Office 365 &sql Azure Integration

**An Overview**

**HCL Technologies**

**SharePoint Service Line – Modern Apps**

Contents

[Introduction: 3](#_Toc458695407)

[Technical Approach: 4](#_Toc458695408)

[Overview: 4](#_Toc458695409)

[Architecture Diagram: 5](#_Toc458695410)

[SPO BCS Limitations: 5](#_Toc458695411)

[SPO Other Limitations: 6](#_Toc458695412)

[Integrating Office 365 & SQL Azure: 6](#_Toc458695413)

[Connecting SQL Azure to Office 365 with BCS: 6](#_Toc458695414)

[Other Integration Options: 9](#_Toc458695415)

[Pros and Cons of Cloud (Office 365 & SQL Azure): 11](#_Toc458695416)

[Summary 12](#_Toc458695417)

[Useful links to refer 12](#_Toc458695418)

Figures

[Figure 1: Integration using BCS 5](#_Toc335141072)

[Figure 2: Creating External Content Types 7](#_Toc335141073)

[Figure 3: Select Data Source Type 7](#_Toc335141074)

[Figure 4: Add metadata connections 7](#_Toc335141075)

[Figure 5: Create List & Form 8](#_Toc335141076)

[Figure 6: Cloud Connector -Architecture 9](#_Toc335141077)

# Introduction:

Microsoft’s cloud computing solutions include Windows Azure Platform at the Platform-as-a-Service (PaaS) front and Office 365 at the Software-as-a-Service (SaaS) side.

Windows Azure is a cloud services operating system that serves as the development, service hosting and service management environment for the Windows Azure platform. Windows Azure provides developers with on-demand compute and storage to host, scale, and manage web applications on the internet through Microsoft datacenters.

Windows Azure offers multiple services to help manage your data in the cloud. SQL Database, formerly known as SQL Azure Database, enables organizations to rapidly create, scale and extend applications into the cloud with familiar tools and the power of Microsoft SQL Server technology.

Microsoft Office 365 is the next generation of Microsoft Business Productivity Online Suite (BPOS) based on the Wave 2010 servers and brings together cloud versions of most trusted communications and collaboration products with the latest version of Office desktop suite. Microsoft Office 365 is a way to use the cloud to make business documents available whenever, wherever needed. It is a collection of programs and services for sending email, storing documents, and collaborating online. Office 365 is subscription-based, so you pay as you go.

While Windows Azure and O365 seem to be independent to each other, it is possible to integrate and make them work together. In this article we will find out all the possible ways by which we can connect office 365 with SQL Azure.

# Technical Approach:

## Overview:

Windows Azure (PaaS) and SharePoint Online (SaaS) can be used together to deliver end-to-end cloud solutions. The Windows Azure cloud allows us to develop and deliver custom code, services, complex data (SQL Azure) and connected back-end integration (AppFabric). These complex operations can be connected through Business Connectivity Services (BCS) in SharePoint Online.

|  |
| --- |
| **Points of Integration** |
| **Azure Integration** | **How** |
| SharePoint Client Object Model | Interact with Windows Azure data in a SharePoint External list. Used to create animated view of External data. |
| Business Connectivity Services(BCS) | Model data from Windows Azure or build external list to SQL Azure using WCF  |
| Silver light | Create UI against Windows Azure services or data |
| Open XML | Manage Windows Azure data in a document |
| REST | Use Rest to interact with Windows Azure data to integrate with SharePoint |
| Office Server Services | Combine with Open XML to auto-gen (Such as PDFs) on a server |
| Workflow/Event Receivers | State or events that tie into Windows Azure services, workflow or data. |
| LINQ | Use for querying Window Azure data objects |

Here when integrating with Windows Azure, SharePoint is consumptive and not being hosted. In other words, SharePoint is not a service that is hosted by Windows Azure, but rather an application that consumes Windows Azure data or services. Windows Azure provides applications or resources that will be consumed by SharePoint artifacts such as a Web Part or Silver light application.

## Architecture Diagram:

The following figure displays how we connect office 365 with SQL Azure using BCS



Figure 1: Integration using BCS

SQL Azure data is exposed using WCF service which is consumed by the external content type. Once data comes in External list it can be displayed in SPO using Content Editor web part or client object Model or Silver light.

**Business Connectivity Services (BCS)**:

Enable connections to external data sources via Windows Communication Foundation (WCF) Web Services endpoints —in both read and write modes. These are often line-of-business (LoB) applications that sit behind the firewall, or are transitioning to the Cloud, like SQL Azure. We can create SharePoint external lists and data columns, the BDC service for WCF connectors and the Secure Store Service is partitioned at the tenant level within the SharePoint Online Administration Center.

Note: BCS within SharePoint Online does not support a direct connection to SQL Azure. A WCF endpoint is required

## SPO BCS Limitations:

* Connect to external data sources via WCF Web Services endpoints in both read and write modes.
* The Business Data Catalogue (BDC) service for WCF connectors and the Secure Store Service partitioned at the customer SharePoint Online Administration Center
* Currently, there are some components of BCS that are not supported, including external data search, rich client integration, profile pages, and direct connectivity to SQL Azure without a WCF endpoint.

## SPO Other Limitations:

Sandbox solutions are not supported in SPO: We can’t connect external systems in SPO via BCS from within sandboxed solutions.

Reason: The sandbox stripes out the user’s security token and this means that the credential mapping (e.g. All Users) defined in the Secure Store Service doesn’t work within in the sandbox.

The suggested work around is that the managed account that runs the user code proxy service (SPUCWorkerProcessProxy.exe) is mapped to the external credentials. But this is not possible in SPO

 Ref: <http://msdn.microsoft.com/en-us/library/ff798353.aspx>

# Integrating Office 365 & SQL Azure:

## Connecting SQL Azure to Office 365 with BCS:

Here we will leverage BCS in SP-O using a WCF service.

The procedure to create an external list in SP-O that uses a cloud-based WCF service is as follows:

* Create a back-end LOB data source that we’ll run our service against

We will create a new Database in SQL Azure, then create table & populate it with data.

* Create a WCF service and deploy to Windows Azure

The WCF service we create will be using the Cloud template and be deployed to our Windows Azure account.

When it is deployed, we will be able to click on the service definition to retrieve the WSDL, which will reflect the web methods we included in it.

* Assess the permissions in the Metadata Store

To set the permissions on the Metadata Store, where the ECTs are stored, simply navigate to the Business Data Connectivity option in SP-O portal, and select Set Metadata Store Permissions. Type in the person we want to have permissions for the ECT, and click Add, and then set the explicit permissions. Click OK when done.

* Create an external content type (ECT) in SharePoint Designer that digests WCF service and creates the external list.
1. Navigate to SP-O site and then click Site Actions and then select Edit in SharePoint Designer.
2. Click External Content Types in the left-hand navigation pane.
3. Click the External Content Type in the ribbon. Add a Name and a Display Name and leave the other options defaulted. Click the ‘Click here to discover…’ link.



Figure 2: Creating External Content Types

1. In the External Data Source Type Selection, select WCF Service from the drop-down and then click OK.



Figure 3: Select Data Source Type

1. Add metadata about the WCF service in the WCF Connection dialog.



 Figure 4: Add metadata connections

1. Once done we have the web methods created exposed in the data connection—so now we can create the ECT and save it to the Metadata Store.
2. Perform the required operation on the web methods like Read item, Read List etc.
3. To create a list based on our new external content type select “Create Lists & Form”



Figure 5: Create List & Form

1. The result is a new external list, which is reflective of the class name properties as the column headers.

We have created an external list using BCS for SP-O using a WCF service talking to a SQL Azure back-end.

## Other Integration Options:

**Cloud Connector**

The Cloud Connector App was especially developed to solve the specific challenges of remotely hosted disconnected systems in the cloud without any direct access to corporate business data and very restricted feature customization capabilities. A data integration strategy needs to be within the foundation of the cloud computing plan. The connector solves many issues that still exists with the *Business Connectivity Services (BCS)* on-premise and in the cloud. It offers a more holistic approach that combines the power of data-, application- and process integration.

Features & Benefits

* Push external line-of-business data to a native SharePoint list into the cloud (or local) - always up-to-date. We don't need to program a web service that allows to pull data from outside or replicate SharePoint libraries.
* No programming or tools required. Setup in a few minutes.
* No installation or customizing in the SharePoint cloud required.
* Bi-directional connections.
* Secure data connection and transfer
* Virtually any data sources supported
* High performance



Figure 6: Cloud Connector -Architecture

Using the Client Object Model and jQuery to Display Data from SQL Azure in SPO

Once we have got an external list, it’s pretty easy to create a new face for it using jQuery and the Client Object Model.

* Created a document library called JavaScript to house all of my jQuery libraries and CSS
* Upload jQuery libraries and CSS templates.
* Create a JavaScript file that referenced these libraries(will be a txt file)
* Create a content editor web part that then referenced the .txt file by editing the web part and pasting the link to the TXT file in the Content Link field.
* Get the context for the SPO site, get the list by its title, define a query and then get the items in the list by using the *getItems* method.
* With an object that now contains the data; we can now walk through the data object and begin to add content to the jQuery accordion control.
* The result is a nicely formatted accordion control that loads the external list data that is stored in SQL Azure (and retrieved via an external content type that communicates with the WCF service).

**Connecting SharePoint to Windows Azure with Silverlight Web Parts**

* Create and deploy the custom Windows Azure WCF service.
* Create a Silverlight-enabled Web Part that can consume the custom Windows Azure service.
* Deploy and use the Silverlight-enabled Web Part in your SharePoint site.

# Pros and Cons of Cloud (Office 365 & SQL Azure):

Some of the benefits of moving to the Microsoft cloud include the following:

* Outsourcing the hassle of installing, managing, patching, and upgrading extremely complex software systems.
* Having predictable and known costs associated with adoption.
* Keeping the lights blinking green and the software up-to-date and secure falls on Microsoft and is backed by service guarantee.
* Reducing cost in not only immediate monetary value but also in efficiency and resource reallocation benefits.
* Backing up and securing your data. After all, Microsoft may not be perfect, but its teams of engineers are extremely specialized and are experts at hosting the software that their colleagues have developed.
* Using the software over the Internet — simply sign up and you’re ready to go. Without the cloud, a SharePoint deployment could take months.

Some of the cons that come along with adopting a cloud solution in general include the following:

* Relying on network and bandwidth. If your Internet provider goes down, then you haven’t any access to your enterprise software and data. Microsoft does not control how you access the Internet and, therefore, cannot account for any failures.
* Having data controlled by someone other than your employees. Your data is hosted in Microsoft’s data center. That can be both a benefit and a detriment. If you feel uncomfortable with your data out there somewhere, then you can either research the Microsoft data centers further or keep your data and applications locally in your own controlled data center.

# Summary

* SQL Azure lets you host a relational database in the cloud
* You can use Windows Azure to build service entry points for the database
* SharePoint Online can consume this data:
1. BCS
2. Excel
3. InfoPath
4. Silverlight

# Useful links to refer

* <http://msdn.microsoft.com/en-us/magazine/gg309179.aspx>
* <http://blogs.msdn.com/b/steve_fox/archive/2011/10/05/using-windows-azure-to-connect-lob-data-to-sharepoint-online-using-business-connectivity-services.aspx>
* <http://www.ilovesharepoint.com/2011_11_01_archive.html>
* <http://msdn.microsoft.com/en-us/library/ff798353.aspx>
* <http://blogs.msdn.com/b/steve_fox/archive/2011/11/12/leveraging-wcf-services-to-connect-bcs-with-sharepoint-online.aspx>
* <http://www.layer2.de/en/products/Pages/Cloud-Connector-for-SharePoint-2010-Office365.aspx>
* <http://blogs.msdn.com/b/steve_fox/archive/2011/10/05/using-windows-azure-to-connect-lob-data-to-sharepoint-online-using-business-connectivity-services.aspx>
* <http://msdn.microsoft.com/en-us/magazine/gg309179.aspx>
* <http://www.wictorwilen.se/Post/SharePoint-Online-and-External-Data-using-JSONP.aspx>