Microsoft Azure - Starter Kits for Partners

Hands on Lab

SharePoint / BI Scenario

Last Update: September 2015





**MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS DOCUMENT.**

The information contained in this document represents the current view of Microsoft Corporation on the issues discussed as of the date of publication. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information presented after the date of publication.

Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Microsoft Corporation.

Microsoft may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Microsoft, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

The descriptions of other companies’ products in this document, if any, are provided only as a convenience to you. Any such references should not be considered an endorsement or support by Microsoft. Microsoft cannot guarantee their accuracy, and the products may change over time. Also, the descriptions are intended as brief highlights to aid understanding, rather than as thorough coverage. For authoritative descriptions of these products, please consult their respective manufacturers.

© 2015 Microsoft Corporation. All rights reserved. Any use or distribution of these materials without express authorization of Microsoft Corp. is strictly prohibited.

Microsoft and Windows are either registered trademarks of Microsoft Corporation in the United States and/or other countries.

The names of actual companies and products mentioned herein may be the trademarks of their respective owners.

Contents

[Overview 4](#_Toc402886978)

[Objectives 4](#_Toc402886979)

[Prerequisites 4](#_Toc402886980)

[Exercises 4](#_Toc402886981)

[Phase 1: Configure Azure 4](#_Toc402886982)

[Phase 2: Configure Domain Controllers 4](#_Toc402886983)

[Phase 3: Configure SQL Server Infrastructure 4](#_Toc402886984)

[Phase 4: Configure SharePoint Servers 4](#_Toc402886985)

[Phase 5: Configure AlwaysOn Availability Groups 4](#_Toc402886986)

[Phase 6: Create Availability Groups on Virtual Machines. 4](#_Toc402886987)

[Phase 7: Creating and using SQL Aliases 4](#_Toc402886988)

[Phase 8: Create and Configure the SharePoint Farm 4](#_Toc402886989)

[Phase 9: Power Pivot for SharePoint 2013 4](#_Toc402886990)

[Configure Excel Services in your SharePoint Server farm 4](#_Toc402886991)

[Configure a Data Model server in Excel Service 4](#_Toc402886992)

[Configure Secure Store 4](#_Toc402886993)

[Download and Configure Microsoft SQL Server 2012 SP1 PowerPivot for Microsoft SharePoint 4](#_Toc402886994)

[Summary 4](#_Toc402886995)

[Additional References 4](#_Toc402886996)

## Overview

Using Microsoft Azure, as your Infrastructure as a Service (IaaS) platform, will enable you to create and manage your infrastructure quickly, provisioning and accessing any host ubiquitously. Grow your business through the cloud-based infrastructure, reducing the costs of licensing, provisioning and backup.

In this hands-on Lab, you will learn how to deploy a SharePoint with SQL Server Always On in Azure.

As of now, this document is focused on the infrastructure aspects for deploying SharePoint on Azure Virtual Machines. The procedures for creating and configuring the SharePoint farm are similar to on-premises as discussed [here](http://technet.microsoft.com/en-us/library/ee805948(v=office.15).aspx#CreateConfigure), this article briefly mention how to deploy PowerPivot for SharePoint (BI).

|  |
| --- |
| **Note:** Due to constant changes in the Azure Portal this document may not reflect the most recent updates. We suggest that you consult the following online links that provide the most up-to-date steps to install and deploy a Sharepoint farm in Azure. It includes steps to deploy:   * Basic Dev/Test Sharepoint farm * High-availability Dev/Test Sharepoint farm * Hybrid cloud Dev/Test farm * High-availability, intranet Sharepoint Production farm   **SharePoint farms hosted in Azure infrastructure services**  <https://azure.microsoft.com/en-us/documentation/articles/virtual-machines-sharepoint-infrastructure-services/> |

**Audience**: IT Pro, Architect, Application Owners and Developers

### Objectives

In this hands-on lab, you will learn how to:

* Configure Azure to create a storage account, cloud services, and a cross-premises virtual network.
* Create and configure the Active Directory Domain Services (AD DS) domain controllers.
* Create and configure the SQL Server virtual machines and prepare them for use with SharePoint
* Create and configure the four SharePoint virtual machines.
* Create availability groups using Windows Server 2012 R2 features.
* Prepare the databases for inclusion in SQL Server Availability Groups.
* Install and deploy SharePoint Farm
* Optionally, deploy PowerPivot for SharePoint.

### Prerequisites

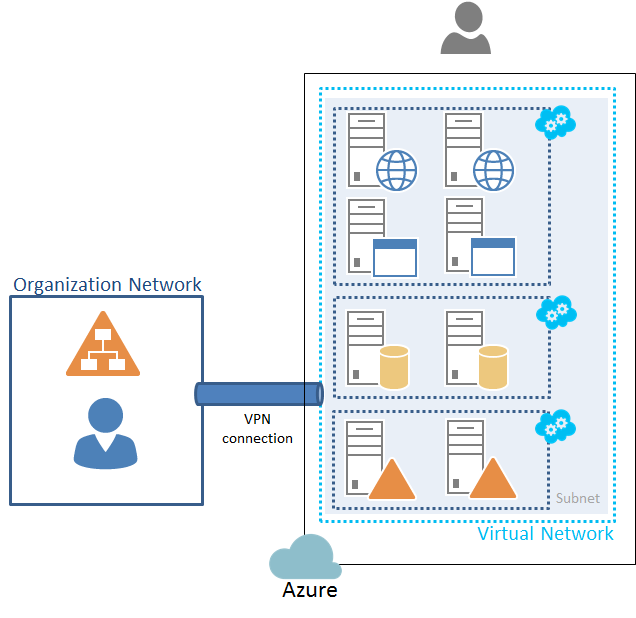
The following is required to complete this hands-on lab:

* A Microsoft Azure subscription - [sign up for a free trial](http://aka.ms/WATK-FreeTrial)

## Exercises

This hands-on lab includes the following exercises:

1. **Phase 1: Configure Azure**. Use the Azure Management Portal to create a storage account, cloud services, a virtual network, and the virtual machines.
2. **Phase 2: Configure Domain Controllers**. Configure the Active Directory Domain Services (AD DS) domain controllers.
3. **Phase 3: Configure SQL Server Infrastructure**. Prepare the SQL Server virtual machines for use with SharePoint.
4. **Phase 4: Configure SharePoint Servers.** Configure the four SharePoint virtual machines.
5. **Phase 5: Configure AlwaysOn Availability Groups**. Create availability groups using Windows Server 2012 R2 features.
6. **Phase 6: Create Availability Groups on Virtual Machines.** Prepare the databases for inclusion in SQL Server Availability Groups.
7. **Phase 7: Create and use SQL Aliases.** Create and use SQL Aliases for SharePoint Databases.
8. **Phase 8:** **Create and Configure SharePoint Farm.** Install SharePoint Farm.
9. **Phase 9:** **PowerPivot for SharePoint 2013.** Install and configure PowerPivot for SharePoint 2013.



Note: If you decide to deploy PowerPivot for SharePoint, you should add an additional server to install Analysis Services in SharePoint Mode.

Estimated time to complete this lab: 7 hours

## Phase 1: Configure Azure

**[Before you begin](javascript:void(0)" \o "Collapse)**

Before you begin configuring Azure components, fill in the following tables. To assist you in the procedures for configuring Azure, print this section and write down the needed information or copy this section to a document and fill it in.

For the names of your new virtual machines, fill in Table 1.

### 

|  |  |  |
| --- | --- | --- |
| Item | Description | Value |
| M1 | A name to assign to the first domain controller (example DC1) |  |
| M2 | A name to assign to the second (backup) domain controller (example DC2) |  |
| M3 | A name to assign to the first SQL Server virtual machine (example SQL1) |  |
| M4 | A name to assign to the second SQL Server virtual machine (example SQL2) |  |
| M5 | A name to assign to the first SharePoint application server virtual machine (example APP1) |  |
| M6 | A name to assign to the second SharePoint application server virtual machine (example APP2) |  |
| M7 | A name to assign to the first SharePoint web server virtual machine (example WEB1) |  |
| M8 | A name to assign to the second SharePoint web server virtual machine (example WEB2) |  |

**Table 1: Virtual machine names**

For the names of Azure components, fill in Table 2.

### 

|  |  |  |
| --- | --- | --- |
| Item | Description | Value |
| N1 | A name to assign to the Azure Virtual Network (example SPFarmNet) |  |
| N2 | A name to assign to your organization network (example OrgNet) |  |
| N3 | The name of a previously configured or new Azure affinity group (example ECoastUSA)  Affinity groups are a way to physically group Azure services together at the same Azure datacenter to increase performance. A virtual network can only be assigned to a single affinity group. |  |
| N4 | A name to assign to the subnet in the virtual network (example SPFarmNet) |  |
| N5 | A proposed name to assign to the cloud service for the domain controllers |  |
| N6 | A proposed name to assign to the cloud service for the SharePoint servers |  |
| N7 | A proposed name to assign to the cloud service for the SQL servers |  |
| N8 | A proposed name to assign to a storage account (example SPFarm) |  |

**Table 2: Names of Azure components**

For the IP addresses and address spaces of the virtual network, fill in Table 3.

### 

|  |  |  |
| --- | --- | --- |
| Item | Description | Value |
| A1 | The public IPv4 address of your VPN device's interface on the Internet |  |
| A2 | The private and public IP address space(s) assigned to your organization network in terms of an address prefix for each space  Note that this address space must include all reachable destinations on your organization network and must not include the address space that you have assigned to the virtual network. |  |
| A3 | The address space (defined in a single private address prefix) for the virtual network |  |
| A4 | The address space of the subnet, based on the overall address space for the virtual network |  |
| A5 | The fourth and fifth possible IP addresses for the address space in Item A4 (these addresses for the two domain controllers will be configured as DNS servers for the subnet) |  |

**Table 3: Addresses and address spaces**

Note that when creating the Azure virtual network, the Azure Management Portal will determine the following:

* The public IPv4 address of the Azure VPN gateway for your virtual network
* The Internet Protocol security (IPsec) pre-shared key for the site-to-site connection

To see these in the Azure Management Portal after you create the virtual network, click **Networks**, click the name of the virtual network, and then click the **Dashboard** menu option.

[**Configure the Azure Virtual Network**](javascript:void(0))

To configure an Azure environment to host the SharePoint 2013 intranet farm, you need the following:

* A new Azure cross-premises virtual network.
* A new affinity group (if needed).
* Three new Azure cloud services for the groups of virtual machines.
* A new storage account to host the virtual machines and added disks.

[**Create a new Azure Virtual Network**](javascript:void(0))

1. Sign in to the [Azure Management Portal](http://manage.windowsazure.com/).
2. From the task bar, click **New** > **Network Services**> **Virtual Network** > **Custom Create**.
3. On the **Virtual Network Details** screen:
   * In **Name**, type the name from Item N1 in Table 2.
   * In **Location**, select the desired region. Your virtual network will be created in the Azure datacenter located in the specified region.
4. Click the next arrow to continue.
5. On the **DNS Servers and VPN Connectivity** page:
   * In **DNS Servers**, configure the following:

### 

|  |  |
| --- | --- |
| DNS Server Name | IP Address |
| SPFARMDNS1 | The first IP address from Item A5 in Table 3. |
| SPFARMDNS2 | The second IP address from Item A5 in Table 3. |

* + In **Site-to-Site Connectivity**, select **Configure a site-to-site VPN**.

1. Click the next arrow to continue.
2. On the **Site-to-Site Connectivity** page:
   * In **Name**, type the name from Item N2 in Table 2.
   * In **VPN device IP address**, type the address from Item A1 in Table 3.
   * In **Address Space**, enter the IP address spaces of your organization network from Item A2 in Table 3, in terms of the prefix (in **Starting IP**) and the prefix length (in **CIDR (Address Count)**). For example, for the address prefix 10.0.0.0/8, "10.0.0.0" is the prefix and "8" is the prefix length.
3. Click the next arrow to continue.
4. On the **Virtual Network Address Spaces** page:
   * In **Address Space**, enter the private IP address space of the virtual network from Item A3 in Table 3, in terms of the prefix (in **Starting IP**) and the prefix length (in **CIDR (Address Count)**).
   * In **Subnets**:
     + Enter the private IP address space of the subnet from Item A4 in Table 3, in terms of the prefix (in **Starting IP**) and the prefix length (in **CIDR (Address Count)**).
     + Enter the name of the subnet from Item N4 in Table 2.
   * Click **Add Gateway Subnet**.
5. Click the check mark to complete.

For more information, see [About Configuring a Virtual Network in the Management Portal](http://msdn.microsoft.com/en-us/library/windowsazure/jj156074.aspx).

To create the site-to-site VPN connection between the new virtual network and an on-premises VPN device, see [Configure a Virtual Network Gateway in the Management Portal](http://msdn.microsoft.com/en-us/library/azure/jj156210.aspx).

[**Create three new Azure Cloud Services**](javascript:void(0))

1. In the task bar of the Azure Management Portal, click **New** > **Compute** > **Cloud Service** > **Quick Create**.
2. In **URL**, enter the name from Item N5 in Table 2 for the cloud service for the domain controllers. This must be a unique name to Azure, so you might have to change it. When you have a unique name, update Item N5 in your Table 2 page with the actual name.
3. In **Region or Affinity Group**, select the region created for the virtual network (Item N3).
4. Click the checkbox next to **Create Cloud Service** to create the cloud service.
5. Click **New** > **Compute** > **Cloud Service** > **Quick Create**.
6. In **URL**, enter the name from Item N6 in Table 2 for the cloud service for the SharePoint servers. This must be a unique name to Azure, so you might have to change it. When you have a unique name, update Item N6 in your Table 2 with the actual name.
7. In **Region or Affinity Group**, select the region created for the virtual network (Item N3 in Table 2).
8. Click the checkbox next to **Create Cloud Service** to create the cloud service.
9. Click **New** > **Compute** > **Cloud Service** > **Quick Create**.
10. In **URL**, enter the name from Item N7 in Table 2 for the cloud service for the SQL servers. This must be a unique name to Azure, so you might have to change it. When you have a unique name, update Item N7 in your Table 2 with the actual name.
11. In **Region or Affinity Group**, select the region created for the virtual network (Item N3 in Table 2).
12. Click the checkbox next to **Create Cloud Service** to create the cloud service.

[**Create a new Azure Storage Account**](javascript:void(0))

1. In the task bar of the Azure Management Portal, click **New** > **Data Services** > **Storage** > **Quick Create**.
2. In **URL**, enter the name from Item N8 in Table 2. This must be a unique name to use for the storage account URL, so you might have to change it. When you have a unique name, update Item N8 in your Table 2 with the actual name.
3. In **Location/Affinity Group**, select the region created for the virtual network (Item N3 in Table 2).
4. In **Replication**, select **Geo-Redundant**. Data loss may occur if you use striped volumes (Windows or Linux) in geo-replicated storage accounts. If a storage outage occurs that requires restoring data from a replicated copy, there is no guarantee that the write order of the stripe disk set would be intact once restored. So pay attention on using Geo-Redundant replication with data and operations that are integrity sensitive.
5. Click the check box to create the storage account.

For more information, see [Create a Storage account on Azure](http://www.windowsazure.com/en-us/manage/services/storage/how-to-create-a-storage-account/).

You need to create the virtual machines listed in Table 4.

### 

|  |  |  |  |
| --- | --- | --- | --- |
| Virtual machine | Cloud service name | Gallery image | Size |
| 1. First domain controller, name from Item M1 in Table 1 (example DC1) | Item N5 in Table 2 | Windows Server 2012 R2 Datacenter | A2 |
| 2. Second domain controller, name from Item M2 in Table 1 (example DC2) | Item N5 in Table 2 | Windows Server 2012 R2 Datacenter | A2 |
| 3. First SQL Server computer, name from Item M3 in Table 1 (example SQL1) | Item N7 in Table 2 | Microsoft SQL Server 2012 SP1 Enterprise on Windows Server 2008 R2 Service Pack 1 | A7 or larger |
| 4. Second SQL Server computer, name from Item M4 in Table 1 (example SQL2) | Item N7 in Table 2 | Microsoft SQL Server 2012 SP1 Enterprise on Windows Server 2008 R2 Service Pack 1 | A7 or larger |
| 5. First SharePoint application server, name from Item M5 in Table 1 (example APP1) | Item N6 in Table 2 | Microsoft SharePoint Server 2013 Trial | A4 |
| 6. Second SharePoint application server, name from Item M6 in Table 1 (example APP2) | Item N6 in Table 2 | Microsoft SharePoint Server 2013 Trial | A4 |
| 7. First SharePoint web server, name from Item M7 in Table 1 (example WEB1) | Item N6 in Table 2 | Microsoft SharePoint Server 2013 Trial | A4 |
| 8. Second SharePoint web server, name from Item M8 in Table 1 (example WEB2) | Item N6 in Table 2 | Microsoft SharePoint Server 2013 Trial | A4 |

**Table 4 – Virtual machines for SharePoint 2013 intranet farm in Azure**

|  |
| --- |
| **ImportantImportant** |
| Create the virtual machines in the order shown in Table 4, starting with the two domain controllers. To verify the assignments, after creating the VM, select each in the Azure Management Portal and click **Dashboard**. The first domain controller should have the fourth possible IP address from the address space assigned to the subnet and the second domain controller should have the fifth possible IP address. |

The process of configuring the creation of a virtual machine can be done quickly. However, the actual provisioning of each machine can take several minutes. To save time, configure the creation of several virtual machines concurrently after the two domain controllers have been provisioned.

Create the virtual machines using the following procedure.

[**Creating a VM from the gallery**](javascript:void(0))

1. In the task bar of the Azure Management Portal, click **New** > **Compute** > **Virtual Machine** > **From Gallery**.
2. On the Choose an image page, select the appropriate image for the virtual machine. Appropriate choices include:
   * Domain controllers (virtual machines 1 and 2 from Table 4): **Windows Server R2 2012 Datacenter**
   * SQL Server computers (virtual machines 3 and 4 from Table 4): **SQL Server 2012 SP1 Enterprise on Windows Server 2012**
   * SharePoint web and application servers (virtual machines 5 through 8 from Table 4): **Microsoft SharePoint Server 2013 Trial**
3. Click the next arrow icon to continue.
4. On the **Virtual machine configuration** page:
   * In **Virtual Machine Name**, type the name of this computer from the appropriate row in Table 1. This name will be displayed in the Azure Management Portal.
   * In **Tier**, select the Basic or Standard tier according to your needs. Basic tier are cheaper but does not provide some important features such as Load balancing or auto-Scaling. In addition, Virtual Machine sizes availability are limited as well as Disk IOPs.
   * In **Size**, select the size for the virtual machine based on the appropriate row in Table 4. Virtual machines can be reconfigured after creation. In the Azure Management Portal, click **Virtual Machines**, click the name of the virtual machine, and then click the **Configure** menu option.
   * In **New User Name**, type a user name for a local administrator account. You will use this name to log on to the machine when initially accessing it through a Remote Desktop connection.
   * In **New Password**, enter a strong password for the local administrator account.
   * In **Confirm Password**, re-enter the strong password.  
       
     Record the local administrator account name and password in a spreadsheet or document for future reference.
5. Click the next arrow to continue.
6. On the **Virtual machine configuration** page:
   * In **Cloud Service**, select the name of the cloud service for the virtual machine based on the appropriate row in Table 4.
   * In **Region/Affinity Group/Virtual Network**, select your newly-created virtual network (Item N1 in Table 2).
   * In **Virtual Network Subnets**, select the name of your subnet (Item N4 in Table 2).
   * In **Storage Account**, select the storage account that you just created (Item N8 in Table 2).
   * In **Availability Set**, for the first machine in a pair of servers with the same role, click **Create availability set** and type a name for the set (such as SPFarmDCs, SPFarmSLQs, SPFarmAPPs, and SPFarmWEBs). For the second machine in a pair of servers with the same role, click the name of the availability set created for the previous server in the pair. This will create four different availability sets for the four different roles, with each set containing both servers for that role.
7. Click the next arrow to continue.
8. On the **Virtual machine configuration** page, in **Endpoints**, click the **PowerShell** row and delete **5986** from the **Public Port** column.

This will tell Azure to generate a random public port for remote Windows PowerShell sessions for this virtual machine. To see this port number in the Azure Management Portal, click **Virtual Machines**, click the name of the virtual machine, and then click the **Dashboard** menu option.

1. Click the check mark to create the machine. Virtual machine creation can take several minutes.

[**Creating and attaching empty disks to a virtual machine**](javascript:void(0))

Domain controllers and SQL Server machines require an extra disk. Table 5 lists the disk size for each virtual machine role.

### 

|  |  |
| --- | --- |
| Virtual Machine Role | Disk Size in GB |
| Domain controllers (virtual machines 1 and 2 from Table 4) | 20 |
| Database servers (virtual machines 3 and 4 from Table 4) | 500 |

**Table 5 – Extra empty data disks**

Use the following procedure to attach an empty data disk to the Azure VMs in Table 5.

[**Attaching an empty disk to a Azure VM**](javascript:void(0))

1. In the Azure Management Portal, click **Virtual Machines**, and then select the virtual machine to which you want to attach the data disk. You must wait until the VM is provisioned and its status is **Running** before attaching the empty disk.
2. On the command bar, click **Attach**, and then click **Attach Empty Disk**. The **Attach Empty Disk** dialog box appears. By default, the storage created earlier is entered into the **Storage Location**. The **File Name** box contains an automatically generated name that does not need to be altered.
3. In **Size**, type the size from the appropriate row in Table 5.
4. Leave the **Host Cache Preference** set to the default value **None**.
5. Click the check mark to attach the empty data disk. Wait until the status is **Running** before continuing (a few minutes).

For additional guidance on creating data disks, see [Disks](http://msdn.microsoft.com/en-us/library/azure/3d15682e-6c44-4967-88dc-0150543e4d83#bkmk_disks).

|  |
| --- |
| **noteNote** |
| This procedure attaches an empty 500 GB disk to the SQL Server VMs. The disk is intended to contain all new databases. However, the system databases (created by default with each SQL Server instance) remain on the system disk. A better practice is to move all databases to the attached disk, including those created before attaching the empty disk. |

[**Initializing an empty disk**](javascript:void(0))

A data disk must be initialized before it can be used. Do the following procedure on the VMs in Table 5.

[**To initialize an empty disk**](javascript:void(0))

1. Log onto the virtual machine with a remote desktop connection using the local administrator account you specified when it was created. For instructions, see [Logging on](http://msdn.microsoft.com/en-us/library/azure/dn275966.aspx#bkmk_loggingOn).
2. In **Server Manager**, in the left pane, click **File and Storage Services**.
3. In the left pane, click **Disks**. In the list of disks, select the empty disk, which is identifiable by its capacity and its **Partition** set to **Unknown**.
4. In the **Volumes** panel, click **To create a volume, start the Volume Wizard**. Follow the steps in the wizard to initialize the disk. Accept all defaults to create a GUID Partition Table (GPT) disk assigned to the drive letter "F". Give the volume a friendly name to distinguish it from other disks.

[**Logging on**](javascript:void(0))

Logging on to a machine with a remote desktop connection depends on whether or not it is joined to a domain. The first time you log onto a virtual machine, you must log on as the local administrator account created with the machine. After you join the VM to a domain, you can log on with domain accounts. For example, you need one user account that has domain privileges in order to configure machines. Similarly, you must have one user account that is used to manage SQL Server hosts.

[**Connecting to a VM after its creation**](javascript:void(0))

The first time you log on to a VM after its creation, use the local administrator account user name and password that you specified when you created the VM.

1. In the Azure Management Portal, in the left panel click **Virtual Machines**.
2. To connect to a VM, click **Running** in the **Status** column next to its name.
3. In the task bar on the bottom of the page, click **Connect**.
4. The Management Portal informs you the **.rdp** file is being retrieved. Click **OK**.
5. The browser dialog appears asking, "Do you want to open or save ComputerName.rdp from manage.windowsazure.com." Click **Open**.
6. In the **Remote Desktop Connection** dialog, click **Connect**.
7. In the **Windows Security** dialog, click **Use another account.**
8. In **User name**, type the name of the VM and user name of the local administrator account created with the VM (a local machine account). Use the following format:   
     
   ComputerName\LocalAdministratorAccountName
9. In **Password**, type the password for the local administrator account.
10. Click **OK**.
11. In the **Remote Desktop Connection** dialog, click **Yes**. The desktop of the new machine appears in a Remote Desktop session window.

[**Connecting to a VM that is joined to a domain**](javascript:void(0))

To connect to a VM after it has joined an AD DS domain, follow the preceding steps. However, the credentials (the name and password) are for a domain account, rather than the local administrator account. The name is in the format DomainName\DomainUserAccountName. For example, contoso\sp\_install.

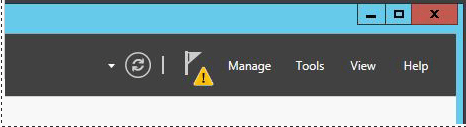
## [Phase 2: Configure Domain Controllers](http://msdn.microsoft.com/en-us/library/azure/dn275963.aspx)

**Configuring Domain Controllers**

You must now configure the domain controller and its backup.

**Add Active Directory Role to the first Domain Controller**

1. Log on to one of the domain controllers VMs created in Phase 1: Configure Azure. For instructions to log on, see [Connecting to a VM after its creation](http://msdn.microsoft.com/en-us/library/azure/9f9a7366-357b-4728-a298-6bdf489bc9e2#bkmk_connectToVMafterCreation).
2. In the left pane of the Server Manager click **Dashboard**. Then click **Add Roles and Features**.
3. In the **Before you begin** page click the **Next** button.
4. In the **Select Installation type**, accept the default and click the **Next** button.
5. In the **Select destination server** page accept the default selection (**Select a server from the server pool** and click **Next**.
6. In the **Select server roles** page, select **Active Directory Domain Services.**, and then click **Next**.
7. In the **Add features that are required for Active Directory Domain Services** dialog, click the **Add features** button.
8. In the **Select server roles** page click the **Next** button.
9. In the **Select features** dialog, accept the defaults, and then click **Next**.
10. In the **AD DS** dialog, click **Next**.
11. In the **Confirm installation settings** dialog, click **Install**. The installation takes several minutes, but you can close the wizard. When the process is complete, a yellow flag will appear in the upper right corner of the Server Manager Dashboard.

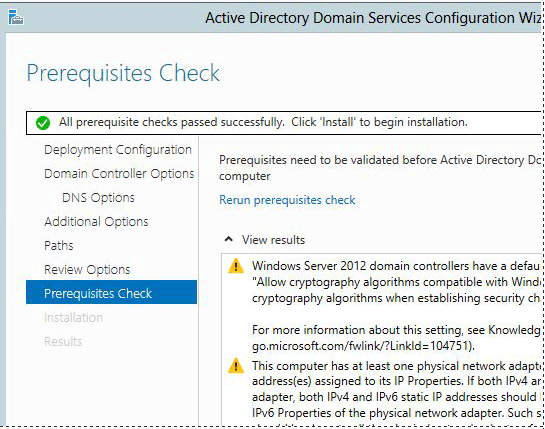


**Promote the VM to Domain Controller**

1. In the Server Manager application, in the upper right corner, a yellow flag alert appears next to the **Manage** link.
2. Click the icon and click **Promote this server to a domain controller**.
3. In the **Deployment Configuration** dialog, select **Add a new forest**.
4. In the **Root domain name** box type **<YourDomainName>**. The name ends with the suffix **.com**, for example **contoso.com**. For an experimental installation, the domain name can be any acceptable string. The name is not validated against other (actual) domain names.
5. In the **Domain Controller Options** dialog, type a password into the **Password** and **Confirm password** boxes. (The password is not used in this scenario.) Then click **Next**.
6. In the **DNS Options** dialog click **Next**. A warning that "a delegation for the DNS cannot be found" appears, but it can be ignored.
7. In the **Additional Options** dialog click **Next**.
8. In the **Paths** dialog, for each of the paths, click the ellipsis to set the path. Create a new folder named **NTDS** on the "F" disk. Set the path for the database folder and the log files folder to the new folder. The paths should be as follows:
   * Database folder: **F:\NTDS**
   * Log files folder: **F:\NTDS**
   * SYSVOL folder: **F:\Windows\SYSVOL**

Click the **Next** button.

1. In the **Review Options** dialog, click the **Next** button. After about 30 seconds, a set of warnings appear that can be ignored.

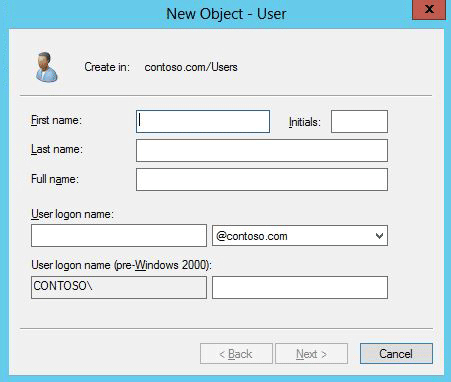


1. In the **Prerequisites Check** dialog, click the **Install** button. The installation takes a few minutes and restarts the VM.

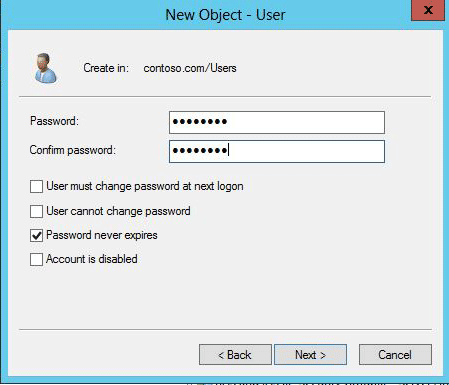
When the VM returns to the running state, it is ready for the next action, which is to add several user accounts to the domain.

**Add new user accounts to the domain**

1. Connect to the virtual machine using the <YourDomainName>\<LocalAdministrator> account.
2. Open the Server Manager. In the upper right corner, click **Tools** and click **Active Directory Users and Computers**.
3. In the **Active Directory Users and Computers** dialog, expand the domain name. In the folder with the domain name (for example, Contoso.com), select the **Users** folder.
4. Click the **Action** menu and click **New**, and click **User**.



1. Use the dialog box to create four new users. The **First Name** and **Last Name** fields can be ignored. Only the **Full name** and **User log on name** fields are required. Each user requires a password. Uncheck the **User must change password at next logon** box. Optionally, select the **Password never expires** checkbox.
   * **sp\_farm**: manages SharePoint farms.
   * **sp\_farm\_db**: user that has **sysadmin** rights on SQL Server instances.
   * **sp\_install**: user that has domain administration rights needed for installing roles and features.
   * **sqlservice**: identity that SQL instances can run as.



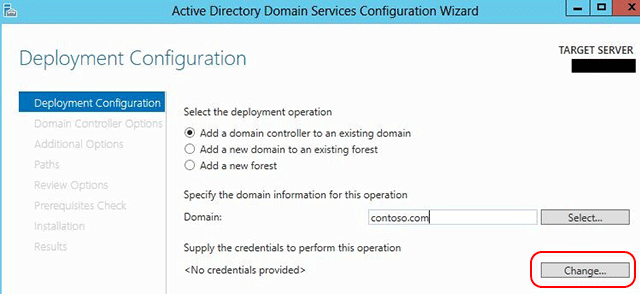
1. In the **Users** panel, select the **sp\_install** user.
2. Right-click the selected node and click **Add to a group**.
3. In the **Select Groups** dialog, type **domain admins**. Then click the **Check Names** button to resolve the name.
4. After the **Domain Admins** group is resolved, click **OK** to add the user to the group and close the dialog.
5. Grant further rights to the **sp\_install** account using the **Active Directory Users and Computers** dialog
   * In the dialog, click **View** and click **Advanced Features**. The option lets you see all hidden containers and hidden tabs in the property windows for AD objects.
   * Right-click your domain name and click **Properties**.
   * In the **Properties** dialog, select the **Security** tab and click the **Advanced** button.
   * In the **Advanced Security settings for <YourDomainName>** window, click **Add**.
   * In the text box, type **<YourDomainName>\sp\_install** and click **OK**.
   * Select **Allow** for **Read all properties** and **Create computer objects**, then click OK three times to close the dialog box.

**Creating a Backup Domain Controller**

To ensure availability when changes are made, create a second VM that serves as a backup domain controller.

**To create a backup Domain Controller**

1. Log onto the VM with the local administrator account that was created with the machine. Use the **MachineName**\**username** format.
2. Initialize the empty disk that was attached to the VM.
3. In the Server Manager dashboard, click **Add Roles and Features**.
4. Use the Wizard to add the **Active Directory Domain Services** Role to the virtual machine.
5. After the role has been added to the VM, click the yellow flag alert that appears next to the **Manage** link
6. In the **Deployment Configuration** page, type in the name of your domain into the **Domain** dialog, for example **contoso.com**.
7. Click the **Change** button.



1. In the **Windows Security** dialog, type the user name **sp\_install** into the **User name** box using the **<YourDomainName>**\**sp\_install** format. For example, **contoso.com\sp\_install**.
   1. Click the **OK** button.
   2. Click the **Next** button.
2. Type in the password for the user.
3. Use the **Add a domain controller to an existing domain** option.
4. Set the identity of the user to one of the **Domain Admins** group users. The operation must be run as a user that has domain-level privileges.
5. In the **Paths** dialog, for each of the paths, click the ellipsis to set the path. Create a new folder named **NTDS** on the "F" disk. Set the path for the database folder and the log files folder to the new folder. The paths should be as follows:
   1. Database folder: **F:\NTDS**
   2. Log files folder: **F:\NTDS**
   3. SYSVOL folder: **F:\Windows\SYSVOL**

Click the **Next** button.

1. Complete the promotion using the Wizard and accepting its defaults.

**Joining a Virtual Machine to a domain**

All Virtual Machines in the SharePoint with SQL Server AlwaysOn deployment must be joined to a domain. To join the VM to the domain, you must have the username and password for one of the user accounts created in [Add new user accounts to the domain](http://msdn.microsoft.com/en-us/library/azure/dn275963.aspx#bkmk_addUsers).

1. Connect to the new VM using the user name and password created with the VM. For more details, see [Connecting to a VM after its creation](http://msdn.microsoft.com/en-us/library/azure/9f9a7366-357b-4728-a298-6bdf489bc9e2#bkmk_connectToVMafterCreation).
2. After connecting, open the **Control Panel**.
3. Click **System and Security**.
4. Click **System**.
5. Under **Computer name, domain, and workgroup settings**, click **Change settings**.
6. In the **System Properties** dialog, in the **Computer Name** tab, click the **Change…** button.
7. Select the **Domain** option, and type the name of your domain, for example **contoso.com**, and click **OK**.
8. In the **Windows Security** dialog, type one of the user names created earlier in the format: username@domain.com, for example **sp\_install@contoso.com**. Type the user's password into the Password box and click **OK**. The machine will be joined to the domain after restarting.

## [Phase 3: Configure SQL Server Infrastructure](http://msdn.microsoft.com/en-us/library/azure/dn275965.aspx)

**Configure a Virtual Machine with SQL Server 2012**

This procedure is executed on both virtual machines that host SQL Server. One VM contains the primary database replica of an availability group. The second VM contains the secondary backup replica. The backup is provided to ensure high-availability.

By default, only the user account that was created with the Virtual Machine (in the Azure Management portal) can access the SQL Server default instance. But the Virtual Machine must be joined to the domain in order to add a domain user account to the list of log ins. Therefore, you must first join the machine to the domain, then log on as the original user in order to access the SQL Server instance.

1. From the Azure Management portal, attach a 500GB empty disk to the VM that hosts SQL Server. For more information, see [Creating and Attaching and Empty Disks to a Virtual Machine](http://msdn.microsoft.com/en-us/library/azure/9f9a7366-357b-4728-a298-6bdf489bc9e2#bkmk_CreateandAttachEmptyDisk).
2. Connect to the VM using the user name and password created with the machine. Use the format **<machineName>\<username>**. By default, the user is a member of the **Administrators** group. For more information, see [Connecting to a VM after its creation](http://msdn.microsoft.com/en-us/library/azure/9f9a7366-357b-4728-a298-6bdf489bc9e2#bkmk_connectToVMafterCreation).
3. Use the [Disk Management](http://support.microsoft.com/kb/309000) tool to partition the disk and assign it the letter **F:**.
4. On the new partition, create three folders named Data, Log, and Backup.
5. Join the VM to the domain. For instructions, see [Joining a Virtual Machine to a domain](http://msdn.microsoft.com/en-us/library/azure/57b7c9ec-eeba-4db9-846d-22b256a5bade#bkmk_JoinVMtoDomain).

When prompted for a user that has permission to join the domain, specify a member of the **Domain Admins** group, such as **sp\_install**. Use the format: **sp\_install**@**<domainName>.com**.

1. Restart the machine to complete the join.
2. Connect to the VM using the user name and password created with the machine. Use the format: **<machineName>.cloudapp.net\<username>**. (At this point, only the user created with the machine has permission to log onto the SQL Server instance.)
3. On the **Start** menu, type **SQL Server Management Studio** and open the tool.
4. Connect to the SQL Server default instance using the Windows account.
5. Right-click the top node—the default instance named after the machine—and click **Properties**.
6. In the **Server Properties** window, click **Database Settings**.
7. Locate the **Database default locations** and set the values for Data, Log, and Backup to the newly created folders in the Data Disk. Only new databases use these locations.
8. Click the **OK** to close the window.
9. In the left pane, expand the **Security** folder.
10. Right-click **Logins**, then click **New login**.
11. In the **Login-New** dialog, type **sp\_farm\_db**. Use the format: **sp\_farm\_db@<domainName>.com**
12. After the name resolves, click **OK** to close the dialog.
13. In the left pane, click **Server Roles**, and select the checkbox for **Sysadmin** role. Click **OK** to close the dialog.
14. Right-click **Logins** and click **New Login**.
15. In the **Login name** box, type **NT Authority\System** and click **OK**.
16. In **Object Explorer**, right-click **NT AUTHORITY\System** and click **Properties**.
17. In the **Securables** page, for the local server, select **Grant** for the following permissions and click **OK**.
    * Alter any availability group
    * Connect SQL
    * View server state
18. Click **OK** and close the **SQL Server Management Studio**.
19. SQL Server requires a port that clients use to access the database server. It also needs a port to manage the high-availability group. By default, the client port number is 1433, and the high availability port is 5022. The Windows Firewall, however, blocks both ports. To open the ports:
    * In the **Start** menu, type **Windows firewall with advanced security**.
    * Click **Inbound Rules**.
    * On the **Action** pane, click **New Rule**.
    * On the **Rule Type** page, select **Program**, and click **Next**.
    * In the **Program** page, select **This program path** and type **C:\Program Files\Microsoft SQL Server\MSSQL12.MSSQLSERVER\MSSQL\Binn\sqlservr.exe** in the text box. Then, click **Next**.
    * On the **Action** page, keep the default ("Allow the connection") and click **Next**.
    * On the **Profile** page, accept the defaults, and click **Next**.
    * On the **Name** page, type **SQL Server ports 1433 and 5022**, and click **Finish**.

## [Phase 4: Configure SharePoint Servers](http://msdn.microsoft.com/en-us/library/azure/dn275962.aspx)

**Configuring the SharePoint VM**

The procedure here is used to configure the SharePoint application. For more guidance, refer to [Deploying SharePoint with SQL Server AlwaysOn in Azure](http://msdn.microsoft.com/en-us/library/azure/dn275959.aspx).

1. Log on to the SharePoint virtual machine using the <**MachineName**>.cloudapp.net\<**UserName**> format. (See [Connecting to a VM after its creation](http://msdn.microsoft.com/en-us/library/azure/9f9a7366-357b-4728-a298-6bdf489bc9e2#bkmk_connectToVMafterCreation).)
2. Join the VM to the domain. (See [Joining a Virtual Machine to a domain](http://msdn.microsoft.com/en-us/library/azure/57b7c9ec-eeba-4db9-846d-22b256a5bade#bkmk_JoinVMtoDomain).)
3. After the VM restarts, from the Management Portal, connect to the machine using the <**DomainName**>.com\sp\_farm\_db user name. The user must have permission to log on to the SQL Server database machine. (See [Connecting to a VM that is Joined to a Domain](http://msdn.microsoft.com/en-us/library/azure/9f9a7366-357b-4728-a298-6bdf489bc9e2#bkmk_connectToDomainVM).)
4. After connecting to the Virtual Machine, navigate to the desktop.
5. Double-click **SharePoint 2013 Products Configuration Wizard**. When asked to allow the program to make changes to the computer, click **Yes**
6. The **Welcome to SharePoint Products** dialog appears. As stated in the dialog, you must have the SQL database server name, as well as the username and password for to access the database. Click **Next**.
7. A **SharePoint Products Configuration Wizard** dialog appear warning that services (such as IIS) will be restarted or reset. Click the **Yes** button.
8. The **Connect to a server farm** dialog appears. Select the appropriate option: either connect to an existing farm, or create a server farm. The first time you run the wizard, select the option to **create** a server farm. If you have already created a farm, then click the option to connect to an existing farm. Then click **Next**.
9. In the **Specify Configuration Database Settings** dialog, enter the name of the primary VM that hosts SQL Server. In the **Username** box type <**domainName**>**.com\sp\_farm\_db** (a user account that is a **sysadmin** on the database server). In the **Password** box type the password.
10. Click the **Next** button.
11. In the **Specify Farm Security Settings** page, type a passphrase, and confirm it. Then click **Next**.
12. The **Configure SharePoint Central Administration Web Application** page appears. Accept the defaults and click **Next**.
13. The **Completing the SharePoint Products Configuration Wizard** page appears. Accept the defaults and click **Next**.
14. The **Configuring SharePoint Products** page appears. Wait until the configuration process completes, about 8 minutes.
15. After the farm is successfully configured, click the **Finish** button. The new administration website starts.
16. To start configuring the SharePoint farm, click **Start the Wizard**.

After concluding the initial setup, more configuration options will be available. Use the documentation found in [Planning for SharePoint 2013 on Azure Infrastructure Services](http://msdn.microsoft.com/en-us/library/azure/dn275958.aspx) to proceed.

## [Phase 5: Configure AlwaysOn Availability Groups](http://msdn.microsoft.com/en-us/library/azure/dn275964.aspx)

**Configuring AlwaysOn Availability Groups**

SharePoint relies on SQL Server to store data for user sites and applications. Because of this reliance on data, it is recommended that you implement the AlwaysOn feature. On Azure Virtual Machines, the feature enables high availability.

|  |
| --- |
| **ImportantImportant** |
| It is important to understand how the failover system works. Without an understanding, your system could fail without a backup. For an overview of the options, see [High Availability and Disaster Recovery for SQL Server in Azure Virtual Machines](http://msdn.microsoft.com/library/windowsazure/jj870962.aspx). |

The procedures here are based on [Test Lab: Create an AlwaysOn Availability Group in Azure End-to-End](http://blogs.msdn.com/b/sqlalwayson/archive/2013/01/23/test-lab-create-an-alwayson-availability-group-in-windows-azure-end-to-end.aspx). A tutorial that uses PowerShell to accomplish the same tasks is found in [Tutorial: AlwaysOn Availability Groups in Azure](http://msdn.microsoft.com/en-us/library/windowsazure/jj870963.aspx)

For troubleshooting the configuration, see [Troubleshoot AlwaysOn Availability Groups Configuration (SQL Server)](http://msdn.microsoft.com/library/ff878308.aspx)

**Enabling Failover Clustering**

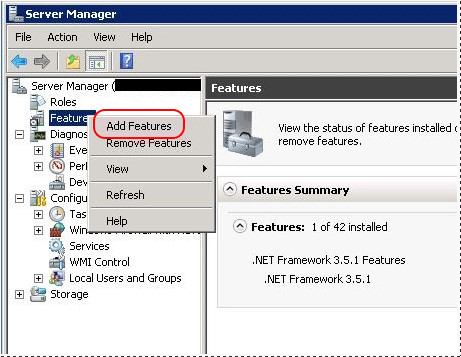
AlwaysOn relies on the Windows Server feature **Windows Server Failover Clustering** (WSFC). The feature allows several machines to participate as a group in a cluster, also known as a a node. When one machine fails, a second machine is ready to take its place. (These high-level details are purposefully simple.) Therefore the first task is to enable the Failover Clustering feature on all of the participating machines. In this scenario, enable the feature on three VMs:

* SQL1
* SQL2
* DC1 (Domain controllers are not recommended as cluster nodes and in Windows Server 2012 they are not supported, use a member server instead)

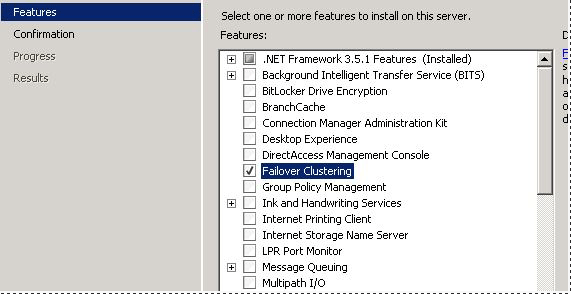
The failover cluster requires at least three VMs and must be using same version and edition of OS. Two machines host SQL Server. The second VM is a synchronous secondary replica, ensuring zero data loss if the primary machine fails. The third machine does not need to host SQL Server (although it could). In this scenario, the third machine functions as a quorum witness in the WSFC. Because the WSFC cluster relies on a quorum to monitor health, there must always be a majority to ensure that the WSFC cluster is online. If only two machines are in a cluster, and one fails, there can be no majority when only one out of two fails. For more information, see [WSFC Quorum Modes and Voting Configuration (SQL Server)](http://msdn.microsoft.com/library/hh270280.aspx).

**To add the Failover Clustering feature to a Virtual Machine**

1. Log on to the Virtual Machine as a member of the **Domain Admins** (for example, **sp\_install**).
2. Open the Server Manager.
3. In the left pane, right-click the **Features** node and click **Add Features**.



1. Select **Failover Clustering** and click **Next**.



1. In the **Confirm Installation Selections** page, click **Install**.
2. When the **Results** page appears, click the **Close** button.

**Creating a Windows Server Failover Cluster (WSFC)**

Due to current non-RFC-compliant behavior by the DHCP in Azure, creation of a WSFC cluster can fail. For details, search for "**WSFC cluster behavior in Azure networking**" in [High Availability and Disaster Recovery for SQL Server in Azure Virtual Machines](http://msdn.microsoft.com/library/windowsazure/jj870962.aspx). However, there is a workaround. To create a cluster, execute the following high-level tasks, on one machine, in the specified order:

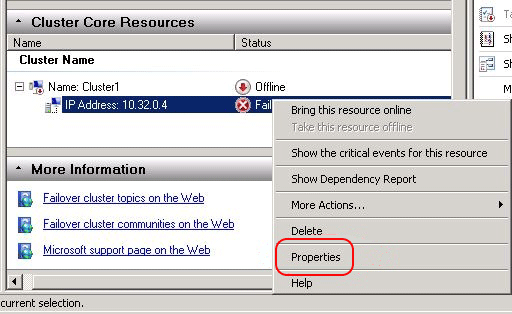
1. Create a single-node cluster on the primary SQL machine.
2. Change the IP address of the cluster to an unused IP address.
3. Bring the cluster name online.
4. Remove the resource that contains the changed IP address.
5. Add the other nodes (VMs with the WSFC feature enabled).

**Create a single-node cluster on the principal SQL machine**

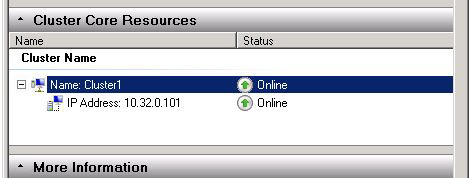
1. Log on to the primary SQL Server host. Use the same identity used to enable clustering on the virtual machines: **sp\_install**.
2. Open **Server Manager** and then expand the **Features** node. Right-click Failover Cluster Manager and click **Create a cluster**.



1. In the **Create Cluster Wizard**, click **Next**. Type the name of the principal SQL Server machine, and click **Add**.
2. In the **Validation Warning** page, click **No, I do not require support from Microsoft for this cluster, and therefore do not want to run the validation tests. When I click Next, continue creating the cluster**. Then, click **Next**.
3. In the **Cluster Name** text box, type **Cluster1**, then click **Next**.
4. In the **Confirmation** page, click **Next** to begin cluster creation. Once the cluster is created, click **Finish**.
5. In **Server Manager**, expand **Failover Cluster Manager**, then click Cluster1.< **domainName**>.com, then scroll down in the center pane, and expand **Cluster Core Resources**. The **Name** and **IP Address** resources appear in the **Failed** state. The IP address resource cannot be brought online because the cluster is assigned the same IP address as that of the machine itself. The result is a duplicate address. Right-click the failed IP Address resource and then click **Properties**.



1. In the **IP Address Properties** box, in the **Adress** box, type in **10.10.2.101 (any unused IP address)**. Then click **OK**,
2. In the **Cluster Core Resources** section, right-click **Name: Cluster1** and click **Bring this resource online**. Wait until both resources are online. When the cluster name resource comes online, it updates the DC server with a new Active Directory (AD) computer account. This AD account is later used to run the availability group clustered service.



1. Now that the AD account is created, bring the cluster name offline. Right-click the **Name: Cluster1** resource and click **Take this resource offline**. In the pop-up confirmation dialog, click **Take Name: Cluster1 offline**.
2. Remove the cluster IP address. Right-click **IP Address: 10.10.2.101** and click **Delete**. The **Name: Cluster1 resource** can no long come online because it depends on the IP address resource. However, an availability group does not depend on the cluster name or IP address in order to work properly. So the cluster name can be left offline.
3. Add the remaining nodes to the cluster. In the browser tree, right-click **Cluster1. <domainName>.com** and click **Add Node**.
4. In the **Add Node Wizard**, click **Next**. In the **Select Servers** page, add all the machines that will participate in the cluster. For the current scenario, add:
   * Backup SQL Server Host
   * Domain controller (use of Domain controllers are not recommended as cluster nodes and in Windows Server 2012 they are not supported, use a member server instead)

If a machine cannot be added, and the error message is "the Remote Registry is not running," do the following. Log on to the machine, open the Services snap-in (services.msc), and enable the Remote Registry. For more information, see [Unable to connect to Remote Registry service](http://technet.microsoft.com/en-us/library/bb266998(v=EXCHG.80).aspx).

After adding the two machines, click **Next**.

1. In the **Validation Warning** page, click **No, I do not require support from Microsoft for this cluster, and therefore do not want to run the validation tests. When I click Next, continue creating the cluster.** Then click **Next** twice to add the nodes. Once the nodes are added to the cluster, click **Finish**.

The Failover Cluster Manager should now show that your cluster has three nodes, and lists them in the **Nodes** container.

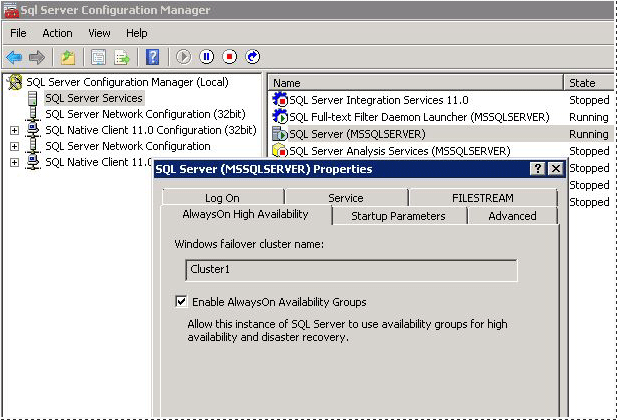
**Create an Availability Group**

The next step is to create an Availability Group using the SQL Server Configuration Manager, and SQL Server Management Studio.

|  |
| --- |
| **noteNote** |
| An availability group in SQL Server differs from an Azure Availability Set. The **Availability Group** contains databases that are highly-available and recoverable. The **Availability Set** allocates Virtual Machines to different fault domains. For more information about fault domains, see [Manage the Availability of Virtual Machines](http://azure.microsoft.com/en-us/documentation/articles/virtual-machines-manage-availability/). |

**To enable Availability Groups on SQL Server**

1. Log on to the primary SQL Server host as sp\_farm\_db@<**domainName**>.com. You can briefly add user to admin group to gain remote access login until you have configured your AG. The user must have **sysadmin** rights on the SQL Server instance.
2. On the **Start** menu, type **SQL Server Configuration Manager**. Open the application.
3. In the left pane, select **SQL Server Services**
4. In the list that appears, double-click **SQL Server (MSSQLSERVER)**.
5. In the **SQL Server (MSSQLSERVER) Properties dialog, click AlwaysOn HighAvailability**.
6. Click the **AlwaysOn High Availability** tab, then select **Enable AlwaysOn Availability Groups**, and then click **Apply**. Click **OK** in the pop-up dialog, and do not close the properties window yet. You will restart the SQL Server service after you change the service account.



1. Next, change the SQL Server service account. Click the **Log On** tab, then type <**DomainName**>\sqlservice in **Account Name**. Fill in and confirm the password, and then click **OK**.
2. In the pop-up window, click **Yes** to restart the SQL Server service.
3. Log on to the second SQL Server host VM, and repeat the process.

## [Phase 6: Create Availability Groups on Virtual Machines](http://msdn.microsoft.com/en-us/library/azure/dn275957.aspx).

**Create the Availability Group and Add Databases**

You can now add databases to the Availability Group. SharePoint creates several databases as part of the initial configuration. Those databases must be prepared, and the preparation consists of two steps. For each database:

1. Take a full backup and a transaction log backup of the database on the primary machine.
2. Restore the full and log backups on the backup machine.

Once the databases have been both backed up and restored, they can be added to the availability group. SQL Server only allows databases that have been backed up (at least once), and restored on another machine, to be in the group.

**Share the Backup Folders**

To enable backup and restore, the backup files (.bak) must be accessible from the second SQL Server host VM. Use the following procedure:

1. Log on to the primary SQL Server Host as <**domainName**>.com\**sp\_farm\_db**.
2. Navigate to the **F:\** disk.
3. Right-click the **Backup** folder and click **Share with** and click **specific people**.
4. In the **File sharing** dialog, type **sqlservice**@<**domainName**>.com.
5. Then click **Add**.
6. Click the **Permission Level** column for the account name, and click **Read/Write**. On the backup SQL Server host, give the account only **Read** permission.
7. Click the **Share** button.
8. Click the **Done** button.

**Backing Up and Restoring a Database**

The procedure here must be repeated for every database that needs to be part of the availability group.

**To back up a database**

1. Log on to the primary SQL Server host machine as <**domainName**>.com\**sp\_farm\_db**.
2. Open **SQL Server Management Studio** and log in as **sp\_farm\_db**.
3. Expand the **Databases** node.
4. Right-click a database to back up and click **Tasks**, then click **Backup**.
5. In the **Source** section, keep **Backup type set to Full**. In the **Destination** section, click **Remove** to remove the default file path for the backup file.
6. In the **File name** text box, type \\<**machineName**>\backup\<**databaseName**>.bak. Then, click **OK**, and then click **OK** again to back up the database. When the backup operation completes, click OK again to close the dialog.
7. Next, take a transaction log backup of the database. In the **Object** Explorer, expand **Databases**, then right-click <**databaseName**>, then point to Tasks, and then click **Back Up**.
8. In **Backup** type, select **Transaction Log**. Keep the **Destination** file path set to the one you specified earlier and click **OK**. Once the backup operation completes, click **OK** again.
9. Keep the remote desktop session open if you have not yet created the Availability group.

**To restore a database**

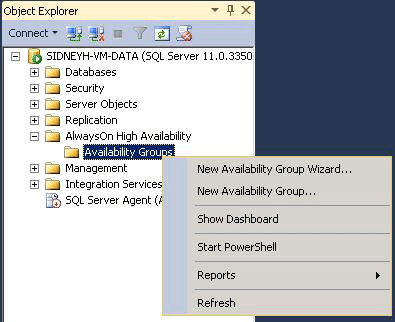
1. Log on to the second SQL Server host machine as <**domainName**>.com\**sp\_farm\_db**.
2. Open **SQL Server Management Studio** and log in as **sp\_farm\_db**.
3. In the **Object Explorer**, right-click **Databases** and click **Restore Database**.
4. In the **Source** section, select **Device**, and click the **…** button
5. In **Select backup devices**, click **Add**.
6. In **Backup file location**, type \\<**machineName**>\backup, then click **Refresh**, then select <**databaseName**>.bak, then click **OK**, and then click **OK** again. You should now see the full backup and the log backup in the **Backup** sets to restore pane.
7. Go to the **Options** page. In **Recovery state** select **RESTORE WITH NORECOVERY**, and then click **OK** to restore the database. Once the restore operation completes, click **OK**.

**Creating an Availability Group**

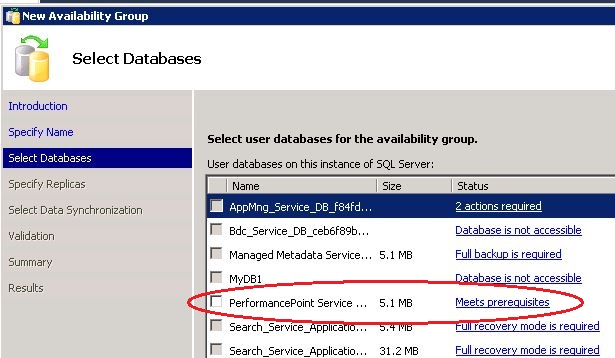
After at least one database is prepared (using the backup and restore method), create an Availability Group.

**To create an Availability Group**

1. Return to the remote desktop session (the primary SQL Server host VM).
2. In **SQL Server Management Studio**, in the **Object Explorer**, right-click **AlwaysOn High Availability** and click **New Availability Group Wizard**.



1. In the Introduction page, click **Next**.
2. In the **Specify Availability Group Name** page, type **AG1** in Availability group name. Then click **Next**.
3. In the **Select Databases** page, select the database that was backed up, and click **Next**. The database meets the prerequisites for an availability group because you have taken at least one full backup on the intended primary replica.



1. In the **Specify Replicas** page, click **Add Replica**.
2. The **Connect to Server** dialog appears. In the **Server name** box type the name of the second SQL Server host: <**BackupMachineName**>. Then click **Connect**.
3. In the **Specify Replicas** page, the backup SQL Server host is listed in **Available Replicas**. For both instances, set the following option values:

### 

|  |  |  |
| --- | --- | --- |
| Server | Option | Value |
| Primary SQL Server | Automatic Failover (Up to 2) | Selected |
| Backup SQL Server | Automatic Failover (Up to 2) | Selected |
| Primary SQL Server | Synchronous Commit (Up to 3) | Selected |
| Backup SQL Server | Synchronous Commit (Up to 3) | Selected |
| Primary SQL Server | Readable Secondary | Yes |
| Backup SQL Server | Readable Secondary | Yes |

1. Click **Next**.
2. On the **Select Initial Data Synchronization** page, select **Join only** and click **Next**. Data synchronization is executed manually by taking the full and transaction backups on the primary server, and restoring it on the backup. You can instead choose to select **Full** to let the **New Availability Group Wizard** perform data synchronization for you. However, synchronization is not recommended for large databases that are found in some enterprises.
3. On the **Validation** page, click **Next**. There is a warning for a missing listener configuration because an availability group listener is not configured. This step is skipped because availability group listeners have only been recently supported and is more complex to configure than on-premises due to characteristics of Azure. You should follow [this article](http://msdn.microsoft.com/en-us/library/azure/dn425027.aspx) to create the AG Listener after you have finished creating the AG in this section. This way you will be easier updated on the recent changes and limitations of AG listener in Azure. After creating and testing the AG listener you can proceed to install SharePoint using SharePoint configuration Wizard. Remember that you should use SQL Aliases since you will be using a <AG listener>,<non-default port> combination.
4. On the **Summary** page, click **Finish**. Once the wizard is finished, inspect the **Results** page to verify that the availability group is successfully created. If so, click **Close** to exit the wizard.
5. From the **Start** menu, open the **Server Manager** again. Expand **Features**, **Failover Cluster Manager**, <**DomainName**>.com, and then expand **Services and applications**. A new clustered service called **AG1** appears in **Cluster1**. **AG1** does not have an IP address by which database clients can connect to the availability group. This lack is by design, because the Azure Virtual Network does not support static IP addresses.

## Phase 7: Creating and using SQL Aliases

Create SQL Aliases for each Web and Application server on the SharePoint farm.

1. On SharePoint Server where you will run the **Configuration Wizard** open **CLICONFIG.exe** from C:\Windows\System32\cliconfg.exe (64 bit version of cliconfig.exe)
2. Enable **TCP/IP** under General tab
3. Click on **Alias** tab
4. Type a **New Name** for the SQL alias
5. Uncheck Dynamically determine port
6. Type the **AG listener** and **Port**

Validate the SQL Alias

1. Create a new text file and rename “TestDBConnection.udl”
2. Double-click to open the file and enter you **SQL Alias** name
3. Use **Windows Integrated Security**
4. You should see all databases when you click on “Select the database on the server”

Perform this on all SharePoint servers.

## Phase 8: Create and Configure the SharePoint Farm

To create and configure the farm, you run the SharePoint Products Configuration Wizard. This wizard automates several configuration tasks, such as creating the configuration database, installing services, and creating the Central Administration website. We recommend that you run the SharePoint Products Configuration Wizard on the server that will host the SharePoint Central Administration website before you run the wizard on the other servers in the farm.

**To run the SharePoint Products Configuration Wizard and configure the farm**

1. Verify that the user account that is performing this procedure is the Setup user account. For information about the Setup user account, see [Initial deployment administrative and service accounts in SharePoint 2013](http://technet.microsoft.com/en-us/library/ee662513(v=office.15).aspx).
2. On the server that will host Central Administration (the application server), click **Start**, point to **All Programs**, and then click **SharePoint 2013 Products**, and then click **SharePoint 2013 Products Configuration Wizard**. If the **User Account Control** dialog box appears, click **Continue**.
3. On the **Welcome to SharePoint Products** page, click **Next**.
4. In the dialog box that notifies you that some services might have to be restarted during configuration, click **Yes**.
5. On the **Connect to a server farm** page, click **Create a new server farm**, and then click **Next**.
6. On the **Specify Configuration Database Settings** page, do the following:
   1. In the **Database server** box, type the name of the **SQL Alias** you created earlier
   2. In the **Database name** box, type a name for your configuration database, or use the default database name. The default name is SharePoint\_Config.
   3. In the **Username** box, type the user name of the server farm account in DOMAIN\user name format.

|  |
| --- |
| **ImportantImportant:** |
| The server farm account is used to create and access your configuration database. It also acts as the application pool identity account for the SharePoint Central Administration application pool, and it is the account under which the SharePoint Timer service runs. The SharePoint Products Configuration Wizard adds this account to the SQL Server Login accounts, the SQL Server **dbcreator** server role, and the SQL Server **securityadmin** server role. The user account that you specify as the service account has to be a domain user account. However, it does not have to be a member of any specific security group on your web servers or your database servers. We recommend that you follow the principle of least-privilege, and specify a user account that is not a member of the Administrators group on your front-end web servers or your database servers. |

* 1. In the **Password** box, type the user password.

1. Click **Next**.
2. On the Specify Farm Security Settings page, type a passphrase, and then click **Next**.
3. On the **Configure SharePoint Central Administration Web Application** page, do the following:
   1. Either select the **Specify port number** check box and type the port number that you want the SharePoint Central Administration web application to use, or leave the **Specify port number** check box cleared if you want to use the default port number.

|  |
| --- |
| **NoteNote:** |
| If you want to access the SharePoint Central Administration website from a remote computer, make sure that you allow access to the port number that you configure in this step. You do this by configuring the inbound rule for **SharePoint Central Administration v4** in Windows Firewall with Advanced Security. |

* 1. Click either **NTLM** or **Negotiate (Kerberos)**.

1. Click **Next**.
2. On the **Completing the SharePoint Products Configuration Wizard** page, click **Next**.
3. On the **Configuration Successful** page, click **Finish**.

|  |
| --- |
| **NoteNote:** |
| If the SharePoint Products Configuration Wizard fails, check the log files on the drive on which SharePoint 2013 is installed, which are located in the %COMMONPROGRAMFILES%\Microsoft Shared\Web Server Extensions\15\LOGS folder. |

1. The Central Administration website will open in a new browser window.

On the Help Make SharePoint Better page, click one of the following options and then click **OK**.

* 1. **Yes, I am willing to participate (Recommended).**
  2. **No, I don’t wish to participate.**

1. On the **Initial Farm Configuration Wizard** page, you have the option to use a wizard to configure services or you can decide to configure services manually. For the purpose of this article, we use the manual option. Click **Cancel**.

The choice that you make here is a matter of personal preference. The Farm Configuration Wizard will configure some services automatically when you run it. However, if you configure services manually, you have greater flexibility in designing your logical architecture.

For information about how to use the wizard to configure services, see [Configure services and service applications in SharePoint 2013](http://technet.microsoft.com/en-us/library/ee794878(v=office.15).aspx). If you are using Microsoft Office Web Apps, see [How Office Web Apps work on-premises with SharePoint 2013](http://technet.microsoft.com/en-us/library/ff431685(v=office.15).aspx).

|  |
| --- |
| **ImportantImportant:** |
| If you are using a DBA-created database, you cannot use the Farm Configuration Wizard, you must use SharePoint Products Configuration Wizard. |

## Phase 9: Power Pivot for SharePoint 2013

Several elements constitute the BI features of SharePoint 2013. In this lab we will address the steps to configure Power Pivot for SharePoint 2013. Additional information about Reporting services, Performance Point services and Visio Services can be found on the following articles:

* [Installing Reporting Services SharePoint Mode for SharePoint 2013](http://technet.microsoft.com/en-us/library/b29d0f45-0068-4c84-bd7e-5b8a9cd1b538(v=sql.110))
* [Test Lab Guide: Configure PerformancePoint Services](http://technet.microsoft.com/en-us/library/jj900678(v=office.15).aspx)
* [Test Lab Guide: Configure Visio Services](http://technet.microsoft.com/en-us/library/jj900483(v=office.15).aspx)

**Summary of steps:**

1. Configure **Excel Services** in your SharePoint Server farm.
2. Configure a data model server in **Excel Services**. You’ll need an instance of Analysis Services 2012 SP1 configured in **Power Pivot mode** in order to do this.
3. Configure **Secure Store** in your SharePoint Server farm.
4. Download and Configure Microsoft SQL Server 2012 SP1 PowerPivot for Microsoft SharePoint (**spPowerPivot.msi**).

You may notice that the Power Pivot configuration tool that we will be using later in this article has options for deploying Excel Services and Secure Store. However, the Power Pivot configuration tool is not able to configure these services using the recommended accounts, so we recommend that you configure these services manually using the links above.

### Configure Excel Services in your SharePoint Server farm

**Configure the application pool account**

For better security, we recommend that you use a separate domain account to run the Excel Services application pool. Have your domain administrator create a domain account to use in running the Excel Services application pool. No specific domain privileges are required for this account.

Before you can use an account to run an application pool, you must register it as a managed account in SharePoint Server. Use the following procedure to register the account.

**To register a managed account**

1. On the SharePoint Central Administration website home page, in the left navigation, click **Security**.
2. On the Security page, under **General Security**, click **Configure managed accounts.**
3. On the Managed Accounts page, click **Register Managed Account**.
4. Type the user name and password of the domain account that you are registering.
5. Optionally, select the **Enable automatic password change** check box if you want SharePoint Server to manage password changes for this account.
6. Click **OK**.

**Grant content database access to the managed account**

You must also grant access to the SharePoint content database for the account that you will use to run the Excel Services application pool. Use the following procedure for each web application that will be associated with Excel Services.

To grant content database access to the managed account

1. On a SharePoint Server application server, click **Start**, click **All Programs**, click **Microsoft SharePoint 2013 Products**, right-click **SharePoint 2013 Management Shell**, and then click **Run as Administrator**.
2. At the Windows PowerShell Command Prompt, type the following (press Enter after each line):

$w = Get-SPWebApplication -identity http://<WebApplication>

$w.GrantAccessToProcessIdentity("<Domain>\<Username>")

|  |
| --- |
| **ImportantImportant:** |
| If in the future you add additional content databases, you must rerun these cmdlets to ensure that Excel Services has access to the new databases. |

Once you have granted content database access to the application pool account, the next step is to start the Excel Calculation Services service.

[**Start the Excel Calculation Services service**](javascript:void(0))

In order to use Excel Services, you must start the Excel Calculation Services service on at least one application server in the farm. Use the following procedure to start the service.

**To start the Excel Calculation Services service**

1. On the Central Administration home page, in the **System Settings** section, click **Manage services on server**.
2. To select the server where you want to start the service, above the **Service** list, click the **Server** drop-down list, and then click **Change Server** and choose the appropriate server.
3. In the **Service** list, click **Start** next to **Excel Calculation Services**.

After the Excel Calculation Services service has been started, the next step is to create an Excel Services service application.

[**Create an Excel Services service application**](javascript:void(0))

Use the following procedure to create an Excel Services service application.

**To create an Excel Services service application**

1. On the Central Administration home page, under **Application Management**, click **Manage service applications**.
2. On the Manage Service Applications page, click **New**, and then click **Excel Services Application**.
3. In the **Name** section, type a name for the service application in the text box.
4. Select the **Create new application pool** option and type a name for the application pool in the text box.
5. Select the **Configurable** option, and from the drop-down list, select the account that you created to run the application pool.
6. Click **OK**.

### Configure a Data Model server in Excel Service

[**Install Analysis Services in SharePoint**](javascript:void(0)) **Mode**

In this step, you run SQL Server Setup to install an Analysis Services server in SharePoint mode. You must be a local administrator to run SQL Server Setup and the computer must be joined to the domain. You should also install on a new different VM in Azure. In a subsequent step, you configure Excel Services to use this server for workbook data models.

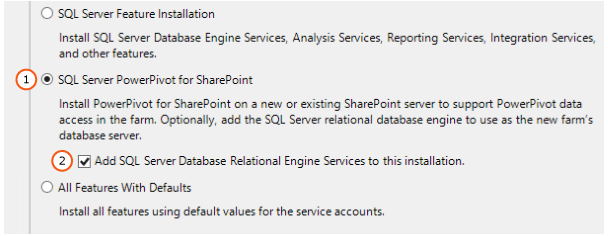
1. Run the SQL Server Installation Wizard (Setup.exe).
2. Click **Installation** in the left navigation.
3. Click **New SQL Server stand-alone installation or add features to an existing installation.**
4. If you see the **Product Key** page, specify the evaluation edition or enter a product key for a licensed copy of the enterprise edition. Click **Next**. For more information on editions, see [Editions and Components of SQL Server 2014](http://msdn.microsoft.com/en-us/library/ms144275.aspx).
5. Review and accept the Microsoft Software License Terms of agreement, and then click **Next**.
6. If you see the **Global Rules** page, review any rules information the setup wizard displays.
7. On the **Microsoft Update** page, it is recommended you use Microsoft Update to check for updates, then click **Next**.
8. The **Install Setup Files** page runs for several minutes. Review any rule warnings or failed rules, and then click **Next**.
9. If you see another **Setup Support Rules**, review any warnings and click **Next**.

**Note**: Because Windows Firewall is enabled, you see a warning to open ports to enable remote access.

1. On the **Setup Role** page, select **SQL Server PowerPivot for SharePoint**. This option installs Analysis Services in SharePoint mode.

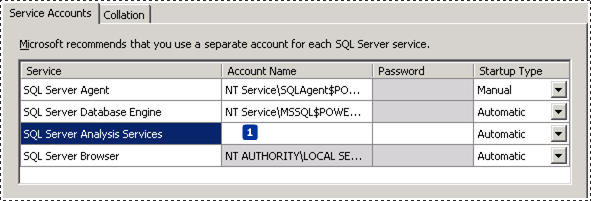
Optionally, you can add an instance of the Database Engine to your installation. You might add the Database Engine when setting up a new farm and need a database server to run the farm’s configuration and content databases. This option also installs SQL Server Management Studio. You should not need to do this as the content databases should run on the AlwaysOn SQL you set earlier.

Click **Next**.



1. In Feature Selection, a read-only list of the features is displayed for informational purposes. You cannot add or remove items the preselected items for this role. Click **Next**.
2. On the **Instance Configuration** page, a read-only instance name of 'PowerPivot' is displayed for informational purposes. This instance name is required and it cannot be modified. However, you can enter a unique Instance ID to specify a descriptive directory name and registry keys. Click **Next**.
3. On the **Server Configuration** page, configure all of the services for Automatic **Startup Type**. Specify the desired domain account and password for **SQL Server Analysis Services**, (1) in the following diagram.
   * For Analysis Services, you can use a **domain user** account or **NetworkService** account. Do not use LocalSystem or LocalService accounts.
   * If you added the SQL Server Database Engine and SQL Server Agent, you can configure the services to run under domain user accounts or under the default virtual account.
   * Never provision service accounts with your own domain user account. Doing so grants the server the same permissions that you have to the resources in your network. If a malicious user compromises the server, that user is logged in under your domain credentials. The user has the permissions to download or use the same data and applications that you do.

Click **Next**.



1. If you are installing the Database Engine, the **Database Engine Configuration** page appears. In Database Engine Configuration, click **Add Current User** to grant your user account administrator permissions on the Database Engine instance.

Click **Next**.

1. On the **Analysis Services Configuration** page, click **Add Current User** to grant your user account administrative permissions. You will need administrative permission to configure the server after Setup is finished.
   * In the same page, add the Windows user account of any person who also requires administrative permissions. For example, any user who wants to connect to the Analysis Services service instance in SQL Server Management Studio to troubleshoot database connection problems must have system administrator permissions. Add the user account of any person who might need to troubleshoot or administer the server now.

|  |
| --- |
| **NoteNote** |
| All service applications that require access to the Analysis Services server instance need to have Analysis Services Administrative permissions. For example, add the service accounts for Excel Services, Power View, and Performance Point Services. Also, add the SharePoint farm account, which is used as the identity of the web application that hosts Central Administration. |

1. Click **Next**.
2. On the **Error Reporting** page, click **Next**.
3. On the **Ready to Install** page, click **Install**.
4. If you see the dialog **Computer Restart Required**, click **OK**.
5. When the installation is complete, click **Close**.
6. Restart the computer.
7. If you have a firewall in your environment, review the SQL Server Books Online topic, [Configure the Windows Firewall to Allow Analysis Services Access](http://msdn.microsoft.com/en-us/library/ms174937.aspx).

**Verify the SQL Server Installation**

Verify that the Analysis Services Service is running.

1. In Microsoft Windows click **Start**, click **All Programs**, and click the **Microsoft SQL Server 2012** group.
2. Click **SQL Server Management Studio**.
3. Connect to the Analysis Services instance, for example **[your server name]\POWERPIVOT**. If you can connect to the instance, you have verified the Service is running.

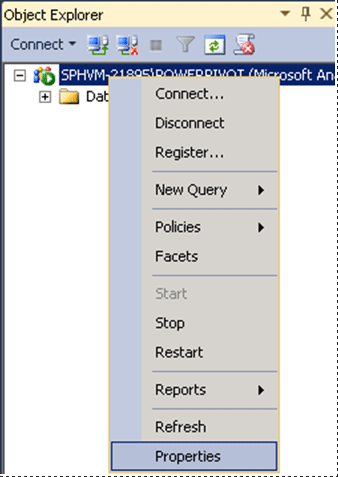
[**Configure Basic Analysis Services SharePoint Integration**](javascript:void(0))

The following steps describe configuration changes needed so you can interact with Excel advanced data models inside a SharePoint document library. Complete these steps after you install SharePoint Server 2013 and SQL Server Analysis Services.

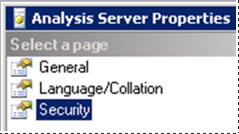
**Grant Excel Services Server Administration Rights on Analysis Services**

You do not need to complete this section if during the Analysis Services installation; you added the Excel Services Application service account as an Analysis Services administrator.

1. On the Analysis Services server, start SQL Server Management Studio and connect to the Analysis Services instance, for example **[MyServer]\POWERPIVOT**.
2. In Object Explorer, Right-click the instance name and click **Properties**.



1. In the left pane, click **Security**. Add the domain login you configured for the Excel Services Application in step 1.



**Configure Excel Services for Analysis Services integration**

1. In SharePoint Central Administration, in the Application Management group, click **Manage Service Applications**.
2. Click the name of your service application, the default is **Excel Services Application.**
3. On the **Manage Excel Services Application** page, click **Data Model Settings**.
4. Click **Add Server.**
5. In **Server Name**, type the Analysis Services server name and the PowerPivot instance name. For example **MyServer\POWERPIVOT**. The PowerPivot instance name is required.

Type a description.

1. Click **Ok**.
2. The changes will take effect in a few minutes or you can **Stop** and **Start** the service **Excel Calculation Services**. To

Another option is to open a command prompt with administrative privileges, and type **iisreset /noforce**.

You can verify the server is recognized by Excel Services by reviewing entries in the ULS log. You will see entries similar to the following:

Excel Services Application Data Model 27 Medium Check Administrator Access ([ServerName]\POWERPIVOT): Pass. f127bd9b-bae3-e0e0-9b48-3f7b5ad1eae6

Excel Services Application Data Model 27 Medium Check Server Version ([ServerName]\POWERPIVOT): Pass (11.0.2809.24 >= 11.0.2800.0). f127bd9b-bae3-e0e0-9b48-3f7b5ad1eae6

Excel Services Application Data Model 27 Medium Check Deployment Mode ([ServerName]\POWERPIVOT): Pass. f127bd9b-bae3-e0e0-9b48-3f7b5ad1eae6

[**Verify the Integration**](javascript:void(0))

The following steps walk you through creating and uploading a new workbook to verify the Analysis Services integration. You will need a SQL Server database to complete the steps. You can create one or use and existing one such as [AdventureWorks2012\_Data.mdf](http://msftdbprodsamples.codeplex.com/downloads/get/165399).

1. **Note:** If you already have an advanced workbook with slicers or filters, you can upload it to your SharePoint document library and verify you are able to interact with the slicers and filters from the document library view.
2. Start a new workbook in Excel.
3. On the Data tab, click **From Other Sources** on the ribbon in the **Get External Data.**
4. Select **From SQL Server**.
5. In the **Data Connection Wizard**, enter the name of the SQL Server instance that has the database you want to use.
6. or Log on credentials, verify that **Use Windows Authentication** is selected, and then click **Next**.
7. Select the database you want to use.
8. Verify that the **Connect to specific table** checkbox is selected.
9. Click the **Enable selection of multiple tables and add tables to the Excel Data Model** checkbox.
10. Select the tables you want to import.
11. Click the checkbox **Import relationships between selected tables**, and then click **Next**. Importing multiple tables from a relational database lets you work with tables that are already related. You save steps because you don't have to build the relationships manually.
12. In the **Save Data Connection File and Finish** page of the wizard,, type a dame for your connection and click **Finish**.
13. The **Import Data** dialog box will appear. Choose **PivotTable Report**, and then click **Ok**.
14. A PivotTable Field List appears in the workbook. On the field list, click the **All** tab
15. Add fields to the Row, Columns, and Value areas in the field list.
16. Add a slicer or a filter to the PivotTable. **Do not skip this step**. A slicer or filter is the element that will help you verify your Analysis Services installation.
17. Save the workbook to a document library on a SharePoint Server 2013 that has Excel Services configured. You can also save the workbook to a file share and then upload it to the SharePoint Server 2013 document library.
18. Click the name of your workbook to view it in SharePoint and click the slicer or change the filter that you previously added. If a data update occurs, you know that Analysis Services is installed and available to Excel Services. If you open the workbook in Excel you will be using a cached copy and not using the Analysis Services server.

### Configure Secure Store

To configure Secure Store, you perform the following steps:

1. Register a managed account in SharePoint Server 2013 to run the Secure Store application pool.
2. Start the Secure Store Service on an application server in the farm.
3. Create a Secure Store Service service application.

To run the application pool, you must have a standard domain account. No specific permissions are required for this account. Once the account has been created in Active Directory, follow these steps to register it with SharePoint Server 2013.

**To register a managed account**

1. On the SharePoint Central Administration Web site home page, in the left navigation, click **Security**.
2. On the Security page, in the **General Security** section, click **Configure managed accounts**.
3. On the Managed Accounts page, click **Register Managed Account**.
4. In the **User name** box, type the name of the account.
5. In the **Password** box, type the password for the account.
6. If you want SharePoint Server 2013 to handle changing the password for the account, select the **Enable automatic password change** box and specify the password change parameters that you want to use.
7. Click **OK**.

Once you have configured the registered account, you must start the Secure Store Service on an application server in the farm. Because Secure Store deals with sensitive information, we recommend that you use a separate application server just for the Secure Store Service for better security.

**To start the Secure Store Service**

1. On the Central Administration home page, in the **System Settings** section, click **Manage services on server**.
2. Above the **Service** list, click the **Server** drop-down list, and then click **Change Server**.
3. Select the application server where you want to run the Secure Store Service.
4. In the **Service** list, click **Start** next to **Secure Store Service**.

Once the service is started, you must create a Secure Store Service service application. Use the following procedure to create the service application.

**To create a Secure Store Service service application**

1. On the Central Administration home page, in the **Application Management** section, click **Manage service applications**.
2. On the Manage Service Applications page, click **New**, and then click **Secure Store Service**.
3. In the **Service Application Name** box, type a name for the service application (for example, **Secure Store Service**).
4. In the **Database Server** box, type the instance of SQL Server where you want to create the Secure Store database.

|  |
| --- |
| **NoteNote:** |
| Because the Secure Store database contains sensitive information, we recommend that you deploy the Secure Store database to a different instance of SQL Server from the rest of SharePoint Server 2013. |

1. Select the **Create new application pool** option and type a name for the application pool in the text box.
2. Select the **Configurable** option, and, from the drop-down list, select the account for which you created the managed account earlier.
3. Click **OK**.

The Secure Store Service has now been configured. The next step is to generate an encryption key for encrypting the Secure Store database.

**Work with encryption keys**

Before using the Secure Store Service, you must generate an encryption key. The key is used to encrypt and decrypt the credentials that are stored in the Secure Store Service database.

**Generate an encryption key**

The first time that you access the Secure Store service application, your only option is to generate a new encryption key. Once the key has been generated, the rest of the Secure Store functionality becomes available.

**To generate a new encryption key**

1. On the Central Administration home page, in the **Application Management** section, click **Manage service applications**.
2. Click the Secure Store service application.
3. In the **Key Management** group, click **Generate New Key**.
4. On the Generate New Key page, type a pass phrase string in the **Pass Phrase** box, and type the same string in the **Confirm Pass Phrase** box. This pass phrase is used to encrypt the Secure Store database.

|  |
| --- |
| **ImportantImportant:** |
| A pass phrase string must be at least eight characters and must have at least three of the following four elements:   * + Uppercase characters   + Lowercase characters   + Numerals   + Any of the following special characters   “! " # $ % & ' ( ) \* + , - . / : ; < = > ? @ [ \ ] ^ \_ ` { | } ~ |
| **ImportantImportant:** |
| The pass phrase that you enter is not stored. Make sure that you write this down and store it in a safe place. You must have it to refresh the key, such as when you add a new application server to the server farm. |

1. Click **OK**.

For security precautions or as part of regular maintenance you may decide to generate a new encryption key and force the Secure Store Service to be re-encrypted based on the new key. You can use this same procedure to do this.

|  |
| --- |
| **CautionCaution:** |
| You should back up the database of the Secure Store Service application before generating a new key. |

### Download and Configure Microsoft SQL Server 2012 SP1 PowerPivot for Microsoft SharePoint

[**Prepare the farm**](javascript:void(0))

In this section, we will prepare the farm for Power Pivot installation. This includes registering a managed account and configuring some Secure Store settings in Central Administration, as well as configuring some security settings in SQL Server. You’ll need to be a farm administrator and a SQL Server administrator to do these steps, as well as a [Shell Administrator](http://technet.microsoft.com/en-us/library/ff607596(v=office.15).aspx) to do the Windows PowerShell steps.

**Register a domain account as a managed account in SharePoint Server 2013**   
The first step is to register the domain account that you want to use to run the Power Pivot application pool as a managed account in SharePoint Server. This will make the account available for us to use when we create the Power Pivot service application later on.

**To register a managed account**

1. In Central Administration, click **Security**.
2. Under **General Security**, click **Configure managed accounts**.
3. Click **Register Managed Account**.
4. Type the username and password of the account that you created to run the Power Pivot application pool.
5. If you want SharePoint Server to handle account password changes for you, select the **Enable password change** check box, and then fill in the details on when you want the password changed.
6. Click **OK**.

**Grant the managed account access to the content database**   
You must also grant access to the SharePoint content database for this account. Use the following procedure for each web application that will be associated with Power Pivot.

**To grant content database access to the managed account**

1. Click **Start**, click **All Programs**, click **Microsoft SharePoint 2013 Products**, right-click **SharePoint 2013 Management Shell**, and then click **Run as Administrator**.
2. At the Windows PowerShell Command Prompt, type the following syntax (press ENTER after each line):

$w = Get-SPWebApplication -identity http://<WebApplication>

$w.GrantAccessToProcessIdentity("<Domain>\<PowerPivotApplicationPoolAccount>")

|  |
| --- |
| **ImportantImportant:** |
| If in the future you add more content databases, you’ll need to rerun these cmdlets to ensure that Power Pivot has access to the new databases. |

**Create a target application in Secure Store**   
The next step is to create a target application in Secure Store for the Power Pivot unattended data refresh account. This target application will be used to refresh the data in your Power Pivot reports.

**To create a target application**

1. In Central Administration, under **Application Management**, click **Manage service applications**.
2. Click the Secure Store service application.
3. In the ribbon, click **New**.
4. In the **Target Application ID** box, type **PowerPivotUnattendedAccount**.

|  |
| --- |
| **NoteNote:** |
| Be sure to use this value so that the Power Pivot for SharePoint 2013 Configuration tool will recognize it. |

1. Type a display name and email address.
2. For **Target Application Type**, make sure **Individual** is selected, and then click **Next**.
3. Ensure **Field Type** is set to **Windows User Name** and **Windows Password**, and click **Next**.
4. Type a name or group for the **Target Application Administrators**, and then click **OK**.
5. Select the target application that you just created, and then in the **Credentials** section of the ribbon, click **Set**.
6. For **Credential Owner**, type the account that you created for the Power Pivot application pool (the one that you just configured as a managed account).
7. Type the user name and password of the credentials that have access to your data sources.
8. Click **OK**.

**Grant access to the SharePoint Server admin content database**   
In order for the Power Pivot Management Dashboard reports to work, the account that runs the Excel Services application pool must have read access to the SharePoint Server admin content database. You configure this in SQL Server.

**To grant admin content database access**

1. Open SQL Server Management Studio and connect to the database engine.
2. Expand **Security** and double-click the account that is used for the Excel Services application pool.
3. On the **User Mapping** page, click the SharePoint Admin content database (usually named SharePoint\_AdminContent\_<GUID>).
4. Select the **SPDataAccess** database role check box.
5. Click **OK**.

The basic farm setup is now complete, so the next thing to do is to install and configure Power Pivot for SharePoint.

[**Configure Power Pivot for SharePoint**](javascript:void(0))

In this section, we’ll configure Power Pivot itself, including installing the Power Pivot software and configuring a service application.

**Install Power Pivot on each server in the farm**   
The first step in configuring Power Pivot in a SharePoint Server farm is to install SQL Server 2012 SP1 PowerPivot for Microsoft SharePoint (sppowerpivot.msi) on each server in the farm.

Use the following procedure to run sppowerpivot.msi on each server in your farm.

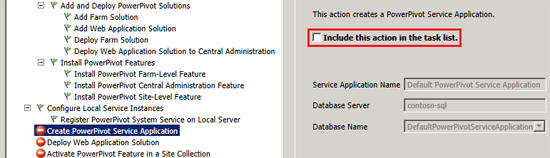
**To install Power Pivot for SharePoint 2013**

1. Double-click sppowerpivot.msi.
2. Follow the wizard to install Power Pivot for SharePoint 2013. When you see the Feature Selection page, leave all of the items selected (this is the default), and then complete the wizard.

**Run the Power Pivot for SharePoint configuration tool**   
Once you have installed sppowerpivot.msi on each server in the farm, the next step is to run the PowerPivot for SharePoint 2013 Configuration tool. Run this program on the server that is running the SharePoint Central Administration website.

**To run the Power Pivot for SharePoint 2013 Configuration tool**

1. Click **Start** > **All Programs** > **Microsoft SQL Server 2012** > **PowerPivot for SharePoint 2013 Configuration**.
2. Choose the **Configure or Repair Power Pivot for SharePoint** option, and then click **OK**.
3. In the Power Pivot Configuration Tool, in the left pane, select **Create PowerPivot Service** application.
4. On the **Parameters** tab, clear the **Include this action in the task list** check box. This will prevent the tool from creating a service application for Power Pivot. We will create the Power Pivot service application manually later.



This will also prevent further configuration steps from being completed, but we’ll come back and do those later as well.

1. Click **Run**.
2. Click **Yes** to confirm.
3. When the action completes, click **OK**, and then click **Exit**.

If you have additional servers in your farm, we’ll eventually run the Power Pivot for SharePoint 2013 Configuration tool on those as well, but first there are a few more steps to do.

**Create a Power Pivot service application**   
We prevented the Power Pivot configuration tool from creating a Power Pivot service application, so that we could create one manually and use the application pool account that we created earlier. Now, we need to go back and create the service application.

**To create a Power Pivot service application**

1. In Central Administration, under **Application Management**, **click Manage service applications**.
2. In the ribbon, click **New**, and then click **SQL Server Power Pivot Service Application**.
3. Type a name for the service application in the **Name** text box.
4. Type a name for the new application pool in the **Application pool name** text box.
5. From the **Configurable** dropdown list, choose the account that you registered as a managed account for the Power Pivot application pool.
6. Click **OK**.

The Power Pivot service application has now been created, though you may need to refresh the page to see it.

**Configure the Power Pivot unattended data refresh account**   
The next step is to configure the Power Pivot unattended data refresh account. We do this in the Power Pivot service application settings by specifying the unattended data refresh account that we created in Secure Store.

**To configure the unattended data refresh account**

1. In Central Administration, under **Application Management**, click **Manage service applications**.
2. Click the Power Pivot service application.
3. Under **Actions**, click **Configure service application settings**.
4. Under **Data Refresh** (about a third of the way down), type the target application ID of the target application that you created (PowerPivotUnattendedAccount) in the **PowerPivot Unattended Data Refresh Account** box.
5. Click **OK**.

**Rerun the Power Pivot for SharePoint 2013 Configuration tool**   
Now that the service application has been created and we’ve configured the unattended data refresh account, we need to run the Power Pivot for SharePoint 2013 Configuration tool once again to complete the remainder of the steps.

**To run the Power Pivot for SharePoint 2013 Configuration tool**

1. Click **Start** > **All Programs** > **Microsoft SQL Server 2012** > **Power Pivot for SharePoint 2013 Configuration**.
2. Choose the **Configure or Repair Power Pivot for SharePoint** option, and then click **OK**.
3. In the Power Pivot Configuration Tool, in the left pane, select **Activate Power Pivot Feature in a Site Collection**.
4. On the **Parameters** tab, choose the site collection where you want to active Power Pivot from the **Site URL** dropdown list.

|  |
| --- |
| **NoteNote:** |
| If you want to activate Power Pivot in more than one site collection, choose one to start with here. It’s easy to add more later. |

1. If you changed the default site collection setting, click **Validate**.
2. Click **Run**, and then click **Yes** to confirm.
3. When the action completes, click **OK**, and then click **Exit**.

Power Pivot setup is now complete and Power Pivot is ready to use.

**Run the Power Pivot for SharePoint Configuration tool on other servers in the farm**   
Because we’ve only run the Power Pivot for SharePoint 2013 Configuration tool on one server, the SQL Server Power Pivot System Service will only be available to run on this server. It’s fine to use Power Pivot this way, but if you have multiple servers in your farm, we recommend that you install the service on all of them. This will allow you to change which server you run the service on (for example, if you want to balance out the services on your farm) or to run the service on multiple servers for greater capacity.

You install the SQL Server Power Pivot System Service by running the Power Pivot for SharePoint 2013 Configuration tool on the other servers in your farm.

Use the following procedure on each server in your farm.

**To run the Power Pivot for SharePoint 2013 Configuration tool**

1. Click **Start** > **All Programs** > **Microsoft SQL Server 2012** > **PowerPivot for SharePoint 2013 Configuration**.
2. Choose the **Configure or Repair Power Pivot for SharePoint** option, and then click **OK**.
3. If you see **Activate PowerPivot Feature in a Site Collection** in the left pane, click on it, and then select the site collection where you want to activate Power Pivot or uncheck the **Include this action in the task list** check box if you don’t want to activate Power Pivot in another site collection.
4. If you made any changes, click **Validate**.
5. Click **Run**, and then click **Yes** to confirm.
6. When the tasks complete, click **OK**, and then click **Exit**.

When you run the Power Pivot for SharePoint 2013 Configuration tool, it installs and turns on the SQL Server Power Pivot System Service on each server where you run it. You probably don’t want this service running on every server, so be sure to go to **Manage services on server** in Central Administration and stop the service on the servers where you don’t want to run it.

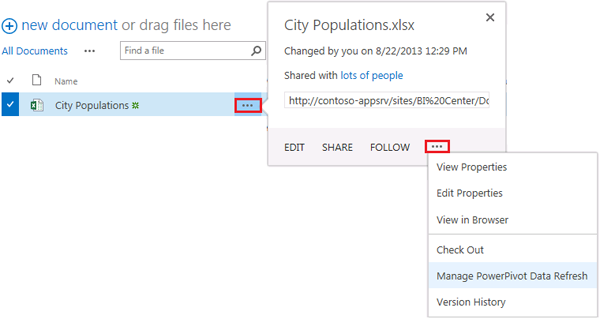
[**Test the Power Pivot for SharePoint installation**](javascript:void(0))

Now that everything is configured, we can do a simple test to see if it’s working. We’ll build a data model in Excel and publish it to a document library in the site collection where we enabled Power Pivot.

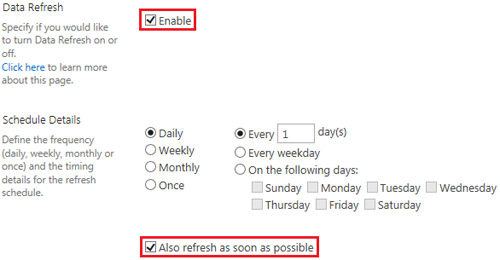
**To create a test workbook**

1. In Excel, on the **Data** tab, choose **From Other Sources**, then choose **From SQL Server**.
2. Type the name of the instance of SQL Server where your data resides.
3. Follow the wizard through to connect to the table that contains your data.
4. When the wizard completes, you should see the **Import Data** dialog box. Choose the **Only Create Connection** option and select the **Add this data to the Data Model** check box.
5. Click **OK**.
6. On the **Power Pivot** tab, click **Manage**.
7. In the **Power Pivot** ribbon, click **PivotTable**.
8. On the **Insert Pivot** dialog box, choose the **Existing Worksheet** option, and then click **OK**.
9. Select the fields that you want in the PivotTable report.
10. Save the workbook to a document library on the site collection where you enabled Power Pivot.

Once you’ve saved the workbook to the document library, you can access the Power Pivot settings by clicking the ellipsis (...) control twice, and then choosing **Manage Power Pivot Data Refresh**.



On the Manage Data Refresh page, select the **Enable** and **Also refresh as soon as possible** check boxes. Also, ensure that the **Use the data refresh account configured by the administrator** option is selected. (This is the default.)



Repeat this several times while making some changes to your data. If the changes show up in the report when it’s rendered in the browser, then data refresh is working properly.

## Summary

In this hands-on Lab, you have learnt how to deploy a SharePoint farm using SQL Server Always On in Azure. In addition, you also learnt how to deploy PowerPivot for SharePoint.

## Additional References

* [Install and configure SharePoint 2013](http://technet.microsoft.com/en-us/library/cc262957.aspx)
* [Overview of AlwaysOn Availability Groups (SQL Server)](http://msdn.microsoft.com/library/ff877884.aspx)
* [Windows Server Failover Clustering (WSFC) with SQL Server](http://msdn.microsoft.com/library/hh270278.aspx)
* [SharePoint on Azure Infrastructure Services](http://msdn.microsoft.com/en-us/library/azure/dn275955.aspx)
* [Install SQL Server BI Features with SharePoint](http://technet.microsoft.com/en-us/library/hh231671.aspx)
* [SQL Server Business Intelligence in Azure Virtual Machines](http://msdn.microsoft.com/library/azure/jj992719.aspx)