

What's New

Appeon PowerBuilder® 2017 R3
FOR WINDOWS

DOCUMENT ID: DC37790-01-1700-01

LAST REVISED: July 26, 2018

Copyright © 2018 by Appeon Limited. 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.

Upgrades are provided only at regularly scheduled software release dates. 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 Limited.

Appeon and other Appeon products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of Appeon Limited.

SAP and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP and SAP affiliate company.

Java and all Java-based marks are trademarks or registered trademarks of Oracle and/or its affiliates in the U.S. and other countries.

Unicode and the Unicode Logo are registered trademarks of Unicode, Inc.

All other company and product names mentioned may be trademarks of the respective companies with which they are associated.

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

Contents

1	New Features in PowerBuilder 2017 R3	1
1.1	RESTClient supports retrieving data into DataWindowChild	1
1.2	JSON data merge and extract	1
1.3	Data encoding	1
1.4	Data encryption	1
1.5	Enhanced PDFlib method	2
1.6	Auto sign in/out	2
1.7	Enhanced PowerBuilder Compiler	2
1.8	Import/Export JSON to/from DataWindows	2
1.9	OAuth 2.0 support	3
1.10	Enhanced source control	3
1.11	PowerBuilder integrated with more PowerServer features	4
1.12	Enhanced PostgreSQL support	4
1.13	SQL Server 2017 support	5
1.14	Enhanced PMinorRevision property	5
1.15	Upgrade from 2017 or 2017 R2	5
2	Bug Fixes & Known Issues in PowerBuilder 2017 R3	6
3	New Features in PowerBuilder 2017 R2	7
3.1	Consume RESTful web services	7
3.2	Source control with SVN & Git	7
3.3	PostgreSQL ODBC driver	7
3.4	Enhanced PDFlib method	8
3.5	MobiLink synchronization supports SQL Anywhere 16 & 17	8
3.6	Enhanced PowerBuilder Compiler	8
3.7	Web Service Proxy (Wizard) available in Standard Edition	9
4	New Features in PowerBuilder 2017	10
4.1	Updated software requirements	10
4.2	Save as PDF using PDFlib	10
4.3	Enhanced OrcaScript	11
4.4	Standalone PowerBuilder Compiler	11
4.5	New RichTextEdit control	11
4.6	Built-in PowerServer Mobile	14
4.7	New license mechanism	14
4.8	Classified features	15

1 New Features in PowerBuilder 2017 R3

About this chapter

This chapter introduces the new features in PowerBuilder 2017 R3.

1.1 RESTClient supports retrieving data into DataWindowChild

RESTClient Retrieve function is enhanced to support retrieving data into the DataWindowChild object using the following syntax:

```
long Retrieve ( dwcontrol, urlname {, data} {, tokenrequest} )
```

For more about the syntax, refer to Section 10.520, “Retrieve” in *PowerScript Reference*.

DataWindowChild with the following presentation styles are unsupported: Composite, Crosstab, OLE 2.0, and RichText.

Note

RESTClient Retrieve function has changed the declaration of the ds/dw argument, from "By reference" in 2017 R2 to "By value" in 2017 R3, therefore, you might need to remove the "ref" declaration and full build the source code, in order for the source code to work after migrated from R2 to R3.

1.2 JSON data merge and extract

A new object called JSONPackage is provided to merge data in a JSON object and extract data from the JSON object:

- The data to be merged can be from DataWindow, DataStore, DataWindowChild, or from an existing JSON. Composite, Crosstab, OLE 2.0, and RichText DataWindows are unsupported.
- The resulted JSON can be posted from the client to the server via HTTPClient, or retrieved from the server to the client via RESTClient.

For more information, refer to Section 2.43, “JSONPackage object” in *Objects and Controls*.

1.3 Data encoding

A new object CoderObject is provided to encode and decode the string or blob data using the main-stream encoders such as Hex, Base64, URL etc.

For more information, refer to Section 2.7, “CoderObject object” in *Objects and Controls*.

1.4 Data encryption

A new object CrypterObject is provided to encrypt and decrypt the string or blob data using the main-stream algorithms.

For more information, refer to Section 2.14, “CrypterObject object” in *Objects and Controls*.

1.5 Enhanced PDFlib method

The following properties and function are provided for the save as PDF method using PDFlib:

- `Export.PDF.NativePDF.UserPassword` property -- for setting the PDF file open password. This feature takes no effect for files created using PDF/A standard. For more about this property, refer to Section 3.3.70, “`Export.PDF.NativePDF.UserPassword`” in *DataWindow Reference*.
- `Export.PDF.NativePDF.MasterPassword` property -- for setting the password for permissions of printing, copying, editing, etc. This feature takes no effect for files created using PDF/A standard. For more about this property, refer to Section 3.3.67, “`Export.PDF.NativePDF.MasterPassword`” in *DataWindow Reference*.
- `Export.PDF.NativePDF.Restrictions` property -- for setting the permissions of printing, copying, editing, etc. This feature takes no effect for files created using PDF/A standard. For more about this property, refer to Section 3.3.69, “`Export.PDF.NativePDF.Restrictions`” in *DataWindow Reference*.
- `SaveNativePDFToBlob` function -- for saving the generated PDF as a blob. For more about this property, refer to Section 9.135, “`SaveNativePDFToBlob`” in *DataWindow Reference*.

1.6 Auto sign in/out

Provides auto sign in/out options in the PowerBuilder login window and the account management window:

- Auto login at startup. This option is selected by default. The user who uses the online license will automatically log in the license when the PowerBuilder IDE is launched. Their user name and password will be saved for the next login.
- Sign out on exit. This option is selected by default in the virtual machine, but not selected by default in the physical machine. The user who uses the online license can automatically log out the license when the PowerBuilder IDE is closed in a normal way and with network connection.

1.7 Enhanced PowerBuilder Compiler

PowerBuilder Compiler supports to use the following deployment options as runtime parameters:

- `ProductVersionNum`. Corresponding runtime parameter: `/vn`
- `FileVersionNum`. Corresponding runtime parameter: `/fvn`

For more details about these parameters, read the PBC user guide (`pbc.pdf`) under `\Appeon\PowerBuilder 17.0\AutoCompiler`.

1.8 Import/Export JSON to/from DataWindows

Three new functions (`ImportJson`, `ImportJsonByKey`, and `ExportJson`) are provided to import JSON to or export JSON from the DataWindow control, DataStore object, or

DataWindowChild object, except for those with the following presentation styles are unsupported: Composite, Crosstab, OLE 2.0, and RichText.

For more about the function, refer to Section 9.96, “ImportJson” in *DataWindow Reference*, Section 9.97, “ImportJsonByKey” in *DataWindow Reference* and Section 9.29, “ExportJson” in *DataWindow Reference*.

The JSON string to be imported into the DataWindow must comply with the simple or standard format in *DataWindow Reference*.

For more code examples about exchanging JSON data between websites and DataWindows, refer to Chapter 18, *Using RESTful Web Services with JSON* in *Application Techniques*.

1.9 OAuth 2.0 support

The following objects are provided to support OAuth 2.0:

- TokenRequest -- gets or sets the properties for the access token request, including the address of the authorization server, the OAuth 2.0 authorization process, the scope of the access request, the secure protocol, the timeout value etc. For more information, refer to Section 2.100, “TokenRequest object” in *Objects and Controls*.
- TokenResponse -- gets the information of the access token response returned by the authorization server, including the access token, the refresh token, the HTTP response header etc. For more information, refer to Section 2.101, “TokenResponse object” in *Objects and Controls*.
- OAuthClient -- provides interfaces for obtaining the access token and protected resources. For more information, refer to Section 2.61, “OAuthClient object” in *Objects and Controls*.
- OAuthRequest -- uses the access token to obtain each property of the HTTP request of the protected resource, including the server address, the request header, the secure protocol, the timeout value etc. For more information, refer to Section 2.62, “OAuthRequest object” in *Objects and Controls*.
- ResourceResponse -- obtains the response information of the protected resource request, including the HTTP response headers and the protected resource returned from the server. For more information, refer to Section 2.82, “ResourceResponse object” in *Objects and Controls*.

RESTClient Retrieve function is extended with "tokenrequest" parameter for supporting OAuth 2.0. For more about the syntax, refer to Section 10.520, “Retrieve” in *PowerScript Reference*.

For more code examples about the OAuth 2.0 grant type, refer to Chapter 19, *Supporting OAuth 2.0 Authorization Server* in *Application Techniques*.

1.10 Enhanced source control

The SVN/Git source control is enhanced in the following aspects:

- (SVN only) Get/Release Lock options are provided to lock the object in the source control server so that other users cannot commit changes to the object.
- SVN Diff and Git Diff options are provided to compare an object in your local directory with the version of the object that was last synchronized with the source control server.
- SVN Show Log and Git Show Log options are provided to display the revision logs that are input during commit.
- Open Containing Folder option is provided to open the folder where the object is located.
- PBL is updated according to the objects in the ws_objects folder, not the PBG files; so PBG files are removed.
- (SVN only) If a target has already been added to SVN source control in R2, this target as well as all files under it must have their binary property removed first, in order to correctly resolve conflicts and merge changes in R3. For detailed instructions, refer to Section 3.2.5.1, “Remove the binary property from a R2 source-controlled workspace” in *Users Guide*.
- OrcaScript can compile source code that is under SVN/Git control. For how to use OrcaScript commands with SVN/Git, refer to Section B.4, “Usage notes for OrcaScript commands with SVN/Git” in *Users Guide*.

For more information about SVN/Git source control, refer to Section 3.2, “Using SVN source control system” in *Users Guide* and Section 3.3, “Using Git source control system” in *Users Guide*.

1.11 PowerBuilder integrated with more PowerServer features

PowerBuilder 2017 R3 is enhanced to integrate with more PowerServer features:

1. The PowerServer (PB Edition) included in the PowerBuilder Universal Edition not only supports PowerServer Mobile, but also supports the PowerServer Web related features, such as deploying Web apps to the .NET-based PowerServer back-end. You will need a free Developer license to use the PowerServer Web related features.
2. The PowerServer Mobile offline features are included in PowerBuilder 2017 R3.

The user manuals for PowerServer Web features and PowerServer Mobile offline features are provided in the PowerServer Help which can be installed with the PowerBuilder Universal Edition.

1.12 Enhanced PostgreSQL support

More connection parameters for PostgreSQL are added to the PBODB initialization file. Search "PostgreSQL" in PBODB170.INI to view the added parameter list, related syntax, and functions.

For more information about PostgreSQL, refer to Section 2.6, “PostgreSQL” in *Connecting to Your Database*.

1.13 SQL Server 2017 support

PowerBuilder 2017 R3 can connect with the Microsoft SQL Server 2017 database through the ODBC, OLE DB, ADO.NET or SQL Native Client interface.

1.14 Enhanced PBMinorRevision property

The Environment PBMinorRevision property is enhanced to reflect the point release of PowerBuilder 2017 R2 and R3 -- 0 indicates GA, 1 indicates R2, and 2 indicates R3.

1.15 Upgrade from 2017 or 2017 R2

You can upgrade PowerBuilder 2017 or 2017 R2 to 2017 R3 by running the setup program. All of your current settings and user preferences will be preserved during the upgrade.

2 Bug Fixes & Known Issues in PowerBuilder 2017 R3

The bug fixes and known issues for PowerBuilder 2017 R3 are listed in the Release Bulletin for PowerBuilder at the following links:

Bug Fixes for PowerBuilder: https://www.appeon.com/support/documents/appeon_online_help/pb2017r3/release_bulletin_for_pb/bug_fixes.html

Known Issues for PowerBuilder: https://www.appeon.com/support/documents/appeon_online_help/pb2017r3/release_bulletin_for_pb/known_issues.html

3 New Features in PowerBuilder 2017 R2

About this chapter

This chapter introduces the new features in PowerBuilder 2017 R2.

3.1 Consume RESTful web services

The following objects are provided to exchange data in JSON and consume the RESTful web services:

- HTTPClient object -- a base object for sending HTTP requests and receiving HTTP responses from a resource identified by a URI. Compared to the Inet object, the HTTPClient object is easier to use and supports more methods (Get/Post/Put/Delete) and more SSL protocols. For more about this object, refer to Section 2.37, “HTTPClient object” in *Objects and Controls*.
- JSONGenerator object -- provides the ability to construct the JSON objects with the root item being either an object or an array. It supports adding values, objects, or arrays to the constructed JSON object. For more about this object, refer to Section 2.42, “JSONGenerator object” in *Objects and Controls*.
- JSONParser object -- provides the ability to load the JSON-formatted data from a string or from a TXT file into a JSON object. For more about this object, refer to Section 2.44, “JSONParser object” in *Objects and Controls*.
- RESTClient object -- provides the ability to access the RESTful Web APIs. It loads the JSON-formatted string returned from the RESTful Web APIs into the DataWindow object. For more about this object, refer to Section 2.83, “RESTClient object” in *Objects and Controls*.

Note that these four objects cannot be referenced in .NET Web Service and .NET Assembly components, otherwise the component will fail to be deployed.

3.2 Source control with SVN & Git

PowerBuilder provides native interface options of Subversion (SVN) and Git for directly performing source control operations and functions in the PowerBuilder IDE, without needing to install the SVN or Git client. All the commonly-used SVN/Git commands are supported, including Connect to Workspace (equivalent to Checkout in SVN and Clone in Git), Add to Source Control, Commit, Revert, Resolve, Update (SVN), Pull (Git), and Push (Git).

For more information, refer to Section 3.2, “Using SVN source control system” in *Users Guide* and Section 3.3, “Using Git source control system” in *Users Guide*.

3.3 PostgreSQL ODBC driver

PowerBuilder supports the PostgreSQL ODBC driver, so PowerBuilder apps can connect with the PostgreSQL 10 (32-bit and 64-bit) (ANSI and Unicode) database through the PostgreSQL ODBC driver.

Note

The PostgreSQL database cannot be used as the data source for stored procedure; and PBODB170.INI must be configured first in order for connecting with the PostgreSQL database through ODBC interface. For how to configure the PBODB170.INI file, see Section A.1, “APPENDIX Adding Functions to the PBODB170 Initialization File” in *Connecting to Your Database*.

3.4 Enhanced PDFlib method

The following properties or settings are provided for the save as PDF method using PDFlib:

- `Export.PDF.NativePDF.ImageFormat` -- enables you to select a format for images in the PDF file when you use the PDFlib method to save data from a `DataWindow` object to a PDF file. For more about this property, refer to Section 3.3.66, “`Export.PDF.NativePDF.ImageFormat`” in *DataWindow Reference*.
- `Export.PDF.NativePDF.PDFStandard` -- enables you to select a PDF conformance level when you use the PDFlib method to save data from a `DataWindow` object to a PDF file. For more about this property, refer to Section 3.3.68, “`Export.PDF.NativePDF.PDFStandard`” in *DataWindow Reference*.
- `Export.PDF.NativePDF.UsePrintSpec` -- enables you to use the page orientation and paper size settings in the Print Specifications tab or in the `Print.Orientation` and `Print.Paper.Size` properties, when you use the PDFlib method to save data from a `DataWindow` object to a PDF file. For more about this property, refer to Section 3.3.71, “`Export.PDF.NativePDF.UsePrintSpec`” in *DataWindow Reference*.

3.5 MobiLink synchronization supports SQL Anywhere 16 & 17

PowerBuilder supports the MobiLink synchronization using SQL Anywhere 16 & 17.

3.6 Enhanced PowerBuilder Compiler

PowerBuilder Compiler supports to use the following deployment options as runtime parameters:

- "Trace information" option. Corresponding runtime parameter: `/tr y|n`
- "Error context information" option. Corresponding runtime parameter: `/in y|n`
- "Optimization" option. Corresponding runtime parameter: `/op 0|1|2`
- "Enable DEBUG symbol" option. Corresponding runtime parameter: `/bg y|n`
- "Generation options" option. Corresponding runtime parameter: `/ge 0|1|2`
- "Execution level" option. Corresponding runtime parameter: `/le 0|1|2`
- "Allow access to protected system UI" option. Corresponding runtime parameter: `/ps y|n`

For more details about these parameters, read the PBC user guide (pbc.pdf) under \Appeon \PowerBuilder 17.0\AutoCompiler.

3.7 Web Service Proxy (Wizard) available in Standard Edition

Starting from PowerBuilder 2017 R2, Web Service Proxy and Web Service Proxy Wizard are available in both the Standard Edition and Universal Edition (In PowerBuilder 2017, they are only available in Universal Edition).

4 New Features in PowerBuilder 2017

About this chapter

This chapter introduces the new features in PowerBuilder 2017.

4.1 Updated software requirements

PowerBuilder 2017 can be installed to the following Windows operating system (Windows XP and Vista are no longer supported):

- Windows 7 32- or 64-bit with SP1
- Windows 8.1
- Windows Server 2008 32- or 64-bit with SP2 (Run-time only, not development)
- Windows Server 2012 R2 (Run-time only, not development)
- Windows Server 2016 (Run-time only, not development)
- Windows 10 (newly supported in 2017)

The following database versions are newly supported in PowerBuilder 2017 compared to PowerBuilder 12.6:

- SQL Server 2016, 2014, 2012, and 2008 R2
- Oracle 12c
- SQL Anywhere 17, 16, and 12
- ASE 16
- Informix 12.x

For the other supported database types and versions, refer to Part II, “Working with Standard Database Interfaces” in *Connecting to Your Database*.

4.2 Save as PDF using PDFlib

In PowerBuilder 2017, an alternative method is provided to directly print data to PDF without needing to install any third-party tool or driver or make any configuration. This method relies on a light-weight software called PDFlib which is automatically installed with PowerBuilder and distributed with the PowerBuilder application, at no additional cost by developer. And PDFlib is automatically packaged into the PowerBuilder application executable without requiring the developer to make any configuration or selection during the building process.

For how to use this method, refer to Section 3.3.63, “Export.PDF.Method” in *DataWindow Reference* and Section 18.3.1.2, “Saving as PDF using PDFlib” in *Users Guide*.

For more about the deployment notes on PDFLib, refer to Section 35.8.2, “Using the PDFlib generator” in *Application Techniques*.

4.3 Enhanced OrcaScript

The OrcaScript tool is greatly enhanced from the following aspects:

- OrcaScript can call DOS commands and the batch file.
For more information, see Section B.1, “About OrcaScript” in *Users Guide*.
- "set appendlib" function is provided to set one library file and one PBD flag at a time, hence reduces the likelihood of messing up the PBD flags for a long library list.
For more information, see Section B.2, “OrcaScript Commands” in *Users Guide*.
- OrcaScript and Orca APIs supports compiling 64-bit executables.
For more information, see Section B.2, “OrcaScript Commands” in *Users Guide*, and PBORCA_ExecutableCreate in ORCA Guide.
- PBORCA_ApplicationRebuild eRebldType argument supports a new PBORCA_REBLD_TYPE enumerated data type -- PBORCA_3PASS.
For more information, see PBORCA_ApplicationRebuild in ORCA Guide.
- OrcaScript generates more detailed logs to help debugging.
- OrcaScript gets rid of the license control so it can be used freely outside PowerBuilder.

4.4 Standalone PowerBuilder Compiler

A standalone tool -- PowerBuilder Compiler (pbc170.exe including OrcaScript) is provided to greatly simplify the build scripts, and makes it possible to automate the entire build and deployment process outside PowerBuilder. You can install this standalone tool by running the AutoCompiler\AutoCompile.exe setup program under the PowerBuilder 2017 installation directory, for example, C:\Program Files\Apeon\PowerBuilder 17.0\AutoCompiler\AutoCompile.exe.

For how to use PBC, read the PBC user guide (pbc.pdf) in the installation folder and Section B.1, “About OrcaScript” in *Users Guide*. For how to use OrcaScript, refer to Appendix B, *The OrcaScript Language* in *Users Guide*.

4.5 New RichTextEdit control

In PowerBuilder 2017, a new RichTextEdit control is added for free use by the PowerBuilder developer and the InfoMaker developer.

This new control is applicable to the RichTextEdit control, RichText DataWindow Object, and the RichText edit style column.

For the PowerBuilder developer, they have the option to use the new control for free, or to continue using the old control (the old control will need to be purchased separately). For the InfoMaker developer, they can only use the new control; they cannot use the old control any more.

The new control has the same functions/events/properties as the old control, except for the following differences and issues (due to the different ways they are designed):

Table 4.1:

Affected area	Type	Description
Double-byte languages	Unsupported	Double-byte languages such as Korean, Japanese are not supported well in the new control.
InputFieldBackColor property	Limitation	For the new control, this property will not take effect until you save the data into a PDF file or print the data; which means, when you preview the UI in the design view or when you run the UI, you will always see the background color is gray, only when you save the data to a PDF file or print the data, you will see the background color is changed to what you set.
Wordwrap property	Limitation	For the new control, this property is always true (it cannot be set to false).
BackColor property	Difference	For the new control, if you set a negative value for the BackColor property, the returned value will be 0 and the color will be black. For the old control, if you set a negative value for the BackColor property, the returned value will be 16777215 and the color will be white.
BottomMargin/ RightMargin/ LeftMargin/ TopMargin properties	Difference	For the new control, if you set a negative value, the returned value will be 0. For the old control, if you set a negative value, the returned value will be the negative value.
Find function	Enhancement	For the new control, the Find function can not only find the text string, but also can find the carriage return characters and some special characters.
GetTextColor/ GetTextStyle functions and font settings	Difference	For the new control, this function will return the settings (text color, font name, text style etc.) of the first character of the selected text if the selected text contains more than one type of settings. For the old control, this property will return empty if the selected text contains more than one type of settings.
GetAlignment/ GetSpacing/ GetParagraphSetting functions	Difference	For the new control, this function will return the alignment (or spacing or paragraph settings) of the paragraph where the insertion point is located if more than one paragraph are selected. For the old control, this property will return null if more than one paragraph with different alignments (or spacing or paragraph settings) are selected.

Affected area	Type	Description
GetParagraphSetting function	Difference	The returned value is different between the new control and the old control due to the different units used.
Visio drawings	New feature	Visio drawings can be pasted or inserted into the new control.
Preview	New feature	All pages can be previewed (by scrolling) in the new control; while only the first page can be previewed in the old control.
SelectedPage function	Limitation	For the new control, the SelectedPage function returns the number of the page which is being viewed (rather than where the insertion point is placed).
ShowHeadFoot function	Difference	When the document is in preview mode, the old control returns 1; while the new control returns 1 and closes the preview mode, and when the parameter is true, the insertion point is placed to the header band; when the parameter is false, the insertion point is placed to the detail band.
ReplaceText function	Limitation	In the new control, the text after replaced will take over the settings (such as font name, font size etc.) carried over by the given string.
SaveDocument function	Limitation	In the new control, when saving to HTML, the image will lose some quality and will be saved as individual files separately from the document and the reference to the image file uses the absolute path,.
CopyRTF function	Difference	The new control will return a longer string which contains more types of information.
Insertion point	Difference	When the user switches the editor band (e.g. from the header/footer band to the detail band), the insertion point is placed to the last column and last row in the new control, while the insertion point's last position will be remembered and restored in the old control, if it is the first time, the insertion point is placed to the first column and first row.
Input field	Difference	The new control limits the length of the input field data to no more than 2000 characters while the old control has no such limitation.
Font	Difference	In the new control, if the user uses a non-English input method to input the English characters, the inserted characters appear like using a different font than the other characters, although they actually use the same font.
Image	Difference	In the new control, when the user tries to manually select an image (only image, no text), the user can only select it by dragging from end to beginning, but cannot select by dragging from beginning to end.
Bullet list alignment	Issue	In the new control, the bullet symbol is cut off in a rich-text edit style column under the preview mode or print mode

Affected area	Type	Description
		because the bullet list is not perfectly aligned with the body text.

4.6 Built-in PowerServer Mobile

PowerServer Mobile (formerly Apeon Mobile) is built into PowerBuilder 2017 Universal edition by default, which enables PowerBuilder developers to deploy their client/server apps to the mobile device.

Note

The Web deployment features are not available in this built-in PowerServer Mobile edition.

This built-in PowerServer Mobile edition is called PowerServer Mobile (PB Edition) which includes the following components:

- PowerServer Toolkit

A set of tools that enable the mobile application generation, maintenance, and upgrades within the PowerBuilder IDE.

IMPORTANT: The PowerServer Toolkit 2017 can deploy apps to PowerServer 2017 or 2016 (in version 2016, PowerServer is called Apeon Server).

- PowerServer Mobile (.NET platform)

A set of server components that are deployed to the .NET IIS application server and that provide the mobile apps with run-time services such as data connectivity, DataWindow support, transaction management, and security.

- PowerServer Help

A set of user documents that help PowerBuilder developers use PowerServer.

Note that PowerServer Mobile needs to be activated with an online or offline license first before it can be used. For detailed instructions, please follow the [Activating PowerServer Mobile \(PB Edition\)](#) (2017) or [Activating PowerServer Mobile \(PB Edition\)](#) (2017 R3) section in the Apeon License User Guide online help.

4.7 New license mechanism

PowerBuilder 2017 licenses are activated and managed in a totally different way than PowerBuilder 12.6 and earlier versions.

In PowerBuilder 2017, the developer is required to activate PowerBuilder with a valid online or offline license. For how to activate, refer to the [Apeon License User Guide](#) (2017) or [Apeon License User Guide](#) (2017 R3) online help.

If you are trying out the product, you should follow the [Trial License Activation Guide](#) (2017) or [Trial License Activation Guide](#) (2017 R3) online help to request and activate the trial license. The process is different.

4.8 Classified features

While new features are added in PowerBuilder, some of the existing features might no longer be needed or recommended for use. You will find the following three classifications of features in the PowerBuilder Help.

- **Discontinued** - feature that has been completely removed from the product.
For example, EAServer projects/targets, PowerBuilder .NET IDE, and Windows Form projects/targets are **discontinued** features.
- **Obsolete** - feature that is available, but is no longer eligible for technical support and will no longer be enhanced.
For example, Web DataWindow is an **obsolete** feature.
- **Stable** – feature that is available and is still eligible for technical support, but will no longer be enhanced.
For example, .NET Assembly is a **stable** feature.