

PowerBuilder and Pocket PowerBuilder Differences

This white paper describes many of the differences between PowerBuilder and Pocket PowerBuilder.

The Pocket PowerBuilder IDE is similar to that of PowerBuilder, so you can leverage your existing PowerBuilder skill-set, reusing knowledge and expertise to build and deploy mobile applications. Pocket PowerBuilder has support for the DataWindow, providing powerful data access and sophisticated programming capabilities on Windows CE devices, tight integration with Adaptive Server Anywhere (ASA), and support for MobiLink, enabling bidirectional data synchronization with heterogeneous enterprise systems.

Many PowerBuilder features are supported without modification. You can reuse code from PowerBuilder applications, although PowerBuilder projects that you import to Pocket PowerBuilder must be re-architected for deployment to handheld devices.

File name changes

Although PowerBuilder and Pocket PowerBuilder use the same file extensions for exported objects, file extensions for libraries, targets, workspaces, and resource files differ to prevent confusion for developers who work with both products. **Table 1** lists the differences between these extensions.

Table 1 Differences in File Extensions

File type	PowerBuilder Extension	Pocket PowerBuilder Extension
Workspace	PBW	PKW
Target	PBT	PKT
Library	PBL	PKL
Dynamic Library	PBD	PKD
Resource File	PBR	PKR

The file name for the Pocket PowerBuilder initialization file is *PKU.INI*. The U stands for Unicode. PowerBuilder supports ANSI character sets. Pocket PowerBuilder primarily supports Unicode, although it does allow you to import and export files with ANSI character sets.

Environment variables

In both PowerBuilder and Pocket PowerBuilder, the enumerated types for the Environment object returned by a GetEnvironment call include the OSType and CPUType. Pocket PowerBuilder has additional values for the GetEnvironment OSType and CPUType. For a Pocket PC platform, the value returned for the GetEnvironment OSType is "WindowsCE!" and the value returned for the CPUType is "ARM!"

System functions for Soft Input Panel

Pocket PowerBuilder has system functions to control the display of the Soft Input Panel (SIP) on a Windows CE device or emulator. These functions are not available in PowerBuilder applications.

Table 2: System functions for displaying and selecting the SIP

Function	Description
SetSIPPreferredState	Displays or hides the SIP used on the Pocket PC or emulator
SetSIPType	Specifies the SIP panel type (keyboard or character recognizer) used by the application on a Pocket PC or emulator

Differences required by target platform

The Windows CE API is a subset of the API for traditional Windows platforms. The most obvious difference between Windows CE and Windows 2000 or Windows XP is the screen size (real estate) available to deployed applications. There are also stylistic differences for applications deployed to Windows CE platforms.

Window Types Supported

Pocket PowerBuilder supports the following window types:

- Main
- Response

Window properties

The following PowerBuilder properties do not exist in Pocket PowerBuilder:

ContextHelp	MinBox	Resizable
ControlMenu	MDIClientColor	RightToLeft
MaxBox	PaletteWindow	WindowState

The default window object size is smaller in Pocket PowerBuilder than in PowerBuilder to fit the size of a Pocket PC screen. You can set default size properties for main and response window objects on a Size tab in the window design Options dialog box. Pocket PowerBuilder does not have a Toolbar tab page in the Properties view for windows and menus. The Toolbar page applies only to MDI windows, which are not supported on the Windows CE platform. The PowerBuilder Pointer property on the Other tab page of the window Properties view also does not exist in Pocket PowerBuilder.

The properties in Table 3 below apply only to Pocket PowerBuilder windows, and not PowerBuilder windows. They are set in the Properties view of the Window painter at design time. You cannot set these properties in a script at runtime.

Table 3: Windows properties for Pocket PowerBuilder only

Property	Description
Close	Adds an OK icon to the title bar of a main or response window that you deploy to a Windows CE platform. By default, when users click OK, user input is confirmed, the window object is destroyed, and the Pocket PowerBuilder application is closed.
SmartMinimize	Adds an X icon to the title bar of a main window that you deploy to a Windows CE platform. By default, when users click the X, the application is removed from the current navigational stack, but remains in memory for quicker availability and enhanced performance. This property and the Close property are mutually exclusive. Selecting one deselects the other. SmartMinimize is not supported on the Pocket PC 2000 emulator
MenuBar	Selecting this property makes room at the bottom of the current window for insertion of a menu. When you set a value for the MenuName property of a window, the MenuBar property is automatically selected.
DefaultSize	Selecting this property changes the size of the current window to the default size selected for the window type on the Size tab of the window design Options dialog box.
ShowSIPButton	Ensures that the SIP button (that is used to open the soft input panel) displays at the bottom of the window when you run the window on a Windows CE platform. This is selected by default.

Unsupported PowerBuilder features

Pocket PowerBuilder Target Types

Pocket PowerBuilder 1.0 supports PowerScript targets only, so the System Tree in Pocket PowerBuilder has a single Workspace tab. Pocket PowerBuilder does not currently support distributed applications or deployment to component transaction servers, and it does not include integrated source control functionality.

Using Pocket PowerBuilder with PocketSOAP

You can use Pocket PowerBuilder in conjunction with PocketSOAP to access online Web services. DLL files that define a PocketSOAP interface for Pocket PowerBuilder are available on the Sybase CodeXchange Web site at <http://powerbuilder.codexchange.sybase.com/>.

Click the Pocket PB hyperlink, open the Pocket SOAP folder, and select the *soapif.zip* file for download. A readme file is included in the Pocket SOAP folder and in the zip file on this Web site. PowerBuilder installs a *Sybase\Shared* folder that contains subdirectories with PowerBuilder DLL files, Web target files, and Java support. There is no shared folder in a Pocket PowerBuilder installation.

Wizard differences

The differences between PowerBuilder and Pocket PowerBuilder functionality are reflected in the wizards provided with the two products. Pocket PowerBuilder has wizards that support the conversion of PowerBuilder and Pocket PowerBuilder targets, as well as a wizard to generate objects that support MobiLink synchronization.

Table 4 describes differences between several PowerBuilder and Pocket PowerBuilder wizards. Wizards that differ only in the extension of the files they create are not included in this table.

Table 4 Wizard Differences

Template Application	Does not support selection of MDI or PFC application types.
Connection Object	Does not support EAServer connection; allows for entry of connection information that not in a database profile.
Application Wizard	Does not support creation of machine code EXE or DLL files; allows selection of deployment platforms; version information is more limited than in PowerBuilder 9 and is valid only for files deployed to the desktop.

Object Types

Pocket PowerBuilder does not support the standard PowerBuilder object types listed in Table 5. Additionally, external visual user objects, standard RichTextEdit and OLE visual controls are also not supported in Pocket PowerBuilder. ClassDefinition and ScriptDefinition objects, and other objects that descend from the ClassDefinitionObject object, are supported on the desktop but not on Windows CE devices or emulators.

Table 5: Unsupported standard class PowerBuilder object types

ADOResultSet	ErrorLogging	Pipeline
Connection	Inet	Profile object types
ContextInformation	InternetResult	SSLCallback
ContextKeyword	JaguarORB	SSLServiceProvider
CorbaUnion	Mail object types	Timing
DynamicDescriptionArea	OLE object types	Trace object types
DynamicStagingArea	TransactionServer	

PowerScript Language Differences

Datatypes

The longlong datatype added for PowerBuilder 9 is not supported in Pocket PowerBuilder.

Functions

The classes of functions listed below are not supported in Pocket PowerBuilder:

- Connection and TransactionServer object functions (such as CreateInstance and Lookup)
- ContextInformation, Inet, and InternetResult object functions (such as GetCompanyName, GetURL, and InternetData)
- CORBA object functions (such as BeginTransaction and Init)
- DBCS functions (such as LenW and PosW)
- DDE functions (such as CloseChannel and GetDataDDE)
- Mail functions (such as MailLogon and MailSend)
- OLE object functions (such as Length and MemberExists)
- Profile and trace functions (such as RoutineList and TraceBegin)
- Print functions (such as PrintOpen and PrintPage)
- RichTextEdit control functions (such as CopyRTF and GetTextColor)
- SSLCallback and SSLServiceProvider object functions (such as GetPin and SetGlobalProperty)

Table 6 lists PowerScript functions that are not currently implemented for the Windows CE platform. Although you are not prevented from coding these functions, if you call any of them at runtime, the calls will either be ignored, return partially valid data, or throw a system exception.

Table 6: Unsupported PowerScript functions

Activate	AddToLibraryList	Arrangesheets
CanUndo	ClassList	ClassName
Connection object functions	ContextInformation objectfunctions	CORBA object functions
CreatePage	DBCS functions	DDE functions
GetActiveSheet	GetArgElement	GetFirstSheet
GetFolder	GetLastReturn	GetLibraryList
GetRecordSet	GetToolbar	Inet object functions
InsertDocument	InsertFile	InternetResult object functions
IsAllArabic	IsAllHebrew	IsAnyArabic
IsAnyHebrew	IsArabic	IsArabicAndNumbers
IsHebrew	IsHebrewAndNumbers	LibraryDirectory
LibraryDirectoryEx	LibraryExport	LibraryImport
LongLong	Mail functions	ObjectAtPointer
OLE object functions	PageCreated	Print functions
Profiling objectfunctions	RichTextEdit functions	SetLibraryList
ShowHelp	ShowPopupHelp	SSLCallback functions
SSLServiceProvider functions	Trace functions	

Method limitations on Windows CE platforms

- The SetPointer function works only with the Hourglass! and Arrow! Values in applications deployed to Windows CE platforms. The pointer is an arrow by default. If you set the pointer to an hourglass in a desktop application, the pointer reverts to an arrow after the script is run. On a Windows CE device you must explicitly call SetPointer a second time to reset the pointer.

- On Windows CE platforms, SetRedraw (FALSE) works only for the ListBox, DropDownListBox, and TreeView controls, however SetRedraw (TRUE) forces a repaint of all control types. This can lead to unexpected performance penalties in applications that you deploy to Pocket PCs or emulators.

Events

The events listed below are not supported on the Windows CE platform. Although you are not prevented from coding these events, if you add script for any of them, it will be ignored at runtime.

CloseQuery	RemoteExec	RemoteRequest
Help	RemoteHotLinkStart	RemoteSend
HotLinkAlarm	RemoteHotLinkStop	ToolbarMoved

DataWindow objects and database support

The DataWindow types supported by Pocket PowerBuilder are:

- Freeform
- Graph
- Grid
- Group
- Tabular

Pocket PowerBuilder does not support Composite, Crosstab, Label, N-Up, OLE 2.0, and RichText.

You can print a DataWindow report from Pocket PowerBuilder only from the desktop. The “Rows to Disk” retrieve option for DataWindow objects is not available in Pocket PowerBuilder.

The ODBC driver for ASA is the database driver installed with Pocket PowerBuilder. Database drivers for OLE/DB and JDBC, as well as DBMS native drivers, are not supported. Pocket PowerBuilder also does not support data pipeline objects. If you need to access an enterprise database from a Pocket PowerBuilder application, you can use MobiLink synchronization technology or convert the enterprise database to an ASA database. To learn more about MobiLink support in Pocket PowerBuilder, visit www.sybase.com/manuals, and download the Pocket PowerBuilder *Resource Guide and Reference*. Visit section “Adding objects for MobiLink synchronization” on page 276 and the chapter on MobiLink synchronization in this guide.

Colors, presentation style, and figures

Some defaults for background colors have been changed from Btn_Face (Gray) in PowerBuilder to Window (White) in Pocket PowerBuilder. Some of the named colors on Windows machines are not supported on Windows CE devices. Unsupported colors are rendered in black on these devices. Controls in Pocket PowerBuilder default to a 2D presentation style. If a 3D look is selected for a control, it might not have the desired appearance when deployed to a Windows CE device. Different versions of the Windows CE platform can vary in their support of 3D controls. Pocket PowerBuilder supports GIFs, BMPs, and stock icons for picture controls. Other picture files (JPGs, WMFs, RLEs, and cursor files) are not currently supported.

Deployment and Runtime Differences

Debug and Deployment Options

For a Pocket PowerBuilder project, you specify deployment options that are not available in PowerBuilder. You can select any or all of the following deployment options:

- Device (ARM)
- Desktop
- PPC Emulator 2000
- PPCEmulator 2002

An application that you deploy to the desktop will look slightly different from the same application deployed to a PDA device or a Pocket PC emulator. The desktop application has its own title bar with a maximize, minimize, and close button. Even if you select Close or SmartMinimize icons for a window, these do not display in the window when it is run or debugged on the desktop. **Desktop deployment is for testing and demonstration purposes only.** When you debug an application, you do not have access to the Tip Watch or Quick Watch views that were added to PowerBuilder in version 9. You must debug an

application from the Pocket PowerBuilder IDE; you cannot run the Pocket PowerBuilder debugger with a deployed application.

Running Applications on an Emulator or PDA Device

If you deploy applications to a Pocket PC emulator, Pocket PowerBuilder starts the emulator. Pocket PowerBuilder has toolbar icons and Tools menu items that launch the Pocket PC 2000 emulator and the Pocket PC 2002 emulator. You can download the Pocket PC 2000 SDK and the Pocket PC 2002 SDK from the Microsoft Web site. These SDKs include emulators that you can use as target platforms for your Pocket PowerBuilder applications.

Using the Windows CE Start Menu

By default, Pocket PowerBuilder deploys applications to the \Program Files directory of a Windows CE device or emulator, but you can change the deployment directory in the Project painter. On a Windows CE device, users can run the Pocket PowerBuilder applications you deploy by clicking on an application executable file in the directory where it is deployed. Users can locate deployed executable files with the Windows CE File Explorer.

You can also deploy applications to a directory that you create in the Start Menu of the Windows CE target platform, such as *\Windows\Start Menu\My Programs*. That way, users will be able to find the deployed applications quickly using the Start Menu. Another option is to use the built-in application list as a selection vehicle for deployed applications.

Using the Built-in Application List

Users can display the built-in application list by selecting *Pocket PowerBuilder 1.0* from the Start menu. The list is empty until you begin deploying applications. The application list includes a directory browser that lets users select a directory containing deployed applications. By default, the application list displays executable files that you deploy to the *\Program Files* directory of the Windows CE device. Users can launch any Pocket PowerBuilder application that you deploy to the device by selecting the application from this application list. The application list is present on the device only if the complete Pocket PowerBuilder runtime package is installed. If you install only the Pocket PowerBuilder runtime DLLs to the device, the application list is not available.

Connecting to an ASA database on Windows CE

In the development environment, the ODBC driver manager provides an interface between the Pocket PowerBuilder ODB interface (pkodb10u.dll) and the ASA version 8 ODBC driver (dbodbc8.dll). The driver manager can handle three types of data source name (DSN) files: system DSNs, user DSNs, and file DSNs. You can create a database connection profile using any of these DSN types.

In applications that you deploy to Win CE, you must use a file DSN because there is no ODBC driver manager.

Using file DSNs

Windows CE does not provide an ODBC driver manager or an ODBC Administrator. On Windows CE, Adaptive Server Anywhere uses ODBC data sources stored in ANSI format files. The file has the same name as the data source, with the extension .dsn, and is usually stored at the root level of the device.

Windows CE also searches for data source files in the following locations:

The directory from which the ODBC driver (dbodbc8.dll) was loaded. This is usually the Windows directory.

The directory specified in the Location key of the Adaptive Server Anywhere section of the registry. This is usually the same as the Adaptive Server Anywhere installation directory. The default installation directory is: \Program Files\Sybase\ASA.

You can specify either the DSN or the FILEDSN keyword to use file data source definitions in a script. On

Windows CE, DSN and FILEDSN are synonyms.

The data source typically specifies the location of the database and the database engine. For more information about defining data sources, see "Defining an ASA data source".

Starting ASA

To start an ASA database on a Pocket PC device or emulator, use the File Explorer (in the Programs folder) to navigate to the location where ASA is installed, usually Program Files\Sybase\ASA . Tap dbsrv8, use the drop-down lists or the Soft Input Panel to complete the Server Startup Options dialog box, and tap OK.

Your application can start the database automatically by specifying the properties of a Transaction object and issuing a CONNECT statement.

Using a different ASA ODBC driver

The default ASA ODBC driver supplied with Adaptive Server Anywhere version 8 is dbodbc8.dll. You can specify a different ASA ODBC driver by including "driver=dbodbcX.dll" in the ConnectString parameter in the DBParm value or in the file data source, where X is the version number of the ASA ODBC driver.

For example, to use an ASA 9 ODBC driver on the Windows CE device, you can set the following DBParm value in a script (SQLCA is the default connection object):

```
SQLCA.DBPARM="ConnectString='DSN=myDSN;driver=dbodbc9.dll;UID=dba;PWD=sql'"  
CONNECT using SQLCA;
```

If you are running your Pocket PowerBuilder applications from the desktop, use the actual name of the driver (for example, "Adaptive Server Anywhere 9.0") in the ConnectString parameter rather than the name of the DLL. Otherwise, the driver you select in the connection string might be ignored.

The DSN you assign must exist in the root directory on the Windows CE device or emulator, or in the \Windows directory or the directory from which the server was started. You can include the driver name in the DSN instead of the DBParm by adding an assignment for the driver in the DSN file:

```
[ODBC]  
enginename=asademo  
databasename=asademo  
databasefile=\Program Files\Sybase\ASA\asademo.db  
start=\Program Files\Sybase\ASA\dbsrv9.exe  
Driver=dbodbc9.dll
```

Preventing the ASA log screen from displaying

You can add a -q switch to the start line in the DSN file to prevent the ASA log screen from displaying when you connect to the database: start=\Program Files\Sybase\ASA\dbsrv8.exe -q

Because the connection might take a few moments, it is a good idea to call the SetPointer function to display the Windows CE version of the hourglass icon when using the -q switch. For information about the use of SetPointer on the Windows CE platform, see the note on "Method limitations on Windows CE platforms" in this appendix.

CAB file creation and distribution

In Pocket PowerBuilder you can generate a CAB file with all the objects from a project and the project executable file. You can use the CAB file to distribute the project to multiple Pocket PC devices.

Resizing and moving controls

Pocket PowerBuilder painters do not allow you to set properties for resizing and moving controls at runtime. However, you can still give users the ability to move and resize controls by modifying these control properties in code.