Contents

[SharePoint 2013: Accessing complex ECT with CSOM 2](#_Toc328126380)

[Description of the Accessing complex external content types with client object model sample 2](#_Toc328126381)

[Prerequisites 2](#_Toc328126382)

[Key components of the sample 2](#_Toc328126383)

[Configure the sample 3](#_Toc328126384)

[Extract the SampleBCSApp.zip files to your hard drive 3](#_Toc328126385)

[Start simulated OData service 3](#_Toc328126386)

[Create a new app in Visual Studio 3](#_Toc328126387)

[Create the external content type 4](#_Toc328126388)

[Get the SharePoint context object to execute the CSOM calls 4](#_Toc328126389)

[Read complex external data using CSOM calls 5](#_Toc328126390)

[Display external data using CSOM and JQuery-UI 6](#_Toc328126391)

[Troubleshooting 7](#_Toc328126392)

[Change log 7](#_Toc328126393)

[Related content 7](#_Toc328126394)

# SharePoint 2013: Accessing complex ECT with CSOM

This documentation is preliminary and is subject to change.

Summary:  Learn how to use client code libraries to manipulate app-scoped external content types.

## Description of the Accessing complex external content types with client object model sample

This project will show how to use Visual Studio 2012 and SharePoint development tools in Visual Studio 2012 to create an app for SharePoint using Business Connectivity Services (BCS) to expose complex data from an external system.

The main objectives for this sample are:

* Set up and use the simulated, self-hosted OData service to provide data that the auto-generation tools in Visual Studio 2012 can use to create external content types
* Create a new app for SharePoint
* Create an external content type that describes data from the self-hosted OData service.
* Use the client object model that has been extended for BCS in SharePoint 2013 to retrieve data by directly calling into the external content type.
* Use JQuery and JQuery-UI to display external data in your app.

This sample will use the Employee data entity found in the self-hosted OData service to 1) display the successful retrieval of the client content from SharePoint, 2) to retrieve the number of records found in the Employee entity, and 3) to display a detailed list of the records found in the Employee entity.

## Prerequisites

This sample requires the following;

* SharePoint 2013 Preview
* Visual Studio 2012
* Internet Information Services (IIS)
* JQuery and JQuery UI. These can be downloaded from <http://jquery.com> and <http:jquery-ui.com>.

## Key components of the sample

The BCSComplexTypeSample.zip file includes the following:

* Visual Studio project files
* Local OData service (CannedDataService)

## Configure the sample

In order to run the samples included in this project, you will need to do the following:

1. Extract the SampleBCSApp.zip file to your hard drive.
2. Start the simulated OData service. This service is hosted by a local instance of IIS. It simply attaches to a port in IIS and provides an OData endpoint that you will use in your app
3. Load Visual Studio project files
4. Build and deploy the project to SharePoint

## Build the project

To build and deploy, press F5.

## Troubleshooting

If you cannot get the “Canned” data service to work, make sure that all the files are in the same folder on your hard drive.

## Change log

First release.

## Related content

* [External content types in SharePoint 2013](http://msdn.microsoft.com/en-us/library/11d7adb5-5388-4517-ae03-beb7be1c6981)
* [Using OData sources with Business Connectivity Services in SharePoint 2013](http://msdn.microsoft.com/en-us/library/7a87e5bf-4428-4055-b113-7665a93e7326)
* [Setting up a SharePoint 2013 development environment for apps](http://msdn.microsoft.com/en-us/library/b0878c12-27c9-4eea-ae3b-7e79e5a8838d)
* <http://www.odata.org>