PowerServer Mobile Offline Tutorials

Appeon® PowerServer® 2020

FOR WINDOWS & UNIX & LINUX

DOCUMENT ID: ADC50002-01-2020-01

LAST REVISED: March 25, 2020

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

Contents

1 Tutorial 1: Set up the Environment

In this tutorial, you will go through the tasks to set up the PowerServer Mobile environment for running the offline mobile application.

1.1 Task 1: Prepare the machine

The simplest scenario will be used in this tutorial, which requires 1 Windows PC and 1 mobile device.

- **1 Windows PC**: used as the development machine, the application server (PowerServer), and the synchronization server (MobiLink Server)
- **1 mobile device**: in this tutorial, 1 Android device is used as the mobile client

The **PowerServer Mobile for .NET** edition will be used to walk you through this tutorial, so please prepare the environment according to the following requirements.

Software requirements for **Windows PC**:

- Windows 7 (32-bit or 64-bit)

  PowerServer Mobile 32-bit version can be installed to the 64-bit OS without any special configurations. But if you have installed **Appeon PowerServer 64-bit** version before please uninstall it first. See Task 1.2: Uninstall previous version of 64-bit PowerServer.

- PowerBuilder 2019 R2
- .NET Framework 4.x

  Download the .NET Framework 4.0 setup program from http://www.microsoft.com/en-us/download/details.aspx?id=17718.

- IIS 7.5

  See Task 1.3: Install IIS and Task 1.4: Configure IIS.

- (for UltraLite only) MobiLink Server 17

  To synchronize a local UltraLite database with the server-side database, you must install MobiLink Server 17 which is included in SQL Anywhere 17. If your UltraLite and MobiLink server are not at the required version (UltraLite 17 and MobiLink server 17), you will need to upgrade them according to the SQL Anywhere 17 help.

  IMPORTANT: If you are new to the MobiLink Server, it will be helpful for you to read this article first: http://support.appeon.com/index.php?/Knowledgebase/Article/View/63/16/how-to-deploy-the-mobilink-offline-application.

- Internet Explorer 8, 9, 10, or 11
- PowerServer 2020

  Install PowerServer and PowerServer Toolkit. See Task 2: Install PowerServer & PowerServer Toolkit.

Software requirements for the Android device:

- Google Android 6.x (32-bit), 7.x, 8.x, 9.x, or 10.x
- Appeon Workspace

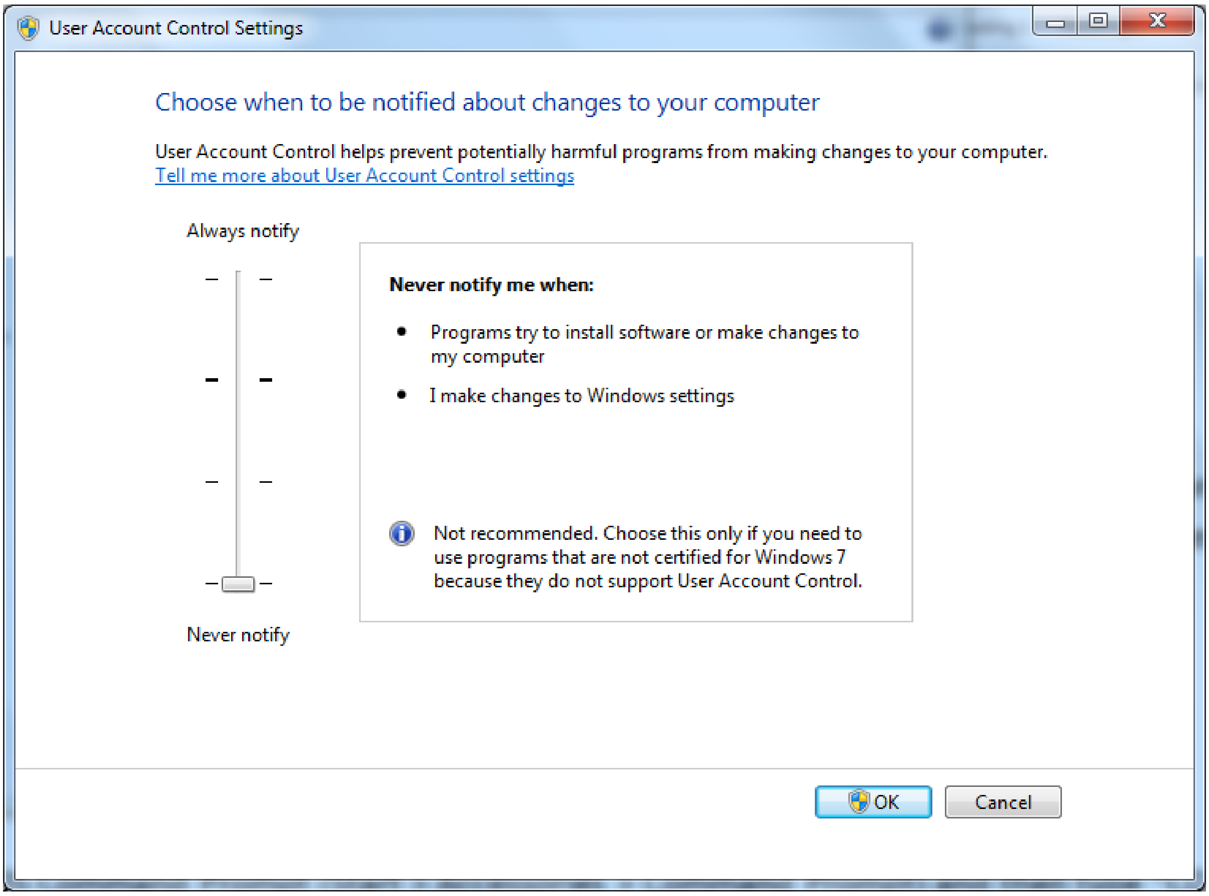
  See Task 3: Install Appeon Workspace (on the Android device).
- UltraLite 17 or SQLite 3.x database

  The database will be automatically downloaded from the server to the mobile device when the application is installed, if you have successfully configured the database in PowerServer Toolkit and deployed it along with the application to server. See Task 3: Configure the offline settings in Tutorial 4 for how to configure the UltraLite or SQLite database in PowerServer Toolkit.

1.1.1 Task 1.1: Disable UAC (User Account Control)

On the Windows PC, go to the **Control Panel** > **User Accounts** and disable UAC by setting the slider to "**Never notify**" (see screenshot below). After you have disabled UAC you **MUST** restart the computer. If you do not restart the computer the change will NOT take effect.

Figure 1.1: Disable UAC



User Account Control Settings

Choose when to be notified about changes to your computer

User Account Control helps prevent potentially harmful programs from making changes to your computer.
Tell me more about User Account Control settings

Always notify

Never notify

Never notify me when:

- Programs try to install software or make changes to my computer
- I make changes to Windows settings

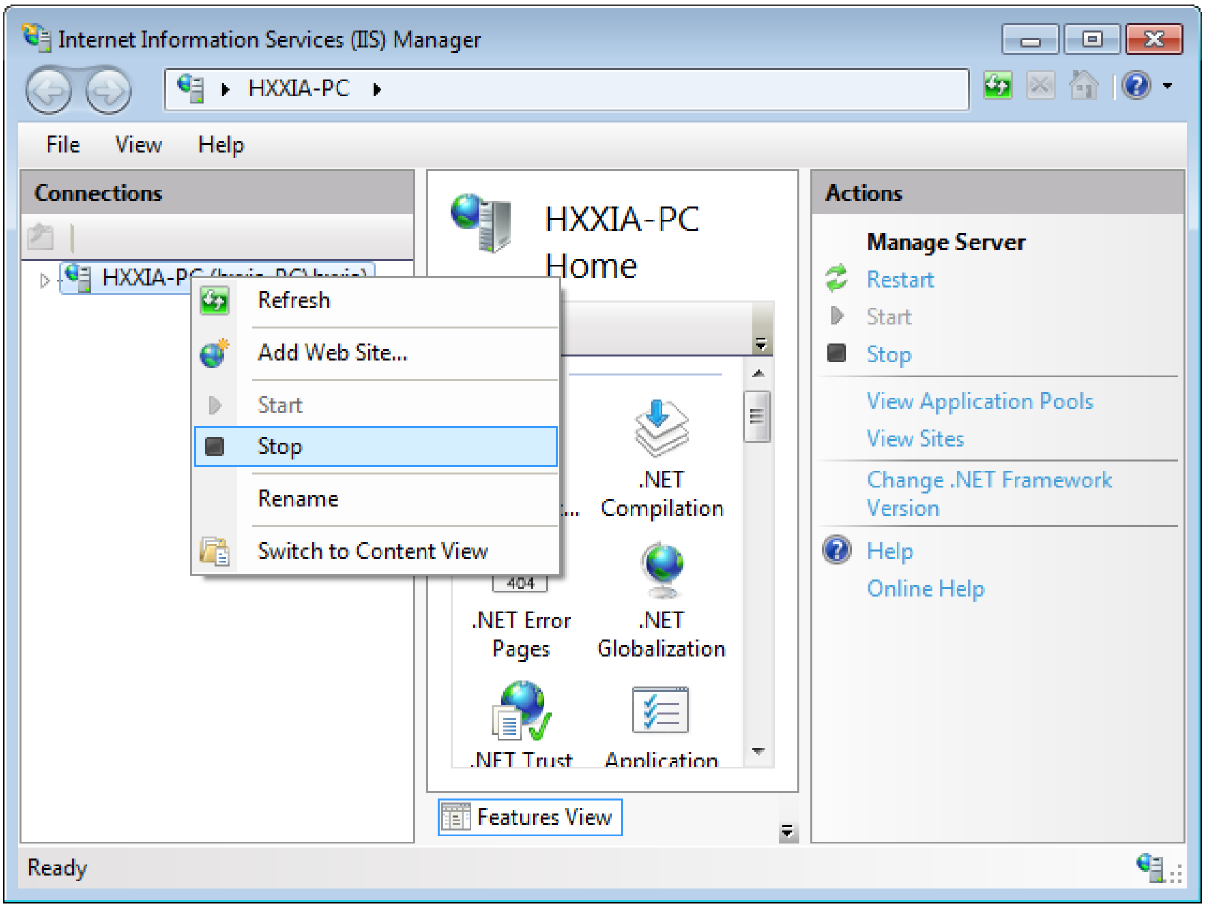
Not recommended. Choose this only if you need to use programs that are not certified for Windows 7 because they do not support User Account Control.

1.1.2 Task 1.2: Uninstall previous version of 64-bit PowerServer

If you are using a 64-bit machine you can install **PowerServer Mobile (32-bit)** without any special configurations. But if you have a previous version of **Appeon PowerServer (64-bit)** already installed on this machine then you must uninstall it first.

Step 1: Open **IIS Manager**, right click the **top** node (not the website node) in the treeview and select **Stop** from the popup menu. This will stop the entire IIS.

Figure 1.2: Stop the entire IIS Manager



Step 2: Close any opened window, especially **PowerBuilder** and **IIS Manager**.

Step 3: Uninstall all of the PowerServer components including **PowerServer**, **PowerServer Toolkit**, and **PowerServer Help**. You will need to uninstall these components one by one.

Step 4: Verify PowerServer is cleanly uninstalled by the following two steps:

- Double check the **Control Panel\Programs\Programs and Features** and make sure no PowerServer component is listed.

- Open a command prompt window and then type **regedit**<Enter>. Double check that no **ADT** or **ASN** keys are listed under HKEY_LOCAL_MACHINE\SOFTWARE\Appeon \<version_number>.

Step 5: Clear the Internet Explorer cache and temporary files.

Step 6: Delete the entire Appeon folder from C:\Users\User_Name\AppData\Roaming\.

Step 7: Delete all the PowerServer application folders from the IIS Web root. For example, under *C:\inetpub\wwwroot\* at minimum you should delete the following folders: appeon, appeon_acf_demo, appeon_code_examples, pet_world, sales_application_demo.

Step 8: Restart the machine.

Step 9: Start IIS by right-clicking the **top** node in the treeview in the IIS Manager and selecting **Start** from the popup menu.

After that, you can proceed to install PowerServer Mobile by following steps in Task 2: Install PowerServer & PowerServer Toolkit.

1.1.3 Task 1.3: Install IIS

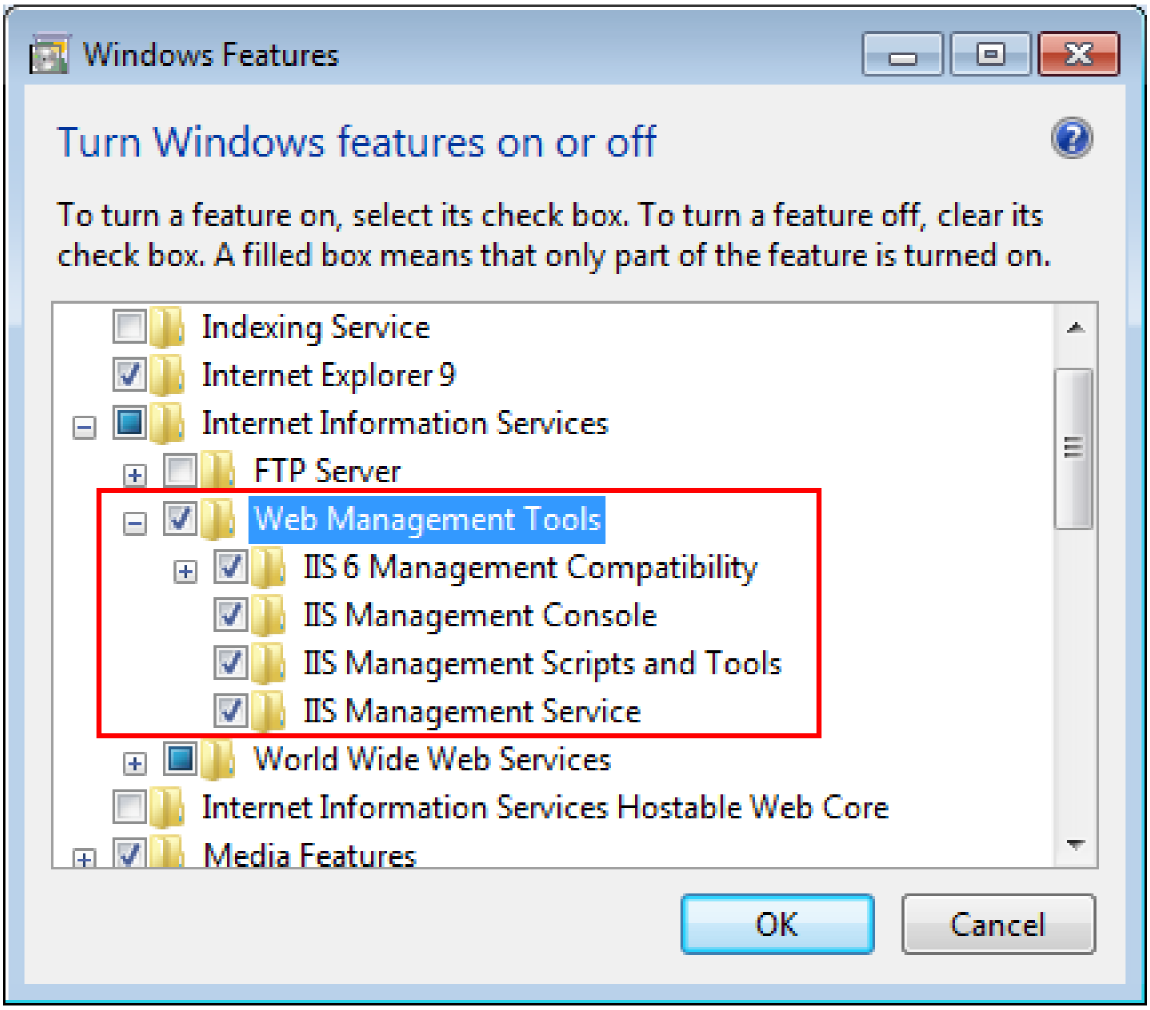
IIS 7.5 is not installed on Windows 7 by default. You need to manually install it. Before you install IIS 7.5 make sure you have installed .NET Framework 4.0. If it is not installed please download the setup program from http://www.microsoft.com/en-us/download/details.aspx?id=17718 and install .NET Framework 4.0 first.

Below are steps for installing IIS.

Step 1: Click **Start** and then click **Control Panel**. In **Control Panel**, click **Programs** and then click **Turn on or off Windows features**.

Step 2: Select the check box of **Internet Information Services**, then expand the list and select the items under **Web Management Tools, Application Development Features** and **Common HTTP Features** according to the figure below. Click **OK** to let Windows finish the install.

Figure 1.3: Select the Web Management Tools



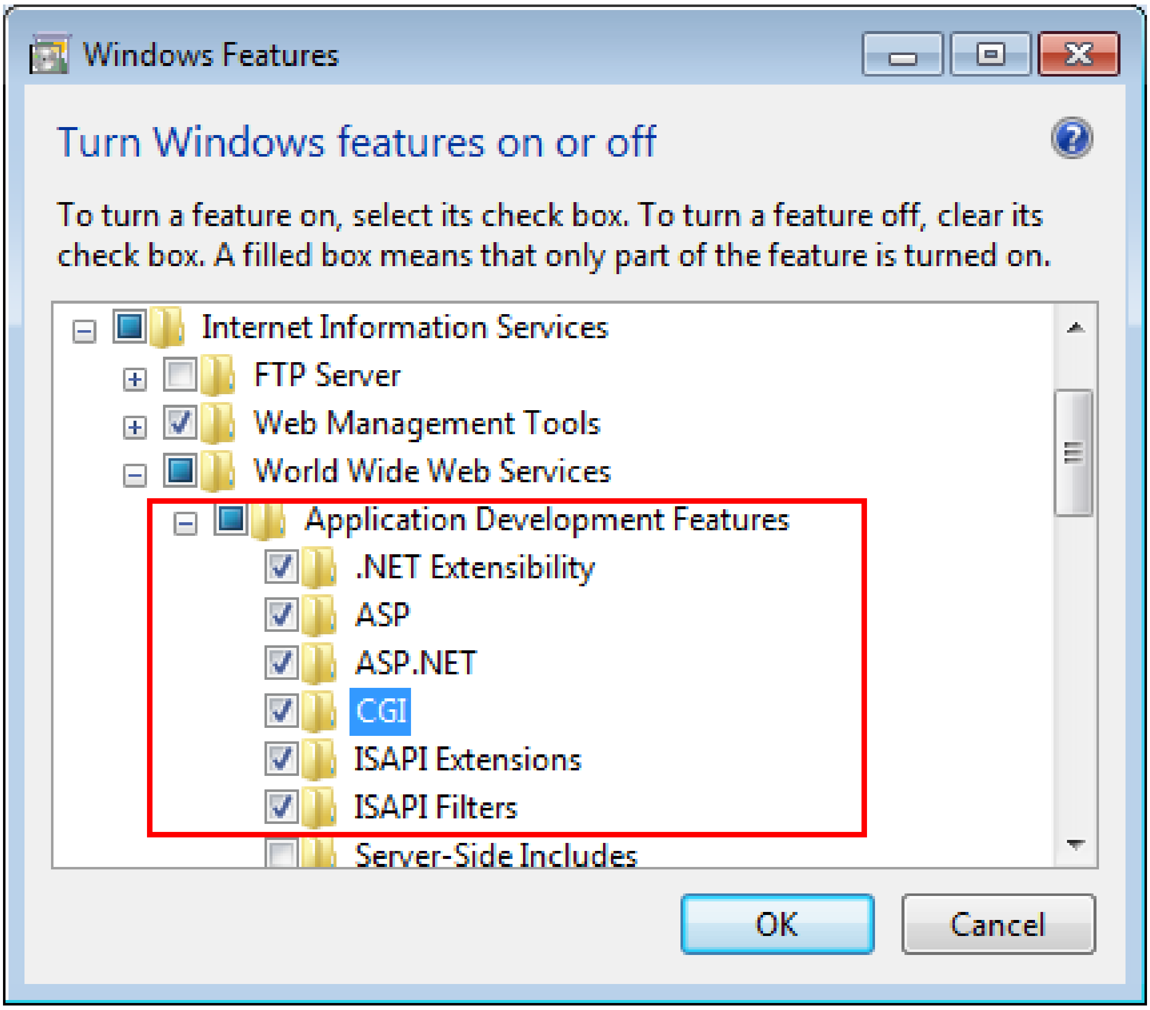
Windows Features

Turn Windows features on or off

To turn a feature on, select its check box. To turn a feature off, clear its check box. A filled box means that only part of the feature is turned on.

- □ Indexing Service
- ☑ Internet Explorer 9
- ■ Internet Information Services
  - □ FTP Server
  - ☑ Web Management Tools
    - ☑ IIS 6 Management Compatibility
    - ☑ IIS Management Console
    - ☑ IIS Management Scripts and Tools
    - ☑ IIS Management Service
  - ■ World Wide Web Services
- □ Internet Information Services Hostable Web Core
- ☑ Media Features

Figure 1.4: Select the Application Development Features



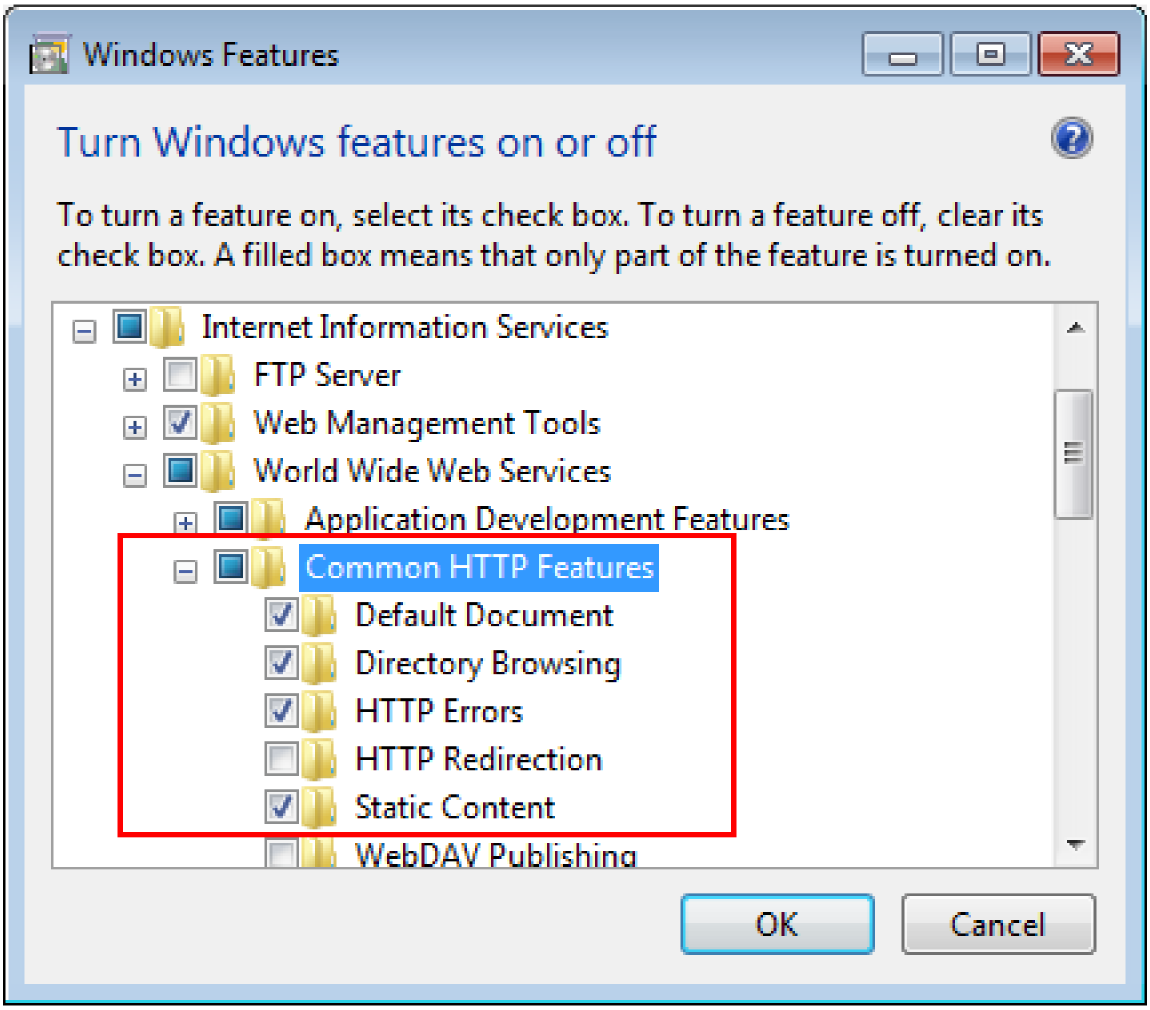
Windows Features

Turn Windows features on or off

To turn a feature on, select its check box. To turn a feature off, clear its check box. A filled box means that only part of the feature is turned on.

- ■ Internet Information Services
  - □ FTP Server
  - ☑ Web Management Tools
  - ■ World Wide Web Services
    - ■ Application Development Features
      - ☑ .NET Extensibility
      - ☑ ASP
      - ☑ ASP.NET
      - ☑ CGI
      - ☑ ISAPI Extensions
      - ☑ ISAPI Filters
      - □ Server-Side Includes

Figure 1.5: Select the Common HTTP Features



Turn Windows features on or off

To turn a feature on, select its check box. To turn a feature off, clear its check box. A filled box means that only part of the feature is turned on.

- ■ Internet Information Services
  - □ FTP Server
  - ☑ Web Management Tools
  - ■ World Wide Web Services
    - ■ Application Development Features
    - ■ Common HTTP Features
      - ☑ Default Document
      - ☑ Directory Browsing
      - ☑ HTTP Errors
      - □ HTTP Redirection
      - ☑ Static Content
      - □ WebDAV Publishing

Step 3: After IIS is installed, go to **IIS Manager**, right click the **Default Web Site**, select **Binding**, and make sure **IP Address** is indicated with an asterisk "*". If not, please click **Edit** and select **All Unassigned** for the IP address, this will mark **IP address** as an asterisk "*".

Step 4: Run http://*IP_Address*:80/ in Internet Explorer. If the IIS welcome screen displays, then IIS is working properly.

Tip: to obtain the IP address of the server, open a command prompt window and then type **ipconfig**<Enter>. Remember this IP address as it is also needed when you configure the mobile app in Appeon Workspace.

If IIS is not working, please re-install IIS or fix the IIS configuration by following the IIS help.

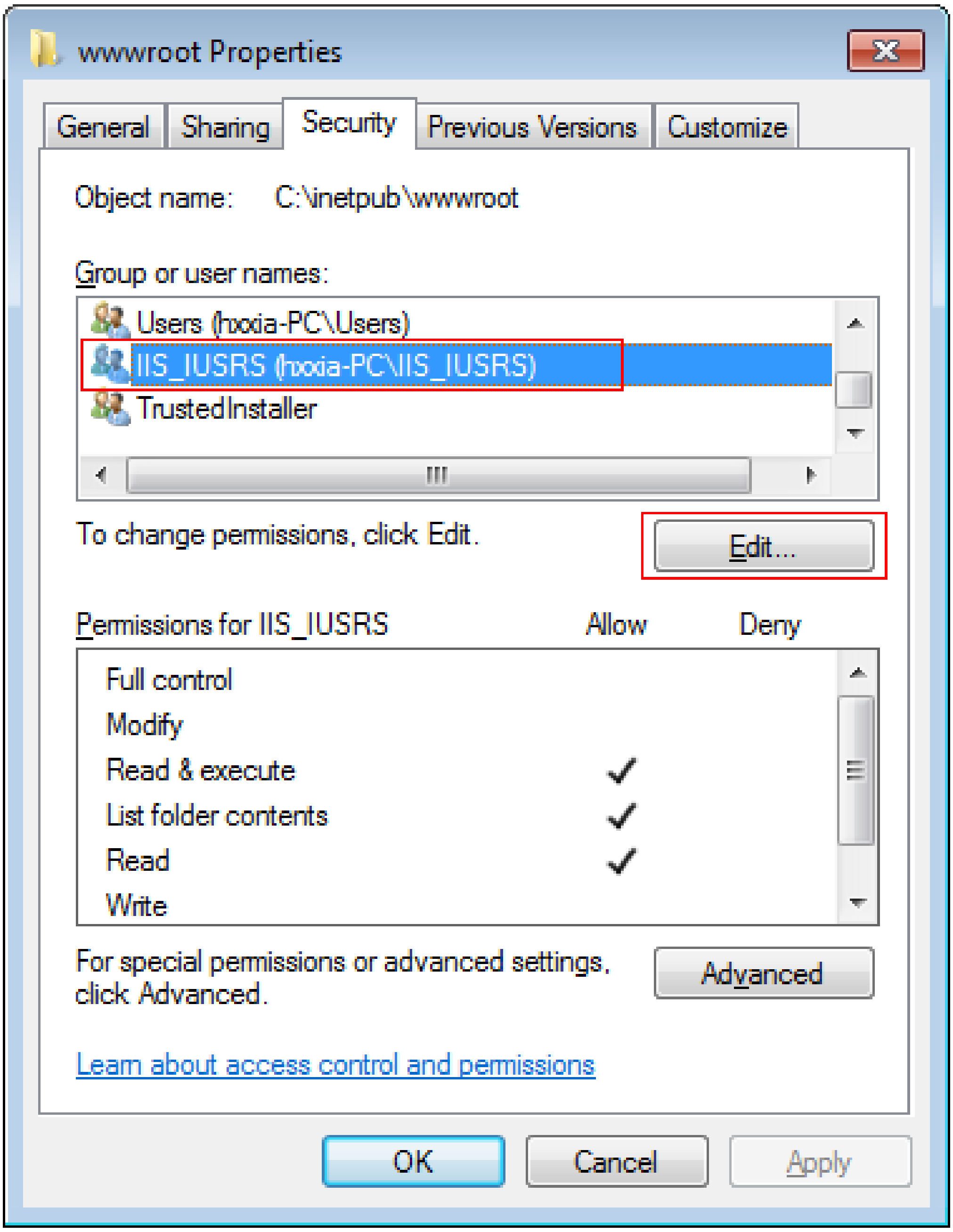
1.1.4 Task 1.4: Configure IIS

Follow steps below to grant **IIS_IUSRS** account with full controls to the IIS Web root folder:

Step 1: Right click **C:\Inetpub\wwwroot** and select **Properties** from the popup menu.

Step 2: On the **Security** tab, select **IIS_IUSRS** in the **Group or user names** list box, and then click the **Edit** button.

Figure 1.6: Select IIS_IUSRS



Object name: C:\inetpub\wwwroot

Group or user names:

- Users (hxxia-PC\Users)
- IIS_IUSRS (hxxia-PC\IIS_IUSRS)
- TrustedInstaller

To change permissions, click Edit.

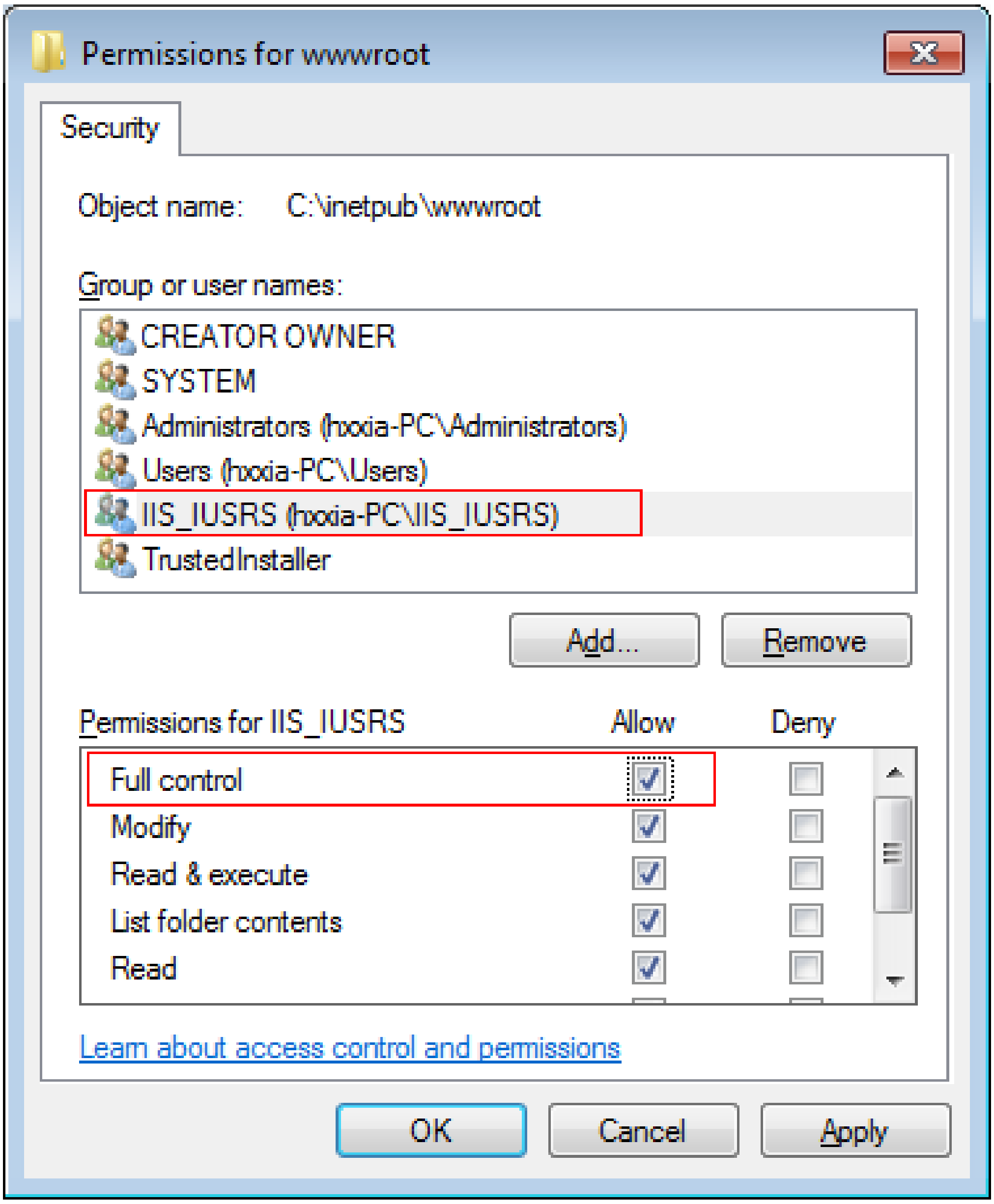
Permissions for IIS_IUSRS	Allow	Deny
Full control		
Modify		
Read & execute	✓	
List folder contents	✓	
Read	✓	
Write		

For special permissions or advanced settings, click Advanced.

Learn about access control and permissions

Step 3: Select **IIS_IUSRS** in the **Group or user names** list box, and then select the **Allow** check box for the **Full control**.

Figure 1.7: Select Full control for IIS_IUSRS



Object name: C:\inetpub\wwwroot

Group or user names:

- CREATOR OWNER
- SYSTEM
- Administrators (hxxia-PC\Administrators)
- Users (hxxia-PC\Users)
- IIS_IUSRS (hxxia-PC\IIS_IUSRS)
- TrustedInstaller

Permissions for IIS_IUSRS	Allow	Deny
Full control	☑	□
Modify	☑	□
Read & execute	☑	□
List folder contents	☑	□
Read	☑	□

Step 4: Click **OK** to save the settings.

1.2 Task 2: Install PowerServer and PowerServer Toolkit

1.2.1 Task 2.1: Install PowerServer and PowerServer Toolkit

On the Windows PC, install **PowerServer** and **PowerServer Toolkit** by the following steps.

Step 1: Start IIS server: open **IIS Manager**, right click the **top** node (not the website node) in the treeview and select **Start** from the popup menu.

Step 2: Launch the PowerServer setup program by double-clicking **setup.exe**. Click **Next** until you reach the screen for selecting the product components. Make sure the checkboxes for **PowerServer for .NET** and **PowerServer Toolkit** are selected.

Step 3: Click **Next** until you reach the screen for specifying the IIS Web site where PowerServer will be installed. Make sure **Select an existing Web Site** and **Default Web Site** are selected.

Step 4: Click **Next**. Choose whether to install the **Demo Applications**. If you have SQL Anywhere database server installed, select to install the demo; otherwise, do not install the demo since the demo requires SQL Anywhere database server to be previously installed.

Step 5: If you selected to install the **Demo Applications**, specify the path for the SQL Anywhere database server engine.

Step 6: Click **Next** until Setup begins copying files for the PowerServer installation.

When PowerServer installation is completed, PowerServer Toolkit installation automatically starts.

Step 7: Click **Next** until Setup begins copying files for the PowerServer Toolkit installation.

Step 8: When the installation is completed, click **Finish**.

1.2.2 Task 2.2: Verify the installation

On the Windows PC, run http://*IP_Address*:80/AEM. If AEM is launched successfully, then PowerServer is installed successfully.

1.3 Task 3: Install Appeon Workspace (on the Android device)

Step 1: Enable the **Unknown resources** option (in **Settings** > **Security**) on the Android device so you can install apps that are not downloaded from Google Play.

Step 2: Make sure your Android device can connect to PowerServer.

Step 3: Visit the Appeon Workspace download center that is posted on PowerServer (http://*Server_IP*:80/AWS), and then click the download button.

1.4 Task 4: Configure the network connection

The offline mobile application will connect with PowerServer for the first time it runs, therefore, make sure your Android device can connect with PowerServer.

Check and make sure the Windows PC (where PowerServer is installed) and the Android device connected to the same Wi-Fi router. To verify this, on the Android device, open the the Web browser and type http://*Server_IP*:80/AEM. If AEM is launched successfully, it means that the PowerServer is properly installed and that the Android device is able to connect to the Windows PC.

NOTE: Mobile Internet is supported by PowerServer Mobile. However, in order for the Appeon Workspace to connect to your PowerServer you will need an external IP address. You can verify that your external IP address is properly working by typing *http://External_IP_Address:80* into a Web browser of any device connected to the Internet. If you get a page not found error or other HTTP error then your network is not configured properly for external access.

2 Tutorial 2: Create a local database

The local database can be either an UltraLite 17 database or a SQLite 3.x database. This section will only provide useful links where you can find detailed instructions to create an UltraLite database or a SQLite database for use on the local mobile device, because this is a non-PowerServer task.

2.1 Create an UltraLite database

There are many different ways to create an UltraLite database for mobile devices, as described in the SQL Anywhere online documentation.

It is recommended that you go through this article where we put together the most relevant topics to guide you through creating a local UltraLite database from the MobiLink synchronization model, creating the MLSync synchronization objects in PowerBuilder, and deploying the app and the database thru PowerServer.

It is also worth mentioning that when you create an UltraLite database, make sure that you keep its character sets to the default UTF-8, and that the character sets used in this UltraLite database and the consolidated database are compatible and with the same length, otherwise data could be lost or become altered in unexpected ways during the synchronization.

2.2 Create a SQLite database

You can either use the SQLite command-line utility named **sqlite3.exe** to create and manage the SQLite database, or choose from the list of management tools to create and manage the database.

3 Tutorial 3: Modify the PB App

This tutorial will walk you through modifying the original PB application so that after deployed to mobile it can connect with the local UltraLite/SQLite database and synchronize the local UltraLite database with the server-side consolidated database.

3.1 Connect with the local database

The following instructions are applicable to both the UltraLite database and the SQLite database.

To connect with the local database, you should first add a special parameter called **EnabledLocalDB** to allow the mobile application to establish a local database connection, this is described in this section, and then you should configure the connection settings for the local database in PowerServer Toolkit, this is described in Task 3: Configure the offline settings in Tutorial 4.

Step 1: Add the **EnabledLocalDB** parameter to the **DBParm** property of the transaction object, and set its value to **True** (or **Yes** or **1**). For example,

```
SQLCA.DBMS = "ODBC"
SQLCA.AutoCommit = False
SQLCA.DBParm =
 "ConnectString='DSN=Mobile15TestDB;UID=DBA;PWD=sql'; EnabledLocalDB='true'"
```

Notes:

- **EnabledLocalDB** could have the following value pairs: 'true'/'false', 'yes'/'no', 1/0. When it is set to 'true', 'yes' or 1, it indicates that the mobile app is able to connect with the local/ offline database on the mobile device.

- The **EnabledLocalDB** parameter is specially provided and supported by PowerServer Mobile only, therefore, it will be ignored when executed in the PB application and the Web application, and will only take effect when executed in the mobile application.

- The **ConnectString** parameter will be ignored when executed in the mobile application, because the connection with the local database (such as user name, password etc.) is configured separately in PowerServer Toolkit. For details, refer to Task 3: Configure the offline settings in Tutorial 4.

Step 2: Choose the method to connect with the local/offline database: by using the PB transaction object or by using the **CacheName** parameter of the PB transaction object, according to the database connection scenarios in the application.

Scenario 1: Connect with one local database in one transaction object. This method requires no special coding. You only need to input the transaction object name and configure the database file in PowerServer Toolkit. For detailed instructions, refer to Task 3: Configure the offline settings in Tutorial 4.

Scenario 2: Connect with more than one local database in one transaction object. This method requires special coding as well as configuration in PowerServer Toolkit. You will first need to create the data source for the corresponding database file in PowerServer

Toolkit; if there are multiple local databases, then you should create multiple data sources; and then you need to add the **CacheName** parameter to the **DBParm** property of the transaction object and set its value to the data source.

For detailed instructions on how to create the data source in PowerServer Toolkit, refer to Task 3: Configure the offline settings in Tutorial 4. The **CacheName** parameter will take precedence over the transaction object if both transaction object and data source are configured in PowerServer Toolkit.

Here is the PowerBuilder code example for using the **CacheName** parameter to connect with the local/offline database. The **CacheName** parameter will be handled differently when executed in the mobile application than when executed in the PB application.

```
SQLCA.DBMS = "ODBC"
SQLCA.AutoCommit = False
SQLCA.DBParm =
 "ConnectString='DSN=Mobile15TestDB;UID=DBA;PWD=sql'; EnabledLocalDB='true';
 CacheName='MyUltralite'"
connect;
……
disconnect;
SQLCA.DBMS = "ODBC"
SQLCA.AutoCommit = False
SQLCA.DBParm =
 "ConnectString='DSN=Mobile15TestDB;UID=DBA;PWD=sql'; EnabledLocalDB='true';
 CacheName='MySQLite'"
connect;
```

3.2 Synchronize data thru MLSync object (for UltraLite only)

Note: The following instructions are applicable to UltraLite database only. If you are using a local SQLite database, since SQLite does not support synchronizing data via MobiLink server, you will need to write scripts to synchronize the SQLite database with the server-side consolidated database.

To synchronize the local UltraLite database with the server-side consolidated database via MobiLink server, use the PB synchronization object **MLSync**. For details, refer to Working with PowerBuilder synchronization objects in the SQL Anywhere online documentation.

When synchronizing the UltraLite database with the consolidated database via Mobilink, the **ScriptVersion** property is required, however this property does not exist in the PB **MLSync** object. As a replacement, you could use the **sv** parameter of the **ExtendedOpt** property, for example:

```
mySync_1.ExtendedOpts = "sv = test"
```

If you do not specify the **sv** parameter of the **ExtendedOpt** property, then PowerServer Mobile will use the value of **Publication** for **ScriptVersion**.

Please note that the **MLSync** object is originally designed by PB to synchronize the local database on the PC only, and the local database on the PC can only be a SQL Anywhere database, and cannot be an UltraLite database; while in Android, the local database can only be an UltraLite database and cannot be a SQL Anywhere database, therefore, some features of the **MLSync** object are implemented differently by PowerServer Mobile in order to work with the UltraLite database on Android. For details, refer to Section 3.3.1.19, “MLSync” in *Supported PB Features for PowerServer Mobile*.

3.3 See also: Reduce SQLite database file size

The SQLite database file may get larger and larger as time goes by. Even if the data is deleted from the database, the database file will not be shrunk, unless you run the VACUUM command. VACUUM will reconstruct the database from scratch and keep the database minimal in size. Thus, it is a recommended good practice to execute VACUUM periodically to reduce the SQLite database file size. But notice that you should use different ways to execute VACUUM for PowerBuilder and PowerServer Mobile.

In PowerBuilder, you can execute the VACUUM command directly, for example:

```
vacuum
```

While in PowerServer Mobile, you will have to use Dynamic SQL Format 1 to execute the VACUUM command, for example:

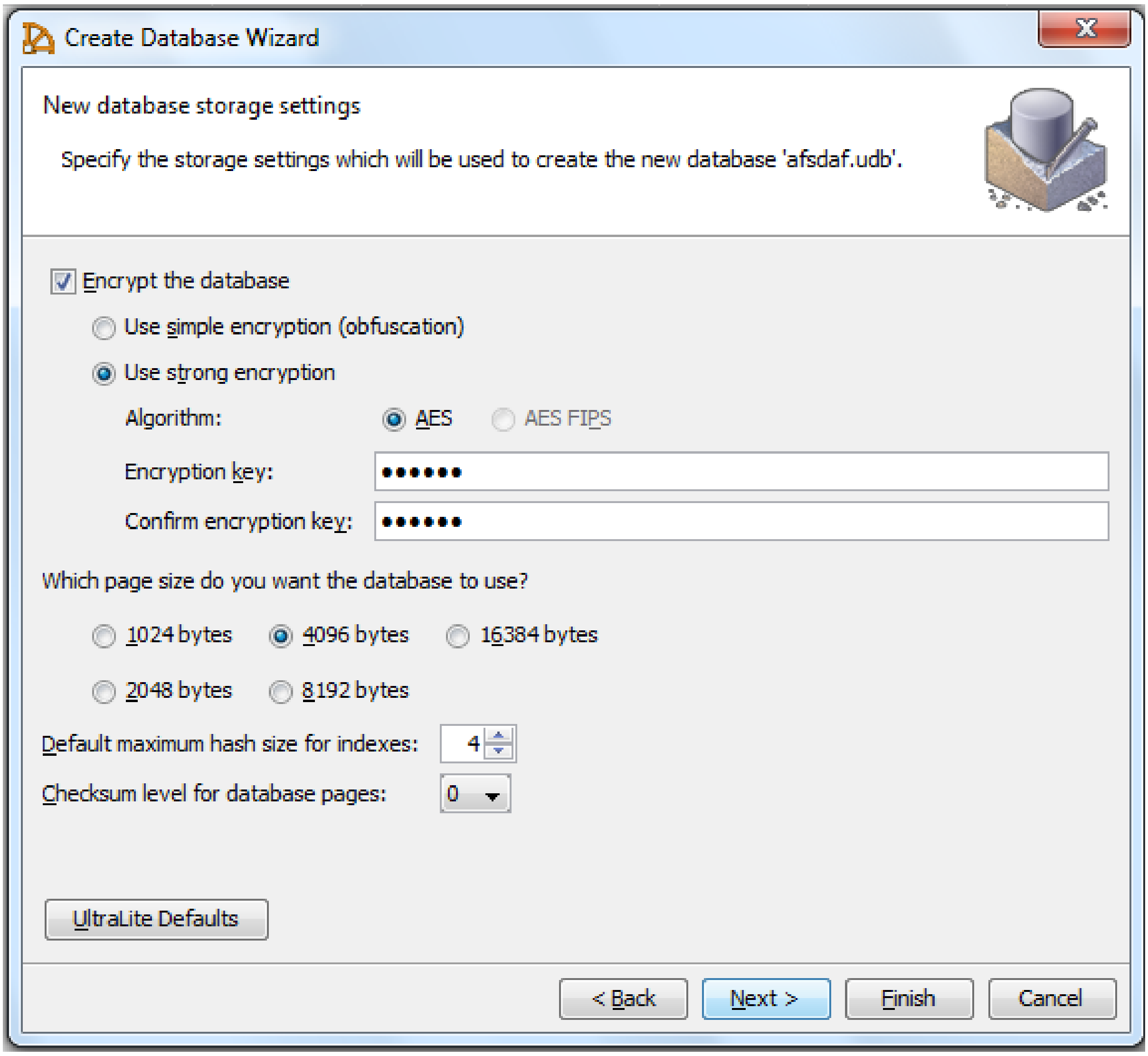
```
execute immediate 'vacuum'
```

Note: before executing VACUUM, make sure the transaction has already been committed, otherwise VACUUM will not be executed successfully.

3.4 See also: Encrypt UltraLite database

The UltraLite database file can be encrypted. You can choose to use a simple or strong encryption for the database file when you create the database in Sybase Central, as shown in the figure below.

Figure 3.1: Encryption type in Sybase Central



Create Database Wizard

New database storage settings

Specify the storage settings which will be used to create the new database 'afsdaf.udb'.

☑ Encrypt the database

○ Use simple encryption (obfuscation)

● Use strong encryption

Algorithm: ● AES ○ AES FIPS

Encryption key: ••••••

Confirm encryption key: ••••••

Which page size do you want the database to use?

○ 1024 bytes ● 4096 bytes ○ 16384 bytes

○ 2048 bytes ○ 8192 bytes

Default maximum hash size for indexes: 4

Checksum level for database pages: 0

Under **Encrypt the database**, either select the **Use simple encryption (obfucscation)** option or the **Use strong encryption** option.

- **Use simple encryption (obfucscation)**: Only fuzzily encrypts the database files and no encryption keys are required in the application scripts.
- **Use strong encryption**: If you have selected this option, configure the **Algorithm** and the **Encryption key** (specified in the application scripts) accordingly.

  The database file will be encrypted by using the key you input. And you will need to input the same encryption key when you configure the database file in PowerServer Toolkit, so that the encrypted database can be accessed by the mobile application. See step 7 in Task 3: Configure the offline settings.

4 Tutorial 4: Config, Deploy & Run the Offline App

The previous tutorial shows you how to specify the database connection syntax so the app will connect with the local database when running on the mobile device. This tutorial will show you how to configure and deploy the database to the PowerServer (so it can be downloaded to the local mobile device when the app is installed on the mobile device) and how to configure and deploy the app as a pure local/offline application so it runs on the mobile device without needing network connections.

The following instructions are applicable to both the UltraLite database and the SQLite database.

4.1 Task 1: Configure the database type

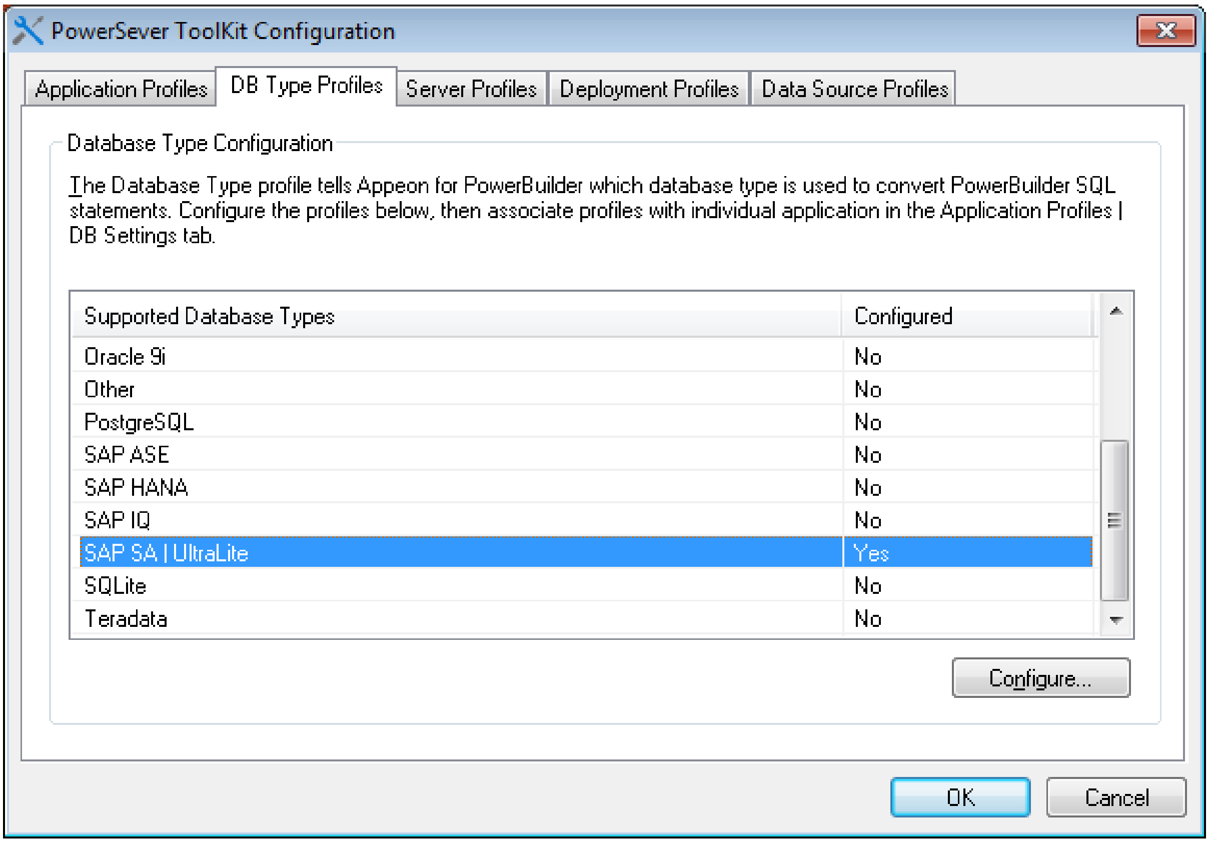
If your mobile application connects with a database, no matter it is a local offline database or a server-side consolidated database, you will need to configure the database type. The database type enables PowerServer Toolkit to apply the correct database driver for generating the correct database syntax. Therefore, the database type is what is important and it is not necessary to specify the actual databases that the application uses, it can be any database so long as they are of the same type.

Step 1: On the PowerServer Toolkit, click the **Configure** icon (🛠).

Step 2: On the **PowerServer Toolkit Configuration** window, click the **DB Type Profile** tab.

Step 3: View the **Configured** column of the database type used by your application, if it displays **Yes**, you can skip the remaining steps in this task. Otherwise, select the database type used by your application, and then click **Configure**.

Figure 4.1: DB Type Profile in ADT



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

Supported Database Types	Configured
Oracle 9i	No
Other	No
PostgreSQL	No
SAP ASE	No
SAP HANA	No
SAP IQ	No
SAP SA | UltraLite	Yes
SQLite	No
Teradata	No

Step 4: On the **Database Type Profile Configuration** window, do the following (take SAP SQL Anywhere as an example):

1. Select the **ODBC Interface** radio button from the **Database Interface** option.

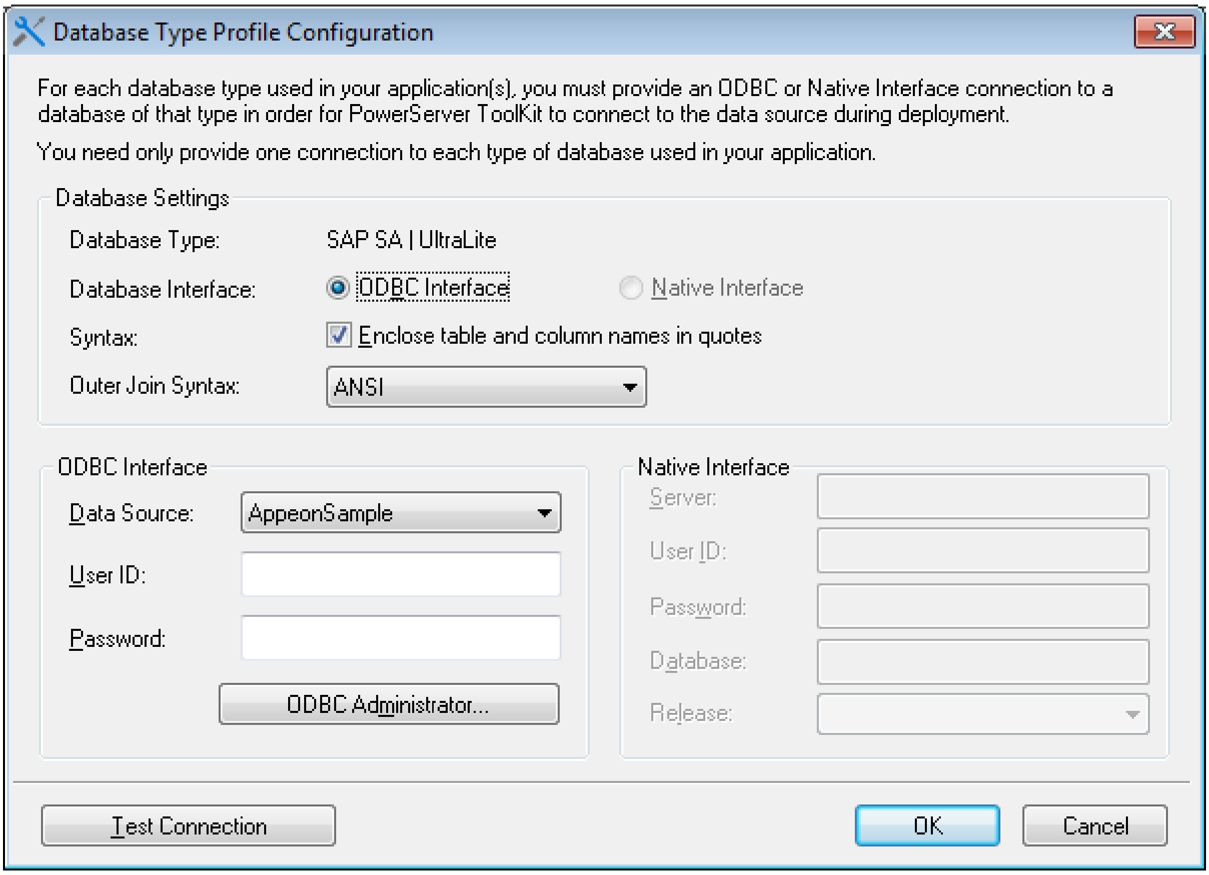
2. Select a data source from the **Data Source** dropdown list box. It can be any data source (ODBC DSN) provided it is the same type as used by your application. It does NOT have to be the actual ODBC DSN used by your particular application. The database type is what is important and that a connection can be established to the specified ODBC DSN.

3. Keep the rest as default, and then click **Test Connection**.

4. Make sure that the database connection is successful.

5. Click **OK**.

Figure 4.2: Database Type Profile Configuration



Database Type Profile Configuration

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

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

Database Settings

Database Type: SAP SA | UltraLite

Database Interface: ● ODBC Interface ○ Native Interface

Syntax: ☑ Enclose table and column names in quotes

Outer Join Syntax: ANSI

ODBC Interface

Data Source: AppeonSample

User ID:

Password:

Now the **Configured** column of your database type will be indicated by **Yes**. You can select the database type when you create the application profile for your application in Task 2: Configure the application.

4.2 Task 2: Configure the application

Step 1: On the PowerServer Toolkit, click the **Config Wizard** ( ) icon.

Step 2: In the welcome screen, click **Next**.

Step 3: Specify the following settings and then click **Next**.

1. Specify the application profile name in the **Application Profile Name** text box. The same text will be displayed in the **Web Folder** text box and the **Mobile App Name** text box. You can modify these fields to use different text.

2. Select **Mobile** or **Universal** from the **Project Type** dropdown list box.

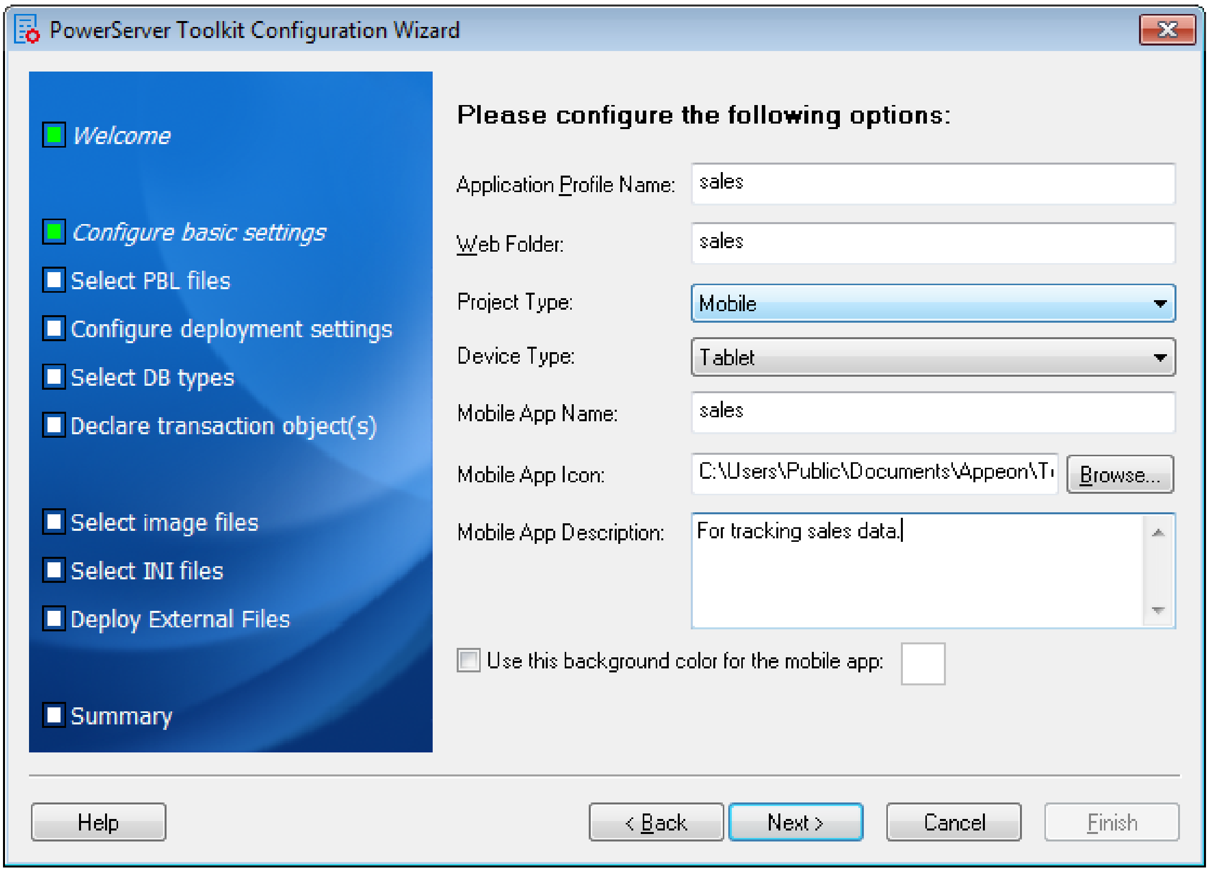
3. Select a device type from the **Device Type** dropdown list box.

   For applications that are designed for tablets, select **Tablet**; for applications that are designed for smartphones, select **Smartphone**; for applications that are designed for both devices, select **Both**.

4. In the **Mobile App Icon** text box, specify an icon for your application by clicking **Browse** to select the image file of the following format: PNG, GIF, JPG, and BMP. The image that you choose will be displayed as the application icon in Appeon Workspace. Recommended size is 86 x 86 pixels (or above) for non-retina display and 172 x 172 pixels (or above) for retina display. To ensure the best display quality in both high-resolution and low-resolution screens, you must supply an image of 172 x 172 or above. The image will be automatically adjusted to fit properly. If you leave this field empty, the default icon will be displayed in Appeon Workspace.

5. Enter a brief description in the **Mobile App Description** text box. This description will be displayed in the Appeon Workspace next to the application icon. If you leave this field empty, no app description will be displayed in Appeon Workspace.

Figure 4.3: PowerServer Toolkit Configuration Wizard



Please configure the following options:

Application Profile Name: sales

Web Folder: sales

Project Type: Mobile

Device Type: Tablet

Mobile App Name: sales

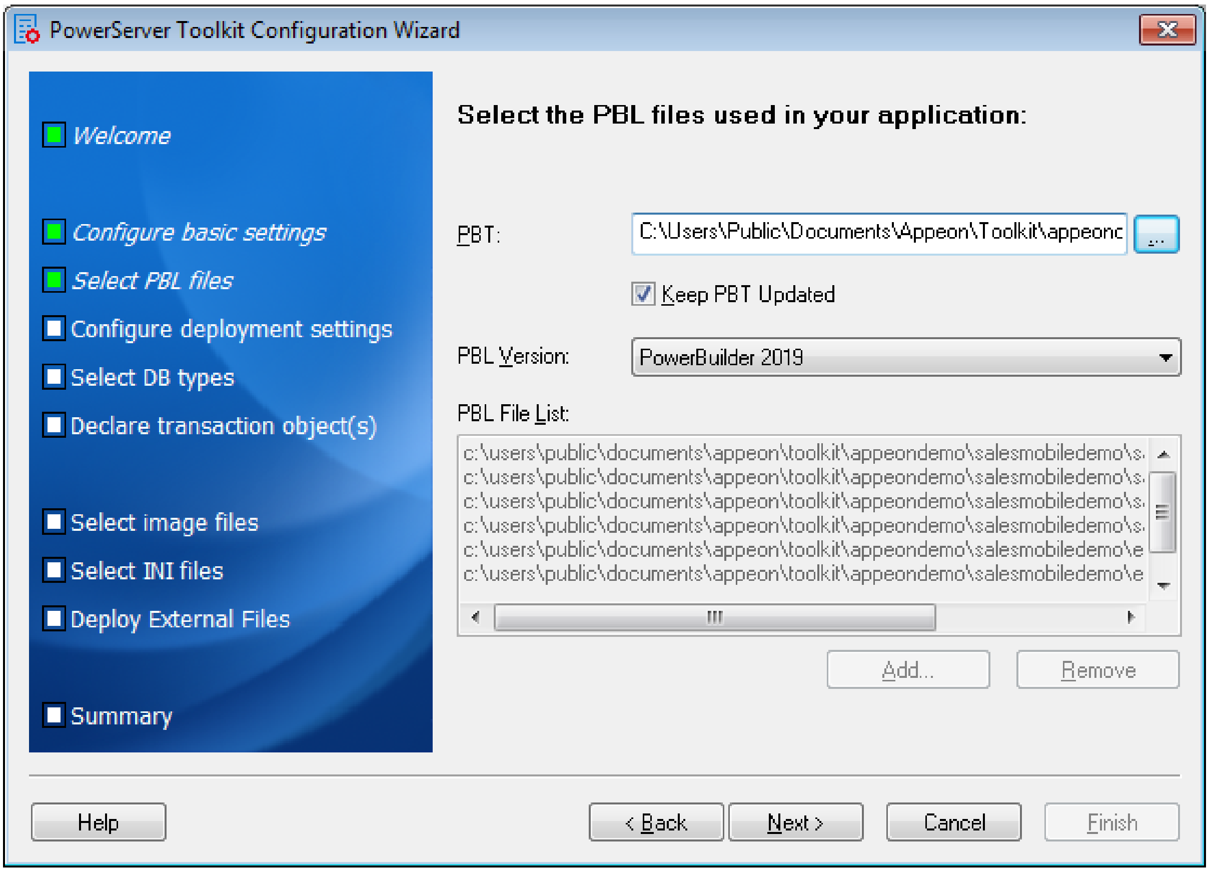
Mobile App Icon: C:\Users\Public\Documents\Appeon\T Browse...

Mobile App Description: For tracking sales data.

□ Use this background color for the mobile app:

Step 4: Add the PBT from the **PBT** dropdown list box (and all related PBLs will be automatically added for you) or add the PBL files individually that are used in your application by clicking the **Add...** button in the **PBL File List** group box, and then select a PBL version from the **PBL Version** dropdown list box. Click **Next**.

Figure 4.4: Add the PBT in ADT Wizard



PowerServer Toolkit Configuration Wizard

Select the PBL files used in your application:

PBT: C:\Users\Public\Documents\Appeon\Toolkit\appeonc

☑ Keep PBT Updated

PBL Version: PowerBuilder 2019

PBL File List:

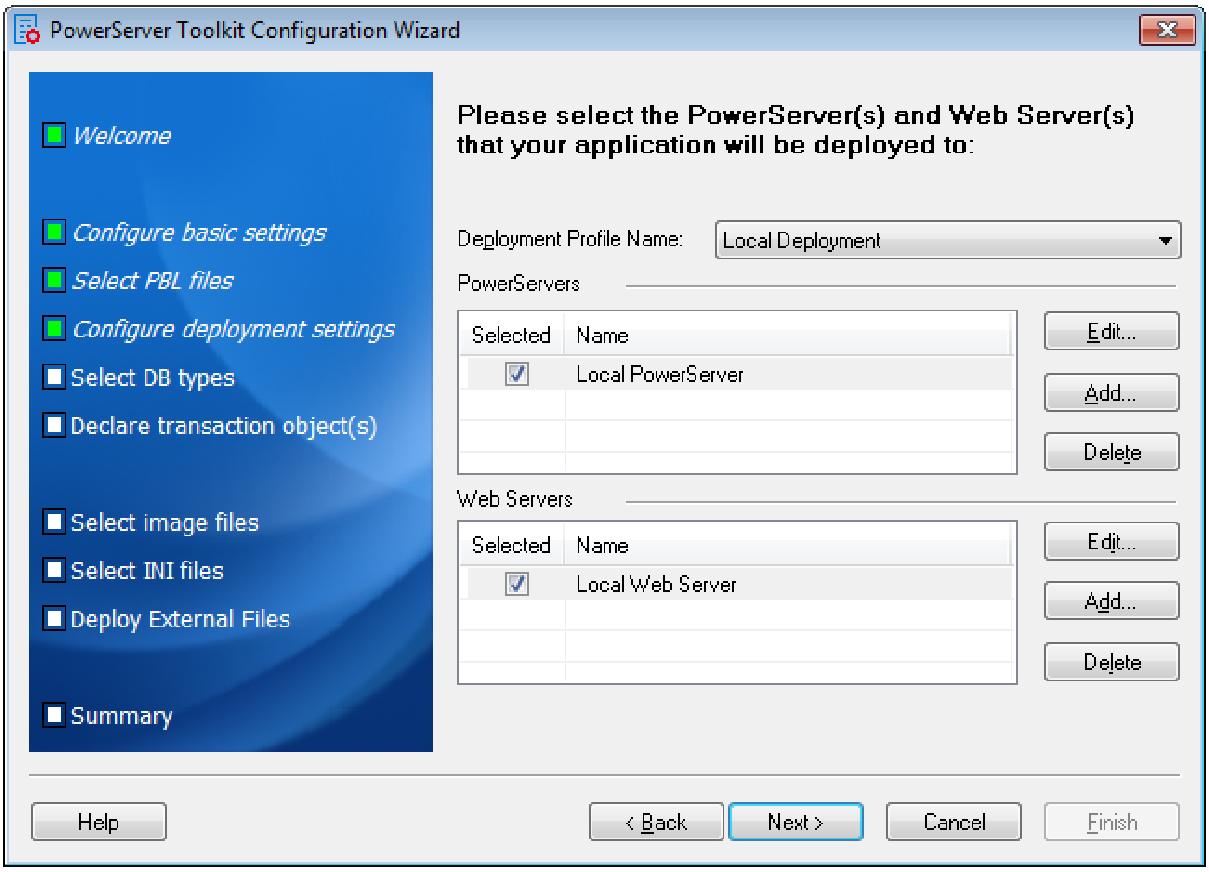
c:\users\public\documents\appeon\toolkit\appeondemo\salesmobiledemo\s:
c:\users\public\documents\appeon\toolkit\appeondemo\salesmobiledemo\s:
c:\users\public\documents\appeon\toolkit\appeondemo\salesmobiledemo\s:
c:\users\public\documents\appeon\toolkit\appeondemo\salesmobiledemo\s:
c:\users\public\documents\appeon\toolkit\appeondemo\salesmobiledemo\e
c:\users\public\documents\appeon\toolkit\appeondemo\salesmobiledemo\e

Step 5: Keep the default settings and click **Next**.

Since you have installed PowerServer Toolkit and PowerServer on the same machine, you can directly use the **Local PowerServer** profile, the **Local Web Server** profile, and the **Local Deployment** profile, all of which are configured automatically by the product setup program.

When you click **Next**, the configuration wizard will automatically test the connection to the server. You will not be able to proceed if the connection test failed.

Figure 4.5: Select the PowerServer(s) and Web Server(s)



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

Deployment Profile Name: Local Deployment

PowerServers

Selected	Name
☑	Local PowerServer

Web Servers

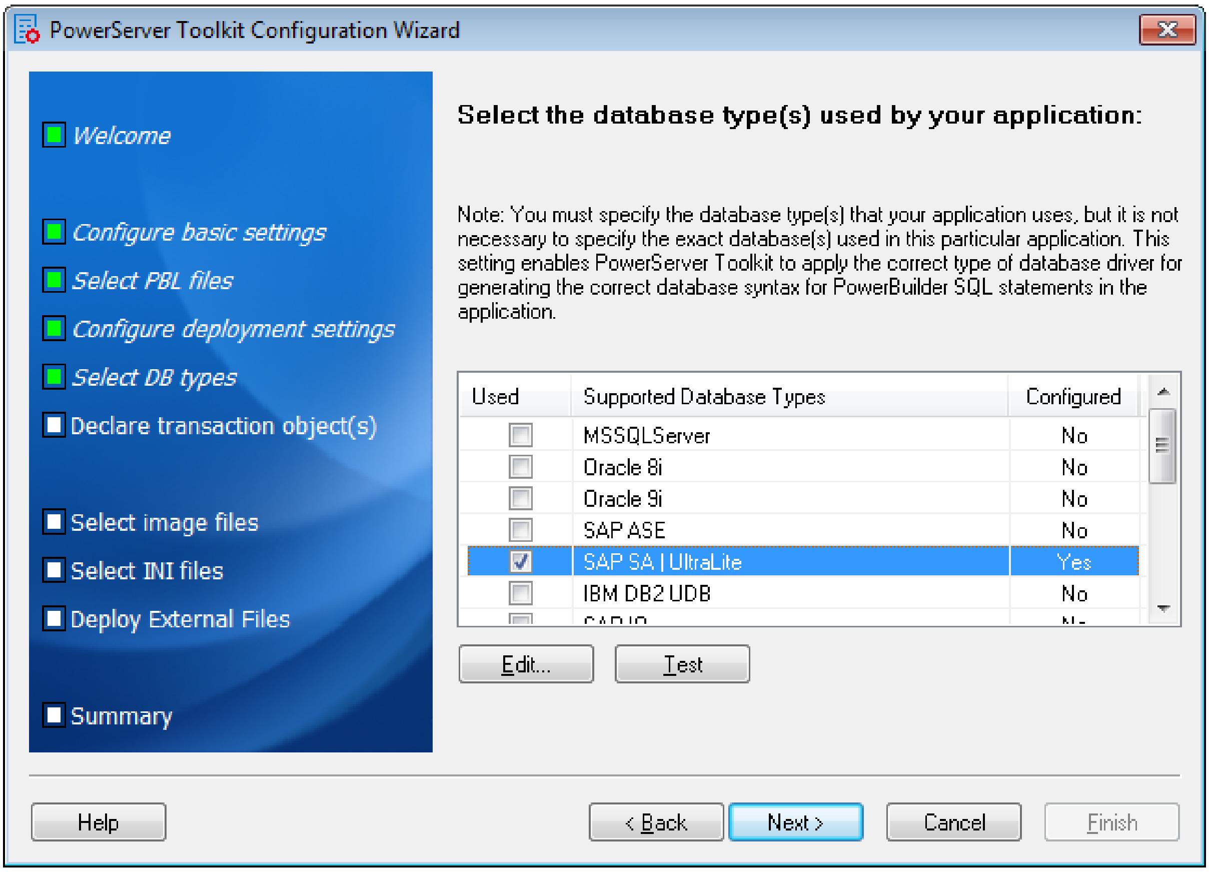
Selected	Name
☑	Local Web Server

Step 6: Select the database type from the **Used** column and click **Next**.

If your mobile application connects with the local offline database only, then select the database type for the local offline database. If your mobile application also connects with the server-side consolidated database, then you will also need to select the database type for your server-side consolidated database.

If the required database type has not been configured in Task 1: Configure the database type, you should highlight the database type, then click **Edit**, and then follow Step 4 in Task 1: Configure the database type to configure it.

Figure 4.6: Select the database type(s)



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

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

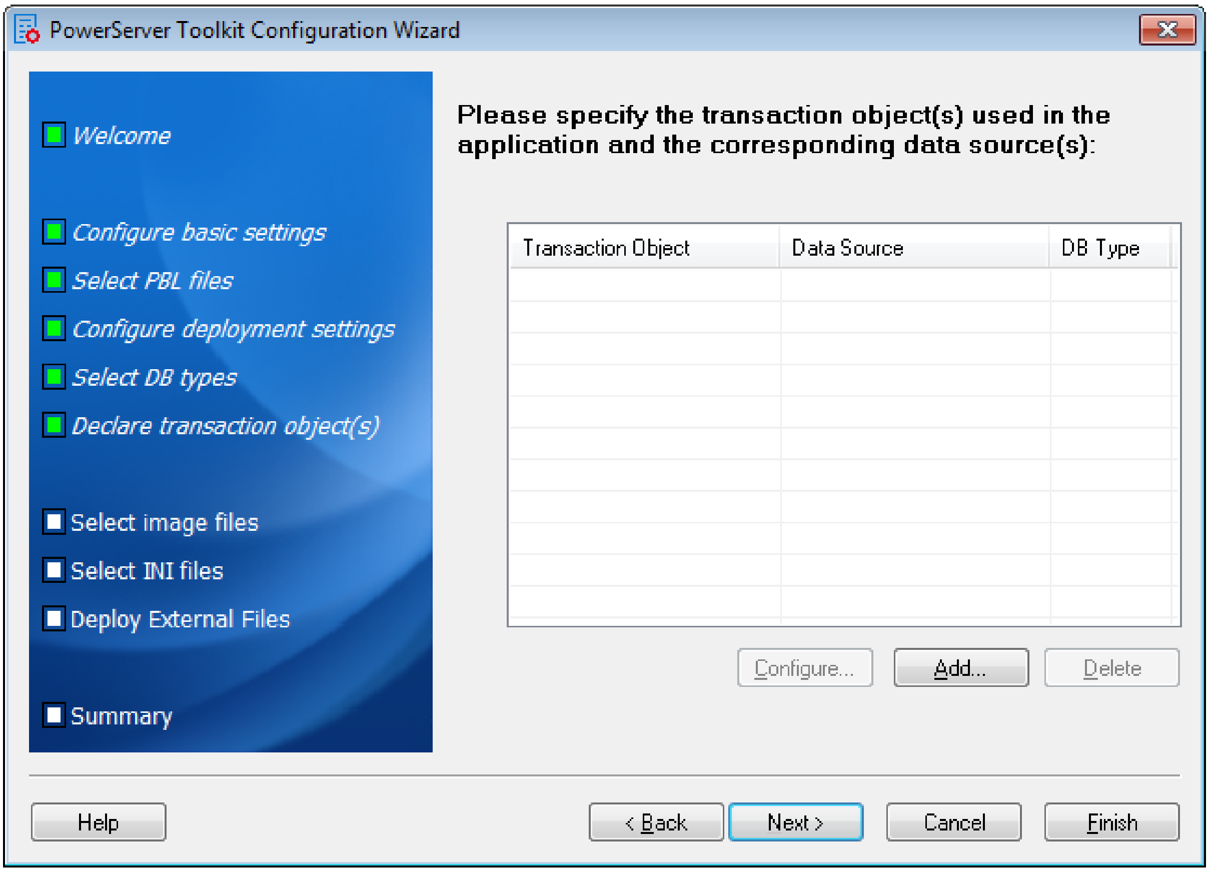
Used	Supported Database Types	Configured
☐	MSSQLServer	No
☐	Oracle 8i	No
☐	Oracle 9i	No
☐	SAP ASE	No
☑	SAP SA | UltraLite	Yes
☐	IBM DB2 UDB	No

Step 7: Click **Add** to specify the transaction object that connects with the server-side consolidated database.

Note: this step is required only if your mobile application connects with the server-side consolidated database. If your mobile application connects with the local offline database only, then skip this step.

For detailed instructions on how to configure the transaction object, please refer to Step 7 in Chapter 2, *Tutorial 2: Config, Deploy & Run the Application* in *PowerServer Mobile Tutorials*.

Figure 4.7: Specify the transaction object



Please specify the transaction object(s) used in the application and the corresponding data source(s):

Transaction Object	Data Source	DB Type

Step 8: Click **Browse** to select the image file or the folder that contains the image files, if any. Click **Next**.

Step 9: Click **Add File** or **Add Directory** to select the INI file, if any. Click **Next**.

Step 10: Click **Add File** or **Add Directory** to select any other external files such as TXT file, PDF file, etc., if any. Click **Next**.

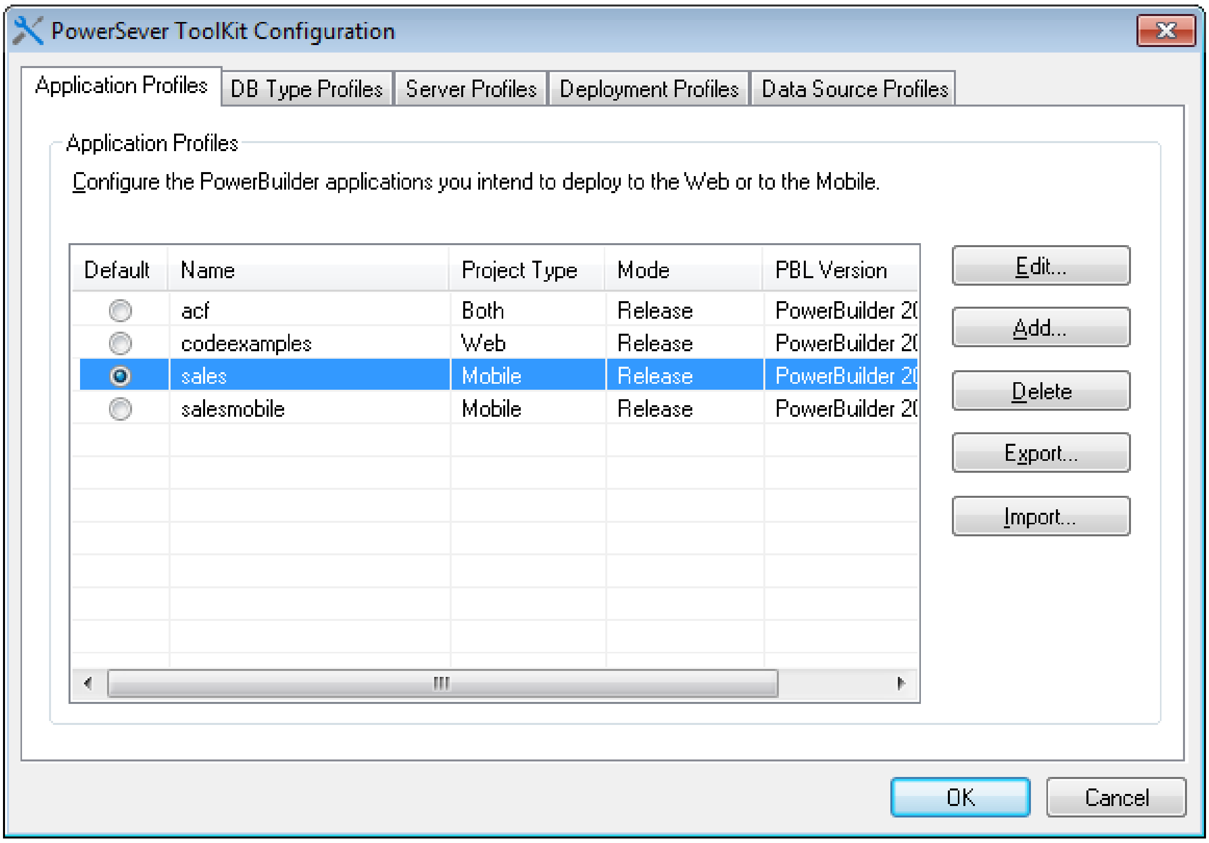
Step 11: In the configuration summary screen, uncheck the **Deploy the application now** option. Click **Finish**.

4.3 Task 3: Configure the offline settings

Step 1: Click the **Configure** ( ) icon on the PowerServer Toolkit. The application you configured just now is listed in the **Application Profiles** tab of the **PowerServer Toolkit Configuration** window.

Step 2: In the **Application Profiles** tab, select the application and then click **Edit**.

Figure 4.8: PowerServer Toolkit Configuration



PowerSever ToolKit Configuration

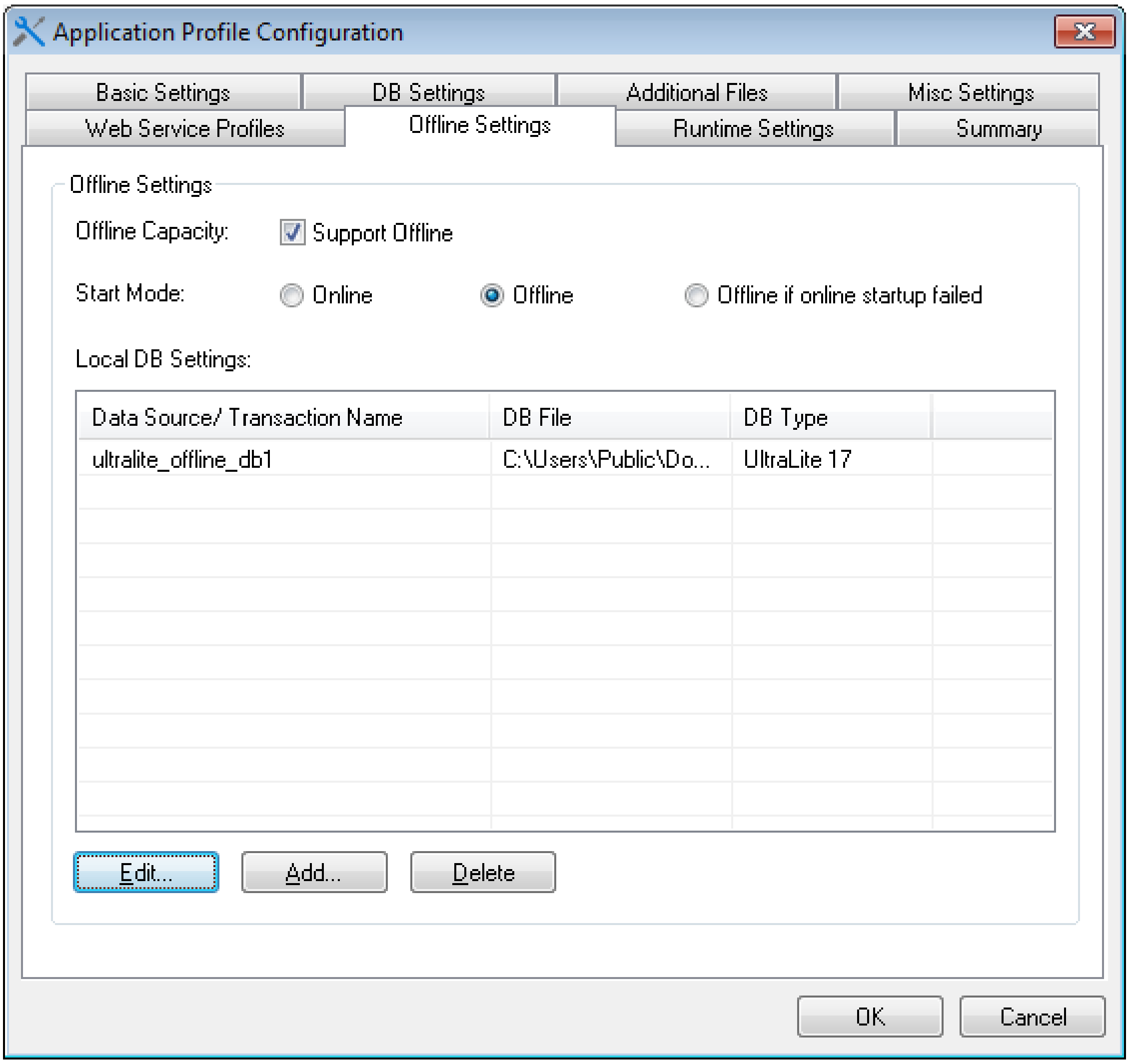
Application Profiles

Configure the PowerBuilder applications you intend to deploy to the Web or to the Mobile.

Default	Name	Project Type	Mode	PBL Version
○	acf	Both	Release	PowerBuilder 2(
○	codeexamples	Web	Release	PowerBuilder 2(
●	sales	Mobile	Release	PowerBuilder 2(
○	salesmobile	Mobile	Release	PowerBuilder 2(

Step 3: In the **Application Profile Configuration** window, select the **Offline Settings** tab.

Figure 4.9: Local DB Settings



Offline Settings

Offline Capacity: ☑ Support Offline

Start Mode: ○ Online ● Offline ○ Offline if online startup failed

Local DB Settings:

Data Source/ Transaction Name	DB File	DB Type
ultralite_offline_db1	C:\Users\Public\Do...	UltraLite 17

Step 4: In the **Offline Capacity** area, select the **Support Offline** option.

Step 5: In the **Start Mode** area, select **Offline if online startup failed** (recommended for the development and test stage), or select **Offline** (recommended for the production stage). Remember to re-deploy the application if the mode is changed.

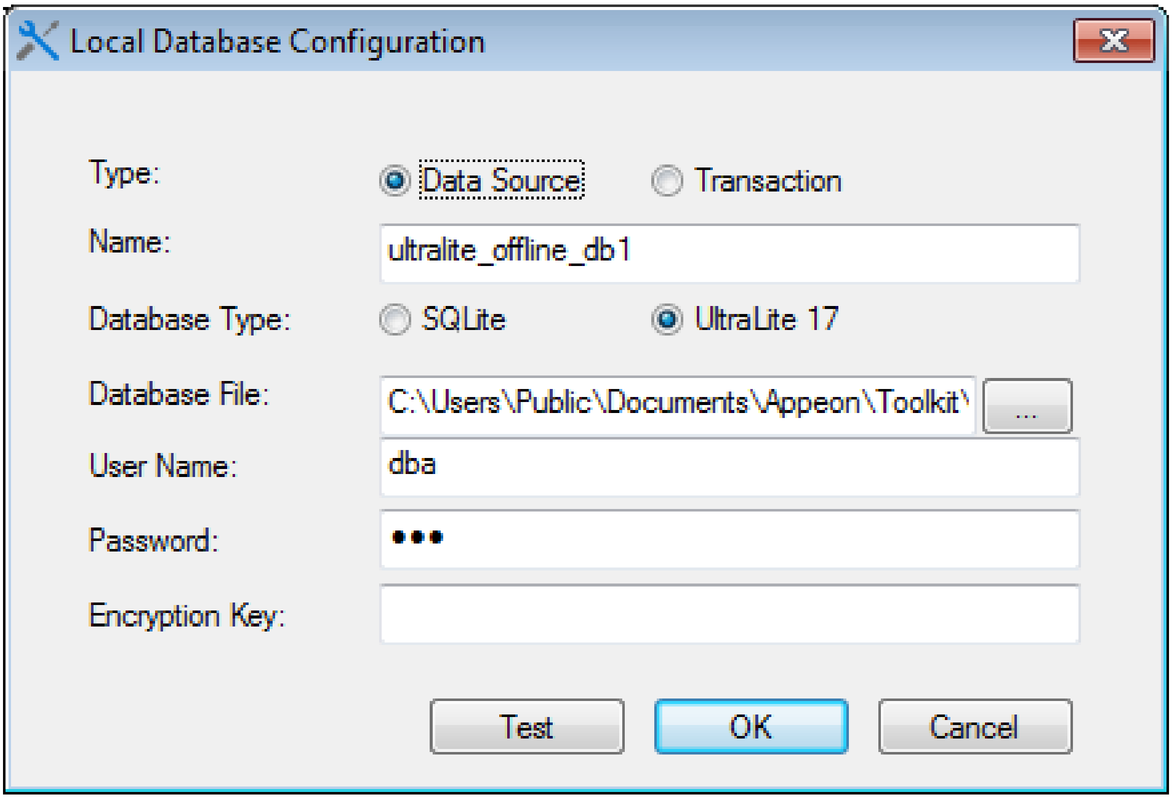
- **Offline if online startup failed** mode allows the application to start as an online application which connects with PowerServer, and if connection with PowerServer failed, the app will start as an offline application. If you are still developing the app and need to test the new functions, this mode is recommended because the app will automatically get the latest changes from PowerServer once it connects with PowerServer successfully during startup.

- **Offline** mode allows the application to start as an offline application which runs locally on the mobile device without connecting with PowerServer, except for the very first time after installed when the configuration information needs to be downloaded to the mobile device. After that, the application will run as a local/offline application without needing any network connections, and will not get updates from PowerServer unless the app calls the relevant APIs specially provided by PowerServer Mobile (for details, see

Tutorial 5: Update the Offline App) or the app is removed from the mobile device and then downloaded and installed again.

Step 6: In the **Local DB Settings** area, click **Add** to specify the local database settings.

Figure 4.10: Local Database Configuration



Local Database Configuration

Type: ● Data Source ○ Transaction

Name: ultralite_offline_db1

Database Type: ○ SQLite ● UltraLite 17

Database File: C:\Users\Public\Documents\Appeon\Toolkit\

User Name: dba

Password: •••

Encryption Key:

Step 7: In the **Local Database Configuration** window, do the following:

1. Select **Data Source** or **Transaction** as the connection type. If you intend to use the transaction object to connect with the local offline database, select **Transaction** here, if you intend to use the **CacheName** DBParm parameter to connect with the local offline database, select **Data Source**. For more details on the connection type, see Step 2 in Connect with the local database in Tutorial 3.

2. Enter the transaction object name or data source name in the **Name** text box. To configure for the **Transaction** type, enter the actual name of the transaction object used in the application. To configure for the **Data Source** type, enter the same name of the **CacheName** DBParm parameter specified in the application.

3. Select **SQLite** or **UltraLite 17** (according to the offline database the application uses) as the database type.

4. Click the browse (**...**) button to select the database file from the **Database File** field. The database file name cannot be "appeondb.db".

5. Enter the user name for the database in the **User Name** text box. This field is effective for UltraLite database only.

6. Enter the password for the database in the **Password** text box. This field is effective for UltraLite database only.

7. Enter the encryption key for the database in the **Encryption Key** text box. This field is effective for UltraLite database only. The encryption key must be the same one you input in the Sybase Central when creating the database. See See also: Encrypt UltraLite database.

8. Click **OK**.

Step 8: Click **OK** to close the **Application Profile Configuration** window.

4.4 Task 4: Deploy the application and the database

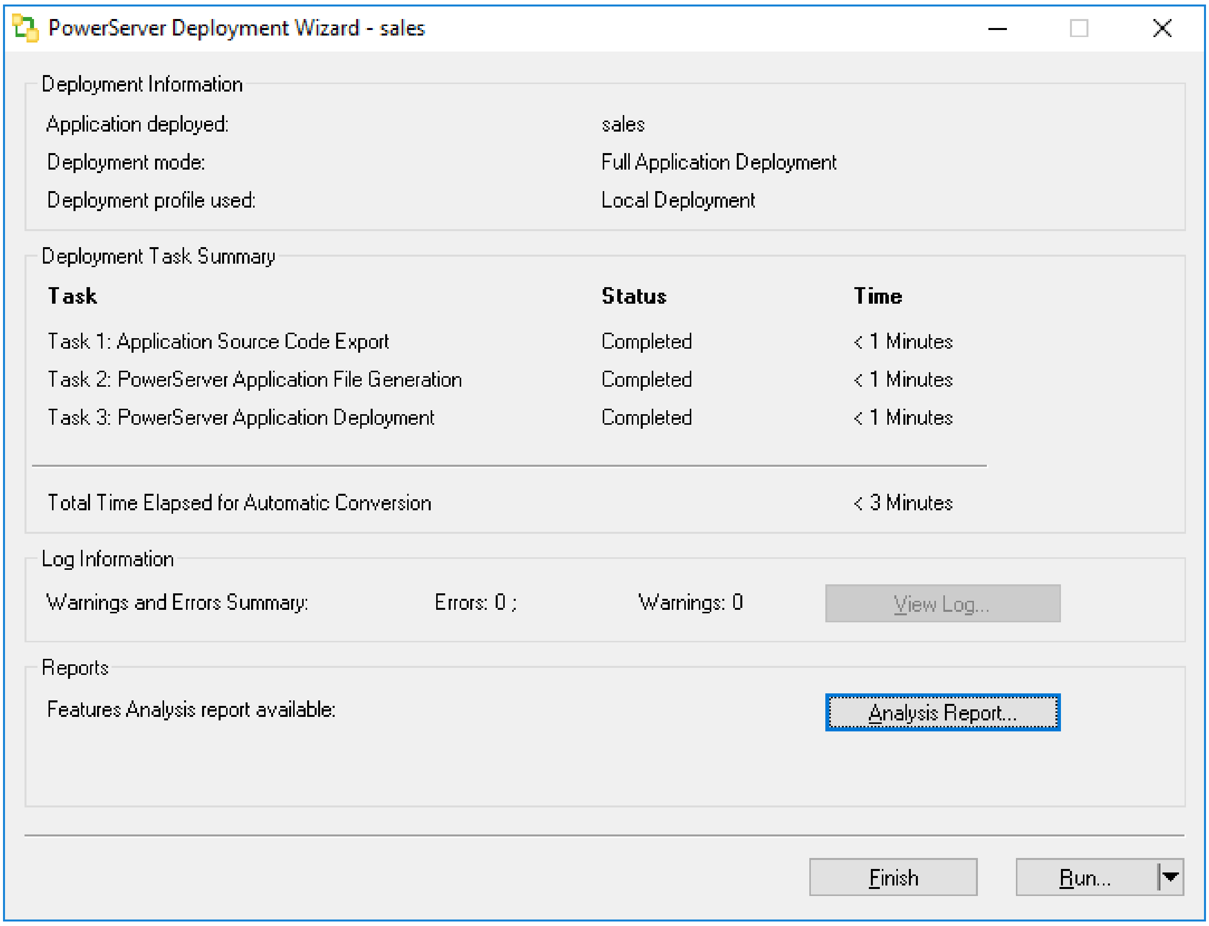
Step 1: Click the **Deploy** icon ( ) on the PowerServer Toolkit.

Step 2: In the **PowerServer Application Deployment Wizard**, make sure the application to be deployed is correct, the deployment profile is **Local Deployment**, and the deployment mode is **Full Application Deployment**. Click **Next**.

Step 3: Click **Finish** when the deployment process is complete.

The application and the database file are deployed to the server successfully.

Figure 4.11: PowerServer Application Deployment Wizard



PowerServer Deployment Wizard - sales

Deployment Information

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

Deployment Task Summary

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

Log Information

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

Reports

Features Analysis report available:

4.5 Task 5: Run the application

Now that you have deployed the application and the database successfully, you are ready to run your offline application with the offline database on the Android device.

Step 1: Configure the network connection. See Task 3: Configure the network connection in Tutorial 1: Set up the Environment.

Network connection is required for installing Appeon Workspace and the offline application, and when running the offline application for the first time after installed (to download the configuration information).

Step 2: Install Appeon Workspace. See Task 4: Install Appeon Workspace in Tutorial 1: Set up the Environment.

Step 3: Tap the **AppeonMobile** icon on your Android device to launch Appeon Workspace.

Step 4: Tap the **New** icon (⊕) to the left of the title bar.

Step 5: In the **App URL** text box, enter the application URL in this format: *http://server_domain/app_name*. For example, if your IIS domain is www.abc.com and you specified "myApplication" in the PowerServer Toolkit configuration as the **Web Folder** name then the URL would be http://www.abc.com/myApplication/.

Step 6: Tap the **Test Connection** button to test the server connections. If successful please proceed to Step 7, otherwise please enter the correct URL.

Step 7: Tap the **Back** icon (⬅) on the title bar to save the information and return to the main screen of the Appeon Workspace.

Once you return to the main screen of the Appeon Workspace, the downloading and installation process of the application the the database occurs automatically.

Step 8: After the application and the database is downloaded, tap the application icon on the main screen to download the configuration information to the mobile device. Make sure there is network connection during the download.

After the configuration information is downloaded to the mobile device, you can run the offline application without network connection.

5 Tutorial 5: Update the Offline App

If you have set the **Start Mode** to **Offline** and deploy the application in PowerServer Toolkit, the application will run as a local/offline app on the mobile device without needing any network connections, except for the first time when the configuration information needs to be downloaded from the PowerServer to the mobile device. In this case, the local/offline application will not be automatically updated, and you will need to call the APIs PowerServer provides to upgrade the application.

PowerServer Mobile provides APIs to check and apply the updates for the local/offline application. For details on how to call these APIs, refer to the section called “of_checkupdate” in *Workarounds & APIs Guide* and the section called “of_applyupdate” in *Workarounds & APIs Guide*.

After getting the updates from PowerServer, you will need to restart the application to make the updates effective.

6 Tutorial 6: Debug the Offline App

The guidelines for developing and debugging the PowerServer mobile applications should also be followed when developing and debugging the offline mobile application. For detailed instructions, refer to Chapter 3, *Tutorial 3: Develop & Debug with PowerServer Mobile* in *PowerServer Mobile Tutorials*.

It is worth mentioning that PowerServer Mobile provides an API to generate the log files on the mobile device. The log files contain detailed information for debugging the app when the log level is set to debug. For how to call the API, refer to the section called “of_log” in *Workarounds & APIs Guide*. After the log file is generated on the mobile device, you can view or send the log file in Appeon Workspace. See Section 4.9, “Configuring logs” in *Appeon Workspace User Guide*.

Index