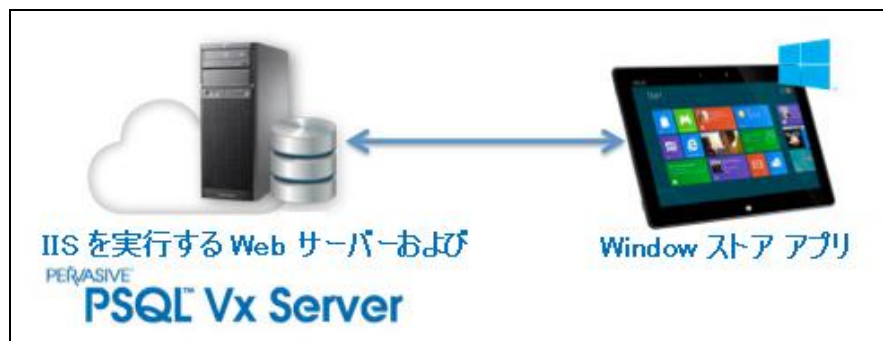


Windows ストア アプリから Pervasive PSQL データへアクセスする

Windows 8 で提供されるさまざまな機能の 1 つに Windows ストア アプリがあります。Pervasive PSQL 自体は Windows ストア アプリではありませんが、この Windows ストア アプリ用のデータ リポジトリとして Pervasive PSQL を使用することができます。Windows ストア アプリを実行しているデバイスに Pervasive PSQL コンポーネントは必要ありません。

ここでは、Windows ストア アプリから Pervasive PSQL データベースのデータへアクセスするための手順について概説します。私のソリューションの目的は、Windows ストア アプリからのデータ アクセスを完全に調査することではありません。とはいえ、この手順の過程で興味深い点がいくつか見つかったので、それらを情報としてお伝えします。

私のソリューションに含めたものは一般に入手可能なソフトウェア コンポーネントで、特別な要素はなにも必要ありませんでした。ローカルにインストールされている Pervasive PSQL サンプル データベース DEMODATA からデータを取得するよう構成された IIS (Internet Information Services) 8 サーバーを使用しました。下図で概略的に示すように、Windows ストア アプリはこの IIS 8 サーバーと通信してデータを取得します。



このように動作する構成になるよう、まずはサーバー環境をセットアップする必要がありました。この環境とは IIS 下で実行する WCF (Windows Communication Foundation) サービス アプリケーションです。これは OData プロデューサーとして機能します。IIS 下で実行する WCF サービス アプリケーションを作成する手順については、[「WCF と IIS を使用して OData プロデューサーを作成する」](#)ドキュメントを参照してください。

上記の事前環境が整ったら、次に (OData コンシューマーとして機能する) Windows ストア アプリを作成します。

ここでは、以下の項目について説明します。

- [環境を設定する](#)
- [Windows ストア アプリを作成する](#)
- [Windows ストア アプリをコンパイルして実行する](#)
- [内部的な動作について](#)
- [最後に](#)

環境を設定する

使用した設定は次のとおりです。サンプル プロジェクト ファイル [ODataWinStoreApp.zip](#) をダウンロードすれば作業にも役立ちます。

- WCF サービス アプリケーション/OData 環境。「[WCF と IIS を使用して OData プロデューサーを作成する](#)」ドキュメントを参照してください。Windows ストア アプリから Pervasive PSQL データにアクセスする前に、同様の環境が機能していなければなりません。そのドキュメントで提供したサンプル プロジェクト ファイルを使用することもできます。
- Windows ストア アプリを作成および実行する開発マシン。これは、Visual Studio 2012 がインストールされた Windows 8 マシンです。Visual Studio は Windows 8 マシンにインストールされている必要があります。そうでない場合、Windows ストア プロジェクト タイプが Visual Studio で使用可能になりません。

Windows ストア アプリを作成する

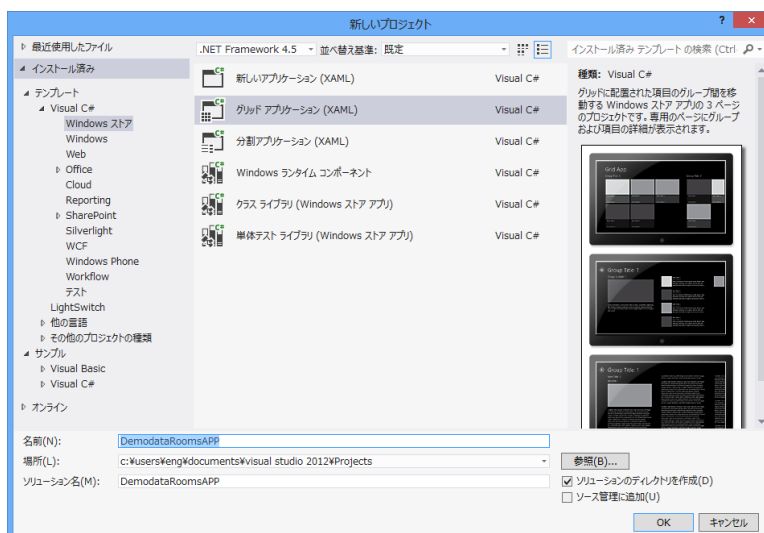
Windows ストア アプリを作成するために使用した手順は次のとおりです。このアプリでは、DEMODATA サンプルデータベースで表した架空の大学におけるビル(棟)と部屋(教室など)を一覧表示します。最初の 5 つの手順ではプロジェクトを設定しています。手順 6 では DEMODATA からデータを取得するためのコードを示します。

私はアプリケーション レイアウトを変更しているため、ご自身で作成されたアプリケーションの外観とは異なる可能性があります。

1. Visual Studio 2012 用の [Windows ストア アプリ用 WCF データ サービス\(英語サイト\)](#) 拡張機能を必ずインストールしておいてください。これは WCF サービス アプリケーションからエンティティ定義を取得するために必要です。
2. スタート ページから Visual Studio 2012 を起動し、[新しいプロジェクト...]を選択します。
3. [テンプレート]カテゴリから、[Visual C#] > [Windows ストア] > [グリッド アプリケーション (XAML)]の順に選択します。

Windows 8 以外のプラットフォームに Visual Studio をインストールした場合は、テンプレート カテゴリに "Windows ストア" は表示されません。

4. プロジェクトとアプリケーションに名前を付けます。私の場合、**DemodataRoomsApp** という名前を付けました。



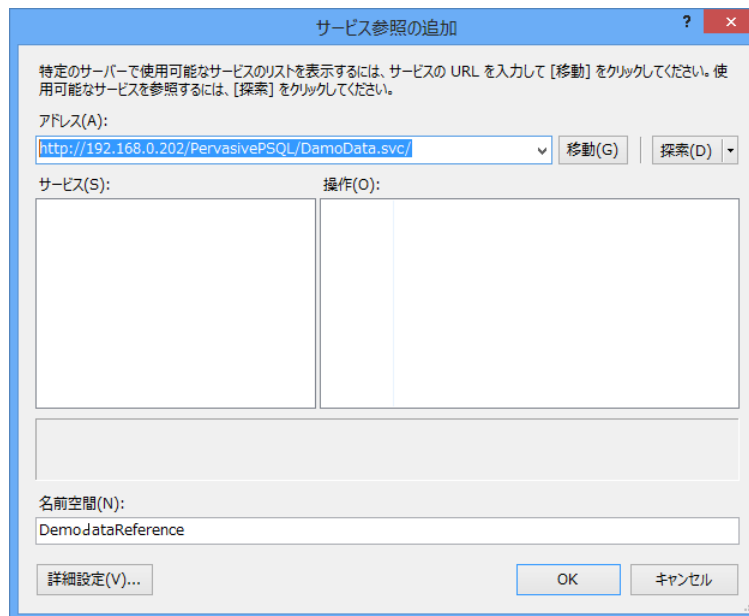
5. WCF サービス アプリケーションを参照するサービスを作成します (WCF サービス アプリケーションが実行していない場合は、すぐに配置して実行してください)。

- a) メニュー バーの[プロジェクト]メニューから[サービス参照の追加]を選択します。
- b) サービスの URL を入力します。「[WCF と IIS を使用して OData プロデューサーを作成する](#)」ドキュメントのプロジェクトを使用する場合は、次のような書式にしてください。

`http://< ホスト名 または IP アドレス>/PervasivePSQL/DemoData.svc/`

- c) [名前空間]には、意味がわかりやすい名前を指定します。私の場合は、**DemodataReference** としました。

画面は次のようになります。



6. 新しいソース ファイルを作成します (**PSQLGetDemoData.cs** という名前を付けました)。このファイルには IIS サーバーから部屋に関する情報を取得するコードが含まれています。

PSQLDemodataModel のコンストラクタには IIS サーバーへの接続に使用する URL が含まれています。

```
this.Context = new DemoDataEntities(new  
Uri("http://192.168.126.130/PervasivePSQL/DemoData.svc/"));
```

上記の URL のホスト名/IP アドレスの部分は、ご自分の IIS サーバーに合わせて変更してください。

GetBuildingNames 関数は room テーブルの全レコードに対してクエリを実行します。私の場合、次のようなコードを使用してクエリを実行しました。

```
var query = (DataServiceQuery<Room>) this.Context.Rooms.Select(room => room);
```

GetRoomsInBuilding 関数は、ビルごとの部屋の一覧についてサーバーにクエリを実行します。

```
var query = (DataServiceQuery<Room>) this.Context.Rooms.Select(room =>
room).Where(room => room.Building_Name == BuildingName);
```

アプリケーションの必要なところでこれらの関数呼び出しを追加しました。また、エンティティフレームワークを使用してレコードをクエリすることがとても簡単だということもわかりました。

Windows ストア アプリをコンパイルして実行する

DemodataRoomsApp.sln という Visual Studio 2012 ソリューションを使用してアプリケーションのビルドおよびデバッグを行いました。IP アドレスはご自分の環境のローカル サーバーの IP アドレスに変更してください。そうしないと、コンパイル エラーで変更を促されます。適切なサーバーを指すよう URL を変更したら、**#error** コンパイラ ディレクティブをコメント行にすることができます。

1 つ注意が必要なのは、**Web** サーバーは時間内にデータを返さないこともあるので、データがアプリケーションで常に表示されるわけではないということです。通常、データはアプリケーションを再実行した後に表示されます。

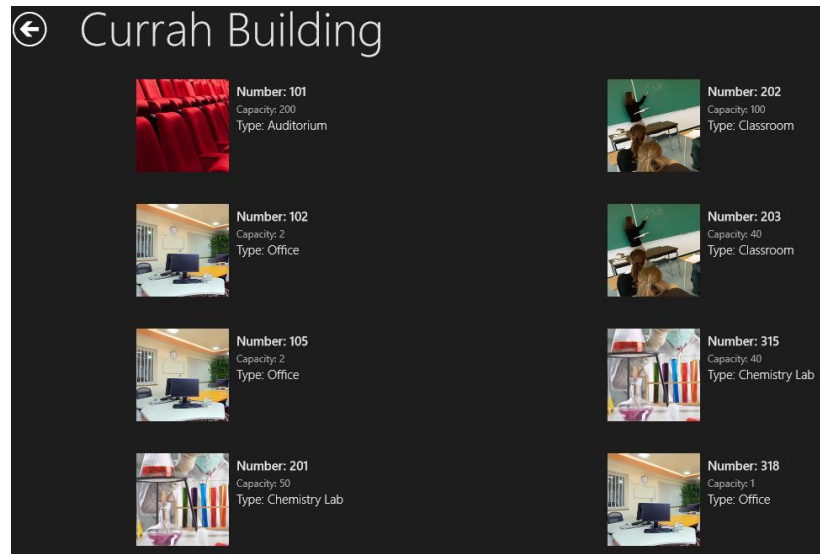
アプリケーションを実行すると、データベース内のすべてのビルを一覧表示する画面が次のように表示されます。



The screenshot shows the application interface for 'DemodataRoomsApp'. It features a 'Buildings >' header and a grid of 24 building entries, arranged in 4 rows and 6 columns. Each entry consists of a building name in bold and a truncated version of the same name below it.

DemodataRoomsApp					
Buildings >					
Roscart Building	Garrison Hall	Kimball Building	Morales Center	Bhargava Building	Currah Building
Roscart Building	Garrison Hall	Kimball Building	Morales Center	Bhargava...	Currah Building
Vander Stoep Hall	Gimlett Building	Klinetob Building	Ojeda Building	Billingsley Building	Eldridge Building
Vander Stoep...	Gimlett Building	Klinetob Building	Ojeda Building	Billingsley...	Eldridge Building
Woodward Building	Harrison-Harper Hall	Martin Center	Quaglino Quarter	Boerner Building	Faske Building
Woodward...	Harrison-Harper...	Martin Center	Quaglino Quarter	Boerner Building	Faske Building
Young Building	Holland Hall	McAnally Building	Bartold Building	Budinszky Building	Gambill Building
Young Building	Holland Hall	McAnally...	Bartold Building	Budinszky...	Gambill Building

"Currah Building" 欄をクリックすると、このビルの子部屋の一覧と各部屋の説明の一部が表示されます。



(Windows ストア アプリを計画するにあたり、さらに情報が必要な場合は、<http://msdn.microsoft.com/en-us/library/windows/apps/hh465427.aspx> を参照してください。)

内部的な動作について

WCF サービス アプリケーションと Windows ストア アプリ間で使用されるプロトコルは OData (Open Data Protocol) です。OData は主に Microsoft によって開発されたオープン スタンドで、ほかのプラットフォーム プロバイダーにも採用されています。Windows プラットフォームの場合、Microsoft はこのプロトコルの詳細を隠すために Visual Studio で上手く抽象概念を作成しています。アプリケーション プログラマーにとっては、ADO.NET に基づいたほかのデータ サービスを使用することと同じです。OData の仕様の詳細については、[OData Web サイト](#)をご覧ください。

興味深いことに、OData クラウドクライアントは非 Microsoft ベースのプラットフォームにも存在します。理論上、Pervasive PSQL クラウドクライアントをサポートしないプラットフォームから Pervasive PSQL データベースと相互に作用するアプリケーションを作成できるということです。Java および JavaScript 向けの OData コンシューマーを作成する方法については、別途提供しているドキュメントを参照してください。

最後に

このドキュメントでは、Windows ストア アプリのデータ リポジトリとして Pervasive PSQL データベースを使用する方法を示しました。また、Windows ストア アプリを実行するマシン (ノート PC、タブレットあるいはスマートフォン) には Pervasive PSQL コンポーネントが必要ないことも示しました。

C# および .NET プログラミングの知識があたりなら、このスタイルのアプリケーションは非常に身近に感じられるでしょう。PSQL へのアクセスは、ほかのデータベース アクセスと同様のプロセスで簡単に行えます。必要なのは、Visual Studio、OData および .NET などの業界標準コンポーネントだけです。

私と同じように C# および .NET プログラミングについて詳しい知識を持たれていないお客様にとって、ここで提供した情報が役立てば幸いです。このドキュメントを参考に、Windows ストア アプリを作成し、Pervasive PSQL データベースをデータリポジトリとして使用できるようにしてください。

Pervasive Software ソフトウェア開発技術者 Jan D.