Contents

[SharePoint 2013: Support and convert SharePoint field types for Windows Phone apps 2](#_Toc329691155)

[Description of the sample 2](#_Toc329691156)

[Prerequisites 2](#_Toc329691157)

[Key components of the sample 2](#_Toc329691158)

[Configure the sample 3](#_Toc329691159)

[Run and test the sample 3](#_Toc329691160)

[Troubleshooting 3](#_Toc329691161)

[Change log 4](#_Toc329691162)

[Related content 4](#_Toc329691163)

# SharePoint 2013: Support and convert SharePoint field types for Windows Phone apps

This documentation is preliminary and is subject to change.

Summary:  Implement data-conversion logic to support SharePoint field types in Windows Phone apps.

In projects based on the Windows Phone SharePoint List Application template, the data of many SharePoint field types is processed and coordinated by default conversion logic for display and manipulation in the Silverlight user interface of a Windows Phone. Developers can also implement their own custom data handling routines. Even if you don't provide conversion logic in your app for Text fields, such fields can still be displayed and edited. Providing your own conversion logic can give you greater control over the format or the structure of the data in those fields. For fields with certain other data types, such as Guid fields, conversion logic is not provided by default. If you understand the mechanism by which conversion logic is provided for fields, it may be fairly simple to provide your own conversion logic..

## Description of the sample

Assume you are creating a Windows Phone app based on a SharePoint list named Product Identifiers, which includes a Product (or Title) field (of type Text) and an Identifier field (of type Guid).

You can create an additional instance of the ListDataProvider class in the project to serve as the data source for the second list (Engineering Team) in the App.xaml.cs file.

Note



SharePoint lists with **Guid** fields must be created programmatically or from a list template that includes **Guid** fields.

In a Windows Phone app created by using the template and based on this simple list, the data in Guid fields is not displayed by default. (In place of that data, a message such as the following is displayed: "No Converter for field type 'Guid' is registered.") In this sample code, you include conversion logic to support Guid fields for a Windows Phone app. You add a class that contains methods to register field value converters to display GUIDs and to generate new GUID values for added list items.

## Prerequisites

This sample requires the following:

* Visual Studio 2010 Express with the new SharePoint templates
* An installation of SharePoint 2013 Preview with administrative privileges

## Key components of the sample

The sample contains the following:

* SPListAppGuidConversion project, which contains the App.xaml.cs file and the CreateProductIdentifiersList.ps1 Windows PowerShell script.
* App.xaml.cs   This file is autogenerated by the Windows Phone SharePoint List Application template. The App.xaml file represents the overall Windows app. The associated code-behind file, App.xaml.cs, includes procedural code to handle life-cycle events for the app. The App.xaml.cs file contains references to SharePoint Server and the list title.
* CreateProductIdentifiersList.ps1   You can run this Windows PowerShell script from the SharePoint Management Shell to create the SharePoint list on which this project is based.

## Configure the sample

Follow these steps to configure the sample.

1. Update the value of TargetSiteUrl in the App.xaml.cs file of the SPListAppGuidConversion solution with the URL of the home page of your SharePoint website.
2. Update the value of the <TargetListTitle> element in the App.xaml.cs file of the SPListAppGuidConversion solution with the title of your target SharePoint list.

## Run and test the sample

* Choose the F5 key to build and run the app.

## Troubleshooting

The following table lists common configuration and environment errors that prevent the sample from running or deploying properly and how to solve them.

| Problem | Solution |
| --- | --- |
| While running the SharePoint List wizard from Visual Studio 2010 Express, an error may occur if developer does not have sufficient privilege on SharePoint site. | Give sufficient privilege to the user account with which developer is running the wizard. |
| Form-based authentication error. | Form-based authentication is not enabled by default. To enable basic form-based authentication for the web application, follow these steps.   * Navigate to Central Administration and ensure you have administrator rights on the server. * Under Application Management, choose Manage Web Applications. * Choose your web application (on which you have your SharePoint site, which you are accessing from your mobile app). * From the ribbon, choose Authentication Providers. * In the Authentication Provider dialog box, choose Default to edit the authentication. * In the Edit Authentication Model window under Claims Authentication Types, choose Basic Authentication. |

## Change log

| Version | Date |
| --- | --- |
| First version | July 16, 2012 |

## Related content

* [SharePoint 2013: Support and convert SharePoint field types for Windows Phone apps](#BM86d6a3f027db435a81bb35ed7382a819)
* **How to: Create a Windows Phone SharePoint 2013 list app**
* **Overview of Windows Phone SharePoint 2013 application templates in Visual Studio**