructor	Supported	Destructor()
DoubleClicked	Supported	DoubleClicked(flags,xpos,ypos)
DragDrop	Supported	DragDrop(source)
DragEnter	Supported	DragEnter(source)
DragLeave	Supported	DragLeave(source)
DragWithin	Supported	DragWithin(source)
GetFocus	Supported	GetFocus()
LoseFocus	Supported	LoseFocus()
RButtonDown	Supported	RButtonDown(flags,xpos,ypos)
Help	Unsupported	None
Other	Unsupported	None

8.1.1.32.3 Functions for V ProgressBar control

Function	Support Level	Example Code
ClassName	Supported	ls_classnm = vpb_1.classname()
Drag	Supported	li_return = vpb_1.drag(Begin!)
GetParent	Supported	lpo_return = vpb_1.GetParent()
Hide	Supported	li_return = vpb_1.Hide()
Move	Supported	li_return = vpb_1.Move(li_x,li_y)
OffsetPos	Supported	li_return = vpb_1.OffsetPos(10)

PointerX	Supported	li_return = vpb_1.PointerX()
PointerY	Supported	li_return = vpb_1.PointerY()
PostEvent	Supported	vpb_1.PostEvent(Clicked!)
Resize	Supported	li_return = vpb_1.Resize(li_x, li_y)
SetFocus	Supported	li_return = vpb_1.SetFocus()
SetPosition	Supported	li_return = vpb_1.SetPosition(ToTop!)
SetRange	Supported	li_return = vpb_1.SetRange (1, 10)
SetRedraw	Supported	li_return = vpb_1.SetRedraw(FALSE)
Show	Supported	li_return = vpb_1.Show()
StepIt	Supported	li_return = vpb_1.StepIt()
TriggerEvent	Supported	li_return = vpb_1.TriggerEvent(Clicked!)
TypeOf	Supported	type_obj = vpb_1.typeof()
GetContextService	Unsupported	None
Print	Unsupported	None

8.1.1.33 VScrollBar control

8.1.1.33.1 Properties for VScrollBar control

Property	Support Level	Example Code
BringToTop	Supported	vsb_1.BringToTop = TRUE
DragAuto	Supported	vsb_1.DragAuto = TRUE
DragIcon	Supported	vsb_1.DragIcon = 'Question!'

Height	Supported	<code>vsb_1.Height = li_value</code>
MaxPosition	Supported	<code>vsb_1.MaxPosition = 120</code>
MinPosition	Supported	<code>vsb_1.MinPosition = 20</code>
Pointer	Supported	<code>vsb_1.Pointer = 'Beam!'</code>
		<code>vsb_1.Pointer = 'd:\archive\IBEAM.BMP.cur'</code>
Position	Supported	<code>vsb_1.Position = 50</code>
StdWidth	Supported	<code>vsb_1.StdHeight = True</code>
TabOrder	Supported	<code>vsb_1.TabOrder = 20</code>
Tag	Supported	<code>vsb_1.Tag = "VScrollBar control"</code>
Visible	Supported	<code>vsb_1.Visible = TRUE</code>
Width	Supported	<code>vsb_1.Width = 700</code>
X	Supported	<code>vsb_1.X = 280</code>
Y	Supported	<code>vsb_1.Y = 1280</code>
ClassDefinition	Unsupported	None

8.1.1.33.2 Events for VScrollBar control

Event	Support Level	Example Code
Constructor	Supported	<code>Constructor()</code>
Destructor	Supported	<code>Destructor()</code>
DragDrop	Supported	<code>DragDrop(source)</code>
DragEnter	Supported	<code>DragEnter(source)</code>
DragLeave	Supported	<code>DragLeave(source)</code>

DragWithin	Supported	DragWithin(source)
GetFocus	Supported	GetFocus()
LineDown	Supported	LineDown()
LineUp	Supported	LineUp()
LoseFocus	Supported	LoseFocus()
Moved	Supported	Moved(scrollpos)
PageDown	Supported	PageDown()
PageUp	Supported	PageUp()
RButtonDown	Supported	RButtonDown(flags,xpos,ypos)
Help	Unsupported	None
Other	Unsupported	None

8.1.1.33.3 Functions for VScrollBar control

Function	Support Level	Example Code
ClassName	Supported	ls_classnm = vsb_1.classname()
Drag	Supported	li_return = vsb_1.drag(Begin!)
GetParent	Supported	lpo_return = vsb_1.GetParent()
Hide	Supported	li_return = vsb_1.Hide()
Move	Supported	li_return = vsb_1.Move(li_x,li_y)
PointerX	Supported	li_return = vsb_1.PointerX()
PointerY	Supported	li_return = vsb_1.PointerY()
PostEvent	Supported	vsb_1.PostEvent(Clicked!)

Resize	Supported	li_return = vsb_1.Resize(li_x, li_y)
SetFocus	Supported	li_return = vsb_1.SetFocus()
SetPosition	Supported	li_return = vsb_1.SetPosition(ToTop!)
SetRedraw	Supported	li_return = vsb_1.SetRedraw(FALSE)
Show	Supported	li_return = vsb_1.Show()
TriggerEvent	Supported	li_return = vsb_1.TriggerEvent(Clicked!)
TypeOf	Supported	type_obj = vsb_1.typeof()
GetContextService	Unsupported	None
Print	Unsupported	None

8.1.1.34 VTrackBar control

8.1.1.34.1 Properties for VTrackBar control

Property	Support Level	Example Code
BringToTop	Supported	vtb_1.BringToTop = TRUE
DragAuto	Supported	vtb_1.DragAuto = TRUE
DragIcon	Supported	vtb_1.DragIcon = 'Question!'
Height	Supported	vtb_1.Height = li_value
LineSize	Supported	vtb_1.LineSize = li_value
MaxPosition	Supported	vtb_1.MaxPosition = 120
MinPosition	Supported	vtb_1.MinPosition = 20
PageSize	Supported	vtb_1.PageSize = li_value
Pointer	Supported	vtb_1.Pointer = 'Beam!'

		vtb_1.Pointer ='d:\archive\IBEAM.BMP.cur'
Position	Supported	vtb_1.Position = 50
Slider	Supported	vtb_1.Slider = TRUE
SliderSize	Supported	vtb_1.SliderSize = li_value
TabOrder	Supported	vtb_1.TabOrder = 20
Tag	Supported	vtb_1.Tag = ls_tag
TickFrequency	Supported	vtb_1.TickFrequency = li_value
TickMarks	Supported	vtb_1.TickMarks = VTicksOnBottom!
Visible	Supported	vtb_1.Visible = TRUE
Width	Supported	vtb_1.Width = li_value
X	Supported	vtb_1.X = li_x
Y	Supported	vtb_1.Y = li_y
ClassDefinition	Unsupported	None

8.1.1.34.2 Events for VTrackBar control

Event	Support Level	Example Code
Constructor	Supported	Constructor()
Destructor	Supported	Destructor()
DragDrop	Supported	DragDrop(source)
DragEnter	Supported	DragEnter(source)
DragLeave	Supported	DragLeave(source)
DragWithin	Supported	DragWithin(source)
GetFocus	Supported	GetFocus()

LineDown	Supported	LineDown()
LineUp	Supported	LineUp()
LoseFocus	Supported	LoseFocus()
Moved	Supported	Moved(scrollpos)
PageDown	Supported	PageDown()
PageUp	Supported	PageUp()
RButtonDown	Supported	RButtonDown(flags,xpos,ypos)
Help	Unsupported	None
Other	Unsupported	None

8.1.1.34.3 Functions for VTrackBar control

Function	Support Level	Example Code
ClassName	Supported	ls_classnm = vtb_1.classname()
Drag	Supported	li_return = vtb_1.drag(Begin!)
GetParent	Supported	lpo_return = vtb_1.GetParent()
Hide	Supported	li_return = vtb_1.Hide()
Move	Supported	li_return = vtb_1.Move(li_x,li_y)
PointerX	Supported	li_return = vtb_1.PointerX()
PointerY	Supported	li_return = vtb_1.PointerY()
PostEvent	Supported	vtb_1.PostEvent(Clicked!)
Resize	Supported	li_return = vtb_1.Resize(li_x, li_y)
SelectionRange	Supported	li_return = vtb_1.SelectionRange(30, 70)

SetFocus	Supported	li_return = vtb_1.SetFocus()
SetPosition	Supported	li_return = vtb_1.SetPosition(ToTop!)
SetRedraw	Supported	li_return = vtb_1.SetRedraw(FALSE)
Show	Supported	li_return = vtb_1.Show()
TriggerEvent	Supported	li_return = vtb_1.TriggerEvent(Clicked!)
TypeOf	Supported	type_obj = vtb_1.typeof()
GetContextService	Unsupported	None
Print	Unsupported	None

8.2 System objects

Supported

There can only be one Application object in an application.

The following are the system objects that Apeon supports:

Application object	DataStore	Connection object	CORBAObject
DataWindowChild	DragObject	DWObject	DynamicStagingArea object
DynamicDescriptionArea object	Environment object	GraphicObject	Graph object
grAxis object	grDispAttr object	Inet object	InternetResult object
ListViewItem object	MenuCascade object	mailFileDescription object	mailMessage object
mailRecipient object	mailSession object	MDIClient object	Menu object
Message object	NonVisualObject object	OLEObject object	PowerObject

Timing object	Transaction object	TreeViewItem object	UserObject object
Window object	WindowObject		

DragObject, GraphicObject, PowerObject and WindowObject cannot be dynamically created (for example, by using the CREATE statement); they must be defined as static objects, created in PowerBuilder painter.

Unsupported

It is strongly recommended that for a given application, no objects have the same name.

The following are the system objects that Apeon does not support:

ADOResultSet	ArrayBounds	ClassDefinition	ClassDefinitionObject
ConnectionInfo	ConnectObject	ContextInformation	ContextKeyword
CORBABadTypeContext	CORBABadInvorder	CORBABadOperation	CORBABadParam
CORBABadTypeCode	CORBACommFailure	CORBACurrent	CORBADataConversion
CORBAFreeMem	CORBAImpLimit	CORBAInitialize	CORBAInternal
CORBAIntFrePos	CORBAInvalidTransaction	CORBAInvFlag	CORBAInvIdentInvOBJRef
CORBAMarshal	CORBANOImplementation	CORBANOMemory	CORBANOPermission
CORBANOResources	CORBANOResponse	CORBAOBJAdapter	CORBAObjectNoTexist
CORBAPersistStore	CORBASystemException	CORBATransactionRequired	CORBATransactionRollback
CORBATranslent	CORBAUnion	CORBAUnknown	CORBAUserException

CPlusPlus	DivideByZeroError	DWRuntimeError	EnumerationDefinition
EnumerationItemDefinition	Error	ErrorLogging	Exception
ExtObject	JaguarORB	NullObjectError	OLERuntimeError
OLEStorage	OLEStream	OLETxnObject	OMControl
OMCustomControl	OMEmbeddedControl	OMObject	OMStorage
OMStream	ORB	PBTocppObject	Pipeline
ProfileCall	ProfileClass	ProfileLine	ProfileRoutine
Profiling	RemoteObject	ResultSet	ResultSets
RuntimeError	ScriptDefinition	Service	SimpleTypeDefinition
SSLCallBack	SSLServiceProvider	SystemFunctions	Throwable
TraceActivityNode	TraceBeginEnd	TraceError	TraceESQL
TraceFile	TraceGarbageCollect	TraceLine	TraceObject
TraceRoutine	TraceTree	TraceTreeError	TraceTreeESQL
TraceTreeGarbageCollect	TraceTreeLine	TraceTreeNode	TraceTreeObject
TraceTreeRoutine	TraceTreeUser	TraceUser	TransactionServer
Transport	TypeDefinition	VariableCardinalityDefinition	VariableDefinition

8.2.1 Supported objects

8.2.1.1 Application object

Important Requirements

- There can only be one Application object in an application.
- An application cannot be given the same name as a PowerBuilder control type. For example, naming an application "datawindow", "editmask" or "commandbutton" is not supported.
- An application name cannot be a reserved word in JavaScript. See Identifiers for more information.
- A commandline argument in the application Open event is supported.
- It is recommended that the use of Application object is limited to the following well-supported actions:
 1. Launching login screens.
 2. Opening and closing MDI, SDI, Main, Response and Popup windows.
 3. Creating instances of one or more transaction objects.

Additional properties of Application object

- You can specify an icon for an application.
- The type of SQLCA can be a transaction user object.
- The type of SQLSA can be a DynamicStagingArea user object.
- The type of SQLDA can be a DynamicDescriptionArea user object.
- The variable type MESSAGE can use the default value "message" or a user-defined message object, and you can also customize the Message object by defining a class user object inherited from the built-in Message object.

The specification of Error will be ignored.

8.2.1.1.1 Properties for Application object

Property	Support Level	Example Code
AppName	Supported	String ls_AppName
		ls_AppName =app_1.AppName
		//Get the AppName property
DisplayName	Supported	String ls_DisplayName

		ls_DisplayName = app_1.DisplayName
MicroHelpDefault	Supported	app_1.MicroHelpDefault = 'Ready'
RightToLeft	Supported	app_1.RightToLeft = TRUE
ToolbarFrameTitle	Supported	ls_value = app_1.ToolbarFrameTitle
ToolbarSheetTitle	Supported	ls_value = app_1.ToolbarSheetTitle
ToolbarTips	Supported	lb_value = app_1.ToolbarTips
ToolbarText	Supported	lb_value = app_1.ToolbarText
ClassDefinition	Unsupported	None
DDETimeOut	Unsupported	None
DWMessageTitle	Unsupported	None
FreeDBLibraries	Unsupported	None
ToolbarPopupMenuText	Unsupported	None
ToolbarUserControl	Unsupported	None

8.2.1.1.2 Events for Application object

Event	Support Level	Example Code
Close	Supported	Close
Idle	Supported	Idle
Open	Supported	Open
		Note: The <i>commandline</i> argument is supported. The Commandline argument cannot include continuous number signs ("#").
ConnectionBegin	Unsupported	None

ConnectionEnd	Unsupported	None
SystemError	Unsupported	None

8.2.1.1.3 Functions for Application object

Function	Support Level	Example Code
ClassName	Supported	ls_class = l_app.ClassName()
GetParent	Supported	lobj_parent = l_app.GetParent()
PostEvent	Supported	l_app.PostEvent (Clicked!)
		l_app.PostEvent ("Clicked!")
TriggerEvent	Supported	l_app.TriggerEvent (Clicked!)
		l_app.TriggerEvent ("Clicked!")
TypeOf	Supported	string ls_applicationtype if l_app.TypeOf()=Application! Then ls_applicationtype = "Application!" end if
GetContextService	Unsupported	None
SetLibraryList	Unsupported	None
SetTransPool	Unsupported	None

Note: in PowerBuilder 9 and above, GetLibrary and SetLibrary are global functions.

8.2.1.2 Connection object

Important Requirements

Connection object can be created dynamically using the CREATE statement.

8.2.1.2.1 Properties for Connection object

Property	Support Level	Example Code
Application	Supported	ls_name = myconnect.application
Driver	Supported	ls_driver = myconnect.Driver
		Myconnect.Driver = "jaguar"
ErrCode	Supported	ll_rc = myconnect.errcode
ErrText	Supported	ll_rc = myconnect.errtext
Location	Supported	ls_location = myconnect.Location myconnect.location = "192.0.0.217:9000"
UserID	Supported	ls_userid = myconnect.UserID
		Myconnect.UserID = "jagadmin"
Password	Supported	ls_pwd = myconnect.Password
		Myconnect.password = ""
ClassDefinition	Unsupported	None
ConnectionString	Unsupported	None
Handle	Unsupported	None
Options	Unsupported	None
Trace	Unsupported	None

8.2.1.2.2 Events for Connection object

Event	Support Level	Example Code
Constructor	Supported	Constructor
Destructor	Supported	Destructor
Error	Unsupported	None

8.2.1.2.3 Functions for Connection object

Function	Support Level	Example Code
ClassName	Supported	ls_name = Myconnect.ClassName()
ConnectToServer	Supported	ll_rc = Myconnect.ConnectToServer()
		Note: The return value is always 0 on the Web.
CreateInstance	Supported	ll_rc = Myconnect.CreateInstance(ln_tools,"simpletest/nvo_to ols")
DisconnectServer	Supported	ll_rc = Myconnect.disconnectserver()
		Note: The return value is always 0 on the Web.
GetParent	Supported	lobj_parent = Myconnect.GetParent()
PostEvent	Supported	lb_rc = Myconnect.PostEvent(event)
TriggerEvent	Supported	li_rc = Myconnect.TriggerEvent(event)
TypeOf	Supported	IF Myconnect.TypeOf() = Connection! THEN ls_rc = "Connection" END IF
GetContextService	Unsupported	None
GetServerInfo	Unsupported	None
Lookup	Unsupported	None
RemoteStopConnectio n	Unsupported	None
RemoteStopListening	Unsupported	None

8.2.1.3 CORBAObject object

Important Requirements

The CORBAObject object can be created dynamically using the CREATE statement.

8.2.1.3.1 Properties for CORBAObject object

Property	Support Level	Example Code
ClassDefinition	Unsupported	None

8.2.1.3.2 Events for CORBAObject object

Event	Support Level	Example Code
Constructor	Supported	Constructor
Destructor	Supported	Destructor

8.2.1.3.3 Functions for CORBAObject object

Function	Support Level	Example Code
ClassName	Supported	ls_name = my_corbaobj.ClassName()
GetParent	Supported	lobj_parent = my_corbaobj.GetParent()
PostEvent	Supported	lb_rc = my_corbaobj.PostEvent(event)
TriggerEvent	Supported	li_rc = my_corbaobj.TriggerEvent(event)
TypeOf	Supported	IF my_corbaobj.TypeOf() = CorbaObject! THEN ls_rc = "Connection" END IF
_Is_A	Unsupported	None
_Narrow	Unsupported	None
GetContextService	Unsupported	None

8.2.1.4 DynamicDescriptionArea object

8.2.1.4.1 Properties for DynamicDescriptionArea object

Property	Support Level	Example Code
NumInputs	Supported	Integer li_input li_input = SQLDA.NumInputs
NumOutputs	Supported	Integer li_output li_output = SQLDA.NumOutputs
InParmType[]	Supported	if SQLDA.InParmType[3] = TypeInteger! then ...
OutParmType[]	Supported	CHOOSE CASE SQLDA.OutParmType[li_Idx]
		On the Web, the value of this property may differ from that in PowerBuilder.
ClassDefinition	Unsupported	None

8.2.1.4.2 Events for DynamicDescriptionArea object

Event	Support Level	Example Code
Constructor	Supported	Constructor
Destructor	Supported	Destructor

8.2.1.4.3 Functions for DynamicDescriptionArea object

Function	Support Level	Example Code
ClassName	Supported	ls_class = dsa_1.ClassName()
GetDynamicDate	Supported	ls_Value = String(SQLDA.GetDynamicDate(li_Idx))

GetDynamicDateTime	Supported	ls_Value = String(SQLDA.GetDynamicDateTime(li_Idx))
GetDynamicNumber	Supported	ls_Value = String(SQLDA.GetDynamicNumber(li_Idx))
GetDynamicString	Supported	ls_Value = SQLDA.GetDynamicString(li_Idx)
GetDynamicTime	Supported	ls_Value = String(SQLDA.GetDynamicTime(li_Idx))
GetParent	Supported	PowerObject lobj_parent_name
		lobj_parent_name = this.GetParent ()
PostEvent	Supported	this.PostEvent (Clicked!)
		this.PostEvent ("Clicked")
SetDynamicParm	Supported	SQLDA.SetDynamicParm(1,2004)
TriggerEvent	Supported	this.TriggerEvent (Clicked!)
		this.TriggerEvent ("Clicked")
TypeOf	Supported	SQLSA.TypeOf()
GetContextService	Unsupported	None

8.2.1.5 DynamicStagingArea object

Important Requirements

The DynamicStagingArea object can be created dynamically using the CREATE statement.

8.2.1.5.1 Properties for DynamicStagingArea object

Property	Support Level	Example Code
ClassDefinition	Unsupported	

8.2.1.5.2 Events for DynamicStagingArea object

Event	Support Level	Example Code
Constructor	Supported	Constructor
Destructor	Supported	Destructor

8.2.1.5.3 Functions for DynamicStagingArea object

Function	Support Level	Example Code
ClassName	Supported	ls_class = dsa_1.ClassName()
Destructor	Supported	Destructor
Constructor	Supported	Constructor
GetParent	Supported	lobj_parent_name = this.GetParent ()
PostEvent	Supported	this.PostEvent (Clicked!)
		this.PostEvent ("Clicked!")
TriggerEvent	Supported	this.TriggerEvent (Clicked!)
		this.TriggerEvent ("Clicked!")
TypeOf()	Supported	SQLSA.TypeOf()
GetContextService	Unsupported	None

8.2.1.6 Environment object

8.2.1.6.1 Properties for Environment object

Property	Support Level	Example Code
CharSet	Supported	None

CPUType	Supported	None
Language	Supported	None
MachineCode	Supported	None
OSFixesRevision	Supported	None
OSMajorRevision	Supported	None
OSMinorRevision	Supported	None
PBFixesRevision	Supported	None
PBMajorRevision	Supported	None
PBMinorRevision	Supported	None
NumberOfColors	Supported	None
ScreenHeight	Supported	None
ScreenWidth	Supported	None
OSType	Supported	None
PBType	Supported	None
Win16	Supported	None
ClassDefinition	Unsupported	None

8.2.1.6.2 *Functions for Environment object*

Function	Support Level	Example Code
ClassName	Supported	string varname varname = ClassName(gd_double)
TypeOf	Supported	dw_1.Typeof()
GetContextService	Unsupported	None

GetParent	Unsupported	None
-----------	-------------	------

8.2.1.7 Graph object

8.2.1.7.1 Properties for Graph object

Property	Support Level	Example Code
BackColor	Supported	gr_1.BackColor = RGB(255, 255, 0)
Border	Supported	gr_1.Border = TRUE
BorderStyle	Supported	gr_1.BorderStyle = StyleLowered!
BringToTop	Supported	gr_1.BringToTop = TRUE
Category	Supported	gr_1.Category.Label = "Types of Products"
CategorySort	Supported	gr_1.CategorySort = Unsorted!
		Note: If the values of Category are multibyte characters (for example, Chinese characters), they are sorted by internal statement number on the Web.
Depth	Supported	gr_1.Depth = 50
DragAuto	Supported	gr_1.DragAuto = TRUE
DragIcon	Supported	gr_1.DragIcon = 'Question!'
Elevation	Supported	gr_1.Elevation = 35
Enabled	Supported	gr_1.Enabled = TRUE
FocusRectangle	Supported	gr_1.FocusRectangle = TRUE
GraphType	Supported	gr_1.GraphType= AreaGraph! Note: The following graph types are unsupported: Area3D!, Bar3DGraph!, Col3DGraph!, Line3D!, Pie3D!
Height	Supported	gr_1.Height = 750

Legend	Supported	gr_1.Legend = AtBottom!
LegendDispAttr	Supported	None
OverlapPercent	Supported	gr_1.OverlapPercent = 10
Perspective	Supported	gr_1.Perspective = 25
PieDispAttr	Supported	None
Pointer	Supported	gr_1.Pointer = 'Beam!'
		gr_1.Pointer = 'd:\archive\IBEAM.BMP'
		None
Rotation	Supported	gr_1.Rotation = -45
Series	Supported	gr_1.Series.Scaletype = Log10!
SeriesSort	Supported	gr_1.SeriesSort = Unsorted!
		<p>Note:</p> <p>1. The result of executing this property on the Web may differ from that in PowerBuilder. This is because changing other properties may affect the SeriesSort property in PowerBuilder but it does not have such effect on the Web.</p> <p>2. If the values of Category are multibyte characters (for example, Chinese characters), they are sorted by internal statement numbers on the Web.</p>
ShadeColor	Supported	dw_1.Object.gr_1.ShadeColor = 16600000
Spacing	Supported	gr_1.Spacing = 120
TabOrder	Supported	gr_1.TabOrder = 10
Tag	Supported	gr_1.SetMicroHelp(This.Tag)
TextColor	Supported	gr_1.Series.DispAttr.TextColor = RGB(0,0,255)
Title	Supported	gr_1.TitleBar = TRUE
		gr_1.Title = "Monthly Report"

TitleDispAttr	Supported	None
Values	Supported	ls_value = dw_1.Object.gr_1.Values
Visible	Supported	gr_1.Visible = TRUE
Width	Supported	gr_1.Width = 750
X	Supported	gr_1.X = 215
Y	Supported	gr_1.Y= 400
ClassDefinition	Unsupported	None

8.2.1.7.2 Events for Graph object

Event	Support Level	Example Code
Clicked	Supported	Clicked
Constructor	Supported	Constructor
Destructor	Supported	Destructor
DoubleClicked	Supported	DoubleClicked
DragDrop	Supported	DragDrop
DragEnter	Supported	DragEnter
DragLeave	Supported	GragLeave
DragWithin	Supported	DragWithin
GetFocus	Supported	GetFocus
LoseFocus	Supported	LoseFocus
RButtonDown	Supported	RButtonDown
Help	Unsupported	None

Other	Unsupported	None
-------	-------------	------

8.2.1.7.3 Functions for Graph object

Function	Support Level	Example Code
AddCategory	Supported	<code>gr_1.AddCategory("PCs")</code>
AddData	Supported	<code>ll_value = gr_1.FindSeries("Costs")</code>
		<code>gr_1.AddData(ll_value, 12, 3)</code>
AddSeries	Supported	<code>li_value = gr_1.AddSeries("Costs")</code>
CategoryCount	Supported	<code>li_value = gr_1.CategoryCount()</code>
CategoryName	Supported	<code>ls_value = gr_1.CategoryName(5)</code>
ClassName	Supported	<code>ls_value = gr_1.ClassName(gd_double)</code>
Clipboard	Supported	<code>gr_1.Clipboard()</code>
DataCount	Supported	<code>ll_value = gr_1.DataCount("Costs")</code>
DeleteCategory	Supported	<code>gr_1.DeleteCategory(CategName)</code>
DeleteData	Supported	<code>gr_1.DeleteData(SeriesNbr, 7)</code>
DeleteSeries	Supported	<code>gr_1.DeleteSeries(ls_name)</code>
Drag	Supported	<code>gr_1.Drag(Begin!)</code>
FindCategory	Supported	<code>li_value = gr_1.FindCategory(sle_ctgry.Text)</code>
FindSeries	Supported	<code>li_value = gr_1.FindSeries(sle_series.Text)</code>
GetData	Supported	<code>data_value = gr_1.GetData(SeriesNbr, ItemNbr, xValue!)</code>
GetDataPieExplode	Supported	<code>gr_1.GetDataPieExplode(series, datapoint, percentage)</code>

		Note: This function takes effect only in the pie graph. In other graphs the return value of GetDataPieExplode is -1.
GetDataStyle	Supported	ll_color = gr_emp_data.GetDataStyle(SeriesNbr, 6, Foreground!, color_nbr)
		On the Web, the GetDataStyle function returns 1 if the specified data point and its series contain data, otherwise it returns -1. This is different from that in PowerBuilder.
GetDataValue	Supported	rtn = gr_1.GetDataValue(SeriesNbr, ItemNbr, data_value)
GetParent	Supported	luo_value = gr_1.GetParent()
GetSeriesStyle	Supported	gr_1.GetSeriesStyle(SeriesName, & line_style, line_width)
Hide	Supported	gr_1.Hide()
ImportClipboard	Supported	gr_1.ImportClipboard()
		Note: 1) The importtype argument of this function is unsupported. 2) When using this function, you should ensure that the data type imported matches the data type that you specified in a DataWindow control, DataStore object or graph control. Otherwise the output on the Web may be different from that in PowerBuilder. For example, if the date type specified in a DataWindow is string, when you import a file with date number (eg, 2006-08-01), on the Web the date data will be recognized as a string (2006), however in PowerBuilder, it will be recognized as a date data (2006-08-01).
ImportFile	Supported	gr_1.ImportFile("D:\EMPLOYEE.TXT", 2, 30, 3)

		<p>Note: 1) The filename argument of this function can only be a text file (TXT).</p> <p>2) The importtype argument of this function is unsupported.</p> <p>3) When using this function, you should ensure that the data type imported matches the data type that you specified in a DataWindow control, DataStore object or graph control. Otherwise the output on the Web may be different from that in PowerBuilder. For example, if the date type specified in a DataWindow is string, when you import a file with date number (eg, 2006-08-01), on the Web the date data will be recognized as a string (2006), however in PowerBuilder, it will be recognized as a date data (2006-08-01).</p>
ImportString	Supported	<p>gr_1.ImportString(ls_Text, 2, 30, 3)</p> <p>Note: 1) The importtype argument of this function is unsupported.</p> <p>2) When using this function, you should ensure that the data type imported matches the data type that you specified in a DataWindow control, DataStore object or graph control. Otherwise the output on the Web may be different from that in PowerBuilder. For example, if the date type specified in a DataWindow is string, when you import a file with date number (eg, 2006-08-01), on the Web the date data will be recognized as a string (2006), however in PowerBuilder, it will be recognized as a date data (2006-08-01).</p>
InsertCategory	Supported	gr_1.InsertCategory("Macs", CategoryNbr)
InsertData	Supported	<p>gr_1.InsertData(SeriesNbr, CategoryNbr + 1, 1250)</p> <p>Note: The data point will be inserted according to the sorting rules of the Category axis.</p>
InsertSeries	Supported	gr_1.InsertSeries("Costs", SeriesNbr)
ModifyData	Supported	gr_1.ModifyData(SeriesNbr, & CategoryNbr, 1250)
Move	Supported	gr_1.Move(150, 200)
ObjectAtPointer	Supported	Object_type = gr_1.ObjectAtPointer(SeriesNbr, ItemNbr)

PointerX	Supported	li_value = gr_1.PointerX()
PointerY	Supported	li_value = gr_1.PointerY()
PostEvent	Supported	gr_1.PostEvent(Clicked!)
Print	Supported	gr_1.Print(1, "1-5", FALSE, TRUE)
Reset	Supported	gr_1.Reset(Series!)
ResetDataColors	Supported	gr_1.ResetDataColors(SeriesNbr, 10)
Resize	Supported	gr_1.Resize(100, 150)
SaveAs	Supported	gr_1.SaveAs()
SeriesCount	Supported	li_value = gr_1.SeriesCount()
SeriesName	Supported	ls_value = gr_1.SeriesName(5)
SetDataPieExplode	Supported	gr_1.SetDataPieExplode(series, datapoint, percentage)
SetDataStyle	Supported	gr_1.SetDataStyle("gr_depts" , SeriesNbr, 6, Background!, 0)
SetFocus	Supported	gr_1.SetFocus()
SetPosition	Supported	gr_1.SetPosition(ToTop!)
SetRedraw	Supported	gr_1.SetRedraw(FALSE)
SetSeriesStyle	Supported	gr_1.SetSeriesStyle("Costs", & SymbolPlus!)
		Note: When using this function to set the overlay style of Graph DataWindow to "Scatter", it returns true and displays the DataWindow with all data points connected by lines. This is different from that in PowerBuilder.
Show	Supported	gr_1.Show()
TriggerEvent	Supported	gr_1.TriggerEvent(Clicked!)
TypeOf	Supported	gr_1.Typeof()

GetContextService	Unsupported	None
-------------------	-------------	------

8.2.1.8 grAxis object

8.2.1.8.1 Properties for grAxis object

Property	Support Level	Example Code
AutoScale	Supported	<code>gr_1.Values.AutoScale = FALSE</code>
		Note: When this property is set to False, the maximum value for the axis cannot be smaller than the minimum value.
DataType	Supported	<code>gr_1.Values.DataType = AdtDate!</code>
		Notes: 1. On the Web, if the datatype of axis is set to <code>AdtDateTime!</code> , a graph can only display the data for one day. 2. It is unsupported to dynamically modify the <code>GraphType</code> property, if the datatype of the axis will be changed after the modification.
DispAttr	Supported	None
DisplayEveryNLabels	Supported	<code>gr_1.Series.DisplayEveryNLabels = 10</code>
DropLines	Supported	<code>gr_1.Series.DropLines = Dash!</code>
Frame	Supported	<code>gr_1.Series.Frame = Dash!</code>
Label	Supported	<code>gr_1.Values.Label = 'Lawsuits per 1000'</code>
LabelDispAttr	Supported	None
MajorDivisions	Supported	<code>gr_1.Values.MajorDivisions = 10</code>
MajorGridLine	Supported	<code>gr_1.Value.MajorGridLine = Dash!</code>
MajorTic	Supported	<code>gr_1.Values.MajorTic = Straddle!</code>
MaximumValue	Supported	<code>gr_1.Values.DataType = AdtDouble!</code>

		gr_1.Values.MaximumValue = 500000.00
MaxValDateTime	Supported	gr_1.Values.DataType = AdtDate!
		gr_1.Values.MaxValDateTime = 12/31/1999
MinimumValue	Supported	gr_1.Values.DataType = AdtDouble!
		gr_1.Values.MinimumValue = 0.00
MinorDivisions	Supported	gr_1.Values.MinorDivisions = 10
MinorGridLine	Supported	gr_1.Value.MinorGridLine = Dot!
MinorTic	Supported	gr_1.Values.MinorTic = Outside!
MinValDateTime	Supported	gr_1.Values.DataType = AdtDate!
		gr_1.Values.MinValDateTime = 01/31/1900
OriginLine	Supported	gr_1.Values.OriginLine = Dash!
PrimaryLine	Supported	gr_1.Values.PrimaryLine = Dash!
RoundTo	Supported	gr_1.Values.DataType = AdtDate!
		gr_1.Values.RoundToUnit = RndMonths!
		gr_1.Values.RoundTo = 6
RoundToUnit	Supported	gr_1.Values.DataType = AdtDate!
		gr_1.Values.RoundToUnit = RndMonths!
		gr_1.Values.RoundTo = 6
ScaleType	Supported	gr_1.Values.ScaleType=Log10!
		<p>Notes:</p> <ol style="list-style-type: none"> 1. Modifying the ScaleType property for an axis on the Web will affect only the involved axis. This may differ from PowerBuilder. 2. Graphs on the Web will be displayed exactly as the settings of MajorDivisions, MinorDivisions, MaximunValue, and RoundTo. There is no displaying difference between the linear and logarithmic scaling.

ScaleValue	Supported	gr_1.Values.ScaleValue = Actual!
SecondaryLine	Supported	gr_1.Values.SecondaryLine = Dash!
ShadeBackEdge	Supported	gr_1.ShadeColor = RGB(240,250,150)
		gr_1.Category.ShadeBackEdge = TRUE
ClassDefinition	Unsupported	None

8.2.1.8.2 Functions for grAxis object

Function	Support Level	Example Code
ClassName	Supported	gr_1.value.ClassName()
GetParent	Supported	PowerObject lpo_returnvalue lpo_returnvalue = gr_1.value.GetParent()
TypeOf	Supported	If gr_1.value.TypeOf() =grAxis! Then ls_returnvalue = "grAxis!" End If
GetContextService	Unsupported	None

8.2.1.9 grDispAttr object

8.2.1.9.1 Properties for grDispAttr object

Property	Support Level	Example Code
Alignment	Supported	gr_1.Alignment = Center!
AutoSize	Supported	gr_1.TitleDispAttr.AutoSize = FALSE
BackColor	Supported	gr_1.BackColor = RGB(255, 255, 0)
DisplayExpression	Supported	gr_1.TitleDispAttr.DisplayExpression = 'title + " " + Today()'

		<p>Note:</p> <p>When specifying the display expression in the Display Expression field, please note that several items in the Columns box may be unsupported in certain situations:</p> <ol style="list-style-type: none"> 1. (For all graphs) The percentofcategory and The percentofgraph are unsupported. 2. (For all graphs) The percentofseries is unsupported when the text object is set to Value Axis Text or Legend. 3. The categorypercentofgraph and percentofseries are unsupported when the text object is set to Pie Graph Labels and there are multiple series. 4. (For pie graph only) The percentofseries is unsupported when the text object is set to Legend.
Escapement	Supported	<code>gr_1.Value.LabelDispAttr.Escapement = 900</code> When the value is a negative number, the text will not be rotated.
FaceName	Supported	<code>gr_1.FaceName = "Arial"</code>
FillPattern	Supported	<code>gr_1.FillPattern = Diamond!</code>
FontCharSet	Supported	<code>gr_1.FontCharSet = ANSI!</code>
FontFamily	Supported	<code>gr_1.FontFamily = Roman!</code>
FontPitch	Supported	<code>gr_1.FontPitch = Fixed!</code>
Format	Supported	<code>gr__1.Values.DispAttr.Format = "0.00"</code>
Italic	Supported	<code>gr_1.Italic = TRUE</code>
TextColor	Supported	<code>gr_1.Series.DispAttr.TextColor = RGB(0,0,255)</code>
TextSize	Supported	<code>gr_1.Values.LabelDispAttr.TextSize = 12</code>
Underline	Supported	<code>gr_1.Values.LabelDispAttr.Underline = TRUE</code>
Weight	Supported	<code>gr_1.Weight = 700</code>
ClassDefinition	Unsupported	None

8.2.1.9.2 Functions for grDispAttr object

Function	Support Level	Example Code
ClassName	Supported	gr_1.ClassName()
GetParent	Supported	luo_value = gr_1.GetParent()
TypeOf	Supported	If gr_1.TypeOf() = grDispAttr! Then ls_returnvalue = "grDispAttr!" End If
GetContextService	Unsupported	None

8.2.1.10 Inet object**8.2.1.10.1 Properties for Inet object**

Property	Support Level	Example Code
ClassDefinition	Unsupported	None

8.2.1.10.2 Events for Inet object

Event	Support Level	Example Code
Constructor	Supported	Constructor
Destructor	Supported	Destructor

8.2.1.10.3 Functions for Inet object

Function	Support Level	Example Code
ClassName	Supported	inet_1 = CREATE inet li_classnm = inet_1.ClassName()

GetParent	Supported	lpo_return = inet_1.GetParent()
GetURL	Supported	inet_1.GetURL()
HyperLinkToURL	Supported	inet_1.HyperlinkToURL("http://www.appeon.com")
PostEvent	Supported	inet_1.PostEvent(Clicked!)
PostURL	Supported	li_return =Inet_1.PostURL()
TriggerEvent	Supported	inet_1.TriggerEvent(Clicked!)
TypeOf	Supported	inet_1.Typeof()
GetContextService	Supported	GetContextService("Internet", inet_base) inet_base.HyperlinkToURL("http://www.appeon.com") Note: This function is supported only when the service name parameter is "Internet".

8.2.1.11 InternetResult object

8.2.1.11.1 Properties for InternetResult object

Property	Support Level	Example Code
ClassDefinition	Unsupported	None

8.2.1.11.2 Events for InternetResult object

Event	Support Level	Example Code
Constructor	Supported	Constructor
Destructor	Supported	Destructor

8.2.1.11.3 Functions for InternetResult object

Function	Support Level	Example Code
ClassName	Supported	internetresult_1 = Create InternetResult li_classnm= internetresult_1.ClassName()
GetParent	Supported	lpo_return = internetresult_1.GetParent()
InternetData	Supported	li_return = internetresult_1.InternetData (data)
InternetStatus	Supported	(Not used)
PostEvent	Supported	Internetresult_1.PostEvent(Clicked!)
TriggerEvent	Supported	Internetresult_1.TriggerEvent(Clicked!)
TypeOf	Supported	Internetresult_1.Typeof()
GetContextService	Unsupported	None

8.2.1.12 ListViewItem object

8.2.1.12.1 Properties for ListViewItem object

Property	Support Level	Example Code
CutHighlighted	Supported	lvi_item1.CutHighlighted = True
Data	Supported	any_value = lvi_item1.Data
DropHighlighted	Supported	lvi_item1.DropHighlighted = True
HasFocus	Supported	lvi_item1.HasFocus = True
ItemX	Supported	None
ItemY	Supported	None

Label	Supported	ls_value = lvi_item1.Label In PowerBuilder, if the string label is too long to display in one line, the string will display in multiple lines. However, in the Web application, the string will only display in one line.
OverlayPictureIndex	Supported	li_value = lvi_item1.OverlayPictureIndex
PictureIndex	Supported	li_value = lvi_item1.PictureIndex
Selected	Supported	lb_value = lvi_item1.Selected
StatePictureIndex	Supported	li_value = lvi_item1.StatePictureIndex
ClassDefinition	Unsupported	None

8.2.1.12.2 Functions for ListViewItem object

Function	Support Level	Example Code
ClassName	Supported	ls_objectname = lvi_item1.ClassName()
TypeOf	Supported	IF lvi_item1.TypeOf() = ListViewItem! THEN ls_returnvalue = "ListViewItem" END IF
GetContextService	Unsupported	None
GetParent	Unsupported	None

8.2.1.13 MDIClient object

Important Requirements

- Multiple MDI windows are supported.
- Defining the size of the client area in a MDI window is supported.
- If there are multiple MDI frames, the second frame window of the Web application will be loaded at a much slower speed up to four times slower than the first one. If the first MDI is closed, the second one will not function.

- When the MDI frame is closed in an Apeon-deployed application, the Web browser will also close automatically.

8.2.1.13.1 Properties for MDIClient object

Property	Support Level	Example Code
BackColor	Supported	w_main.mdi_1.BackColor = ll_value
Height	Supported	w_main.mdi_1.Height = li_value
MicroHelpHeight	Supported	li_return = w_main.mdi_1.MicroHelpHeight
Tag	Supported	w_main.mdi_1.Tag = ls_value
Width	Supported	w_main.mdi_1.Width = li_value
X	Supported	w_main.mdi_1.X = li_value
Y	Supported	w_main.mdi_1.Y = li_value
BringToTop	Unsupported	None
ClassDefinition	Unsupported	None
Visible	Unsupported	None

8.2.1.13.2 Functions for MDIClient object

Function	Support Level	Example Code
ClassName	Supported	ls_classname = w_main.mdi_1.Classname()
GetParent	Supported	PowerObject lpo_returnvalue
		lpo_returnvalue = this.GetParent()
Hide	Supported	w_main.Hide()
Move	Supported	w_main.Move(150,200)
Resize	Supported	w_main.Resize(100, 150)

SetRedraw	Supported	w_main.SetRedraw(FALSE)
Show	Supported	m_main.Show()
TypeOf	Supported	ls_type = w_main.mdi_1.TypeOf()
GetContextService	Unsupported	None

8.2.1.14 mailFileDescription object

8.2.1.14.1 Properties for mailFileDescription object

Property	Support Level	Example Code
FileType	Supported	mailFileDescription mattach
		mAttach.FileType = mailattach!
Filename	Supported	mAttach.Filename = ls_filename
Pathname	Supported	mAttach.Pathname = ls_pathname
Position	Supported	mAttach.Position = ll_position
ClassDefinition	Unsupported	None

8.2.1.14.2 Functions for mailFileDescription object

Function	Support Level	Example Code
ClassName	Supported	ls_classnm = mAttach.classname()
TypeOf	Supported	type_obj = mAttach.typeof()
GetContextService	Unsupported	None
GetParent	Unsupported	None

8.2.1.15 mailMessage object

8.2.1.15.1 Properties for mailMessage object

Property	Support Level	Example Code
AttachmentFile[]	Supported	mMsg.AttachmentFile[1] = mAttach
ConversationID	Supported	ls_ID = mMsg.ConversationID
DateReceived	Supported	mMsg.DateReceived = ls_today
MessageSent	Supported	lb_sent = mMsg.MessageSent
MessageType	Supported	ls_type = mMsg.MessageType
NoteText	Supported	ls_note = mMsg.NoteText
ReceiptRequested	Supported	lb_return = mMsg.ReceiptRequested
Recipient[]	Supported	mailRecipient recip
		mMsg.Recipient[1] = recip
Subject	Supported	mMsg.subject = ls_subject
Unread	Supported	mMsg.Unread = false
ClassDefinition	Unsupported	None

8.2.1.15.2 Functions for mailMessage object

Function	Support Level	Example Code
ClassName	Supported	ls_classnm = mMsg.classname()
TypeOf	Supported	type_obj = mMsg.typeof()
GetContextService	Unsupported	None
GetParent	Unsupported	None

8.2.1.16 mailRecipient object**8.2.1.16.1 Properties for mailRecipient object**

Property	Support Level	Example Code
Address	Supported	mRecip.Address = ls_address
EntryID	Supported	ls_return = string(mRecip.EntryID)
Name	Supported	mRecip.Name = ls_name
RecipientType	Supported	mRecip.RecipientType = mailTo!

8.2.1.16.2 Functions for mailRecipient object

Function	Support Level	Example Code
ClassName	Supported	ls_classnm = mRecip.classname()
TypeOf	Supported	type_obj = mRecip.typeof()
GetContextService	Unsupported	None
GetParent	Unsupported	None

8.2.1.17 mailSession object**8.2.1.17.1 Properties for mailSession object**

Property	Support Level	Example Code
MessageID[]	Supported	ls_message = mSes.MessageID[li_i]
SessionID	Supported	ll_sessionid = mSes.SessionID

ClassDefinition	Unsupported	None
-----------------	-------------	------

8.2.1.17.2 Events for mailSession object

Event	Support Level	Example Code
Constructor	Supported	Constructor
Destructor	Supported	Destructor

8.2.1.17.3 Functions for mailSession object

Function	Support Level	Example Code
ClassName	Supported	ls_classnm = mSes.classname()
GetParent	Supported	lobj_parent = mSes.GetParent
mailAddress	Supported	mRet = mSes.mailAddress()
mailDeleteMessage	Supported	mRet = mSes.mailDeleteMessage(sID)
mailGetMessages	Supported	mRet = mSes.mailGetMessages()
mailHandle	Supported	mHandle = mSes.mailHandle()
mailLogoff	Supported	mRet = mSes.mailLogoff()
mailLogon	Supported	mRet = mSes.mailLogon ()
mailReadMessage	Supported	mRet = mSes.mailReadMessage(sMessageID, mMsg , mailEntireMessage!, TRUE)
mailRecipientDetails	Supported	mRet = mSes.mailRecipientDetails(mMsg.Recipient[1],TRUE)
mailResolveRecipient	Supported	mRet = mSes.mailResolveRecipient(mRecip,true)
mailSaveMessage	Supported	mRet = mSes.mailSaveMessage(mMsg.ConversationID,

		mMsg)
mailSend	Supported	mRet = mSes.mailsend (mMsg)
PostEvent	Supported	lb_return = mSes.PostEvent("ue_user")
TriggerEvent	Supported	lb_return = mSes.TriggerEvent("ue_user")
TypeOf	Supported	type_obj = mSes.typeof()
GetContextService	Unsupported	None

8.2.1.18 Timing object

8.2.1.18.1 Properties for Timing object

Property	Support Level	Example Code
Interval	Supported	ld_return = timing_1.interval
Running	Supported	lb_return = timing_1.running
ClassDefinition	Unsupported	None

8.2.1.18.2 Events for Timing object

Event	Support Level	Example Code
Constructor	Supported	Constructor
Destructor	Supported	Destructor
Timer	Supported	Timer(10, w_main)

8.2.1.18.3 Functions for Timing object

Function	Support Level	Example Code
----------	---------------	--------------

ClassName	Supported	timing_1 = CREATE uo_timer
		ls_classnm = timing_1.ClassName()
GetParent	Supported	lobj_parent_name = timing_1.GetParent()
PostEvent	Supported	timing_1.PostEvent(Clicked!)
Start	Supported	timing_1.Start(60)
Stop	Supported	timing_1.Stop()
TriggerEvent	Supported	timing_1.TriggerEvent(Clicked!)
TypeOf	Supported	timing_1.Typeof()
GetContextService	Unsupported	None

8.2.1.19 Menu object

Important Requirements

- Menus can be dynamically created (for example, using the CREATE statement).
- When a menu is dynamically created, it is unsupported to create multiple items to the menu at one time.
- Use of Main menu should be limited to no more than five layers of submenus for a single menu item.
- More levels of dropdown menus under a menu bar item are supported.
- Apeon supports user-defined menu events and functions.

8.2.1.19.1 Properties for Menu object

Property	Support Level	Example Code
Checked	Supported	m_function.m_testitemforfunction.Checked = lb_value
Default	Supported	m_function.m_testitemforfunction.Default = TRUE
Enabled	Supported	m_function.m_testitemforfunction.Enabled = lb_value

Item []	Supported	Menu lmenu_item[] lmenu_item = m_menu.Item[]
		Support using Item[] in different ways:
		lmenu_onemenu = m_main.Item[1].Item[1] m_main.Item[1].Item[1].text = "open a Window" m_main.Item[1].Item[1].triggerevent(clicked)
		Dynamically adding menu items with Item[] is supported.
MicroHelp	Supported	m_function.m_testitemforfunction.MicroHelp = ls_value
ParentWindow	Supported	w_value = m_function.Parentwindow
ShiftToRight	Supported	m_function.m_testitemforfunction.ShiftToRight = true
Shortcut	Supported	None
Tag	Supported	m_function.m_testitemforfunction.Tag = ls_value
ToolbarItemBarIndex	Supported	m_function.m_testitemforfunction.ToolbarItemBarIndex = li_value
		Note: If the property ToolbarItemBarIndex is set to 0, the toolbar does not display in PowerBuilder but displays on the Web.
ToolbarItemDown	Supported	m_function.m_testitemforfunction.ToolbarItemDown = true
ToolbarItemDown Name	Supported	m_function.m_testitemforfunction.ToolbarItemDownName = ls_value
ToolbarItemName	Supported	m_function.m_testitemforfunction.ToolbarItemName = ls_value
ToolbarItemOrder	Supported	m_function.m_testitemforfunction.ToolbarItemOrder = li_value
ToolbarItemSpace	Supported	m_function.m_testitemforfunction.ToolBarItemSpace = 5
ToolbarItemText	Supported	m_function.m_testitemforfunction.ToolbarItemText = ls_value
ToolbarItemVisible	Supported	m_function.m_testitemforfunction.ToolbarItemVisible = lb_value

Visible	Supported	<code>m_function.m_testitemforfunction.Visible = lb_value</code>
ClassDefinition	Unsupported	None
MenuItemType	Unsupported	None
MergeOption	Unsupported	None

8.2.1.19.2 Events for Menu object

Event	Support Level	Example Code
Clicked	Supported	Clicked
		Note: The Clicked event cannot be triggered for a menu item that has a submenu.
Selected	Supported	Selected
Help	Unsupported	None

8.2.1.19.3 Functions for Menu object

Function	Support Level	Example Code
Check	Supported	<code>ll_returnvalue = m_function.m_testitemforfunction.Check()</code>
ClassName	Supported	<code>ls_returnvalue = m_function.m_testitemforfunction.ClassName()</code>
Disable	Supported	<code>ll_returnvalue = m_function.m_testitemforfunction.Disable()</code>
Enable	Supported	<code>ll_returnvalue = m_function.m_testitemforfunction.Enable()</code>
GetParent	Supported	<code>lobj_returnvalue = m_function.GetParent()</code>
Hide	Supported	<code>ll_returnvalue = m_function.m_testitemforfunction.Hide()</code>
PopupMenu	Supported	<code>ll_returnvalue =</code>

		<code>m_function.m_testitemforfunction.PopMenu(xlocation, ylocation)</code>
PostEvent	Supported	<code>ll_returnvalue = m_function.m_testitemforfunction.PostEvent(event)</code>
Show	Supported	<code>ll_returnvalue = m_function.m_testitemforfunction.Show()</code>
TriggerEvent	Supported	<code>ll_returnvalue = m_function.m_testitemforfunction.TriggerEvent(event)</code>
TypeOf	Supported	<pre>If m_function.m_testitemforfunction.TypeOf() = menu! Then ls_returnvalue = "menu" end if</pre>
Uncheck	Supported	<code>ll_returnvalue = m_function.m_testitemforfunction.Uncheck()</code>
GetContextService	Unsupported	None

8.2.1.20 Menu features

Supported

- Menu object may be implemented with the following functionalities:
 1. Modifying menu items based on user rights using the enable and visible properties.
 2. User-defined functions in menu object.
- It is supported to have a menu object as a structure variable, or define a structure/declare a structure instance in a menu object.
- It is supported to declare a menu type array.
- It is supported to qualify the item of a child menu with the class name of the parent menu. For example, `m_city` is the menu for the window `w_citysheet`, and is inherited from `m_main`. `m_file` is an item of the menu `m_city`.
The following two lines of code have the same effect on the Web:

```
m_main.m_file.enabled = FALSE
m_city.m_file.enabled = FALSE
```
- It is supported to declare a menu variable, assign value to it, and pass the variable as argument. For example:

```
menu lm_onemenu
lm_onemenu = create menu
lm_onemenu.text = "Open a window" //or
```

```
lm_ onemenu = m_main.m_file //or  
lm_ onemenu = m_main.Item[1].Item[1]
```

A window object can use the ChangeMenu function to dynamically change the menu. See the ChangeMenu functions for [Window object](#).

- Switching between different menus is allowed. However, only one menu can be displayed at one time. For example, the MDI window menu can be dynamically refreshed when a sheet is opened or closed.
- Append the names of open sheets (window list) to a menu item.
- Using the UpArrow/DownArrow/LeftArrow/RightArrow to change the currently selected menu item is supported.
- Shortcut keys for menu items are supported.

Unsupported

- In PowerBuilder, the menu items always show in front of main window or popup window. In Web applications, the menu items may get hidden behind a main window or a popup window.
- Different from PowerBuilder as follows: In the case of multiple Web windows loading the same menu, when a window (window A) accesses the menu object, the menu object accessed is not the menu in the window A, but the menu in the window that is most recently opened.
- It is not supported to access the menu of the window that currently does not have focus.

8.2.1.20.1 Toolbar features

Supported

- The toolbar item synchronizes with the relevant menu item. If the property of the menu item changes, the property of the toolbar item changes accordingly.
- Multiple toolbars are supported.
- Multiple toolbars can be displayed at the same time, but docked toolbar or toolbar toggle are not supported.
- Toolbar can only be displayed in the MDI and MDIHelp windows.

Unsupported

- In Web applications, the toolbars are always placed directly below the menu. The Web toolbar cannot be dragged around or placed at any other place.
- In PowerBuilder, if the user right mouse clicks over a toolbar, a popup menu is displayed with the available options, for example: Left, Right, Floating, Show Text, Show PowerTips. The popup menu will not display in Web applications. The toolbar item cannot show text but can show PowerTips.
- Clicking the toolbar item will not trigger the Selected event for Menu object.
- If the ShiftToRight property is set to TRUE, menu objects may shift to right and toolbar items will automatically adjust its places according to the menu changes in

PowerBuilder. However, toolbar items will not automatically adjust its places on the Web.

8.2.1.21 MenuCascade object

Important requirements

- It is unsupported to place a cascading menu in another cascading menu.
- When you extend a cascading menu, the first dropdown item automatically get focus on the Web. Pressing the blank key can trigger the Click event of the item with focus and the focus can be shifted between items if you pressing the Up and Down arrow keys.
- Changes to a Web menu will be updated automatically without calling the Hide and Show functions for the Menu.

8.2.1.21.1 Properties for MenuCascade object

Property	Support Level	Example Code
Checked	Supported	<code>m_function.m_testitemforfunction.Checked = lb_value</code>
Column	Supported	<code>m_menu.Columns = li_count</code>
Default	Supported	<code>m_menu.Default = TRUE</code>
DropDown	Supported	<code>m_menu.DropDown = False</code>
Enabled	Supported	<code>m_function.m_testitemforfunction.Enabled = lb_value</code>
Item []	Supported	<code>lmenu_item = m_menu.Item[]</code>
MicroHelp	Supported	<code>m_function.m_testitemforfunction.MicroHelp = ls_value</code>
ParentWindow	Supported	<code>w_value = m_function.Parentwindow</code>
ShiftToRight	Supported	<code>m_menu.ShiftToRight = True</code>
Shortcut	Supported	<code>li_retuen = m_menu.Shortcut</code>
Tag	Supported	<code>m_function.m_testitemforfunction.Tag = ls_value</code>
Text	Supported	<code>m_function.m_testitemforfunction.Text = ls_value</code>

ToolBarItemDown	Supported	m_function.m_testitemforfunction.ToolBarItemDown = lb_flag
ToolBarItemDownName	Supported	m_function.m_testitemforfunction.ToolBarItemDownName = ls_value
ToolBarItemBarIndex	Supported	m_function.m_testitemforfunction.ToolBarItemBarIndex = li_value
		Note: If the property ToolBarItemBarIndex is set to 0, the toolbar does not display in PowerBuilder but displays on the Web.
ToolBarItemName	Supported	m_function.m_testitemforfunction.ToolBarItemName = ls_value
ToolBarItemOrder	Supported	m_function.m_testitemforfunction.ToolBarItemOrder = li_value
		Note: the value of ToolBarItemOrder cannot be a negative number.
ToolBarItemText	Supported	m_function.m_testitemforfunction.ToolBarItemText = ls_value
ToolBarItemVisible	Supported	m_function.m_testitemforfunction.ToolBarItemVisible = lb_value
Visible	Supported	m_function.m_testitemforfunction.Visible = lb_value
ClassDefinition	Unsupported	None
CurrentItem	Unsupported	None
MenuItemType	Unsupported	None
MergeOption	Unsupported	None

8.2.1.21.2 Events for MenuCascade object

Event	Support Level	Example Code
-------	---------------	--------------

Clicked	Supported	Clicked
		Note: The Clicked event cannot be triggered for a menu item that has a submenu.
Selected	Supported	Selected
Help	Unsupported	None

8.2.1.21.3 Functions for MenuCascade object

Function	Support Level	Example Code
Check	Supported	ll_returnvalue = m_function.m_testitemforfunction.Check()
ClassName	Supported	ls_returnvalue = m_function.m_testitemforfunction.ClassName()
Disable	Supported	ll_returnvalue = m_function.m_testitemforfunction.Disable()
Enable	Supported	ll_returnvalue = m_function.m_testitemforfunction.Enable()
GetParent	Supported	lobj_returnvalue = m_function.GetParent()
Hide	Supported	ll_returnvalue = m_function.m_testitemforfunction.Hide()
PopupMenu	Supported	ll_returnvalue = m_function.m_testitemforfunction.PopupMenu(xlocation, ylocation)
PostEvent	Supported	ll_returnvalue = m_function.m_testitemforfunction.PostEvent(event)
Show	Supported	ll_returnvalue = m_function.m_testitemforfunction.Show()
TriggerEvent	Supported	ll_returnvalue = m_function.m_testitemforfunction.TriggerEvent(event)

TypeOf	Supported	If m_function.m_testitemforfunction.TypeOf() = menu! Then ls_returnvalue = "menu" end if
Uncheck	Supported	ll_returnvalue = m_function.m_testitemforfunction.Uncheck()
GetContextService	Unsupported	None

8.2.1.22 Message object

Important Requirements

Both the system message and the user-defined message are supported.

8.2.1.22.1 Properties for Message object

Property	Support Level	Example Code
DoubleParm	Supported	OpenWithParm(w_1, ld_value) ld_value = Message.DoubleParm
LongParm	Supported	ls_value = Message.LongParm
PowerObjectParm	Supported	OpenWithParm(w_1, s_value) ls_value = Message.PowerObjectParm
Processed	Supported	None
StringParm	Supported	OpenWithParm(w_1, ls_value) ls_value = Message.StringParm
WordParm	Supported	None
ClassDefinition	Unsupported	None
Handle	Unsupported	None
Number	Unsupported	None

ReturnValue	Unsupported	None
-------------	-------------	------

8.2.1.22.2 Events for Message object

Event	Support Level	Example Code
Constructor	Supported	None
Destructor	Supported	None

8.2.1.22.3 Functions for Message object

Function	Support Level	Example Code
ClassName	Supported	string varname varname = ClassName(gd_double)
GetParent	Supported	lobj_parent = cb_1.GetParent()
PostEvent	Supported	cb_1.PostEvent(Clicked!)
TriggerEvent	Supported	cb_1.TriggerEvent(Clicked!)
TypeOf	Supported	dw_1.Typeof()
GetContextService	Unsupported	None

8.2.1.23 NonVisualObject object

8.2.1.23.1 Properties for NonVisualObject object

Property	Support Level	Example Code
ClassDefinition	Unsupported	None

8.2.1.23.2 Events for NonVisualObject object

Event	Support Level	Example Code
Constructor	Supported	Constructor
Destructor	Supported	Destructor

8.2.1.23.3 Functions for NonVisualObject object

Function	Support Level	Example Code
ClassName	Supported	nvo_test.classname()
GetParent	Supported	lobj_parent = nvo_test.GetParent()
PostEvent	Supported	nvo_test.PostEvent(Constructor!)
TriggerEvent	Supported	nvo_test.TriggerEvent(Constructor!)
TypeOf	Supported	If nvo_test.typeof()=nonvisualobject! Then ls_value='nonvisualobject!' End If
GetContextService	Unsupported	None

8.2.1.24 OLEObject object**Unsupported**

The shortcut key for OCX is unsupported.

It is unsupported to have blob, DateTime, Date, or Time parameters in functions of OLEObject object.

The datatype of the property value cannot be Datetime, Date, or Time.

8.2.1.24.1 Properties for OLEObject object

Properties	Support Level	Example Code
------------	---------------	--------------

ClassDefinition	Unsupported	None
Handle	Unsupported	None

8.2.1.24.2 Events for OLEObject object

Event	Support Level	Example Code
Constructor	Supported	Constructor
Destructor	Supported	Destructor
Error	Unsupported	None
ExternalException	Unsupported	None

8.2.1.24.3 Functions for OLEObject object

Function	Support Level	Coding Examples
ClassName	Supported	ls_return = ole_1.ClassName()
ConnectToNewObject	Supported	OLEObject myoleobject myoleobject = CREATE OLEObject li_return = myoleobject.ConnectToNewObject("excel.application")
ConnectToObject	Supported	OLEObject myoleobject myoleobject = CREATE OLEObject li_return = myoleobject.ConnectToObject("c:\excel\expense.xls")
DisconnectObject	Supported	li_return = myoleobject.DisconnectObject()
GetAutomationNativePointer	Supported	UnsignedLong lul_oleptr li_return = ocx_spell.GetNativePointer(lul_oleptr)

IsAlive	Supported	IF ole_1.IsAlive() THEN return
PostEvent	Supported	ole_1.PostEvent("Error")
ReleaseAutomationNativePointer	Supported	li_return = ocx_spell.ReleaseNativePointer(lul_oleptr)
SetAutomationPointer	Supported	li_return = oleChild.SetAutomationPointer(myoleobject)
TriggerEvent	Supported	li_return = ole_1.TriggerEvent("Error")
TypeOf	Supported	None
ConnectToNewRemoteObject	Unsupported	None
ConnectToRemoteObject	Unsupported	None
GetContextService	Unsupported	None
GetParent	Unsupported	None
SetAutomationLocale	Unsupported	None
SetAutomationTimeout	Unsupported	None

8.2.1.25 Transaction object

Important Requirements

- The Transaction object can be global or local.
- The user-defined Transaction object is supported.
- The type of SQLCA can be transaction or transaction user object. It is supported if the user defines a transaction user object (for example, u_trans) and sets the type of SQLCA to the transaction user object (for example, u_trans).
- An application can have multiple Transactions Objects, and the Transaction Objects can be connected to different type of databases.
- The specification of isolation level for a database profile in PowerBuilder is not supported.

8.2.1.25.1 Properties for Transaction object

Appeon ignores the dynamic property settings for the connection cache and Transaction Object except for the AutoCommit property.

Property	Support Level	Example Code
AutoCommit	Supported	sqlca.AutoCommit = lb_value
		Notes: 1). If the database server is Oracle and the driver is native driver, the AutoCommit setting is always interpreted as FALSE. 2). The AutoCommit property is the only property that may be dynamically modified.
Database	Supported	ls_value = sqlca.Database
DBMS	Supported	ls_value = sqlca.DBMS
DBPass	Supported	None
DBParm	Supported	i_sqlca.DBParm= "ConnectionString= 'DSN=AppeonSample;UID=dba;PWD=sql!'" //DBMS-specific parameters
LogID	Supported	ls_value = sqlca.LogID
LogPass	Supported	ls_value = sqlca.LogPass
ServerName	Supported	ls_value = sqlca.ServerName
SQLCode	Supported	ll_value = sqlca.SQLCode
SQLDBCCode	Supported	ll_value = sqlca.SQLDBCCode
		Notes: 1). In PowerBuilder, if sqlca.SQLCode=100, sqlca.SQLDBCCode = 3. In Appeon, when sqlca.SQLCode=100, sqlca.SQLDBCCode = 0. 2). In the Web application, SQLDBCCode returns a different value from PowerBuilder. It is recommended not to use SQLDBCCode in statements such as IF...ELSE.

SQLErrText	Supported	ls_value = sqlca.SQLErrText
SQLNRows	Supported	ll_value = sqlca.SQLNRows
UserID	Supported	ls_value = sqlca.UserID
ClassDefinition	Unsupported	None
Lock	Unsupported	None
SQLReturnData	Unsupported	None

8.2.1.25.2 Events for Transaction object

Event	Support Level	Example Code
Constructor	Supported	Constructor
Destructor	Supported	Destructor

8.2.1.25.3 Functions for Transaction object

Function	Support Level	Example Code
ClassName	Supported	ls_returnvalue = sqlca.ClassName()
TypeOf	Supported	if this.TypeOf() = transaction! Then ls_returnvalue = "transaction!" end if
DBHandle	Supported	None
PostEvent	Supported	None
SyntaxFromSQL	Supported	ls_dw_syntax = SQLCA.SyntaxFromSQL(ls_sql_syntax, ls_style, ls_dw_err)

		<p>Notes:</p> <ol style="list-style-type: none"> 1. The following syntax is unsupported: ls_dw_syntax = SyntaxFromSQL(sqlca, ls_sql_syntax, ls_style, ls_dw_err) 2. The data type of computed columns in SyntaxFromSQL cannot be the Apeon unsupported data type. 3. The length of return value on the Web is different from PowerBuilder 4. The return value of the SyntaxFromSQL function on the Web contains a column name for each column although it is not in PowerBuilder. 5. The function cannot generate correct source code for TreeView DataWindow.
TriggerEvent	Supported	None
GetContextService	Unsupported	None
GetParent	Unsupported	None

8.2.1.26 TreeViewItem object

8.2.1.26.1 Properties for TreeViewItem object

Property	Support Level	Example Code
Bold	Supported	tv_treeviewitem.Bold = TRUE
Children	Supported	tv_treeviewitem.Children = FALSE
Data	Supported	tv_treeviewitem.Data = sle_prop.text
Expanded	Supported	tv_treeviewitem.Expanded = TRUE
		Note: the property is supported but is read-only.
ExpandedOnce	Supported	tv_treeviewitem.ExpandedOnce = TRUE
		Note: the property is supported but is read-only.

HasFocus	Supported	tv_treeviewitem.HasFocus = TRUE
		Note: the property is supported but is read-only.
ItemHandle	Supported	tv_treeviewitem.Itemhandle = long(sle_prop.text)
		Note: This property is supported but is read-only.
OverlayPictureIndex	Supported	l_tvi.OverlayPictureIndex = 2
Label	Supported	tv_treeviewitem.Label = sle_prop.text
Level	Supported	tv_treeviewitem.Level = long(sle_prop.text)
PictureIndex	Supported	tv_treeviewitem.PictureIndex = long(sle_prop.text)
SelectedPictureIndex	Supported	tv_treeviewitem.SelectedPictureIndex = long(sle_prop.text)
Selected	Supported	lb_value = tv_treeviewitem.Selected
StatePictureIndex	Supported	li_value = tv_treeviewitem.StatePictureIndex
ClassDefinition	Unsupported	None
CutHighLighted	Unsupported	None
DropHighLighted	Unsupported	None

8.2.1.26.2 Functions for TreeViewItem object

Function	Support Level	Example Code
ClassName	Supported	ls_returnvalue = tv_treeviewitem.ClassName()
TypeOf	Supported	If ltvi_1.TypeOf() = TreeViewItem! Then ls_returnvalue = "TreeViewItem!" End If
GetContextService	Unsupported	None

GetParent	Unsupported	None
-----------	-------------	------

8.2.1.27 UserObject object

Important Requirements

The UserObject object can be dynamically created (for example, by using the CREATE statement).

8.2.1.27.1 Properties for UserObject object

Property	Support Level	Example Code
BackColor	Supported	uo_test.BackColor = ll_value
Border	Supported	uo_test.Border = lb_value
BorderStyle	Supported	uo_test.BorderStyle = StyleBox!
		uo_test.BorderStyle = StyleLowered!
		uo_test.BorderStyle = StyleRaised!
		Note: StyleShadowBox! is unsupported.
ClassName	Supported	ls_value = uo_test.ClassName
BringToTop	Supported	uo_test.BringToTop = TRUE
Control[]	Supported	WindowObject lobj_control[]
		lobj_control[] = uo_test.control[]
DragAuto	Supported	uo_test.DragAuto=TRUE
DragIcon	Supported	uo_test.DragIcon='c:\archive\arrow.ico'
Enabled	Supported	uo_test.Enabled = TRUE
Height	Supported	uo_test.Height = 300
HScrollBar	Supported	uo_1.HScrollBar = false
Pointer	Supported	uo_test.Pointer='Help!'

		uo_test.Pointer='d:\archive\IBEAM.BMP.cur'
PictureName	Supported	uo_test.PictureName = "c:\pictures\pb1.bmp"
PowerTipText	Supported	uo_test.PowerTipText="UserObject PowerTip"
TabBackColor	Supported	uo_tab.tab_page1.TabBackColor = RGB(0,0,255)
TabOrder	Supported	uo_test.TabOrder = li_value
TabTextColor	Supported	uo_tab.tab_page1.TabTextColor =5566
Tag	Supported	uo_test.Tag = ls_value
Text	Supported	uo_test.Text = ls_value
Visible	Supported	uo_test.Visible = lb_value
VScrollBar	Supported	uo_test.VScrollBar = lb_value
Width	Supported	uo_test.Width = li_value
X	Supported	uo_test.X = li_value
Y	Supported	uo_test.Y = li_value
ClassDefinition	Unsupported	None
ColumnsPerPage	Unsupported	None
LibraryName	Unsupported	None
LinesPerPage	Unsupported	None
ObjectType	Unsupported	None
PictureMaskColor	Unsupported	None
Style	Unsupported	None
UnitsPerColumn	Unsupported	None
UnitsPerLine	Unsupported	None

8.2.1.27.2 Events for UserObject object

Event	Support Level	Example Code
Constructor	Supported	Constructor
Destructor	Supported	Destructor
DragDrop	Supported	DragDrop
DragEnter	Supported	DragEnter
DragLeave	Supported	DragLeave
DragWithin	Unsupported	DragWithin
RButtonDown	Supported	RButtonDown
Help	Unsupported	None
Other	Unsupported	None

8.2.1.27.3 Functions for UserObject object

Function	Support Level	Example Code
ClassName	Supported	uo_test.classname()
Drag	Supported	uo_test.Drag(Begin!)
GetParent	Supported	lobj_parent_name = this.GetParent()
Hide	Supported	uo_test.Hide()
Move	Supported	uo_test.Move(x1,y1)
PointerX	Supported	uo_test.PointerX()
PointerY	Supported	uo_test.PointerY()
PostEvent	Supported	uo_test.PostEvent(Clicked!)
Resize	Supported	uo_test.Resize(x1,y1)

SetFocus	Supported	uo_test.SetFocus()
SetPosition	Supported	uo_test.SetPosition(ToTop!)
		uo_two.SetPosition(Behind!, uo_three)
SetRedraw	Supported	li_return= uo_test.SetRedraw(TRUE)
Show	Supported	uo_test.Show()
TriggerEvent	Supported	uo_test.TriggerEvent(Clicked!)
TypeOf	Supported	If uo_test.typeof()=CheckBox! Then ls_value='CheckBox!' End If
AddItem	Unsupported	None
CreatePage	Unsupported	None
DeleteItem	Unsupported	None
GetContextService	Unsupported	None
InsertItem	Unsupported	None
PageCreated	Unsupported	None
Print	Unsupported	None

8.2.1.28 Window object

Important Requirements

- There are six window types: Child, Main, MDI, MDIHelp, Popup, and Response.
- Window object cannot be dynamically created (for example, using the CREATE statement); it must be defined in the PowerBuilder painter.
- Main Window can have an associated menu.
- User-defined window events and control events are supported.

8.2.1.28.1 Properties for Window object

Property	Support Level	Example Code
BackColor	Supported	w_1.BackColor = RGB(0, 0, 255)
Border	Supported	w_1.Border = TRUE
BringToTop	Supported	w_1.BringToTop = TRUE
Center	Supported	w_1.Center = TRUE
		Note: This property is unsupported for an MDI frame window.
Control []	Supported	Control[] graphicobject lobj_control lobj_control = w_1.control[1]
		Note: This property cannot be dynamically changed.
ControlMenu	Supported	w_1.ControlMenu= TRUE
Enabled	Supported	w_1.Enabled = TRUE
Height	Supported	w_1.Height = 750
		//Specifies the height of the window w_1
		Note: The Height property is unsupported for an MDI frame window.
HScrollBar	Supported	w_1.HScrollBar = TRUE
Icon	Supported	None
MaxBox	Supported	w_1.MaxBox= TRUE
MenuID	Supported	Menu menuvar = w_1.MenuID
MenuName	Supported	ls_value = w_1.MenuName
MinBox	Supported	w_1.MinBox = TRUE
Pointer	Supported	w_1.pointer = 'Icon!'
		w_1.pointer ='d:\archive\IBEAM.BMP.cur'

Resizable	Supported	w_1.Resizable = TRUE
		<p>Notes:</p> <ol style="list-style-type: none"> 1. For MDI frame window, the Resizable property will always be TRUE. 2. At the execution of the ArrangeSheet function, no matter whether the Resizable property of each open sheet is set to TRUE or FALSE, all open sheets will be tiled and resized so that they do not overlap. 3. After setting this property to True, setting Border properties will not be effective.
RightToLeft	Supported	w_1.RightToLeft = TRUE
		Note: If the RightToLeft property of a MDI Window is set to Yes, the content in the window except for the content in the status bar displays from right to left.
Tag	Supported	w_1.SetMicroHelp(This.Tag) //Set or get the Tag property
Title	Supported	w_1.Title = "Monthly Report"
TitleBar	Supported	w_1.TitleBar = TRUE
ToolBarVisible	Supported	w_1.ToolBarVisible = TRUE
		Note: The ToolBarVisible is unsupported for Popup and Main windows.
Visible	Supported	w_1.Visible = TRUE
		Note: This property is unsupported for sheets whose window type is Response, Popup or Main windows.
VScrollBar	Supported	w_1.VScrollBar = TRUE
Width	Supported	w_1.Width = 750 //Set or get the Width property
		Note: The Width property is unsupported for an MDI frame window.
WindowState	Supported	w_1.WindowState = Maximized!

		Note: On the Web, the initial window state for an MDI frame window is not based on the WindowState propert. It will be decided by the Internet Explorer state.
WindowType	Supported	IF dw_1.WindowType=Main! THEN MessaMessageBox("Window Type", "This is a main window.") END IF
X	Supported	w_1.X = 215 //Set or get the X property Notes: The X property is not supported for the MDI frame window.
Y	Supported	w_1.Y = 215 //Set or get the Y property Notes: The Y property is not supported for the MDI frame window.
ClassDefinition	Unsupported	None
ClientEdge	Unsupported	None
ColumnsPerPage	Unsupported	None
ContextHelp	Unsupported	None
KeyboardIcon	Unsupported	None
LinesPerPage	Unsupported	None
PaletteWindow	Unsupported	None
ToolbarAlignment	Unsupported	None
ToolbarHeight	Unsupported	None
ToolbarWidth	Unsupported	None
ToolbarX	Unsupported	None
ToolbarY	Unsupported	None

UnitsPerColumn	Unsupported	None
UnitsPerLine	Unsupported	None

8.2.1.28.2 Events for Window object

Event	Support Level	Example Code
Activate	Supported	Activate
		Note: Activate and Deactivate events are unsupported for dialogue boxes.
Clicked	Supported	Clicked
Close	Supported	Close
		Note: the Close event is handled the same way as Post in Apeon. In the Web application, the return value of the Close event may be different from that in PowerBuilder.
CloseQuery	Supported	CloseQuery
Deactivate	Supported	Deactivate
		Notes: 1). With the Web applications, the Activate/Deactivate events will not be triggered if the user shifts the focus from a window in the current application to a place outside the application. 2). Activate and Deactivate events are unsupported for dialogue boxes.
DoubleClicked	Supported	DoubleClicked
DragDrop	Supported	DragDrop
DragEnter	Supported	DragEnter
DragLeave	Supported	DragLeave

DragWithin	Supported	DragWithin
Hide	Supported	Hide
HotLinkAlarm	Supported	HotLinkAlarm
Key	Supported	Key
MouseDown	Supported	MouseDown
MouseMove	Supported	MouseMove
MouseUp	Supported	MouseUp
Open	Supported	Open
RButtonDown	Supported	RButtonDown
Resize	Supported	Resize
Show	Supported	Show
Timer	Supported	Timer
Help	Unsupported	None
Other	Unsupported	None
RemoteExec	Unsupported	None
RemoteHotLinkStart	Unsupported	None
RemoteHotLinkStop	Unsupported	None
RemoteRequest	Unsupported	None
RemoteSend	Unsupported	None
SystemKey	Unsupported	None
ToolbarMoved	Unsupported	None

8.2.1.28.3 Functions for Window object

Function	Support Level	Example Code
ArrangeSheets	Supported	w_mdi.ArrangeSheets(arrangetype)
ChangeMenu	Supported	Parent.ChangeMenu(m_test_menu2) Notes: 1). It is unsupported to use a menu item as the <i>menuname</i> argument. 2). The menu you want to make the current menu and the current menu cannot be the same menu.
ClassName	Supported	ls_value = w_test.ClassName()
CloseChannel	Supported	w_test.CloseChannel()
CloseUserObject	Supported	w_test.CloseUserObject()
ExecRemote	Supported	ExecRemote("[Save()]", "Excel", "REGION.XLS")
GetActiveSheet	Supported	w_test.GetActiveSheet()
GetDataDDE	Supported	w_test.GetDataDDE(Str20)
GetDataDDEOrigin	Supported	string ls_name GetCommandDDEOrigin(ls_name)
GetFirstSheet	Supported	w_test.GetFirstSheet()
GetNextSheet	Supported	w_test.GetNextSheet()
GetRemote	Supported	GetRemote("[Save()]", "Excel", "REGION.XLS")
GetParent	Supported	w_test.GetParent()
Hide	Supported	Parent.Hide()
Move	Supported	w_test.Move(X,Y)
OpenChannel	Supported	long handle handle = OpenChannel("Excel", "REGION.XLS")
OpenUserObject	Supported	w_test.OpenUserObject()

		Note: The userobjecttype argument cannot be a system control.
OpenUserObjectWithParm	Supported	w_test.OpenUserObjectWithParm()
ParentWindow	Supported	w_parent = w_test.parentwindow()
PointerX	Supported	li_dist = w_city.PointerX()
PointerY	Supported	li_dist = w_city.PointerY()
PostEvent	Supported	w_main.PostEvent(Clicked!)
Resize	Supported	w_test.Resize(300,250)
RespondRemote	Supported	IF GetDataDDE(Value) = 1 THEN RespondRemote(TRUE) END IF
SetFocus	Supported	w_test.SetFocus()
		Unsupported: SetFocus(w_test)
SetMicroHelp	Supported	w_test.SetMicroHelp ("Preview")
SetPosition	Supported	w_test.SetPosition(position [, precedingwindow])
		Note: The window with property TopMost may not always be the top one in Apeon.
SetRedraw	Supported	li_return= w_test.SetRedraw(FALSE)
SetRemote	Supported	SetRemote("R5C7", "4500", handle)
Show	Supported	Parent.Show()
StartHotLink	Supported	StartHotLink("Any", "MyPBApp", "Any")
StopHotLink	Supported	StopHotLink("Any", "MyPBApp", "Any")
TriggerEvent	Supported	w_test.TriggerEvent(close!)

TypeOf	Supported	if this.typeof()=window! Then messagebox("show", "test typeof()") end if
WorkSpaceHeight	Supported	Height = W_employee.WorkSpaceHeight()
WorkSpaceWidth	Supported	Width = W_employee.WorkSpaceWidth()
WorkSpaceX	Supported	w_test.WorkSpaceX()
WorkSpaceY	Supported	w_test.WorkSpaceY()
GetCommandDDE	Unsupported	None
GetCommandDDEOrigin	Unsupported	None
GetContextService	Unsupported	None
GetToolbar	Unsupported	None
GetToolbarPos	Unsupported	None
Print	Unsupported	None
SetDataDDE	Unsupported	None
SetToolbar	Unsupported	None
SetToolbarPos	Unsupported	None
StartServerDDE	Unsupported	None
StopServerDDE	Unsupported	None

8.2.1.29 Window types

The six window types, including Child, Main, MDI, MDIHelp, Popup, and Response, are supported well by Apeon with the following behavior differences.

- MDI and MDIHelp:** If the MDI window for a Web application is closed, the whole Web application is closed. However, with a PowerBuilder application, if there is a

Main window open before the MDI window is opened, the application returns to the Main window when the MDI window is closed.

For a Web application, a MDI sheet will be always behind the windows that are opened with the function `Open`

For a Web application, events that should be triggered by clicking the Title bar of an MDI frame window will be triggered by clicking the Title bar of Internet Explorer

- **Child:** In PowerBuilder, a Child window is always above its parent window. With Web applications, a Child window can be behind its parent window.

8.2.1.30 Opening and closing windows

Opening Windows

- Open functions

```
Open ( windowvar, windowtype {, parent } )
```

Refer to the [Window functions](#) section of System Functions for the supported syntax for opening the windows.

In the Web application, if window A is opened by the `Open` function whereas window B is opened by the `OpenSheet` function, window A will always display over window B.

- Triggering of events

When a window is open, the Constructor events for the instance variables and controls are executed in the same order on the Web as in PowerBuilder:

1. Executing Constructor events for all instance variables is prior to executing Constructor events for all controls.
2. For the instance variables/controls, the events are triggered by the following order: outside --> inside and top -->bottom.
3. If a control is a container (tab/tab page/visual user object), the Constructor event for the container is triggered first. Then all the controls in the container are subsequently triggered.

Events triggered for the window object follows the order `Open >> Resize >> Show >> Activate`.

- Front-to-back order

The same rule applies in PowerBuilder: the front-to-back order of all the controls in the window is determined by the `Control[]` property of the window and the `BringToTop` property of each specific control.

Closing Windows

- Close functions

Refer to the [Window functions](#) section for the supported syntax for closing the windows.

- Triggering of events

When a window is closed, the Destructor events of the variables and controls in the window are executed in the same order on the Web as in PowerBuilder:

1. The Destructor events for the controls are triggered in the same order as the Constructor events.
2. Executing Destructor events for all controls happens prior to executing Destructor events for all instance variables.
3. The Destructor events for the instance variables are triggered in the opposite order as the Constructor events.

8.2.1.31 User operation in windows

Supported

The following major user operations are supported.

- Maximize
- Restore
- Minimize
- Single-click
- Double-click
- Drag
- Close
- Switching between controls in the window with the Tab key
- Switch between windows by pressing the Ctrl + Tab key

Unsupported

- It is unsupported to scroll a window using the PageUp/PageDown, upArrow/downArrow, or leftArrow/rightArrow keys
- It is unsupported to switch between controls in a window by pressing the Enter/UpArrow/DownArrow/LeftArrow/RightArrow key

8.2.1.32 Window variables

Window Variables

- Window type arrays are supported, and window type variable that is a structure member is supported.
- The declaration and assignment of window variable are all supported.

- It is supported to dynamically create a window instance by using CREATE. For example, the following syntax is supported:

```

window lwin_owindow
lwin_owindow = create Window
lwin_owindow.title = "This is a city processing window"

```

8.2.2 Unsupported objects

Unsupported

- It is strongly recommended that for a given application, no objects have the same name.
- Apeon does not support the following system objects, including its events, properties and functions:

ADOResultSet	ArrayBounds	ClassDefinition	ClassDefinitionObject
ConnectionInfo	ConnectObject	ContextInformation	ContextKeyword
CORBABadTypeCon text	CORBABadInvorder	CORBABadOperation	CORBABadParam
CORBABadTypeCod e	CORBACommFailur e	CORBACurrent	CORBADataConversion
CORBAFreeMem	CORBAImpLimit	CORBAInitialize	CORBAInternal
CORBAIntFrePos	CORBAInvalidTrans action	CORBAInvFlag	CORBAInvIdentInvOBJ Ref
CORBAMarshal	CORBANoImplemen t	CORBANoMemory	CORBANoPermission
CORBANoResources	CORBANoResponse	CORBAOBJAdapter	CORBAObjectNoTexist
CORBAPersistStore	CORBASystemExce ption	CORBATransactionRe quired	CORBATransactionRoll edback
CORBATranslent	CORBAUnion	CORBAUnknown	CORBAUserException
CPlusPlus	DivideByZeroError	DWRuntimeError	EnumerationDefinition

EnumerationItemDefinition	Error	ErrorLogging	Exception
ExtObject	JaguarORB	NullObjectError	OLERuntimeError
OLEStorage	OLEStream	OLETxnObject	OMControl
OMCustomControl	OMEmbeddedControl	OMObject	OMStorage
OMStream	ORB	PBTocppObject	Pipeline
ProfileCall	ProfileClass	ProfileLine	ProfileRoutine
Profiling	RemoteObject	ResultSet	ResultSets
RuntimeError	ScriptDefinition	Service	SimpleTypeDefinition
SSLCallBack	SSLServiceProvider	SystemFunctions	Throwable
TraceActivityNode	TraceBeginEnd	TraceError	TraceESQL
TraceFile	TraceGarbageCollect	TraceLine	TraceObject
TraceRoutine	TraceTree	TraceTreeError	TraceTreeESQL
TraceTreeGarbageCollect	TraceTreeLine	TraceTreeNode	TraceTreeObject
TraceTreeRoutine	TraceTreeUser	TraceUser	TransactionServer
Transport	TypeDefinition	VariableCardinalityDefinition	VariableDefinition

9 PowerScript Reference

9.1 PowerScript Topics

9.1.1 Overview

The PowerBuilder language is PowerScript. PowerScript is used in scripts and user-defined functions to build PowerBuilder applications.

During conversion of a PowerBuilder application, script compiled in PowerScript is converted to JavaScript.

In the following section, PowerScript refers to the PowerBuilder programming language, while JavaScript refers to the programming language used for PowerBuilder-to-Web conversion.

9.1.2 Object-Oriented programming

The application may make use of the powerful object-oriented programming techniques that are typically found in PowerBuilder applications:

- Inheritance
Creating a new control in a child object by copying and pasting an existing control in the PowerBuilder painter is unsupported if the existing control is inherited from the parent object.
- Encapsulation
- Polymorphism

9.1.3 Language basics

9.1.3.1 Comments

Supported

1) Common Comments

- Double-slash method: Code // Comment
A= B + C// Comment
- Slash-and-asterisk method: /* Comment */
A= /* comment */ B + C

2) Embedded comments

```
// Comment1 /*Comment2
/* Comment1 // Comment2 */
/* Comment1 /* Comment2*/ Comment3 */
```

For example:

```
A = B + C /* This comment starts here.
```

```

/* This is the start of a nested comment. The nested comment ends here. */
The first comment ends here.
*/ + D + E + F

```

Unsupported

None.

9.1.3.2 Identifiers

Supported

Identifiers in Appeon refer to global/instance/local variable names, or object names, or **menu names & menu item names**. Note that menu names & menu item names are regarded as identifiers.

Rules for identifiers:

- Identifiers can be reserved words in JavaScript, apart from the identifier word "Object".
- Can have up to 40 characters but no spaces.
- Must start with a letter or an _ (underscore) and can include any combination of the letters, numbers and special characters listed in [Appeon supported identifiers](#).
- In the case of duplicate objects, objects earlier in the PBL list overwrite those later in the PBL list.

Unsupported

- Objects of different types cannot have the same names.
- Objects of the same types, even if they are in different PBLs, cannot have the same name.
- The “Ω” will not be automatically converted to “ω” in Appeon.
- Identifiers cannot be reserved words in Appeon: appeondatawindow, appeondatastore, appeonservice, appeon_nvo_db_update, appeonnextfuncs, appeonfileservice, ejbserial, ejbobject and parse_retval_object.

9.1.3.2.1 Appeon supported identifiers

Letters, numbers or special characters with Unicode listed below or within the intervals are supported.

- All intervals are closed intervals.
- Cannot start with the letter, number or special character corresponding to the Unicode (or within the intervals) marked with "*".

30-39 *	41-5A	5F	61-7A	B7 *	C0-D6
D8-F6	F8-131	134-13E	141-148	14A-17E	180-1C3
1CD-1F0	1F4-1F5	1FA-217	250-2A8	2BB-2C1 *	2D0-2D1 *
300-345 *	360-361 *	386	387*	388-38A	38C
38E-3A1	3A3-3CE	3D0-3D6	3DA	3DC	3DE
3.00E+00	3E2-3F3	401-40C	40E-44F	451-45C	45E-481

483-486 *	490-4C4	4C7-4C8	4CB-4CC	4D0-4EB	4EE-4F5
4F8-4F9	531-556	559 *	561-586	591-5A1 *	5A3-5B9 *
5BB-5BD *	5BF *	5C1-5C2 *	5C4 *	5D0-5EA	5F0-5F2
621-63A	640 *	641-64A	64B-652 *	660-669 *	670 *
671-6B7	6BA-6BE	6C0-6CE	6D0-6D3	6D5	6D6-6E8 *
6EA-6ED *	6F0-6F9 *	901-903 *	905-939	93C-94D *	951-954 *
958-961	962-963 *	966-96F *	981-983 *	985-98C	98F-990
993-9A8	9AA-9B0	9B2	9B6-9B9	9BC *	9BE-9C4 *
9C7-9C8 *	9CB-9CD *	9D7 *	9DC-9DD	9DF-9E1	9E2-9E3 *
9E6-9EF *	9F0-9F1	A02 *	A05-A0A	A0F-A10	A13-A28
A2A-A30	A32-A33	A35-A36	A38-A39	A3C *	A3E-A42 *
A47-A48 *	A4B-A4D *	A59-A5C	A5E-A5E	A66-A74 *	A81-A83 *
A85-A8B	A8D	A8F-A91	A93-AA8	AAA-AB0	AB2-AB3
AB5-AB9	ABC-AC5 *	AC7-AC9 *	ACB-ACD *	AE0	AE6-AEF *
B01-B03 *	B05-B0C	B0F-B10	B13-B28	B2A-B30	B32-B33
B36-B39	B3C-B43 *	B47-B48 *	B4B-B4D *	B56-B57 *	B5C-B5D
B5F-B61	B66-B6F *	B82-B83 *	B85-B8A	B8E-B90	B92-B95
B99-B9A	B9C	B9E-B9F	BA3-BA4	BA8-BAA	BAE-BB5
BB7-BB9	BBE-BC2 *	BC6-BC8 *	BCA-BCD *	BD7 *	BE7-BEF *
C01-C03 *	C05-C0C	C0E-C10	C12-C28	C2A-C33	C35-C39
C3E-C44 *	C46-C48 *	C4A-C4D *	C55-C56 *	C60-C61	C66-C6F *
C82-C83 *	C85-C8C	C8E-C90	C92-CA8	CAA-CB3	CB5-CB9
CBE-CC4 *	CC6-CC8 *	CCA-CCD *	CD5-CD6 *	CDE	CE0-CE1
CE6-CEF *	D02-D03 *	D05-D0C	D0E-D10	D12-D28	D2A-D39
D3E-D43 *	D46-D48 *	D4A-D4D *	D57 *	D60-D61	D66-D6F *
E01-E2E	E30	E31 *	E32-E33	E34-E3A *	E40-E45
E46-E4E *	E50-E59 *	E81-E82	E84	E87-E88	E8A
E8D	E94-E97	E99-E9F	EA1-EA3	EA5	EA7
EAA-EAB	EAD-EAE	EB0-EB9 *	EBB-EBD *	EC0-EC4 *	EC6 *
EC8-ECD *	ED0-ED9 *	F18-F19 *	F20-F29 *	F35 *	F37 *
F39 *	F3E-F3F *	F40-F47	F49-F69	F71-F84 *	F86-F8B *
F90-F95 *	F97 *	F99-FAD *	FB1-FB7 *	FB9 *	10A0-10C5
10D0-10F6	1100	1102-1103	1105-1107	1109	110B-110C
110E-1112	113C	113E	1140	114C	114E
1150	1154-1155	1159	115F-1161	1163	1165
1167	1169	116D-116E	1172-1173	1175	119E
11A8	11AB	11AE-11AF	11B7-11B8	11BA	11BC-11C2
11EB	11F0	11F9	1E00-1E9B	1EA0-1EF9	1F00-1F15
1F18-1F1D	1F20-1F45	1F48-1F4D	1F50-1F57	1F59	1F5B
1F5D	1F5F-1F7D	1F80-1FB4	1FB6-1FBC	1FBE *	1FC2-1FC4
1FC6-1FCC	1FD0-1FD3	1FD6-1FDB	1FE0-1FEC	1FF2-1FF4	1FF6-1FFC
20D0-20DC*	20E1*	2126 *	212A-212B *	212E *	2180-2182 *
3005 *	3007	3021-302F *	3031-3035 *	3041-3094	3099-309A *
309D-309E *	30A1-30FA	30FC-30FE	3105-312C	4E00-9FA5	AC00-D7A3

9.1.3.3 Labels

Labels and GOTO statements are unsupported

9.1.3.4 Special ASCII characters

Supported

ASCII character	To Specify This	Enter This
Common ASCII characters	Newline	~n
	Tab	~t
	Carriage return	~r In Appeon, "~r" is considered a newline character.
	Formfeed	~f
	Backspace	~b
	Double quote	~"
	Single quote	~'
Any ASCII character	Tilde	~~
	Hexadecimal	~h###
	Octal	~o####

Unsupported

- 1) In a Web application, a character whose ASCII value is greater than 127 cannot be saved to the database.
- 2) In PowerBuilder, for characters whose ASCII value is greater than 128, the equal operator will consider them the same. However, in Appeon, the equal operator will not consider them the same.
- 3) Common ASCII character: Vertical tab (~v)
- 4) Any ASCII character: Decimal (~####)
Note: Appeon ignores the escape character ("~") specified in the PowerBuilder painter. In SQL statements, "~" is handled as escape character.

9.1.3.5 Null values

Supported

NULL means undefined or unknown. It is not the same as an empty string, a zero, or a date of 0000-00-00. For example, NULL is neither 0 nor "" (empty string).

Null in PowerBuilder is directly translated into Null in JavaScript.

Unsupported

Expressions involving Null values may arrive at different values in JavaScript from their values in PowerScript. For example, the expression $A + B$ in PowerScript will return Null if any of A or B is Null. However, in JavaScript, the expression will not return Null even if A or B is Null. >In addition, an arithmetic or relational operation involving a null value always returns null in PowerBuilder. As shown in the following table, the return value is not always null in JavaScript:

Operation	Return Value
-----------	--------------

Assuming SetNull(A), SetNull(B)		PowerBuilder	JavaScript
Arithmetic	A+1	Null	1
	A+B	Null	0
	A*B	Null	0
Relational	A=1	Null	FALSE
	A<>1	Null	TRUE
	NOT (A=1)	Null	TRUE
	A=A	Null	TRUE
	A=B	Null	TRUE
	IsNull(A=1)	TRUE	FALSE
String concatenation	A+"ABC"	Null	"NULLABC"
	A+B	Null	"NULLNULL"

9.1.3.6 Reserved words

Supported Reserved Words

and	call	case	catch
choose	close	commit	connect
constant	continue	create	cursor
declare	delete	describe	disconnect
do	dynamic	else	elseif
end	event	execute	exit
FALSE	fetch	first	for
forward	from	function	global
halt	if	immediate	insert
into	is	last	loop
next	not	of	on
open	or	parent	post
prior	prepare	ref	return
rollback	rpcfunc	select	step
subroutine	super	then	this
to	trigger	TRUE	try
type	until	update	using
while	with	within	selectblob
updateblob			

- The reserved word HALT is supported, but there cannot be any code after the Halt statement. For example, in the following script, "close(parent)" will be ignored in the Apeon conversion process.

```
Halt //supported
```

```
Close(parent) //this will be ignored
```

PROCEDURE can only be used in the DECLARE Procedure SQL statement

Unsupported Reserved Words

enumerated	external	finally	goto
indirect	intrinsic	library	system
systemread	systemwrite	throw	throws

9.1.3.7 Pronouns

Supported

The following pronouns in PowerScript are supported:

- Parent - refers to the object that contains the current object.
Using Parent in the script for a visual user object is supported.
- This - refers to the window, user object, menu, application object, or control that owns the current script.
- Super - refer to the immediate ancestor for a descendant object or control.

Unsupported

None.

9.1.3.8 Statement continuation & separation

Supported

- The statement continuation character (&) and syntax are supported. The syntax is as follows:

Start of statement &
more statement &
end of statement

- The statement separation character (;) and syntax are supported. The syntax is as follows:

Statement1; statement2

Note: White Space (Blanks, tabs, formfeeds, and comments) in the statement is supported and treated the same way as in PowerBuilder.

Unsupported

None.

9.1.4 Data types

9.1.4.1 Standard data types

Supported

Apeon supports all standard PowerBuilder data types as outlined in the table below:

Blob	Boolean	Char or character	Date	DateTime
Decimal* or Dec	Double	Integer or Int	Long	Real
String	Time	UnsignedInteger	UnsignedInt	UInt
UnsignedLong	ULong			

Decimal in Datawindow

Since Apeon 6.0, the decimal precision extends to 28 digits in a DataWindow by enabling the 28-digital Decimal option in AEM.

1. Precision

With 28-digit Decimal option, the decimal precision in DataWindow supports up to 28 digits and ranges from -39,614,081,257,132,168,796,771,975,167 to +39,614,081,257,132,168,796,771,975,167.

2. Function Limitations

28-digit Decimal supports the following arithmetic operators: Arithmetic operators: “+”, “-“, “*”, “/”

Relational operators: =, >, <, <>, >=, <=

28-digit Decimal is supported in the following DataWindow expressions: Abs, Avg, CumulativeSum, Medium Sign, Sum,

3. Function Differences

In the calculation, values/variables of Double or Real data type are handled as values/variables of Decimal data type. Constants in an expression will be handled as Double datatype if the expression is being dynamically modified and contains relational operators (=, >, <, <>, >=, <=). Constants in an expression will be handled as Decimal data type if the expression is being dynamically modified and contains arithmetic operators (“+”, “-“, “*”, “/”). Appending the letter D in uppercase or lowercase to identify a number as a decimal constant in a DataWindow expression is unsupported in 28-digit decimal. Decimal displaying in the DataWindow in Apeon may differ from that in PowerBuilder. Sometimes Decimal displays in scientific notation in PowerBuilder but displays in standard notation in Apeon. If exceeding 28 digits, the decimal number cannot be correctly displayed on the Web.

Differences

- **UnsignedLong (32-bit) and UnsignedInteger (16-bit)**
The UnsignedLong (32-bit) and UnsignedInteger (16-bit) data types are handled as Long and Integer. Therefore, the supported range for UnsignedLong is from 0 to 4,294,967,295, and the supported range for UnsignedInteger is from 0 to 65,535.
- **Decimal, Real and Double**
In operations that contain calculations of precision, values/variables of Decimal or Real data type are handled as values/variables of Double data type.
In operations that do not contain calculations of precision, values/variables of Double or Real data type are handled as values/variables of Decimal data type.
- You must ensure that the result of a numeric expression will not cause overflow in PowerBuilder. Otherwise the result on the Web is different from that in PowerBuilder.

9.1.4.2 Any data type

Any data type is fully supported, with the exception as follows:

Assigning the return value of a String function to an Any variable that has not been previously assigned is unsupported.

Using Any in the OCX is unsupported.

Using Any array in a dot notation is unsupported. To work around it, you can use Any variable instead of Any array. For example:

Unsupported code:

```
la_data[] = dw_1.object.data
```

Supported code:

```
any la_data1  
la_data1 = dw_1.object.data
```

9.1.4.3 System object data types**Supported**

- Autoinstantiated object data types:
NVOs (Class User Object) with their Autoinstantiate property checked;
Structure data types;
Autoinstantiated system objects: ListViewItem, ListViewItem.
- Nonautoinstantiated object data types:
NVOs (Class User Object) with their Autoinstantiate property unchecked;
Visual controls (user-defined visual controls and system controls)
Nonautoinstantiated system objects, including: Application, DataStore,
DataWindowChild, DynamicStagingArea, MDIClient, Menu, Message, Transaction
Object, Window.
- PowerObject classes:
The supported PowerObject classes include DragObject, DWOBJECT, GraphicObject,
PowerObject, and WindowObject.
Note: In the Web application, because the declaration of a PowerObject does not
specify the object type, the initial value of the variable is NULL. Therefore, when the
IsValid function is executed, IsValid returns NULL.

Unsupported

- Among all the supported system objects and system controls, if the data type is a non-global Transaction Object, the data type is not supported.
- If the data type is an unsupported system object or system control, it is unsupported.
- Do not refer to a DataWindow object by using global functions or instance variables. Otherwise the executing result will have error on the Web.

9.1.4.4 Enumerated data types**Supported**

All system-defined enumerated data types and values are supported, unless the enumerated data type belongs to an unsupported feature.

Unsupported

Enumerated data types have predefined sets of values. Most of the PowerBuilder enumerated data types have default values. Using the default values of enumerated data types is unsupported.

9.1.4.5 Forced conversion between data types

Supported

1. Forced conversion between string and char data type:
 - Assigning strings to char arrays
 - Assigning char arrays to strings
 - When a string literal is assigned to a char variable, the first character of the string literal is assigned to the variable. For example:
`char c = "xyz" //results in the character x being assigned to the char variable c.`
 - Special characters (such as new line, form feed, octal, hex, etc.) can be assigned to char variables using string conversion. For example:
`char c = "~n"`
 - A char variable assigned to a string variable results in a one-character string.
 - Expressions using both strings and char arrays promote the chars to strings before evaluation. For example:
`char c
if (c = "x") then
// promotes the contents of c to a string before comparison with the string "x"`
 - Using chars in PowerScript functions. All PowerScript functions that take strings also take chars, subject to the above described conversion rules.
2. Forced conversion between numeric data types:
 - The order of precedence in PowerBuilder regarding numeric data types is supported. The following is the order of precedence from highest to lowest (based on the range of values for each data type):
(High) Double >> Real >> Decimal >> UnsignedLong >> Long >> UnsignedInteger >> Integer (Low)
 - If operands in an expression have different data types, the value whose type has lower precedence is converted to the data type with higher precedence. For example:
`int x; x=2.4`
PowerScript result: the value of x is actually 2, because x is defined as an integer. PowerBuilder will cut the digit after the decimal point, then assign 2 to x.
 - Unsigned type has precedence over signed. So if one operand is signed and the other is unsigned, both are promoted to the unsigned version of the higher type. For example, if one operator is a Long and another an UnsignedInteger, both are promoted to UnsignedLong.
 - Transform other numeric data types recessive to integer or int data. A force conversion from other numeric data to int/long data will arrive at the same value in PowerScript and JavaScript.

Unsupported

- In PowerBuilder, when Time data type is automatically converted into DateTime, the date is always 1900-01-01. Thus, the DataWindow always gets a value of DateTime type: Retrieve (1900-01-01 xx:xx:xx). However, if connected to a JDBC SQL Server, the default date is the current day. In other words, DataWindow gets Retrieve(today xx:xx:xx) when Time data type is automatically converted into DateTime.

Difference

When assigning a numeric number to an Any data, a forced conversion from Any to Integer may arrive at different values in PowerBuilder and on the Web. For example,

```
Integer li_value
```

```
any la_value = 3.5
```

```
li_value = Integer(la_value) // li_value will be 3 on the Web
and 4 in PowerBuilder.
```

9.1.5 Declarations

9.1.5.1 Variables and constants

Variable scope	Supported
	Global, local and instance
	Unsupported
	Shared
Variable declaration syntax	Supported
	datatype { { size } } { { precision } } variablename { = value } {, variablename2 { = value2 } }
	The datatype can be any standard type and system object type. Declaring multiple variables of the same type at one time is supported. For example: integer li_a=5, li_b=10. Naming variables with Non-English characters or numbers.
	Unsupported
	When a global decimal variable is declared, or a decimal constant is used to declare a variable, the specification of precision for the global decimal or the decimal constant will be ignored. A variable having the same name as a global variable is or a global function is unsupported. A global variable cannot have the same name as a control in a window. For example, if there is a GroupBox control named as gb_1, the following syntax is unsupported: Global Boolean gb_1. Placing a cursor or stored procedure declaration in the declaration of instance variables is unsupported.
Global scope operator (::)	Unsupported
	Referring to a global variable by using the global scope operator (::) before the variable name is unsupported; (i.e. The syntax with "::globalname" is unsupported).
Constant type	Supported

	All of the standard data types
Constant declaration syntax	Supported
	<p><code>CONSTANT datatype constname = value</code></p> <p>The constant can only be public.</p> <p>It is supported if the <i>value</i> is an expression. For example:</p> <pre>constant date ld_date = today()</pre>
Access to instance variables	Unsupported
	<p>While the instance variable is passed by reference to a script and before the execution of this script is finished, the value of this instance variable is changed by another script, however, Apeon cannot capture the changes.</p>
Initial values of variables and constants	Supported
	<p>When a variable or constant is declared, a default initial value is automatically assigned or an initial value can be specified in the declaration.</p> <p>The initial values of enumerated data types are converted to null in JavaScript. Except for the initial values of enumerated data types and the Any data type, the default initial values in PowerScript are supported.</p>
	<p>Unsupported</p> <p>There are different rules in PowerBuilder than in JavaScript if specifying an expression as an initial value:</p> <p>It is unsupported to use the instance constants of a non-instantiated object.</p> <p>With the syntax <code>datatype variable = expression</code>, in PowerBuilder, the expression's value is assigned to the variable when the script is compiled (not during execution). In JavaScript, the expression's value is set to the variable during execution. For example, if the declaration is the following:</p> <pre>date d_date = Today()</pre> <p>The value of <code>d_date</code> is the date when the script is compiled in PowerBuilder, and it is the date when the application is running in JavaScript.</p> <p>Therefore, making the declaration of a variable and assigning it with the initial value in separate PowerScript statements is recommended. For example:</p> <pre>date d_date d_date = Today()</pre>

9.1.5.2 Arrays

9.1.5.2.1 Declaration syntax

The declaration syntax for arrays is supported:

```
{ access } datatype variablename { d1, ..., dn } { =
{ valuelist } }
```

- The access is always PUBLIC.

- The datatype cannot be an unsupported data type (refer to the [Data types](#) section). For decimals, you can specify the precision of the data by including an optional value in brackets after the datatype. For example, `decimal {2} ld_prices[]`.
- Both variable-size arrays and fixed-size arrays are supported.

```
date ld_birthdays[ ]
string ls_array[10 ]
```
- Arrays with specified lower bound and upper bound are supported, even if the lower bound is a non-integer or a negative value. If the lower bound is not an integer, it will be rounded off.

```
string ls_name[-10 to 15]
```
- Both single-dimensional and multi-dimensional arrays are supported.

```
integer ls_array[10]
integer li_score[2,3]
```
- Using TO to change array index values is supported.

```
integer li_staff[100, 0 to 20, - 3 to 5]
```
- Multiple arrays of the same type can be declared simultaneously.

```
string ls_array1[10], ls_array2[5], ls_array3[100]
```

Assigning an array to an array is unsupported if the array elements of Any data type. For example,

```
any la_1[3], la_2[4]
...
la_1[1] = la_2[4]           // Unsupported
```

9.1.5.2.2 Initialization and assignment

Supported

- Each element of an array can be initialized to the same default value as its underlying data type. The default value for string data type elements is ""; the default value for numeric data type element is 0.
- The default length of variable-size arrays is 0, which means that the array does not have any elements. Initializing several elements of variable-size array is supported if the element after the several elements is assigned with a value. Example:

```
integer li_array [ ]
li_array[8]=8           //The values of li_array[1], ...,
li_array[7] are initialized to the default value 0
```

- Using arraylists to assign values to an array, or assign values to array elements separately, is supported. There can be expression(s) in the arraylist. The result will be the same as in PowerBuilder.

```
li_array[10] = {1,2,3,4,5}
li_array[3,2] = {1,2,3,4,5}
ld_date = {today(),relativedate(today(),1), 2002-12-31}
```

- Assigning one array to the another is supported. The result will be the same as in PowerBuilder.

- Assigning a variable-size array to another variable-size array. Example:

```
integer li_test1[]={1,2,3,4,5}
integer li_test2[] = {10,20,21,22,23,24,25,26,27}
... .
```

```
li_test2=li_test1
```

If the array type is NVO, the assignment operation does not trigger the Constructor event.

- Assigning a fixed-size array to another fixed-size array. It can be between one-dimensional arrays, or between one-dimensional array and a multi-dimensional array, or between multi-dimensional arrays. Example:

```
integer li_test0[10] = {21,22,23,24,25,26,27,28}
integer li_test1[5] = {1,2,3,4,5}
integer li_test2[2,6] = {11,12,13,14,15,16,17,18}
integer li_test3[2,3,5] = {11,12,13,14,15,16,17,18}
```

```
li_test1=li_test0 //assign a one-dimensional
array to another one-dimensional array
li_test2=li_test1 //assign a one-dimensional
array to a multi-dimensional array
li_test1=li_test2 //assign a two-dimensional
array to a one-dimensional array
li_test3=li_test2 //assign a multi-dimensional
array to another multi-dimensional array
```

- Assigning a variable-size array to a fixed-size array. Example:

```
integer li_test0[] = {21,22,23,24,25,26,27,28}
integer li_test1[10] = {1,2,3,4,5}
li_test1=li_test0
```

- Assigning a fixed-size array to a variable-size array. Example:

```
integer li_test1[10] = {1,2,3,4,5}
integer li_test0[]
li_test0[30] = 100
li_test0=li_test1 //the element number of the
li_test0 will be 10
```

- Reinitializing arrays with the following steps is supported:
 1. Declare a dummy array of the same type (never put any values into the array).
 2. Simply set originalarray = dummyarray.
 This will clear out the original array and cause UpperBound to return the correct value. The same rule applies to PowerBuilder.
- The index of an array can be an expression. Example:


```
for A = 1 to upperbound(ls_array)
ls_array[A,1] = .....
end for
ls_array[integer(ltvi_item.data)] =123
```

Unsupported

- Assigning values between instance variables of structure type is unsupported. For example, the following lines of code are unsupported:

```
Str_a L1
Str_b L2
L1 = L2
```

Assigning structure arrays to non-structure arrays is unsupported. For example, the following lines of codes are unsupported:

```
str_dwstruct lst_dwstr[]
powerobject po_ary[]
po_ary = lst_dwstr // Unsupported
```

9.1.5.2.3 *Passing arrays as arguments*

Passing a variable-size or fixed-size array as an argument is supported. For example:

```
uf_convertarray(a) //integer a[]
uf_convertarray(a) //integer a[10]
```

The following two syntaxes will get the same result for one-dimensional, multi-dimensional, or variable-size arrays:

```
li_upper = upperbound(ls_array) // string ls_array[]
li_upper = upperbound(ls_array []) // string ls_array[]
```

9.1.5.2.4 *Complex arrays*

- Structure arrays are supported. The declaration, initialization and assignment of structure arrays comply with the general rules for standard arrays.
- UserObject arrays are supported. Please refer to the [User objects](#) section for details.
- Nested arrays are supported.
- Enumerated type arrays are supported.

9.1.5.2.5 *Unsupported*

If arguments of a function are arrays, arguments and corresponding actual parameters cannot be of different dimensions.

9.1.5.3 External functions

Declare external functions

Supported

- Declaring and calling local external functions
- Declaring and calling global external functions
- Data types of external function arguments can be the following:
Array, Boolean, Blob, Char, Date, DateTime, Double, Decimal, Integer, Long, Real, String, Structure, and Void
- Data types of external function return values can be the following:
Array, Boolean, Blob, Char, Date, DateTime, Double, Decimal, Integer, Long, Real, String, Structure, Unsigned Integer, Unsigned Long, and Void

9.1.6 Operators & expressions

Supported operators and expressions

- Arithmetic operators for numeric data types: +, -, *, /, ^, ++, --, +=, -=, *=, /=, ^=
- Relational operators for all data types: =, >, <, <>, >=, <=, NOT, AND, OR
- The concatenation operator for String data types: +
- DataWindow expressions are supported

- The supported features of operators & expressions are related to [Null values](#) and [Forced conversion between data types](#).

Precedence of supported operators

These operators will follow the order of precedence listed in the table below:

Order	Operator	Purpose
1	()	Grouping
2	+, -, NOT	Unary plus (indicates positive number), unary minus (indicates negative number), Negation
3	^	Exponentiation
4	*, /	Multiplication and division
5	+, -	Addition and subtraction; string concatenation
6	>, <, <=, >=	Relational operators
7	=, <>	Relational operators
8	AND	Logical and
9	OR	Logical or

Unsupported

Avoid data overflow in the application. If there is data overflow, the Web application handles the data differently from the PowerBuilder application.

9.1.7 Structures

9.1.7.1 Definition and declaration of structures

Defining a structure in the Structure painter or an object painter is supported:

- If the structure is defined in the Structure painter, the structure is global structure, and instances of the structure can be declared in a script or in an object's instance variable.
- If the structure is defined in an object painter, the structure is an object structure, and instances of the structure can only be declared in the object's instance variables and script. An object structure can be defined in any of the following painters: Application, Window, Menu, Function or UserObject.

Declaring global or local instances for global structures is supported. Declaring instance and local instances for object structures is also supported. Several instances for a structure can be declared at one time.

Example: `str_emp_data str_emp1, str_emp2`

Declaring structure arrays is supported. The arrays can be single or multi-dimensional, variable-sized or fixed sized. The supported features for structure arrays are the same as the supported features for any other arrays. For example:

```
s_employee lstr_temp[10]
s_employee lstr_employee[]
```

If an object structure's name is identical to that of a global structure, the object structure has higher priority than the global structure (same as in PowerBuilder).

9.1.7.2 Referring to structure variables

The following syntaxes are supported:

```
structurename.variable //referring to a structure variable by using dot notations
objectname.structurename.variable //referring to a variable of an object structure
```

For example:

```
str_emp1.emp_lname = "Jones"
This.str_cust1.name
w_customer.str_cust1.name
```

9.1.7.3 Initialization and assignment of structure variables

Initialization and assignment

For each variable of a structure, the variable will be initialized with default values according to its data type. If it is a string type variable, the default value is ""; if it is numeric type variable, the default value is 0.

A structure instance can be assigned when it is declared or after it is declared. Example:

```
s_employee lstr_ouremployee1, lstr_outemployee2
s_employee lstr_youremployee = lstr_ouremployee1
lstr_youremployee = lstr_ouremployee2
```

As in PowerBuilder, when assigning one structure to another, the whole structure is copied so that there are two copies of the structure.

9.1.7.4 Passing structures as arguments

Passing structures as arguments by value, by reference, or as read-only is supported.

9.1.7.5 Complex structures

The following types of complex structures are supported:

- Structures which have one or more of their structure variables being referred to another structure.
- One or more structure variables which are autoinstantiated or non-autoinstantiated objects. Example:

```
s_employee lstr_employee[] //the structure s_employee is defined with a
DataWindow type variable
long ll_employee
lstr_employee[1].adw_employee=w_employee.dw_1
ll_employee=lstr_employee[1].adw_employee.Retrieve()
```

- One or more structure variables which are arrays that have no unsupported features.

9.1.7.6 Unsupported

- Cross-definition of structures. For example, structure A has a structure variable that is structure B, while structure B has a structure variable that is structure A.
- Inheritance of an object structure is unsupported; inheritance of an object structure instances is unsupported as well.
- If one or more member variables in the structure are unsupported data types, Appeon still supports the structure as long as unsupported variables are not used.
- If there are member variables of enumerated data type in the structure, the default values of the enumerated variables are unsupported.

9.1.8 User objects

9.1.8.1 User objects

Important Requirements

- For standard class user objects:
 1. Standard class user objects can only inherit from the following non-visual system objects: DataStore, Transaction Object, DynamicStagingArea and OLEObject.
 2. Non-visual standard class user objects must be defined in a PowerBuilder painter. They can be dynamically created (for example, by using the CREATE statement).
 3. If a non-visual object is a local variable, the Destructor event in the non-visual object cannot be triggered unless there is a Destroy statement for the non-visual object as well.
- For custom class user objects:
 1. Non-visual custom class user objects must be defined in PowerBuilder painter. They can be dynamically created (for example, by using the CREATE statement).
 2. The specification of EAServer Project properties is unsupported.
 3. If a non-visual object is a local variable, the Destructor event in the non-visual object cannot be triggered unless there is a Destroy statement for the non-visual object as well.
- For standard and custom visual user objects:
 1. Must be defined in PowerBuilder painter.
 2. The SetFocus function is not supported for custom visual user objects, but is supported for standard visual user objects.

Standard Visual Object is an extension of the visual system object (control), and it is used to customize the function of the visual system object (control).

Supported

- Custom class user objects
- Custom visual user objects
- Standard class user objects
- Standard visual user objects

Unsupported

1. External visual user objects
In the PowerBuilder application, the Destructor event sequence for a user object will be triggered in accordance with the Control[] property of the user object. In the Web application, the Destructor event sequence is unsupported.
2. Nonvisual objects (Custom Class Objects and Standard Class Objects)
To insert nonvisual object(s) between objects (window, user object, NVO, application), you can select any items on the *Insert menu* | *Object menu* of the PowerBuilder painter. However, Apeon does not support this.

9.1.8.2 Autoinstantiated NVO

- Declaring an autoinstantiated NVO
 1. Declaring an autoinstantiated user object creates an instance of that object (just like a structure), and the Constructor events are triggered for the instance variables.
 2. If an instance variable contains an autoinstantiated NVO(b) and the Constructor event is triggered for the instance variable, an instance of NVO(b) is automatically created, and the Constructor events are triggered for the instance variables of NVO(b).

- Assignment for autoinstantiated NVO
 1. When an autoinstantiated object is assigned to another autoinstantiated object, the whole object is copied to the second variable. For example:

```
n_cst_string lnv_string1, lnv_string2
lnv_string2 =
lnv_string1           //lnv_string2 is a
copy of lnv_string 1
```

2. Assigning a NonVisualObject object to an autoinstantiated NVO or a NonVisualObject object is **unsupported** (please note that NonVisualObject is a system object and it is different from an NVO (non visual user object)). For example:

```
NonVisualObject lnv_test           //lnv_test
is a NonVisualObject object

n_cst_string
lnv_string           //n_cst_string is an
autoinstantiated NVO

lnv_test =
lnv_string           //unsupported
```

- Defining autoinstantiated NVO

The following can be included in the definition of an autoinstantiated NVO:

1. Instance variable. The instance variable can be an autoinstantiated NVO, an object, or have the same name as a window instance variable
 2. System function or user defined function or object function
 3. Constructor event, or object event or user-defined event
 4. Using a Destructor event in the definition of an autoinstantiated NVO is unsupported.
- Autoinstantiated NVO array
 1. If the autoinstantiated NVO array is a fixed-size array and the array is declared, instances of each NVO are created, the instance variables of each NVO are instantiated, and the Constructor event is triggered for each instance variable. Example:

```
n_cst_string lnv_string[10]
```

1. If the autoinstantiated NVO array is variable-size array, the NVO instances are not created when the array is declared. When an array element is named (an NVO), the instances of the element and the foregoing elements are created, the instance variables of each NVO are instantiated, and the Constructor event is triggered for each instance variable. Example:

```
n_cst_string lnv_string[]
lnv_string[10].is_source = ls_model
```

- Autoinstantiated NVO as a structure member

When the structure is declared, an instance of the autoinstantiated NVO is created. When the structure is called, the Constructor event is not triggered for the NVO. In PowerBuilder, the Constructor event is triggered when the NVO is first used.

- Inheritance

Inheritance is supported if the ancestor of an autoinstantiated NVO is a nonautoinstantiated NVO.

9.1.8.3 Nonautoinstantiated NVO

- Declaring a non-autoinstantiated NVO
 1. Declaring a non-autoinstantiated user object does not create an object instance until there is a CREATE statement. Declaring an object variable declares an object reference. Example:


```
n_base lnv_base //n_
base is a nonautoinstantiated NVO

lnv_base = Create n_base //Create an
instance of n_base
```
 2. When the object instance is created, the instance variables of the NVO are instantiated, and the Constructor event is triggered for each instance variable.
 3. Instantiating an ancestor variable with an instance of one of its descendants is supported. Example:

```

n_base lnv_base //n_base is a
nonautoinstantiated NVO

lnv_base = Create using
"n_cst_sqlspy" //n_cst_sqlspy is a
descendant of n_base

```

- Assignment for nonautoinstantiated NVO

1. When a non-autoinstantiated object is assigned to another non-autoinstantiated object, a reference to the object instance is copied. Only one copy of the object exists. For example:

```

n_cst_string lnv_string1, lnv_string2
lnv_string2 = lnv_string1 //Both
point to same object instance

```

2. Unlike autoinstantiated NVOs, assigning a NonVisualObject object to a non-autoinstantiated NVO or assign a non-autoinstantiated NVO to a NonVisualObject object is **supported**. For example:

```

NonVisualObject lnv_test //lnv_test
is a NonVisualObject object

n_cst_string
lnv_string2 //n_cst_string is a
non-autoinstantiated NVO

lnv_test =
lnv_string2 //supported

```

- Defining non-autoinstantiated NVOs

The following can be included in the definition of a non-autoinstantiated NVO:

1. Instance variables. An instance variable can be an autoinstantiated NVO, an object, or have the same name as a window instance variable
2. System, user defined, or object functions
3. Constructor, Destructor, object, and user-defined events. The Destructor event cannot be triggered unless there is a Destroy statement for the object as well.

- Non-autoinstantiated NVO array

There can be fixed-size or variable size non-autoinstantiated NVO arrays. When the array is declared, an instance of the object is not created. A non-autoinstantiated NVO will only be created when there is a CREATE statement for the NVO.

- Non-autoinstantiated NVO as a structure member

1. Defining a non-autoinstantiated NVO in a structure is supported. Example:

```

global type str_model from structure
string s_emplid
Date ld_inputday
n_cst_base lnv_base

```

```
end type
```

- When the structure is declared, an instance of the non-autoinstantiated NVO is not created. A non-autoinstantiated NVO will only be created when there is a CREATE statement for the NVO.

9.1.9 Calling functions and events

9.1.9.1 Syntax for calling functions and events

Supported

- The syntax used to call all PowerBuilder functions and events:

```
{objectname.} {type} {calltype} {when} name  
( { argumentlist } )
```

If the *calltype* argument is DYNAMIC, it is unsupported to specify a reference argument in the argumentlist.

- The syntax used to call functions and events in an object's ancestor:

```
{ objectname. } ancestorclass ::{ type } { when } name  
( { argumentlist } )
```

- If the function name is not qualified with an object or a control, PowerBuilder searches for the function and executes the first one it finds that matches the name and arguments. This is supported in Apeon.

Unsupported

- PowerBuilder searches for events in the object's ancestor hierarchy until it gets to the top ancestor or finds an event that overrides its ancestor. However, Apeon searches for events from the object to the window.
- Referring to a global function by using the global scope operator (::) before the function name is unsupported.

9.1.9.2 Triggering & Posting

Triggering	Supported
	Triggering for functions Triggering for events
	Unsupported
	For application and message objects, triggering for functions and events are unsupported.
Posting	Supported
	Posting for functions Posting for events
	Unsupported

	<p>For complex objects, such as application, message, and DataWindow objects, posting for functions and events are unsupported.</p> <p>Posting function B that is called inside function A is unsupported. For example:</p> <pre>Function A () { Post Function B () //unsupported to post function B }</pre> <p>Posting a function containing reference arguments, which is a local variable is unsupported.</p>
Post function	<p>Supported</p> <p>There are three POST syntax supported, and two of them are supported with limitations.</p> <ul style="list-style-type: none"> • Post Close(window) • Post Open(window) <p>limitations: the <i>window</i> argument cannot be an array variable. For example:</p> <pre>//Unsupported window winname[2] Post open(winname[1],....) - Post open(windowvariable)</pre> <ul style="list-style-type: none"> • Post user_function() <p>Limitations: The reference argument cannot be a local variable. For example:</p> <pre>//Unsupported Integer gf_string (ref string as_parm1) Post gf_String(ls_Parm) // ls_parm is a local variable.</pre>

9.1.9.3 Static & dynamic calls

Supported

- Static calls to functions
- Static calls to events
- Dynamic calls to functions
- Dynamic calls to events

Unsupported

- Dynamic calling for overloaded functions is unsupported.
- Dynamically calling a function that has an argument passed by reference is unsupported.

- If a function is dynamically called, its return value cannot be passed as an argument of another function.

9.1.9.4 Overloading, overriding, and extending functions and events

Supported

- Function overriding is supported.
- Extending and overriding events are supported.

Unsupported

- Overloading system functions is unsupported.
- Dynamic calling for overloaded functions is unsupported.
- Overloading a function that has a dot notation as an argument is unsupported. For example, overloading the following function is unsupported:

```
wf_getname(dw_1.object.s_id[1])
```

- Using the local variable AncestorReturnValue in an event of a descendent object is unsupported, unless the event of the descendent object is an extended event from the ancestor object.
- The following scenario is unsupported:

```
In object A (parent object), function g() calls function  
f(type1 arg 1).  
g()  
{  
f(type1 arg1);  
}
```

In object B (child object), function f(type2 arg2) is the overloading function of function f(type1 arg 1), and object B inherits function g() from object A.

- Calling function g() in object B is **unsupported**. In other words, it is unsupported for a child object to call an inherited function from the parent object, where in the parent object, the function calls another function, which has an overloading function in the child object.

9.1.9.5 Passing arguments to functions and events

Passing arguments

There are three ways to pass arguments to functions and events:

- By value
- By reference
- Read-only

Arguments can be passed with one **limitation** that each function or event can have a maximum of 20 arguments. If the number of arguments exceeds 20, the arguments after the 20th argument will be invalid.

Unsupported

- If both function A and function B have an argument passed by reference, calling function A that has one argument calling to function B, while the argument (that is passed by reference) in the two functions uses the same variable, is unsupported.
- Function A has two arguments passed by reference. It is unsupported for the two arguments to use the same variable.
- Passing a property dot notation as the function argument is unsupported if the property refers to an object, however, this can be worked around.
- Passing an argument that is an object property by reference is unsupported.

9.1.9.6 Using return values of functions and events

Supported

To use the return value, assign it to a variable of the appropriate data type or call the function where you can use a value of that data type:

- Return values for built-in PowerScript functions
- User-defined functions or events that have return values
- Return values for system events
- User-defined events that have return values
- The function has parameters of standard data type passed by reference and the return value is used as a condition in RETURN, IF ... THEN, CHOOSE...ASE, or DO ... LOOP statements.
- The function has parameters of object data type passed by value and the return value is used as a condition in RETURN, IF ... THEN, CHOOSE...ASE, or DO ... LOOP statements.
- The return value of one function is used as the parameter of another function, for example, func1(func2()).

To use cascaded call and return values:

- It is supported to get/set the property of an object that is the return value of a function:
Syntax: function.property
For example: ParentWindow().Enabled = TRUE
- It is supported to call the function of an object that is the return value of a function.
Syntax: function1.function2
For example: ParentWindow().Hide()
Ls_test = String(m_main.GetParent().ClassName())

Limitations of using cascaded call and return values:

- It is unsupported to use the DYNAMIC keyword
- Except the first call in the chain of cascaded calls, it is unsupported to use reference arguments.

9.1.10 Document Interface

Important Requirements

The following are limitations for using MDI and SDI:

- When a sheet window is open within MDI, the toolbar of the sheet window will be added as a whole new row below the toolbar of the MDI window. (This is more limited than in PowerBuilder, where you can choose for the sheet toolbar to be added after the MDI toolbar, on the left/right of the screen, to be floating, etc.)
- SDI can have only one menu and one toolbar.

Supported

- Both MDI and SDI are supported.
- An MDI window can have its menu and toolbar, and every sheet window can have its own menu and toolbar, exactly like in PowerBuilder.
- When a sheet window is opened within MDI, the menu of the sheet window will replace the menu of the MDI window. (This is called a menu switch, just as it is in PowerBuilder).
- Appending the names of open sheets (Window list) to a menu item is supported.

Unsupported

- Multiple MDI windows are unsupported. An application cannot have more than one MDI window.
- Multiple MDI frames are unsupported.

9.2 PowerScript statements

Supported

The following table shows the supported PowerScript statements with examples:

Statement	Examples
Assignment	<pre>a = b + 2</pre> <p>Note: There must not be any space between the following operators: ++, --, +=, -=, *=, /=, ^=.</p>
CALL	<pre>CALL ancestorobject {controlname}::event Call super:: eventname</pre> <p>Example: <code>Call super::clicked</code></p> <p>Note: It is supported to use the local variable AncestorReturnValue in an event of a descendent object, if the AncestorReturnValue is generated in a Call Super statement.</p> <pre>Call windowname:: eventname Example: Call w_parent::ue_ok</pre> <pre>Call windowname`Controlname:: eventname Example: Call w_parent`dw_1::ue_retrieve</pre>
CHOOSE...CASE	<pre>CHOOSE CASE testexpression CASE expressionlist statementblock { CASE expressionlist statementblock ... </pre>

	<p>CASE expressionlist statementblock } CASE ELSE statementblock } END CHOOSE</p> <p>Notes: Expressionlist can be one of the following expressions: 1). A single value 2). A list of values separated by commas (such as 2, 4, 6, 8) 3). A TO clause (such as 1 TO 30) 4). IS followed by a relational operator and comparison value (such as IS>5) 5). Functions 6). Any combination of the above with an implied OR between expressions (such as 1, 3, 5, 7, 9, 27 TO 33, IS >42)</p>
CONTINUE	<pre>integer A=1, B=1 DO WHILE A < 10 A ++ IF A < 3 THEN CONTINUE B+=A LOOP</pre>
CREATE	<p>CREATE Support creating object instances for all objects except for PowerObject object, GraphicObject object, WindowObject object, DragObject object, DrawObject object, Function_object object and SystemFunctions object.</p> <p>CREATE USING Support dynamically creating object instances except for:</p> <ol style="list-style-type: none"> 1). Dynamically creating PowerObject object, GraphicObject object, WindowObject object, DragObject object, DrawObject object, Function_object object and SystemFunctions object. Example: UserObject luo_1 luo_1 = create using "PowerObject" 2). Dynamically creating Transaction object. Example: lds_main = Create using "Transaction" 3). Dynamically creating object instances for nested objects. Example: w_main`cb_1 lcb lcb = Create using "w_main`cb_1"
DESTROY	<p>DESTROY DBTrans</p> <p>Supported:</p> <p>The Destroy statement in non-visual system objects (DataStore,</p>

	<p>DynamicStagingArea, and Transaction Object) and non_visual user objects is supported.</p> <p>Example:</p> <pre>Destroy lnv_string // lnv_string = create n_cst_string</pre> <p>The Destroy statement in visual controls and visual user objects is supported.</p> <p>Example:</p> <pre>commandbutton lcb_1 lcb_1 = create using "cb_2" Destroy lcb_1</pre> <p>Note: DESTROY statement is unnecessary for a Web application since an object variable will be automatically destroyed when the Web application exits.</p>
DO...LOOP	<p>Four formats of Do...Loop:</p> <p>Do...Until</p> <p>Do...While</p> <p>Loop...Until</p> <p>Loop...While</p> <pre>DO UNTIL a > 15 a = (a + 1) * b LOOP</pre> <pre>Integer a = 1, b = 1 DO WHILE a <= 15 a = (a + 1) * b LOOP</pre> <pre>Integer a = 1, b = 1 DO a = (a + 1) * b LOOP UNTIL a > 15</pre> <pre>Integer a = 1, b = 1 DO a = (a + 1) * b LOOP WHILE a <= 15</pre> <p>Nesting of Do...Loop statement. Example:</p> <pre>Int li_array[100,50,200] FOR i = 1 to 100 FOR j = 1 to 50 FOR k = 1 to 200 ll_array[i,j,k] = i + j + k NEXT NEXT NEXT</pre>

	<p>Nesting of Do...Loop statement and For...Next statement. Example:</p> <pre>FOR ll_i = 5 to 25 DO UNTIL ll_j > 15 ll_j ++ LOOP ll_j = 1 NEXT</pre>
EXIT	<pre>DO WHILE a < 10 a ++ IF a > 3 THEN EXIT b += a LOOP</pre>
FOR...NEXT	<p>Integer a=1 Integer start, end, increment ... For n=start TO end STEP increment a*=n Next</p> <p>1). End the FOR loop with the keywords END FOR instead of NEXT. Example: <pre>FOR ll_i = 5 to 25 ll_j = ll_j+10 END FOR</pre> </p> <p>2). Using a positive or negative variable for the step increment. Example: <pre>FOR N = 5 TO 25 STEP 5 A = A+10 NEXT</pre> </p> <p>3). Nesting of the For...Next statements or For ... Next statement with Do ... Loop statement. Example: <pre>Int li_array[100,50,200] FOR i = 1 to 100 FOR j = 1 to 50 FOR k = 1 to 200 ll_array[i,j,k]= i + j + k NEXT NEXT NEXT</pre> </p>
HALT	<pre>IF sle_password.Text <> CorrectPassword THEN HALT CLOSE</pre> <p>Notes:</p> <ul style="list-style-type: none"> • There cannot be any code following the Halt statement. • The reserved word HALT is supported.
IF...THEN	<pre>IF num >= 1 THEN result = 1 ELSE result = 0</pre>

	<pre> IF num >= 1 THEN result = 1 ELSEIF num <= -1 THEN result = -1 ELSE result = 0 END IF </pre>
RETURN	RETURN 0

Unsupported

- The following statements are unsupported:

GOTO	THROW	THROWS	TRY...CATCH...FINALLY...END TRY
------	-------	--------	---------------------------------

9.3 Using PowerBuilder Source Editor

A script that is manually added in the Edit Source window will not be converted to the Web. For example, the following script that follows "on w_1.create"; in the Edit Source window will have no effect: `MessageBox ("Welcome", "Welcome to Appeon!")`.

For a descendant object, it is unsupported to declare the event of its ancestor object in the Edit Source window.

10 Embedded SQL

10.1 Database server and data types

Supported

The supported database servers and data types are listed in the table below:

MS SQL Server	Supported		
	binary*	bit	bigint
	char	datetime	decimal
	float	int	image*
	money	nchar	numeric
	ntext	nvarchar	real
	small datetime	small money	smallint
	sql_variant	text	tinyint
	timestamp	unique identifier	varbinary *
	varchar		
You can only use the default value for the user_quoted_identifier property.			
Unsupported The data types that are marked with a "*" symbol can only be used in SELECTBLOB and UPDATEBLOB SQL statements.			
<i>binary & varbinary</i> On the Web, the data length of the two types can exceed 255 digits. Digits that exceed 255 can either be read from or updated to the database by SELECTBLOB or UPDATEBLOB statement.			
Sybase ASE Server	Supported		
	binary	bit	char
	date	datetime	decimal
	float	Image	int
	money	nchar	nvarchar
	numeric	real	smalldatetime
	smallint	smallmoney	text
	timestamp	tinyint	varbinary
	varchar		
Unsupported In a column, data of timestamp type cannot be displayed correctly. Timestamp data type is unsupported when dynamically creating DataWindows.			
Sybase	Supported		

ASA Server	binary*	bit	bigint
	char	date	decimal
	double	float	integer
	long_binary*	long_varchar	money
	numeric	smallint	smallmoney
	time	timestamp	tinyint
	unsigned_bigint	unsigned_int	unsigned_smallint
	varbinary*	varchar	
	Unsupported The data types that are marked with a "*" symbol can only be used in SELECTBLOB and UPDATEBLOB SQL statements.		
Oracle Server	Supported		
	blob *	char	character
	clob*	date	dec
	decimal	float	integer
	interval_day_to_second	interval_year_to_month	long
	nchar	nvarchar2	number
	numeric	raw*	real
	ref cursor	rowid*	timestamp
	timestampwithtimezone	timestampwithlocaltimezone	urowid*
	varchar	varchar2	
	Unsupported The data types that are marked with a "*" symbol can only be used in SELECTBLOB and UPDATEBLOB SQL statements.		
IBM DB2 Server	Supported		
	bigint	blob*	char
	clob*	date	dbclob*
	decimal	double	graphic
	integer	real	smallint
	time	timestamp	varchar
	vargraphic		
Unsupported The data types that are marked with a "*" symbol can only be used in SELECTBLOB and UPDATEBLOB SQL statements. Use Double variables instead of Real variables when obtaining float data from the database.			
Sybase IQ	Supported		

	bigint	binary*	blob*
	char	clob	date
	datetime	decimal	float
	hs_blockmapidentity*	hs_vdoidentity*	hs_vdorecid*
	image*	int	long_binary*
	long_varchar	money	numeric
	oldbit	real	rowid
	smalldatetime	smallint	sysname
	time	timestamp	text
	tinyint	uniqueidentifier*	uniqueidentifierstr
	unsigned bigint	unsigned int	unsigned smallint
	varbinary*	varchar	xact_id
	xml		
	Unsupported The data types that are marked with a "*" symbol can only be used in SELECTBLOB and UPDATEBLOB SQL statements.		
Informix	Supported		
	blob*	boolean	byte*
	char	clob*	date
	datetime	decimal	float
	int8	integer	interval
	lvarchar	money	nchar
	nvarchar	real	serial
	serial8	smallint	text
	time	varchar	
	Unsupported The data types that are marked with a "*" symbol can only be used in SELECTBLOB and UPDATEBLOB SQL statements.		

10.2 Operators

Supported

The following are the SQL operators that can be used:

=	<	>	<=	>=	<>	between	exists
in	like	not between	not exists	not in	not like	is	is not
=all	!=all	<all	>all	<=all	>=all	=any	!=any
<any	>any	<=any	>=any	!=any		~	

Note: Concatenation operator (+) for String datatype is supported.

Unsupported

"^" is unsupported.

10.3 Transaction management statements

Supported

Statement	Examples
COMMIT	Commit
CONNECT	Connect
DISCONNECT	Disconnect
RollBack	Rollback

Unsupported

None.

10.4 Non-cursor statements

Requirements

1. For SQL Server database, Apeon Server cannot recognize SQL statements with double quotation marks if the SET QUOTED_IDENTIFIER is OFF.
2. For SQL Server and ASE database with JDBC driver, use "SET NOCOUNT ON" before the calls to SQL statements when you create a trigger object. Otherwise, update to the database on the Web may fail with an error "Failed to update database due to java.sql.SQLException... cannot insert the value NULL...".
3. In SQL statements, if table names and column names are in double quotation mark, string constant should be quoted as single quotation marks.

Supported

Apeon currently supports six types of non-cursor SQL statements:

DELETE	INSERT	SELECT
SELECTBLOB	UPDATE	UPDATEBLOB

The following table provides more information for supported non-cursor SQL statements:

Type	Supported	Details

SELECT	Retrieval list (SELECT clause)	<p>The retrieval list can be column names, IDs, or aliases. The column names can be expressed as characters (for example, "empid") or as dot notation (for example, emp.empid). Example:</p> <pre>select s_emplid, s_emplname //Column names select e.emp_lname //Table aliases select s_emplid + s_emplname as emplinfo //Column (expression) aliases</pre> <p>The retrieval list can be expressions that are functions, sub queries, arithmetic operators or any combination of columns, constants, and expressions. Example:</p> <pre>select s_emplid + s_emplname as emplinfo //Concatenation of strings select substring(s_emplid + s_emplname,1,3) as emplinfo,getdate() //Function expression select f_salary / 12 as f_persalary //Operation expression</pre> <p>The retrieval list can be an asterisk that represents all columns in one table. Example:</p> <pre>select * //Asterisk symbol representing all columns</pre>
	FROM clause	<p>The FROM clause can contain a single table (view) or multiple tables (views). The table can be expressed using table name or table alias. Example:</p> <pre>from Employee, viewbonus from Employee a, viewbonus b</pre> <p>The tables can be locked or unlocked.</p>
	WHERE clause	The WHERE clause can contain any of the following:
		Retrieval parameters
		Standard comparison operators (=, >, <, <>, >=, <=)
		Standard logical operators (NOT, AND, OR)
		Special operators (UNION, BETWEEN, IN, LIKE, IS NULL)
Join conditions		
Special characters ('&', '~', '[]', '!~!', '""', '!~@~~~~!', etc.)		
HAVING clause	<i>Supported</i>	
GROUP BY clause	<i>Supported</i>	
COMPUTED clause	<i>Supported</i>	

	FOR UPDATE clause	<i>Supported</i>
	Variables list (INTO statement)	The Variables list can be variable(s) or reference(s) to a control property. Example: into: ls_emplid into: sle_1.Text
	Example	SELECT f1, f2, ..., fn into :v1, :v2, ..., :vn FROM table WHERE w1 = :p1 and w2 = :p2.prop and/or ... and/or wn = :pn
SELECTBLOB	SELECTBLOB INTO clause	<i>Supported</i> On the Web, if the result set of SELECTBLOB contains several rows, the first row is returned and the return value of SQLNRows is 1 for every supported DBMS. This is different from PowerBuilder.
	FROM clause	<i>Supported</i>
	WHERE clause	<i>Supported</i>
	Example	Blob Emp_id_pic SELECTBLOB Emp_pic INTO :Emp_id_pic FROM Employee WHERE Employee.Emp_Num = 100 USING Emp_tran ; p_1.SetPicture(Emp_id_pic)
INSERT	INSERT INTO clause	The INSERT INTO clause can be either table (view) name(s) or a column list.
	VALUES clause	The VALUES clause can contain any of the following:
		All supported data types in allowed scope
		Space, special characters ('<', '!', '@', '#', '\$', '%', '^', '&', '*')
	Initial value, a single record, multiple records (up to 500)	
Validation	The validation will be automatically done by the system.	
Example	INSERT INTO table VALUES ("v1", 12.3, :p1, :p2, ..., :pn)	
UPDATE	SET clause	The SET clause can update a single record or multiple records (up to 200). It can contain special characters including '<', '!', '@', '#', '\$', '%', '^', '&', '*'.

	Example	<pre>UPDATE table SET f1 = :p1, f2 = :p2 WHERE w1 = :p3 and/or ... wn = :pn</pre>
UPDATEBLOB	WHERE clause	<p><i>Supported</i></p> <p>If UPDATEBLOB statements update data without a WHERE clause, on the Web all the data will be updated to the database, and the return value of SQLNRows is equal to the number of rows that you have updated.</p> <p>On the Web, the <i>BlobVariable</i> could be NULL.</p>
	Example	<pre>UPDATEBLOB Employee SET emp_pic = :Emp_id_pic WHERE Emp_num = 100 USING Emp_tran ;</pre>
DELETE	DELETE FROM clause	<p>The FROM clause can contain a single table (view) or multiple tables (views). The table can be expressed using a table name or table alias.</p> <p>The DELETE statement can delete a single record or multiple records.</p>
	WHERE clause	The WHERE clause can contain any of the following:
		Retrieval parameters
		Standard comparison operators (=, >, <, <>, >=, <=)
Standard logical operators (NOT, AND, OR)		
Special operators (UNION, BETWEEN, IN, LIKE, IS NULL)		
Join conditions		
Special characters ('&', '~', '[]', '!~!', '""', '!~@~!~!', etc.)		
Example	<pre>DELETE FROM table WHERE f1 = '1' and f2 = :p1 ... fn = :pn</pre>	

Unsupported

- Select statements cannot contain outer join syntax.
- Updateblob statements cannot update multiple blob columns at one time.
- It is unsupported to parenthesize SELECT statements when using EXCEPT ALL or INTERSECT in SQL statements for DB2 database. For example:

```
(SELECT DEP_ID FROM PUB_T_DEPARTMENT) EXCEPT ALL (SELECT
DEP_ID FROM PUB_T_EMPLOYEE)
```

(SELECT DEP_ID FROM PUB_T_DEPARTMENT) INTERSECT (SELECT DEP_ID FROM PUB_T_EMPLOYEE) //Unsupported

SELECT DEP_ID FROM PUB_T_DEPARTMENT EXCEPT ALL SELECT DEP_ID FROM PUB_T_EMPLOYEE
 SELECT DEP_ID FROM PUB_T_DEPARTMENT INTERSECT SELECT DEP_ID FROM PUB_T_EMPLOYEE //Supported

- If the database is Sybase ASE or SQL Server, SQL statements can contain column name(s) enclosed in double quotes (same as in PowerBuilder). If the database is not Sybase ASE or SQL Server, SQL statements cannot contain column name(s) enclosed in double quotes (same as in PowerBuilder).

Using a COMPUTE BY clause in the SELECT statement is unsupported if an application uses a JDBC driver.

10.5 Cursor statements

Supported

- The following statements for retrieving and updating cursors are supported:

CLOSE	DECLARE	DELETE	FETCH
FETCH FIRST	FETCH LAST	FETCH NEXT	FETCH PRIOR
OPEN	UPDATE		

- Local cursors are supported.
- Global and instance cursors are supported.

Unsupported

- The Cursor SQL statement UPDATE Where Current is unsupported.

```
Syntax:
UPDATE TableName SetStatement WHERE CURRENT OF CursorName;
```

- The Cursor SQL Statement DELETE Where Current is unsupported.

```
Syntax:
DELETE FROM TableName WHERE CURRENT OF CursorName;
```

- If a cursor is declared for retrieving rows from a table, modifying (inserting, deleting, or updating) the table during the cursor open close period is unsupported. Otherwise, the data retrieved is different on the Web than in the PowerBuilder application.

Placing the cursor declaration syntax in a statement block that may not be executed at runtime is unsupported. In PowerBuilder, cursor declaration syntax is treated the same way as variable declaration, so the syntax will not be skipped although the statement block is not executed. However, in the Web application, the syntax may be skipped and cause errors.

10.6 Database stored procedures

SQL statements	Supported	
	Statements	Examples

	<p>DECLARE</p> <p>EXECUTE</p> <p>FETCH</p> <p>FETCH FIRST</p> <p>FETCH LAST</p> <p>FETCH NEXT</p> <p>FETCH PRIOR</p> <p>CLOSE</p> <p>Notes</p> <p>Input & output parameters are supported.</p> <p>Return value for stored procedure is supported.</p>	<p>DECLARE lproc_1 PROCEDURE FOR StoreProcedure @f1 = :p1 IN, @f2 = :p2 OUT, ... @fn = :pn USING trans_obj;</p> <p>EXECUTE lproc_1;</p> <p>FETCH lproc_1 INTO :v1, :v2, :v3, ...;</p> <p>FETCH FIRST lcur_1 INTO :v1, :v2, :v3, ...;</p> <p>FETCH LAST lcur_1 INTO :v1, :v2, :v3, ...;</p> <p>FETCH NEXT lcur_1 INTO :v1, :v2, :v3, ...;</p> <p>FETCH PRIOR lcur_1 INTO :v1, :v2, :v3, ...;</p> <p>CLOSE lproc_1;</p>
Declaration syntax	<p>All Supported, Except That...</p> <p>Placing the stored procedure declaration syntax in a statement block that may not be executed at runtime is unsupported. In PowerBuilder, stored procedure declaration syntax is treated the same way as variable declaration, so the syntax will not be skipped although the statement block is not executed. However, in the Web application, the syntax may be skipped and cause errors. For example:</p> <pre> if li_length = 10 then DECLARE proc_empl PROCEDURE FOR dbo.java_debug_request debugger = a1, request = a2, out_request = a3 ; ... End if OPEN proc_empl; FETCH proc_empl INTO :ls_emplid; ... </pre> <p>In a Web application with the above syntax, if the li_length is not 10, the cursor declaration syntax cannot be read, and errors occur.</p>	
Arguments of stored procedures	<p>Supported</p> <p>Input, output & inout parameters are supported. The data type of the parameters should match the corresponding data type in database when you declare a stored procedure.</p> <p>Multiple result sets for a stored procedure.</p> <hr/> <p>Unsupported (for Oracle database)</p> <p>Parameters can not have default values</p> <p>Parameters cannot be expressions.</p>	

Calling stored procedures	<p>Supported</p> <p>Apeon provides nearly full support for calling stored procedures, except for the following requirements and unsupported features.</p> <p>Requirements (For Informix database)</p> <p>The sequence of input parameters must stay the same as that in the Informix database. For example:</p> <pre>Create procedure ProcName(Variable1 int,Variable2 varchar(10), ...) return varchar(10) End procedure ProcName Call ProcName(10,"Apeon", ...)</pre>
	<p>Unsupported</p> <p>Using an expression as a parameter for calling to an Oracle stored procedure.</p> <p>Using default values for parameters in an Oracle stored procedure.</p> <p>Apeon does not support calling an ASA stored procedure with output arguments.</p> <p>For all databases, Apeon does not support calling an overloading stored procedure.</p>

10.7 Dynamic SQL

Supported

- Dynamic SQL Format 1: executing a SQL statement does not produce a result set and does not require input parameters. Example code:

```
EXECUTE IMMEDIATE :strSQL USING trans_obj;
/*Executing a SQL statement does not produce a result set and does
not require input parameters*/
```

Note:

1. In EXECUTE IMMEDIATE SQL statement, if the number of fetched row(s) is 0, the SQLCODE in the transaction object is 0 in PowerBuilder while it is 100 in Apeon.
 2. Using the syntax EXECUTE IMMEDIATE "set transaction isolation level n" is unsupported.
- Dynamic SQL Format 2: executing a SQL statement that does not produce a result set but does require input parameters. Example code:

```
INT emp_id = 56
String fname = "jack";
PREPARE sqlsa FROM "Delete From employee Where emp_id=? And
fname=?"
EXECUTE sqlsa USING :emp_id, :fname;
/*Executing a SQL statement that does not produce a result set but
does require input parameters*/
```

- **Dynamic SQL Format 3:** Use this format to execute a SQL statement that produces a result set in which the input parameters and result set columns are known at compile time. Example code:

```

DECLARE Cursor | Procedure DYNAMIC CURSOR | PROCEDURE
FOR DynamicStagingArea ;
PREPARE DynamicStagingArea FROM SQLStatement {USING
TransactionObject} ;
OPEN DYNAMIC Cursor {USING ParameterList} ;
EXECUTE DYNAMIC Procedure {USING ParameterList} ;
FETCH Cursor | Procedure INTO HostVariableList ;
CLOSE Cursor | Procedure ;
/*Use this format to execute a SQL statement that produces a
result set in which the input parameters and result set columns
are known at compile time*/

```

Note: The default Transaction object name SQLCA is supported.

- **Dynamic SQL Format 4:** executing a SQL statement that produces a result set in which the number of input parameters, result set columns, or both, are unknown at compile time. Example code:

```

DECLARE Cursor | Procedure DYNAMIC CURSOR | PROCEDURE
FOR DynamicStagingArea ;
PREPARE DynamicStagingArea FROM SQLStatement { USING
TransactionObject} ;
DESCRIBE DynamicStagingArea INTO DynamicDescriptionArea ;
OPEN DYNAMIC Cursor | Procedure USING DESCRIPTOR
DynamicDescriptionArea ;
EXECUTE DYNAMIC Cursor | Procedure USING DESCRIPTOR
DynamicDescriptionArea ;
FETCH Cursor | Procedure USING DESCRIPTOR DynamicDescriptionArea ;
CLOSE Cursor | Procedure ;
/ *Use this format to execute a SQL statement that produces a
result set in which the input parameters and result set columns
are unknown at compile time*/

```

Notes:

1. The default transaction object name SQLDA is supported.
2. If you need a DynamicStagingArea variable other than SQLSA, you must declare it and instantiate it with the CREATE statement before using it.

Difference

Calling Fetch Prior when the cursor position is on the first row or calling Fetch Next when the cursor position is on the last row returns a different sqlbcode than in PowerBuilder and on the Web.

11 Functions

11.1 System functions

Supported

The following are the categories with some or all of their functions supported:

Array Functions	Blob Function	Data Type Checking and Conversion Functions
Date, Day, and Time Functions	DDE Client Functions	File Functions
International Functions	Miscellaneous Functions	Numeric Functions
Print and Printer Functions	Registry Functions	String Functions
System and Environment Functions	Timing Functions	Window Functions

Unsupported

- The following are the unsupported system function categories that do not contain any supported individual functions:

Class Definition Functions	DDE Server Functions	Garbage Collection Functions
Help Functions	Library Functions	Shared Object Functions (SharedObject Functions)
Tracing Functions		

- Having a parent object dynamically call a system function that is overloaded by its child object is unsupported.

11.1.1 Supported types

Supported

The following are the categories with some or all of their functions supported:

Array Functions	Blob Function	Data Type Checking and Conversion Functions
Date, Day, and Time Functions	DDE Client Functions	File Functions
International Functions	Miscellaneous Functions	Numeric Functions
Print and Printer Functions	Registry Functions	String Functions

System and Environment Functions	Timing Functions	Window Functions
----------------------------------	------------------	------------------

Unsupported

The following are the unsupported system function categories that do not contain any supported individual functions:

Class Definition Functions	DDE Server Functions	Garbage Collection Functions
Help Functions	Library Functions	Shared Object Functions (SharedObject Functions)
Tracing Functions		

- Having a parent object dynamically call a system function that is overloaded by its child object is unsupported.

11.1.1.1 Array functions

Function	Support Level	Coding Examples
LowerBound	Supported	Int b[-5 to 2,5],a[5],c[]
		li_lower = LowerBound(b,2)
		li_lower = LowerBound(b,1)
		li_lower = LowerBound(a)
		li_lower = LowerBound(a,1)
		li_lower = LowerBound(c)
UpperBound	Supported	Int b[-15 to -5,5],a[5],c[]
		li_upper = UpperBound(b,2)
		li_upper = UpperBound(b,1)
		li_upper = UpperBound(a)
		li_upper = UpperBound(a,1)
		li_upper = UpperBound(c)

11.1.1.2 Blob functions

Function	Support Level	Coding Examples
Blob	Supported	blob lb_data = blob("convert string to blob")

		string ls_data
		lb_data = blob(ls_data)
BlobEdit	Supported	ulong ll_var
		blob lb_data
		ll_var = BlobEdit(lb_data, 1, "String")
		ll_var = BlobEdit(lb_data, 3, 12345)
		ll_var = BlobEdit(lb_data, 9, 2004-12-03)
		ll_var = BlobEdit(lb_data, 1, blob("blob"))
		Note: The <i>data</i> argument cannot be Any data type.
BlobMid	Supported	blob lb_data,lb_subdata
		lb_subdata = BlobMid(lb_data,3)
		lb_subdata = BlobMid(lb_data,5)
Len	Supported	long ll_len
		blob lb_data,lb_subdata
		ll_len = len(lb_data)
		ll_len = len(lb_subdata)
Lenw	Supported	long ll_len
		blob lb_data,lb_subdata
		ll_len = LenW(lb_data)
		ll_len = LenW(lb_subdata)
String	Supported	blob lb_data,lb_subdata
		string ls_data
		ls_data = string(lb_data)
		ls_data = string(lb_subdata)

11.1.1.3 Data type checking and conversion functions

Function	Support Level	Coding Examples
Asc	Supported	li_test = Asc('adfsd')
		li_test = Asc('~nern')
		li_test = Asc(ls_test)
		li_test = Asc(wf_getstring())
		//wf_getstring() is a function that has a string return value
Char	Supported	ls_test1 = Char(65)
		ls_test1 = Char("~n~~djfkndl")
Dec	Supported	ldec_test = Dec("1.234567")
		//Return the string as a decimal
		ldec_test = Dec("1234567")
		ldec_test = Dec(dw_1.object.data[1,2])
Double	Supported	ldb_test = Double("78.7956")
		//Return the string as a double
Integer	Supported	li_test = Integer("93")
		//Return the string as an integer
Long	Supported	ll_test = long("99.88")
		//Return the string as a long
		ll_test = long (16119,26930)
		//Convert the two UnsignedIntegers into a long
Real	Supported	lr_test = Real ("88.56")
		//Return the string as a real
Date	Supported	ld_test = Date(ldt_test)
		//ldt-test is a datetime variable
		ld_test = Date(now())

		ld_test = Date(ls_test)
		//ls_test is a string variable
		ld_test = Date (2003, 4, 1)
		Note: if the argument contains an invalid date, Date returns NULL in PowerBuilder, whereas in the Web application, it returns an empty string ("").
DateTime	Supported	ldt_test = Datetime(ld_test)
		ldt_test = Datetime(ld_test, lt_test)
		Note: After conversion, the microsecond portion in the <i>time</i> argument will be omitted.
IsDate	Supported	If IsDate("Jan 32, 1993") = TRUE Then...
		//Determines whether the string is a valid date
IsNull	Supported	Integer a, b...
		lb_value = isnull(a+b)
		//If the value of expression a+b is null, the lb_value is set as True; otherwise, False
IsNumber	Supported	If IsNumber("23.45") Then ...
		//Return True since the value of the string is a number
		Note: PowerBuilder does not support 1E123 (more than two digits after E). Functions such as IsNumber ("1E123") in PowerBuilder return FALSE, but after conversion, IsNumber returns TRUE.
IsTime	Supported	IsTime(timevalue)
		If IsTime("23: 11") Then ...
		//hh:mmIf
		IsTime("23: 11:33") Then ...
		//hh:mm:ssIf
		IsTime("23: 11:33.123456") Then ...

		//hh:mm:ss.xxxxxx
String	Supported	String(<i>data</i> {, <i>format</i> })
		ls_test = string(1993-05-17, "mm/dd/yyyy")
		//Convert a date to string
		ls_test = string(07:12:28,"hh:mm:ss")
		//Convert a time to string
		ls_test = string(44.56, "\$#,##0.00")
		//Convert a numeric to string
		ls_test = string("gf", "@*@")
		//Convert a string to formatted string
		Notes: 1. String(<i>data</i> , <i>format</i>) cannot be used in DataWindow property expression. 2. When a format is specified in the String function, make sure that the format is one of the supported types for EditMask. The string argument cannot contain special case conversion formats such as "!!!", "aaa" 3. The format argument cannot be [currency].
Time	Supported	lt_test = time("23:00") Note: After conversion, the microsecond portion of the time will be omitted.
LongLong	Unsupported	None

11.1.1.4 Date, Day and Time functions

Function	Support Level	Coding Examples
Day	Supported	li_test = Day(2003-04-01)
		li_test = Day(ld_today)
		li_test = Day(today())
DayName	Supported	ls_test = DayName(2003-04-01)

		ls_test = DayName(ld_today)
		ls_test = DayName(today())
DayNumber	Supported	li_test = DayNumber(2003-04-01)
		li_test = DayNumber (ld_today)
		li_test = DayNumber (today())
DaysAfter	Supported	ll_test = DaysAfter(2003-04-01, 2003-04-01)
		ll_test = DaysAfter(ld_test1, today())
Hour	Supported	Hour(<i>time</i>)
		li_test = Hour(21:00:00)
		li_test = Hour(lt_test)
		li_test = Hour(Now())
Minute	Supported	li_test = Minute (21:00:00)
		li_test = Minute (lt_test)
		li_test = Minute (Now())
Month	Supported	li_test = Month (2003-04-01)
		li_test = Month (ld_today)
		li_test = Month (today())
Now	Supported	ldt_test = Now()
RelativeDate	Supported	ld_test = RelativeDate (2003-04-01, 27)
		ld_test = RelativeDate (ld_test1, li_after)
		ld_test = RelativeDate (today(), li_after)
RelativeTime	Supported	lt_test = RelativeTime (21:00:00, 60)
		lt_test = RelativeTime (ld_test1, li_after)
		lt_test = RelativeTime (now(), li_after)
Second	Supported	li_test = Second (21:00:00)
		li_test = Second (lt_test)

		li_test = Second (Now())
SecondsAfter	Supported	ll_test = SecondsAfter (21:00:00, 09:00:00)
		ll_test = SecondsAfter (lt_test1, now())
Today	Supported	ld_test = Today()
Year	Supported	li_test = Year (2003-04-01)
		li_test = Year (ld_today)
		li_test = Year (today())

11.1.1.5 DDE client functions and events

Function	Support Level	Coding Examples
CloseChannel	Supported	CloseChannel(handle, Handle(w_sheet))
ExecRemote	Supported	None
GetDataDDE	Supported	string Str20
		GetDataDDE(Str20)
GetDataDDEOrigin	Supported	string WhichAppl, WhatTopic, WhatLoc
		GetDataDDEOrigin(WhichAppl, WhatTopic, WhatLoc)
GetRemote	Supported	None
OpenChannel	Supported	long handle
		handle = OpenChannel("Excel", "REGION.XLS")
		Note: It is recommended to determine whether the execution succeeds by its returning -1, -9 or a positive integer. Do not assume that it has failed because it returns a negative integer.
RespondRemote	Supported	IF GetDataDDE(Value) = 1 THEN
		RespondRemote(TRUE)
SetRemote	Supported	None
StartHotLink	Supported	StartHotLink("Any", "MyPBApp", "Any")

StopHotLink	Supported	StopHotLink("Any", "MyPBApp", "Any")
-------------	-----------	--------------------------------------

Event	Support Level	Coding Examples
HotLinkAlarm	Unsupported	None

11.1.1.6 File functions

Important Requirements

On the Web, the reference of the relative path is the Desktop directory, however in PowerBuilder, the reference of the relative path is the target application's current directory.

Function	Support Level	Coding Examples
ChangeDirectory	Supported	int li_return
		li_return = ChangeDirectory(sle_path.text)
CreateDirectory	Supported	int li_return
		li_return = CreateDirectory(sle_path.text)
DirectoryExists	Supported	if DirectoryExists(sle_path.text) then ls_tosting = "The directory exists!"
FileClose	Supported	li_return = FileClose(li_FileNum)
FileDelete	Supported	if FileDelete(sle_filename.text) then ls_tosting = "True:FileDelete "+sle_filename.text
FileExists	Supported	if FileExists(sle_filename.text) then ls_tosting = "The file exists!"
FileLength	Supported	long ll_FileLen
		ll_FileLen = FileLength(sle_filename.text)
FileMove	Supported	li_return = FileMove(sle_filename.text,sle_2.text)
		li_FileNum = FileOpen("C:\DEPT\EMPLOYEE.DAT", StreamMode!, Write!, LockWrite!, Replace!)
FileRead	Supported	blob Emp_Id_Pic
		li_bytes = FileRead(li_FileNum, Emp_Id_Pic)

FileSeek	Supported	ll_FileLen = FileSeek(li_FileNum, 14, FromCurrent!)
		FileSeek(li_FileNum, -14, FromEnd!)
FileWrite	Supported	blob Emp_Id_Pic
		FileWrite(li_FileNum, "New Employees")
		FileWrite(li_FileNum, emp_id_pic)
GetCurrentDirectory	Supported	sle_1.text = GetCurrentDirectory()
		Note: GetCurrentDirectory returns full path name of Desktop on the Web, and it returns the full path name of the current directory in PowerBuilder.
GetFileOpenName	Supported	li_return = GetFileOpenName (sle_po.text, sle_path.text, ls_file[], "TXT", "Text Files (*.TXT)*.TXT, Doc Files (*.DOC)*.DOC")
		li_return = GetFileOpenName (sle_po.text, sle_path.text, sle_filename.text, "TXT", "Text Files (*.txt)*.txt" , sle_2.text)
GetFileSaveName	Supported	li_return = GetFileSaveName (sle_po.text, sle_path.text, ls_file[], "TXT", "Text Files (*.TXT)*.TXT, Doc Files (*.DOC)*.DOC")
		li_return = GetFileSaveName (sle_po.text, sle_path.text, sle_filename.text, "TXT", "Text Files (*.txt)*.txt" , sle_2.text)
GetFolder	Supported	li_return = GetFolder("Folder", sle_path.text)
RemoveDirectory	Supported	li_return = RemoveDirectory(sle_path.text)
FileOpen	Unsupported	None

11.1.1.7 International functions

Function	Support Level	Coding Examples
Reverse	Supported	ls_return = Reverse(ls_test)
IsAllArabic	Unsupported	None
IsAllHebrew	Unsupported	None
IsAnyArabic	Unsupported	None

IsAnyHebrew	Unsupported	None
IsArabic	Unsupported	None
IsArabicAndNumbers	Unsupported	None
IsHebrew	Unsupported	None
IsHebrewAndNumbers	Unsupported	None
FromAnsi	Unsupported	None
FromUnicode	Unsupported	None
ToAnsi	Unsupported	None
ToUnicode	Unsupported	None

11.1.1.8 Miscellaneous functions

Function	Support Level	Coding Examples
Beep	Supported	Beep(5)
ChooseColor	Supported	li_color = ChooseColor(red, custom [])
ClassName	Supported	ls_classname = ClassName(li_array) Note: If the argument is a numeric data type, the function returns number.
DebugBreak	Supported	IF IsNull(auo_ext) THEN DebugBreak()
DraggedObject	Supported	DragObject poj_ctr
		poj_ctr = DraggedObject()
GetFolder	Supported	integer li_result
		li_result = GetFolder("my targets", ls_path)
IntHigh	Supported	li_high = IntHigh(ll_value)
IntLow	Supported	li_low = IntLow(ll_value)
IsValid	Supported	IF IsValid(w_emp) = FALSE THEN Open(w_emp)

KeyDown	Supported	KeyDown (keycode) Notes: 1. In an application deployed with Apeon, the function returns TRUE only when the key is being pressed and FALSE if not. However, the KeyDown function returns TRUE once the key is pressed and FALSE if not. 2. The following enumerated Keycode value are unsupported: KeyApps, KeyLeftWindows, KeyNull, KeyPrintScreen, KeyRightWindows.
MessageBox	Supported	li_return = MessageBox("Title1','Text1',Information!,OK!,1)
PixelsToUnits	Supported	li_return = PixelsToUnits(35, XPixelsToUnits!)
RGB	Supported	ll_color = RGB(255, 255, 255) The RGB value scope supported: 0~16777215 The custom color scope supported: 16777216~33554431 Note: If the color value is -2 or -1 (Transparent), the color display effect will be different in the Web from in the PowerBuilder application. If it is a color unsupported (the color value is less than -2), the color will be replaced with ButtonFace color.
SetNull	Supported	SetNull(ls_test)
SetPointer	Supported	SetPointer(Cross!)
Sleep	Supported	Sleep (5)
UnitsToPixels	Supported	li_return = UnitsToPixels(350, YUnitsToPixels!)
PopulateError	Unsupported	None
SignalError	Unsupported	None

11.1.1.9 Numeric functions

Function	Support Level	Coding Examples
Abs	Supported	ldec_return = Abs(-15725.12)
		ldec_return = Abs(ai_num)

ASin	Supported	ldb_return = ASin(.84147)
		ldb_return = ASin(af_num)
ACos	Supported	ldb_return = ACos(.84147)
		ldb_return = ACos(af_num)
ATan	Supported	ldb_return = ATan(.84147)
		ldb_return = ATan(af_num)
Ceiling	Supported	li_return = Ceiling(3558.5)
		li_return = Ceiling(af_num)
Cos	Supported	ldb_return = Cos(10586.3)
		ldb_return = Cos(af_num)
Exp	Supported	ldb_return = Exp(17438.15)
		ldb_return = Exp(af_num)
Fact	Supported	ldb_return = Fact(14)
		ldb_return = Fact(af_num)
Int	Supported	li_return = Int(8314.11)
		li_return = Int(af_num)
Log	Supported	ldb_return = Log(7628)
		ldb_return = Log(af_num)
LogTen	Supported	ldb_return = LogTen(30975.5)
		ldb_return = LogTen(af_num)
Max	Supported	ldec_return = Max(1019,21120)
		ldec_return = Max(af_num,bf_num)
Min	Supported	ldec_return = Min(1019,21120)
		ldec_return = Min(af_num,bf_num)
Mod	Supported	ldec_return = Mod(32526,8261.15)
		ldec_return = Mod(af_num,bf_num)

Pi	Supported	ldb_return = Pi(20852)
		ldb_return = Pi(af_num)
Rand	Supported	ldec_return = Rand(14888)
		ldec_return = Rand(af_num)
Randomize	Supported	Randomize(0)
Round	Supported	ldec_return = Round(6655.16973,3)
		ldec_return = Round(af_num,b_num)
		Executing Round in SQL statements, decimals will be ignored on the Web.
Sign	Supported	li_return = Sign(0)
		li_return = Sign(-543534)
		li_return = Sign(4563)
		li_return = Sign(af_num)
Sin	Supported	ldb_return = Sin(-751)
		ldb_return = Sin(751)
		ldb_return = Sin(af_num)
Sqrt	Supported	ldb_return = Sqrt(740752012)
		ldb_return = Sqrt(af_num)
Tan	Supported	ldb_return = Tan(28713.4)
		ldb_return = Tan(af_num)
Truncate	Supported	ldec_return = Truncate(21133.24473,3)
		ldec_return = Truncate(af_num)

11.1.1.10 Print functions

Function	Support Level	Example Code
Print	Supported	Print(Job, "Sybase Corporation", 5000) Note: You can call Print in DataWindow to workaround the PrintDataWindow function.

PrintBitmap	Supported	PrintBitmap(li_job, "d:\PB\BITMAP1.BMP", & 50,100, 0,0)
PrintCancel	Supported	PrintCancel(li_job)
PrintClose	Supported	PrintClose(li_job)
PrintDataWindow	Supported	PrintDataWindow(job, dw_EmpHeader)
PrintDefineFont	Supported	PrintDefineFont(li_job, 1, "Courier 10Cpi", -18, & 400, Default!, Decorative!, FALSE, FALSE) Note: Using user-defined fonts that cannot be supported by the system is unsupported.
PrintLine	Supported	PrintLine(li_job,0,5,7500,5,10)
PrintOpen	Supported	Ulong li_job = PrintOpen("Phone List")
PrintOval	Supported	PrintOval(li_job, 4000, 3000, 1000, 1000, 10)
PrintPage	Supported	PrintPage(li_job)
PrintRect	Supported	PrintRect(li_job, 4000,3000, 1000,1000, 125)
PrintRoundRect	Supported	PrintRoundRect(li_job, 1000,1000, 6250,9000, &300,300, 125)
PrintScreen	Supported	PrintScreen(li_job,500,1000,5000,5000)
PrintSetFont	Supported	PrintSetFont(li_job, 2)
PrintSetSpacing	Supported	PrintSetSpacing(li_job, 1.5)
PrintSetup	Supported	PrintSetup()
PrintText	Supported	PrintText(li_job,"PowerBuilder", 3700, 10,3)
PrintWidth	Supported	Int W = PrintWidth(li_job,"PowerBuilder")
PrintX	Supported	Int LocX = PrintX(li_job)
PrintY	Supported	Int LocY = PrintY(li_job)
PrintSend	Unsupported	Note: This function is obsolete for PowerBuilder, so Apeon does not support it either

11.1.1.11 Printer functions

Function	Support Level	Example Code
----------	---------------	--------------

PrintGetPrinter	Supported	String ls_fullstring=PrintGetPrinter()
PrintGetPrinters	Supported	String ls_prntrs = PrintGetPrinters ()
PrintSetPrinter	Supported	String ls_setprn
		PrintSetPrinter (ls_setprn)
PrintSetupPrinter	Supported	PrintSetupPrinter ()

11.1.1.12 Registry functions

Function	Support Level	Coding Examples
RegistryDelete	Supported	RegistryDelete("HKEY_LOCAL_MACHINE\Software\MyApp.Settings\Fonts", "Title")
RegistryGet	Supported	RegistryGet("HKEY_USERS\MyApp.Settings\Fonts", "NameOfEntryNum", RegULONG!, ul_num) Note: The ValueType argument: can be RegString!, RegExpandString!, ReguLong!, ReguLongBigEndian!, or RegMultiString!. cannot be RegMultiString! when implementing this function on the client. cannot be RegBinary!, RegMultiString! or RegLink! when implementing this function on the server.
RegistryKeys	Supported	li_rtn = RegistryKeys("HKEY_CLASSES_ROOT\MyApp", &ls_subkeylist)
RegistrySet	Supported	RegistrySet("HKEY_USERS\MyApp.Settings\Fonts", "NameOfEntryNum", RegULONG!, ul_num) Notes: 1). The ValueType argument: can be RegString!, RegExpandString!, ReguLong!, ReguLongBigEndian!, or RegMultiString!. cannot be RegMultiString! when implementing this function on the client. cannot be RegBinary!, RegMultiString! or RegLink! when implementing this function on the server. 2). The ValueType argument must be used in the syntax.

RegistryValues	Supported	string ls_valuearray[]
		RegistryValues("HKEY_LOCAL_MACHINE\Software\MyApp.Settings\Fonts", ls_valuearray)

When setting the value for a key and value name in the system registry, the value name will be converted into lowercase. Therefore, the functions are case insensitive.

Naming rules for the files, sections, registry keys used in the Registry and Profile functions:

- Must start with an ‘_’ (underscore), a '\$' (dollar sign), or a letter.
- Can include any combination of characters, numbers, '.' (point), '_' (underscore), '-' (dash), space, '\' (backslash) and '\$' (dollar sign).
- Cannot include single quotation marks, quotation marks, '&' (ampersand sign), or '/' (slash).

Functionality difference in different Restisry Mode:

If the Registry Mode in AEM is set to "Use client machine Windows registry", the registry functions are supported the same way as in PowerBuilder.

If the Registry Mode in AEM is set to "Use Appeon registry emulation", Web applications cannot use RegistryGet and RegistryValues for accessing the Windows registry directly. The Registry functions are implemented as workarounds. They do not read from the actual Windows registry. Instead, Appeon creates a mock registry in the Appeon Server database, which initially has no values. Therefore, you must first set a value using the RegistrySet function before trying to use RegistryGet and RegistryValues functions. In this case the registry functions are only supported to work with existing registry keys, and it is not supported to create new keys during application runtime

11.1.1.13 String functions

11.1.1.13.1 String functions for PowerBuilder 10

Functions	Support Level	Coding Examples
AscA	Supported	li_test = AscA('adfsd')
		li_test = AscA(ls_test)
Asc	Supported	li_test = Asc('~nern')
		li_test = Asc(wf_getstring())
CharA	Supported	ls_test = CharA(65)
		ls_test = CharA("~n~~djfkSDL")
Char	Supported	ls_test = Char(65)
		ls_test = Char("~n~~djfkSDL")

Fill	Supported	ls_fill = Fill('-',10)
		ls_fill = Fill('hello', 6)
FillA	Supported	ls_fill = FillA('-',10)
		ls_fill = FillA('hello', 6)
FillW	Supported	ls_fill = FillW('-',10)
		ls_fill = FillW('hello', 6)
LastPos	Supported	ll_return = LastPos('Apeon Apeon', 'Ap')
		ll_return = LastPos('Apeon Apeon', 'Ap',4)
Left	Supported	ls_return = string(Left('z{uDPk7#k',9))
LeftA	Supported	ls_return = string(LeftA('z{uDPk7#k',9))
LeftW	Supported	ls_return = string(LeftW('z{uDPk7#k',9))
LeftTrim	Supported	ls_return = string(LeftTrim(' fdfsd'))
		ls_return = string(LeftTrim(' 34ttrtertgre'))
LeftTrimW	Supported	ls_return = string(LeftTrimW(' fdfsd'))
		ls_return = string(LeftTrimW(' 34ttrtertgre'))
Len	Supported	ls_return = string(Len('gfgdfgsdrgdfg'))
LenA	Supported	ls_return = string(LenA('gfgdfgsdrgdfg'))
LenW	Supported	ls_return = string(LenW('gfgdfgsdrgdfg'))
Lower	Supported	ls_return = Lower("You ARE Welcome!")
Match	Supported	ls_return = string(Match('ABBBC,ABB*C'))
MatchW	Supported	ls_return = string(MatchW('ABBBC,ABB*C'))
Mid	Supported	ls_return = string(Mid('Apeon Apeon',5,2))

MidA	Supported	ls_return = string(MidA('Appeon Appeon',5,2))
MidW	Supported	ls_return = string(MidW('Appeon Appeon',5,2))
Pos	Supported	ls_return = string(Pos('Appeon Appeon','on'))
		ls_return = string(Pos('Appeon Appeon','peon',2))
PosA	Supported	ls_return = string(PosA('Appeon Appeon','on'))
		ls_return = string(PosA('Appeon Appeon','peon',2))
PosW	Supported	ls_return = string(PosW('Appeon Appeon','on'))
		ls_return = string(PosW('Appeon Appeon','peon',2))
Replace	Supported	ls_return = string(Replace('BABE RUTH', 1, 4, 'BABY'))
PeplaceA	Supported	ls_return = string(ReplaceA('BABE RUTH', 1, 4, 'BABY'))
ReplaceW	Supported	ls_return = string(ReplaceW('BABE RUTH', 1, 4, 'BABY'))
Reverse	Supported	ls_return = string(Reverse('vDI%Qv'))
Right	Supported	ls_return = string(Right('Davis', 4))
RightA	Supported	ls_return = string(RightA('Davis', 4))
RightW	Supported	ls_return = string(RightW('Davis', 4))
RightTrim	Supported	ls_return = string(RightTrim('fsdjd'))
		ls_return = string(RightTrim('fsdfjdlfsd'))
RightTrimW	Supported	ls_return = string(RightTrimW('fsdjd'))
		ls_return = string(RightTrimW('fsdfjdlfsd'))
Space	Supported	ls_return = string(Space(14))
Trim	Supported	ls_return = string(Trim('uifusd'))

TrimW	Supported	ls_return = string(TrimW('uifusd'))
WordCap	Supported	ls_return = string(WordCap('how do you do?'))
Upper	Supported	ls_return = string(Upper('how do you do?'))

11.1.1.13.2 String functions for PowerBuilder 9

Functions	Support Level	Coding Examples
Asc	Supported	li_test = Asc('adfsd')
		li_test = Asc('~nern')
		li_test = Asc(ls_test)
		li_test = Asc(wf_getstring())
Char	Supported	ls_test1 = Char(65)
		ls_test1 = Char("~n~djfkSDL")
Fill	Supported	ls_fill = Fill('-+',10)
		ls_fill = Fill('hello', 6)
FillW	Supported	ls_fill = FillW('-+',10)
		ls_fill = FillW('hello', 6)
LastPos	Supported	ll_return = LastPos('Appeon Appeon', 'Ap')
		ll_return = LastPos('Appeon Appeon', 'Ap',4)
Left	Supported	ls_return = string(Left('z{uDPk7#k',9))
LeftW	Supported	ls_return = string(LeftW('z{uDPk7#k',9))
LeftTrim	Supported	ls_return = string(LeftTrim(' fdfsdf'))
		ls_return = string(LeftTrim(' 34ttrtertgre'))
LeftTrimW	Supported	ls_return = string(LeftTrimW(' fdfsdf'))
		ls_return = string(LeftTrimW(' 34ttrtertgre'))
Len	Supported	ls_return = string(Len('gfgdfgsdrgdfg'))

LenW	Supported	ls_return = string(LenW('gfgdfgsdrgdfg'))
Lower	Supported	ls_return = Lower("You ARE Welcome!")
Match	Supported	ls_return = string(Match('ABBBC,ABB*C'))
MatchW	Supported	ls_return = string(MatchW('ABBBC,ABB*C'))
Mid	Supported	ls_return = string(Mid('Appeon Appeon',5,2))
MidW	Supported	ls_return = string(MidW('Appeon Appeon',5,2))
Pos	Supported	ls_return = string(Pos('Appeon Appeon','on'))
		ls_return = string(Pos('Appeon Appeon','peon',2))
PosW	Supported	ls_return = string(PosW('Appeon Appeon','on'))
		ls_return = string(PosW('Appeon Appeon','peon',2))
Replace	Supported	ls_return = string(Replace('BABE RUTH', 1, 4, 'BABY'))
ReplaceW	Supported	ls_return = string(ReplaceW('BABE RUTH', 1, 4, 'BABY'))
Reverse	Supported	ls_return = string(Reverse('vDI%Qv'))
Right	Supported	ls_return = string(Right('Davis', 4))
RightW	Supported	ls_return = string(RightW('Davis', 4))
RightTrim	Supported	ls_return = string(RightTrim('fsdjd'))
		ls_return = string(RightTrim('fsdfjdlfsd'))
RightTrimW	Supported	ls_return = string(RightTrimW('fsdjd'))

		ls_return = string(RightTrimW('fsdfjdlfsd'))
Space	Supported	ls_return = string(Space(14))
Trim	Supported	ls_return = string(Trim('uifusd'))
TrimW	Supported	ls_return = string(TrimW('uifusd'))
WordCap	Supported	ls_return = string(WordCap('how do you do?'))
Upper	Supported	ls_return = string(Upper('how do you do?'))

11.1.1.14 System and environment functions

Function	Support Level	Coding Examples
Clipboard	Supported	string ls_CoName
		ls_CoName = Clipboard()
CommandParm	Supported	String ls_command_line
		ls_command_line = CommandParm()
GetEnvironment	Supported	environment env
		integer rtn
		rtn = GetEnvironment(env)
GetFocus	Supported	Lobj_control = GetFocus()
GetApplication	Supported	application app
		app = GetApplication()
Handle	Supported	Note: the Handle property is partially supported
		Supported syntax: Handle (objectname)
		Unsupported syntax: Handle (objectname, previous)
Post	Supported	Post(Handle(w_date), 277, 3, 0)

		Note: the <i>message#</i> argument cannot be a user defined message number.
ProfileInt	Supported	ProfileInt ("C:\PROFILE.INI", "PB", "maximized", 3)
ProfileString	Supported	ProfileString ("C:\PROFILE.INI", "Employee", "Name", "None")
Run	Supported	run("notepad")
		run("notepad.exe")
		run("C:\winnt\system32\notepad")
		run("C:\winnt\system32\notepad.exe")
Send	Supported	Send(Handle(w_emp), 277, 2, 0)
		Note: the <i>message#</i> argument cannot be a user defined message number.
SetProfileString	Supported	SetProfileString("C:\PROFILE.INI", "Position", "Title", "MGR")
Yield	Supported	Yield()
Restart	Unsupported	None
SignalError	Unsupported	None

Naming rules for the files, sections, registry keys used in the Registry and Profile functions:

- Must start with an '_' (underscore) or a '\$' (dollar sign) or a letter.
- Can include any combination of characters, numbers, '.' (point), '_' (underscore), '-' (dash), space, '\' (backslash) and '\$' (dollar sign).
- Cannot include single quotation mark, quotation mark, '&' (ampersand sign), or '/' (slash).
- Note: files that have the same names cannot be used in Profile functions.

11.1.1.15 Timing functions

Function	Support Level	Coding Examples
CPU	Supported	long ll_start
		ll_start = Cpu()
Idle	Supported	Idle(300)

Timer	Supported	Timer(10, w_main)
Start	Unsupported	None

11.1.1.16 Window functions

Function	Support Level	Coding Examples
Close	Supported	close(w_commandbutton)
		close(parent)
CloseWithReturn	Supported	CloseWithReturn(parent, "return value") Notes: 1. In PowerBuilder, the return value is Null when there are two Nulls in the parameter. In this case, the return value in JavaScript is not Null. 2. On the Web, if the <i>returnvalue</i> parameter is NULL, the return value will be stored in the PowerObjectParm property of the Message object.
Open	Supported	Open(w_main, parent)

<p>OpenSheet</p>	<p>Supported</p>	<p>OpenSheet(w_main,w_parent) OpenSheet(lw_main , "w_main", w_frame , 2, Cascaded!)</p> <p>Notes:</p> <ol style="list-style-type: none"> 1. When opening the same local window variables more than once within the same function, the result is unpredictable. Avoid using the OpenSheet window function in the following manner: <pre>w_11 ww opensheet(ww , "w_111", w_2 , 2, Cascaded!) opensheet (ww, "w_11", w_2, 0, Cascaded!)</pre> <ol style="list-style-type: none"> 2. The following syntax is supported: <pre>OpenSheet (sheetrefvar {, windowtype }, mdiframe {, position {, arrangeopen } })</pre> <ol style="list-style-type: none"> 3. Apeon supports appending a window list to a menu item on the Web. 4. The ArrangeOpen argument can be Cascaded!, Layerd!, Original!. For the argument is Cascaded!, The size of the sheet on the Web is different from in PowerBuilder.
<p>OpenSheetWithParm</p>	<p>Supported</p>	<p>String ls_str OpenSheetWithParm(w_main,ls_str,w_parent)</p> <p>Note: The window type in the function cannot be MDI or MDIHelp.</p> <p>The following syntax is supported:</p> <pre>OpenSheetWithParm (sheetrefvar, parameter {, windowtype }, mdiframe {, position {, arrangeopen } })</pre> <p>Apeon supports appending a window list to a menu item on the Web.</p>

OpenWithParm	Supported	<p>OpenWithParm(w_main,w_parent)</p> <p>Notes:</p> <ol style="list-style-type: none"> 1. The window type in the function cannot be MDI or MDIHelp. 2. On the Web, if the <i>parameter</i> argument is NULL, the argument will be stored in the PowerObjectParm property of the Message object.
--------------	-----------	--

11.1.2 Unsupported types

Unsupported

The following are the unsupported system function categories that do not contain any supported individual functions:

Class Definition Functions	DDE Server Functions	Garbage Collection Functions
Help Functions	Library Functions	Shared Object Functions (SharedObject Functions)
Tracing Functions		

- Having a parent object dynamically call a system function that is overloaded by its child object is unsupported.

11.1.2.1 Class definition functions

This system function type is not supported.

11.1.2.2 DDE server functions

This system function type is not supported.

11.1.2.3 Garbage collection functions

This system function type is not supported.

11.1.2.4 Help functions

This system function type is not supported.

11.1.2.5 Library functions

This system function type is not supported.

11.1.2.6 Shared object (SharedObject) functions

This system function type is not supported.

11.1.2.7 Tracing functions

This system function type is not supported.

11.1.2.8 User functions**Supported**

- Object Functions
- Global Functions
- Global and local External Functions
- Global and local Remote Procedure Calls (RPC)

Unsupported

Dynamic creation of global functions.

12 Events

12.1 Event types

Supported

- User-defined events can be attached to the Application object.
- User-defined events without an ID are supported.
- User-defined events with an ID are supported to the extent that the system message is supported.

Unsupported

- User-defined system messages are not supported.
- The system messages specific for the unsupported system objects/controls are unsupported.
- Selecting the same event ID to declare two events that have different names is not supported.
- Using the local variable AncestorReturnValue in an event of a descendent object is unsupported unless the event of the descendent object is an extended event from the ancestor object, or the AncestorReturnValue is generated in Call Super statement.
- If a system event involves a UI operation, the subsequent event will not be triggered until the UI operation is done.

Avoid the following code, because it will cause system error to the deployed application:

An event in an event sequence destroys some container (such as closing a window or destroying a user object), and then a subsequent event in the sequence calls to the container or controls/objects in the container.

For example, in case `uo_1` contains `cb_1` and there is an event sequence (`ue_1` □ `ue_2` □ `ue_3`), if `uo_1` is deleted in `ue_2` but `cb_1` is called in `ue_3`, the Internet Explorer that runs the sequence will crash.

12.2 System messages

Supported

The following table lists the supported system messages. The Notes in the table indicate that the event and event ID are partly supported:

Object/Control	Event ID	System Event using the ID
All	<code>pbm_constructor</code>	Constructor
	<code>pbm_destructor</code>	Destructor
	<code>pbm_keydown</code>	Key

Controls and windows	pbm_rbuttondown	RButtonDown
Window	pbm_activate	Activate
	pbm_close	Close
	pbm_closequery	CloseQuery
	pbm_deactivate	Deactivate
	pbm_dragdrop	DragDrop
	pbm_dragenter	DragEnter
	pbm_dragleave	DragLeave
	pbm_dragwithin	DragWithin
	pbm_hidewindow	Hide
	pbm_lbuttonclk	Clicked
	pbm_lbuttondblclk	DoubleClicked
	pbm_lbuttondown	MouseDown
	pbm_lbuttonup	MouseUp
	pbm_mousemove	MouseMove
	pbm_open	Open
	pbm_showwindow	Show
	pbm_size	Resize
pbm_timer	Timer	
CheckBox, CommandButton,	pbm_bnclicked	Clicked
Picture PictureBox, RadioButton, StaticText	pbm_bndragdrop	DragDrop
	pbm_bndragenter	DragEnter
	pbm_bndragleave	DragLeave
	pbm_bndragwithin	DragWithin
	pbm_bnkillfocus	LoseFocus

	pbm_bnsetfocus	GetFocus
DropDownListBox/	pbm_cbncillfocus	LoseFocus
DropDownPictureListBox	pbm_cbnmodified	Modified
	pbm_cbnselchange	SelectionChanged
	pbm_cbnsetfocus	GetFocus
DataWindow, DataStore	pbm_dwnchanging	EditChanged
	pbm_dwnbuttonclicked	ButtonClicked
	pbm_dwnbuttonclicking	ButtonClicking
	pbm_dwndragdrop	DragDrop
	pbm_dwndragenter	DragEnter
	pbm_dwndragleave	DragLeave
	pbm_dwndragwithin	DragWithin
	pbm_dwndberror	DBError
	pbm_dwnhscroll	ScrollHorizontal
	pbm_dwnitemchange	ItemChanged
	pbm_dwnitemchange focus	ItemFocusChanged
	pbm_dwnitemvalidationerror	ItemError
	pbm_dwnkillfocus	LoseFocus
	pbm_dwnlbuttonclk	Clicked
	pbm_dwnlbuttondblclk	DoubleClicked
	pbm_dwnrbuttondown	RButtonDown
	pbm_dwnresize	Resize
	pbm_dwnretrieveend	RetrieveEnd
	pbm_dwnretrievestart	RetrieveStart
pbm_dwnrowchange	RowFocusChanged	
pbm_dwnrowchanging	RowFocusChanging	

	pbm_dwnsetfocus	GetFocus
	pbm_dwnupdateend	UpdateEnd
	pbm_dwnupdatestart	UpdateStart
SingleLineEdit, EditMask, MultiLineEdit, StaticText	pbm_endragdrop	DragDrop
	pbm_endragenter	DragEnter
	pbm_endragleave	DragLeave
	pbm_endragwithin	DragWithin
	pbm_enkilfocus	LoseFocus
	pbm_enmodified	Modified
	pbm_ensetfocus	GetFocus
OLEControl	(none)	Clicked
	(none)	PropertyChanged
	(none)	PropertyRequestEdit
	pbm_omnclose	Close
	pbm_constructor	Constructor
	pbm_doubleclick	DoubleClicked
	pbm_omndragdrop	DragDrop
	pbm_omndragenter	DragEnter
	pbm_omndragleave	DragLeave
	pbm_omndragwithin	DragWithin
	pbm_omnrename	Rename
	pbm_omnsave	Save
	pbm_omnsaveobject	SaveObject
	pbm_omnviewchange	ViewChange
HScrollBar, HTrackBar, VScrollBar, VTrackBar	pbm_sbndragdrop	DragDrop
	pbm_sbndragenter	DragEnter

	pbm_sbndragleave	DragLeave
	pbm_sbndragwithin	DragWithin
	pbm_sbnlinedown	LineDown, LineRight
	pbm_sbnlineup	LineLeft, LineUp
	pbm_sbthumbtrack	Moved
	pbm_sbnpagedown	PageDown, PageRight
	pbm_sbnpageup	PageLeft, PageUp
ListBox, PictureListBox	pbm_lbndblclk	DoubleClicked
	pbm_lbndragdrop	DragDrop
	pbm_lbndragenter	DragEnter
	pbm_lbndragleave	DragLeave
	pbm_lbndragwithin	DragWithin
ListView	pbm_lvnbegindrag	BeginDrag
	pbm_lvnbeginlabeledit	BeginLabelEdit
	pbm_lvnbeginrightdrag	BeginRightDrag
	pbm_lvncolumnclick	ColumnClick
	pbm_lvnclicked	Clicked
	pbm_lvndeleteallitems	DeleteAllItems
	pbm_lvndeleteitem	DeleteItem
	pbm_lvndoubleclicked	DoubleClicked
	pbm_lvndragdrop	DragDrop
	pbm_lvndragenter	DragEnter
	pbm_lvndragleave	DragLeave
	pbm_lvndragwithin	DragWithin
	pbm_lvnendlabeledit	EndLabelEdit
	pbm_lvnsetfocus	GetFocus

	pbm_lvnitemchanging	ItemChanging
	pbm_lvnkillfocus	LoseFocus
	pbm_lvnrclicked	RightClicked
	pbm_lvnrdoubleclicked	RightDoubleClicked
Tab	pbm_tnclicked	Clicked
	pbm_tndoubleclicked	DoubleClicked
	pbm_tnrclicked	RightClicked
	pbm_tnrdoubleclicked	RightDoubleClicked
	pbm_tndragdrop	DragDrop
	pbm_tndragenter	DragEnter
	pbm_tndragleave	DragLeave
	pbm_tndragwithin	DragWithin
	pbm_tcnsetfocus	GetFocus
	pbm_tcnsetchanging	SelectionChanging
TreeView	pbm_tvnbegin drag	BeginDrag
	pbm_tvnbeginlabeledit	BeginLabelEdit
	pbm_tvnbeginrightdrag	BeginRightDrag
	pbm_tvnclicked	Clicked
	pbm_tvndeleteitem	DeleteItem
	pbm_tvndoubleclicked	DoubleClicked
	pbm_tvndragdrop	DragDrop
	pbm_tvndragenter	DragEnter
	pbm_tvndragleave	DragLeave
	pbm_tvndragwithin	DragWithin
	pbm_tvnitemexpanded	ItemExpanded
	pbm_tvnitemexpanding	ItemExpanding

	pbm_tvniitempopulate	ItemPopulate
	pbm_tvntkillfocus	LoseFocus
	pbm_tvnselchanged	SelectionChanged
	pbm_tvnselchanging	SelectionChanging
	pbm_tvnssetfocus	GetFocus
	pbm_tvnrclicked	RightClicked
	pbm_tvnenlabeledit	EndLabelEdit
Application	(None)	Open
	(None)	Close
	(none)	Idle
User Object	pbm_uondragdrop	DragDrop
	pbm_uondragenter	DragEnter
	pbm_uondragleave	DragLeave
	pbm_uondragwithin	DragWithin
Menu	(None)	Clicked
	(None)	Selected

Unsupported

The following table lists the unsupported system messages for the supported system objects/controls:

Object/Control	Event ID	System Event using the ID
Controls and windows	pbm_other	Other
Window	pbm_help	Help
	pbm_ddedata	HotLinkAlarm
	pbm_ddeexecute	RemoteExec
	pbm_ddeadvise	RemoteHotLinkStart
	pbm_ddeunadvise	RemoteHotLinkStop

	pbm_dderequest	RemoteRequest
	pbm_ddepoke	RemoteSend
	pbm_syskeydown	SystemKey
	pbm_tbnmoved	ToolbarMoved
DataWindow, DataStore	(none)	Error
	pbm_dwnretrieverow	RetrieveRow
	pbm_dwnvscroll	ScrollVertical
ListView	pbm_lvnititemactivate	ItemActivate
	pbm_lvnsort	Sort
Application	(none)	IdleSystemError
OLEControl	(none)	Error
	(none)	ExternalException
	pbm_help	Help
	pbm_other	Other

12.3 System message (non-standard EventID)

Supported

The following table lists the supported system messages. The Notes in the table indicate that the event and event ID are partly supported:

Object/Control	Event ID (non-standard)	System Event using the ID
All	pbm_contextmenu	ContextMenu
		Note: Event bubbling is unsupported.
DataWindow	pbm_dwnkey	KeyDown
	pbm_dwndropdown	DropDown
	pbm_dwnrbuttonup	Rbuttonup
	pbm_dwnprocessenter	Processenter

	pbm_rbuttondown	
	pbm_lbuttondown	
	pbm_rbuttonup	
	pbm_lbuttonup	

Unsupported

Except for the system messages listed in the table above, all other system messages with non-standard event IDs are unsupported by Appeon. For example, pbm_exchange event is unsupported. Refer to PowerBuilder Help for more details.

13 DataWindow

13.1 DataWindow data sources

Supported

The following five PowerBuilder data sources are supported:

- Quick Select
- SQL Select
- Query
- Stored Procedure
- External

Notes:

1. You can group your stored procedure in ASE Server.
2. If a DataWindow uses an external data source, the content of the DataWindow can only be saved as TXT type.

Unsupported

- Dynamically changing the SQL statement of an external data source is unsupported.
- If a DataWindow uses a stored procedure as its data source and the stored procedure has multiple result sets, the deployed DataWindow always takes the first result set.
- If a DataWindow uses a stored procedure as its data source, the Table.Select *property* cannot be modified, and the SetSQLSelect function cannot be executed.
- If a DataWindow uses an ASE stored procedure as its data source, the AutoCommit property must be set to TRUE and the chain must be off.

13.2 Using SQL statements in DataWindows

Requirements

If you specify the Table Owner in the Specify Update Properties window in PowerBuilder, please make sure the Enclose table and column names in Quote option is not selected in Apeon Developer during the deployment or errors will occur.

If you specify the argument name in Specify Retrieval Arguments window in PowerBuilder, use ASCII characters.

Do not use FOR UPDATE non-cursor statement in the SQL statement for DataWindows.

13.3 DataWindow presentation styles

CrossTab, Composite,	Supported
----------------------	-----------

Freeform, Graph, Grid, Group, Label, N-Up, Tabular, TreeView and RichText presentation styles	<p>DataWindow objects and controls in a DataWindow can be dynamically created (e.g. by using the CREATE statement).</p> <p>CrossTab, Composite, Freeform, Graph, Grid, Group, Label, N-Up, Tabular and RichText style DataWindows are implemented in XML in Apeon deployed applications.</p> <p>Shortcut keys “Ctrl + Home” and “Ctrl + End” are supported in Grid, Freeform, Tabular, TreeView and N-up DataWindows.</p> <p>The Web UI and behavior are identical with the original PowerBuilder DataWindows, except for a few differences listed below.</p>
	<p>Differences</p>
	<p>For all supported DataWindows, it ignores the upper or lower case when sorting by the column names.</p> <p>For <i>CrossTab</i> style</p> <p>Refer to the detailed description of differences CrossTab DataWindow.</p>
	<p>For <i>Composite</i> style</p>
	<p>Refer to the detailed description of differences Composite DataWindow</p>
	<p>For <i>Freeform</i> style:</p>
	<p>When a user adjusts a scrollbar up and down, before the user releases the scrollbar, the DataWindow's contents are scrolled up and down in correspondence with the scrollbar's position. This does not happen in PowerBuilder applications.</p>
<p>For <i>Grid</i> style:</p>	

	<p>1) Web applications support dragging and dropping columns in different locations on a Grid DataWindow. You can select a column header and drag it over other columns and drop the column in any location within the DataWindow. The selected column will be highlighted.</p> <p>2) Sorting a column or selecting all the items by clicking the column header is unsupported.</p> <p>3) In PowerBuilder, the tab sequence in a Grid DataWindow object is always left to right (except for right-to-left operating systems). Changing the tab value to any number other than 0 has no effect. In the Web Grid DataWindow, changing the tab value to any number other than 0 does have an effect. If the user presses Tab, the focus will change according to the predefined tab order.</p> <p>4) The color of the DataWindow border will be changed as the background color changes in the PowerBuilder application. This is not so in the Web application.</p>
	For <i>Group</i> style:
	Refer to the detailed description of differences Grouping in DataWindow .
	For <i>Graph</i> style:
	Refer to the detailed description of differences Graph DataWindow .
	For <i>Label</i> style:
	RowFocusChange event is unsupported.
	For <i>N-Up</i> style:
	When dynamically changing the height of DataWindow, the number of pages is recalculated on the Web but it is not recalculated in PowerBuilder.
	For <i>TreeView</i> style:
	Refer to the detailed description of differences TreeView DataWindow .
	For <i>RichText</i> style:

	<p>Refer to the detailed description of differences RichText DataWindow.</p> <p>For more DataWindow user operation differences, refer to DataWindow enhancements and differences.</p>
Other DataWindow presentation styles	Unsupported
	OLE DataWindow

13.3.1 Composite DataWindow

Nested reports in a Composite DataWindow	Supported			
	Nested reports that are of the following presentation styles can be added into the Composite DataWindow, and the Web Composite DataWindow is implemented in XML:			
	CrossTab	Freeform	Graph	Grid
	Group	Label	N-Up	Tabular
	RichText	TreeView		
	Placing reports in the Group band is supported.			
	Placing Group reports in any DataWindow band is supported.			
Retrieval arguments	Supported			
	Retrieval arguments of the nested report are supported in the composite DataWindow.			
Row-scrolling functions	Unsupported			
	<p>Calling the following DataWindow functions in nested reports is unsupported:</p> <p>ScrollToRow, ScrollPriorPage, ScrollNextPage, ScrollPriorRow ScrollNxtRow</p>			
Performance	Differences			

differences between PowerBuilder and Web	<p>On the Web, if the Y property of a nested report is negative, the header band of the nested report overlaps the detail band.</p> <p>On the Web, if the Trail_Footer property of a nested report control is set to True, the footer band of the nested report will be displayed after the summary band. If the Trail_Footer property is set to False, the footer band will always be displayed at the bottom of current DataWindow band.</p> <p>On the Web, if the content in a Composite DataWindow cannot be displayed in one page horizontally, it will be displayed on a separate page.</p> <p>If the Visible property of a nested report is dynamically changed from TRUE to FALSE, the nested report is counted as visible in PowerBuilder, but invisible on the Web, when the Composite DataWindow breaks pages or counts the height of the detail band.</p> <p>The NewPage property will have effect in all DataWindow bands on the Web. However, it can only have effect in the detail band in PowerBuilder.</p> <p>In the Detail band, the Height.Autosize property cannot be changed by using the Modify function.</p> <p>In PowerBuilder, if the Summary band cannot be displayed within one page, the last row of the detail band and the whole summary band will be displayed in the next new page. However, on the Web, the summary band that cannot be fully displayed within the rest space of the page will be ignored.</p> <p>If you set different print scales for a Composite DataWindow and the nested reports in the DataWindow, the scale for the nested reports will have no effect; instead, the scale for the Composite DataWindow will be effective for the DataWindow and the reports.</p> <p>Scrolling a composite DataWindow using the PageUp/PageDown, UpArrow/DownArrow, or LeftArrow/RightArrow keys is unsupported.</p> <p>If the width of the content of a nested crosstab DataWindow cannot be displayed within the report control, the exceeding content will be ignored on the Web.</p>
---	--

13.3.2 CrossTab DataWindow

Controls in CrossTab DataWindow	Supported			
	CrossTab DataWindow supports the following controls:			
	Button	Text	Picture	GroupBox
	Line	Oval	Rectangle	RoundRectangle
	Computed Field	Graph		

	GroupBox is visible before inputting any data and is invisible after inputting data.			
	Unsupported			
	Column	Report	TableBlob	
Specified properties of CrossTab DataWindow	Supported			
	Crosstab.SourceNames(r/w)	Crosstab.Rows(r)	Crosstab.Columns(r)	Crosstab.Values(r)
	Crosstab.StaticMode (r)	Table.CrosstabData (r)		
	Properties marked with "(r)" can be read - but not changed - in script. Properties marked with "(r/w)" can be read or changed in script. It is supported to use expressions like sum (units for crosstab) only in CrossTab.Values. It is unsupported to use such expressions in other properties that can contain expressions.			
	Unsupported			
	Help.TypeID.SetCrosstab			
Specified methods of CrossTab DataWindow	Supported			
	GetMessageText			
	Unsupported			
	CrosstabDialog			
Specified expression functions of CrossTab DataWindow	Supported			
	CrosstabAvg	CrosstabCount	CrosstabMax	CrosstabMin
	CrosstabSum			
Performance	Differences			

differences between PowerBuilder and Web	<p>Adding columns in CrossTab DataWindow is unsupported, If you add a column in CrossTab DataWindow by "copy" and "paste" in PowerBuilder, the new column will not display on the Web.</p> <p>When the footer band of a Crosstab DataWindow is higher than the DataWindow, detail band displays in PowerBuilder while footer band displays on the Web.</p> <p>If the argument of SetFilter() is NULL, in the Specify Filter dialog it displays the columns in detail band in PowerBuilder. On the Web, it displays the columns in detail band as well as the dynamically generated columns.</p> <p>After dynamically modifying Expression property of a computed field control in CrossTab DataWindow, data will be recalculated according to the new expression on the Web. In PowerBuilder, the data will not be recalculated.</p> <p>When dynamically modifying the properties that affect position, it automatically refreshes the interface on the Web but it does not in PowerBuilder.</p> <p>After retrieving data in CrossTab DataWindow, the position of controls (excluding Text control, Column control and Computed Filed control) will be rearranged on the Web. This is different from in that in PowerBuilder.</p>
Unsupported features	<p>Unsupported</p> <p>SlideLeft property is unsupported in CrossTab DataWindow.</p> <p>For graph control in CrossTab DataWindow, Values cannot contain aggregate functions.</p> <p>For a CrossTab nested report with arguments, retrieval arguments are unsupported.</p>

13.3.3 Grouping in DataWindow

Grouped DataWindow objects can be created in two ways:

- Use the Group presentation style to create a grouped DataWindow object from scratch.
- Take an existing DataWindow object and define grouping

Group bands	Supported
	Group Header band & Group Trailer band
	Unsupported
	None

Group bands options in the PowerBuilder painter	Supported										
	<table border="1"> <thead> <tr> <th>Color</th> <th>Height</th> <th>Group Definition</th> <th>Group Sort*</th> <th>Reset Page Count</th> </tr> </thead> <tbody> <tr> <td>New Page on Group Break</td> <td>Autosize Height*</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	Color	Height	Group Definition	Group Sort*	Reset Page Count	New Page on Group Break	Autosize Height*			
	Color	Height	Group Definition	Group Sort*	Reset Page Count						
	New Page on Group Break	Autosize Height*									
	<p><i>Group Sort</i></p> <p>If a DataWindow contains multiple groups and more than one group is specified with sort criterion in the Group Sort option, the criterion of the group with the largest group band ID will be valid for all the groups on the Web, while the other criteria will be ignored.</p> <p>It is only support to use aggregate functions in Group sort.</p> <p><i>Autosize Height</i></p> <p>the Autosize Height option is unsupported in Group Header and Group Trailer bands.</p>										
	Unsupported										
None											
Properties for the Group keyword	Supported										
	<p>New page</p> <p>SyntaxFromSQL: Group (colnum1, colnum2 NewPage)</p> <p>ResetPageCount</p> <p>SyntaxFromSQL: Group (col1 {col2 ...} ... ResetPageCount)</p>										
	Unsupported										
	None										
DataWindow functions for grouping	Supported										
	Groupcalc *, FindGroupChange										
	<p>Groupcalc</p> <p>In Web applications, the Groupcalc function will be executed automatically when values in a group are changed by dot notation or SetItem method.</p>										
	Unsupported										
	None										

DataWindow expressions for grouping	Supported				
	Last	First	Percent	CumulativeSum	CumulativePercent
	Unsupported				
	Large	Median	Mode	Small	Stdev
	Stdevp	Var	Varp		

13.3.4 Graph DataWindow

Controls in Graph DataWindow	Supported
	Graph DataWindow supports the following controls: Button Column, Computed field, Graph, GroupBox, Line, Oval, OLE, Rectangle, RoundedRectangle, Report, Picture, Text
	Unsupported
	None.
Graph axes	Differences
	Category axis
	<ol style="list-style-type: none"> When setting the categories on the Category axis, you should ensure that the category type and the Category axis type are the same. If they are not, Appeon adopts the first category type you input for this axis and will ignore the one that does not match along with the following. If the expressions are for graphs (for example, sum (units for graph)), do not use it for the category axis.
	Value axis
	When setting the values on the Value axis, you should ensure that the value type and the value axis type are the same. If they are not, Appeon adopts the first category type you input for this axis and will ignore the one that does not match along with the following.
Properties	Differences

for Graph DataWindow	<p>The following DataWindow object properties can be read but not changed in Graph DataWindow:</p> <p>Bandname.Text, HorizontalScrollMaximum, HorizontalScrollMaximum2, QueryMode, QuerySort, Rows_Per_Detail, VerticalScrollMaximum, Zoom</p>
Performance differences between PowerBuilder and Web	<p>Differences</p> <p>On the Web, the UI will be automatically refreshed after executing RowsCopy, RowsDiscard, RowsMove and Sort. This is different from in PowerBuilder.</p> <p>Executing GetObjectAtPoint function of DataWindow control does not return the row number.</p> <p>When multiple series are of different data types, the series will be sorted and displayed according to the order of the String data type.</p> <p>It is unsupported to have a nested report in a Graph DataWindow.</p> <p>It is unsupported to operate the properties before drawing a Graph DataWindow.</p> <p>It is unsupported to use overlays in graphs in the DataWindow.</p> <p>The value of the header height cannot be smaller than 0.</p> <p>In pie graph, if there is negative data in a series, the graph displays differently on the Web from in PowerBuilder. In PowerBuilder, the total percentage exceeds 100%. But Apeon ignores the negative data, it display 100% on the Web.</p>

13.3.5 RichText DataWindow

DataWindow control properties in RichText DataWindow	<p>Unsupported</p> <p>DragAuto property is unsupported.</p>
DataWindow control events in RichText DataWindow	<p>Differences</p> <p>For ItemFocusChange event:</p> <ol style="list-style-type: none"> 1. After retrieving data in a RichText DataWindow on the Web, it triggers the ItemFocusChange event. 2. Executing SetRow function triggers the ItemFocusChange event on the Web. <p>For RowFocusChanged event:</p>

	<p>1. On the Web, executing InsertRow or DeleteRow may not trigger the RowFocusChanged event. This is different from that in PowerBuilder.</p> <p>2. Executing Clipboard, ImportFile and ImportString functions do not trigger the RowFocusChanged event on the Web. This is different from that in PowerBuilder.</p>
	For KeyDown event:
	Some keystrokes do not affect the formatting of documents in RichText DataWindow on the Web. For example, pressing Esc after updating data in an inputfield reverts to the old data in PowerBuilder but pressing the Esc key on the Web does not affect the updating of the data and the changes will be made.
	Unsupported
	PrintStart and Printed events are unsupported.
DataWindow control functions in RichText DataWindow	Supported
	The following functions of DataWindow control are effective only in RichText DataWindow: CopyRTF, InsertDocument, PasteRTF, ShowHeadFoot
	Differences
	For Clicked function, Xpos argument specifies the distance of the pointer from the left side of the DataWindow workspace and Ypos argument specifies the distance of the pointer from the top of the DataWindow workspace on the Web. This is different from the actual behavior of PowerBuilder.
	On the Web, the return value of GroupCalc function in RichText DataWindow is the same as in DataWindows of other presentation styles, which differs from that in PowerBuilder.
	On the Web, some functions have differences in triggering events in RichText DataWindow with that in PowerBuilder. They behave the same as in DataWindows of other presentation styles. These functions are AcceptText, ShareDataOff, SetSQLSelect, SetSort, Sort, SetFilter, Filter, SetFormat, RowsCopy, RowsDiscard and RowsMove.
	Unsupported
	GetClickedRow and GetBandAtPointer functions are unsupported.
DataWindow object	Supported

properties for RichText DataWindow	<p>The following DataWindow object properties have effect only in RichText DataWindow:</p> <p>Detail.Text, Header.Text, Footer.Text, RichText.property</p>
Performance differences between PowerBuilder and Web	<p>Differences</p> <p>For RichText DataWindows, clicking the header or footer bands triggers the GetFocus event on the Web. This is different from that in PowerBuilder.</p> <p>It will not verify the type of the column if it has the initial value when inserting a column on the Web. This is different from that in PowerBuilder.</p> <p>If the value of a column is empty, it displays “ ?? ” in the column on the Web. This is different from that in PowerBuilder.</p> <p>On the Web, if all data is filtered when executing SetFilter, it displays “??”. This is different from that in PowerBuilder.</p> <p>Double clicking a picture in a RichText DataWindow triggers DoubleClicked event on the Web. This is different from that in PowerBuilder.</p> <p>On the Web, if the focus is in a computed field, executing GetColumnName returns only the column name excluding the name of the computed field. This is different from that in PowerBuilder.</p> <p>If there are multiple DataWindows of RichText and other presentation styles in a Window, the last-created RichText DataWindow is always on the top.</p> <p>If the size of the pasted picture is larger than the RichTextEdit control, the horizontal and vertical scroll bars will be automatically displayed on the Web, though the HScrollBar and VScrollBar properties are set to False.</p> <p>Alignment buttons on the Web are only effective if the WordWrap property is set to True. This is different from that in PowerBuilder.</p> <p>The display of scroll bars in RichText DataWindow is the same as that in RichTextEdit control. Also, if the WordWrap property of RichTextEdit control is set to True on the Web, it does not display the horizontal scroll bar.</p>
Unsupported	Unsupported

features

It is unsupported to use Apeon DataWindow menu.

It is unsupported to modify the column value in the pop-up window when right clicking an input field in a RichText DataWindow.

It is unsupported to dynamically create RichText DataWindows. (For example, using Create statement.)

It is unsupported to use RichText DataWindow as NVO, DataStore and DropDownDataWindow.

End identifier is unsupported.

13.3.6 TreeView DataWindow

DataWindow object properties for TreeView presentation style	Supported		
	CollapsedTreeNodeName	DefaultExpandToLevel	ExpandedTreeNodeName
	Indent	Level	SelectNodeByMouse
	ShowNodeConnectLines	ShowLeafNodeConnectLines	ShowTreeNodeIcon
	StateIconAlignMode	TreeNodeIconName	
	Unsupported		
	AccessibleDescription	AccessibleName	AccessibleRole
	QueryMode	QuerySort	QueryClear
	RtoLLayout		
	DataWindow control functions for TreeView DataWindow	Supported	
Collapse		CollapseAll	CollapseAllChildren
CollapseLevel		Expand	ExpandAll
ExpandAllChildren		ExpandLevel	IsExpanded
SelectTreeNode			

DataWindow control events for TreeView presentation style	Supported		
	Collapsing	Collapseed	Expanding
	Expanded	TreeNodeSelecting	TreeNodeSelected
	Unsupported		
	None		
Performance differences between PowerBuilder and Web	<p>Limitations</p> <p>Performance differences between PowerBuilder and Web</p> <p>Limitations</p> <p>All those limitations in Group DataWindow also impact TreeView DataWindows.</p> <p>Treeview DataWindow cannot be dynamically created.</p> <p>Collapsing, Collapsed, Expanding, and Expanded events cannot be triggered if the nested TreeView DataWindows.</p> <p>If the nest level for a TreeView DataWindow is 3, the user may not expand and collapse correctly tree node when clicking the mouse.</p> <p>The height for the Tree.level band cannot be negative value.</p> <p>You can not set a column to current column using SetColumn function if the column cannot be focused.</p> <p>If the ScrollNextPage function is called in the last page, the current row on the Web will be set to the last row. This is different from that in PowerBuilder.</p> <p>On the Web, the SetSort function sorts data before groups data. This is different from that in PowrBuilder.</p> <p>If the Shortcut in PowerBuilder application is the same as the shortcut of IE 7, shorcut in IE 7 has high priority than shortcut settings in PowerBuilder.</p>		
Event	Difference		

sequence difference	<p>The <code>TreeNodeSelecting</code>, <code>TreeNodeSelected</code>, <code>RowFocusChanging</code> and <code>RowFocusChanged</code> events cannot be triggered in the preview mode.</p> <p>Selecting Tree Node on the Web will sequentially trigger the events <code>TreeNodeSelecting</code>, <code>TreeNodeSelected</code>, <code>RowFocusChanging</code>, <code>RowFocusChanged</code>.</p> <p>Clicking the TreeView DataWindow on the Web will trigger the following events in sequence: <code>LButtonDown</code>, <code>RowFocusChanging</code>, <code>RowFocusChanged</code>, <code>TreeNodeDelecting</code> and <code>TreeNodeSelected</code>.</p> <p>On the Web calling ScrollNextPage function will trigger the following event in sequence: <code>Rowfocuschanging</code>, <code>Rowfocuschanged</code>, and <code>ScrollVertial</code>.</p> <p>Collapsing or expanding the tree node will trigger the <code>collapsing(/expanding)</code>, <code>collapsed(/expanded)</code> and <code>Resize</code> events on the Web in sequence.</p> <p>Do not assign values to rows in the grouping column in a loop. You can use dot notation to directly assign values to corresponding rows.</p>
----------------------------	--

13.4 Displaying and validating data

Data validation

Apeon supports data validation in the following two ways:

- Validating whether the data is of a correct data type and in the allowed data scope
- User-defined validation expression

The validation expressions must be DataWindow expressions that Apeon supports (see [DataWindow operators and expressions](#) section).

Define display format

- For Web applications, formats specified in the PowerBuilder application has a higher priority than formats specified in the client machine.
- The format property is supported for column edit style `Edit` and `EditMask`. The display formats are listed in the following table:

Using sections	Supported
	Each type of display format can have multiple sections, with each section corresponding to a form of the number, string, date, or time. Only one section is required. On the web, the sections can be no longer than two. For example:
	<code>\$\$,##0;(\$\$,##0)</code> //Supported
	<code>\$\$,##0;(\$\$,##0);(\$\$,##0.00)</code> //Unsupported
Number	Supported

	<p>Special characters: "#", "0", "%", "\$", "E-", "E+", "e-", "e+", "~"</p> <p>Keywords: [General], [Currency]</p> <p>Separator: ",", "."</p> <p>Default format: ###,###.00</p>
	Unsupported
	<p>If the currency is not US dollars, it is better to specify the format with the dollar symbol replaced with the actual currency symbol to ensure that the currency symbol displays correctly in the deployed Web application. For example, if the format is ¥#,##0.00, then do not rely on the setting of the client machine to add the actual currency symbol.</p>
String	Supported
	<p>Special characters: "d", "dd", "ddd", "dddd", "m", "mm", "mmm", "mmmm", "yy", "yyyy"</p> <p>Separator: "-", ":", "/", ":", " "(space), ", "</p> <p>Separator: Chinese characters for year, month or day can be separators</p> <p>Keywords: [ShortDate], [LongDate]</p> <p>Default format: mm/dd/yy</p>
	Unsupported
	None
Date	Supported
	<p>Special characters: "d", "dd", "ddd", "dddd", "m", "mm", "mmm", "mmmm", "yy", "yyyy"</p> <p>Separator: "-", ":", "/", ":", " "(space), ", "</p> <p>Separator: Chinese characters for year, month or day can be separators</p> <p>Keywords: [ShortDate], [LongDate]</p> <p>Default format: mm/dd/yy</p>
	Unsupported
	None.
Time	Supported

	<p>Special characters: "h", "hh", "m", "mm", "s", "ss", "ffffff", "AM/PM", "am/pm", "A/P", "a/p"</p> <p>Separators: "-", ":", "/", ".", " "(space), ",", "</p> <p>Keywords: [Time]</p> <p>Default format: hh:mm:ss:fff</p>
	Unsupported
	Chinese characters as AM and PM symbols in the Control Panel Regional Options are unsupported for Web applications.
DateTime	Supported
	<p>Separators: "/", "-", " " (space)</p> <p>Default format: mm/dd/yyyy hh:mm:ss:ffff</p> <p>When setting the date format to mm/dd/yyyy, an invalid date will turn to 01/01/1900 in Apeon while 00/00/0000 in PowerBuilder.</p>
	Unsupported
	None.

13.5 Dynamic DataWindow

DataWindow objects and all entities in them have a set of properties. You can look at and change the values of these properties during execution using DataWindow methods or property expressions. You can also create DataWindow objects during execution. Apeon 5.1 supports dynamically creating DataWindows with the following presentation style: FreeForm, Tabular, Grid, Graph, CrossTab, N-Up, Label, Composite, Group. However, dynamically created RichText and TreeView DataWindows are unsupported. Also DataWindow cannot be dynamically created if the database type is Informix.

Important: Configuration required for supporting dynamic DataWindows

Configuration at the database server is required for support of dynamic DataWindows: find the following SQL files in .%Apeon_Server_Installation_Path%\apeon\sql\dynamicsql, and install the corresponding SQL for the database you are using. Note that you should execute the SQL statement in the database console and executing it in PowerBuilder may cause errors.

- install_apeon_syntaxfromsql_ORACLE.sql
- install_apeon_syntaxfromsql_MSSQL.sql
- install_apeon_syntaxfromsql_ASE.sql
- install_apeon_syntaxfromsql_ASA.sql
- install_apeon_syntaxfromsql_DB2.sql

- install_appeon_syntaxfromsql_IQ.sql

You can uninstall the SQL files and disable the support of dynamic DataWindows by running the following files in the same folder:

- uninstall_appeon_syntaxfromsql_ORACLE.sql
- uninstall_appeon_syntaxfromsql_MSSQL.sql
- uninstall_appeon_syntaxfromsql_ASE.sql
- uninstall_appeon_syntaxfromsql_ASA.sql
- uninstall_appeon_syntaxfromsql_DB2.sql
- uninstall_appeon_syntaxfromsql_IQ.sql

Notes:

1. When installing or uninstalling Apeon-provided SQL files, you must use the corresponding SQL executing tool of each database as listed below:

ASA	Interactive SQL
SQL Server	Query Analyzer
DB2	Command Editor
Oracle	SQLPlus WorkSheet
Sybase IQ	Interactive SQL
SQL Server	Query Analyzer

2. After installing the SQL files, you should restart Apeon Server or redeploy the application to disable the caches.
3. When you use the transaction SyntaxFromSQL method, please note the following differences between PowerBuilder and the Web:
 - The length of return value on the Web is different from PowerBuilder, because the return value on the Web carries the names of all columns while the return value in PowerBuilder does not.
 - Text, the object keyword, is unsupported if there are calculation involved in SQL statements, For example, “ Select emp_ID,emp_Name,salary/12,"Dept="+emp_deptname from employee”.
4. When using transaction SyntaxFromSQL method for DB2 on the Web, please note that the generated DataWindow contains no primary key information and cannot be updated on the Web under the following conditions:
 - The database table contains only a simple index.
 - The database table contains a primary key that uses alias.
 - Access a table that contains a primary key as the table owner. (for example, using SELECT * FROM DB2ADMIN.PUB_T_DEPARTMENT)

Executing SaveAs function for dynamically created DataWindows

1. A dynamic DataWindow can only be saved as TXT, HTML or XLS file. Saving it to WMF will give the error "DataObject information does not exist". The reason is that SaveAs to TXT, HTML or XLS is directly executed at the client while SaveAs to other formats is executed by the server. Apeon Server cannot save the DataWindow since it cannot get the dynamically-created SQL.
2. Timestamp data type is unsupported when dynamically creating DataWindows.
3. It is unsupported to have stored procedure with input parameters as the data source when dynamically creating DataWindow.
4. It is unsupported to use EXECUTE statement to execute the SQL when dynamically creating DataWindows. For example,

```
Execute ('select * from employee'). //Unsupported
```
5. If the DataWindow object is created dynamically, the SetSQLSelect function is unsupported.

Methods of dynamically creating DataWindows

1. Using DataWindow Create method
Supported syntax: Integer dwcontrol.Create (string syntax {, string errorbuffer })
Unsupported: Using PBSELECT statement in *syntax* to create DataWindows is unsupported.
2. Using transaction SyntaxFromSQL method for ASE, ASA, SQL Server and Oracle database type.
Supported syntax: transaction.SyntaxFromSQL(sqlselect, presentation, err).

When you use the transaction SyntaxFromSQL method, please note the following differences between PowerBuilder and Web: The length of return value on the Web is different from PowerBuilder, because the return value on the Web carries the names of all columns while the return value in PowerBuilder does not.

Note : It is unsupported to dynamically create DataWindow if the column alias specified in the SQL has the same name as the column name in the quoted table. For example, if a table t1 has the column c1, c2 and c3, syntax like " SELECT c1= c2* 10, c3 FROM t1" is unsupported.

Methods of dynamically modifying DataWindow properties

1. Using property expression
Please refer to DataWindow | [DataWindow operators and expressions](#) for detailed information.

2. Using DataWindow Modify function
Supported syntax: string dwcontrol.Modify(string modstring)

When you use the Modify function, be aware of that: If the Modify function is used to modify the WHERE clause of the DataWindow object's SQL SELECT statement, make sure that the new SQL SELECT statement is correct in syntax. Otherwise, syntax errors can occur because PowerBuilder does not validate the statement whereas JavaScript does.

You should verify that the data types of both sides of the equal symbol in the expression are compatible.

Adding or deleting controls in DataWindow objects

You can use Modify method to create and destroy the controls in a DataWindow object while noting the following issues.

1. When you use this method to adding controls in a DataWindow, the name of the controls should not be same.
2. If it is failed to add or delete a control in the DataWindow object, the error message reported on the Web may be different from in PowerBuilder.

13.6 DataWindow operators and expressions

DataWindow operators	Supported
	Arithmetic operators: +, -, *, /, (), ^ Relational operators: =, >, <, <>, >=, <=, BETWEEN...AND, IN, LIKE, and any of these operators in combination with NOT Logical operators for all data types: NOT, AND, OR Concatenation for string data types: +
	Unsupported
	Subtraction (-). The setting of the DashesInIdentifiers property is ignored. For example, "A-B" always means subtract B from A.
Operator precedence in DataWindow expressions	Difference from PowerBuilder
	The precedence of "AND" is higher than that of "OR" (in PowerBuilder, "AND" and "OR" have the same precedence). The operators >, <, <=, >= are of higher precedence than = and <> (in PowerBuilder, the operators >, <, <=, >=, =, <> are of the same precedence).
	Note: You can use parentheses to make sure the Web application has correct precedence effect as in the PowerBuilder application. Original expression: a OR b AND c Modified expression: (a OR b) AND c
Supported	Supported

& Unsupported expressions	Conditional expressions for property values (such as Visible, X, Y, Width, Height, BackgroundColor, TextColor)					
	Expressions for computed fields					
	Validation rules					
Functions used in DataWindow expressions	Filter and sort criteria					
	Note: The information on the limitations for these types of expression is provided in Specific Requirements for Using Expressions .					
	Unsupported					
Functions used in DataWindow expressions	Series and values in graphs					
	Columns, rows, and values in crosstabs					
	Supported					
	Abs	Acos	Asc	Asin	Atan	Avg
	Bitmap	Ceiling	Describe	Char	Case	Cos
	Count	CrosstabAvg	CrosstabCount	CrosstabMax	CrosstabMin	CrosstabSum
	CumulativePercent	CumulativeSum	CurrentRow	Date	DateTime	Day
	DayName	DayNumber	DaysAfter	Exp	Fact	Fill
	First	GetText	GetRow	Hour	If	Int
	Integer	IsDate	IsNull	IsNumber	IsRowModified	IsRowNew
	IsSelected	IsTime	Last	Large	LastPos	Left
	LeftTrim	Len	Log	LogTen	Long	LookUpDisplay
	Lower	Match	Max	Median	Mid	Min
	Minute	Mod	Month	Now	Number	Page
	PageCount	Percent	Pi	Pos	Rand	Real
RelativeDate	RelativeTime	Replace	RGB	Right	RightTrim	
Round	RowCount	Second	SecondsAfter	Sign	Sin	

RowHeight	Small	Space	Sqrt	Stdevp	Stdev
String	Sum	Tan	Time	Today	Trim
Truncate	Upper	WordCap	Year		
In addition to the above functions, User functions can be used in DataWindow expressions.					
Unsupported					
Mode	PageAcross	PageCountA cross	ProfileInt	ProfileStrin g	Var
VarP					
In addition to the above functions, System functions and External functions cannot be used in DataWindow expressions.					

13.6.1 Specific Requirements for Using Expressions

Rule for all DataWindow expressions

- Expressions involving Null values may arrive at different values in JavaScript from their values in PowerScript. For more details, please refer to the [Null values](#) section.
- Expressions cannot contain "~t".
- Expressions cannot be used in cursor or stored procedure arguments.
- For nested structures, please verify that:
 - It is supported to have single quotes nested with single quotes or double quotes nested with single quotes.
 - It is unsupported to have single quotes nested with double quotes or double quotes nested with double quotes.
 - A multi-layer nested structure may cause problems.
- If using user-defined global functions in DataWindow expressions,
 - Arguments and returned values can only be simple data types: Int/Integer, Boolean, String, Character, Long, UnsignedLong, UnsignedInteger.
 - Arguments cannot be references.
 - If the computed field expression is a computed field or global function, the formatting for the computed field will not take effect on the Web.

Sort and Filter expressions

- Unsupported functions: Avg, Count, Max, Min & Sum
- Behavioral difference: In a Web application, if a Find, Filter or Sort expression contains any special characters (for example, ".", "", "/"), the execution result may

differ from PowerBuilder. In a Web application, the DataWindow rows may display in a different order from PowerBuilder

Property expressions

1. Overlapped quotes, for example, "sdf~"sdf", "dfg'sdf"
2. In the Web application, modifying a DataWindow property in a DataWindow expression may conflict with the settings in the application's source code

Decimal Precision in DataWindow expression

28-digit Decimal is only supported in the following DataWindow expressions: Abs, Avg, CumulativeSum, Medium Sign, Sum.

13.7 DataWindow object and the properties

13.7.1 DataWindow object

Controls in a DataWindow	Supported & Unsupported			
	Refer to the Controls in a DataWindow and their properties section for details.			
Column edit styles	Supported			
	CheckBox	DropDownDataWindow	DropDownListBox	Edit
	EditMask	RadioButtons		
	Unsupported			
	None			
DataWindow bands	Supported			
	All DataWindow bands, including Header band, Detail band, Summary band, Footer band, Trailer band, and Tree.Level band are supported.			
	All DataWindow bands can contain any supported control.			
	Unsupported			
	The height of the Header band cannot be smaller than 0.			
	Notes			
1) Do not use the band name as the name of the DataWindow object, otherwise the execution result on Web will be different from that on PowerBuilder.				
2) If the content in the summary band cannot be displayed fully on the last page of the deployed DataWindow, an additional page will be displayed to show the rest of the content.				

Other	Supported
	DataWindows that contain Blob columns are unsupported.

13.7.2 DataWindow object properties

Supported

If a property is marked as "supported (r)" or "(r)", it means that the property can be read in script, but not changed in script, and setting the property in the painter has no effect in the converted application.

If a property is marked as "supported (r/w)" or "(r/w)", it means that the property can be read or changed in script, and setting the property is effective in the painter.

The table below lists the supported properties for DataWindow object:

Property	What's Supported	What's Unsupported
Attributes	Supported (r)	
Bandname.property	(r/w): Detail.Color, Detail.Height, Detail.Height.Autosize, Detail.Pointer, Footer.Color, Footer.Height, Footer.Pointer, Header.Color, Header.Height, Header.Pointer, Summary.Color, Summary.Height, Summary.Pointer, Trailer.Color, Trailer.Height, Trailer.Pointer; Height.Autosize	
Bandname.Text	Supported (r/w)	
Bands	Supported	
BitmapName	Supported	
Color	Supported (r/w)	
Column.count	Supported	
Crosstab.property	(r/w):Crosstab.SourceNames (r): Rows, Columns, Values, StaticMode, CrossTabData	
Data	Supported	
Data.XML	Supported	
Data.XMLDTD	Supported	
Data.XMLSchema	Supported	

FirstRowOnPage	Supported	
Grid.ColumnMove	Supported	
Grid.Lines	Fully supported (r/w).	
HTMLDW	Supported	
HTMLGen.property	Supported	
HTMLTable.property	Supported	
Label.property	Supported	
LastRowOnPage	Supported (r)	
Message.Title	Supported	
Name	Supported	
Nested	Supported	
Objects	Fully supported (r) Note: In the Web application, the returned object names are listed in a different order from that in PowerBuilder.	
Pointer	Fully Supported (r/w)	
Print.property	Supported (r/w): Color, Collate, Columns.Width, Copies, DocumentName, Duplex, FilenNme, Margin.Botton, Margin.Left, Margin.Right, Margin.Top, Orientation, Page.Range, Page.RangeInclude, Paper.Size, Paper.Source, Preview, Preview.Rulers, Preview.Zoom, PrinterName, Prompt, Quality, Scale Notes: In PowerBuilder, Copies can support printing multiple copies even if the printer does not support it. In Apeon, Copies will take effect only when the printer supports printing multiple copies. The value of MarginBottom, MarginLeft, printMarginTop, MarginRight, PreviewZoom, Scale cannot be negative. In PowerBuilder, if a printer driver does not support scaling, scaling is not supported in either	CanUserDefaultPrinter, ClipText, Columns, CustomPage.Width, CustomPage.Length, OverridePrintJob

	<p>DataWindow printing or preview. However, in the same scenario on the Web, scaling can work in DataWindow preview.</p> <p>When executing PDF printing, if the DocumentName is an empty string, on the Web, the DataWindow will be automatically saved as naming DataWindow.pdf.</p> <p>FileName cannot be an empty string.</p>	
Print.Buttons	Supported	
Print.Preview.Buttons	Supported	
Printer	Supported (r/w)	
Processing	Supported(r)	
ReadOnly	Supported (r)	
Rows_Per_Detail	Supported	
QueryClear	<p>Supported(r/w)</p> <p>Note: By setting the QueryClear property to Yes, events that are triggered by the property on the web are different from in PowerBuilder.</p>	
QueryMode	<p>Supported(r/w):</p> <p>Extending support: For columns of all types (except string), if the QueryMode is set to Yes, calling Find functions will take effect on the Web but not in PowerBuilder.</p> <p>Limitations:</p> <p>In the query mode, calling the following functions is not recommended because the result of calling functions on the Web is different from in PowerBuilder: InsertRow, DeleteRow, ResetUpdate, RowsDiscard, Update, SetItemStatus, Sort, GroupSort, CopyData, MoveData, GetFullState, SetFullState, GetChanges, SetChanges.</p> <p>If the QueryMode is set to Yes, it is unsupported to insert or delete rows in a DataWindow.</p> <p>If the QueryMode is set to Yes, retrieving data by column alias will cause errors on the Web.</p> <p>QueryMode is unsupported in TreeView DataWindow.</p> <p>If the QueryMode is set to Yes, executing</p>	

	DataWindow SaveAs function will not reset the query criteria on the Web.	
QuerySort	Supported(r/w)	
RichText.property	Supported	
Selected	Supported	
Selected.Data	Supported	
Selected.Mouse	Supported (r/w)	If the Selected.Mouse is set to Yes, it is unsupported to select multiple columns by pressing Shift and Ctrl keys.
ShowDefinition	Supported	
SlideLeft	Supported	
SlideUp	Supported	
Sparse	Supported	
Storage	Supported	
StoragePageSize	Supported	
Syntax	Supported	
Syntax.Data	Supported	
Syntax.Modified	Supported	
Table.property	(r/w): Arguments, CrosstabData, Data.Storage, Delete.Argument, Delete.Method, Delete.Type, Filter, Select.Attribute, Update.Argument, Update.Method, Update.Type, Select, GridColumns, Insert.Argument, Insert.Method, Insert.Type,	Table.property is not supported for External data sources.

	<p>Select.Attribute, Sort, Update.Argument, Update.Method, Update.Type, UpdateKeyInPlace, UpdateTable, UpdateWhere</p> <p>(r) :SQLSelect, Procedure</p> <p>In PowerBuilder, the Table.Select property contains the escape character "~" But after the execution of the SetTransObject function, "~" will be removed. During parsing of the application, "~" will be translated into a newline return. After parsing, no "~" will be found in the Table.Select property.</p> <p>In PowerBuilder, by default, the Ascending criteria will be included in the value of the Table.Sort property. However, in the Web application, the Sort is a string containing the sort criteria set by the user. Even though the string will not include the default Ascending criteria, the result will be the same as in PowerBuilder.</p> <p>For a Web application that has more than one user accessing the same tables concurrently, set the DataWindow Table.UpdateWhere property to 1 ("Key and updatable columns") to avoid data loss.</p>	
VerticalScrollMaximum	Supported	
VerticalScrollPosition	Supported	
Zoom	<p>Fully supported (r/w)</p> <p>The effect of a zoom factor appears differently in PowerBuilder than in the Web application.</p>	
Units	<p>Supported</p> <p>Only the type of PowerBuilder units is supported</p>	

Unsupported

None of the Graphical DataWindow properties are supported.

The table below lists the unsupported properties for DataWindow objects:

Data.HTML.Table	Data.XMLWeb	HideGrayLine	Data.XSLFO
Font.Bias	Help.property	HorizontalScrollMaximum	HorizontalScrollMaximum2
HorizontalScrollPosition	HorizontalScrollPosition	HorizontalScrollSplit	HTMLDW

tion	tion2		
OLE.Client.property	Retrieve.AsNeeded	Row.Resize	Table.sqlaction.property
Timer_Interval			

13.7.3 Controls in a DataWindow and their properties

Avoid placing controls above or behind other controls in a DataWindow, as overlapping controls are displayed differently on the Web than in PowerBuilder. For example, if there is a Text control behind a DropDownDataWindow column in a DataWindow, the DropDownDataWindow field cannot be pulled down correctly.

13.7.3.1 Button control

Supported

If a property is marked as "supported (r)" or "(r)", it means that the property can be read - but not changed - in script, and properties set in the painter will not take effect in the converted application.

If a property is marked as "supported (r/w)" or "(r/w)", it means that the property can be read or changed in script, and properties set in the painter will take effect.

Property	What's Supported	What's Unsupported
Action	(r/w): User Defined, Retrieve(Yield), Retrieve, PageNext, PagePrior, PageFirst, PageLast, Sort, Filter, DeleteRow, AppendRow, InsertRow, Update, SaveRowsAs, Print, Cancel, Preview, PreviewWithRulers, QueryMode, QuerySort, Query, Clear	This property cannot be set by using DataWindow expression.
Attributes	Supported (r)	None
Background.property	Fully supported (r/w): Color, Mode This property can be set by using DataWindow expression.	None
Band	Supported (r): Band, Background, Foreground	None
Color	Fully supported (r/w)	This property can be set by using DataWindow expression.
DefaultPicture	Supported (r/w)	This property cannot be set by using

		DataWindow expression.
Filename	Fully supported (r/w)	None
Font.property	(r/w): Charset, Escapement, Face, Family, Height, Italic, Pitch, Strikethrough, Underline, Weight Note: 1) When the value of Escapement is a negative number, the text will not be rotated. 2) The text rotated can be displayed out of the control in PowerBuilder, but not in the Web application.	Width
Height	Fully supported (r/w) This property can be set by using DataWindow expression.	
HTextAlign	Fully supported (r/w)	
Name	Supported (r)	
Pointer	Fully supported (r/w)	
SlideLeft	Supported (r/w)	
SlideUp	Supported (r/w)	
Tag	Fully supported (r/w)	
Text	Fully supported (r/w)	
Type	Supported (r)	
VTextAlign	Supported (r/w)	
Visible	Fully supported (r/w) This property can be set by using DataWindow expression.	
Width	Fully supported (r/w) This property can be set by using DataWindow expression.	

X	Fully supported (r/w) This property can be set by using DataWindow expression.
Y	Fully supported (r/w) This property can be set by using DataWindow expression.

Unsupported

HideSnaked	Moveable	Resizable	SuppressEventProcessing
------------	----------	-----------	-------------------------

13.7.3.2 Column control

Supported

If a property is marked as "supported (r)" or "(r)", it means that the property can be read - but not changed - in script, and properties set in the painter will not take effect in the converted application.

If a property is marked as "supported (r/w)" or "(r/w)", it means that the property can be read or changed in script, and properties set in the painter will take effect.

Property	What's Supported	What's Unsupported
Accelerator	Fully supported (r/w)	None
Alignment	Left!, Center!, Right!, Justify! In PowerBuilder, when the Edit.password property is YES and the alignment of the column's text is center-aligned, the text is left-aligned when it is being edited. After the editing, the text is center-aligned. In the Web application, the text is always center-aligned.	None
Attributes	Supported (r)	None
Background. <i>property</i>	Fully supported (r/w): Color, Mode	None
Band	(r): Band	None
BitmapName	Fully supported (r/w)	None

Border	(r/w): NoBorder!, Box!, Lowered!, Raised!, ShadowBox!, Underline!, ResizeBorder!	None
CheckBox. <i>property</i>	(r/w): LeftText, Off, On, Other, Scale, Text, 3D, ThreeD	It is unsupported to dynamically change the edit style of a column by modifying this property.
Color	Fully supported (r/w) Set this property using DataWindow expression.	None
ColType	Supported (r) dw_control.object.columnname.coltype dw_control.Describe("columnname.coltype")	None
Criteria. <i>property</i>	Fully supported (r/w)	None
dbName	Supported (r) Example: dw_control.object.columnname.dbName dw_control.Describe("columnname.dbName")	None
dddw. <i>property</i>	(r/w): AllowEdit, AutoHScroll, AutoRetrieve, DataColumn, DisplayColumn, HscrollBar, Limit, Name, NilIsNull, PercentWidth, Required, UseAsBorder, VscrollBar, Case Lines, ShowList. (r): HSplitScroll	It is unsupported to dynamically change the edit style of a column by modifying this property.

	<p>The size and font size of Web DropDownDataWindows are determined by the size of the deployed DataWindow; the width of Web DropDownDataWindows, if set to be smaller than the column width, will be displayed in the same width as the column.</p> <p>When executing Modify function to change the Dddw.property on the Web, the value variable must be contained in single quotation marks. For example,</p> <pre>dw_1.Modify("""emp_id.dddw.name=dw_dddw_dropdown""") //Unsupported</pre> <pre>dw_1.Modify("""emp_id.dddw.name='dw_dddw_dropdown'""") //Supported</pre>	
<i>ddlb.property</i>	(r/w): AllowEdit, AutoHScroll, Case, Limit, NilIsNull, Required, ShowList, Sorted, UseAsBorder, VScrollBar	It is unsupported to dynamically change the edit style of a column by modifying this property.
<i>Edit.property</i>	(r/w): AutoHScroll, AutoSelect, AutoVScroll, Case, CodeTable, DisplayOnly, Format, HscrollBar, Limit, Name, NilIsNull, Password, Required, Style, VscrollBar, FocusRectangle, ValidateCode	<p>Modifying ValidateCode takes no effect.</p> <p>It is unsupported to dynamically change the edit style of a column by modifying this property.</p>
<i>EditMask.property</i>	(r/w): AutoSkip, CodeTable, Mask, ReadOnly, Required, Spin, SpinIncr, SpinRange, UseFormat, FocusRectangle	If a Mask is set as a four-digit string, the column will only retrieve the first four digits of the entered value when you use SetItem to set a value in the EditMask column, which is different from what occurs in PowerBuilder.

	<p>In the Web application, if the Mask of a column is set to [General], the values in the column are all displayed as integers when the column gets focus.</p> <p>In the Web application, if the Mask is set to General by default, the displaying data of a column with the EditMask edit style will be converted into an integer when the column gets focus.</p>	It is unsupported to dynamically change the edit style of a column by modifying this property.
Font. <i>property</i>	<p>(r/w): Charset, Escapement, Face, Family, Height, Italic, Pitch, Strikethrough, Underline, Weight</p> <p>Note: 1) When the value of Escapement is a negative number, the text will not be rotated.</p> <p>2) The text rotated can be displayed out of the control in PowerBuilder, but not in the Web application.</p>	Width
Format	Edit	None
Height	Set the Height property using a DataWindow expression.	None
Height.AutoSize	Supported (r)	None
HideSnaked	Supported (r)	None
Identity	Fully supported (r/w)	None
ID	Supported (r)	None
Initial	Fully supported (r/w)	None
Key	Fully supported (r/w)	None
LineRemove	Supported (r)	None
Multiline	Fully supported (r/w)	None
Name	Supported (r)	None
Pointer	Supported (r/w)	None
Protect	This property can be set by using DataWindow expression.	None
RadioButtons.pr operty	Fully supported (r/w): 3D, Columns, LeftText, Scale	It is unsupported to dynamically change

		the edit style of a column by modifying this property.
RightToLeft	Supported This property is partially supported. It only supports the right-to-left display of characters.	None
SlideLeft	Supported (r/w)	None
SlideUp	Supported (r/w)	None
TabSequence	Fully supported (r/w)	None
Tag	Fully supported (r/w)	None
Type	Supported (r)	None
Update	Fully supported (r/w)	None
Validation	Supported (r)	This property cannot be dynamically changed.
ValidationMsg	Supported (r/w)	None
Values	Supported (r/w) This property is supported for columns with the following edit styles: DropDownListBox, CheckBox, and RadioButtons.	None
Visible	Supported (r/w) This property can be set by using DataWindow expression.	None
Width	Fully supported (r/w) This property can be set by using DataWindow expression.	None
Width.Autosize	Supported (r)	None
X	Supported (r/w)	None
Y	Supported (r/w)	None

Unsupported

Moveable	Resizable	HTML. <i>property</i>
----------	-----------	-----------------------

Notes:

1. Automatically adjusting the height of DropDownDataWindow and DropDownListBox objects in Column controls is unsupported.
2. Make sure that the length of data you specified for Column control in PowerBuilder exactly matches the length of data of the corresponding column in the database.

It is unsupported to nest a DropDownDataWindow with another DropDownDataWindow.

13.7.3.3 Computed Field control**Supported**

If a property is marked as "supported (r)" or "(r)", it means that the property can be read - but not changed - in script, and properties set in the painter will not take effect in the converted application.

If a property is marked as "supported(r/w)" or "(r/w)", it means that the property can be read or changed in script, and properties set in the painter will take effect.

Property	What's Supported	What's Unsupported
Alignment	Fully supported (r/w): Left!, Center!, Right!, Justify!	None
Attributes	Supported (r)	None
Background.property	Fully supported (r/w): Color, Mode	None
Band	Fully supported	None
Border	(r/w): NoBorder!, Box!, Lowered!, Raised!, ShadowBox!, Underline!, ResizeBorder!	None
Color	Fully supported (r/w) This property can be set by using DataWindow expression.	None
ColType	Supported (r)	None

Expression	Supported (r/w) After dynamically modifying this property in CrossTab DataWindow, data will be recalculated according to the new expression on the Web. In PowerBuilder, the data will not be recalculated.	None
Font.property	(r/w):Charset, Escapement, Face, Family, Height, Italic, Pitch, Strikethrough, Underline, Weight Note: 1) When the value of Escapement is a negative number, the text will not be rotated. 2) The text rotated can be displayed out of the control in PowerBuilder, but not in the Web application.	Width
Format	Supported (r/w)	None
Height	Fully supported (r/w) This property can be set by using DataWindow expression.	None
Height.Autosize	Supported	None
HideSnaked	Supported (r)	None
HTML.property	(r): AppendedHTML, Link, LinkArgs, LinkTarget, ValueIsHTML	None
LineRemove	Supported (r)	None
Multiline	Supported (r/w)	None
Name	Supported (r)	None
Pointer	Fully supported (r/w):	None
SlideLeft	Supported (r/w)	None
SlideUp	Supported (r/w)	None
Tag	Fully supported (r/w)	None
Type	Supported (r)	None

Visible	Fully supported (r/w) This property can be set by using DataWindow expression.	None
Width	Fully supported (r/w) This property can be set by using DataWindow expression.	None
Width.Autosize	Supported	None
X	Fully supported (r/w) This property can be set by using DataWindow expression.	
Y	Fully supported (r/w) This property can be set by using DataWindow expression.	None

Unsupported

Moveable	Resizable
----------	-----------

13.7.3.4 Graph control

Supported

If a property is marked as "supported (r)" or "(r)", it means that the property can be read - but not changed - in script, and properties set in the painter will not take effect in the converted application.

If a property is marked as "supported (r/w)" or "(r/w)", it means that the property can be read or changed in script, and properties set in the painter will take effect.

Property	What's Supported	What's Unsupported
Attributes	Fully supported	None
Axis	Fully supported	None
Axis.property	Fully supported	None
Axis.DispAttr	Fully supported	None

BackColor	Fully supported	None
Border	Fully supported	None
Category	Fully supported	None
Category.property	Fully supported	None
Category.DispAttr	Fully supported	None
Color	Fully supported	None
Depth	Fully supported	None
Elevation	Fully supported	None
GraphType	(r/w): AreaGraph!, BarGraph!, BarStackGraph!, Bar3DObjGraph!, BarStack3DObjGraph!, Col3DObjGraph!, ColStack3DObjGraph!, ColGraph!, ColStackGraph!, LineGraph!, PieGraph!, ScatterGraph!	Area3D!, Bar3DGraph!, Col3DGraph!, Line3D!, Pie3D!
Height	Fully supported	None
HideSnaked	Supported	None
Legend	Fully supported	None
Legend.DispAttr.property	Fully supported	None
Name	Fully supported	None
OverlapPercent	Fully supported	None
Perspective	Fully supported	None
Pie.DispAttr.fontproperty	Fully supported	None
Pointer	Fully supported	None
Range	Fully supported	None
Rotation	Fully supported	None

Series	Fully supported	None
Series.property	Fully supported	None
Series.DispAttr	Fully supported	None
ShadeColor	Fully supported	None
SizeToDisplay	Fully supported	None
SlideLeft	Fully supported	None
SlideUp	Fully supported	None
Spacing	Fully supported	None
Tag	Fully supported	None
Title	Fully supported	None
Title.DispAttr.font property	Fully supported	None
Type	Supported	None
Values	Fully supported	None
Values.property	Fully supported	None
Values.DispAttr	Fully supported	None
Visible	Fully supported	None
Width	Fully supported	None
X	Fully supported	None
Y	Fully supported	None

Unsupported

Moveable	Resizable
----------	-----------

13.7.3.5 GroupBox control

Supported

If a property is marked as "supported (r)" or "(r)", it means that the property can be read - but not changed - in script, and properties set in the painter will not take effect in the converted application.

If a property is marked as "supported(r/w)" or "(r/w)", it means that the property can be read or changed in script, and properties set in the painter will take effect.

Property	What's Supported	What's Unsupported
Attributes	Supported (r)	None
Background. <i>property</i>	Fully supported (r/w): Color, Mode	None
Band	Supported (r)	None
Border	Supported (r)	None
Color	Fully supported (r/w)	None
Font. <i>property</i>	(r/w): Charset, Escapement, Face, Family, Height, Italic, Pitch, Strikethrough, Underline, Weight Note: 1) When the value of Escapement is a negative number, the text will not be rotated. 2) The text rotated can be displayed out of the control in PowerBuilder, but not in the Web application.	Width
Height	Fully supported (r/w)	None
HideSnaked	Supported (r)	None
Name	Supported (r)	None
Pointer	Fully supported (r/w):	None
SlideLeft	Supported (r/w):	None
SlideUp	Supported (r/w):	None
Tag	Fully supported (r/w)	None
Text	Fully supported (r/w)	None

Type	Supported (r)	None
Visible	Fully supported (r/w)	None
Width	Fully supported (r/w)	None
X	Fully supported (r/w)	None
Y	Fully supported (r/w)	None

Unsupported

Moveable	Resizable
----------	-----------

13.7.3.6 Line control

Supported

If a property is marked as "supported (r)" or "(r)", it means that the property can be read - but not changed - in script, and properties set in the painter will not take effect in the converted application.

If a property is marked as "supported(r/w)" or "(r/w)", it means that the property can be read or changed in script, and properties set in the painter will take effect.

Property	What's Supported	What's Unsupported
Attributes	Supported (r)	None
Background. <i>property</i>	(r/w): Color, Mode	None
Band	Supported (r)	None
Name	Supported (r)	None
Pen. <i>property</i>	Fully supported (r/w)	None
Pointer	Fully supported (r/w)	None
SlideLeft	Fully supported (r/w)	None
SlideUp	Fully supported (r/w)	None
Tag	Fully supported (r/w)	None

Type	Supported	None
Visible	Fully supported (r/w)	None
X1, X2	Fully supported (r/w)	None
Y1, Y2	Fully supported (r/w)	None

Unsupported

Moveable	Resizable
----------	-----------

13.7.3.7 Oval control

Supported

If a property is marked as "supported (r)" or "(r)", it means that the property can be read - but not changed - in script, and properties set in the painter will not take effect in the converted application.

If a property is marked as "supported(r/w)" or "(r/w)", it means that the property can be read or changed in script, and properties set in the painter will take effect.

Property	What's Supported	What's Unsupported
Attributes	Supported (r)	None
Background. <i>property</i>	Fully supported (r/w): Color, Mode	Note
Band	Supported (r)	Note
Brush. <i>property</i>	(r/w): Color, Hatch	Note
Height	Supported (r/w)	Note
HideSnaked	Supported (r)	Note
Name	Fully supported	Note
Pen. <i>property</i>	(r/w): Color, Style, Width	Note
Pointer	Fully supported (r/w)	Note
SlideLeft	Supported (r/w)	Note

SlideUp	Supported (r/w)	Note
Tag	Fully supported (r/w)	Note
Type	Supported (r)	Note
Visible	Fully supported (r/w)	Note
X	Fully supported (r/w)	Note
Y	Fully supported (r/w)	Note

Unsupported

Moveable	Resizable
----------	-----------

13.7.3.8 Picture control

Supported

If a property is marked as "supported (r)" or "(r)", it means that the property can be read - but not changed - in script, and properties set in the painter will not take effect in the converted application.

If a property is marked as "supported(r/w)" or "(r/w)", it means that the property can be read or changed in script, and properties set in the painter will take effect.

Property	What's Supported	What's Unsupported
Attributes	Supported (r)	None
Band	Supported (r)	None
Border	Fully supported (r/w)	None
Filename	Fully supported (r/w)	None
Height	Fully supported (r/w)	None
HideSnaked	Supported (r)	None
HTML. <i>property</i>	(r): AppendedHTML, Link, LinkArgs, LinkTarget	None

Invert	Supported (r)	None
Name	Supported (r)	None
Pointer	Supported (r/w):	None
SlideLeft	Supported (r/w)	None
SlideUp	Supported (r/w)	None
Tag	Fully supported (r/w)	None
Type	Supported (r)	None
Visible	Fully supported (r/w)	None
Width	Fullysupported (r/w)	None
X	Fullysupported (r/w)	None
Y	Fullysupported (r/w)	None

Unsupported

Moveable	Resizable
----------	-----------

13.7.3.9 Rectangle control

Supported

If a property is marked as "supported (r)" or "(r)", it means that the property can be read - but not changed - in script, and properties set in the painter will not take effect in the converted application.

If a property is marked as "supported(r/w)" or "(r/w)", it means that the property can be read or changed in script, and properties set in the painter will take effect.

Property	What's Supported	What's Unsupported
Attributes	Supported (r)	None
Background. <i>property</i>	Fully supported (r/w): Color, Mode	None
Band	Supported (r)	None

Brush. <i>property</i>	(r/w): Color, Hatch	None
Height	Fully supported (r/w)	None
HideSnaked	Supported (r)	None
Name	Supported (r)	None
Pen. <i>property</i>	(r/w): Color, Style, Width	None
Pointer	Fully supported (r/w)	None
SlideLeft	Supported (r/w)	None
SlideUp	Supported (r/w)	None
Tag	Fully supported (r/w)	None
Type	Supported (r)	None
Visible	Fully supported (r/w)	None
X	Fully supported (r/w)	None
Y	Fully supported (r/w)	None

Unsupported

Moveable	Resizable
----------	-----------

13.7.3.10 Report control

Supported

If a property is marked as "supported (r)" or "(r)", it means that the property can be read - but not changed - in script, and properties set in the painter will not take effect in the converted application.

If a property is marked as "supported(r/w)" or "(r/w)", it means that the property can be read or changed in script, and properties set in the painter will take effect.

Property	What's Supported	What's Unsupported
----------	------------------	--------------------

Attributes	Fully supported	None
Band	Fully supported	None
Border	Fully supported	None
DataObject	Supported (r/w)	DataWindows with the following presentation styles cannot be the DataObject: RichText, OLE and TreeView.
Height	Fully supported	None
Name	Fully supported	None
Nested Arguments	Fully supported	None
NewPage	Supported	None
Pointer	Fully supported	None
SlideLeft	Supported	None
SlideUp	Supported	None
Tag	Fully supported	None
Trail footer	Supported	None
Type	Fully supported	None
Visible	Fully supported	None
X	Fully supported	None
Y	Fully supported	None

Unsupported

Criteria	HideSnaked	Moveable	Resizable
----------	------------	----------	-----------

Note:

It is unsupported to assign a nested report to a DataWindow object.

Unsupported example:

```
Dwobject ldwo
```

```
String ls_1
Ldwo = Dw_1.object.dw_report[1] // dw_report is a nested
report.
Ls_1 = ldwo.object.lastname[2]
```

Supported example:

```
Ls_1 = Dw_1.object.dw_report[1].object.lastname[2]
```

13.7.3.11 RoundedRectangle control**Supported**

If a property is marked as "supported (r)" or "(r)", it means that the property can be read - but not changed - in script, and properties set in the painter will not take effect in the converted application.

If a property is marked as "supported(r/w)" or "(r/w)", it means that the property can be read or changed in script, and properties set in the painter will take effect.

Property	What's Supported	What's Unsupported
Attributes	Supported (r)	None
Background. <i>property</i>	(r/w): Color, Mode	None
Band	Supported (r)	None
Brush. <i>property</i>	(r/w): Color, Hatch	None
Height	Fully supported (r/w)	None
HideSnaked	Supported (r)	None
Name	Supported (r)	None
Pen. <i>property</i>	(r/w): Color, Style, Width	None
Pointer	Fully supported(r/w)	None
SlideLeft	Supported (r/w)	None
SlideUp	Supported (r/w)	None
Tag	Fully supported (r/w)	None
Type	Supported (r)	None

Visible	Fully supported (r/w)	None
X	Fully supported (r/w)	None
Y	Fully supported (r/w)	None

Unsupported

Moveable	Resizable
----------	-----------

13.7.3.12 Text control

Supported

If a property is marked as "supported (r)" or "(r)", it means that the property can be read - but not changed - in script, and properties set in the painter will not take effect in the converted application.

If a property is marked as "supported(r/w)" or "(r/w)", it means that the property can be read or changed in script, and properties set in the painter will take effect.

Property	What's Supported	What's Unsupported
Alignment	Fully supported (r/w): Left!, Center!, Right!, Justify!	None
Attributes	Supported (r)	None
Background. <i>property</i>	Fully supported (r/w): Color, Mode	None
Band	Supported (r): Band	Background, Foreground
Border	(r/w): NoBorder!, Box!, Lowered!, Raised!, ShadowBox!, Underline!, ResizeBorder!	None
Color	Supported (r/w) This property can be set by using DataWindow expression.	None

Font. <i>property</i>	(r/w): Charset, Escapement, Face, Family, Height, Italic, Pitch, Strikethrough, Underline, Weight Note: 1) When the value of Escapement is a negative number, the text will not be rotated. 2) The text rotated can be displayed out of the control in PowerBuilder, but not in the Web application.	Width
Height	Fully supported (r/w) This property can be set by using DataWindow expression.	None
HideSnaked	Supported (r)	None
HTML. <i>property</i>	(r): AppendedHTML, Link, LinkArgs, LinkTarget, ValueIsHTML	None
Name	Supported (r)	None
Pointer	Fully supported(r/w)	None
SlideLeft	Supported (r/w)	None
SlideUp	Supported (r/w)	None
Tag	Fully supported (r/w)	None
Text	Supported (r/w)	None
Type	Supported (r)	None
Visible	Fully supported (r/w) This property can be set by using DataWindow expression.	None
Width	Fully supported (r/w) This property can be set by using DataWindow expression.	None
X	Fully supported (r/w) This property can be set by using DataWindow expression.	None

Y	Fully supported (r/w) This property can be set by using DataWindow expression.	None
---	---	------

Unsupported

Height.AutoSize	Moveable	Resizable
-----------------	----------	-----------

13.7.3.13 Unsupported controls

The properties of all the following unsupported controls are unsupported: OLE and TableBlob.

13.8 DataWindow data and property expressions

Supported

- The following DataWindow data expressions are supported:
 - Syntax for one or all data items in a named column
dwcontrol.Object.dwcolumnname{.buffer} {.datasource}{[rownum]}
 - Syntax for selected data in a named column
dwcontrol.Object.dwcolumnname{.Primary}{.datasource}.Selected
 - Syntax for a range of data in a named column
dwcontrol.Object.columnname{.buffer}{.datasource}[startrow,endrow]
 - Syntax for a single data item in a DataWindow
dwcontrol.Object.Data { .buffer } { .datasource } [rownum, colnum]
 - Syntax for data in a block of rows and columns
dwcontrol.Object.Data{.buffer}{.datasource}[startrow,startcol, endrow, endcol]
 - Syntax for data in a single row or all rows
dwcontrol.Object.Data{.buffer}{.datasource}{[rownum]}
 - Syntax for all data from selected rows
dwcontrol.Object.Data{.Primary}{.datasource}.Selected
- The expression for accessing the text displayed in the column header is supported:
dwcontrol.Object.ColumnName.Text
- Basic syntax for DataWindow property expressions is supported:
dwcontrol.Object.dwcontrolname{.property}.property{=value}

- Setting or getting the value of a dot notation is supported, except that there is structure data in the expression

Unsupported

- Getting the data value from a filter buffer according to the row number is unsupported. This is because data is stored in a different order in Apeon than in PowerBuilder. For example, the following script is unsupported:

```
dw_control.object.filter[startrow, startcol, endrow,
endcol] //unsupported
```

- Setting the value of a dot notation to a structure array, or setting a structure array to a dot notation, is unsupported. For example, the following syntax is unsupported:

```
struct1 t[10]
t = dw_1.object.data[1]
```

Note: The Apeon Unsupported Features Analysis tool cannot detect unsupported features in DataWindow dot notations. You must make sure the specifications about dot notations are satisfied in the application. Otherwise, it will cause errors. For more information on undetected features, refer to the [Undetected features](#) section.

- Selected and Data are unsupported to be directly accessed by a DataWindow object via the dot notation. For example:

```
ldwo_target.Primary = ldwo_column.selected //Unsupported
ldwo_target.Primary = dw_1.object.id.selected //Supported
```

- It is unsupported to contain Data in a dot notation to obtain the data of a column. For example:

Supported example:

```
string ls_an
dwobject dwo
dwo = dw_1.object.id
ls_an = string(dwo.data.primary[1])
```

Unsupported example:

```
string ls_an
dwobject dwo
dwo = dw_1.object.id
ls_an = string(dwo.primary[1])
```

It is unsupported to use the following syntax in obtaining data in a row of a report.

```
dw_1.Object.dw_report[1] //Unsupported
```

13.9 DataWindow constants

Supported

In PowerBuilder, constants are defined in the DataWindow control for values of properties and arguments for methods. They are sets of values associated with enumerated data

types. Values for enumerated data types always end with an exclamation point. The following list is the PowerBuilder constants Apeon supports:

Constant	Supported Values	Unsupported Values
Alignment	Left!	None
	Center!	
	Right!	
	Justify!	
Band	Detail!	None
	Header!	
	Footer!	
Border	NoBorder!	ShadowBox!
	Box!	Underline!
	Lowered!	ResizeBorder!
	Raised!	None
BorderStyle	StyleBox!	StyleShadowBox!
	StyleLowered!	
	StyleRaised!	
CharSet	CharSetAnsi!	CharSetDBCS-Japanese!
	CharSetUnicode!	
	CharSetAnsiHebrew!	
	CharSetAnsiArabic!	
DWBuffer	Primary!	None
	Delete!	
	Filter!	
DWItemStatus	NotModified!	None

	DataModified!	None
	New!	Note: It is not recommended to change status in script. Frequent or inappropriate user modification may result in unpredictable problems. The following operation is not recommended for new rows with no specified values for their columns:
	NewModified!	Changing a row's status from New! to NewModified and updating in script.
SaveAsType	Excel!	CSV!, SYLK!, WKS!, WK1!, DIF!, dBASE2!, dBASE3!, SQLInsert!, Clipboard!, PDF!, PSReport!, XSLFO!, Excel8!, EMF!
	Excel5!	
	HTMLTable!	
	Text!	
	WMF!	
	XML!	
SQLPreviewFunction	PreviewFunctionRetrieve!	PreviewFunctionReselectRow!
		PreviewFunctionUpdate!
SQLPreviewType	PreviewSelect!	PreviewInsert!
		PreviewDelete!
		PreviewUpdate!

If a constant is set to an unsupported value, Apeon will use the default enumerated value as the constant value. For example, unsupported BorderStyle will be read as NoBorder.

Unsupported

The following constants are not supported:

ConnectionSource	DriverType	DWConflictResolution	FillPattern
grColorType	grDataType	grObjectType	grSymbolType

LineStyle	RowFocusInd		
-----------	-------------	--	--

13.10 DataWindow control

13.10.1 Properties of DataWindow control

Supported

The supported properties of DataWindow controls are listed in the table below:

Border	BorderStyle*	BringToTop	ControlMenu
DataObject	DragAuto	DragIcon	Enabled
Height	HScrollBar*	LiveScroll	Icon
MaxBox	MinBox	Object	Resizable*
RightToLeft	TabOrder	Tag	Title
TitleBar	Visible	VScrollBar*	Width
X	Y		

- **BorderStyle**
Three types of BorderStyle are supported: Box, 3D Raised, 3D Lowered, but Shadow Box is unsupported.
- **VScrollBar or HScrollBar**
In PowerBuilder, when the property VScrollBar or HScrollBar is changed from visible to invisible or from invisible to visible, the Resize event will be triggered. However, in the same scenario on the Web, the Resize event will not be triggered.
- **Resizable**
The Resizable property cannot be dynamically changed.
After setting this property to True, setting Border properties will not be effective.

Unsupported

The unsupported properties of DataWindow controls are listed in the table below:
ClassDefinition and HSplitScroll

13.10.2 Events of DataWindow control

Events for HTML DataWindows	Supported			
	ButtonClicked	ButtonClicking	Clicked	Constructor
	Collapsed	Collapsing	DBError*	Destructor
	DoubleClicked	DragDrop	DragEnter	DragLeave
	DragWithin	EditChanged	Expanded	Expanding
	GetFocus	ItemChanged	ItemError	ItemFocusChanged
	LoseFocus	PrintPage	RButtonDown	Resize
	RetrieveEnd	RetrieveStart	RowFocusChanged	RowFocusChanging
	<i>DBError</i>			
	The <i>sqlsyntax</i> and <i>row</i> arguments do not work in the DBError event. The <i>row</i> argument is always considered to be 0.			
	The GetSQLPreview function will be ignored if used in the DBError event.			
	<i>SQLPreview</i>			
	Only the value <i>previewselect!</i> can be used as the SQLPreview <i>sqltype</i> argument.			
	Unsupported			
Error	Help	Other	PrintEnd	
PrintStart	RetrieveRow	ScrollHorizontal	ScrollVertical	
Using instance variables or global variables to store the dwo argument (argument type is dwobject) for a DataWindow event is unsupported.				
Triggering of DataWindow events	Supported			
	Nearly full support is available for Triggering supported DataWindow events, except for the following limitations and differences.			
	Limitations & differences			

(For **ItemChanged** event) In a Web application, if the ItemChanged event is triggered for a field whose new value does not pass validation, the focus will not go back to the column being edited.

(For **Resize** event) When the Width or Height of a DataWindow control is reset, the Resize event will be triggered, regardless of whether the value is changed or remains the same.

(For **RowFocusChanged** event) In PowerBuilder, RowFocusChanged will be triggered twice upon execution of the RowsCopy or RowsMove function. However, in the same scenario on the Web, the RowsFocusChanged event will only be triggered once.

(For **SQLPreview** event) The SQLPreview event can only be triggered by the Retrieve function, not by Update or ReselectRow.

If a DataWindow event requires interaction with the server, its subsequent event may not be triggered due to the time delay caused by the interaction with the server.

For label DataWindows, clicking different lines cannot trigger the RowFocusChanging and RowFocusChanged events.

13.10.3 Functions of DataWindow control

There are some limitations or differences for calling functions marked with the "*" symbol. Refer to the Limitations and Differences for Calling Functions section.

Functions for DataWindows	Supported				
	AcceptText	CategoryCount	CategoryName	ClassName	Clear
ClearValues	Clipboard	Copy	CopyRTF	Collapse	
CollapseAll	CollapseAllChildren	CollapseLevel	Cut	DataCount	
DeletedCount	DeleteRow	Describe *	Drag	Expand	
ExpandAll	ExpandAllChildren	ExpandLevel	Filter	FilteredCount	
Find	FindCategory	FindGroupChange	FindNext	FindSeries	
GetBandAtPointer	GetBorderStyle	GetChanges	GetChild	GetClickedColumn	
GetClickedRow*	GetColumn	GetColumnName	GetData	GetDataPieExplode	
GetDataStyle	GetDataValue	GetFormat	GetFullState	GetItemDate	
GetItemDataTime	GetItemDecimal	GetItemFormattedString	GetItemUnformattedString	GetItemNumber	

GetItemStatus	GetItemString	GetItemTime	GetNextModified	GetObjectAtPointer
GetParent	GetRow	GetRowFromRowId	GetSelectedRow	GetSeriesStyle
GetSQLPreview	GetSQLSelect	GetText	GetValidate	GetValue*
GroupCalc	Hide	ImportClipboard	ImportFile	InsertDocument
InsertRow	ImportString	IsExpanded	IsSelected	LineCount
ModifiedCount	Modify*	ObjectAtPointer	Paste	PasteRTF
Position	PointerX	PointerY	PostEvent	Print*
PrintCancel	Reset*	ResetDataColors	ResetUpdate	Resize
Retrieve*	RowCount	RowsCopy*	RowsDiscard*	RowsMove
SaveAs*	SaveAsAscii	ScrollNextPage	ScrollNextRow	ScrollPriorPage
ScrollPriorRow	ScrollToRow	SelectedLength	SelectedStart	SelectedText
SelectedLine	SelectText	SelectTextLine	SelectTextAll*	SelectTextWors
SetChanges	SeriesCount	SeriesName	SetBorderStyle	SetColumn
SetDataPieExplode	SetDataStyle	SetDetailHeight*	SetFilter	SetFocus
SetFormat	SetFullState	SetItem	SetItemStatus	SetPosition*
SetRedraw	SetRow	SetRowFocusIndicator	SetSeriesStyle	SetSort*
SetSQLPreview	SetSQLSelect*	SetTabOrder	SetText	SetTransObject
SetValidate	SetValue*	ShareData	ShowHeadFoot	ShareDataOff
Show	Sort*	TextLine	TriggerEvent*	Typeof
Undo	Update			
There are some limitations or differences for calling functions marked with the "*" symbol. Refer to the Limitations and Differences for Calling Functions section.				
Unsupported				
CanUndo	CrosstabDialog	DBCcancel	DBErrorCode	DBErrorMessage
GenerateHTMLForm	GenerateResultSet	GetContextService	GetMessageText	GetStateStatus
GetTrans	GetUpdateStatus	OLEActivate	ResetInk	ReselectRow
SaveInk	SaveInkPic	Scroll	SelectRow	ResetTransObject
SetActionCode	SetHTMLAction	SetTrans		

Limitations and Differences for Calling Functions

Limitations

- GetChild

In earlier versions of PowerBuilder, the `GetChild` function can be used in a DataWindow for referring to a child DataWindow of the DataWindow. Such usage is unsupported in PowerBuilder 9 and above, and is also not supported in Apeon.

- **GetValue**

The *column* argument only works when the edit style of the column is `CheckBox`, `DropDownListBox`, `Edit`, or `RadioButton`.

- **ImportFile, ImportString, ImportClipboard**

1. The *filename* argument of these functions can only be a text file (TXT).
2. The *importtype* argument of these functions is unsupported.
3. When using these functions, you should ensure that the data type imported matches the data type that you specified in a DataWindow control, DataStore object or graph control. Otherwise the output on the Web may be different from that in PowerBuilder. For example, if the date type specified in a DataWindow is string, when you import a file with date number (eg, 2006-08-01), on the Web the date data will be recognized as a string (2006), however in PowerBuilder, it will be recognized as a date data (2006-08-01).

- **Modify**

Refer to [Dynamic DataWindow](#).

- **Print**

- Arguments specified in the Print function will be ignored in a Web application. Even if the *CancelDialog* argument is set to `TRUE` or `FALSE`, no Cancel dialog will be displayed on the Web to cancel printing.
- When the Print function is executed for a DataWindow that contains a Text control with `BorderType` as `None` and actual length of the content larger than the length of the control, the printing result shows the content in the Text control the same way as the control is displayed in the application, although in PowerBuilder, the printing result shows the full content in the Text control.
- For controls, print function is only supported when called in Graph controls with the following syntax.
integer objectname.Print (long printjobnumber, integer x, integer y {, integer width, integer height })

- **SaveAs**

- On the Web, `Nillable` attribute is unsupported in an XML schema.
- The supported SaveAs syntax:
ll_testvalue = dw_test.SaveAs(filename,saveastype, colheading)

In the syntax:

The *colheading* argument does not take any effect.

The value of a *saveastype* argument can be `TEXT`, `HTMLTable`, `WMF` (Windows only), `EXCEL`, `EXCEL5`, `XML` or `PDF`.

The unsupported SaveAs syntax:

dw_test.SaveAs([filename,]graphcontrol[, saveastype, colheading])

- **Reset**

The unsupported syntax: `Reset(dwcontrol)`.

The supported syntax: `dwcontrol.Reset`.

- **SetPosition**

The supported syntax:

`dwcontrol.SetPosition (position {, precedingobject })`

the *position* can be `ToTop!`, `ToBottom!`, but cannot be `Behind!`

The unsupported syntax

`dwcontrol.SetPosition (objectname, band, bringtofront)`

- **SetSQLSelect**

1. If a DataWindow uses a stored procedure as its data source, the `SetSQLSelect` function cannot be executed for the DataWindow.
2. You should create transaction instance for DataWindow control and DataStore object before executing this function. Otherwise `SetSQLSelect` cannot be verified in the Web application and it causes the execution to fail.

- **SetValue**

The `SetValue` *column* argument only works if the edit style of the column is `DropDownListBox`.

- **SetSort**

1. If the `Format` argument of the `SetSort` contains sorting criteria for more than one column, the criteria of the columns must be separated with commas.
2. Using “`desc`” or “`asc`” as the `Order` will not have any effect. Please use the standard `Order` value `A` for ascending or `D` for descending order.

- **TriggerEvent**

The following `TriggerEvent` syntax is unsupported:
`object1.TriggerEvent (object2, event)`.

2. Differences

- **Describe**

If there is no value for a property in the `propertylist` argument, `Describe` returns a question mark (?) in PowerBuilder, but returns the default value (for example, `Arrow!`) of the property on the Web.

- **GetClickedRow**

In a Web application, if the user clicks or double-clicks any area within a DataWindow, the `GetClickedRow` function returns the same value as in PowerBuilder. If the `GetClickedRow` is called in some control or window for a DataWindow, the return value is different on the Web than in PowerBuilder. For example, if the `dw_1.GetClickedRow()` function is called in the `Clicked` event of a `CommandButton`, it returns -1 in PowerBuilder and 0 on the Web.

- **GetItemDecimal**

The return value of the *GetItemDecimal* function will omit the zero(s) to the right of the decimal value on the Web. For example: 3000.100 will be returned as 3000.1 on the Web but 3000.100 in PowerBuilder.

- Retrieve
 - Because Apeon Web applications cannot check whether the retrieve result set matches the DataWindow definition syntax, the DBError event will not be triggered.
 - If the data type of an EditMask control is numeric, the value of the control cannot be set to "0".
- SetDetailHight

Setting detail.height.autosize to true after specifying the height by this function, executing Describe returns the auto sized height on the Web. But in PowerBuilder it returns the height specified by this function.
- RowDiscard

For Composite, Group, Label and RichText DataWindows, if the current row is the start row, calling RowDiscard will not trigger RowFocusChanged on the Web. This is different from that in PowerBuilder.
- SaveAsAscii

When executing the SaveAsAscii function in PowerBuilder, invisible controls which lay in rows will not be saved as empty rows or " " (quotation marks). However, these invisible controls will be ignored on the Web.
- SelectTextAll

If the focus is in the header or footer band, executing "SelectTextAll (detail!)" on the Web returns -1 and no text will be selected. However, in PowerBuilder, the text in the band, which the focus is in, will be selected.
- ShareData and ShareDataOff
 1. Supported:

```
CONNECT USING SQLCA;
dw_corp.SetTransObject(SQLCA)
dw_corp.Retrieve()
dw_corp.ShareData(dw_emp)
dw_corp.ShareData(dw_dept)
... // Some processing
dw_emp.ShareDataOff()
```
 2. In Web applications, if any data in two data-sharing Group DataWindows are changed, the primary Group DataWindow will be re-grouped and re-sorted automatically, while the secondary Group DataWindow will only be re-grouped.
 3. The row and column status is shared between the primary DataWindow and secondary DataWindow on the Web, although not shared in PowerBuilder.
- Sort

1. Different from the Sort in PowerBuilder, the Web Sort function considers the "~" symbol to have precedence over characters. For this reason, the sort result may be a little different in Web and PowerBuilder applications.
2. In the Web application, when the user sorts a DataWindow on a specified column, rows containing special characters (e.g. ".", "", "/") will be sorted in different order than in PowerBuilder.

13.11 DataStore object

A DataStore is a nonvisual DataWindow control. DataStores act just like DataWindow controls except that many of the visual properties associated with DataWindow controls do not apply to DataStores. Because you can print DataStores, PowerBuilder provides some events and functions for DataStores that pertain to the visual presentation of the data.

However, graph functions such as CategoryCount, CategoryName, GetData, SeriesCount, and so forth depend on the visual graph control, which is not created for a DataStore object. These functions return an error value or an empty string when used with DataStores.

13.11.1 Properties of DataStore object

Supported

The supported properties of DataStore are listed in the table below:

DataObject	Object
------------	--------

Unsupported

The unsupported property of DataStore is ClassDefinition.

13.11.2 Events of DataStore object

Supported

The following DataStore events are supported:

Constructor	DBError	Destructor*	ItemChanged
ItemError	PrintPage	RetrieveEnd	RetrieveStart
SQLPreview*	UpdateEnd	UpdateStart	

- **Destructor**
If a non-visual object is a local variable, the Destructor event in the non-visual object cannot be triggered unless there is a Destroy statement for the non-visual object as well.

- SQLPreview
The SQLPreview event can be triggered by Retrieve method, but cannot be triggered by the Update or ReselectRow methods.

Unsupported

The following DataStore events are unsupported:

Error	PrintEnd	PrintStart	RetrieveRow
-------	----------	------------	-------------

13.11.3 Functions of DataStore object

Functions for Composite, Crosstab, Freeform, Graph, Grid, Group, Label, N-Up, Nested, RichText and Tabular DataStores	Supported				
	AcceptText	ClassName	ClearValues	Create	DeletedCount
	DeleteRow	FindGroupChange	Describe	Filter	FilteredCount
	Find	FindRequired	GenerateHTMLForm	GetChanges	GetChild
	GetColumn	GetColumnName	GetFormat	GetFullState	GetItemDate
	GetItemDateTime	GetItemDecimal	GetItemNumber	GetItemStatus	GetItemString
	GetItemTime	GetNextModified	GetParent	GetRow	GetRowFromRowId
	GetRowIdFromRow	GetSelectedRow	GetSQLSelect	GetText	GetValidate
	GetValue*	GroupCalc	ImportClipboard*	ImportFile*	ImportString*
	InsertRow	IsSelected	ModifiedCount	Modify*	PostEvent
	Print	Reset	ResetUpdate	Retrieve	RowCount
	RowsCopy	RowsDiscard	RowsMove*	SaveAs*	SaveAsAscii
	SetDetailHeight	SelectRow	SetChanges	SetColumn	SetFilter
	SetFormat	SetFullState	SetItem	SetItemStatus	SetPosition
	SetRow	SetSort*	SetSQLPreview	SetSQLSelect*	SetText
	SetTransObject	SetValidate	SetValue*	ShareData	ShareDataOff
	Sort	TriggerEvent*	TypeOf	Update	
	Unsupported				

	CategoryCount	CategoryName	Clipboard	CopyRTF	CreateFrom
	DataCount	DBCcancel	FindCategory	FindSeries	GenerateResultSet
	GetBorderStyle	GetContextService	GetClickedColumn	GetClickedRow	GetData
	GetDataPieExplore	GetDataStyle	GetDateValue	GetSeriesStyle	GetStateStatus
	GetTrans	InsertDocument	PasteRTF	PrintCancel	ReselectRow
	ResetDataColors	ResetTransObject	SeriesCount	SeriesName	SetBorderStyle
	SetDataPieExplore	SetDataStyle	SetHTMLAction	SetSeriesStyle	SetTrans

Limitations and Differences for Calling Functions

- GetValue

The *column* argument works only if the edit style of the column is CheckBox, DropDownListBox, Edit, or RadioButton.
- ImportFile, ImportString, ImportClipboard
 1. The *filename* argument of these functions can only be a text file (TXT).
 2. The *importtype* argument of these functions is unsupported.
 3. When using these functions, you should ensure that the data type imported matches the data type that you specified in a DataWindow control, DataStore object or graph control. Otherwise the output on the Web may be different from that in PowerBuilder. For example, if the date type specified in a DataWindow is string, when you import a file with date number (eg, 2006-08-01), on the Web the date data will be recognized as a string (2006), however in PowerBuilder, it will be recognized as a date data (2006-08-01).
- Modify

If the Modify function is used to modify the WHERE clause of the DataWindow object's SQL SELECT statement, please make sure that the new SQL SELECT statement is correct in syntax. Otherwise, syntax errors can occur on the Web because PowerBuilder does not validate the statement whereas JavaScript does.
- RowsMove

Events cannot be triggered while the application moves data within the primary buffer of one DataWindow.
- SaveAs
 1. The *saveastype* argument can be TEXT, HTMLTable, WMF (EAServer only), EXCEL, EXCEL5 or XML.
 2. The supported SaveAs syntax:


```
ll_testvalue = dw_test.SaveAs(filename,saveastype,
```

```
colheading)
```

The unsupported SaveAs syntax:

```
dw_test.SaveAs([filename, ]graphcontrol[, saveastype,
colheading])
```

3. On the Web, Nillable attribute is unsupported in an XML schema.

- SaveAsAscii

Retainnewlinechar argument is unsupported.

- SetValue

The *column* argument works only if the edit style of the column is DropDownListBox.

- SetSort

Using “desc” or “asc” as the Order will not have any effect. Please use the standard Order value A for ascending or D for descending order.

- TriggerEvent

The following TriggerEvent syntax is unsupported:

```
object1.TriggerEvent (object2, event)
```

SetSQLSelect

You should create transaction instance for DataWindow control and DataStore object before executing this function. Otherwise SetSQLSelect cannot be verified in the Web application and it causes the execution to fail.

13.12 DataWindowChild object

13.12.1 Properties of DataWindowChild object

Supported

None.

Unsupported

The unsupported property of DataWindowChild is ClassDefinition.

13.12.2 Functions for DataWindowChild

Functions for DataWindowChild	Supported				
	AcceptText	ClassName	DeletedCount	DeleteRow	Describe
	Filter	FilteredCount	Find	FindGroupChange	FindRequired
	GetBandAtPointer	GetBorderStyle	GetChanges	GetChild	GetClickedColumn
	GetClickedRoww	GetColumn	GetColumnName	GetFormat	GetItemDate

	GetItemDateT ime	GetItemDecimal	GetItemNumber	GetItemStatus	GetItemString
	GetItemTime	GetNextModified	GetObjectAtPointer	GetParent	GetRow
	GetSelectedR ow	GetSQLPreview	GetSQLSelect	GetText	GetValidate
	GetValue	GroupCalc	ImportClipboard*	ImportFile*	ImportString*
	InsertRow	IsSelected	ModifiedCount	Modify*	Reset
	ResetUpdate	Retrieve	RowCount	RowsCopy	RowsDiscard
	RowsMove	SaveAs*	ScrollNextPage	ScrollNextRow	ScrollPriorPa ge
	ScrollPriorRo w	ScrollToRow	SetChanges	SetColumn	SetFilter
	SetDetailHeig ht	SetFormat	SetItem	SetItemStatus	SetPosition*
	SetRedraw	SetRow	SetRowFocusIndica tor	SetSort*	SetSQLPreview
	SetSQLSelect	SetTabOrder	SetText	SetTransObject	SetValue
	SetValidate	ShareData	SharedDataOff	Sort	TypeOf
	Update				
Unsupported					
	CrosstabDialog	DBCcancel	DBErrorCo de	DBErrorMessa ge	GetContextService
	GetRowIDFromRow	GetRowFrom RowID	GetTrans	GetUpdateStatu s	OLEActivate
	ReselectRow	ResetTransO bject	SelectRow	SetBorderStyle	SetTrans

Limitations and Differences for Calling Functions

- GetValue

The column argument does not work only if the edit style of the column is CheckBox, DropDownListBox, Edit, or RadioButton.

- ImportFile, ImportString, ImportClipboard

1. The *filename* argument of these functions can only be a text file (TXT).
2. The *importtype* argument of these functions is unsupported.
3. When using these functions, you should ensure that the data type imported matches the data type that you specified in a DataWindow control, DataStore object or graph control. Otherwise the output on the Web may be different from that in PowerBuilder. For example, if the date type specified in a DataWindow is string, when you import a file with date number (eg, 2006-08-01), on the Web the

date data will be recognized as a string (2006), however in PowerBuilder, it will be recognized as a date data (2006-08-01).

- Modify

If the Modify function is used to modify the WHERE clause of the DataWindow object's SQL SELECT statement, please make sure that the new SQL SELECT statement is correct in syntax. Otherwise, syntax errors can occur because PowerBuilder does not validate the statement whereas JavaScript does.

- SaveAs

1. The *saveastype* argument can be TEXT, HTMLTable, WMF (EAServer only), EXCEL, EXCEL5 or XML
2. On the Web, Nillable attribute is unsupported in an XML schema.
3. The supported SaveAs syntax:

```
ll_testvalue = dw_test.SaveAs(filename,saveastype,  
colheading)
```

The unsupported SaveAs syntax:

```
dw_test.SaveAs([filename, ]graphcontrol[, saveastype,  
colheading])
```

- SetSQLSelect

You should create transaction instance for DataWindow control and DataStore object before executing this function. Otherwise SetSQLSelect cannot be verified in the Web application and it causes the execution to fail.

- SetSort

Using “desc” or “asc” as the Order will not have any effect. Please use the standard Order value A for ascending or D for descending order

- SetPosition

The following syntax is supported:

```
dwcontrol.SetPosition (position {, precedingobject } )  
the position can be ToTop!, ToBottom!, and cannot be Behind!
```

The following syntax is unsupported:

```
dwcontrol.SetPosition (objectname, band, bringtofront )
```

- Sort

If the items in a Web DropDownDataWindow are double-byte (such as, Chinese, Korean, or Japanese), sorting the DropDownDataWindow has a different result from PowerBuilder. This is because the sorting method of PowerScript and JavaScript is different. For example, if the charset is Chinese, PowerScript sorts by the spelling while JavaScript sorts by Unicode.

13.13 DataWindow performance considerations

Strong suggestion: reduce usage of DataWindow SQLPreview event

Each time the DataWindow SQLPreview event is triggered, the Web application will interact with Appeon Server twice, which costs 1-2 seconds. Therefore, Appeon recommends you minimize writing script into the SQLPreview event of the DataWindow.

Suggestion: use Describe and Modify to get and set DataWindow object properties

Dot notation is much less efficient than Describe and Modify functions. In general, the Describe and Modify functions are about two to three times faster than dot notation. Therefore, Appeon recommends you use Describe function to replace dot notation that gets the DataWindow object properties, and use the Modify function to replace dot notation that sets the DataWindow object properties.

Suggestion: minimize modifying DataWindow DataObject

It takes 0.3 seconds each time the DataObject property of DataWindow/DataStore is modified during application run time. Therefore, modifying the DataObject property frequently could slow down performance significantly. Appeon recommends you minimize modifying the DataObject property.

14 DBParm parameters in Database

Supported

The following DBParm parameters are supported: CommitOnDisconnect, CacheName, DelimitIdentifier, DateTimeAllowed, DisableBind and TrimSpaces.

- Disablebind parameter is unsupported in MS SQL Server.
- In PowerBuilder, the connection cache name only applies when a PowerBuilder NVO is deployed to EAServer. In Apeon, the connection cache specified applies for the connection of the deployed application to the database. Settings to the other parameters in the script will be ignored.
- To support DelimitIdentifier parameter, you must set the "Enclose table and column names in double quotes" option in the Apeon Developer | DB Type Profile Configuration page before deployment.
- If table and column names are keywords that must be enclosed in double quotation marks (with DelimitIdentifier set to "yes"), the SQL statement cannot be a PowerBuilder SELECT statement, otherwise, PDF Print and Save As PDF will fail.
- The DelimitIdentifier parameter is unsupported when the updated table name of DataWindow contains the owner name. In this case, be sure to clear the "Enclose table and column names in double quotes" check box in Apeon Developer.
- It is unsupported to dynamically set the DelimitIdentifier parameter. For all supported DBParm parameters (excluding DelimitIdentifier) you can only dynamically set them before executing Connect.

Unsupported

Except the parameters above, other DBParm parameters are unsupported.

15 Web Services Client

Apeon supports using the SoapConnection object and SoapException object to implement the Web Service. UDDI Proxy is currently unsupported.

Requirements

In order to run the Web application contains Web Service, it is required to:

1. Configure Web Service in Apeon Developer. Refer to the section Application profile settings in working with Apeon Developer,
2. Install Soap ToolKit 3.0 in every client machine. You can download this tool from www.microsoft.com.

15.1 SoapConnection object

SoapConnection object contains the following methods. GenerateProxy is currently unsupported.

Method	Support Level	Example Code
CreateInstance	Supported	<pre>syb_myport myproxy long ret ret = Conn.CreateInstance(myproxy, "syb_myport", "http://my.server/soap/myport")</pre>
SetOptions	Supported	<pre>lOpt=Conn.SetOptions("SoapLog=~"airportweather.log~ "")</pre>
SetProxyServer Options	Supported	<pre>long ll_return string ls_string ls_string = "address='http://Srvr:8080/endpoint'," ls_string += "userID='MyName', password='mypass'" ll_return = Conn.SetProxyServerOptions (ls_string)</pre>
GenerateProxy	Unsupported	None

15.2 SoapException object

Apeon supports the following properties and functions of SoapConnection object.

Exception property	Supported
	Text

	Contains the text of the error message.
Exception function	Supported
	GetMessage
	Returns the error message from objects of type RuntimeException.
	SetMessage
	Sets an error message for an object of type RuntimeException.

16 Undetected features

16.1 Overview

Appeon provides an Unsupported Features Analysis tool for detecting unsupported features in a PowerBuilder application before generating Web files for the application. The Unsupported Features Analysis scans each object within a PowerBuilder application for unsupported features and generates a report that, when combined with the *Appeon Migration User Guide* and *Appeon Help*, provides a foundation for bringing a PowerBuilder application to specification.

The Appeon Unsupported Features Analysis is capable of highlighting the majority of unsupported features contained in the PowerBuilder application; however, there are some features that the Unsupported Features Analysis will not detect, and thus will not list in the Analysis Report. Approximately 5% of PowerBuilder features that are not supported by Appeon may not be detected in the Unsupported Features Analysis.

An application containing unsupported features can still be converted to the Web and work correctly, depending on the type and number of unsupported features. If the Web application does not work correctly but the Unsupported Features Analysis does not indicate any unsupported features that are likely to cause such a problem, the problem may be caused by an unsupported feature that has not been detected. In this case, it is strongly recommended that you carefully read the undetected features listed below and examine your application to determine if it contains any of these features.

16.2 Undetected Features Listing

The following are unsupported features that are not detected during Unsupported Features Analysis.

Naming conventions	<ul style="list-style-type: none"> • Duplicate object names. No two objects should have the same name in an application, whether they are of the same type or not. The Unsupported Features Analysis cannot detect whether object names are duplicated in an application. • More than one Application object. Having more than one Application object in an application is unsupported. Unsupported Features Analysis cannot detect whether more than one Application object is present. • Applications named as objects or controls. Application names that have the same name as PowerBuilder control/object types are not supported. Unsupported Features Analysis cannot detect whether this unsupported issue is present in an application.
Null values	<ul style="list-style-type: none"> • Unsupported operation • Expressions with Null value

ASCII characters	<p>Unsupported special ASCII characters:</p> <ul style="list-style-type: none"> • Vertical tab (~v)
Non-visual UserObject	A NonVisualObject object assigned to an autoinstantiated NVO or an autoinstantiated NVO assigned to a NonVisualObject object.
Variables and constants	Instance variables have identical names as global variables.
Forced conversion	Forced conversion between types
Window	Multiple MDI windows in an application.
Overloading, overriding functions	<ul style="list-style-type: none"> • Dynamic calling for overloaded functions • Dynamic calling for overriding functions
Using the return value of some supported functions	The return value of the Open or OpenSheet functions
User functions	Remote procedure call (RPC)
Operators	The operator '^' with embedded SQL statements.
Stored procedure	<ul style="list-style-type: none"> • DB2 stored procedures • Stored procedures placed inside an Oracle package • Oracle stored procedures with Apeon unsupported features
Cursor declare requirement	If a cursor is declared for retrieving rows from table X, the table X (insert, delete, update) is modified during the cursor declare-close period.
Cursor statements	<p>The following two syntaxes:</p> <ol style="list-style-type: none"> 1. UPDATE TableName SetStatement WHERE CURRENT OF CursorName; 2. DELETE FROM TableName WHERE CURRENT OF CursorName;
DataWindow data expressions	<ul style="list-style-type: none"> • Unsupported features in the DataWindow dot notations. You must make sure the specifications about dot notations are satisfied in the application. Otherwise, it will cause errors. • The return values of the DataWindow dot notations are assigned to user objects.
Partly supported	Some partly supported features (such as DataWindow Modify and Describe functions).

features	
User interface interactions	Unsupported features in the UI such as shortcut key.
Enumerated data type	Default values of enumerate type variable
Data source	The data sources of dynamically created DataWindows are store procedures with input parameters.
Dynamic calls	<ul style="list-style-type: none"> • Dynamically call the method of a menu object. • Dynamically call the method that contains reference arguments.
System events	There are some system events cannot be supported. For details, refer to the table below.

Unsupported and undetected system events:

cbgetcount	custom38	emgetwordbreakproc	mbuttondown	querynewpalette	vbxevent01
cbgetcursel	custom39	emlimittext	mbuttonup	queryopen	vbxevent02
cbgetdroppedcontrolrect	custom40	emlinefromchar	mdiactivate	queuesync	vbxevent03
cbgetdroppedstate	custom41	emlineindex	mdicascade	quit	vbxevent04
cbgeteditssel	custom42	emlinelength	mdiconarrange	rcresult	vbxevent05
cbgetextendedui	custom43	emlinescroll	mdicreate	renderallformats	vbxevent06
cbgetitemdata	custom44	emreplacesel	mdidestroy	renderformat	vbxevent07
cbgetitemheight	custom45	emscroll	mdigetactive	renhscroll	vbxevent08
cbgetlbtextlen	custom46	emsetfont	mdimaximize	renvscroll	vbxevent09
cbinsertstring	custom47	emsethandle	mdinext	sbnbottom	vbxevent10
cblimittext	custom48	emsetmodify	mdirestore	sbn dragdrop	vbxevent11
cbncloseup	custom49	emsetpasswordchar	mdisetmenu	sbn dragenter	vbxevent12
cbndblclk	custom50	emsetreadonly	mditile	sbn dragleave	vbxevent13

cbndelayedrecogfail	custom51	emsetrect	measureitem	sbnndragwithin	vbxevent14
cbndragdrop	custom52	emsetrectnp	menuchar	sbnendscroll	vbxevent15
cbndragenter	custom53	emsetsel	menuselect	sbnkillfocus	vbxevent16
cbndragleave	custom54	emsettabstops	mmjoy1buttondown	sbnlinedown	vbxevent17
cbndragwithin	custom55	emsetwordbreak	mmjoy1buttonup	sbnlineup	vbxevent18
cbndropdown	custom56	emsetwordbreakproc	mmjoy1move	sbnpagedown	vbxevent19
cbneditchange	custom57	emundo	mmjoy1zmove	sbnpageup	vbxevent20
cbneditupdate	custom58	enable	mmmcinotify	sbnsetfocus	vbxevent21
cbnendrec	custom59	enchange	mmmimclose	sbnthumbposition	vbxevent22
cbnerrspace	custom60	endsession	mmmimdata	sbnthumbtrack	vbxevent23
cbnkillfocus	custom61	enerrspace	mmmimerror	sbntop	vbxevent24
cbnmodified	custom62	enhscroll	mmmimlongdata	setcursor	vbxevent25
cbnrresult	custom63	enmaxtext	mmmimlongerror	setfont	vbxevent26
cbnselchange	custom64	enteridle	mmmimopen	settext	vbxevent27
cbnselendcancel	custom65	enupdate	mmmomclose	sizeclipboard	vbxevent28
cbnselendok	custom66	envscroll	mmmomdone	skb	vbxevent29
cbnsetfocus	custom67	erasebkgn	mmmomopen	spoolerstatus	vbxevent30
cbresetcontent	custom68	fontchange	mmwimclose	syschar	vbxevent31
cbselectstring	custom69	getdlgcode	mmwimdata	syscolorchange	vbxevent32
cbsetcursel	custom70	getfont	mmwimopen	syscommand	vbxevent33
cbseteditself	custom71	getminmaxinfo	mmwomclose	sysdeadchar	vbxevent34
cbsetextendedui	custom72	gettext	mmwomdone	syskeydown	vbxevent35
cbsetitemdata	custom73	gettextlength	mmwomopen	systemerror	vbxevent36
cbsetitemheight	custom74	globalrchange	mouseactivate	sysykeyup	vbxevent37

cbshowdropdown	custom75	heditctl	move	tbnmoved	vbxevent38
changechain	ddeack	help	ncactivate	tcnClicked	vbxevent39
char	ddeadvise	hookrcresult	ncalcsize	tcndoubleclicked	vbxevent40
chartitem	ddedata	hscroll	nccreate	tcndragdrop	vbxevent41
childactivate	ddeexecute	hscrollclipboard	ncdestroy	tcndragenter	vbxevent42
commnotify	ddepoke	iconerasebkgn	nchittest	tcndragleave	vbxevent43
compacting	dderequest	initdialog	nclbuttondblclk	tcndragwithin	vbxevent44
compareitem	ddeterminate	initmenu	nclbuttondown	tcnenterkey	vbxevent45
ctlcolor	ddeunadvise	initmenupopup	ncmbuttondblclk	tcnkeydown	vbxevent46
custom01	deactivate	keyup	ncmbuttondown	tcnkillfocus	vbxevent47
custom02	deadchar	lbaddstring	ncmbuttonup	tcnrClicked	vbxevent48
custom03	deleteitem	lbdeletestring	ncmousemove	tcnrdoubleclicked	vbxevent49
custom04	destroyclipboard	lbidir	ncpaint	tcnSelchanged	vbxevent50
custom05	devmodechange	lbfindstring	ncrbuttondblclk	tcnSelchanging	vkeytoitem
custom06	drawclipboard	lbfindstringexact	ncrbuttondown	tcnsetfocus	vscroll
custom07	drawitem	lbgetcaretindex	ncrbuttonup	timechange	vscrollclipboard
custom08	dropfiles	lbgetcount	nextdlgctl	tvnenterkey	windowposchanged
custom09	dwcloseddropdown	lbgetcursel	omnclose	uondragdrop	windowposchanging
custom10	dwescape	lbgethorizontalextext	omndatachange	uondragenter	wininichange
custom11	dwnbackabout	lbgetitemdata	omndragdrop	uondragleave	Selected
custom12	dwndropdown	lbgetitemheight	omndragenter	uondragwithin	OMNDragLeave
custom13	dwngraphcreate	lbgetitemrect	omndragwithin	uonexternal01	nclbuttonup