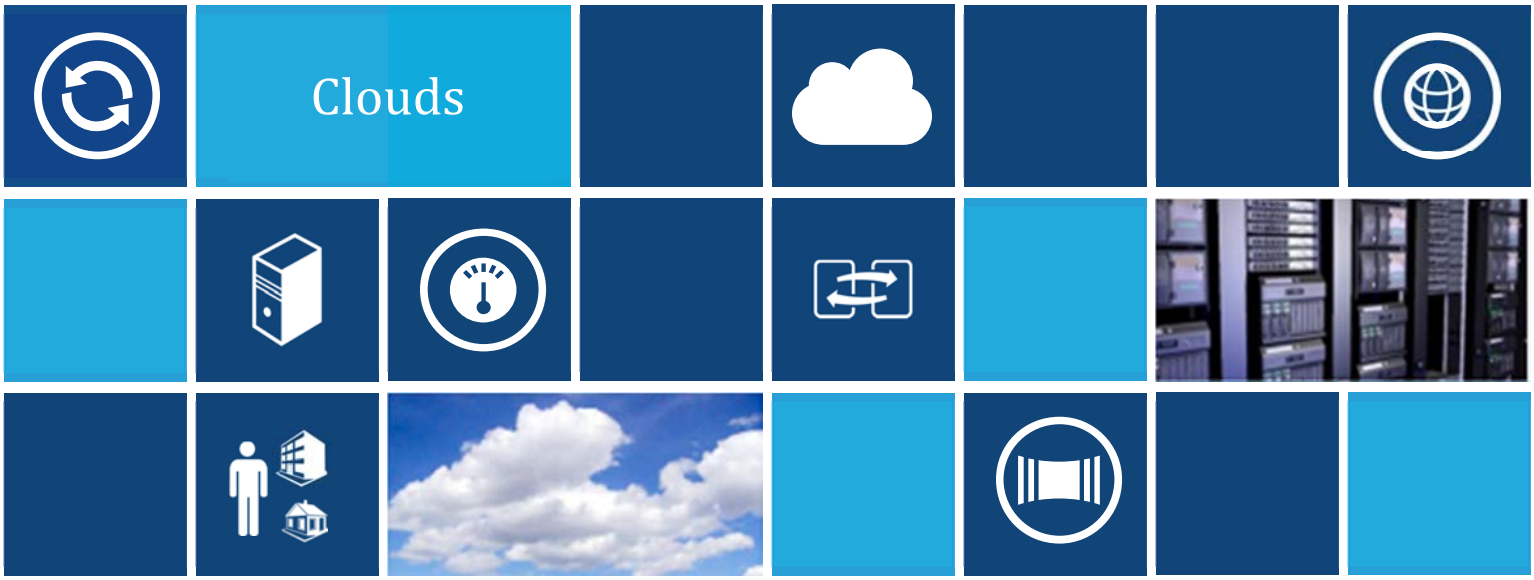
BUILT FOR THE FUTURE, READY NOW.



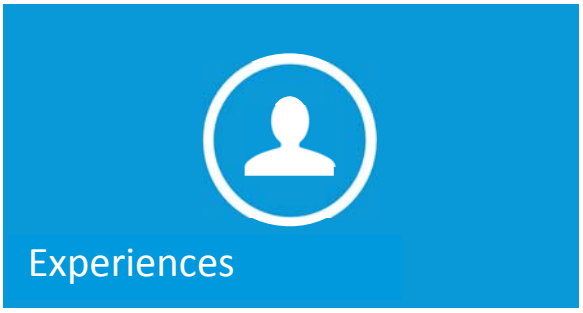
Microsoft Private Cloud: Evaluation Guide

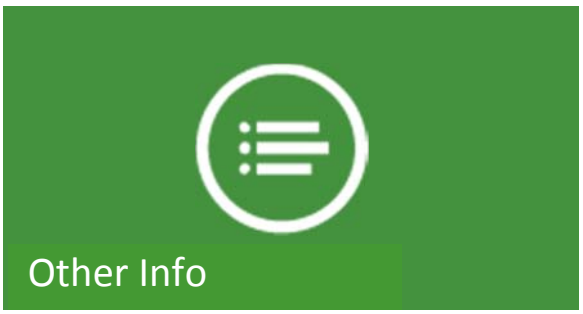
Microsoft System Center 2012 SP1

Windows Server 2012

Copyright Information

Evaluation Guide Contents





Introduction

Using the Evaluation Guide

This Private Cloud Evaluation Guide is built to give you the necessary information to experience the Microsoft Private Cloud built on System Center 2012 SP1 and powered by Windows Server 2012 where you will go through detailed scenarios in a proof of concept environment. This guide is designed to be consumed as you wish and each of the hands-on portions of the guide are designed as "Experiences." Each of the Experiences are independent so that you may build and test each of them in any order you wish.

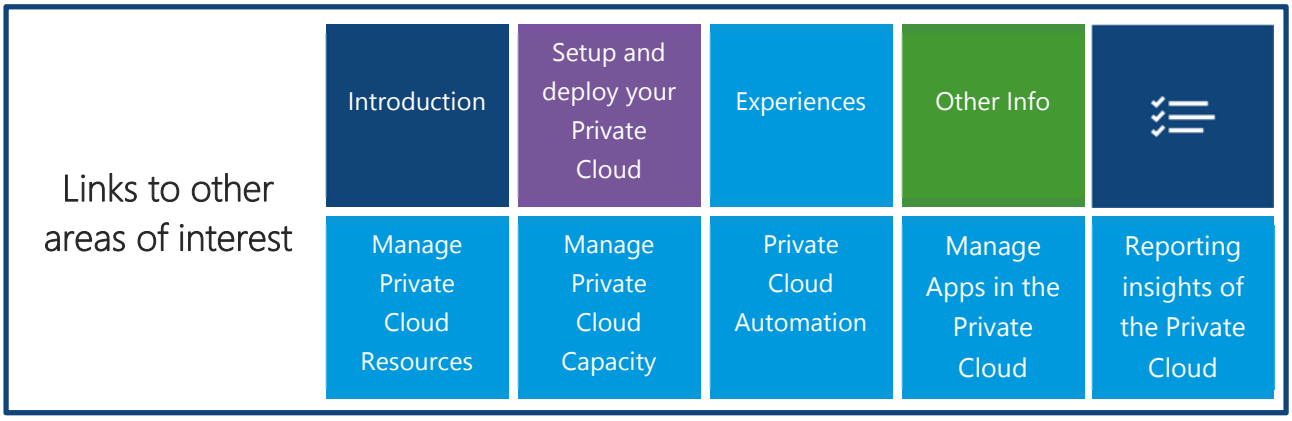
This guide is intentionally built so that you can run it on -lower-level hardware for the purpose of evaluation. It is also possible to follow this guide and change the scenario to your own environment and parameters.

Navigating the document on your terms

This document is designed for you to navigate much easier and locate the information that is most relevant to you. Each of the **4 major chapters – Introduction, Setup and deploy, Experiences and Other info – are collapsible and expandable**.

You will also find that you may jump to any of the other chapters or experiences from the end of any section. The following navigation window allows you to click on the desired tile which links directly to that point in the document. **Simply press "ctrl+click" the tile to view that section**.

 **This tile takes you back to the Evaluation Guide Contents page**.



What is Microsoft's Private Cloud?

The Microsoft Cloud OS

Cloud computing is happening. It is being driven by the exponential growth in modern applications, data and devices. These trends require organizations to quickly scale and deliver continuous services. Simultaneously, innovation is happening across storage, network and compute technologies. With these new requirements and innovations, IT is under pressure to deliver capacity on demand, whether for short-term projects or longer-term initiatives. At the same time, organizations have higher expectations for service levels – services must be always up, always on, with no planned or unplanned downtime.

To meet these challenges, a new modern platform is required. In the era of cloud computing, Microsoft delivers this with the Cloud OS. The Cloud OS takes on a broader role than just managing a server. Microsoft's vision for the Cloud OS is focused on the transformation of the datacenter, enabling modern applications, unlocking insights on any data and empowering people-centric IT.

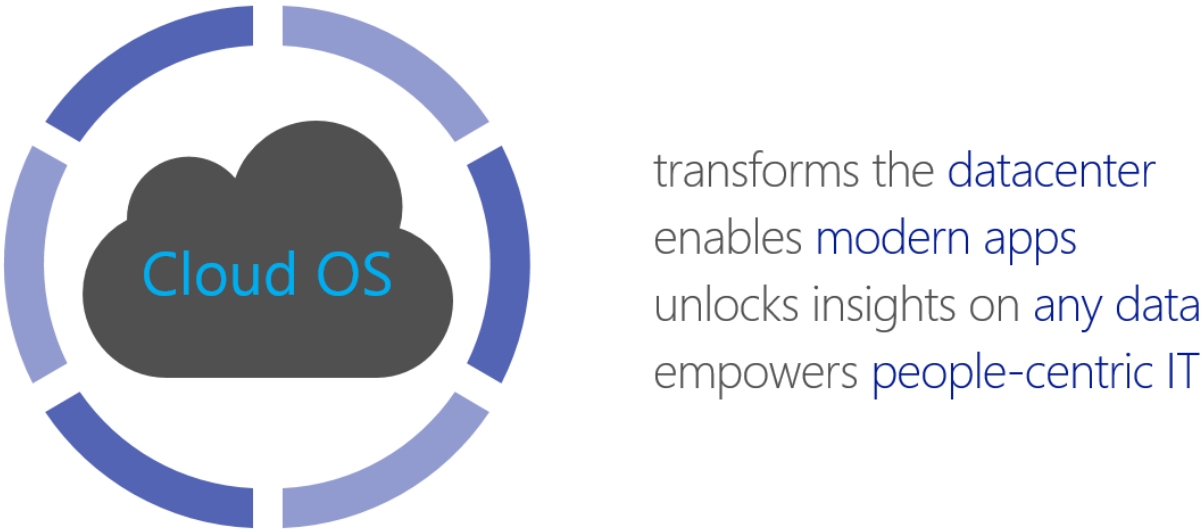


Figure 1: The Cloud OS delivers on four fundamental tenets of the next generation of IT

The Cloud OS requires a comprehensive execution of vision crossing a wide range of enterprise technologies, including Windows Azure, SQL Server, System Center and Windows Server. This whitepaper sets out a step on that journey by focusing on the transformation of the datacenter. Fundamental to the modern datacenter, is the ability to build private cloud infrastructures and quickly scale within the datacenter as well as agile and secure consumption of Windows Azure and service provider cloud capacity. This paper explains Windows Server 2012 as the foundation for the Private Cloud and how System Center 2012 SP1, makes the management of private and public clouds consistent and seamless.

The Private Cloud in your datacenter

A private cloud aims to deliver many of the characteristics of public cloud computing such as scalability and elasticity, the pooling of shared infrastructure, user self-service, and higher levels of overall application availability and reliability. However, by taking a private cloud approach, organizations can deliver on these goals while still using their own physical resources allowing them to maintain complete control over their data and processes.

Figure 2: The private cloud delivers the broad benefits of cloud computing with added levels of control and customization

With shared infrastructure, compute, storage and even networking assets are virtualized and pooled together. This approach is a shift from today's model in which an application owner will likely know (and probably fund) the exact hardware on which their application is physically deployed. In a private cloud environment, application owners will only care that adequate capacity to run the application has been provided from the shared pool. Once deployed, services can scale up and down based on demand. IT no longer needs to over-provision infrastructure capacity to accommodate potential spikes in load.

Datacenter administrators and application owners can view resource utilization as well as the chargeback information related to their allocated capacity. By giving application owners better visibility over their resource usage organizations are able to more easily drive and deliver the right behavior to conserve capacity where possible. A self-service interface to which standardized services are published from central IT, empowers application owners and other internal customers to be able to easily provision resources when needed. Automation helps to drive efficiency in service delivery thereby freeing up IT Pros to focus on other tasks. Self-service also frees up application owners as well. They are empowered to directly access resources; no complicated approval process; no need to wait for the overworked IT team to eventually get to the request.

A private cloud deployment captures all of these characteristics, most of which are also in common with public and partner hosted clouds, but, the private cloud also allows organizations to maintain total control over their applications and data. These are common blockers for businesses looking at public cloud solutions. Ultimately, most customers will find portions of their organization that live best in a private deployment, portions that live best in a public deployment and in some cases portions that are best served by partner delivered clouds. The world will exist in a hybrid state for the foreseeable future and it is important for companies to evaluate their particular needs to determine the best combination for them.

Why System Center 2012 SP1 for your Private Cloud

Over the last twenty years, Microsoft has built some of the world's largest datacenters running many of the world's largest cloud hosted services. The expertise that has been built up in efficiently running these massive data centers forms the foundation for the Microsoft private cloud offering. Built on System Center 2012 SP1 and Windows Server 2012, a Microsoft private cloud allows datacenter administrators to deploy a flexible and responsive infrastructure that is designed to simplify day-to-day tasks and to enable management of applications at the service level, rather than the level of individual servers.

System Center 2012 SP1 makes it easy for customers to build private clouds while taking advantage of currently deployed IT hardware and building an infrastructure for the future. This comprehensive offering was designed around the following principles.

These principles are surfaced through a core set of capabilities in Windows Server 2012 and System Center 2012 SP1, allowing datacenter administrators and application owners to access and consume IT services from these private clouds.

The rest of this paper will describe the private cloud capabilities provided by System Center 2012 SP1. For the datacenter administrator, this includes infrastructure management, service delivery and automation. For the application owner or business unit IT professional, this includes self-service application management, monitoring and reporting.

The table in Figure 3 highlights these capabilities, how they align to industry concepts and System Center 2012 SP1 components.

Customer Scenario	Industry Capability	System Center 2012 Component
Application Management	Application Self-Service	App Controller, Service Manager
	Application Performance Management	Operations Manager
	Application Management Across Clouds	Virtual Machine Manager
Service Delivery & Automation	IT Service Management & Reporting	Service Manager
	Process Automation & Orchestration	Orchestrator
Infrastructure Management	Cloud Creation & Delegation	Virtual Machine Manager
	Data Protection & Disaster Recovery	Data Protection Manager, Orchestrator
	Monitoring	Operations Manager
	Configuration & Compliance	Configuration Manager

Figure 3: Mapping of industry cloud computing concepts to System Center 2012 SP1 products and features.

Components of the Private Cloud

Overview

Highly virtualized computing gives you the benefits of increased utilization of your servers, power savings, and reduced server footprint. We are familiar with these as they are in datacenters today. However, this does not equal the private cloud.

A private cloud provides all of those benefits plus highly integrated and automated management, scalable and elastic platforms, and self-service IT infrastructure.

Through a highly automated infrastructure, an organization can reduce operational costs by automating many tasks that previously required manual intervention. Through the service catalog an organization can provide a self-service IT infrastructure to business units and departments with an SLA. This forces service-level discussion and removes the burden to procure, provision, and manage infrastructure on a per-application, ad-hoc basis. With a scalable and elastic infrastructure, an organization can enable faster delivery of capacity as resource needs change. By utilizing a Windows Server 2012 infrastructure along with System Center 2012 SP1, the Microsoft private cloud allows you to deploy a flexible and responsive infrastructure, designed to simplify day-to-day tasks and enable management of applications at the service level, rather than that of individual servers.

With the release of Service Pack 1, many new features have been made available to increase scalability, elasticity and automation of the Private Cloud. Wider integration with many other products including Windows Azure means more options for IT professionals wishing to take advantage of the Public and Private cloud to future proof their business.

Reference Links

- Download Microsoft Windows Server 2012
- Download Microsoft System Center 2012 SP1
- Download SC2012 Configuration Manager and Endpoint Protection SP1
- What's new in System Center 2012 SP1
- What's new for Orchestrator in System Center 2012 SP1
- What's new for App Controller in System Center 2012 SP1
- What's new for Virtual Machine Manager in System Center 2012 SP1
- What's new for Operations Manager in System Center 2012 SP1
- What's new for Service Manager in System Center 2012 SP1
- What's new for Data Protection Manager in System Center 2012 SP1

Microsoft Windows Server 2012

Microsoft Windows Server 2012 is the operating system that runs your private cloud. In this evaluation guide it will run as the host hypervisor, the OS for the different infrastructure components, and for the application that we will deploy, manage, and monitor.

Hyper-V

Hyper-V is the server virtualization technology within Windows Server 2012. It allows you to run virtual machines on top of the host Operating System when your hardware is capable of running virtualized loads.

System Center Orchestrator

System Center Orchestrator is a workflow engine that allows you to automate the creation, deployment and monitoring of resources in your private cloud. Better said, Orchestrator provides you with orchestration, integration and automation of IT processes. Orchestrator works in conjunction with all the System Center components and can also be used to automate components from 3rd party applications like HP ILO and OA, VMware vSphere and many others (for a full list of Orchestrator 2012 SP1 integration packs see http://technet.microsoft.com/en-us/library/hh295851.aspx).

Included as part of Orchestrator 2012 SP1 is the Service Provider Framework which enables service providers to offer Infrastructure as a Service (IaaS). The provider may have already provided a front-end portal for clients to interact with, and Service Provider Foundation allows those clients access to their resources on the hosting provider's system without any change to the portal.

Components	Description
Management Server	The management server is the communication layer between the Runbook Designer and the orchestration database
Runbook Server	A runbook server is where an instance of a runbook runs. A runbook is a collection of actions bundled together that Orchestrator will run performing various automated actions. Runbook servers communicate directly with the orchestration database. You can deploy multiple runbook servers per Orchestrator installation to increase capacity and redundancy.
Orchestration Database	The database is a Microsoft SQL Server database that contains all of the deployed runbooks, the status of running runbooks, log files, and configuration data for Orchestrator.
Runbook Designer	The Runbook Designer is the tool used to build, edit, and manage Orchestrator runbooks. One runbook or different runbooks together form your workflow(s).
Runbook Tester	Runbook Tester is a run-time tool used to test runbooks developed in the Runbook Designer. This tool allows you to test your runbooks before taking them into production.
Orchestration Console	The Orchestration console lets you start or stop runbooks and view real-time status on a web browser. This is a Silverlight-based web console.

Orchestrator Web Service	The Orchestrator web service is a Representational State Transfer (REST)-based service that enables custom applications to connect to Orchestrator to start and stop runbooks, and retrieve information about operations by using custom applications or scripts. The Orchestration console uses this web service to interact with Orchestrator.
Deployment Manager	Deployment Manager is a tool used to deploy integration packs (IPs), runbook servers, and Runbook Designers
Integration pack (IP)	An integration pack is a collection of custom activities specific to a product or technology. Microsoft and other companies provide integration packs with activities to interact with their product from an Orchestrator runbook.
Orchestrator Integration Toolkit	The Orchestrator Integration Toolkit lets you extend your library of activities beyond the collection of standard activities and integration packs. The Integration Toolkit has wizard-based tools to create new activities and integration packs for Orchestrator. Developers can also use the Integration Toolkit to create integration packs from custom activities that they build by using the Orchestrator SDK.

New Capabilities
• Exchange Administrator Integration Pack for Orchestrator in System Center 2012 SP1 • Exchange Users Integration Pack for Orchestrator in System Center 2012 SP1 • Representational State Transfer (REST) Integration Pack Guide for Orchestrator in System Center 2012 SP1

System Center App Controller

System Center App Controller is a Silverlight web-based interface that allows you to manage, build, configure and deploy services both on the private and the public cloud. With this interface, you will have a common self-service experience through your different clouds. This interface is mainly used to provide self-service capabilities for your application owners.

The ability to control and manage applications and services within the private cloud is critical. A key requirement, as organizations begin using hybrid apps, will be the ability to connect with services in other clouds and to manage them through a single management experience.

Components	Description
App Controller Server	The App Controller Server runs the web-based Silverlight application to manage, build, configure and deploy services both on your private cloud and the public cloud.
Database	The database that contains the necessary information for the connection to your Azure subscriptions and your Virtual Machine Manager service(s).
PowerShell Module	The App Controller PowerShell Module provides administrators with the ability to automate App Controller administration.

New Capabilities
• Upload a virtual hard disk or image to Windows Azure from a VMM library or network share • Migrate a virtual machine from VMM to Windows Azure • Add a Service Provider Framework (SPF) hosting provider connection. SPF enables service providers to offer Infrastructure as a Service (IaaS) to their clients.

System Center Virtual Machine Manager

System Center Virtual Machine Manager is the component that provides you with virtual machine management AND service deployment. This component comes with support for multi-hypervisor environments and is the system that allows you to define, create and manage your private cloud environment(s).

Components	Description
VMM Management Server	The computer on which the Virtual Machine Manager service runs and which processes commands and controls communications with the VMM database, the library server, and virtual machine hosts.
VMM Console	The Virtual Machine Manager Console is the graphical user interface to your VMM environment.
VMM Self-Service Portal	A website used to deploy and request virtual machines.
VMM Database	The VMM database stores all the Virtual Machine Manager configuration and information regarding the hosts and virtual guests.
VMM Library Server	The VMM Library Server is a catalog of resources containing all the ISO files, virtual hard disks, templates and profiles used to deploy virtual machines and services.

New Capabilities
• Support for Network Virtualization with the support for using DHCP to assign customer IP addresses • Support for VHDX format with functionality to convert from VHD to VHDX • Support for file shares using SMB 3.0

System Center Operations Manager

System Center Operations Manager provides you with deep application diagnostics and infrastructure monitoring of your private cloud components. It can offer you a thorough overview of the performance and availability of applications deployed in your datacenter, private or public cloud.

Components	Description
Management Server	The Operations Manager Management Server is the focal point for administering the management group and communicating with the database. When you open the Operations console and connect to a management group,

	you connect to a management server for that management group. Depending on the size of your computing environment, a management group can contain a single management server or multiple management servers.
Operations Console	The Operations Manager is the graphical user interface that will give you the single pane of glass monitoring of your private cloud.
Operations Manager Database	The OpsMgr database is where all the collected data like performance & event data, alerts etc. are stored.
Operations Manager Data Warehouse	The OpsMgr data warehouse is used for long term reporting.
Operations Manager Web Console	The OpsMgr web console provides a browser-based alternative to the OpsMgr console.
Operations Manager Advisor	Browser-based console to provide deep insight in your .Net applications
Gateway Server	A gateway server is used to monitor untrusted environments like a DMZ Audit Collection Server ACS is used for collecting and auditing security events.

New Capabilities

The diagnostic and monitoring capabilities have been expanded in System Center 2012 Operations Manager SP1. New features include:

- Monitoring Windows Services built on the .NET Framework
- Automatic discovery of ASP.NET MVC3 and MVC Applications
- Enabled APM of SharePoint 2010
- Integration with Team Foundation Server 2010 and 2012

System Center Service Manager

Service Manager provides an integrated platform for automating and adapting your organization's IT service management best practices, such as those found in Microsoft Operations Framework (MOF) and Information Technology Infrastructure Library (ITIL). It provides built-in processes for incident and problem resolution, change-control, and release management.

Components	Description
Management Server	Contains the main software part of a Service Manager installation. You can use the Service Manager management server to manage incidents, changes, users, and tasks.
Database	The database that contains Service Manager configuration items (CI) from the IT Enterprise; work items, such as incidents, change requests, and the configuration for the product itself. This is the Service Manager implementation of a Configuration Management Database (CMDB).
Data warehouse management server	The computer that hosts the server piece of the data warehouse.

Data warehouse databases	Databases that provide long-term storage of the business data that Service Manager generates. These databases are also used for reporting.
Service Manager console	The user interface (UI) piece that is used by both the help desk analyst and the help desk administrator to perform Service Manager functions, such as incidents, changes, and tasks. This part is installed automatically when you deploy a Service Manager management server. In addition, you can manually install the Service Manager console as a stand-alone part on a computer.
Self-Service Portal	A web-based interface into Service Manager.

New Capabilities
• Chargeback helps you can apply cloud-based pricing to your Virtual Machine Manager fabric • Improved Operations Manager Integration • SQL Server 2012 Support • Windows Server 2012 and Windows 8 Support

System Center Data Protection Manager

Data Protection Manager (DPM) enables disk-based and tape-based data protection and recovery for servers such as SQL Server, Exchange Server, SharePoint, virtual servers, file servers, and support for Windows desktops and laptops. DPM can also centrally manage system state and Bare Metal Recovery (BMR) which will allow us to protect the servers that contain our virtualized infrastructure.

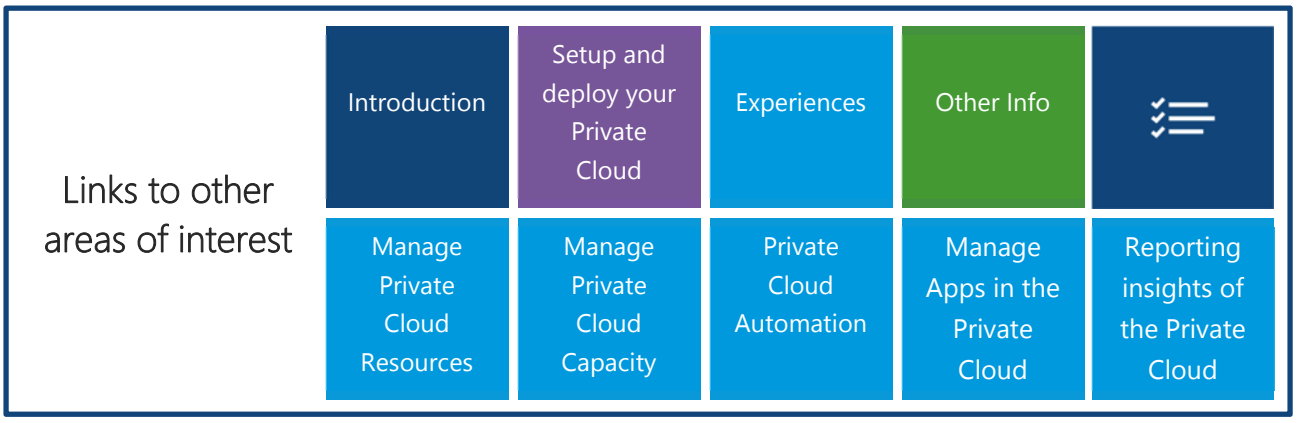
Components	Description
DPM server	The DPM server contains the program files of the Data Protection Manager installation. This server will be responsible for all the protection and recovery jobs
Database	The DPM database will contain all the information of your Data Protection Manager environment. All protection group information, agent information, recovery points and so on are stored here.
Central Console	Operations Management is used as the platform for the central console and is used to manage your SCDPM server (or multiple DPM servers)
Storage Pool	The Storage Pool contains your storage that will be used to take disk-to-disk backups

New Capabilities

- Cluster Shared Volume (CSV) 2.0 Support for improved performance on backing up virtual machines on CSVs.
- Hyper-V protection over remote SMB Shares.
- Scale Out support for Hyper-V machines.
- Protection of Windows 8 de-duplicated volumes.
- Support for Live Migration.
- Integration with Windows Azure Online Backup.
- Support for SQL 2012 Always-On feature.
- Support for Resilient File System (ReFS).

Summary

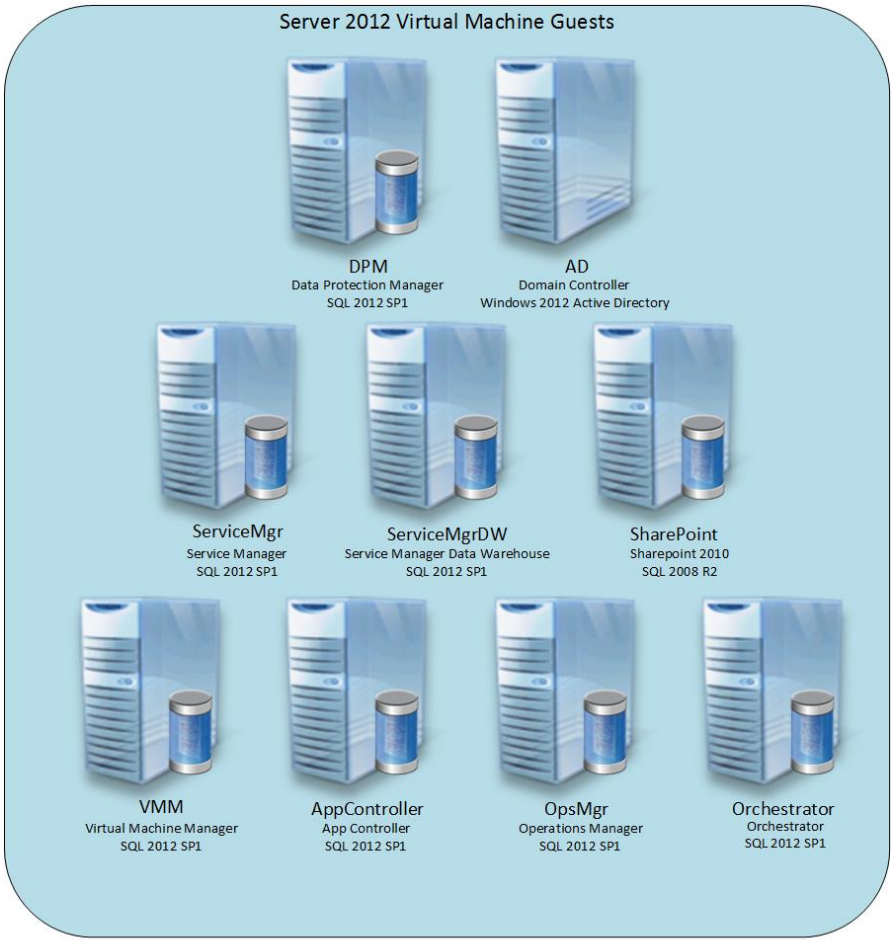
In this first chapter, you reviewed the components that are needed to deploy a private cloud. Now you know all the building blocks to start with your evaluation of the Private Cloud.



Setup and deploy your Private Cloud

Architecture

Here is the overview of what hardware we are going to use for our private cloud evaluation. This can be different from your environment but this is the minimum required to evaluate the Private Cloud.



Installing the Private Cloud

Implementing or upgrading your Private Cloud

This document covers two situations where you may be deploying System Center 2012 SP1 into a Lab or sandbox environment for evaluation of the Private Cloud. This may be your first time evaluating System Center 2012 and Windows Server 2012, so you have an environment that is empty. Or this may be a follow up from a previous evaluation using the Private Cloud Evaluation guide that was released with the System Center 2012 RTM package.

In future there will also be an option to install the full System Center 2012 Suite using the Microsoft Cloud Accelerator Toolkit.

Both of these situations can be covered in the Install and Upgrade sections below. If you wish to install a Private Cloud environment from a blank environment, follow the steps outlined below in the Installing the Private Cloud section. If you wish to upgrade an existing Private Cloud evaluation environment delivered via the Unified Installer or the previous Private Cloud Evaluation guide, then follow the steps outlined in the section Upgrading the Private Cloud.

If you have previously deployed the Private Cloud either using the previous Evaluation Guide or using the Unified Installer that was released with System Center 2012 RTM, you may have less hardware than the diagram above. To upgrade an RTM environment to SP1 and implement the scenarios in later sections of the document you will need 2 additional hosts, and 2 network cards per host. Each server must have Windows Server 2012 installed, and the Hyper-V role enabled.

More detail on setting up the hardware can be found in the section labeled Upgrading the Private Cloud.

Prerequisites 

Physical servers

The following minimum physical servers are required:

Name	CPU	Memory	Disk	NIC's	Main IP	Cluster IP
HyperV02	4	16 GB	1TB	2 Physical NICs	192.168.1.2	172.24.100.2
HyperV03	4	16 GB	1TB	2 Physical NICs	192.168.1.21	172.24.100.21
HyperV04	4	16 GB	1TB	2 Physical NICs	192.168.1.22	172.24.100.22

All servers have a subnet mask of 255.255.255.0 on both network adapters.

Virtual servers

The following virtual servers are required:

Name	Function	CPU	Memory	Disk	IP
Orchestrator	Orchestrator server	2 vCPU	4 GB	60GB	192.168.1.3
VMM	Virtual Machine Manager server	2 vCPU	4 GB	90GB	192.168.1.4
AppController	App Controller server	2 vCPU	4 GB	60GB	192.168.1.5
OpsMgr	Operations Manager server	2 vCPU	4 GB	60GB	192.168.1.6
ServiceMgr	Service Manager server	2 vCPU	4 GB	60GB	192.168.1.7
ServiceMgrDW	Service Manager Data Warehouse server	2 vCPU	4 GB	60GB	192.168.1.8
DPM	Data Protection Manager server	2 vCPU	4 GB	60GB	192.168.1.9
AD	Domain Controller	2 vCPU	2 GB	60GB	192.168.1.10
SharePoint	SharePoint 2010 Server	2 vCPU	4 GB	60GB	192.168.1.11

When you are going to deploy the infrastructure and work through the different exercises, you can use your own IP addresses and better hardware. If you work with different IP ranges, you need to take into account that for some of the exercises, you will also need to change the ranges.

Installing the Private Cloud – fresh install

Requirements

If you are installing the Private Cloud with Server 2012 and System Center 2012 SP1 with a blank environment you will require the following prerequisites:

Item	Detail
Hardware	3 Hosts meeting the minimum specification as detailed in the Architecture section
Operating Systems	Windows Server 2012 http://www.microsoft.com/en-us/server-cloud/windows-server/trial.aspx Windows Server 2008 R2 SP1 http://www.microsoft.com/en-us/download/details.aspx?id=11093
Software	System Center 2012 SP1 http://www.microsoft.com/en-us/download/details.aspx?id=34607 (BETA) SQL 2012 SP1 http://www.microsoft.com/betaexperience/pd/SQL2012EvalCTA/enus/default.aspx SQL 2008 R2 SP2 http://www.microsoft.com/en-us/download/details.aspx?id=6362

Each product within the System Center suite and SQL also have their own prerequisites. Please refer to the documentation for each product for guidance on installation.

Reference Links

- Windows Server
  - Download Windows Server 2012
  - Installing Windows Server 2012
  - How to Install Active Directory Domain Services
- SQL Server
  - Download Microsoft SQL Server 2012 SP1
  - How to install SQL Server 2008 R2
  - Quick Start Installation of SQL Server 2012 SP1
- System Center
  - Download System Center 2012 SP1
  - Install System Center 2012 SP1
  - How to Install Virtual Machine Manager Management Server
  - Single-Server Deployment of Operations Manager
  - Installing Service Manager on Two Computers
  - Self-Service Portal Deployment for System Center 2012 Service Manager
  - How to Install Orchestrator on a Single Computer
  - Installing App Controller
  - Installing Data Protection Manager

Installation process

Once you have the prerequisites downloaded the installation process can commence. Follow each of the steps as detailed in the sequence below.

Install Windows Server 2012 hosts

Install Windows Server 2012 on the two physical hosts.

Once the hosts are built complete the following actions on each host:

- Assign the IPv4 addresses as per the table in the Private Cloud Architecture section in this document
- Rename the hosts to match the naming convention in the Private Cloud Architecture section in this document
- Add the Hyper-V Role (see appendices for detailed instructions)

Once these steps are complete, configure the Virtual Network on each host. It is important that each host has identical Virtual Network settings. To configure the Virtual Network, follow the instructions below on each host:

1. Open the Hyper-V Manager console
2. Click on Connect to Server in the right hand pane
3. Type localhost and click OK
4. Right click on the Host name and select Virtual Switch Manager
5. Select New Virtual Network Switch, click on External and click Create Virtual Switch
6. Set the Name to "External Virtual Network"
7. Ensure the physical network adapter is selected and the tick box *allow management operating system to share this network adapter* is ticked.
8. Click OK.

Once the Hosts are configured, complete these steps:

1. Create a virtual machine for each of the servers as detailed in the Private Cloud Architecture section, ensuring the settings match the table. (See Appendices for instructions on creating a virtual machine). Ensure when you create these virtual machines that you distribute the machines across the 3 physical hosts HyperV02, HyperV03 and HyperV04.
2. Install Windows Server 2012 on every virtual machine, except the SharePoint server
3. Install Windows Server 2008 R2 on the SharePoint server
4. Configure each virtual machine as per the table in the Private Cloud Architecture section in this document.

5. Configure the network settings on each virtual machine as per the table in the Private Cloud Architecture section. See the appendices for steps on setting the IPv4 address on a Windows 2012 server.
6. Prepare Active Directory on the server "AD".
7. Add all servers to the new domain. See the appendices for steps on how to add a Windows 2012 server to the domain.

Once these steps are complete, the lab you have constructed will be ready for deploying System Center 2012.

Group policies

To make the evaluation easier, we've created a Group Policy on domain level to enable and disable a few settings. By default we disabled all Windows Firewalls on the server machines and enabled remote desktop for easy access. It is not necessary to disable Windows Firewall and you can review the requirements for Windows Firewall on the different TechNet pages for the components.

Passwords

All passwords in this evaluation guide are the same. For every account, we are using the password: "pass@word1"

Installing System Center 2012 SP1

Now that the lab is ready for System Center 2012, you can start deploying the different products in the suite on each of the virtual machines in the lab. Install each product in the following order:

- Virtual Machine Manager
- Operations Manager
- Service Manager
- Orchestrator
- App Controller
- Data Protection Manager

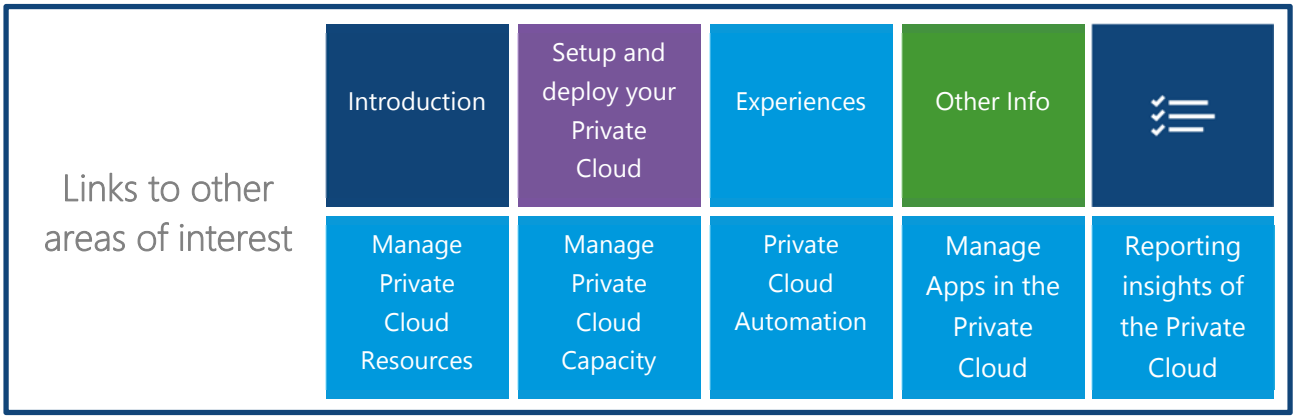
You'll notice below that all items that should be installed on the virtual machine with that product have been included. If you've already installed that product on the virtual machine, you may skip installing it again.

Virtual Machine Installation Guidance	
Virtual Machine Manager	1. Install SQL 2012 SP1 on the virtual machine designated for Virtual Machine Manager as per the installation instructions found here http://technet.microsoft.com/en-us/sqlserver/ff898410.aspx 2. Install System Center 2012 Virtual Machine Manager SP1 as per the installation instructions found here http://technet.microsoft.com/en-us/library/gg610669.aspx.

Operations Manager	1. Install SQL 2012 SP1 on the virtual machine designated for Operations Manager as per the installation instructions found here http://technet.microsoft.com/en-us/sqlserver/ff898410.aspx 2. Install System Center 2012 Operations Manager SP1 as per the installation instructions found here http://technet.microsoft.com/en-us/library/hh298609.aspx.
Service Manager	1. Install SQL 2012 SP1 on the virtual machine designated for Service Manager as per the installation instructions found here http://technet.microsoft.com/en-us/sqlserver/ff898410.aspx. 2. Install SQL 2012 SP1 on the virtual machine designated for Service Manager Data Warehouse using the same instructions. Ensure that as well as the Database engine, that the Reporting Services and Analysis Service are also installed with SQL 2012. 3. Install SQL 2008 R2 on the virtual machine designated for SharePoint as per the installation instructions found here http://technet.microsoft.com/en-us/sqlserver/ff398089.aspx 4. Install System Center 2012 Service Manager SP1 as per the installation instructions found here http://technet.microsoft.com/en-us/library/hh305220.aspx. Use the two-computer scenario. 5. Install the Self Service Portal for System Center 2012 Service Manager SP1 using the instructions found here http://technet.microsoft.com/en-us/library/hh495575.aspx.
Orchestrator	1. Install SQL 2012 SP1 on the virtual machine designated for Orchestrator as per the installation instructions found here http://technet.microsoft.com/en-us/sqlserver/ff898410.aspx. 2. Install System Center 2012 Orchestrator SP1 as per the installation instructions found here http://technet.microsoft.com/en-us/library/hh420337.aspx.
App Controller	1. Install SQL 2012 SP1 on the virtual machine designated for App Controller as per the installation instructions found here http://technet.microsoft.com/en-us/sqlserver/ff898410.aspx. 2. Install System Center 2012 App Controller SP1 as per the installation instructions found here http://technet.microsoft.com/en-us/library/jj871059.aspx.
Data Protection Manager	1. Install System Center 2012 Data Protection Manager SP1 as per the installation instructions found here http://technet.microsoft.com/en-us/library/hh758153.aspx. 2. DPM has SQL 2008 R2 as part of the product download and this will be installed during the install phase.

Summary

Once you have completed each of the installations as documented above you will have a Private Cloud environment ready to perform the scenarios available in this document. Once you are ready to start these scenarios, please continue to the Scenario Prerequisite section.



Upgrade your existing Private Cloud to SP1

If you have previously deployed System Center 2012, and wish to reuse that environment to re-evaluate the Private Cloud with System Center 2012 SP1, this is entirely possible. To accomplish this you will need to complete several tasks to take the environment to a state where the scenario information in the subsequent sections is possible to complete.

This upgrade process presumes that the environment you are currently running is built using the previous version of the Private Cloud Evaluation Guide that was released with the RTM of System Center 2012. Before starting this process, ensure that all of the latest Windows Updates and Service Packs are applied to the environment. Specifically ensure that all System Center 2012 installations are updated to Update Rollup 2.

Upgrade the hardware

In the previous Evaluation Guide there was a requirement for a single physical host. This is now increased to three physical hosts with the following minimum specification:

Name	CPU	Memory	Disk	Network Cards	Main IP
HyperV02	4	16 GB	1TB	1 Physical NIC	192.168.1.2
HyperV03	4	16 GB	1TB	1 Physical NIC	192.168.1.21
HyperV04	4	16 GB	1TB	1 Physical NIC	192.168.1.22

To upgrade the hardware, add the two new physical hosts into your Evaluation Lab environment, and build them using Windows Server 2012. Install Windows Server 2012 on the two physical hosts.

Important Note

Note that each server requires two physical network cards to complete the scenarios detailed in the later sections of this document.

Once they have been built, set the IP addresses as per the above table and add them to the domain you have currently in your lab environment. See the Appendices for details on how to edit a Windows Server 2012 server IP settings and add it to a domain.

Once they are built, add the Hyper-V role to each of the Windows Server 2012 hosts and set the Virtual Network settings. It is important that each host has identical Virtual Network settings. To configure the Virtual Network, follow the instructions below on each host:

1. Open the Hyper-V Manager console
2. Click on Connect to Server in the right hand pane
3. Type localhost and click OK
4. Right click on the Host name and select Virtual Switch Manager
5. Select New Virtual Network Switch, click on External and click Create Virtual Switch
6. Set the Name to External Virtual Network

7. Ensure the physical network adapter is selected and the tick box *allow management operating system to share this network adapter* is ticked.
8. Click OK.

Reference Links

- Description of Update Rollup 2 for System Center 2012
- Installing Microsoft Windows Server 2012
- Evaluating Microsoft Windows Server 2012

Upgrade System Center 2012 to SP1

Once the new Windows 2012 Hosts are in place, it's time to start upgrading the Private Cloud to System Center 2012 SP1. This must be completed in a particular sequence and this sequence is documented in the "Upgrade Sequencing for System Center 2012 Service Pack 1" found http://technet.microsoft.com/en-us/library/jj628191.aspx. The Installation Guidance below follows this sequence.

Installation Guidance	
Orchestrator	To upgrade Orchestrator to SP1 follow the document "Upgrading System Center 2012 - Orchestrator to System Center 2012 SP1" which can be found http://technet.microsoft.com/en-US/library/jj900231.aspx.
Service Manager	To upgrade Service Manager to SP1 follow the document "Upgrading to System Center 2012 - Service Manager" which can be found http://technet.microsoft.com/en-us/library/hh519584.aspx.
Data Protection Manager	To upgrade Data Protection Manager to SP1 follow the document "Upgrading from System Center 2012 - Data Protection Manager" which can be found http://technet.microsoft.com/en-us/library/jj650934.aspx.
Operations Manager	To upgrade Operations Manager to SP1 follow the document "Upgrading System Center 2012 - Operations Manager to Operations Manager in System Center 2012 Service Pack 1" which can be found http://technet.microsoft.com/en-us/library/jj899854.aspx.
Virtual Machine Manager	To upgrade Virtual Machine Manager to SP1 follow the document "Upgrading to VMM in System Center 2012 SP1" which can be found http://technet.microsoft.com/en-us/library/jj870890.aspx.
App Controller	To upgrade App Controller to SP1 follow the document "Deploying App Controller in System Center 2012 Service Pack 1 (SP1)" which can be found http://technet.microsoft.com/en-us/library/jj871059.aspx.

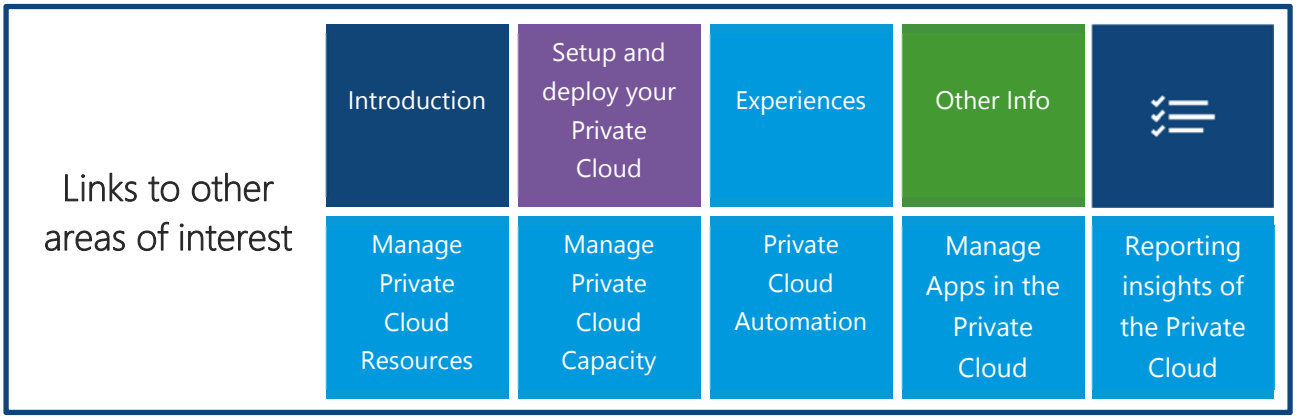
Physical hosts

Once the System Center 2012 suite has been upgraded to SP1, migrate each of the virtual machines currently running on HyperV02 to HyperV03 and HyperV04. Once these virtual machines have been removed from HyperV02, rebuild the HyperV02 host to Windows Server 2012.

Summary

Once these upgrades have been completed you should have the following:

- 3 physical hosts with Windows Server 2012
- 7 Virtual Machines with System Center 2012 SP1



Configuring the Private Cloud experiences

This document follows a series of scenarios that a typical business would perform to implement and take advantage of a Server 2012 and System Center 2012 SP1-based private cloud. In these scenarios we will use the personas of Jeff and Debbie.

Jeff is the Data Center admin for Contoso Electronics, a small firm that is growing rapidly beyond its current infrastructure. Increased sales and market share means the current IT infrastructure cannot cope with the requirements for the business, and Jeff needs to take advantage of many of the features of the private cloud to increase the ability of the companies' infrastructure to respond to growing demand.

Debbie is a Business Unit owner and leads a team of developers responsible for delivering new retail applications to the business. She maintains several environments for developing the new applications and requires the infrastructure to be made available at short notice when new projects are provisioned. She also needs to keep an eye on the cost of these environments as each project must be charged on the resources they consume.

During these scenarios, these people will be involved in decision making and taking advantage of the new capabilities of System Center 2012 SP1, and Server 2012.

Prerequisites

The following prerequisites are required to be in place before you can start the evaluation of the Microsoft private cloud solution. All of these steps will take place on **HyperV02**, **HyperV03** or **Hyperv04**. It is assumed that the **Hyper-V** role has already been added to these hosts, that they have been added to the **Contoso.com**, and they have the same **IP address** as outlined in the hardware recommendations in the previous chapter.

Required software for scenarios

The following software will be required for to complete the following stories. The software can be obtained from the Microsoft website and links are provided.

- Windows Server 2012

Microsoft Windows Server 2012 Evaluation **VHD**. This is required for the build of **Guest01** and should be downloaded and extracted to the folder **C:\VirtualMachines** on **HyperV02**.

http://technet.microsoft.com/en-nz/evalcenter/hh670538.aspx

- SQL Server 2012 SP1

Microsoft SQL Server 2012 with SP1 Evaluation. This is require for the build of **Guest01** and should be downloaded and extracted to the folder **C:\VirtualMachines** on **HyperV02**.

http://technet.microsoft.com/en-us/evalcenter/hh225126.aspx

- SysInternals Tools

Microsoft Sys Internals VHDTool. This is required for the generation of a large VHD on **Guest01**. To make the file easily accessible to **Guest01**, the file should be stored on the SMB 3.0 share **\\HyperV02\SMB3Share** on **HyperV02**

http://archive.msdn.microsoft.com/vhdtool

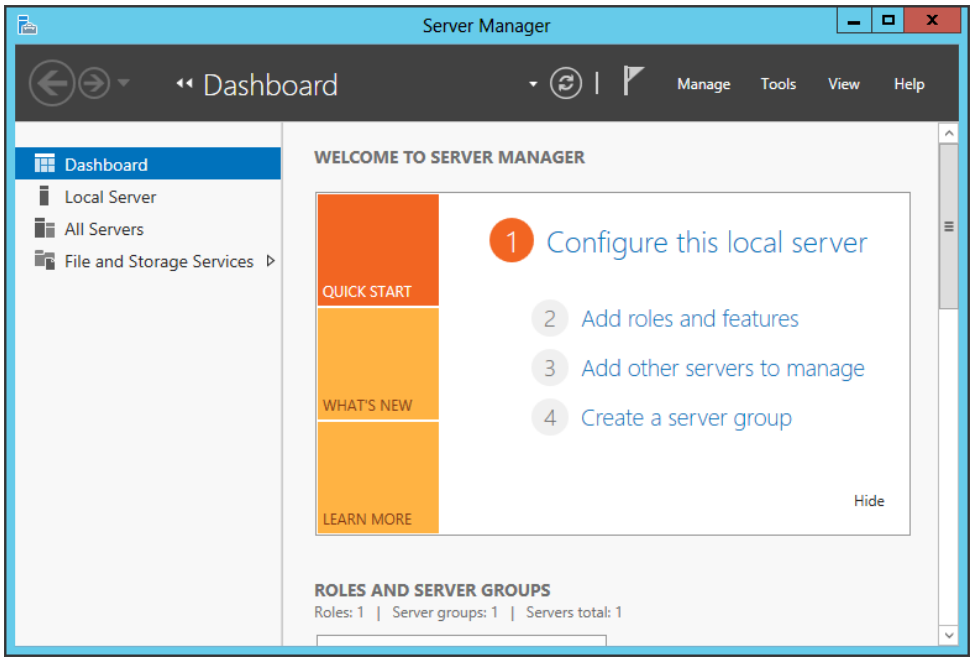
Configuring a SMB 3.0 share

Before **Jeff** can implement a highly available private cloud solution he is going to need a commonly accessible storage platform that can host his highly available services. There are numerous ways that this can be achieved but for the purposes of his evaluation Jeff would like to use a new feature in **Microsoft Windows Server 2012.** That feature is **SMB 3.0** and is available as an installable feature.

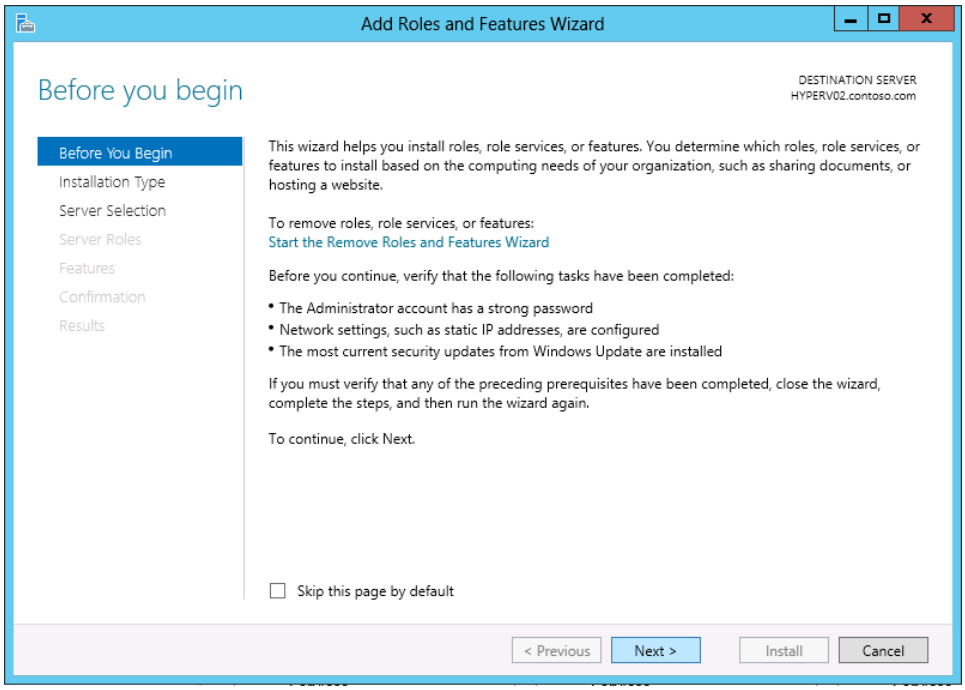
The following steps should be completed on the server **HyperV02**

Create a SMB 3.0 share

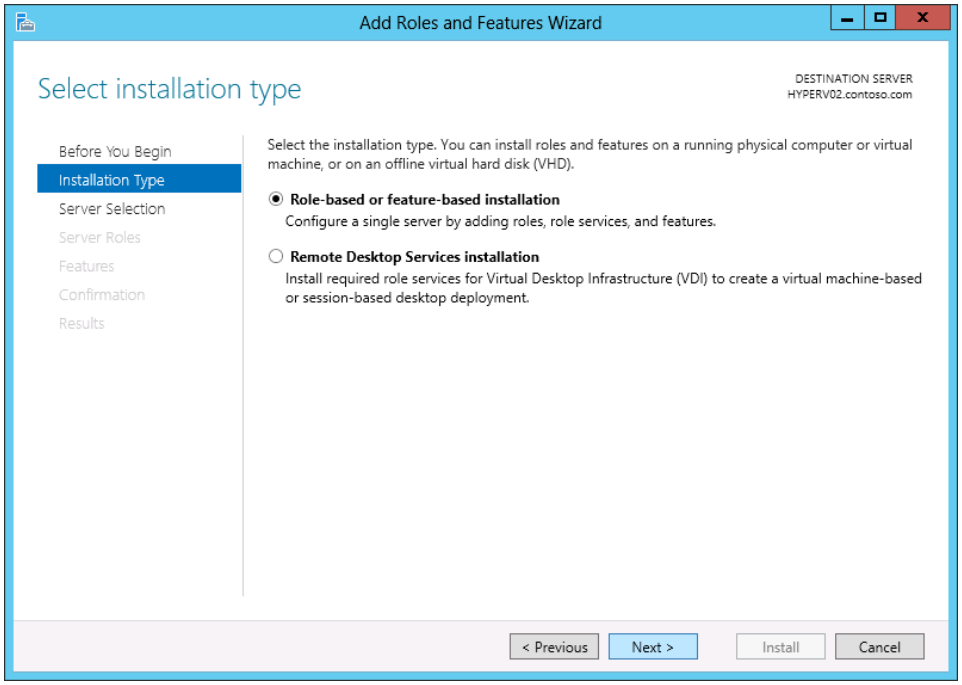
1. Open Server Manager, confirm that the Dashboard is open and select Add Roles and Features



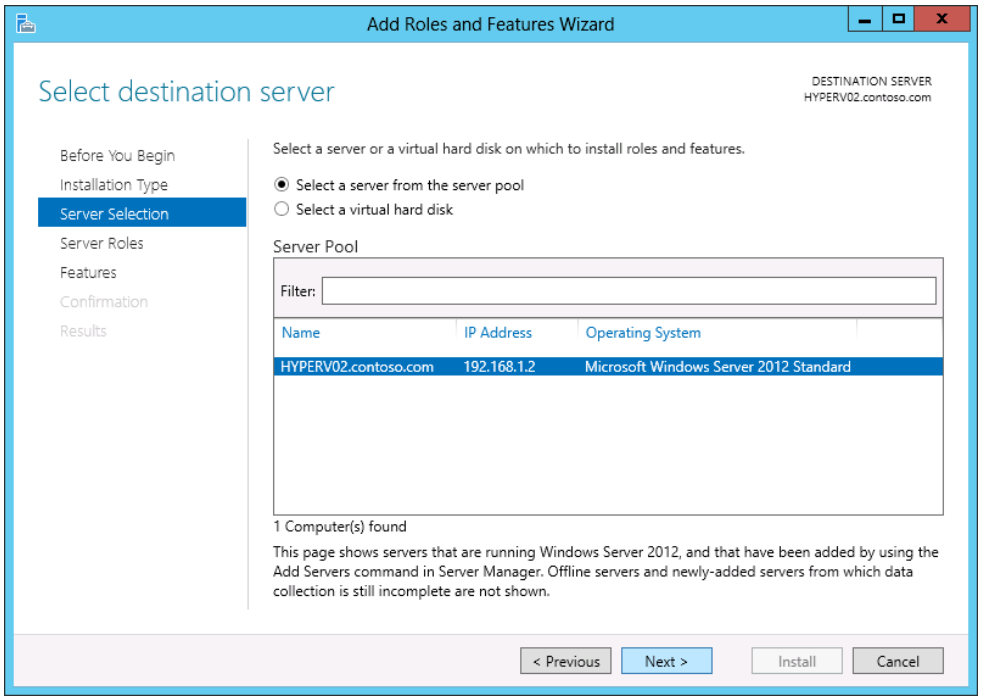
1. In the **Before You Begin** screen, select **Next**



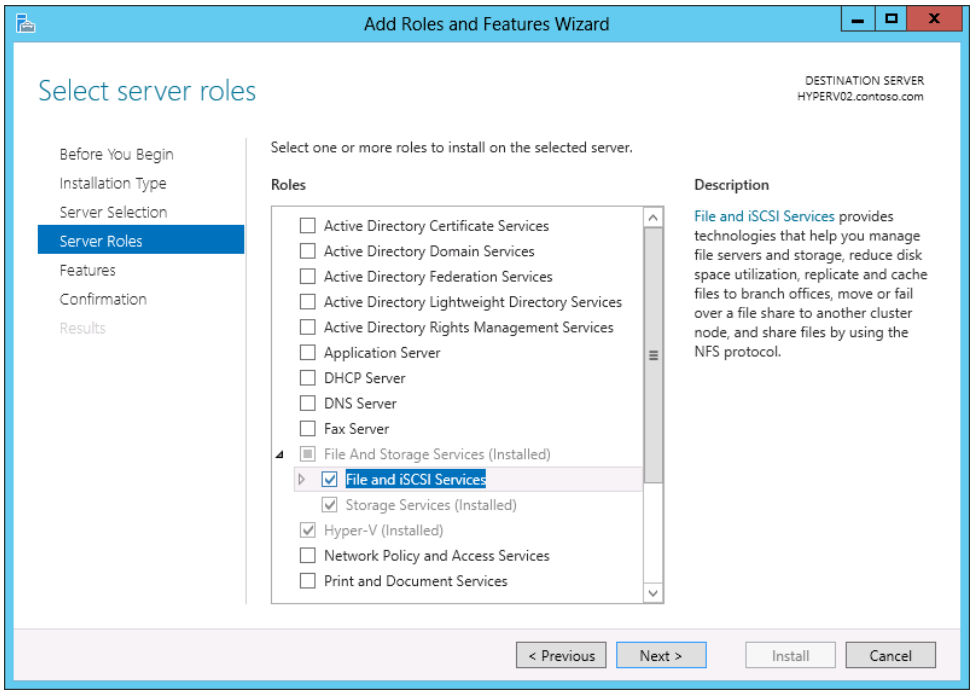
2. In the Installation Type screen, tick Role-based or feature-based installation. Then select Next



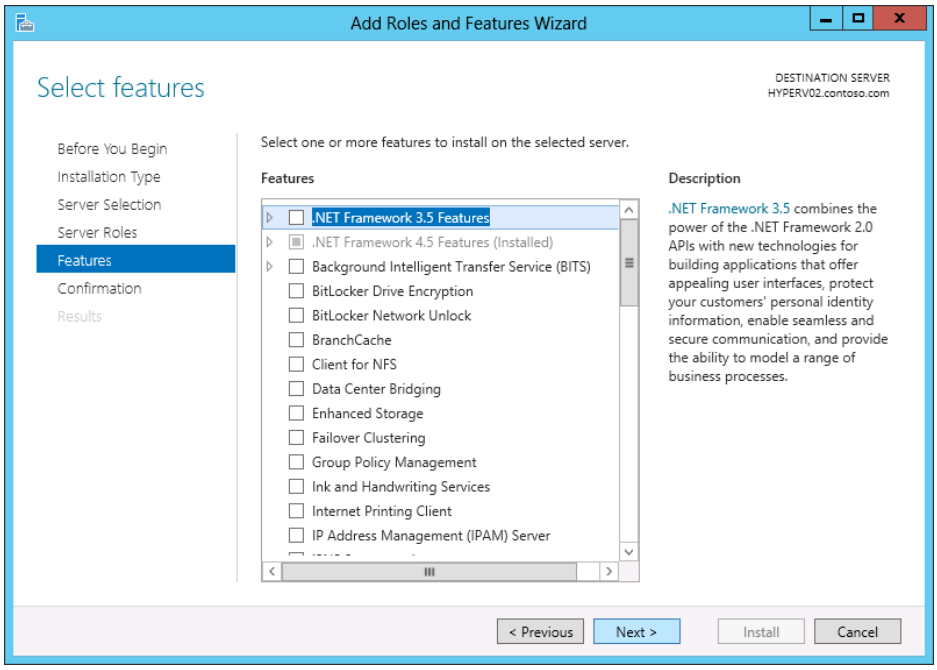
3. In the Server Selection screen select HyperV02 from the Server Pool, select Next



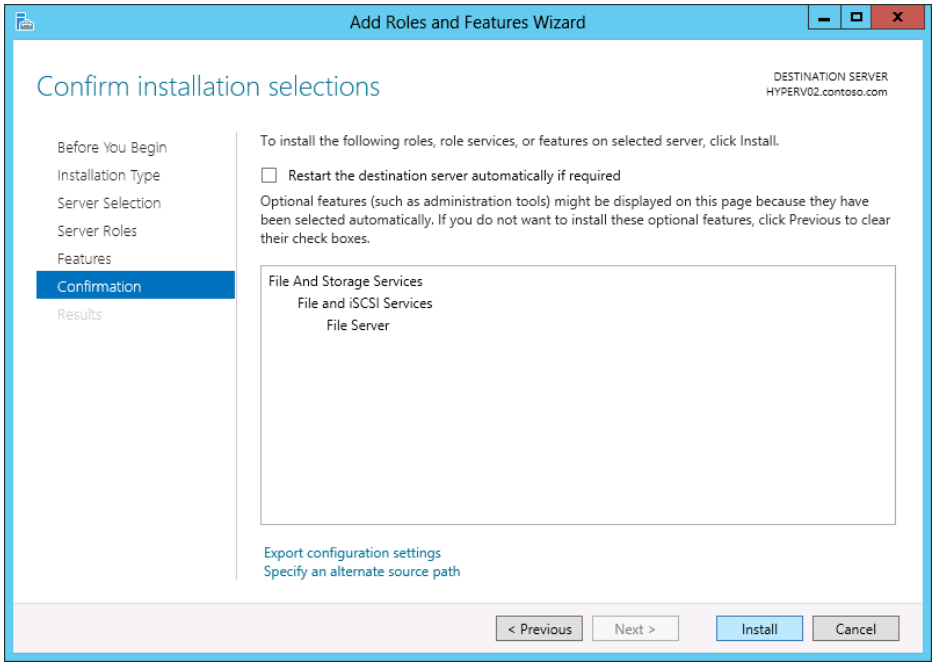
4. In the Server Roles screen, expand File and Storage Services, tick File and iSCSI Services, select Next



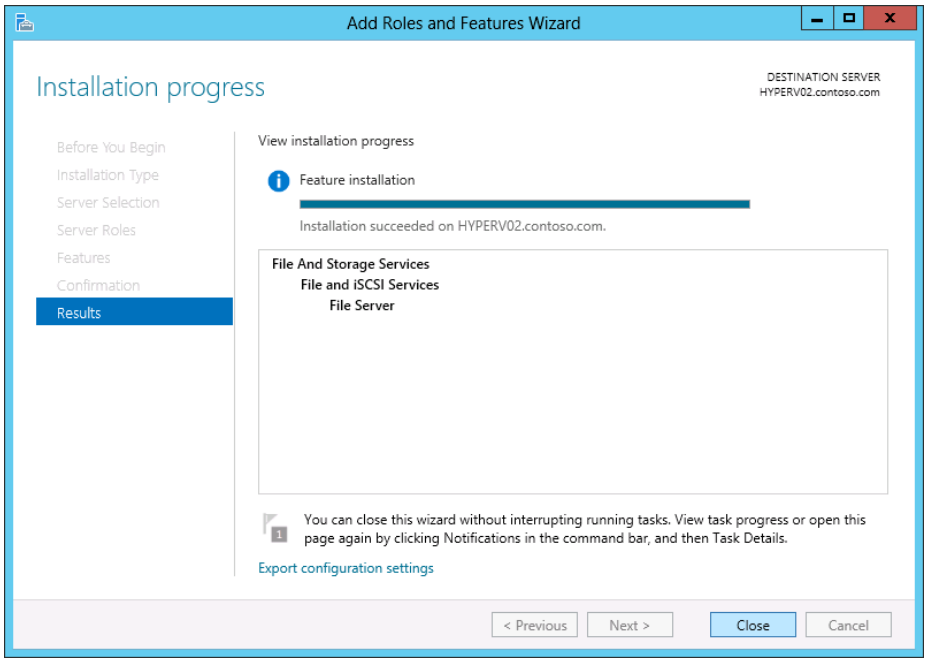
5. In the **Features** screen select **Next**



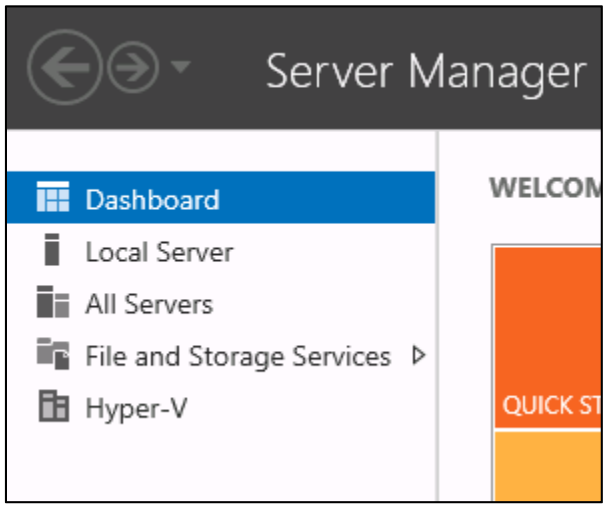
6. In the **Confirmation** screen click on **Install**



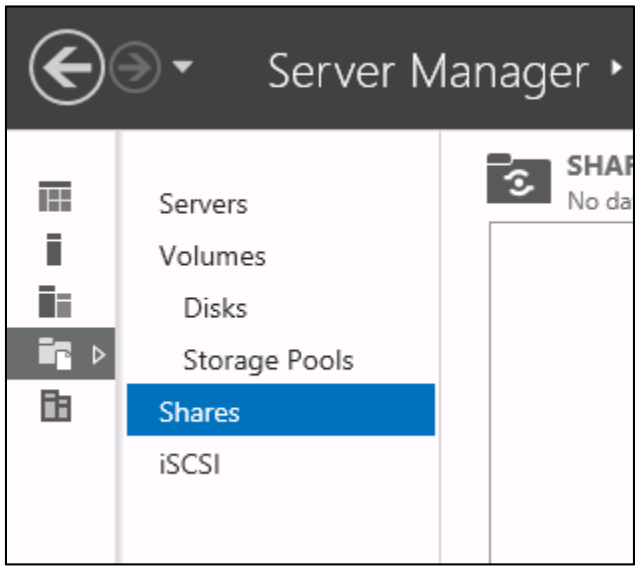
7. On the **Results** screen confirm that the installation was successful. Then select **Close**



8. In Server Manager select File and Storage Services

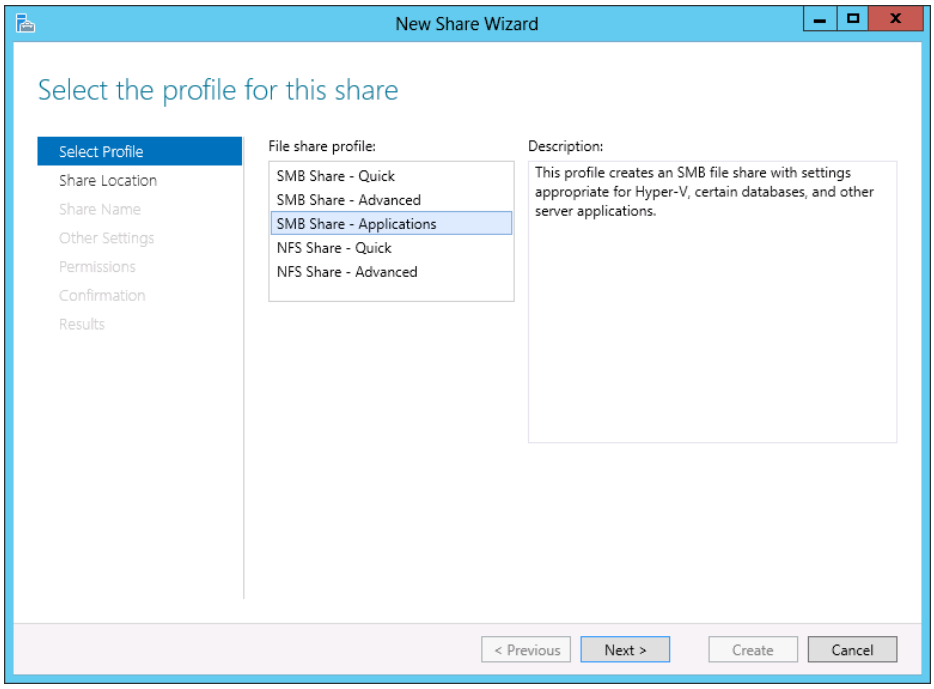


9. Click **Shares**

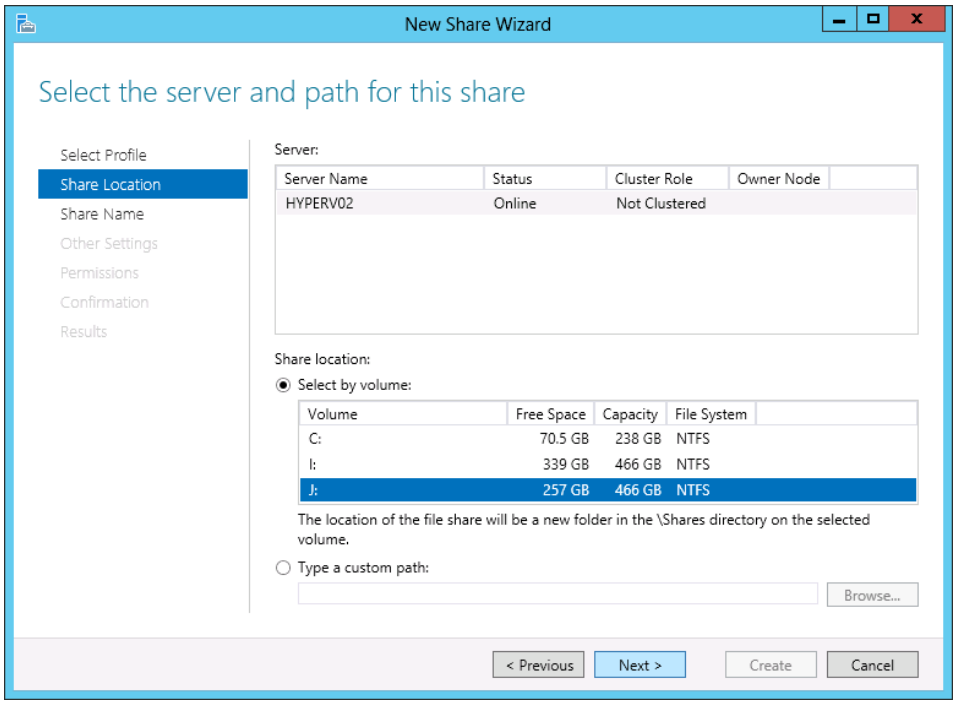


10. Select **Tasks** and then **New Share...**

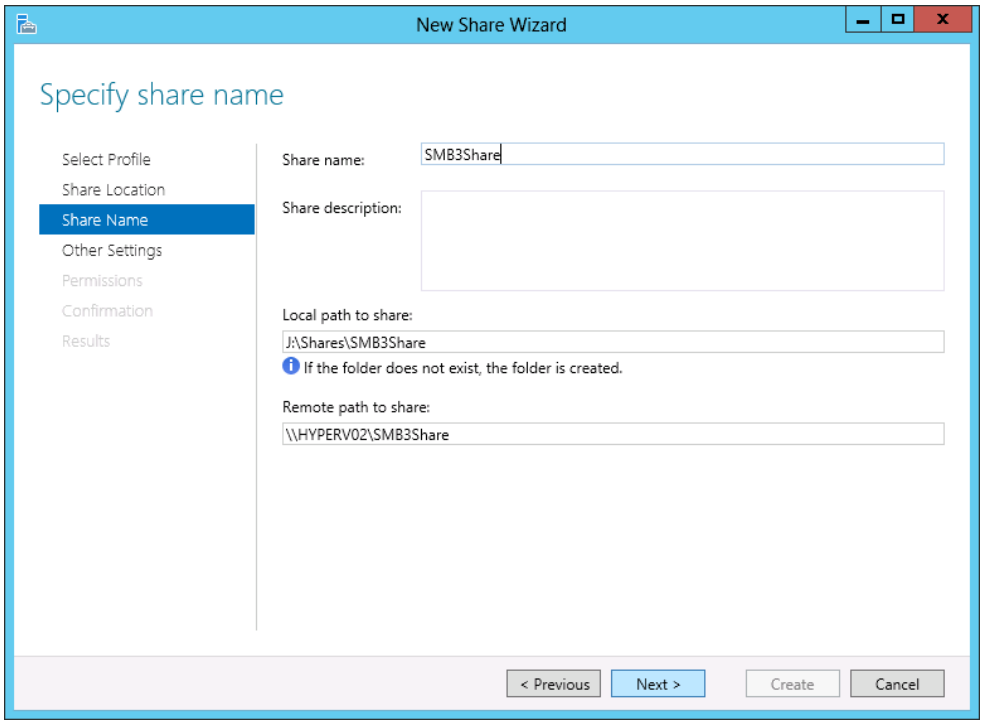
11. On the Select Profile screen, select SMB Share – Applications



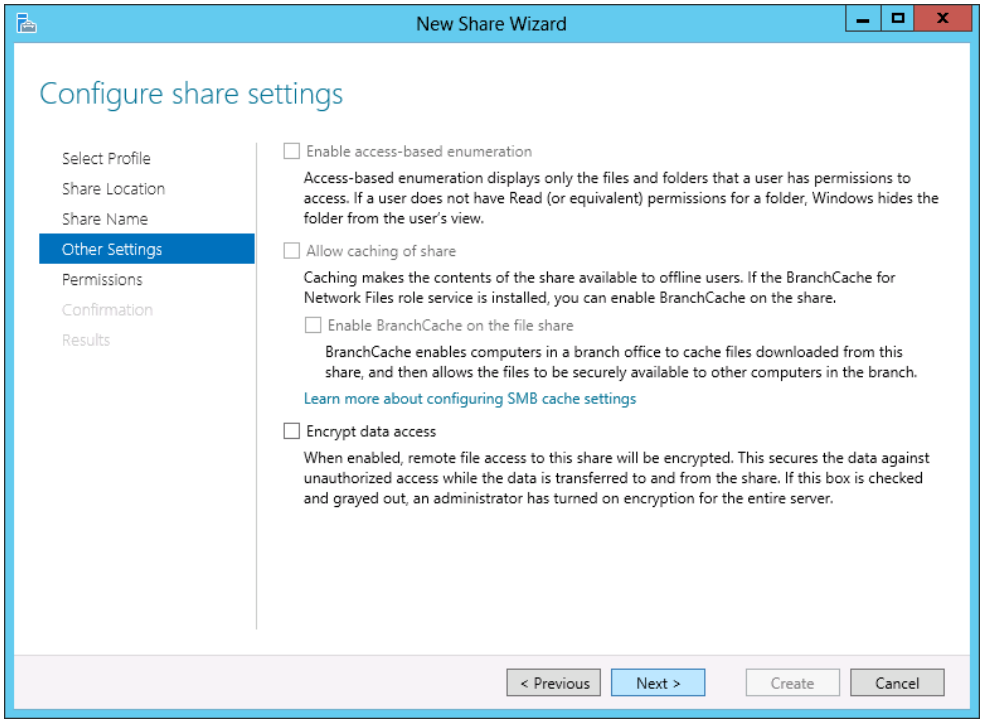
12. In the **Share Location** screen select **HyperV02**, then select the drive that you would like to host your share on (ensure the drive has 100GB of free space). Then select **Next**



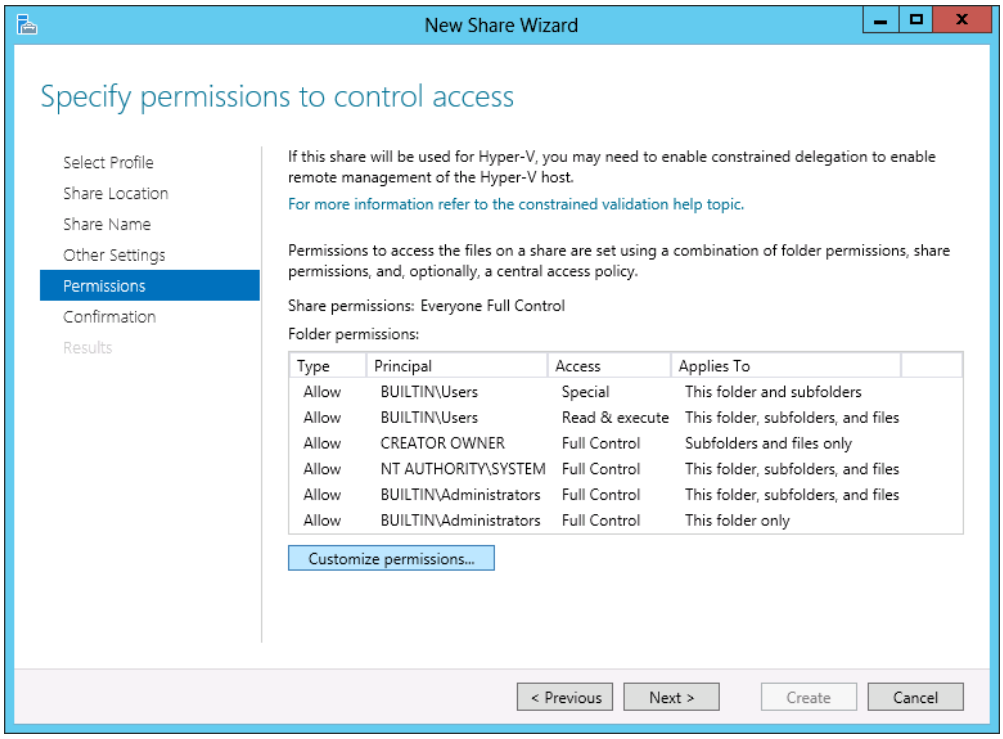
13. In the **Share Name** screen, input the share name **SMB3Share**. Then select **Next**



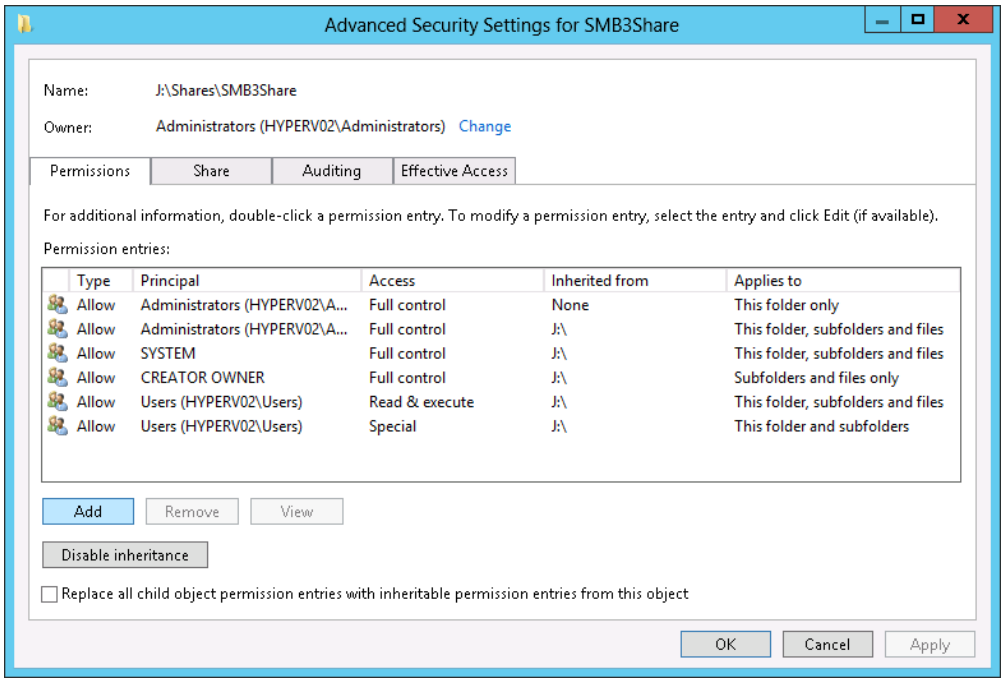
14. On the **Other Settings** screen select **Next** without changing any settings



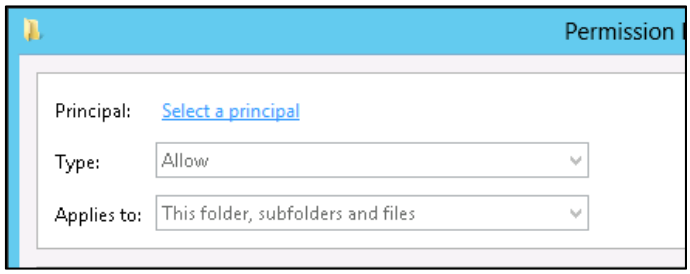
15. On the Permissions screen page select Customize Permissions



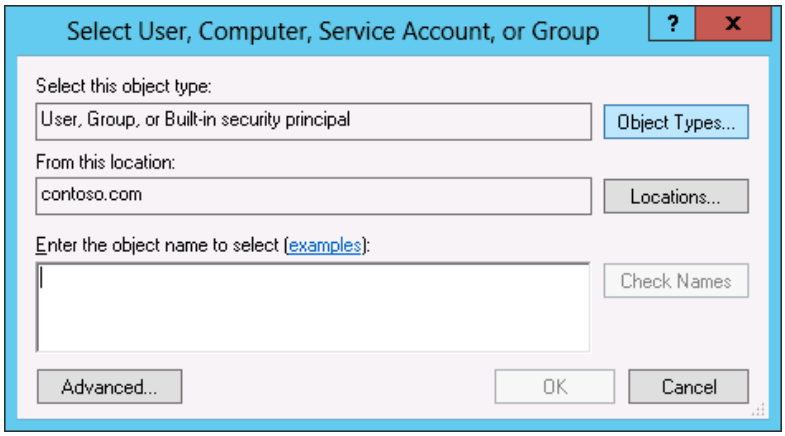
16. In the **Advanced Security Settings for SMB3Share** window, select **Add**



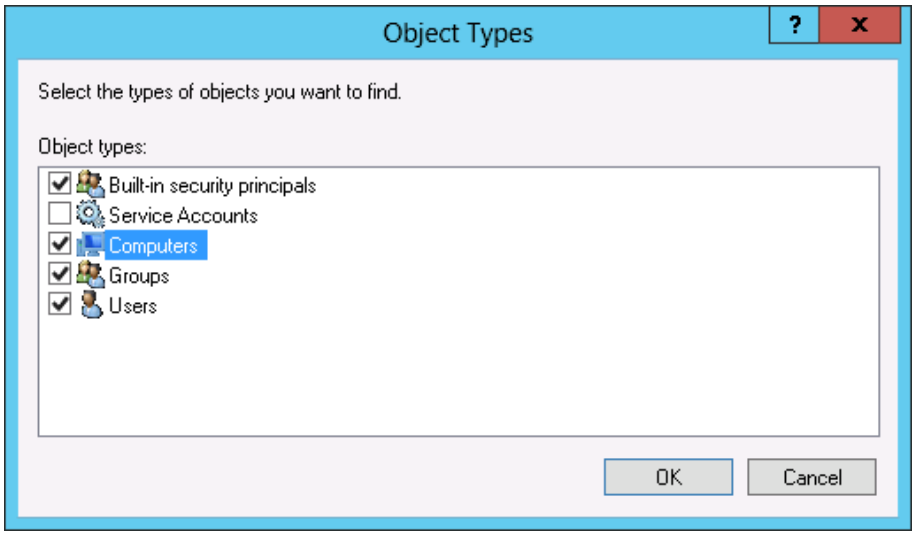
17. Click on Select a Principal



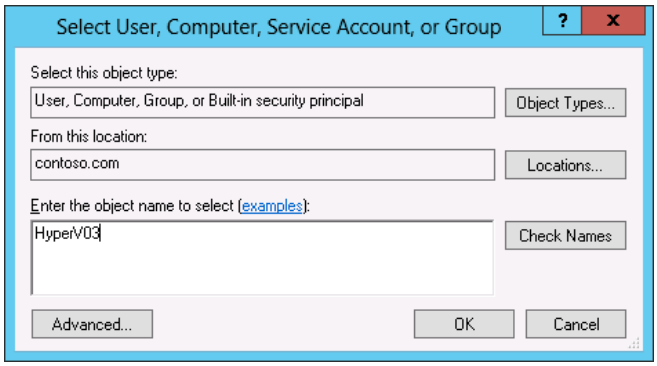
18. Select Object Types



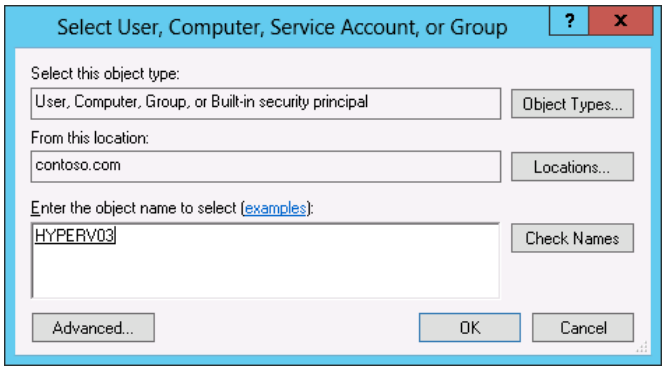
19. Tick **Computers** select **OK**



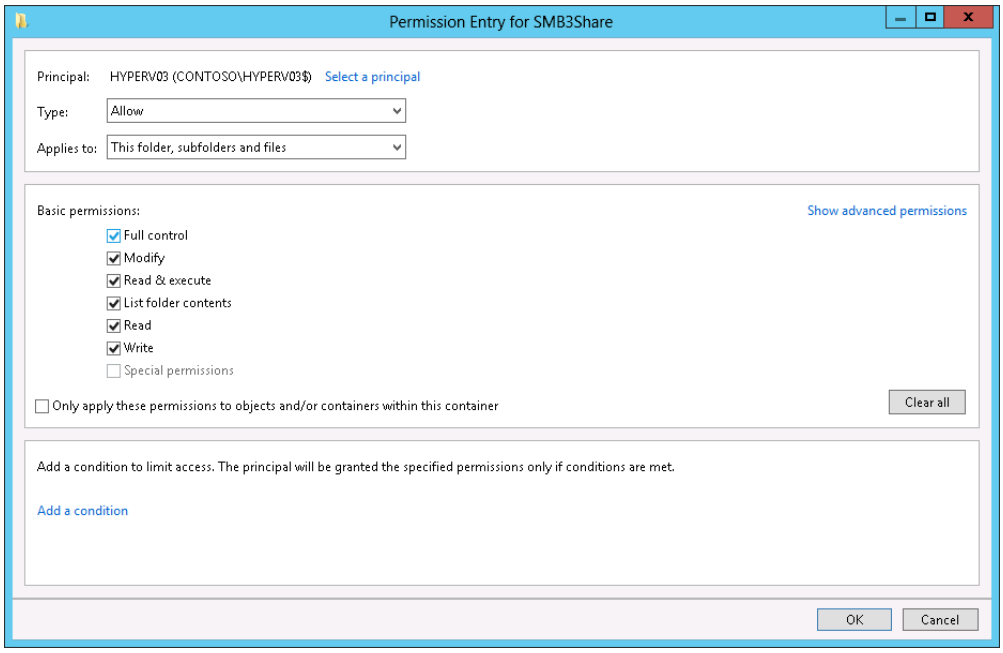
20. Fill in the name of your first Hyper-V host (in this example **HyperV03**), then select **Check Names**



21. Confirm the servers name has been found correctly then click on **OK**

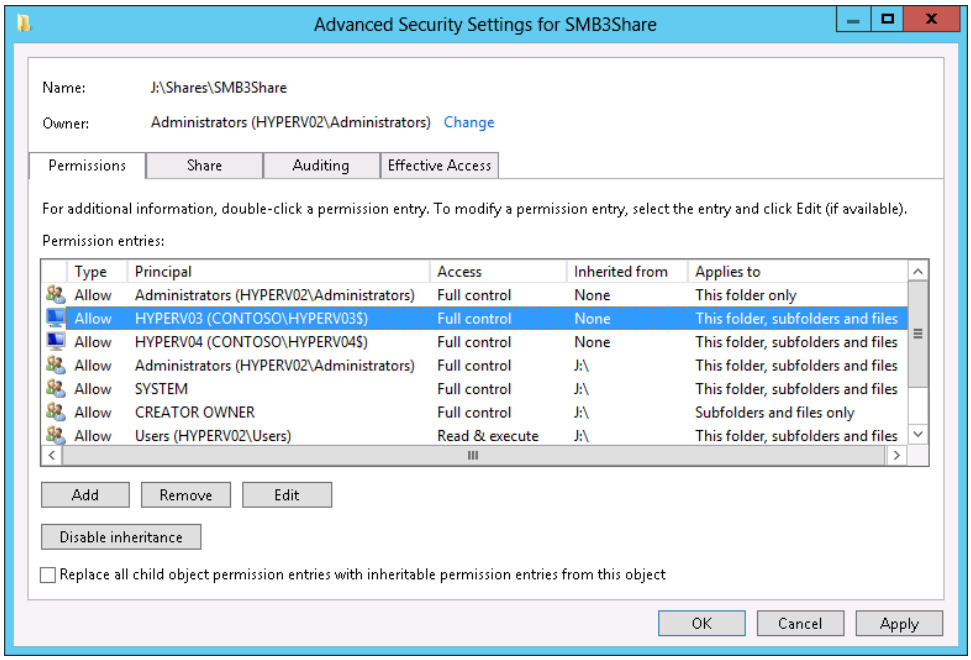


22. This will return you to the permissions for **SMB3Share**, tick **Full Control**, select **OK**

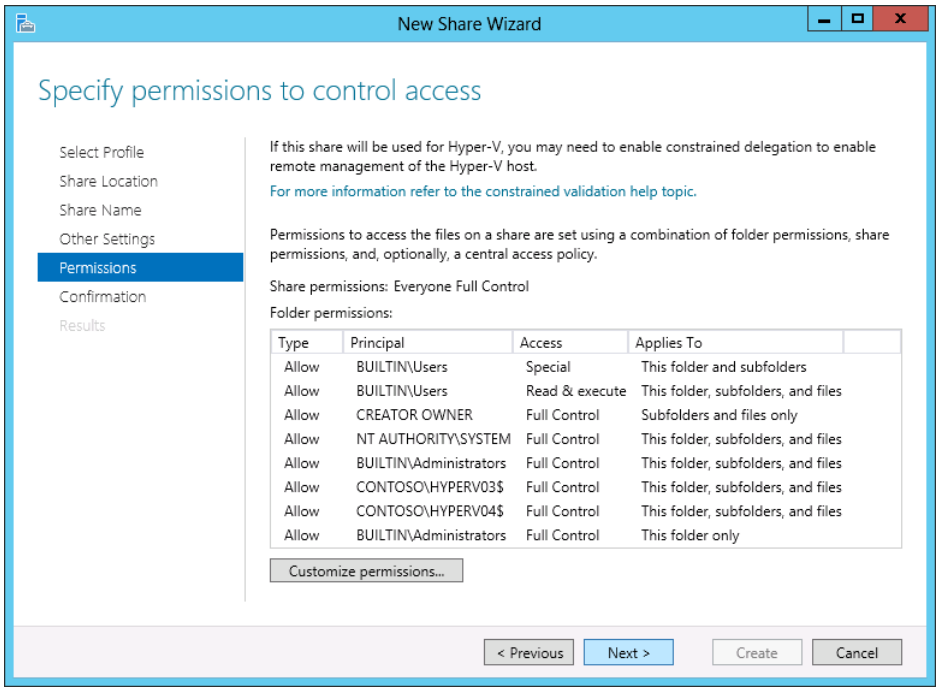


23. Repeat above steps **18-22** for **HyperV04**

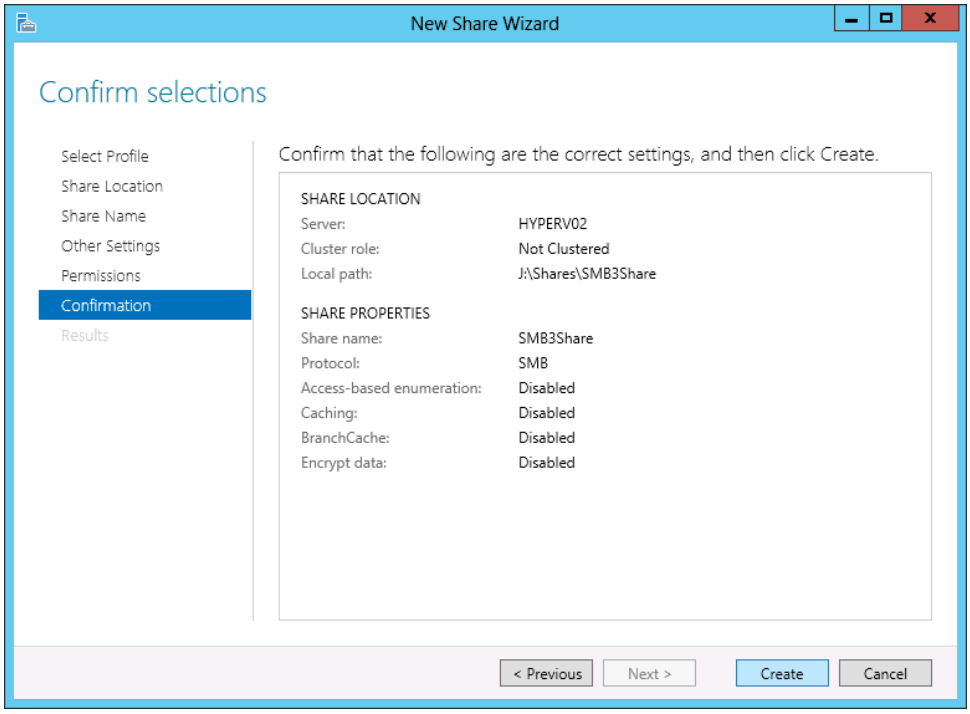
24. Confirm that **HyperV03** and **HyperV04** are listed in the permissions, select **OK**



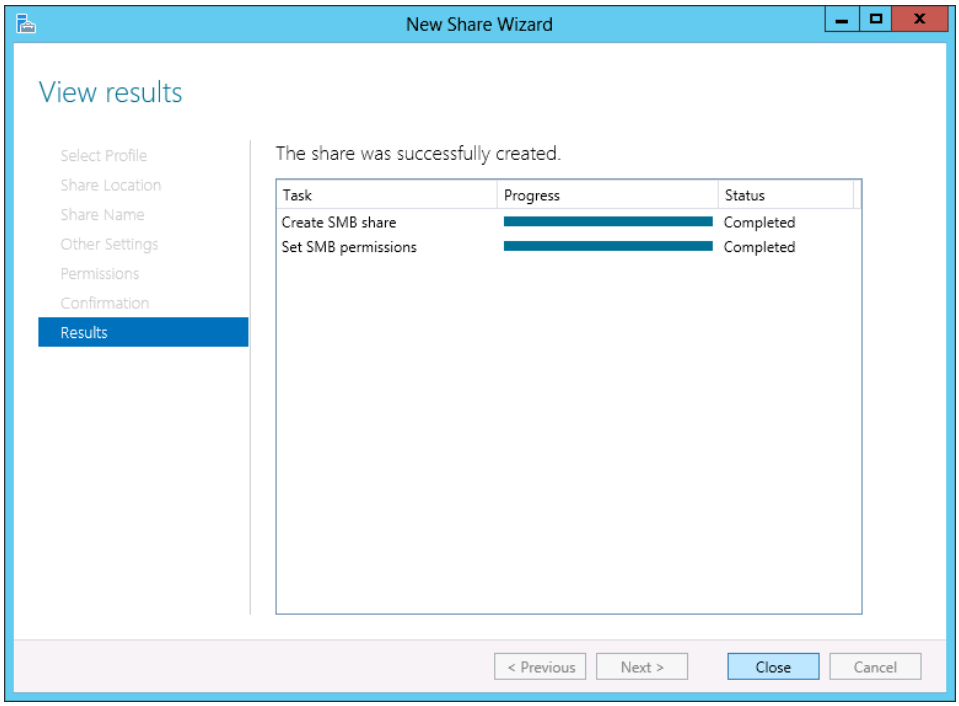
25. On the **Permissions** screen click on **Next**



26. On the **Confirmation** screen click on **Create**



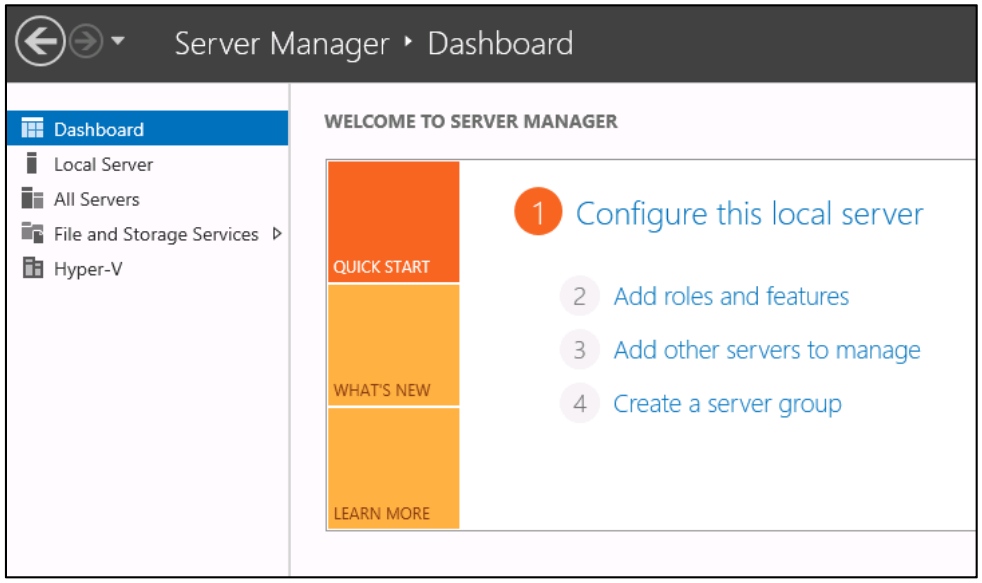
27. Confirm that the action has been completed successfully. Select **Close**



This now completes the creation of an SMB 3.0 share. This will be used as the storage target for the cluster which will be configured in the next step.

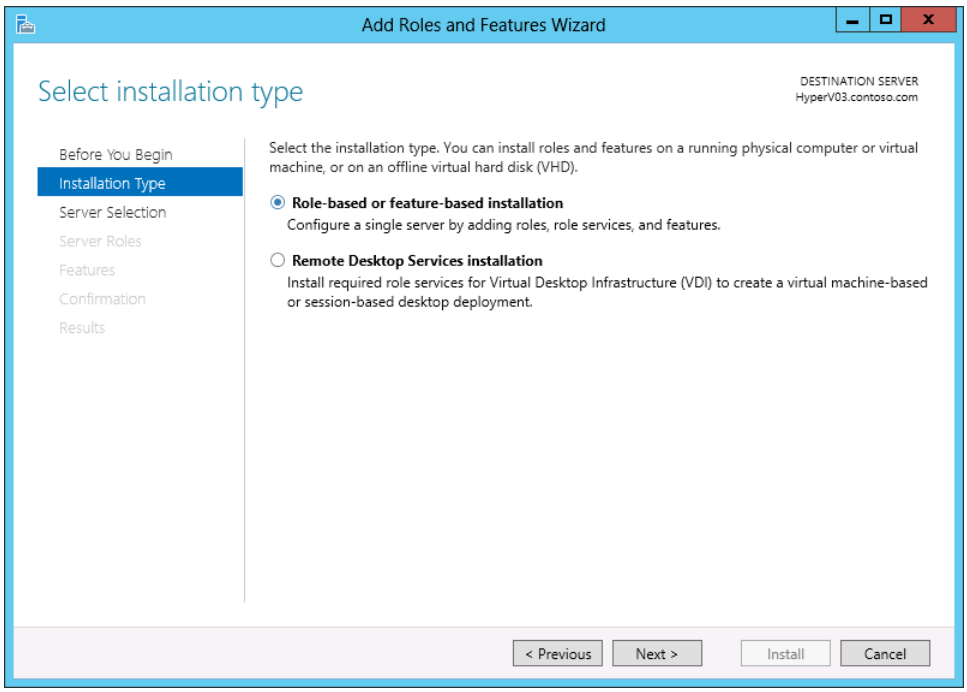
Install the Microsoft Failover Cluster

1. Load **Server Manager**, confirm that the **Dashboard** is open and then select **Add roles and features**

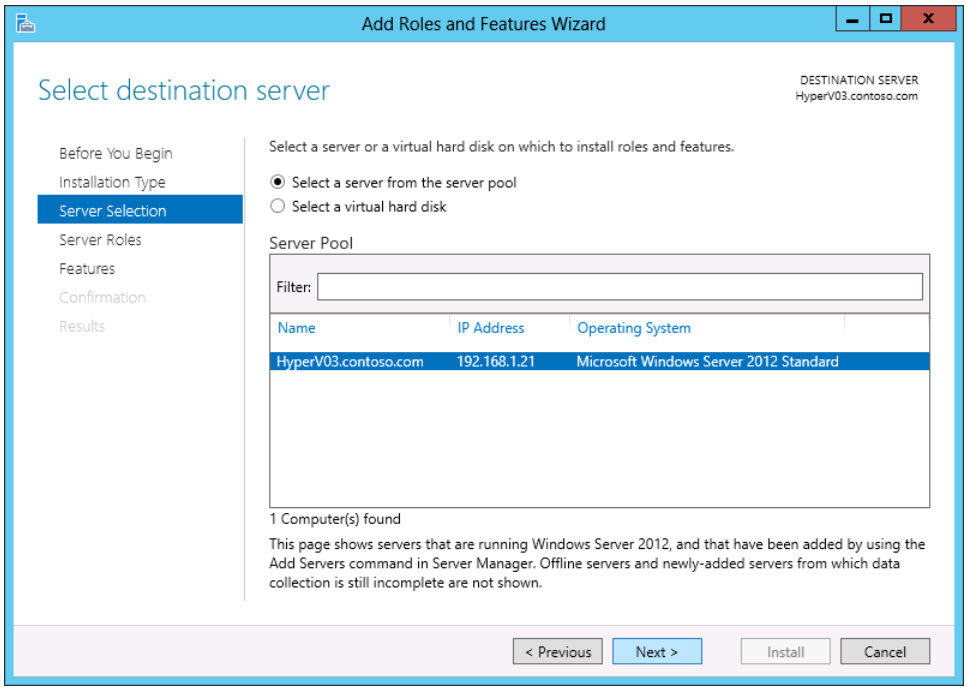


2. On the **Before You Begin** window click Next

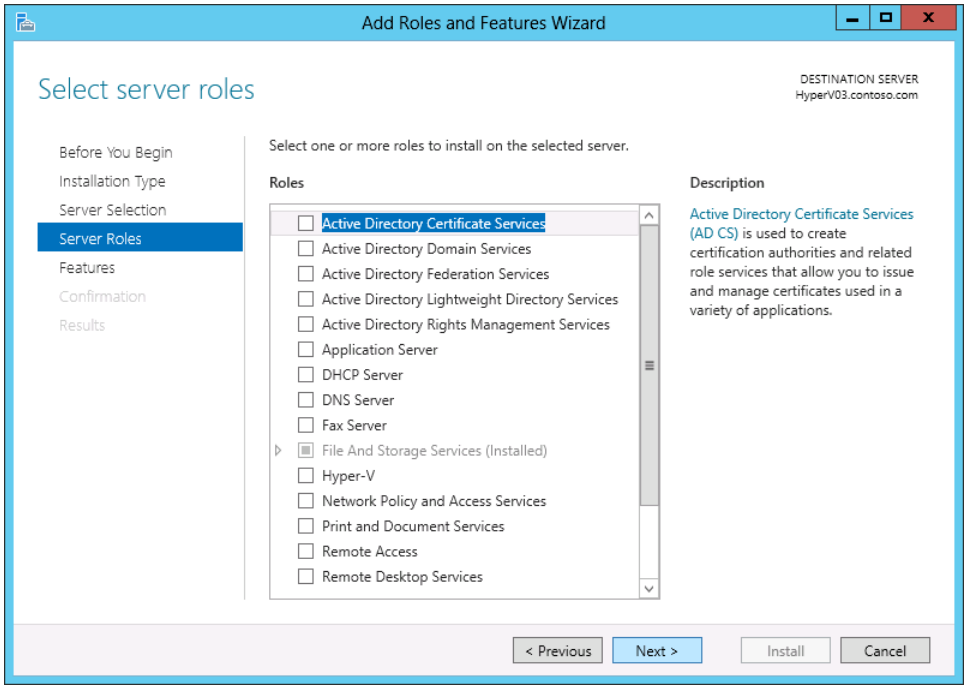
3. In the Installation Type screen, tick Role-based or feature-based installation. Then select Next



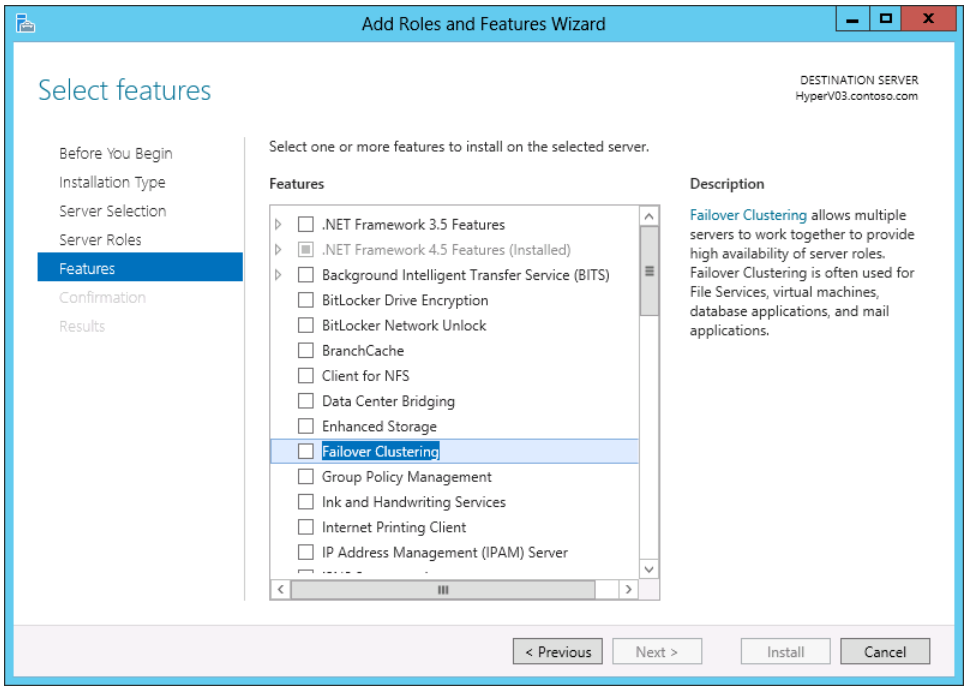
4. On the **Server Selection** screen select **HyperV03** from the **Server Pool**, then select **Next**



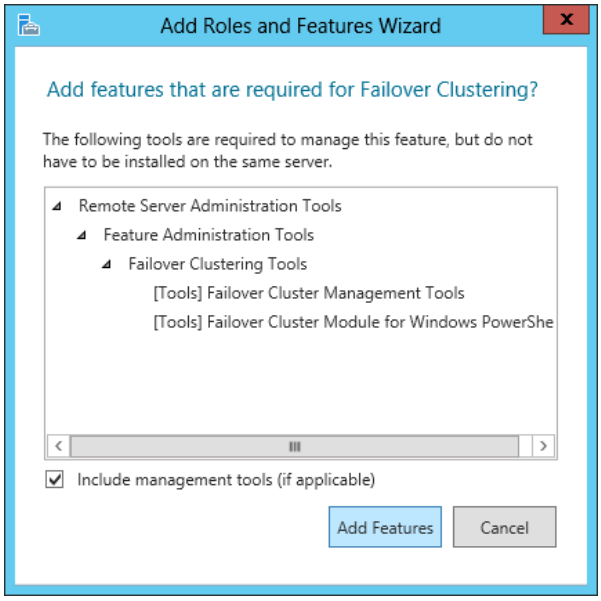
5. On the **Server Roles** screen, click on **Next**



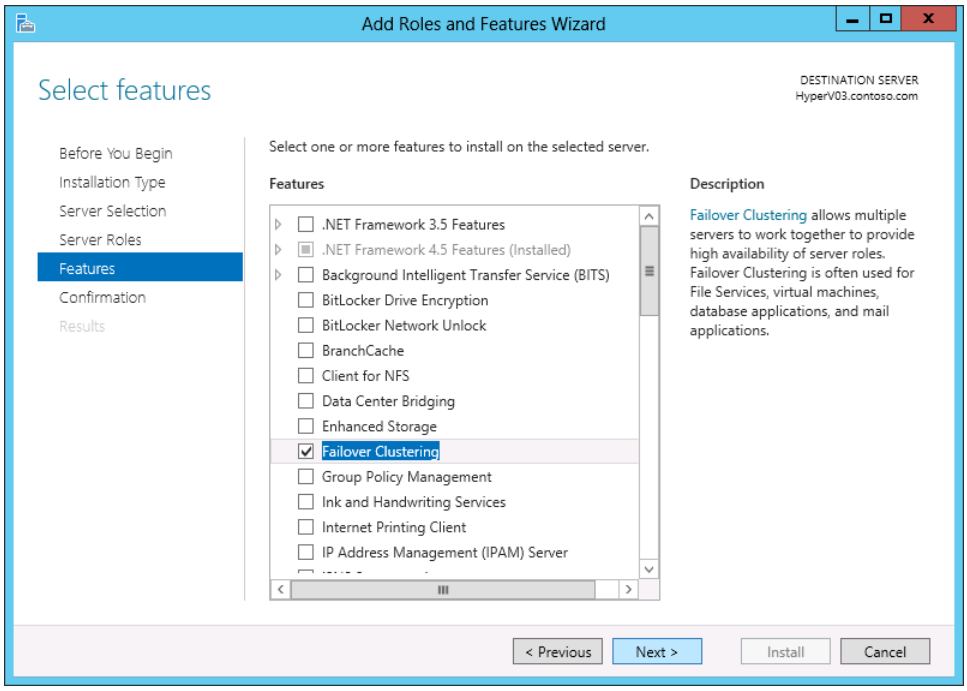
6. In the Features screen tick Failover Clustering



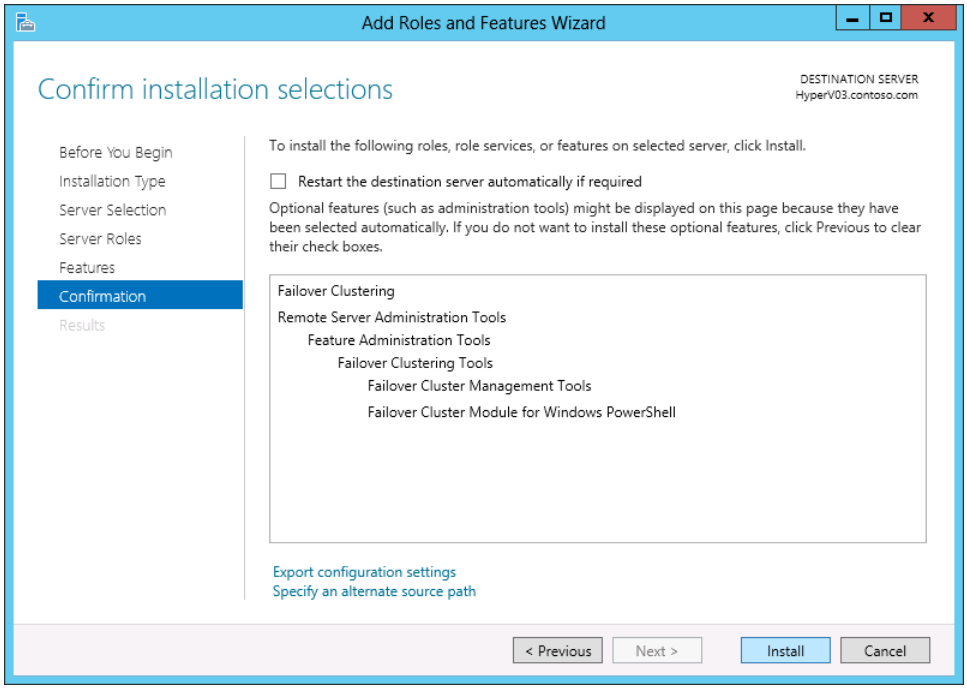
7. The Add features that are required for Failover Clustering window will open, click on Add Features



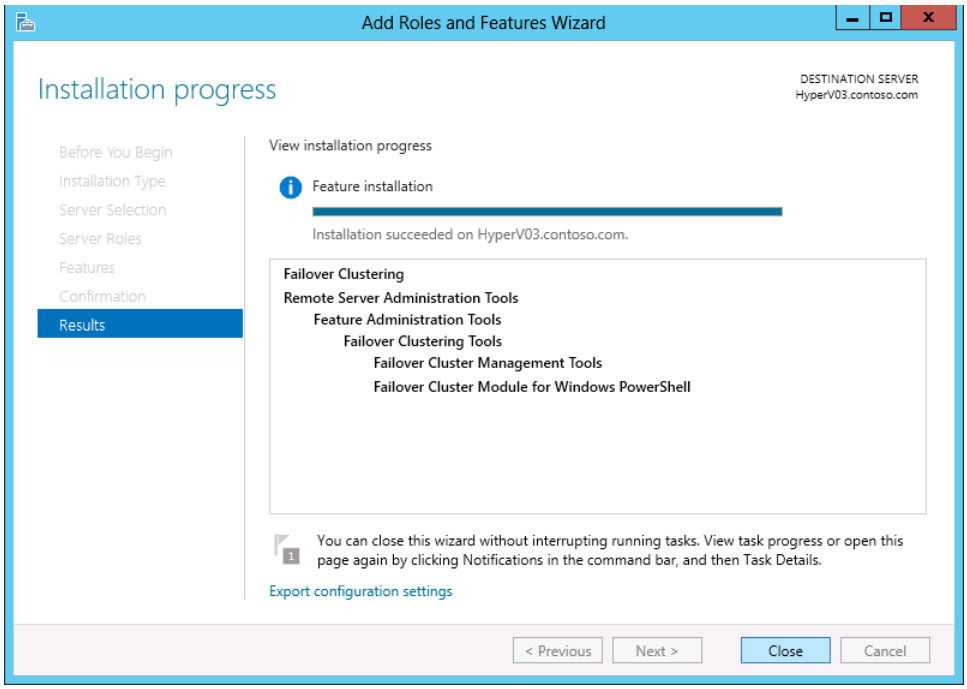
8. On the **Features** screen select **Next**



9. In the **Confirmation** screen click **Install**



10. Review the **Results** to confirm that the installation was successful. Click on **Close** then reboot **HyperV03**

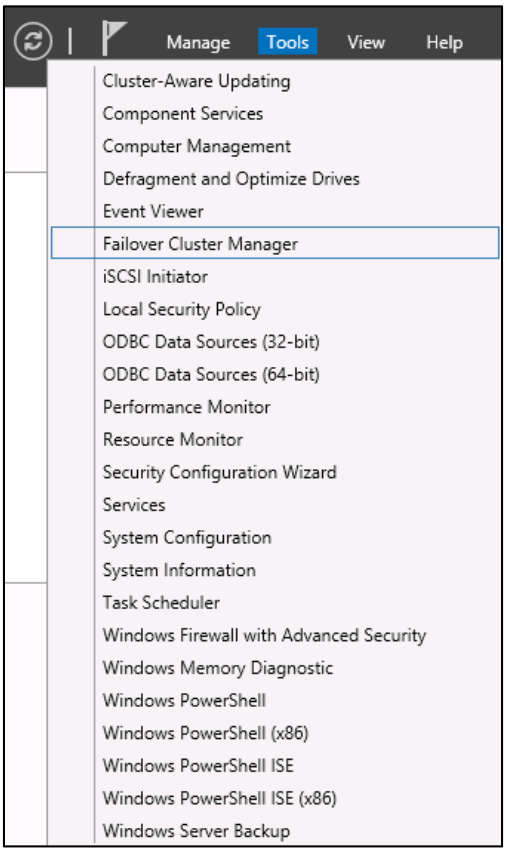


11. Once **HyperV03** has finished its reboot, log in as **administrator** to finalize the install

12. Repeat steps **1-10** on **HyperV04**

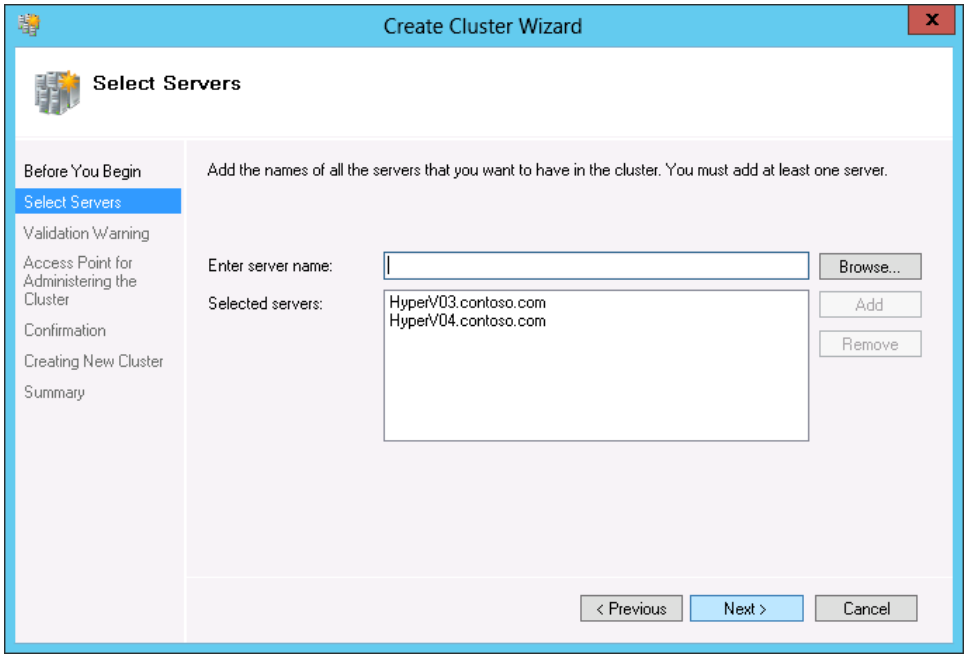
Configuring the Microsoft **Failover Cluster** on HyperV03 and HyperV04

1. From **HyperV03** open **Server Manager** and confirm that the **Dashboard** is open. From the **Tools** menu items, select **Failover Cluster Manager**



2. Once the Failover Cluster Manager loads, Click on Create Cluster and click Next

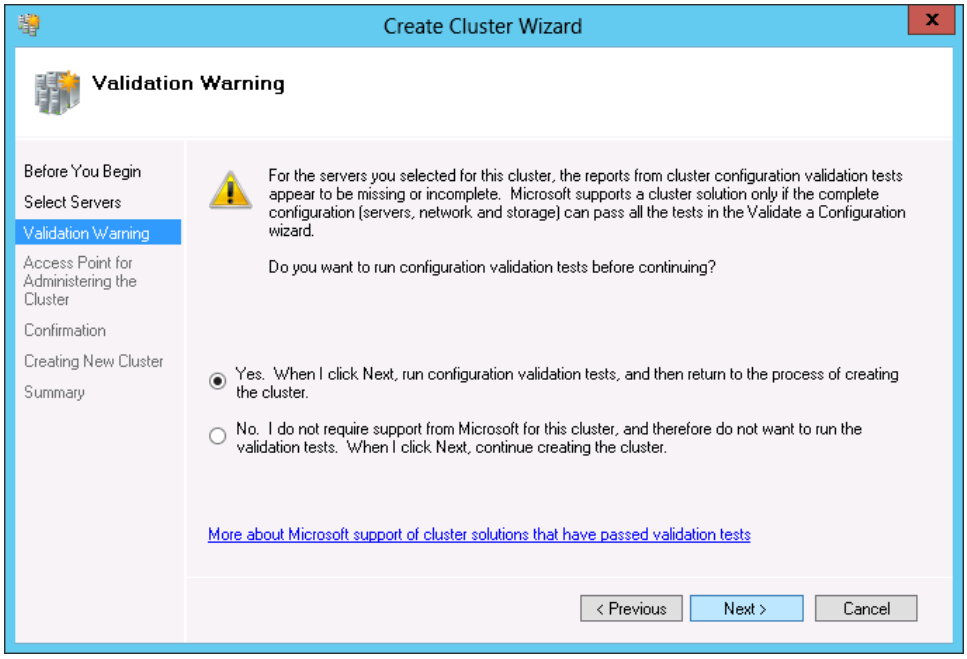
3. In the **Create Cluster Wizard,** Type **HyperV03; HyperV04** and click **Add**. The server names will verify and then press **Next**



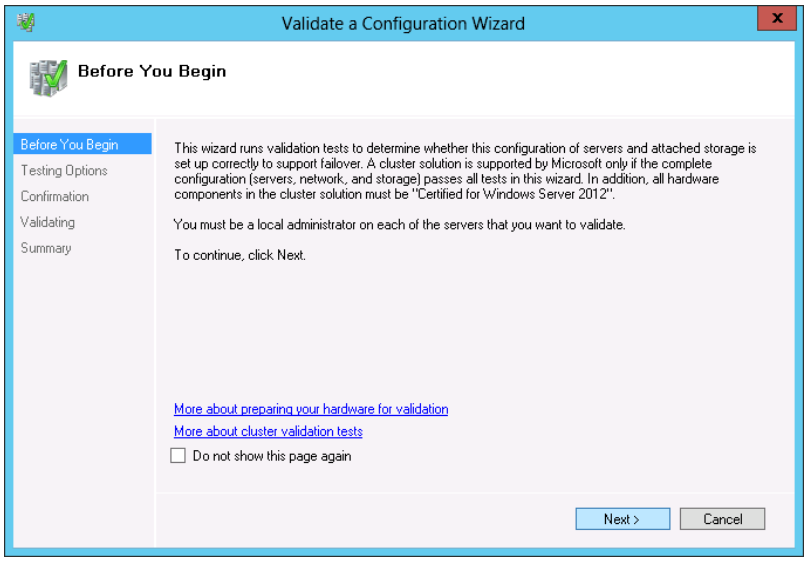
4. On the **Create Cluster Wizard** click **Next** to run cluster validation checks against **HyperV03** and **HyperV04**

Important Note

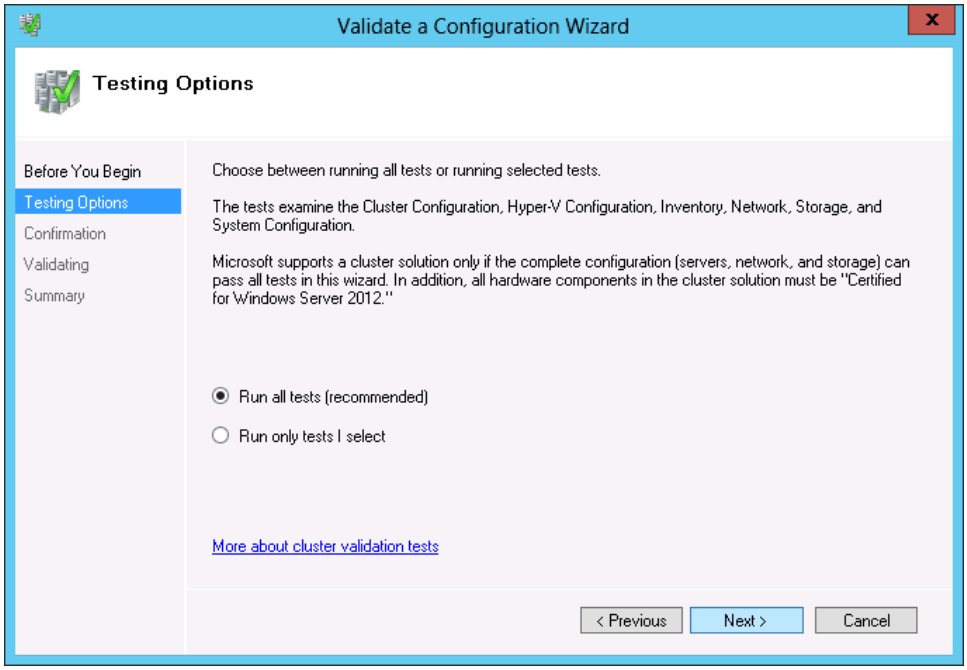
You now have the option to validate the cluster. Microsoft Support will depend on the outcome of this validation test. It is recommended to familiarize yourself with the process during this evaluation guide.



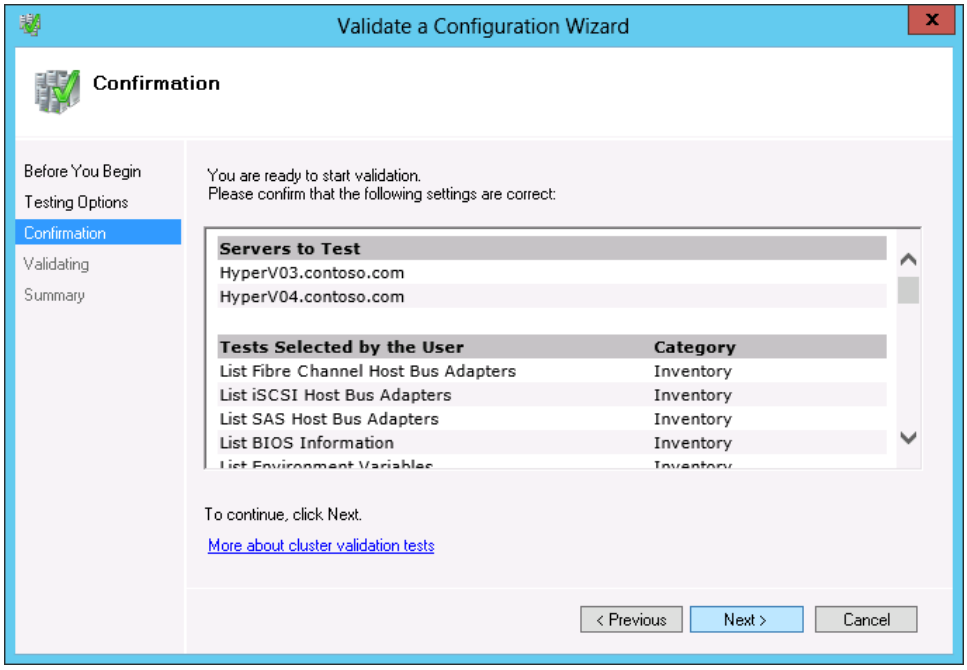
5. On the **Before You Begin** screen of the Validate a Configuration Wizard, click Next



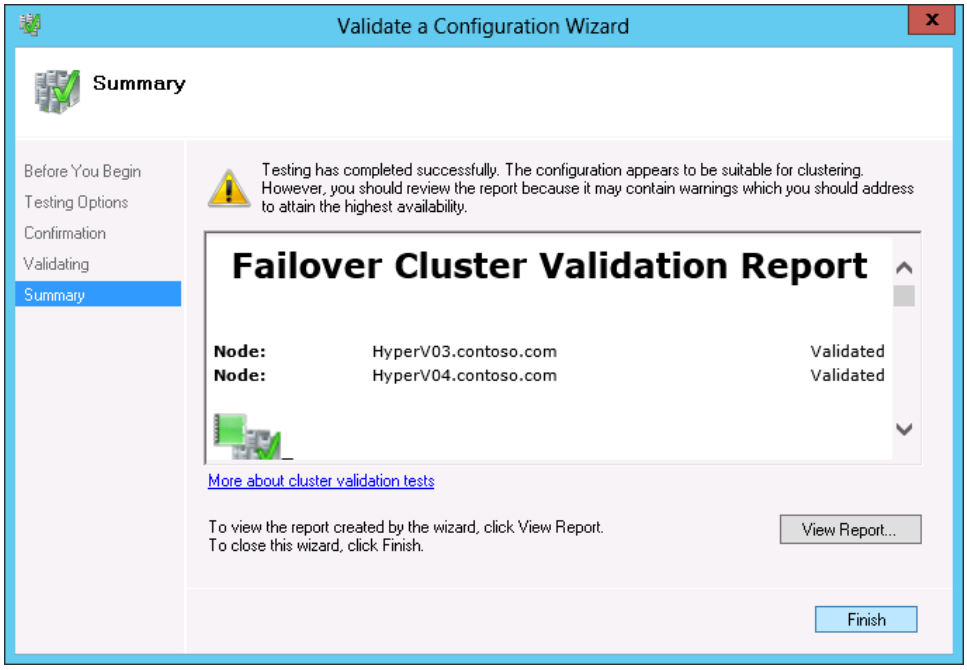
6. On the **Testing Options** screen click **Next** to run all tests



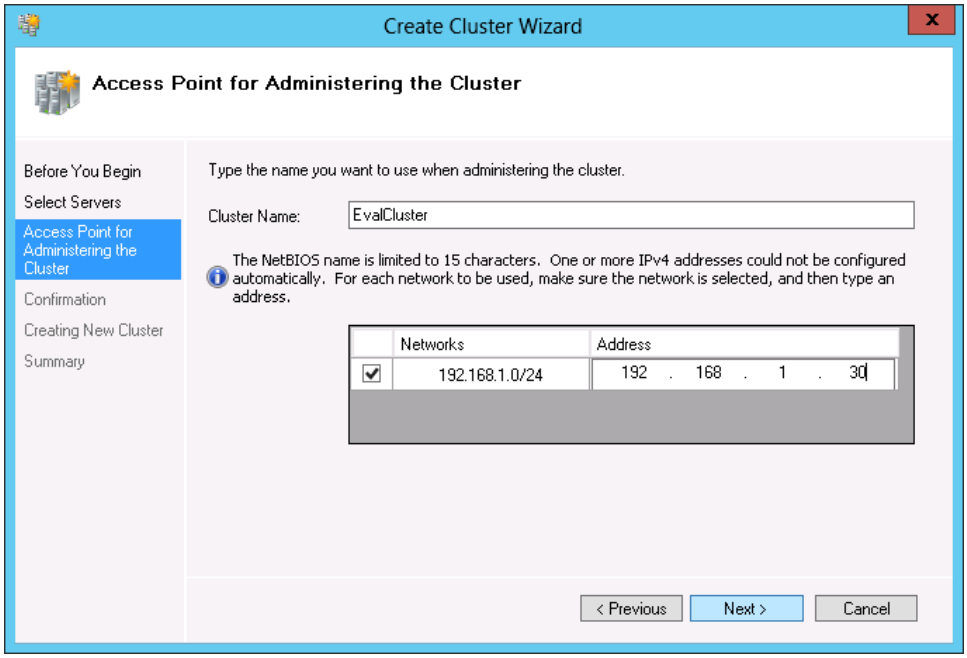
7. On the **Confirmation** screen click **Next**



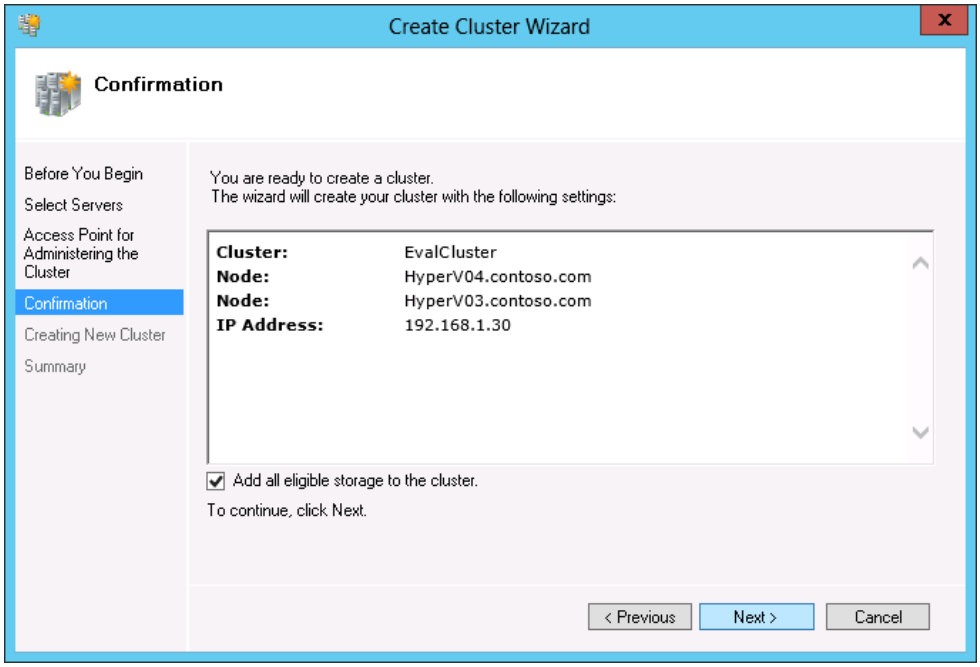
8. On the **Summary** screen make sure both nodes are validated and click **Finish**. (In this example warnings are expected due to hardware limitations of the environment)



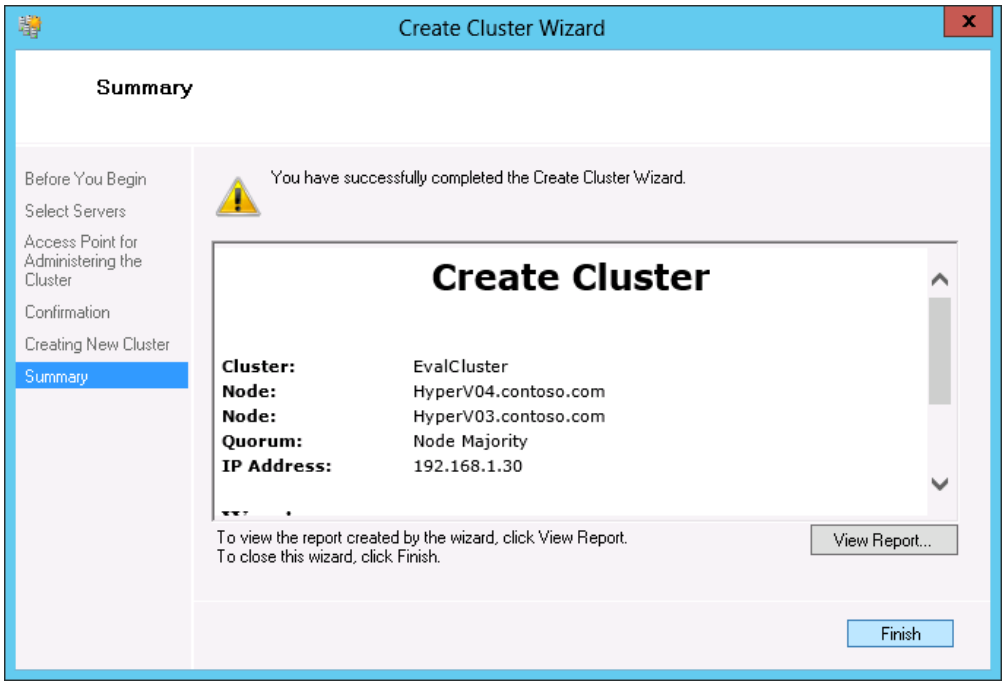
9. On the Access Point for Administering the Cluster screen of the Create Cluster Wizard, enter the cluster name EvalCluster, enter the IP address 192.168.1.30 and click Next



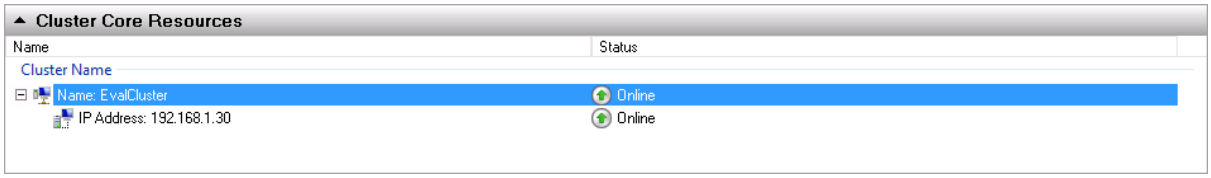
10. On the **Confirmation** screen, review the entries are correct and click **Next**



11. On the **Summary** screen click **Finish**



12. The cluster is now created, from the **Failover Cluster Manager** you should see the resources online

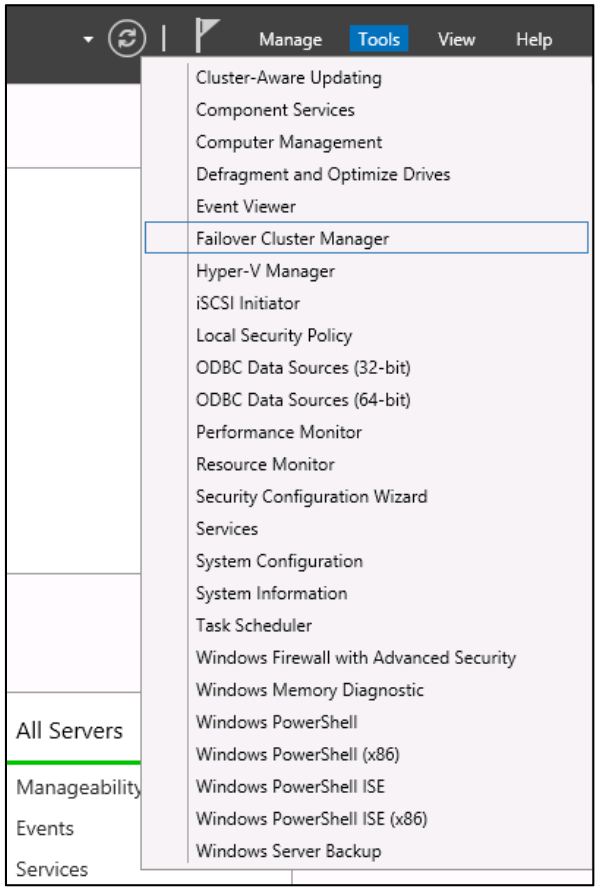


At the end of this exercise we have created a Microsoft Failover Cluster. This technology will allow you to build resilient and fault redundant applications in your private cloud.

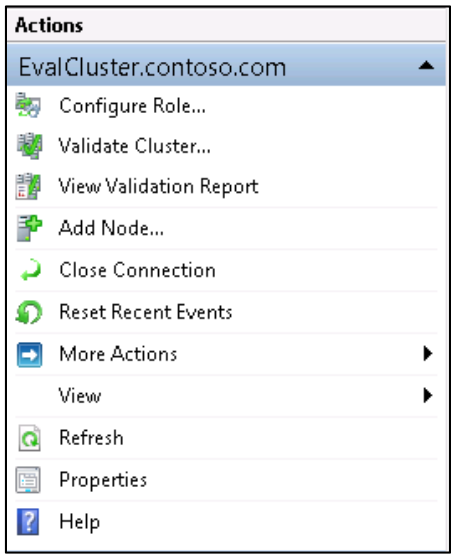
Configure Microsoft Hyper-V Replication

As **HyperV03** and **HyperV04** are already part of a **Failover Cluster**, the standard Hyper-V replication between two hosts cannot be used. Instead a Hyper-V **Replica Broker** will need to be used. The Hyper-V **Replica Broker** is an additional role that needs to be added to the **Failover Cluster**:

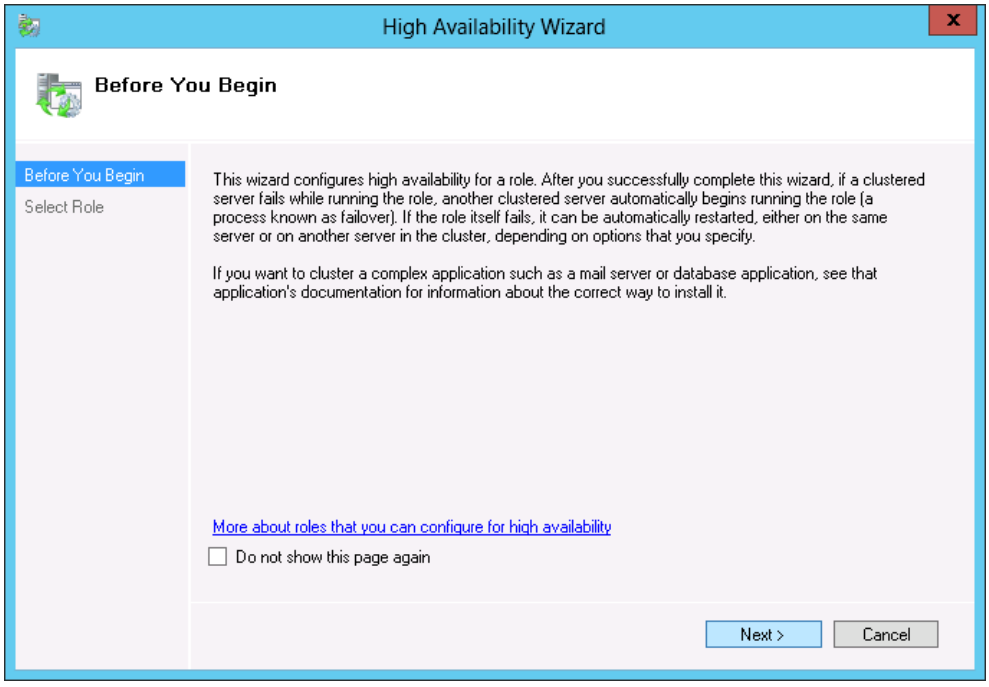
1. From **HyperV03** open **Server Manager** and confirm that the **Dashboard** is open. From the **Tools** menu items, select **Failover Cluster Manager**



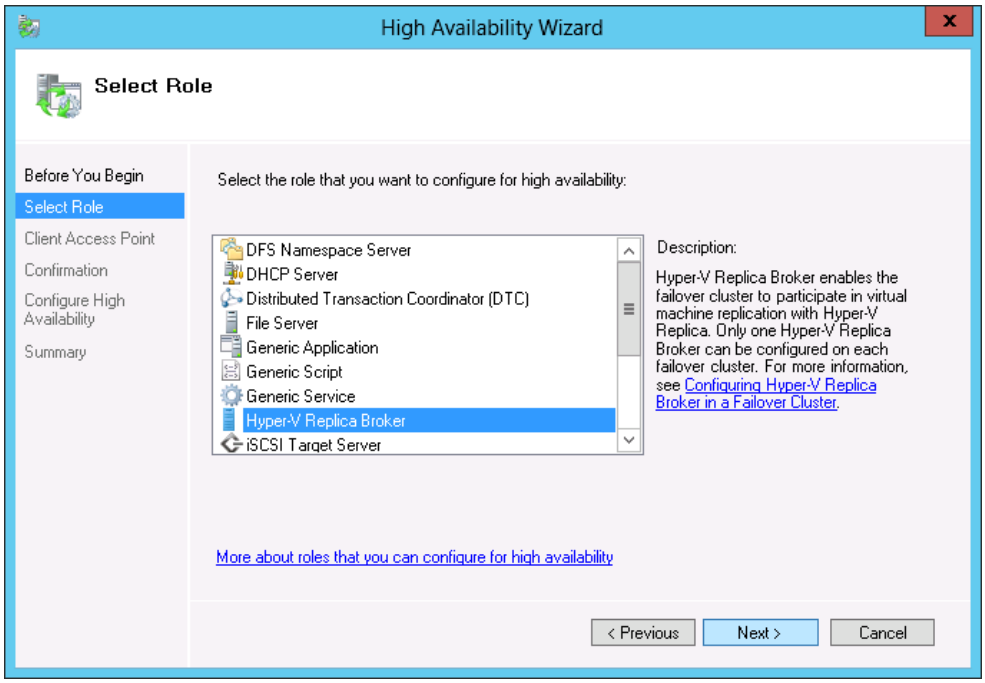
2. In the Failover Cluster Console click on EvalCluster, then in the Actions column select Configure Role



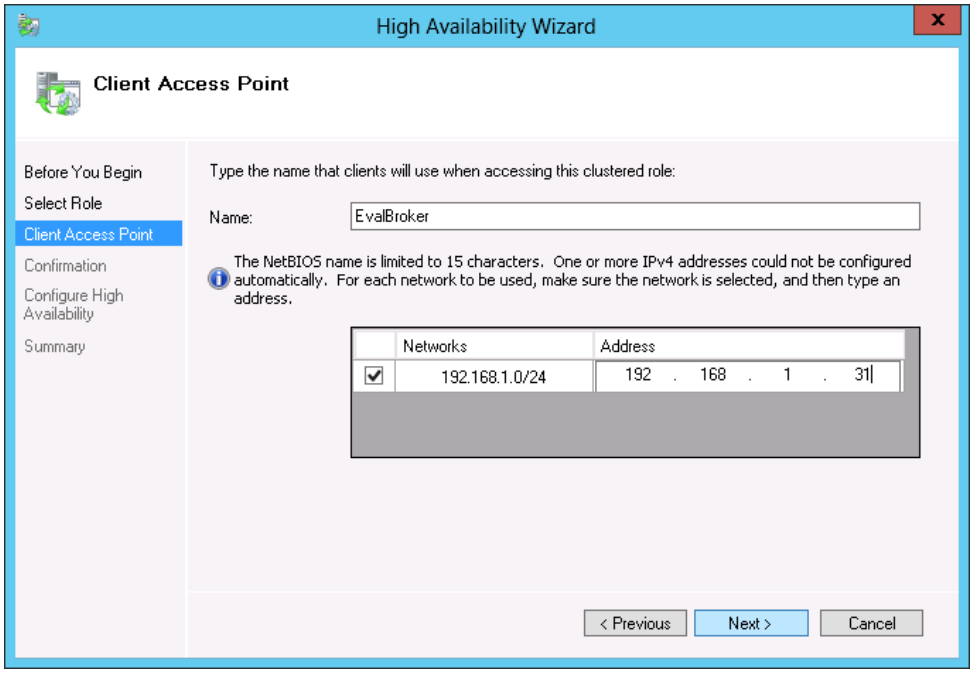
3. In the High Availability Wizard, on the Before You Begin window, click on Next



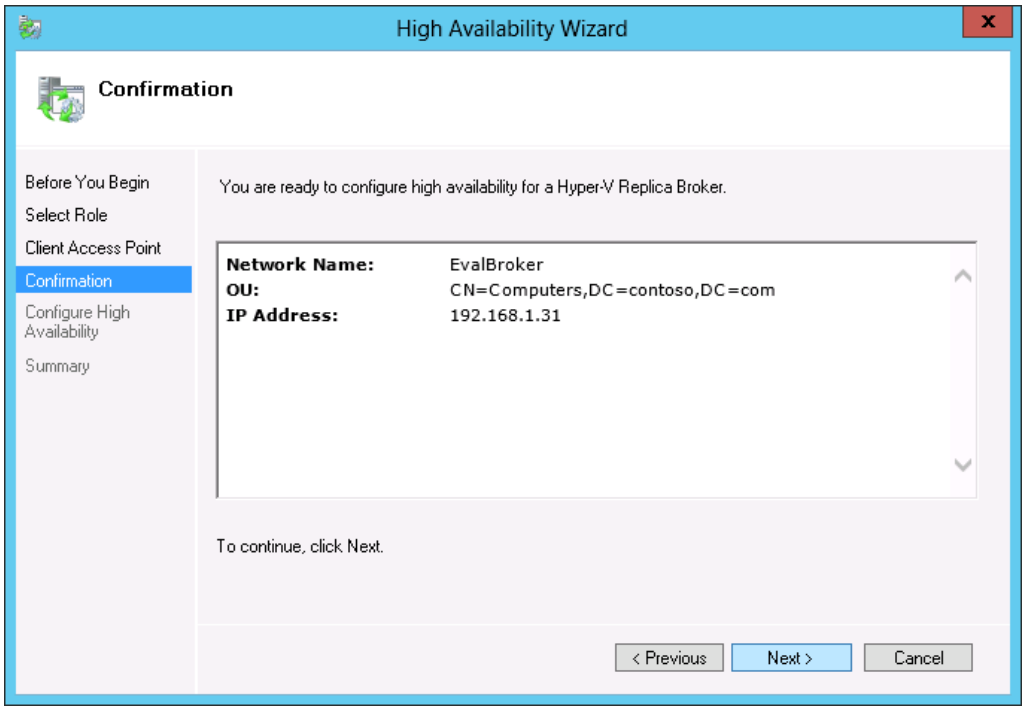
4. Select Hyper-V Replica Broker, then select Next



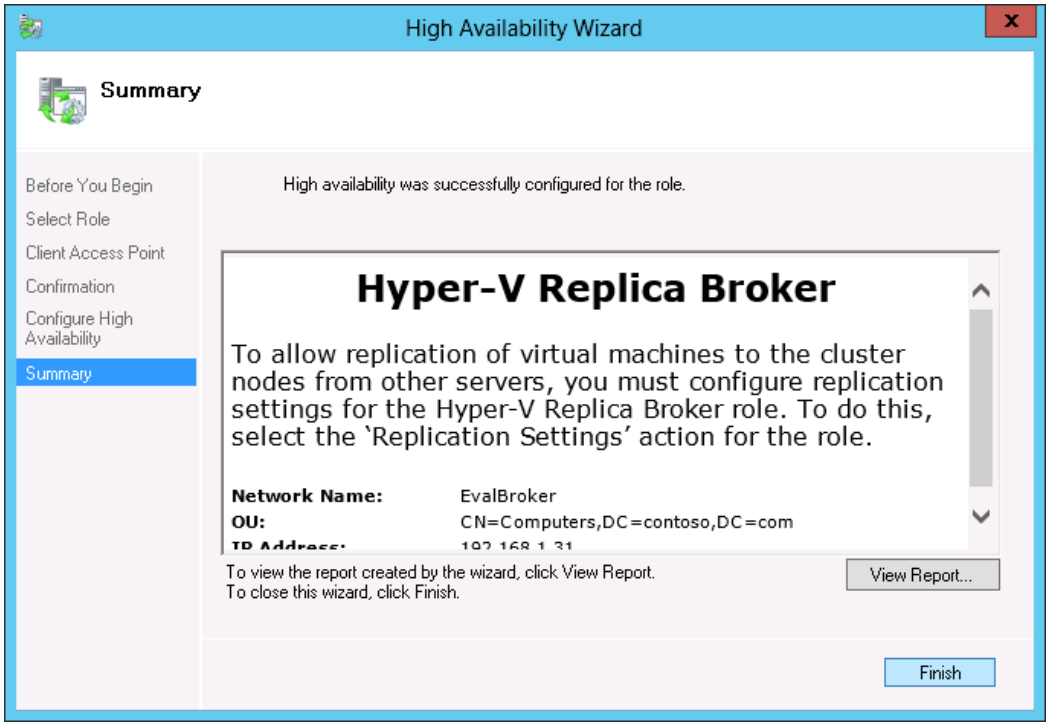
5. In the **Client Access Point** screen, type in the name **EvalBroker**, and for the IP address input **192.168.1.31**. Then select **Next**



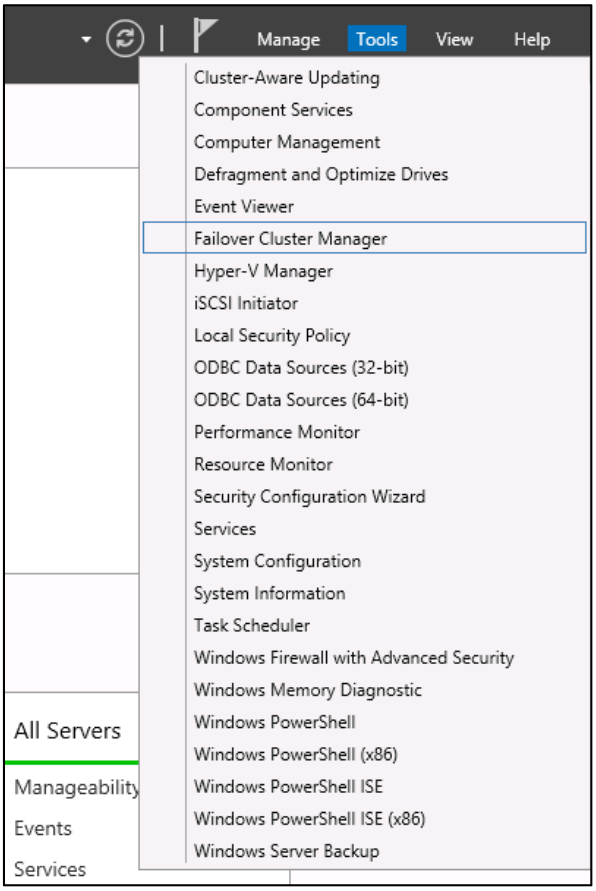
6. On the **Confirmation** screen click **Next**



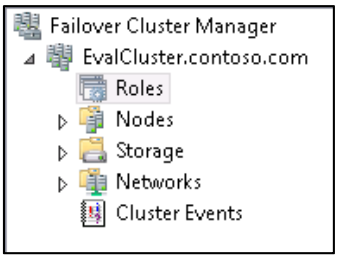
7. On the **Summary** screen click on **Finish**



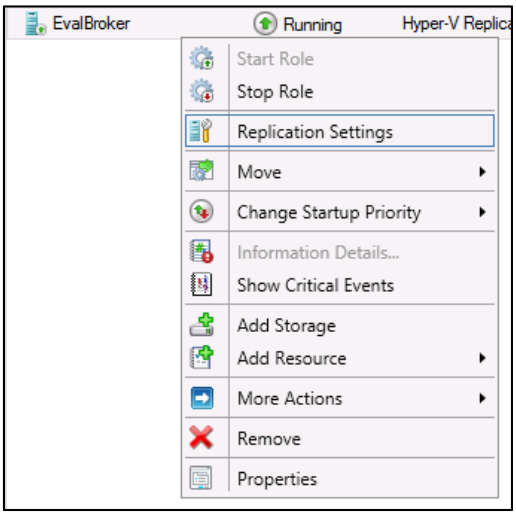
8. From **HyperV03** open **Server Manager** and confirm that the **Dashboard** is open. From the **Tools** menu items, select **Failover Cluster Manager**



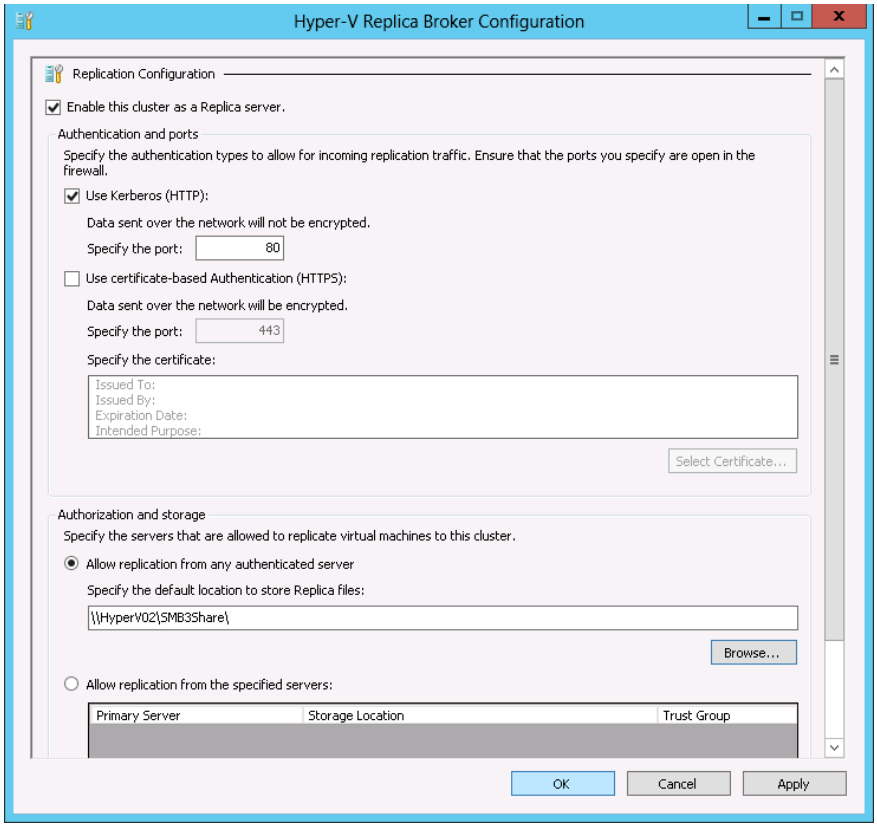
9. Expand **EvalCluster**, then click on **Roles**



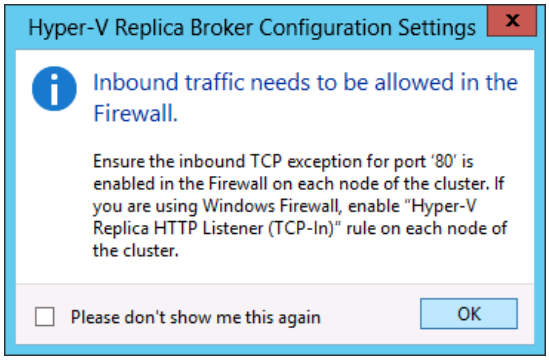
10. Right click on EvalBroker then select Replication Settings



11. Tick the Enable this cluster as a Replica server, the Use Kerberos (HTTP), and the Allow replication from any authenticated server boxes. In the Specify the default location to store Replica files box input \\HyperV02\SMB3Share then select OK

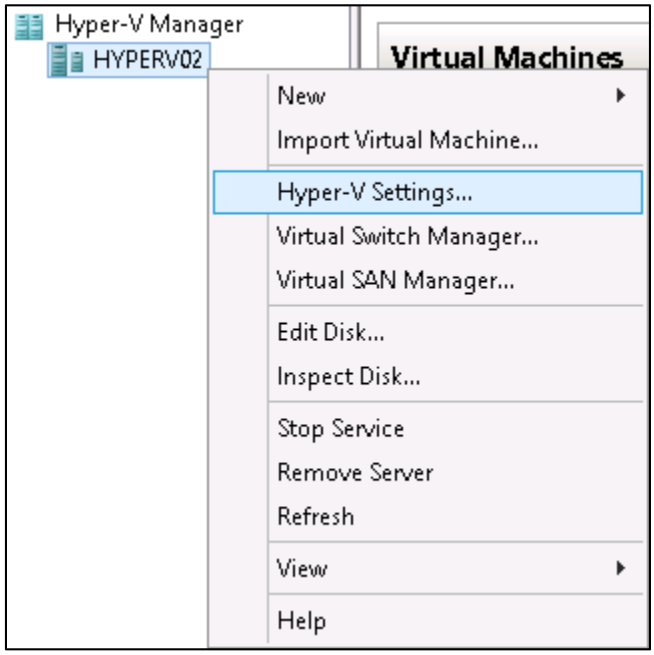


12. Take note of the **Inbound traffic needs to be allowed in the Firewall** information box. As **Windows Firewall** has been disabled on all hosts this information can be disregarded. Select **OK**

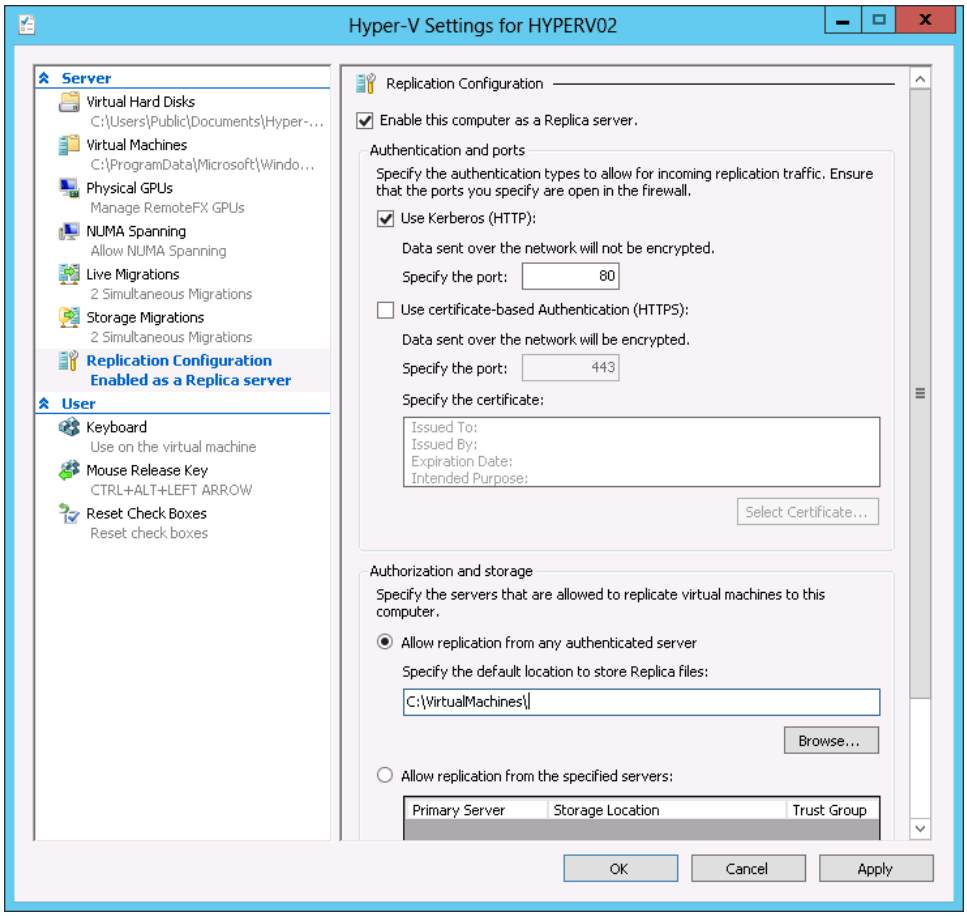


The next step is to configure the other end of the replication partnership. This is to be completed on **HyperV02**.

13. Open **Hyper-V Manager** on **HyperV02,** right click on **HyperV02** in the left hand column, and then select **Hyper-V Settings…**



14. On the Hyper-V Settings for HyperV02 window, select Replication Configuration. Tick the boxes Enable this computer as a Replica server, then Use Kerberos (HTTP), and Allow replication from any authenticated server. In the Specify the default location to store Replica files: box input C:\VirtualMachines (create the C:\VirtualMachines folder if it does not already exist). Select OK

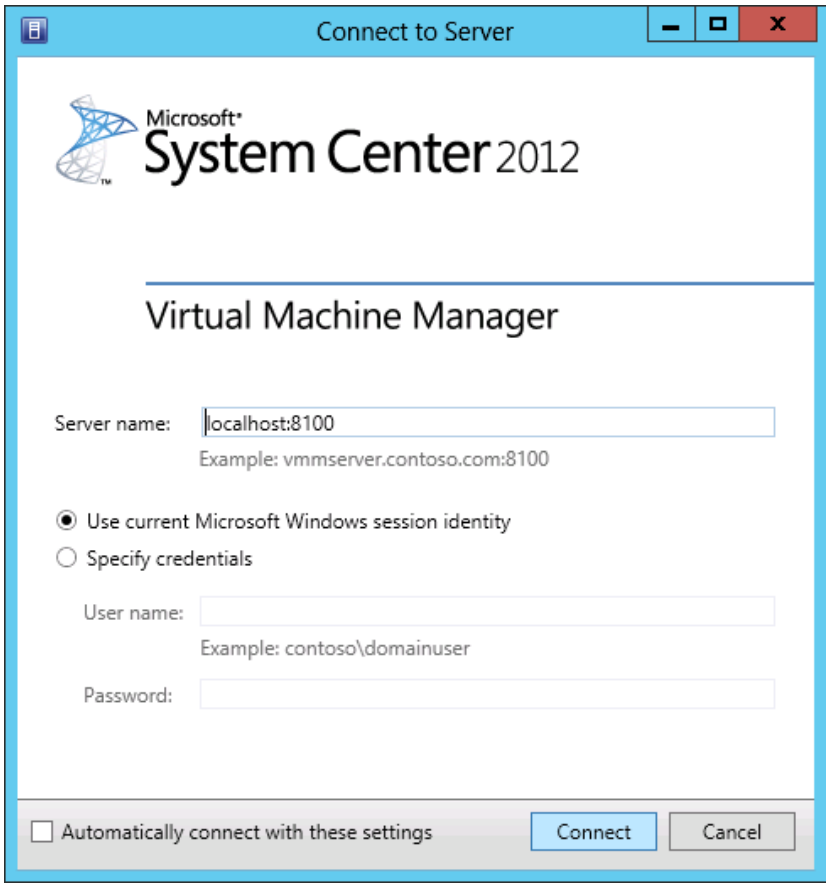


15. Take note of the **Inbound traffic needs to be allowed in the Firewall** information box. As **Windows Firewall** has been disabled on all hosts this information can be disregarded. Select **OK**

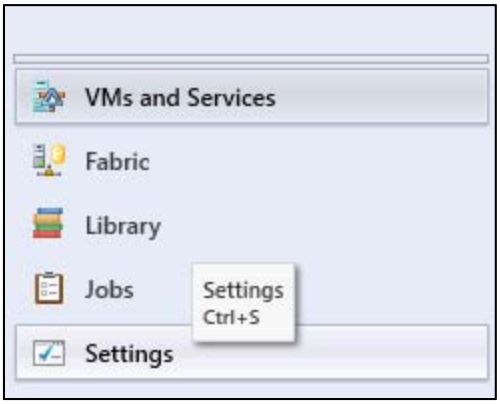
This concludes the configuration of the replication partnership between **EvalCluster** and **HyperV02**. No Hyper-V guests will be replicated until it has been configured accordingly. The configuration of the guests will be covered in **Story 1 Scenario 4.**

Add EvalCluster and HyperV02 to System Center 2012 VMM SP1

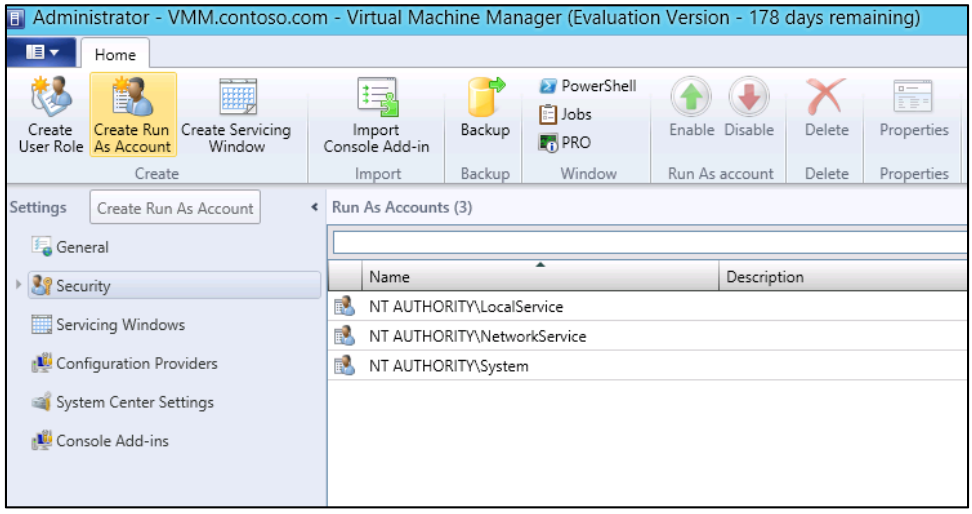
1. Login to the **VMM** server using the **Contoso\Jeff** account that was previously created and start **Virtual Machine Manager**



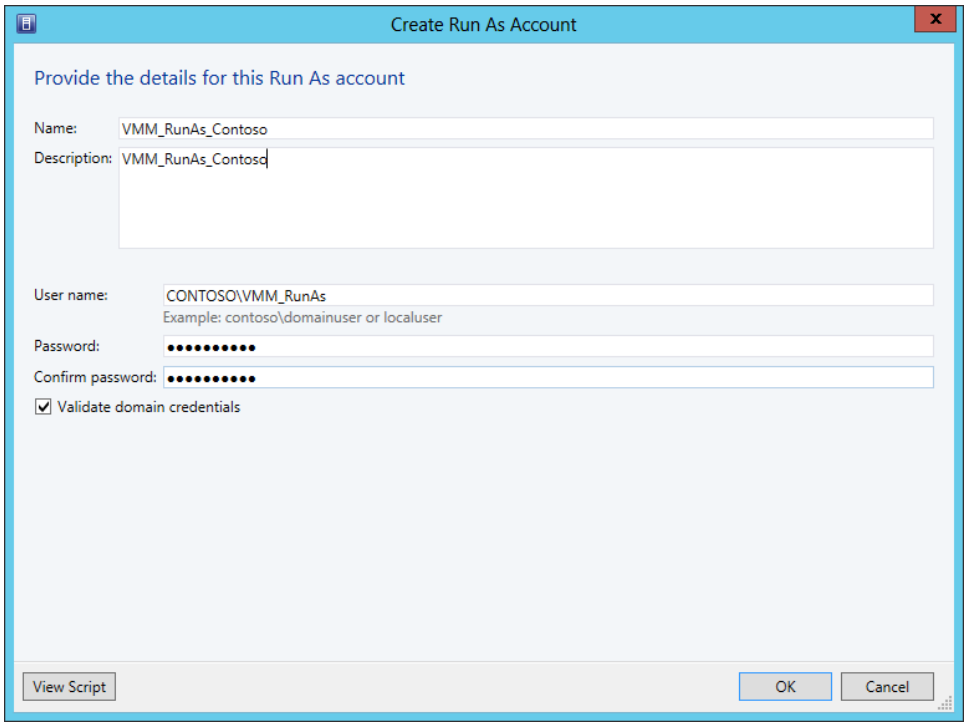
2. In the Virtual Machine Manager console click on **Settings** in the left hand column



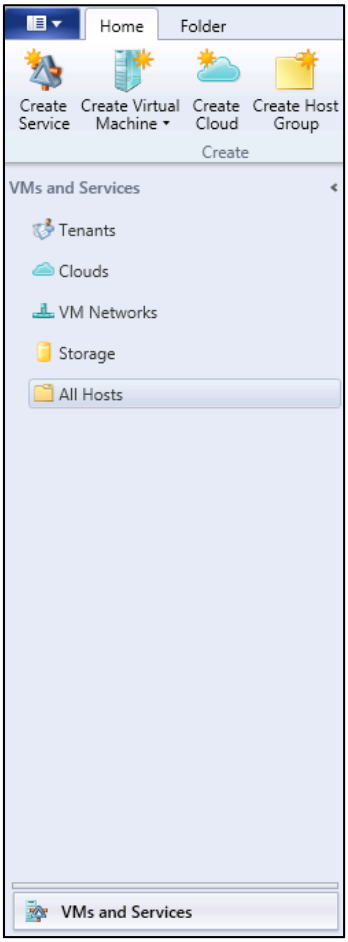
3. Click on **Create Run As Account** in the top ribbon (if not there, expand **Security**, Click on **Run As Accounts**, click on **Create Run As Account**)



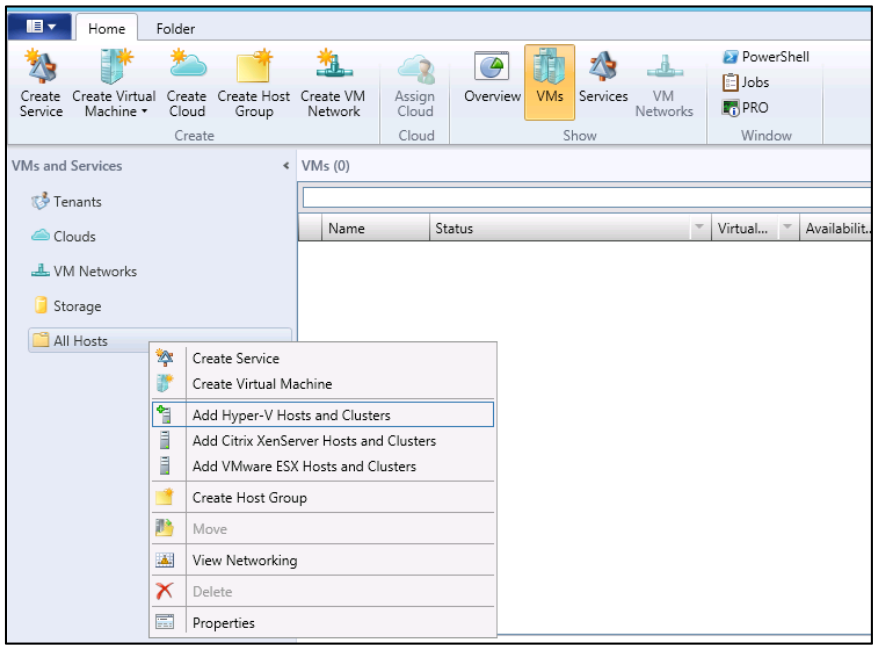
4. For this **Create Run As Account** we will be using the previously created Domain Admin contoso\VMM_Runas. For the **Name** and **Description** type VMM_RunAs_Contoso. The **User name** will be **contoso\VMM_Runas** with the **password** set to **pass@word1**, click on **OK** and the account is now created



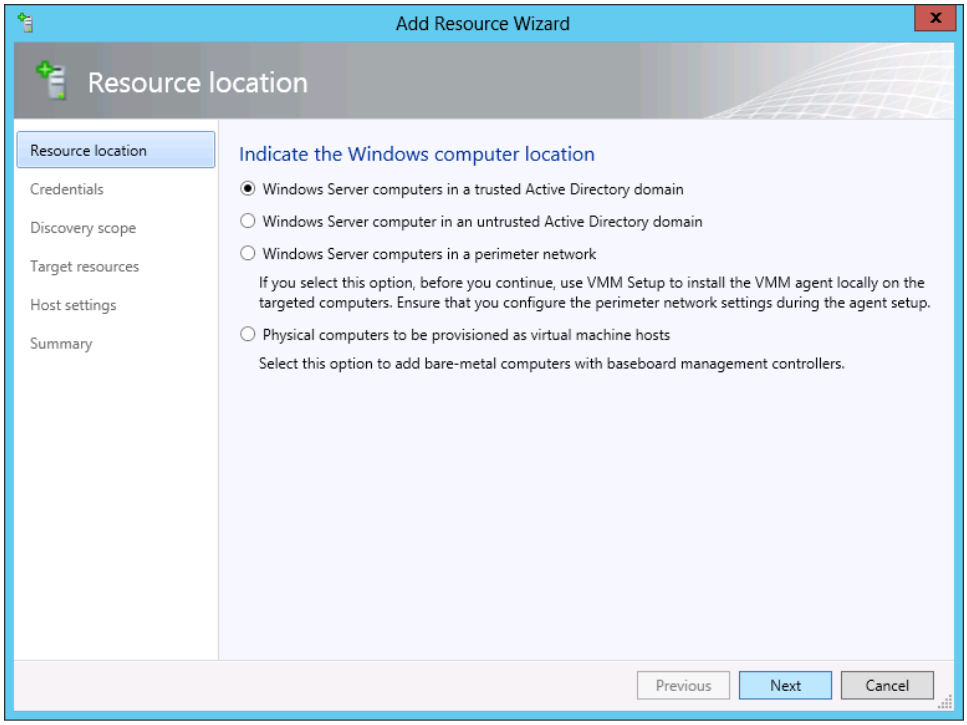
5. From the **VMM** server log in using the **contoso\Jeff** account and start the **Virtual Machine Manager** console. Click on **VMs and Services** in the left column



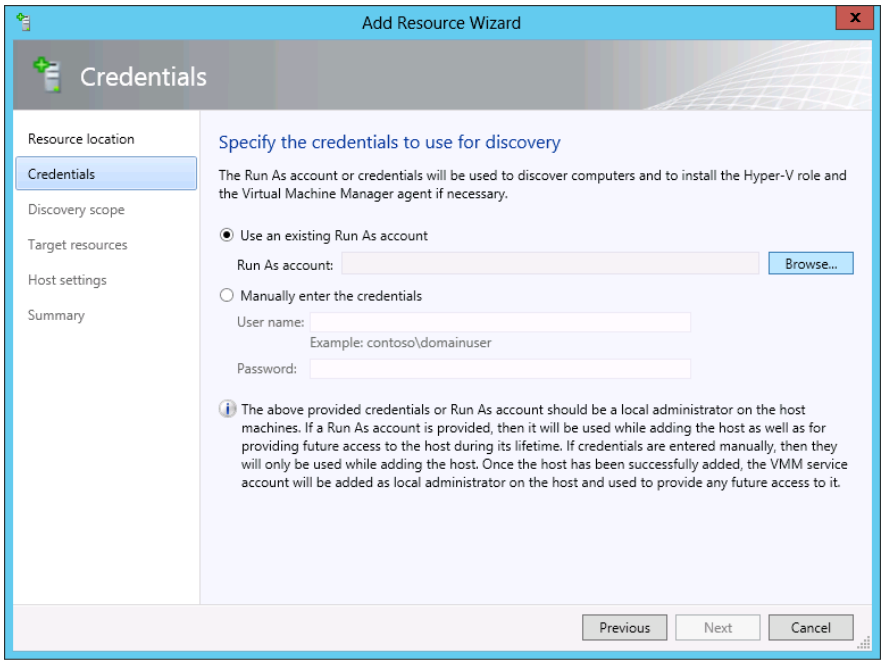
6. Right click on All Hosts and choose the add Hyper-V Hosts and Clusters option



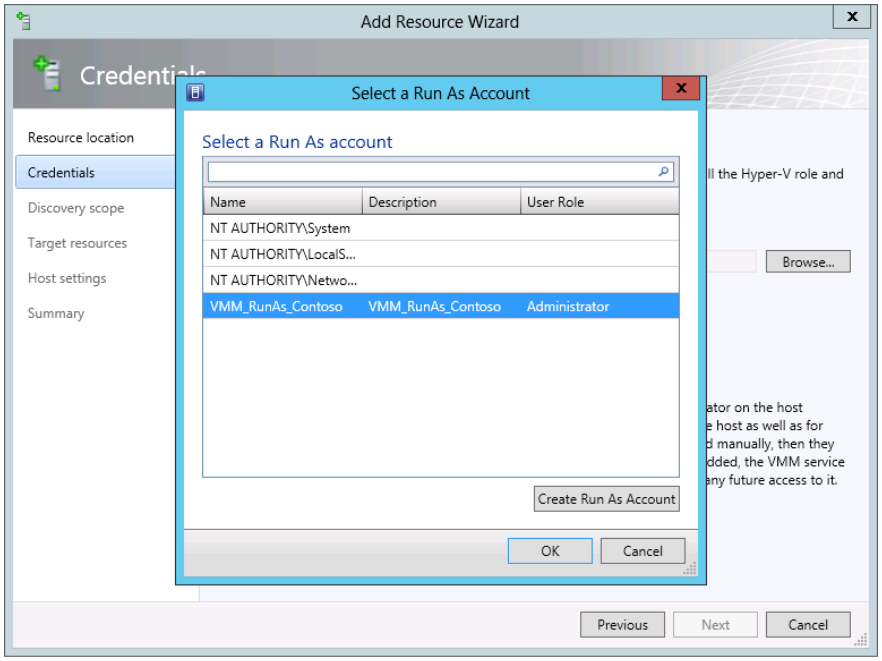
7. On the **Add Resource Wizard** ensure the Windows Server computers in a trusted Active Directory domain is chosen and click Next



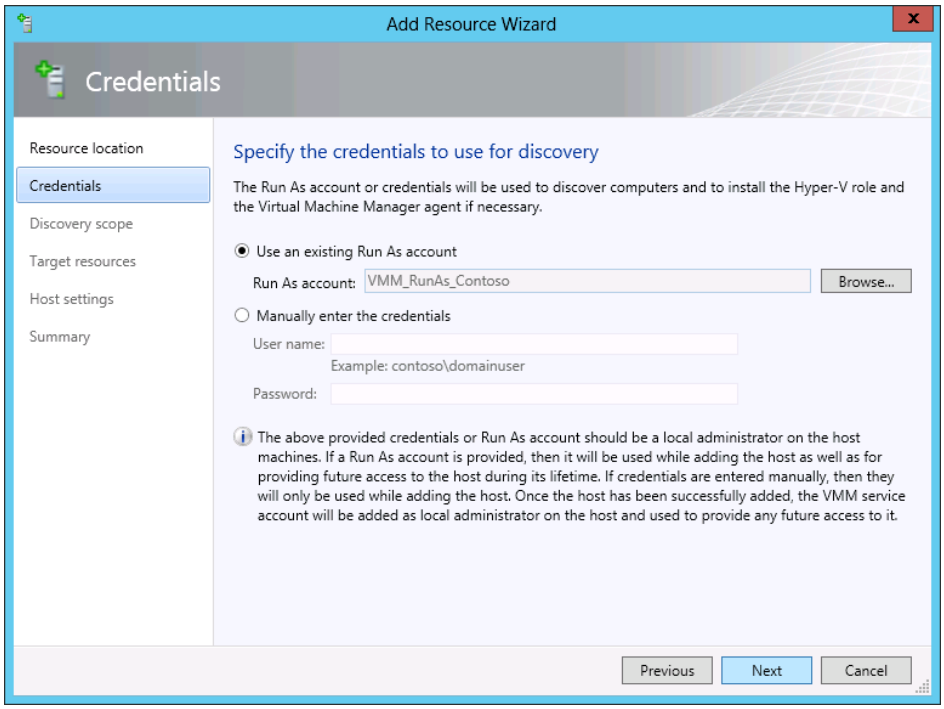
8. On the Credentials screen ensure Use an existing Run As Account is checked and click on Browse



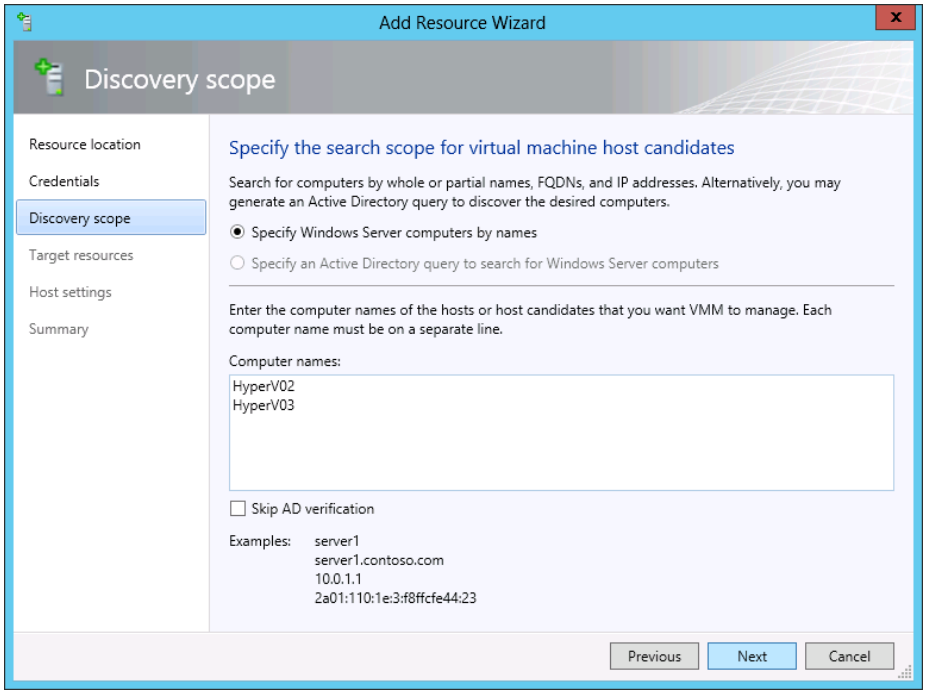
9. On the Select a Run As account options choose the user VMM_RunAs_Contoso and hit OK



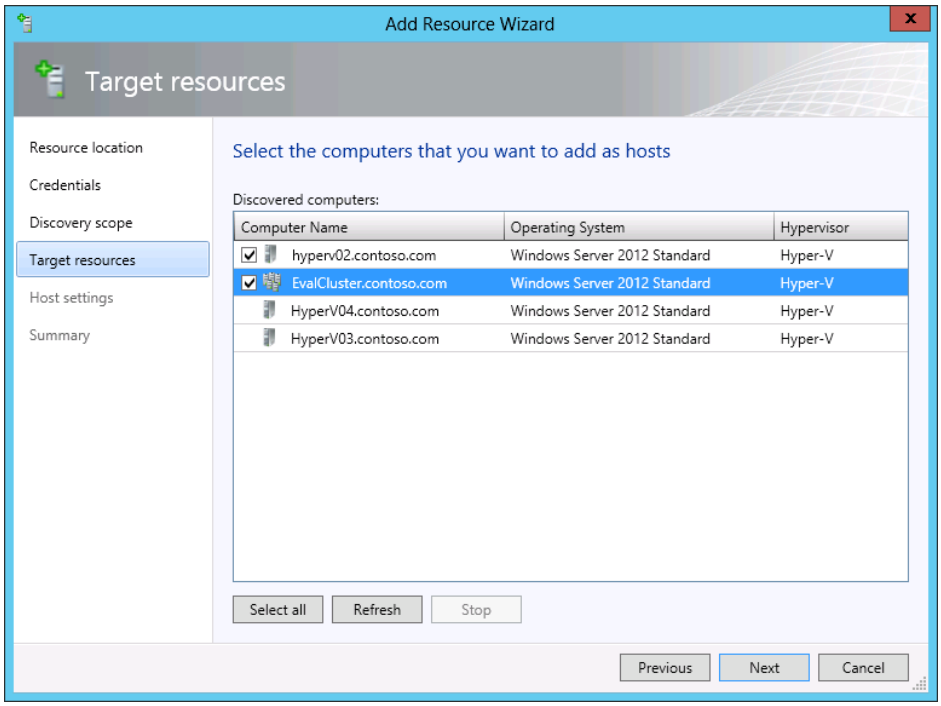
10. Choose **Next** to ok the Run As account



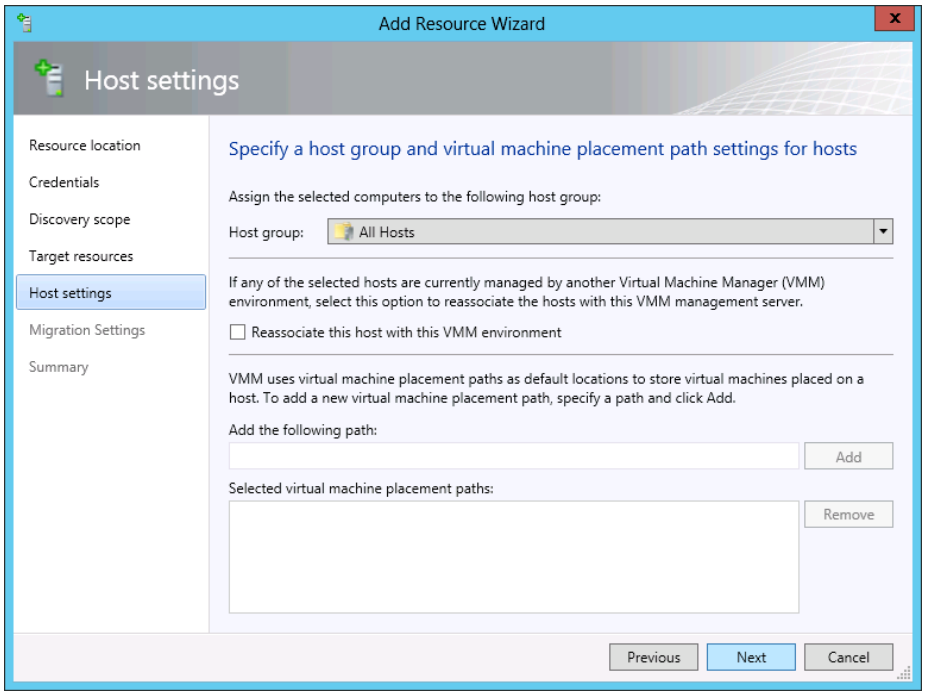
11. In the **Discovery scope** screen add the servers **HyperV02** and **HyperV03** and click **Next**



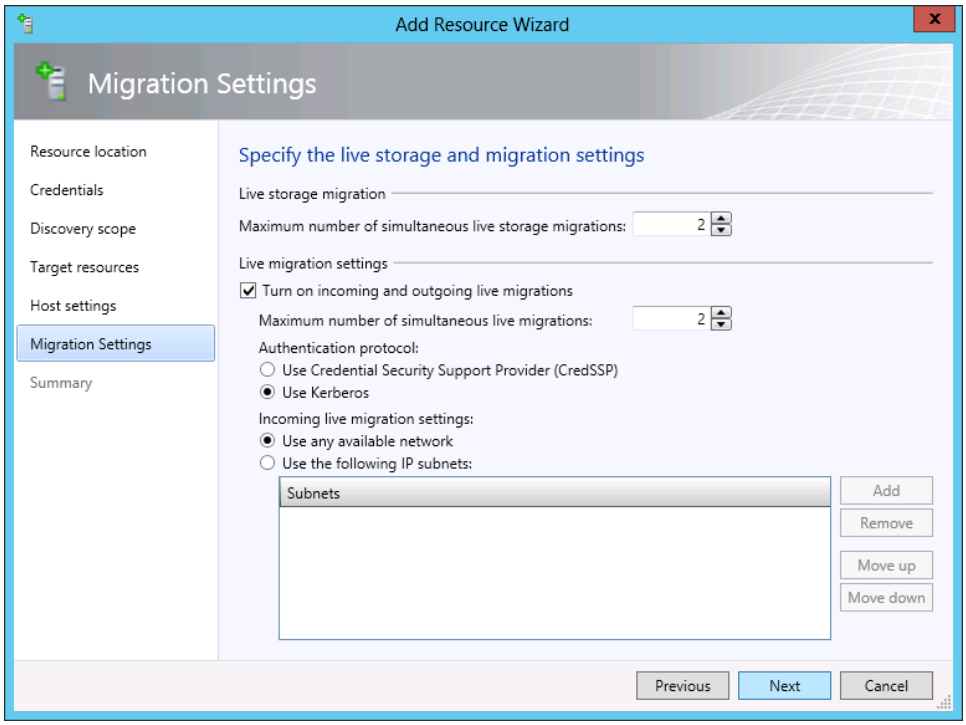
12. The **Virtual Machine Manager** will now confirm the computer names entered. Notice in this example that although **HyperV03** was selected as a single host, **Virtual Machine Manager** detects it as a **Cluster** and allows you to select it as a whole entity. Tick the boxes for both **HyperV02** and **EvalCluster** and click **Next**



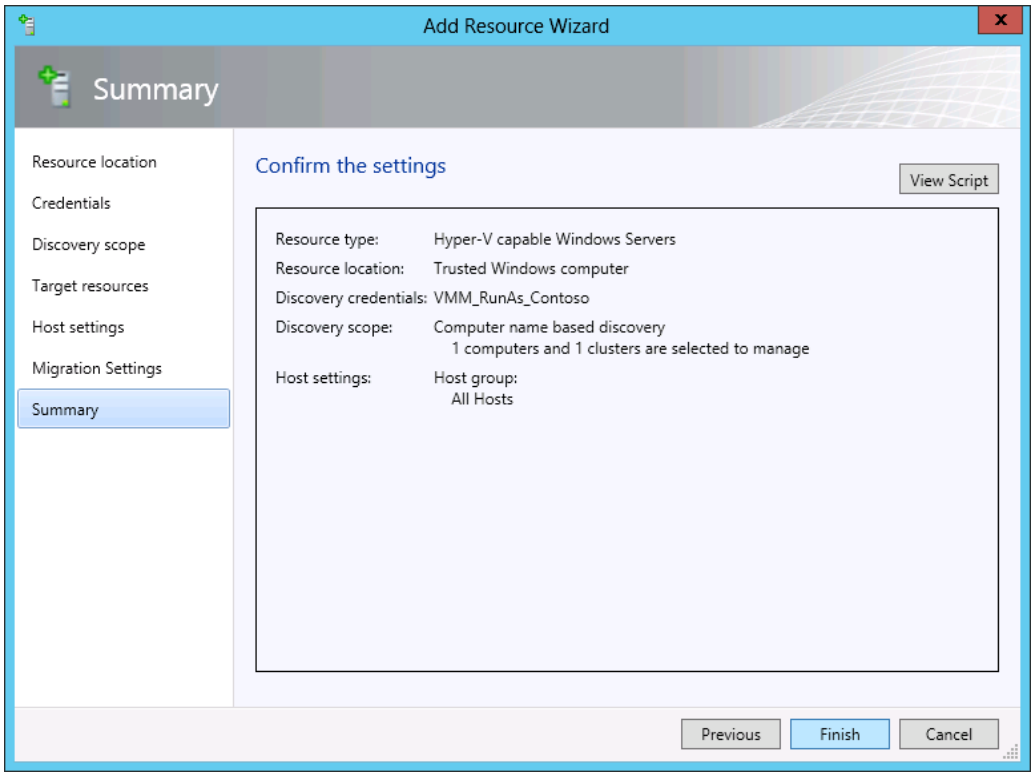
13. On the **Host Settings** options just click **Next**



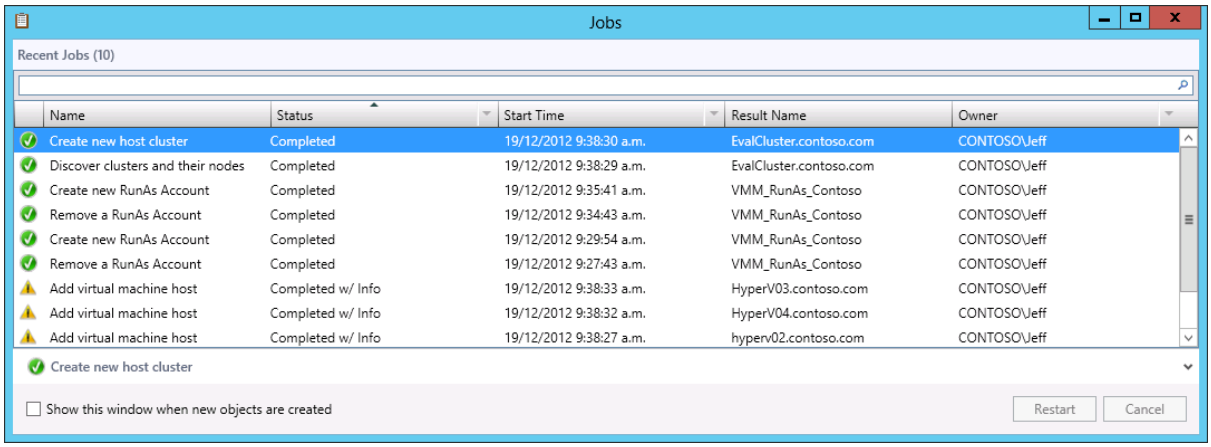
14. On Migration Settings tick the Turn on incoming and outgoing live migrations. Tick Use Kerberos and click Next



15. Review the **Summary** page and click **Finish**



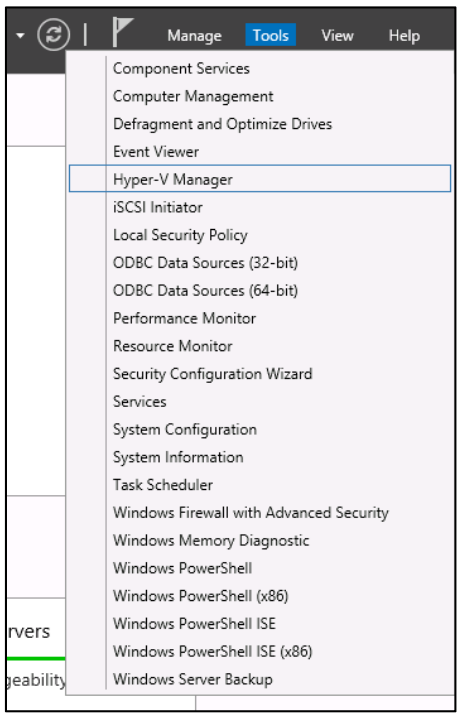
16. You will now see the jobs running and complete. In this example you will receive warnings around Multipath IO not being enabled; this is to be expected and can be ignored. Close this window



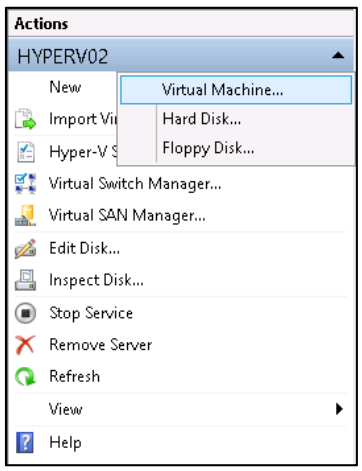
Create a Virtual Guest on HyperV02

Log onto **HyperV02** as an **administrator**, copy the previously downloaded Server 2012 Evaluation VHD from Microsoft website (http://technet.microsoft.com/en-nz/evalcenter/hh670538.aspx) to an easily accessible location. In this example we are using the folder **C:\VirtualMachines**

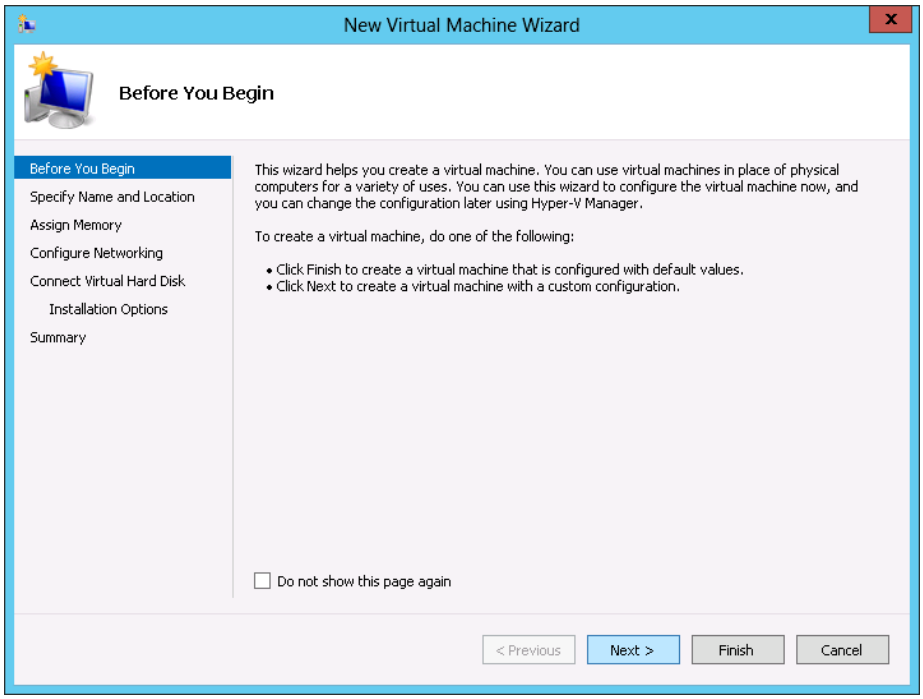
1. On HyperV02, open the Hyper-V Manager console from Server Manager



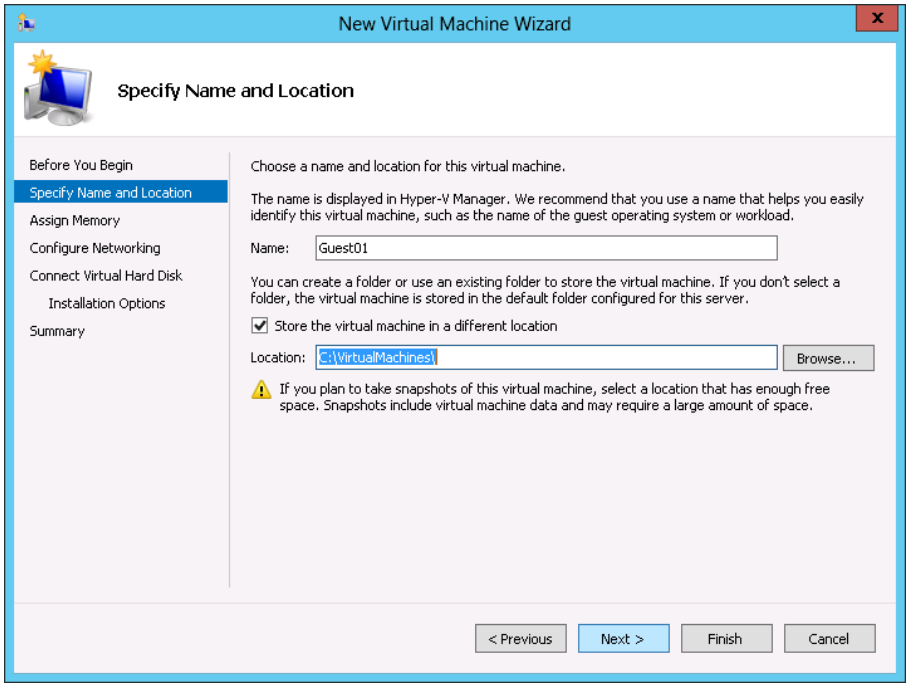
2. On the **Actions** column (right hand side), click on **New**, then select **Virtual Machine…**



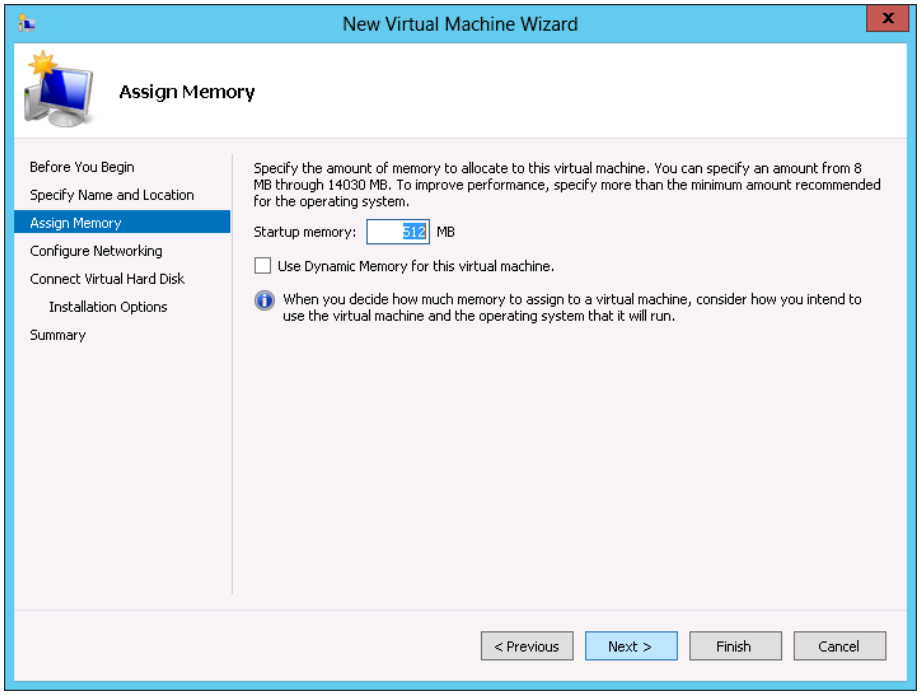
3. In the **Before You Begin** screen, select **Next**



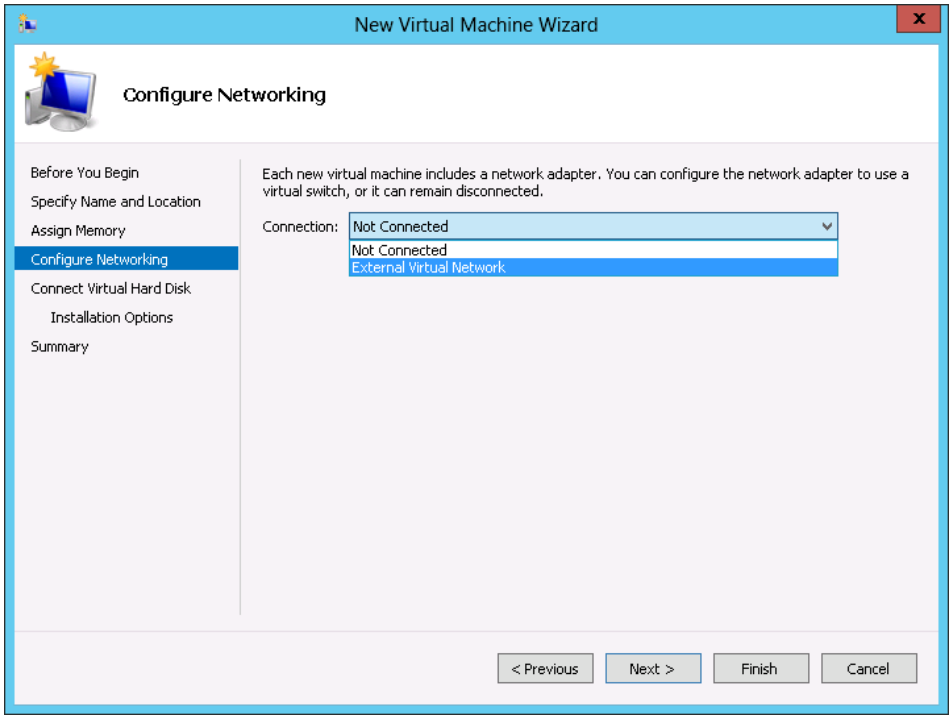
4. On the Specify Name and Location window, in the Name: box input Guest01. Tick Store Virtual Machine in a different location, and in the Location: box, input C:\VirtualMachines. Select Next



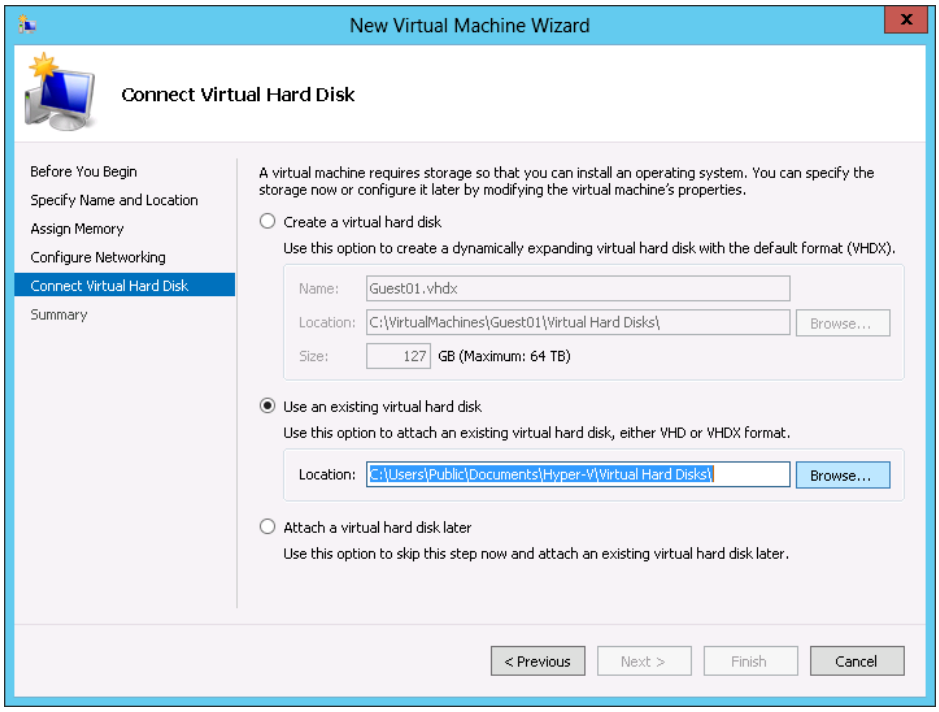
5. On the **Assign Memory** window leave all settings at **default** (do not change default of **512MB** or tick the box **Use Dynamic Memory for this Virtual Machine**). Select **Next**



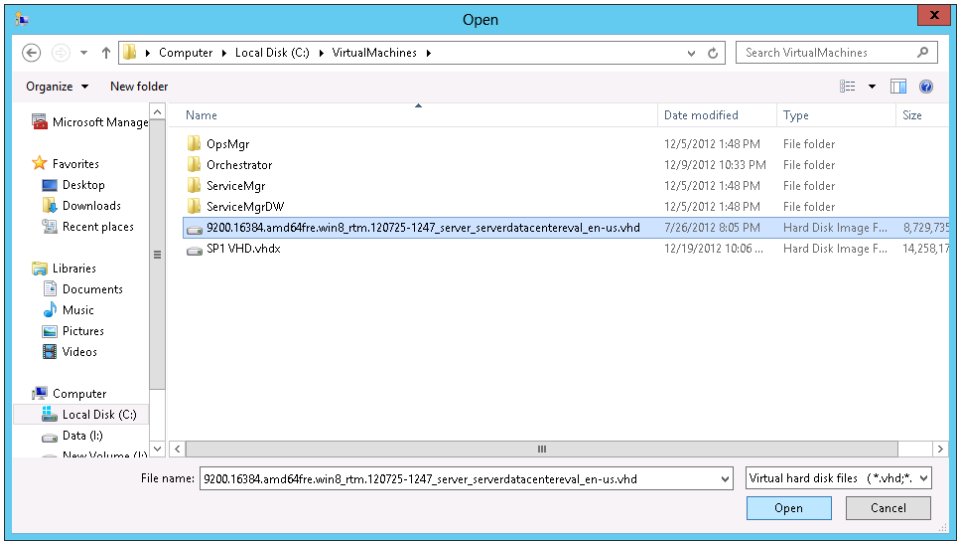
6. On the **Configure Networking** window, select the network **External Virtual Network** from the drop down box. Select **Next**



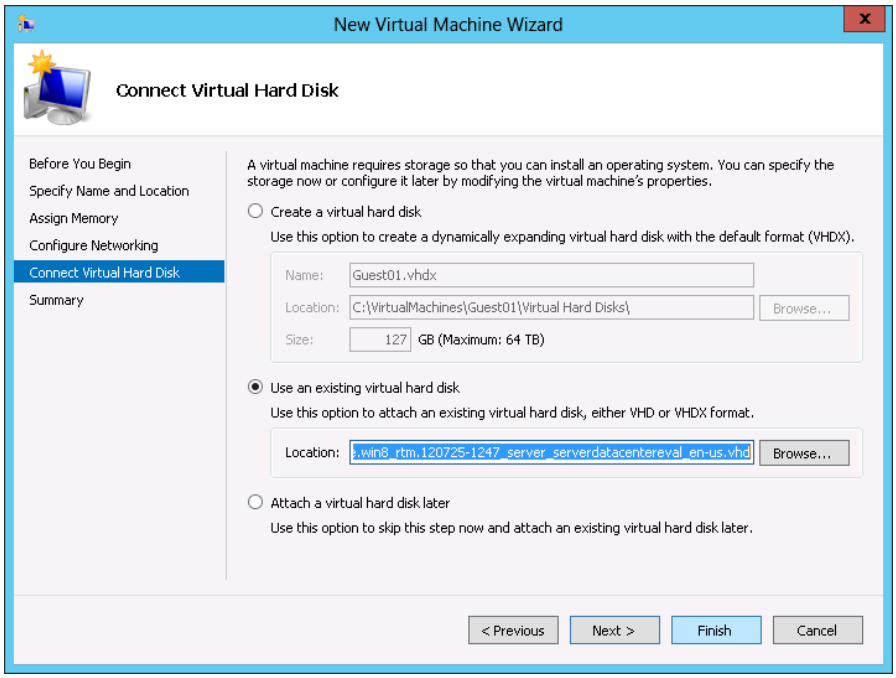
7. Tick the Use an existing virtual hard disk, then select Browse…



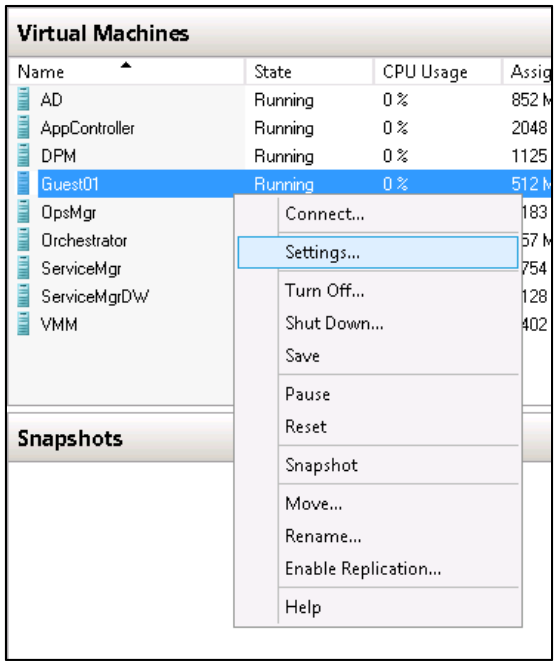
8. Browse to **C:\VirtualMachines**, select the Microsoft Windows server 2012 Evaluation VHD file, then select **Open**



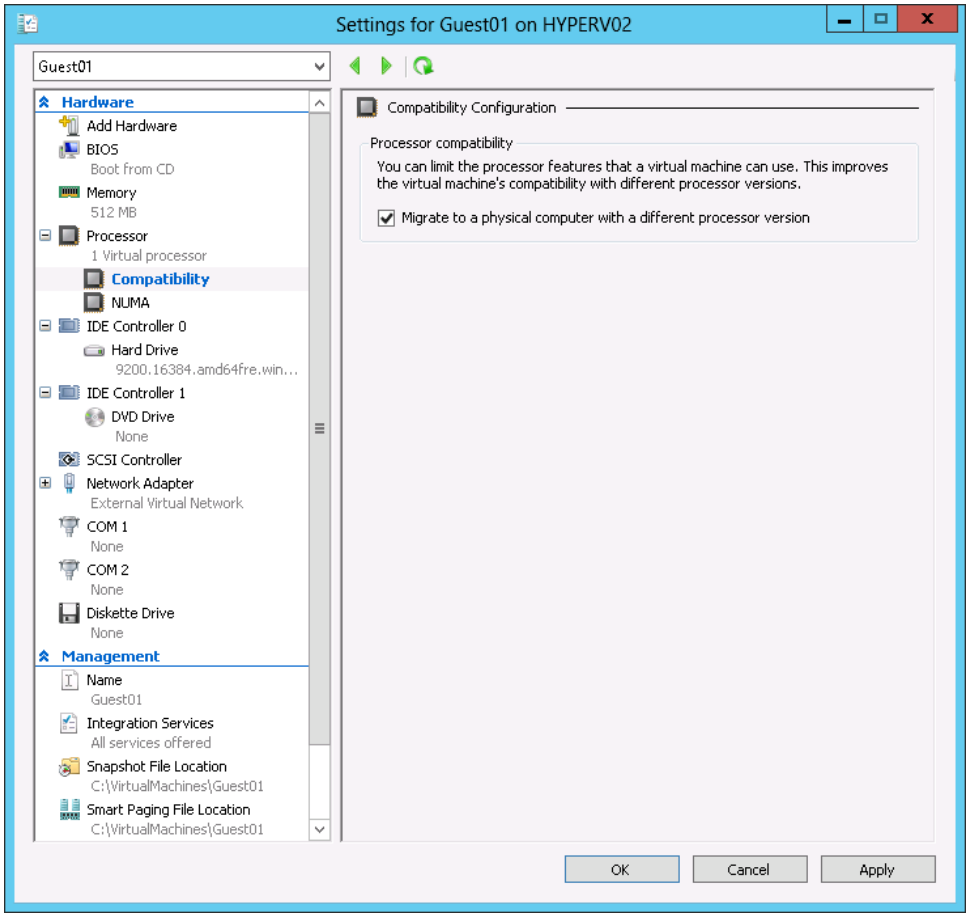
9. Select **Finish**



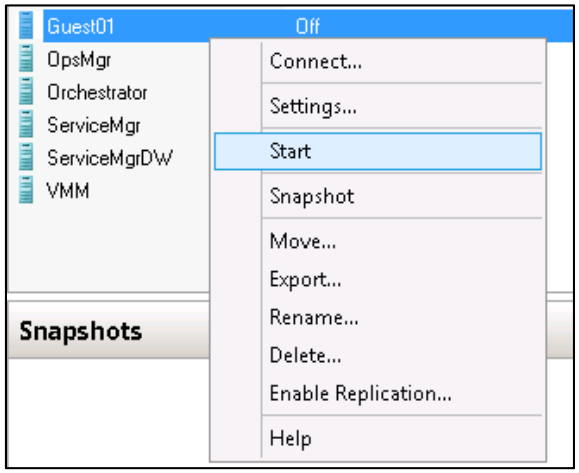
10. Within **Hyper-V Manager** on **HyperV02** right click **Guest01** and then click on **Settings...**



11. Within the Window Settings for Guest01 on HYPERV02, under the Hardware column expand Processor, select Compatibility, and tick the box Migrate to a physical computer with a different processor version. Click on OK

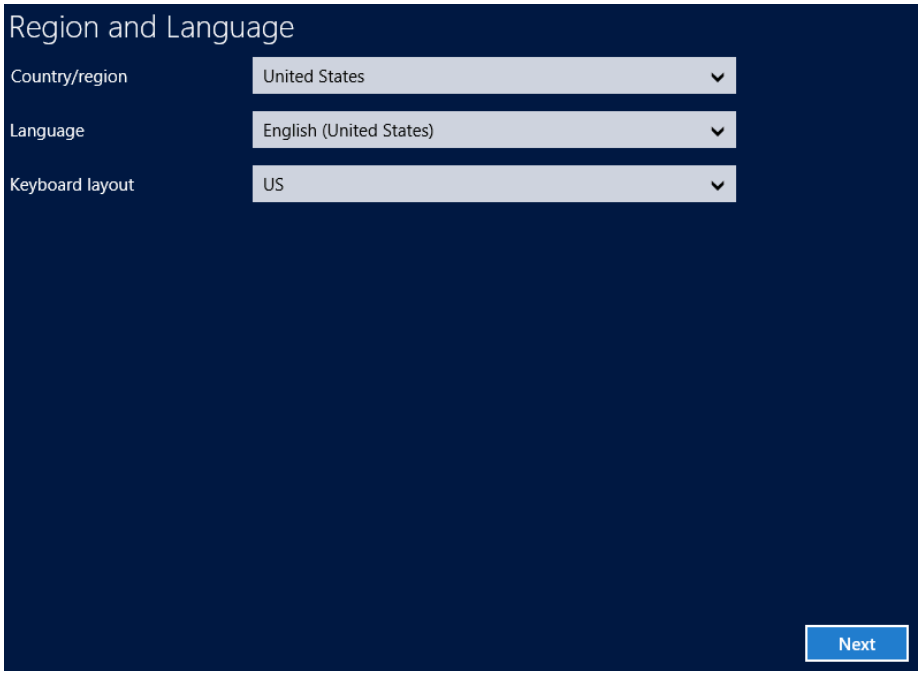


12. Within **Hyper-V Manager** on **HyperV02** right click **Guest01** and then click on **Start**

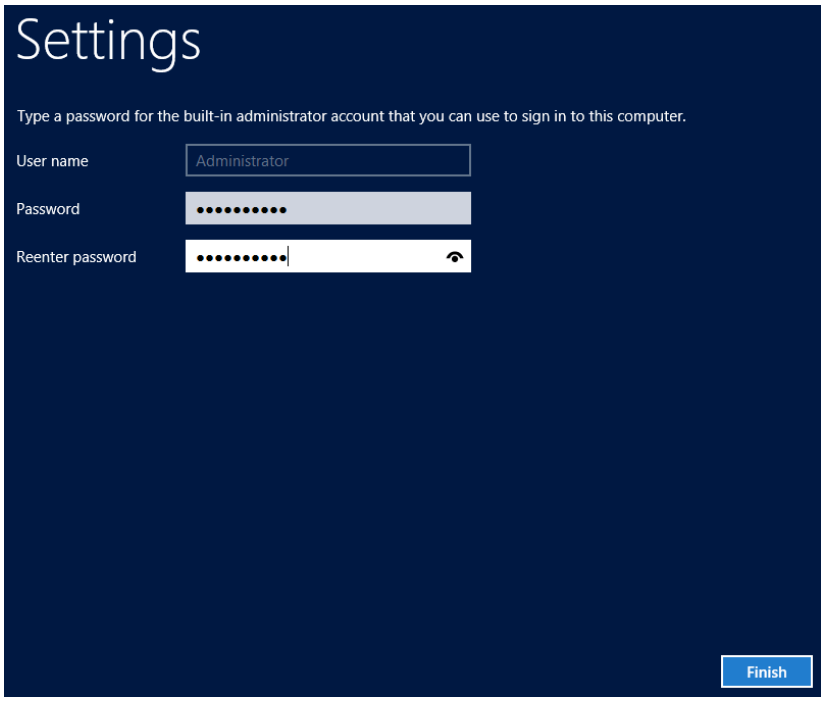


13. Connect to **Guest01** by double clicking its name in **Hyper-V Manager**. Wait for OS to load and the License Agreement to be displayed, tick **I accept the license terms for using Windows**. Then click on **Accept**

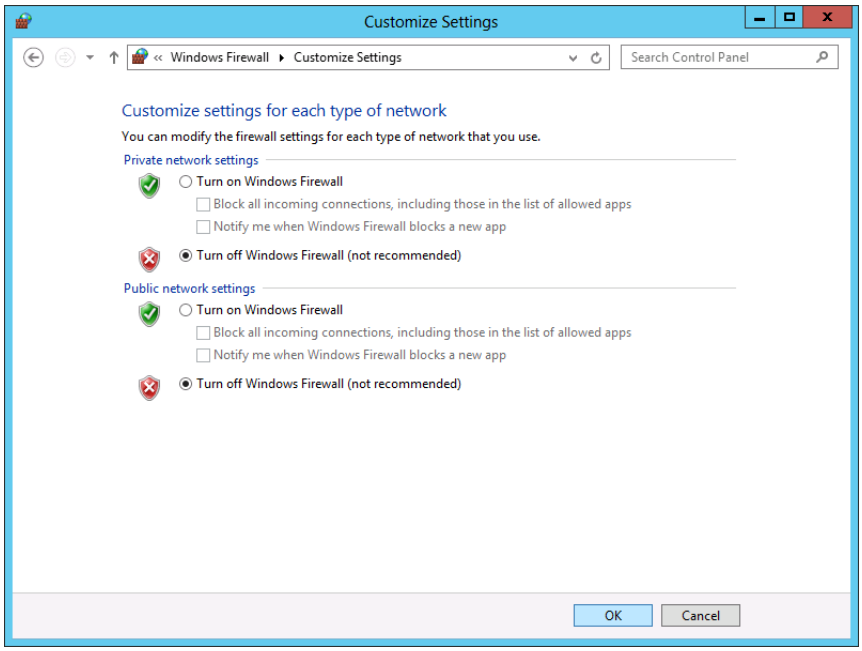
14. Select the correct **Regional and Language** settings, select **Next**



15. Input the password **pass@word1** twice, then click on **Finish**



16. Log into the **Guest01**, open **Windows Firewall** and disable the firewall for **Private** and **Public**. Click on **OK**



17. Log off of **Guest01**

Install SQL 2012 SP1

All the experiences in this guide will use SQL Server 2012 SP1 in their scenarios. To install SQL Server 2012 complete the following instructions:

1. Log onto Guest01 as GUEST01\ADMINISTRATOR and the password pass@word1
2. Install **SQL 2012 SP1** on the **Guest01** virtual machine from the media downloaded previously. For further information on the prerequisites and installing **SQL Server 2012** refer to the following articles on MSDN:

Reference Links	
SQL Server prerequisites	http://msdn.microsoft.com/en-us/library/ms143506(SQL.110).aspx
SQL Server installation	http://msdn.microsoft.com/en-us/library/bb500395.aspx

3. Ensure the following roles and features are installed during the SQL Server installation
    a. Database services
    b. Management Tools – Complete

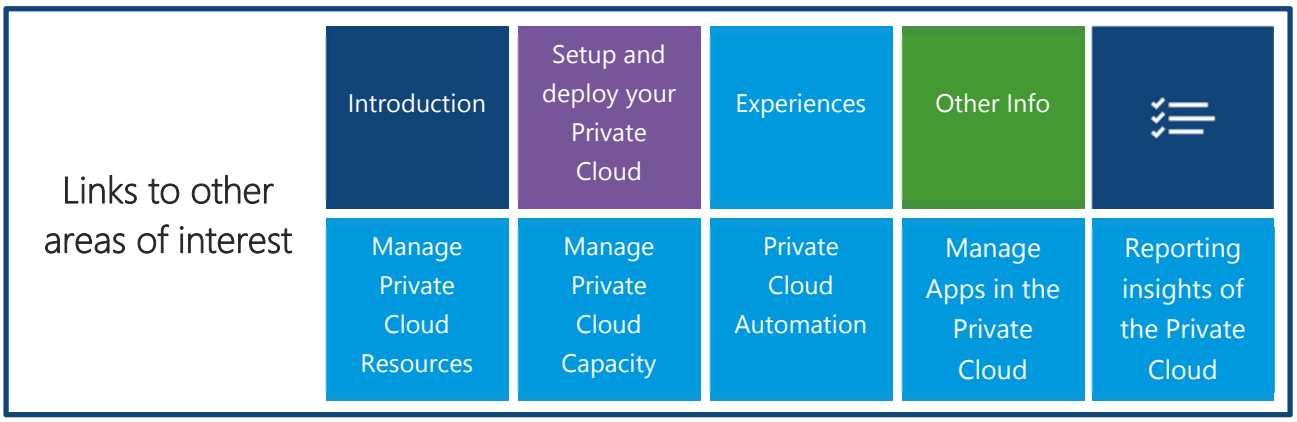
Synopsis

Jeff has now configured his infrastructure to take advantage of many of the new and exciting features of Server 2012 and System Center 2102 SP1. So far he has accomplished the following:

- Set up a new failover cluster which will give him the ability to create Highly Available virtual infrastructure
- Configured a connection broker which will allow him to configure Hyper-V Replicas of his virtual machines
- Configured a SMB 3.0 share for storing virtual machine operating files
- Configured Virtual Machine Manager to manage his virtual infrastructure
- Built a virtual guest to host new applications and services

These roles and features that Jeff has configured will allow him to start taking advantage of many of the new capabilities available when Windows Server 2012 and System Center 2012 SP1 come together. The following scenarios will walk Jeff through step by step guides of how to configure and utilize features such as:

- Shared Nothing Live Migration
- Storage Pooling using Just a Bunch Of Disks (JBOD)
- Business Continuity through High Availability and Hyper-V Replica
- Network Traffic management through Quality of Service in Hyper-V



Experiences

The following experiences are designed to help you experience the key features of System Center 2012 SP1 and Windows Server 2012. They highlight the most important new functionality of System Center and take you through how you might use these features in your own network.

The experiences are designed to be consumed as you wish- you can go through them systematically or start with the one that intrigues you the most and try out others as you wish. They can be done in any order, at any time. You will want to ensure that you've installed and configured your network as described in "Setup and Deploy your Private Cloud." The steps included in "Configuring the Private Cloud Experiences" are important to be able to successfully follow the step-by-step instructions.

The five experiences are:

1. Manage Private Cloud resources
2. Manage Private Cloud capacity
3. Private Cloud automation
4. Manage Apps in the Private Cloud
5. Reporting insights of the Private Cloud

Inside each Experience are scenarios. Each of the scenarios will allow you to evaluate and identify how the System Center Private Cloud can assist in alleviating problems in the current environment, or provide you with the basis for a plan to take advantage of Windows Server 2012 and System Center 2012.

Experience 1: Manage Private Cloud resources

This experience will take you through some of the features that are provided by the Microsoft Private Cloud. These features will be invaluable in creating a robust, low cost and resilient infrastructure, while also delivering on the core tenets of the Private Cloud:

Scalable and Elastic

Shared Resources

Always Up, Always On

Prerequisites

To evaluate this experience you will require the following resources in place in your Eval Guide lab. Refer to the Scenario Prerequisites for additional information on how to configure these items:

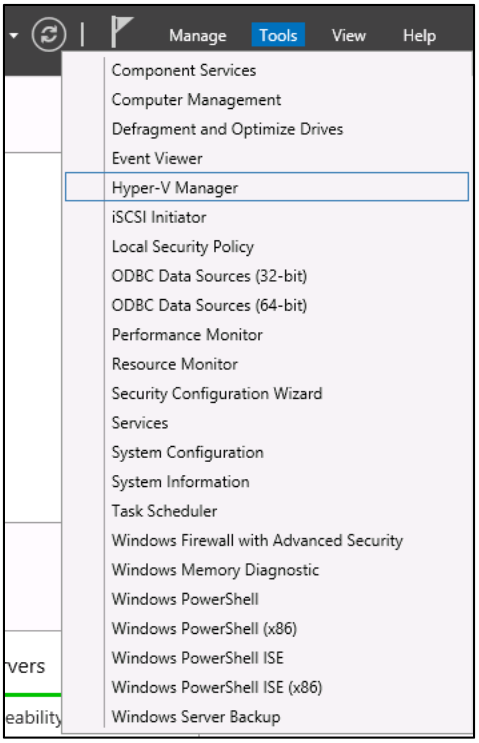
- The **HyperV** role installed on **HyperV02**, **HyperV03**, and **HyperV04**
- An SMB3.0 share on HyperV02
- A **Failover Cluster** installed and configured on **HyperV03** and **HyperV04**
- The Hyper-V Cluster Broker installed and configured on the Failover Cluster
- **Guest01** prebuilt and running on **HyperV02**

Scenario: Verifying VM storage infrastructure

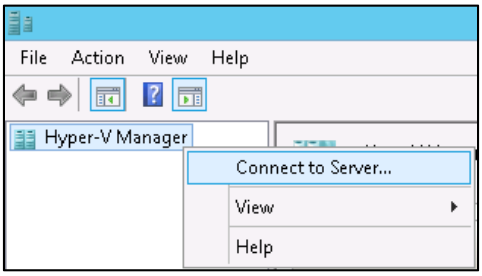
Jeff has identified that his virtual infrastructure appears to require alteration to take advantage of the new features of the private cloud he is implementing. He investigates the current configuration of his virtual infrastructure to see where he can best implement these features.

In this scenario Jeff will take a look at the resources of **Guest01** which is currently hosted on **HyperV02**. This is typical non-highly available Hyper-V installation running local, non-shared direct attached storage. In this scenario Jeff will confirm the location of the storage for **Guest01**

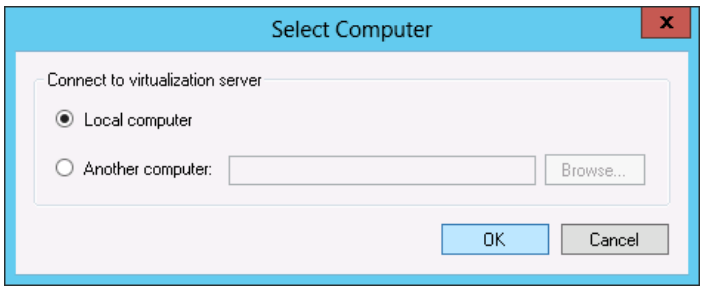
1. From **HyperV02** open **Server Manager** and confirm that the **Dashboard** is open. From the **Tools** menu items, select **Hyper-V Manager**



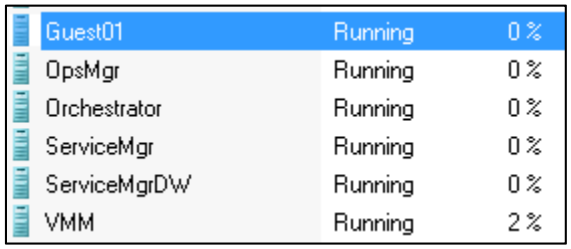
2. If HyperV02 is not listed under Hyper-V Manager, right click Hyper-V Manager, then select Connect to Server…

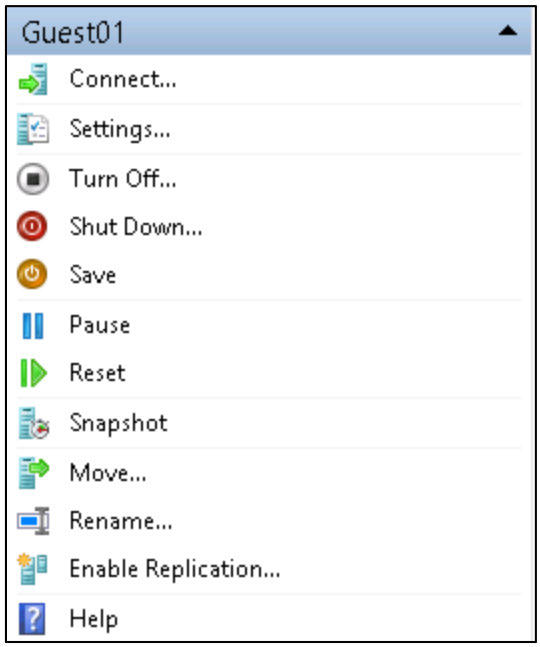


3. Tick Local computer, select OK

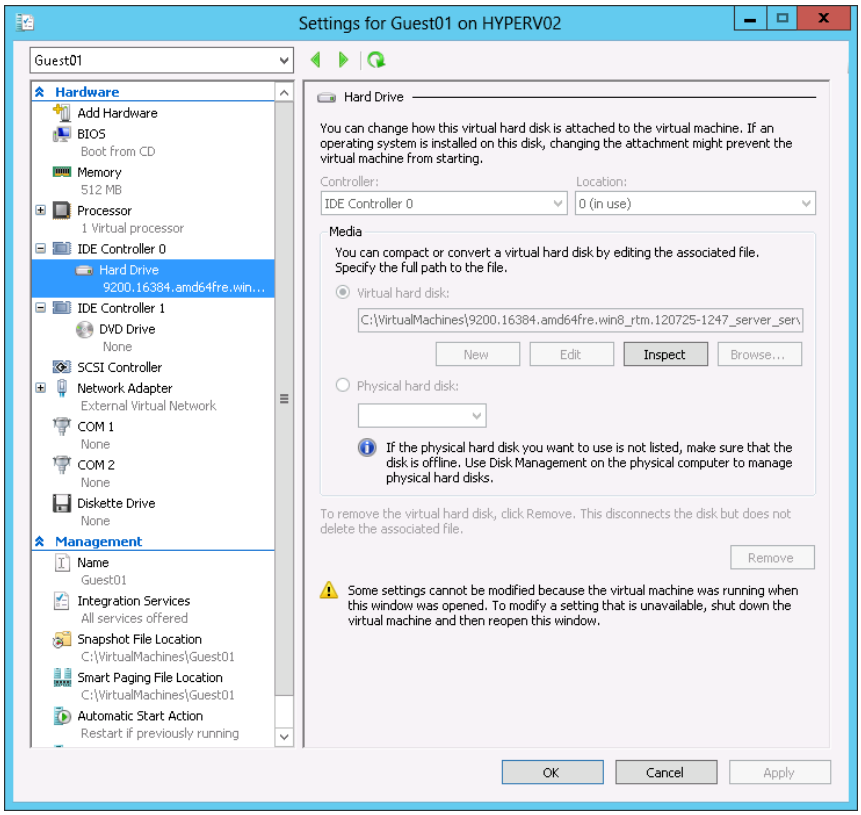


4. This will return you to the **Hyper-V Manager.** Select the Virtual Machine **Guest01**, and in the right hand column click on **Settings…**





5. Expand **IDE Controller 0** in the **Hardware** column and select **Hard Drive**



6. Take note of the path under **Virtual hard disk**.

As Jeff can see the storage for this particular virtual machine is located on the local C: drive of the server **HyperV02**. In the event of an interruption to service, this virtual machine would not be available, and in the case of a hardware failure, the data may well be lost. Through the following steps Jeff will migrate the storage of **Guest01** to a shared storage platform and host the virtual machine on a Microsoft Failover Cluster to take advantage of the Availability and Resiliency that clustering gives.

Scenario: Migrating VM storage with SMB 3.0

Debbie has discussed with Jeff the importance of a particular server in the Contoso environment that houses much of the code and development work her project team are preparing for release. Debbie needs to ensure that the data is always available to her project teams who work in several locations and in different time zones, so high availability is very important. She also needs to ensure that in the event of hardware fault, the server will continue operating.

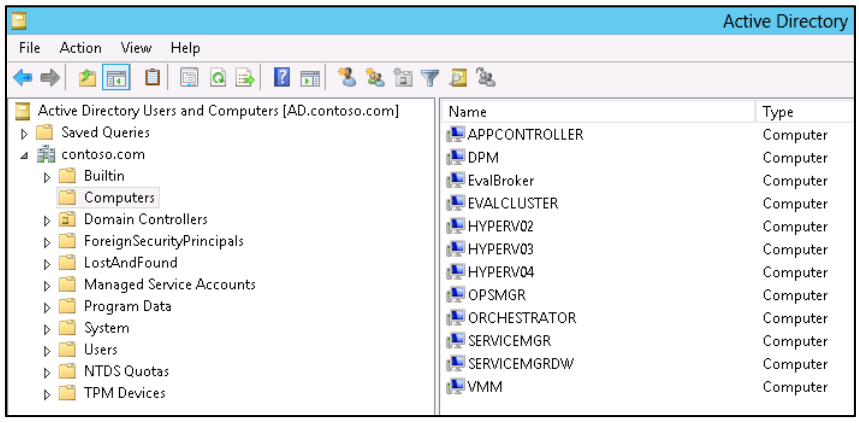
Jeff decides to migrate the server Guest01 to the cluster to ensure that the server is always on, even if one physical server fails, and also if he needs to shut down hosts for maintenance. He also decides to take advantage of the SMB 3.0 storage for the virtual machines, reducing the cost of implementing the highly available solution.

In this scenario Jeff will migrate the server **Guest01** from **HyperV02**, to the cluster **EvalCluster**. The purpose for this is to provide a stable, highly available platform that is resilient to hardware failure and loss of service. Prior to configuring this, some specific changes have to be made to the SMB 3.0 share. These changes will allow live migration to take place to and from the SMB 3.0 Share

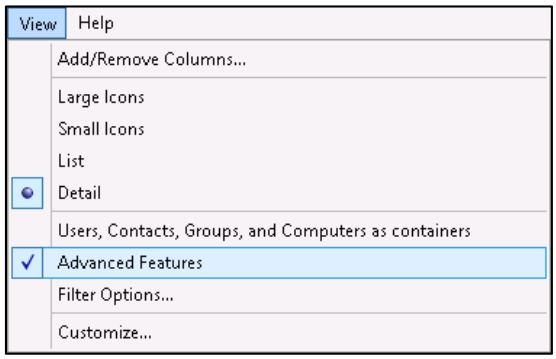
Storage of Virtual Machines on remote SMB 3.0 shares is a new feature in Hyper-V 3.0 and provides the following features:

- Ease of provisioning and management. You can manage file shares instead of storage fabric and logical unit numbers (LUNs).
- Increased flexibility. You can dynamically migrate virtual machines or databases in the data center.
- Ability to take advantage of existing investment in a converged network. You can use your existing converged network with no specialized storage networking hardware.
- Reduced capital expenditures. Capital expenses (acquisition costs) are reduced.
- Reduced operating expenditures. You can reduce operating costs because there is no need for specialized storage expertise.

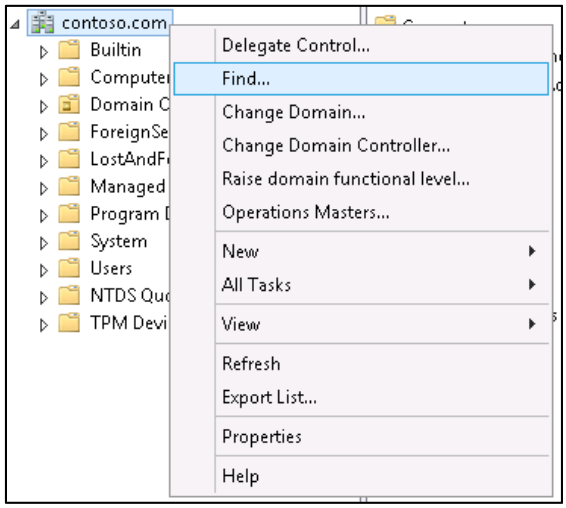
1. Connect to the Active Directory server **AD** as an administrator and load **Active Directory Users and Computers**



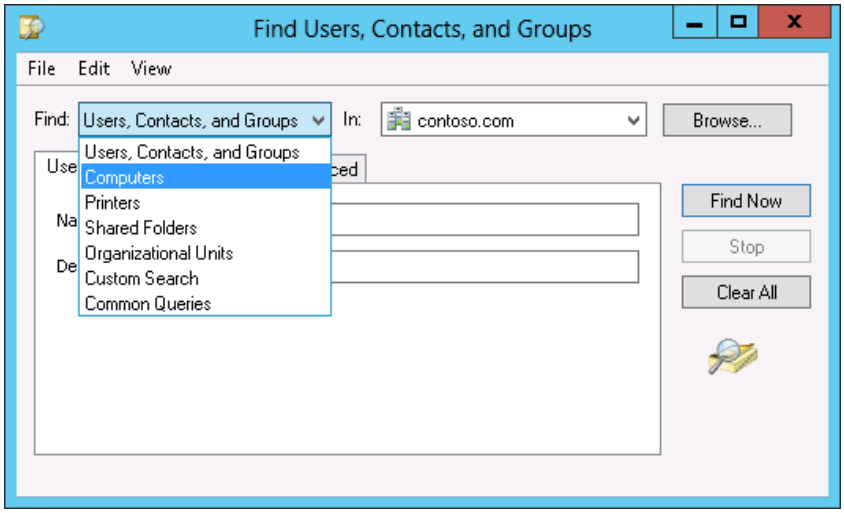
2. Click on View and tick Advanced Features



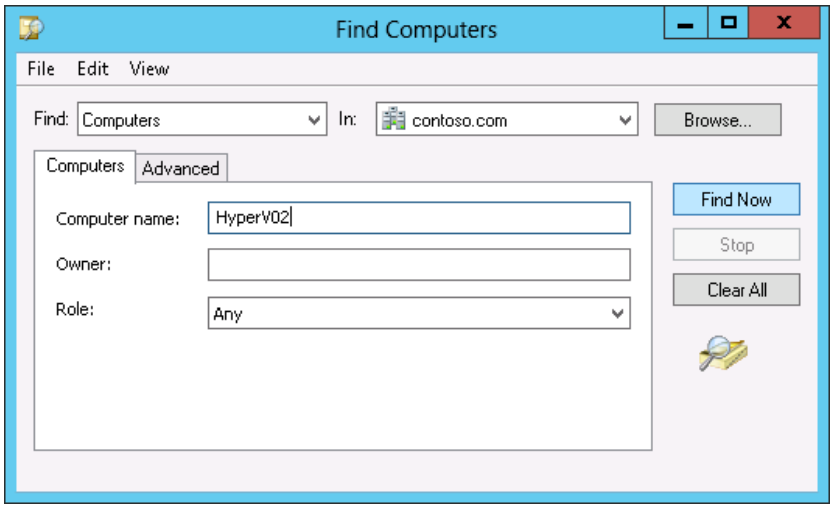
3. Next, right click the **Contoso.com** domain and select **Find**



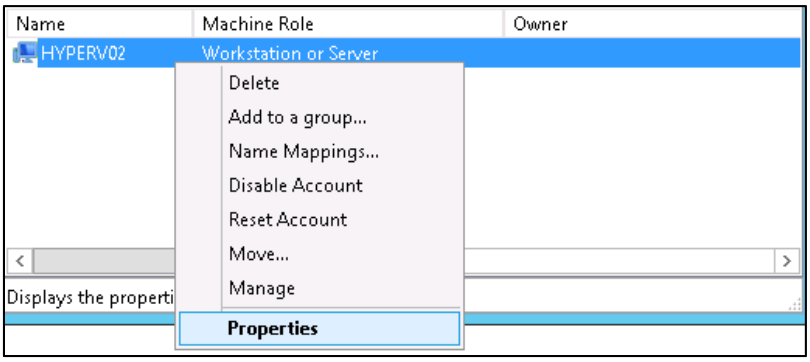
4. Change the Find: box from Users, Contacts and Groups to Computers



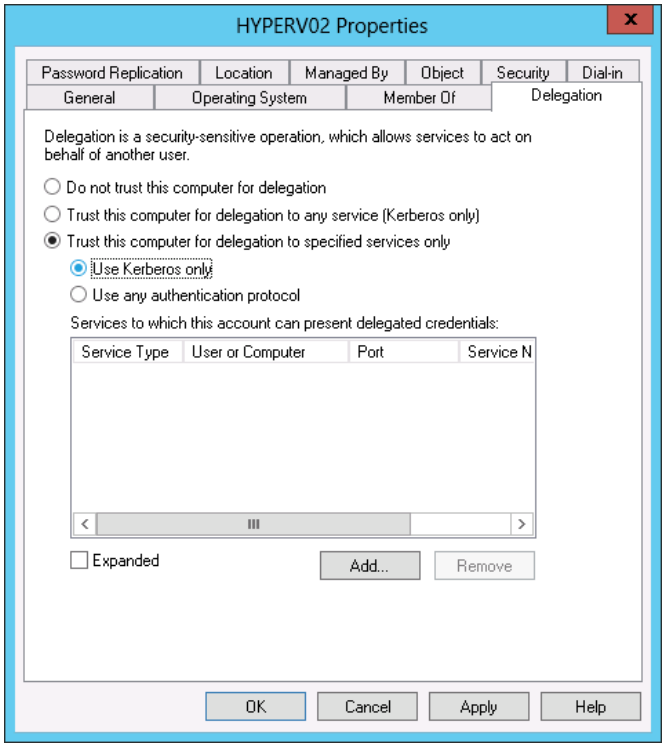
5. Type in **HyperV02**, then click on **Find Now**



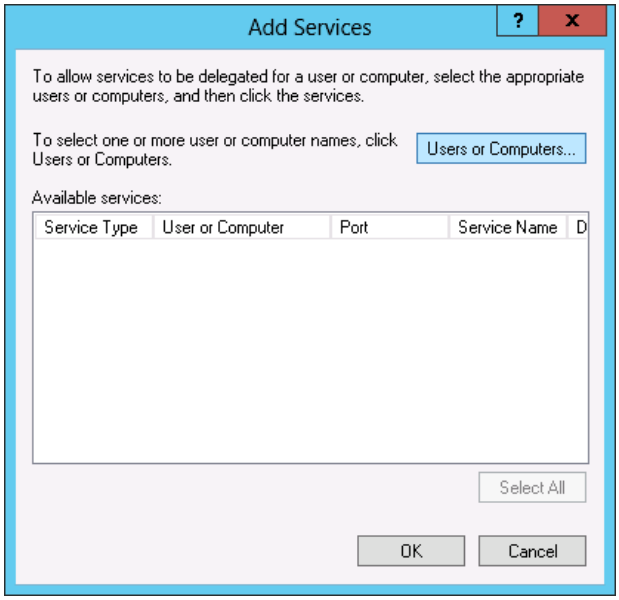
6. Right click on **HyperV02** then select **Properties**



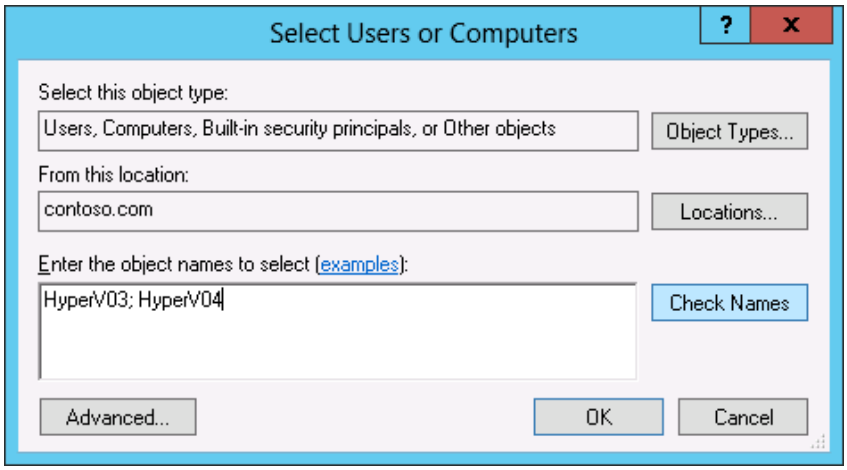
7. Click on the Delegation tab, tick Trust this computer for delegation to specified services only, and tick Use Kerberos only. Click Add...



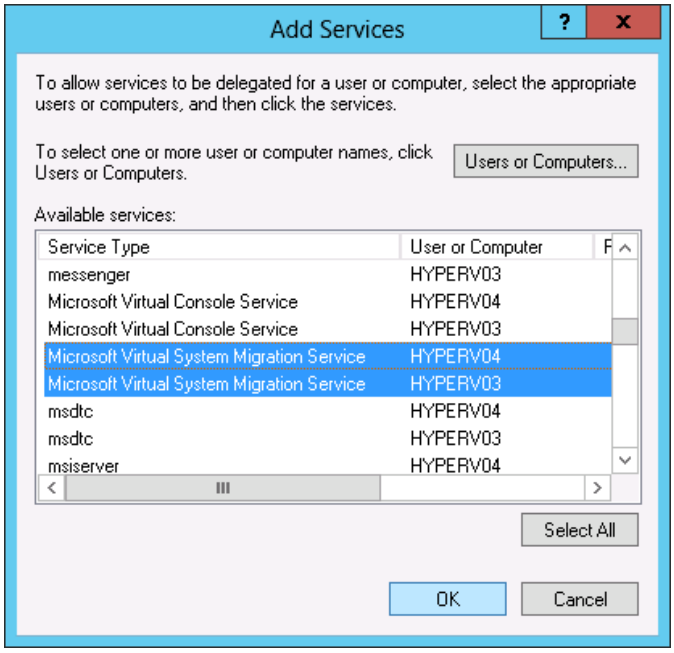
8. Click Users or Computers...



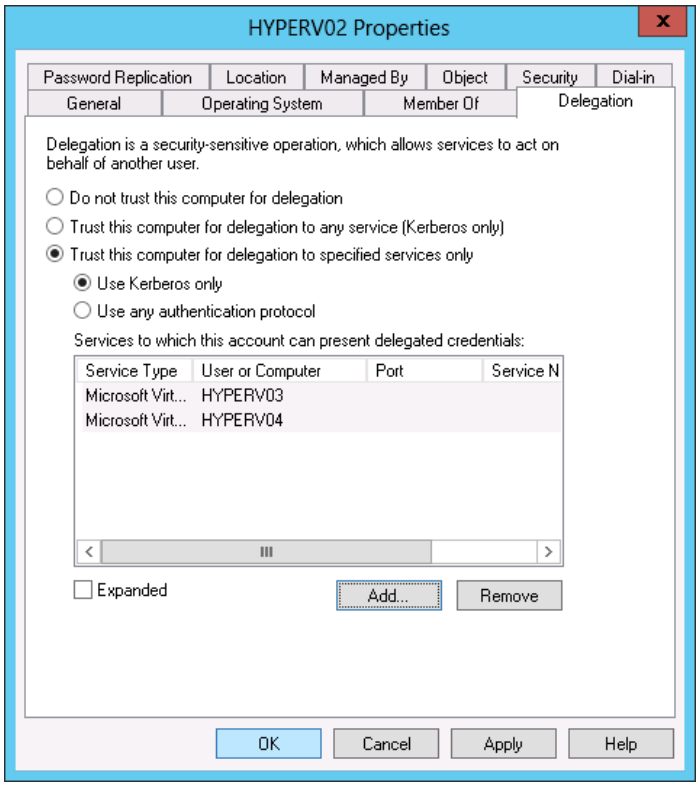
9. In the object name box, type in **HyperV03; HyperV04**, click **OK**



10. Scroll down and select **Microsoft Virtual System Migration Service** for **HyperV03** and **HyperV04** (hold the ctrl key to select multiple items), click **OK**



11. Click **OK** to close the **HyperV02** properties



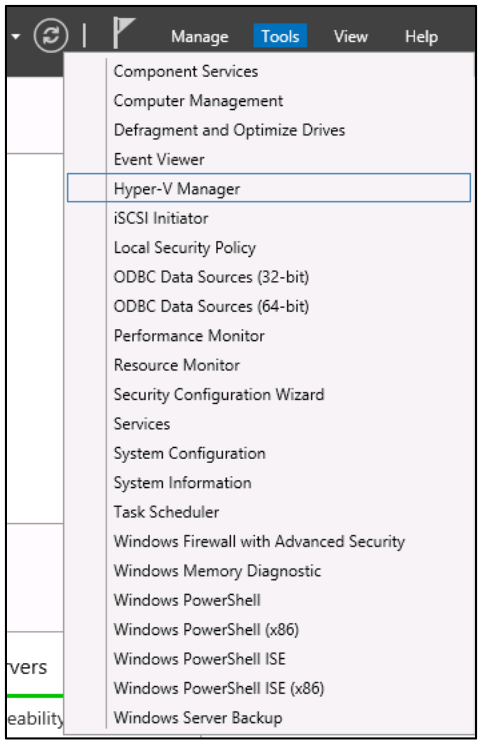
12. Repeat the above steps **5** – **11** for **HyperV03** and **HyperV04**. The following delegation roles will need to be added to each server:

13. HyperV03
    a. CIFS HyperV02
    b. Microsoft Virtual System Migration Service HyperV02
    c. Microsoft Virtual System Migration Service HyperV04

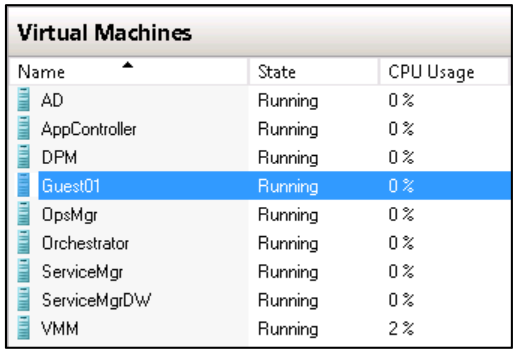
14. HyperV04
    a. CIFS HyperV02
    b. Microsoft Virtual System Migration Service HyperV02
    c. Microsoft Virtual System Migration Service HyperV03

Next Jeff will need confirmation that **Guest01** experiences no loss of service during the migration. A simple way to check this is to set up a persistent ping to **Guest01** and confirm that no pings are dropped during the transfer.

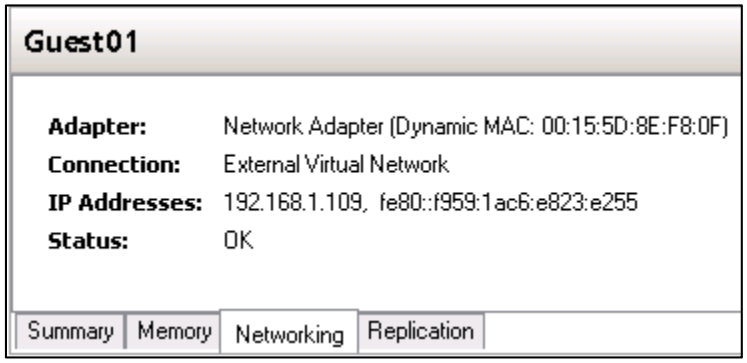
1. From **HyperV02** open **Server Manager** and confirm that the **Dashboard** is open. From the **Tools** menu items, select **Hyper-V Manager**



2. Click on Guest01 in the Virtual Machines pane

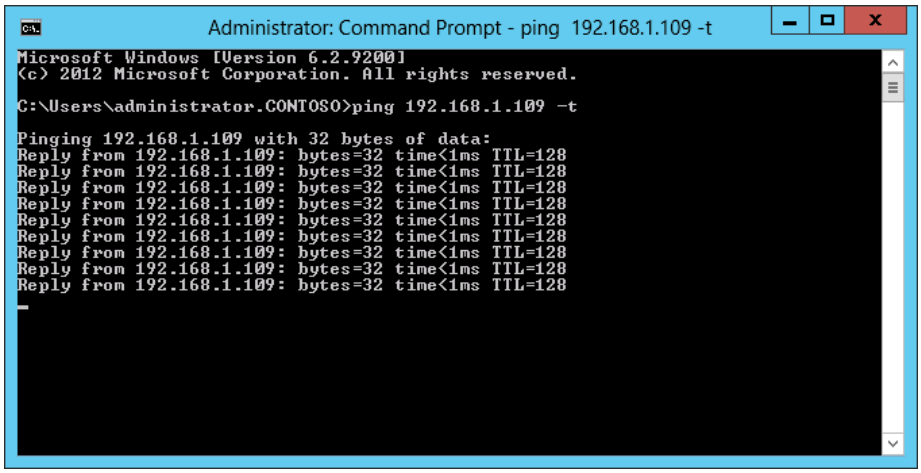


3. In the **Guest01** box at the bottom of the console, click on the **Networking** tab, and take a note of the IP address. This IP address is the network address for **Guest01** and will be used in the next step. In this example the IP address is **192.168.1.109**



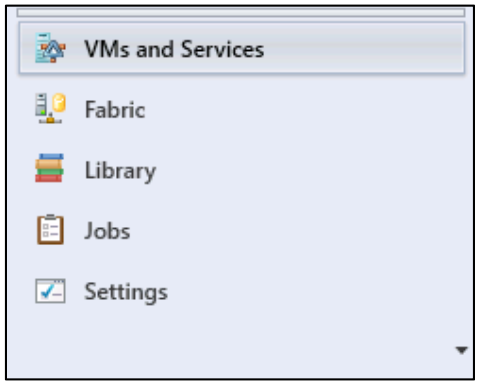
4. On **HyperV02**, load the **Command Prompt**, and type in the command **ping <Guest01 IP> -t** then press return. This will now set up a persistent ping to the **Guest01**. In the previous step the IP address was **192.168.1.109**, so in this example the command used would be **ping 192.168.1.109 -t**

5. Leave the **Command Prompt** box open so you can review the results of the ping

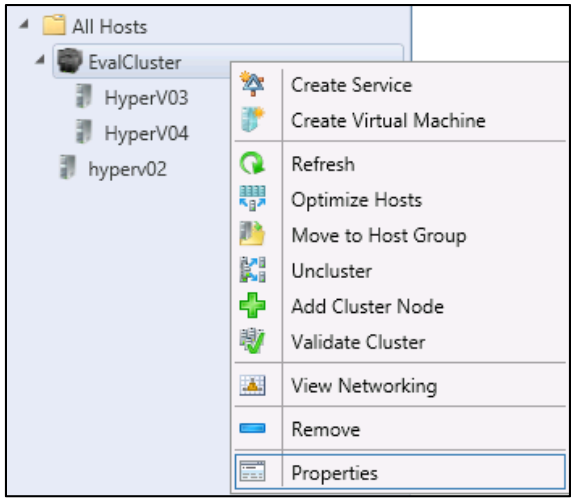


The next steps will make the SMB 3.0 share available to the HyperV cluster. This will be used for the shared storage of **Guest01**

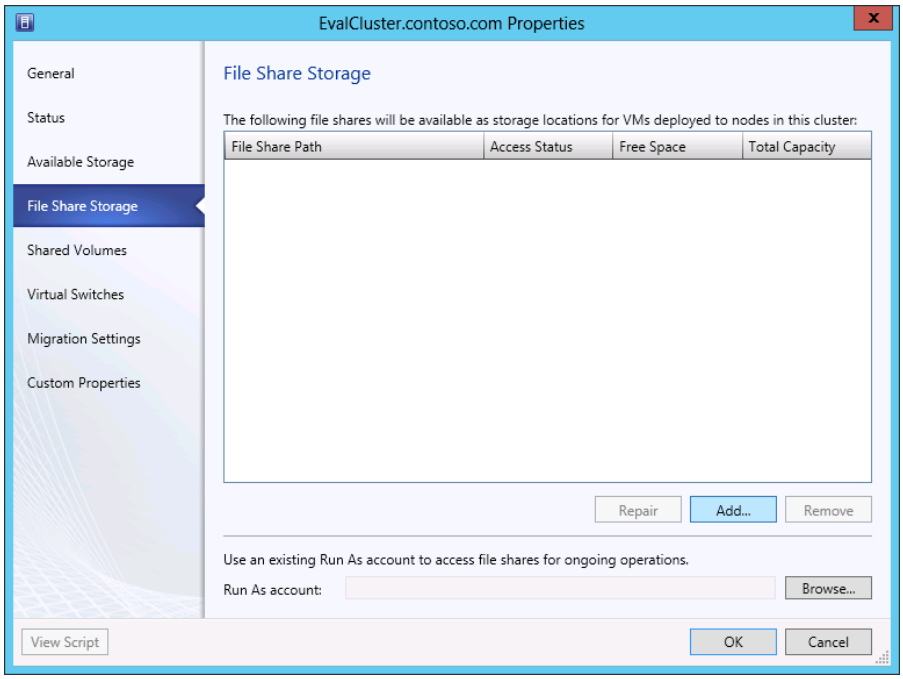
1. Load the Virtual Machine Management Console and log in as Jeff. Click on VMs and Services



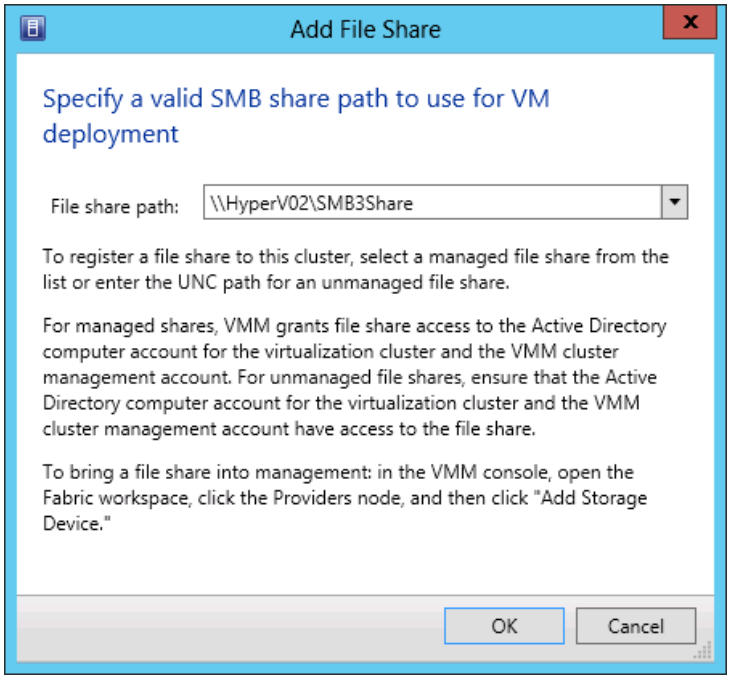
2. Expand **All Hosts**, right click on **EvalCluster**, then select **Properties**



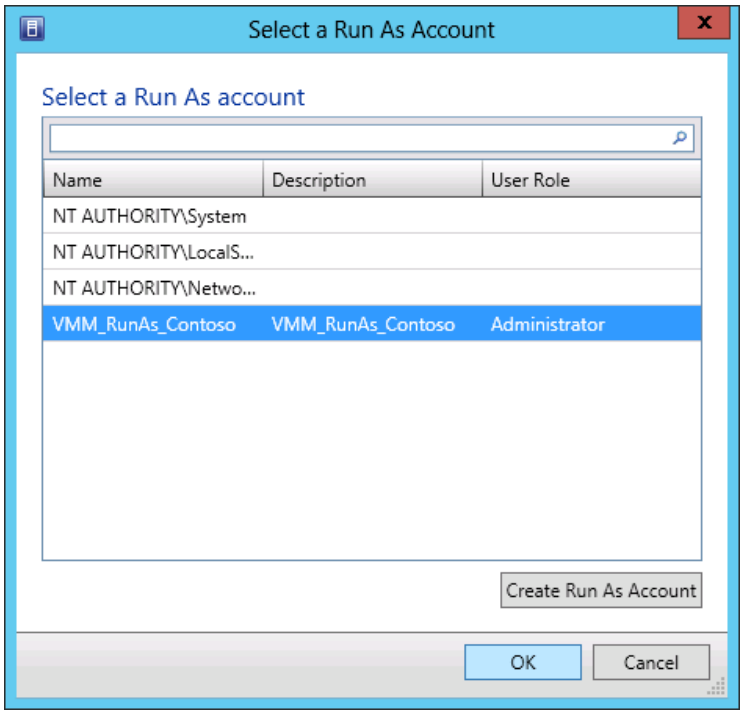
3. Click on File Share Storage, click on Add



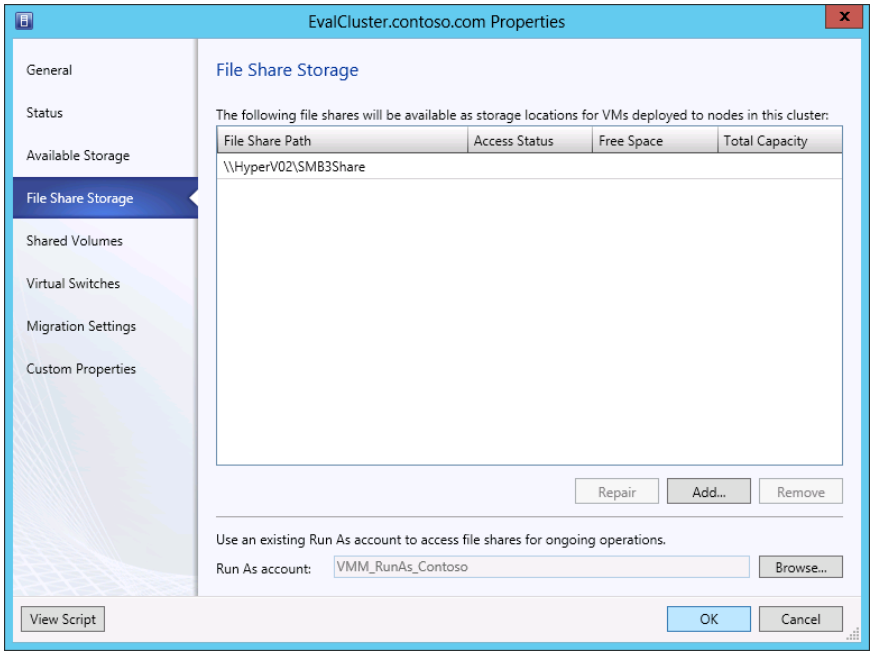
4. On the Add file share window in the File share path, type \\Hyperv02\SMB3Share click OK



5. Click on the **Browse** button next to the **Run As account** box, select **VMM_RunAs_Contoso** Click on **OK**

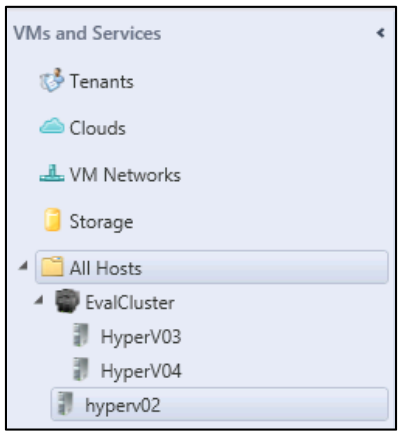


6. Confirm your settings then click on **OK**

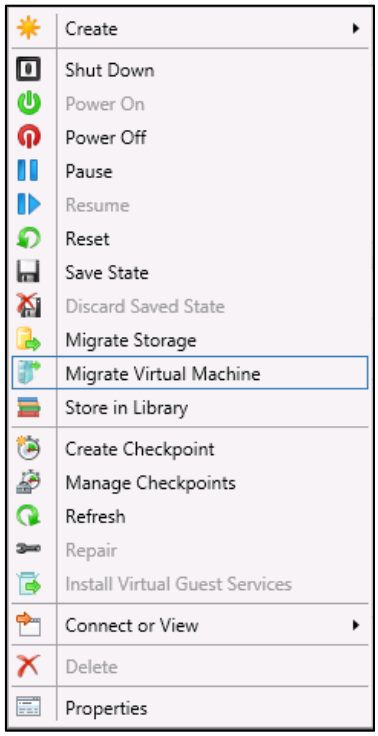


This final step will start the migration of **Guest01** from **HyperV02** to **EvalCluster**.

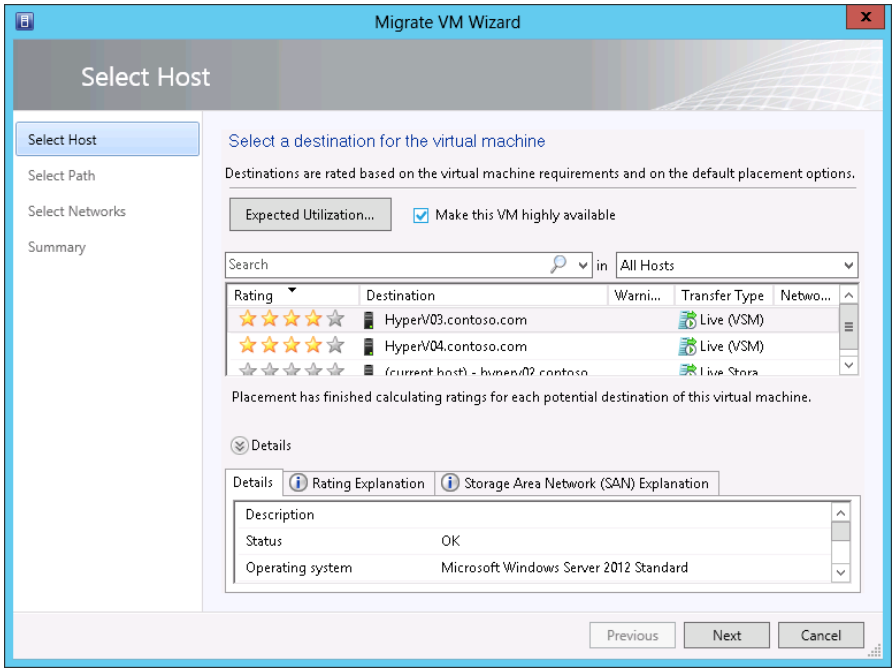
1. Load the Virtual Machine Management Console and log in as Jeff. Expand All Hosts, click on HyperV02



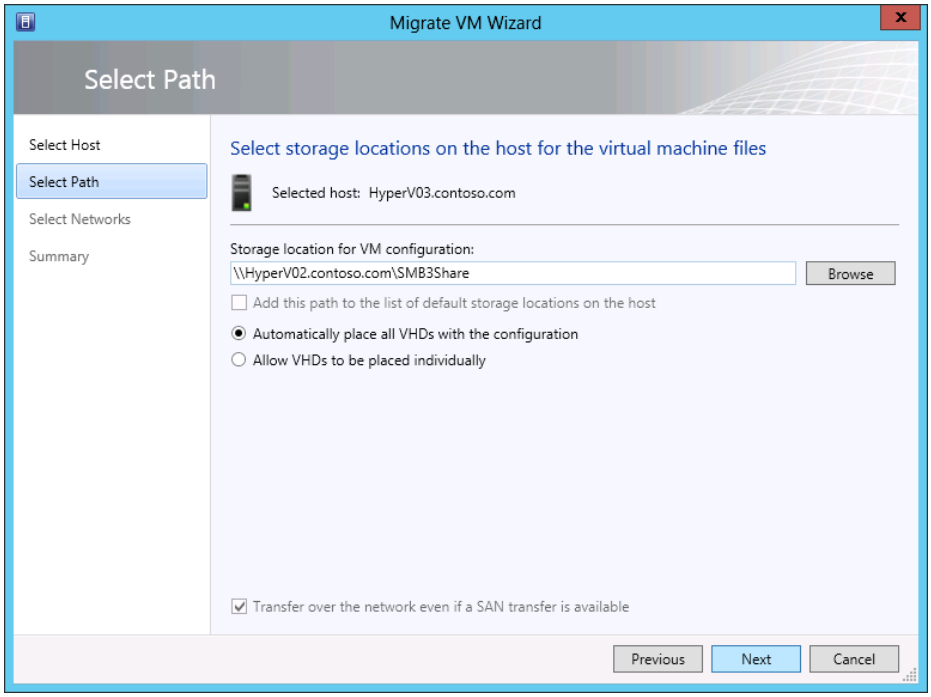
2. In the **VMs** pane right click on **Guest01** then select **Migrate Virtual Machine**



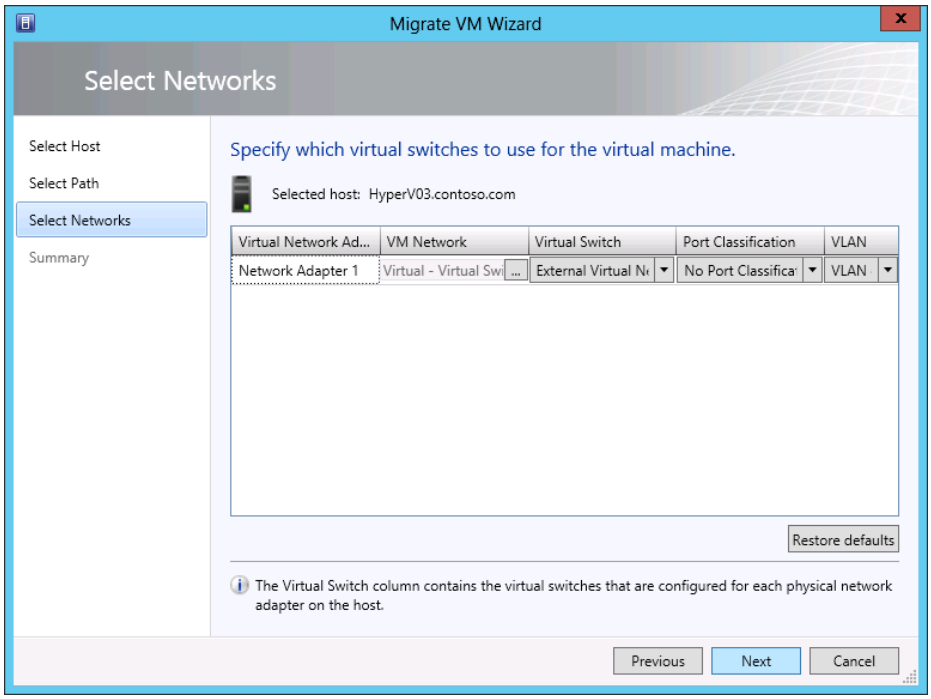
3. Wait for **Virtual Machine Manager** to assess your servers and provide the scores for each of your hosts. Tick the box **Make this VM Highly available**, and **Virtual Machine Manager** will now reassess the servers and score their suitability from a high availability perspective. Select **HyperV03** then click on **Next**



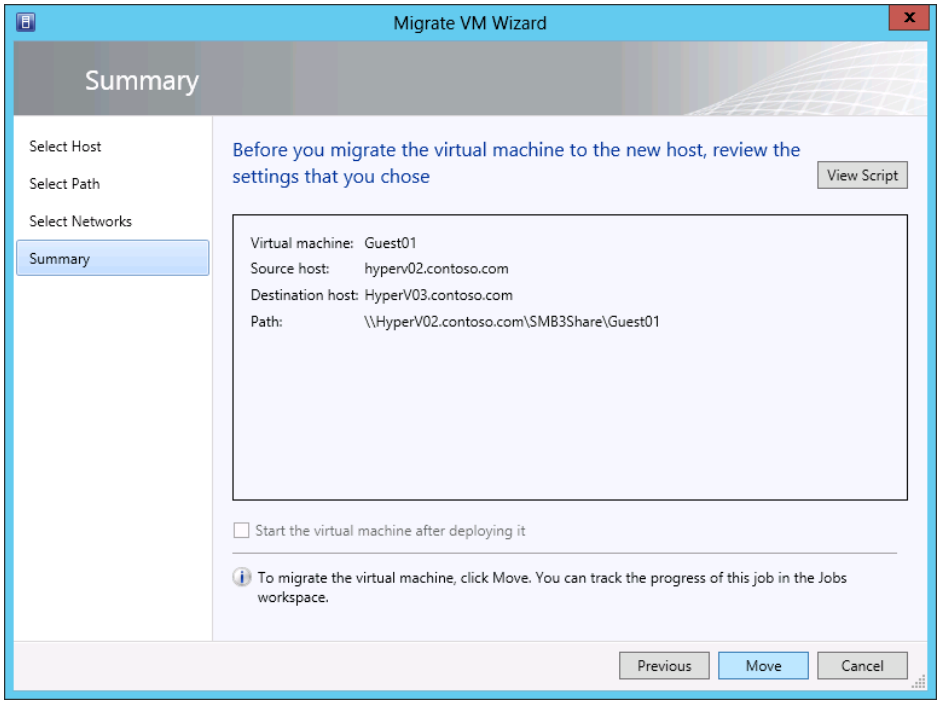
4. Confirm that the **SMB 3.0** share on **HyperV02** is being used, don't change any other settings, click **Next**



5. On the **Select Networks** window, leave the settings are default then click **Next**



6. Review your settings, then finally click **Move**



While migration is taking place, look at the ping results in the command prompt on **HyperV02**. You should find that there are very little, if any dropped network packets during the migration. Small fluctuations in ping times and a small number of dropped packets are expected during the final handover of the migration.

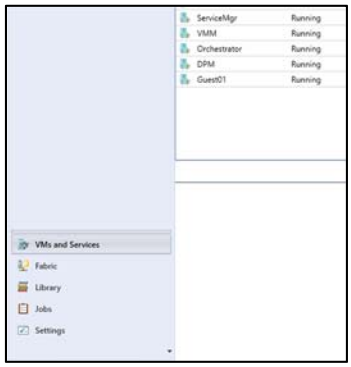
With the completion of this step Jeff has successfully migrated the non-highly available Guest01 to the newly created highly available failover cluster. Not only is Guest01 now highly available, but it was migrated with no downtime or loss of service. The virtual machine was also migrated with no shared storage between the two nodes – an excellent example of Shared Nothing Live Migration. Virtual machines can be migrated between different Hyper-V 3.0 hosts that share no commonly accessible storage with no downtime.

Scenario: Using remote SMB storage shares on a VHD

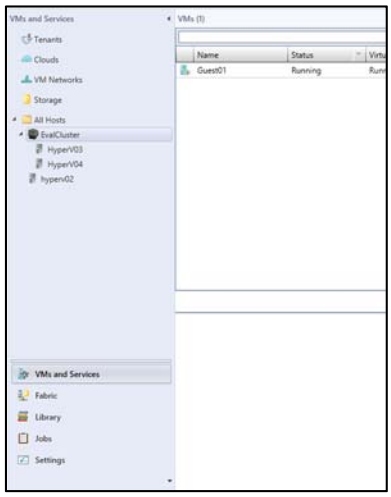
Now that Jeff has performed the migration, he investigates the virtual machine settings in his recently implemented Virtual Machine Manager environment. Jeff will confirm that the storage of **Guest01** is now located on the SMB 3.0 share located on **HyperV02**, and that the VM is running on the cluster **EvalCluster**.

1. Load the **Virtual Machine Management** console and enter the username **contoso\Jeff** and the password **pass@word1**. Click on **Connect**

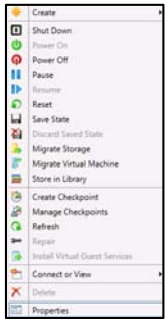
2. Click on VMs and Services



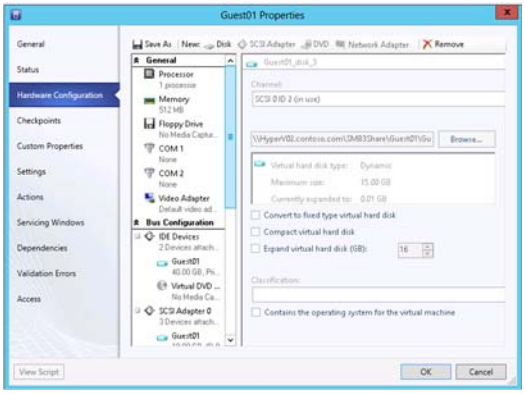
3. Expand All Hosts and click on EvalCluster



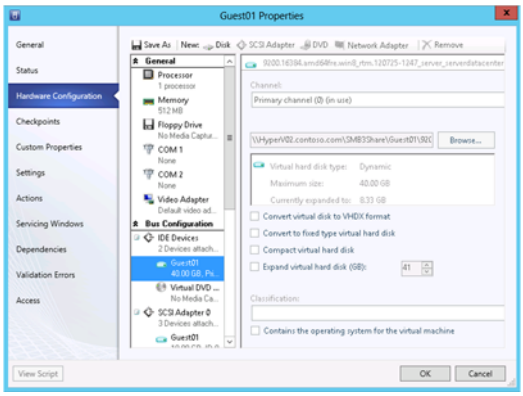
4. Right click on **Guest01**



5. Left click on Guest01 and select Properties then click on Hardware Configuration



6. Click on the Hard Drive 0 under IDE Controller



7. Confirm that the location of the drive is now **\\HyperV02\SMB3Share**
8. Click on **OK**

At the end of this scenario Jeff has confirmed that **Guest01** has been migrated from **HyperV02** to **EvalCluster** and is utilizing a remote SMB Share for storage of the virtual hard disk.

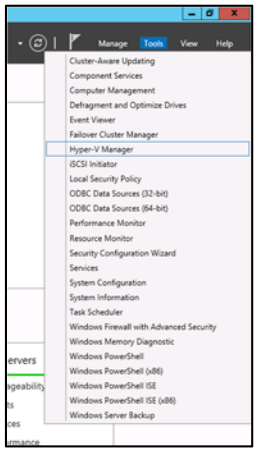
Scenario: Using replication for shared nothing failovers

Now that the Guest01 server hosting the critical data is highly available, Jeff decides to implement another feature of Hyper-V in Server 2012 to ensure business continuity in event of a total site failure. Setting up a replica allows for a shared nothing failover to another Hyper-V host using Windows Server 2012. This means a quick and simple way of ensuring continuity with the virtual infrastructure, especially if the hosts are in different geographic locations.

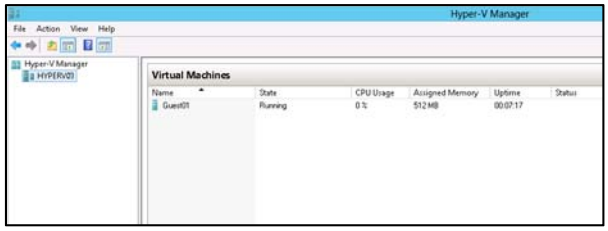
Jeff also notes that one of the features of Hyper-V Replica is the ability to change the IP address of the virtual machine when the failover occurs. This ensures the server can communicate on a different subnet with no intervention from the administrators during failover.

In this scenario we will replicate **Guest01** from **EvalCluster** back to **HyperV02**.

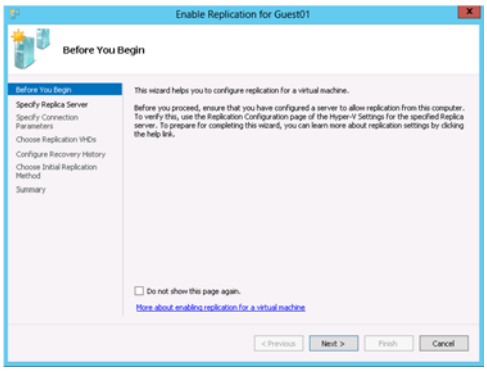
1. Load the **Hyper-V** console on **HyperV03**



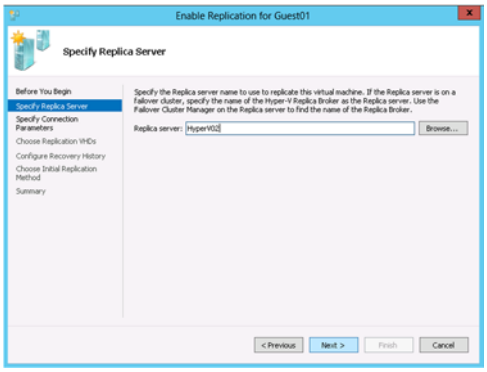
2. Right click Guest01 and select Enable Replication



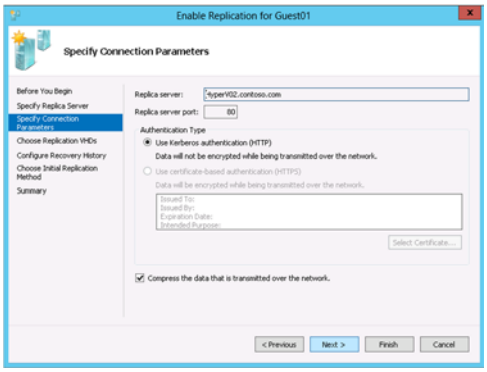
3. Click on **Next**



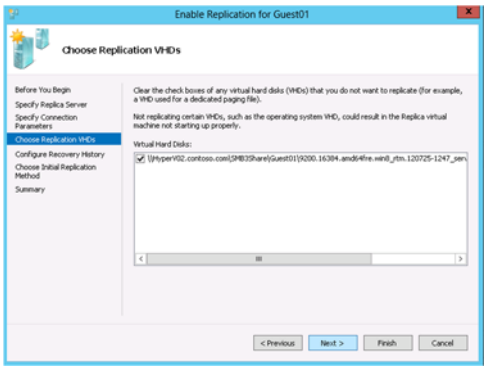
4. Type in the name **HyperV02**



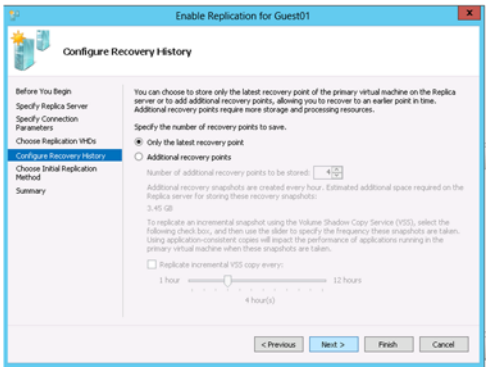
5. Click on **Next**



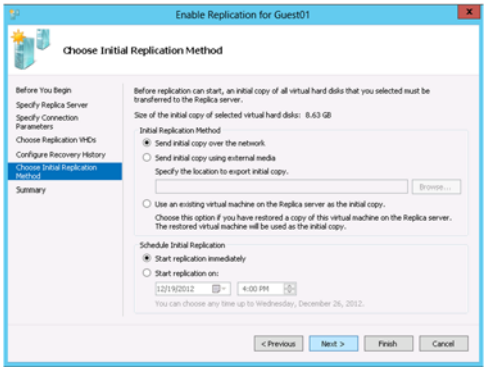
6. Click on **Next**



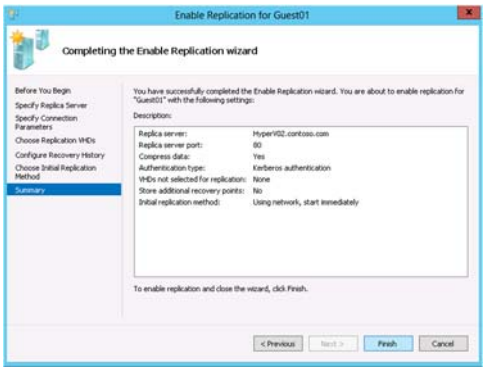
7. Click on **Next**



8. Leave these settings default, click on **Next**. (For the purposes of the evaluation guide these settings can be kept at default, although it would be good to familiarize yourself with the delivery options for your initial replication.)



9. Click on **Finish**

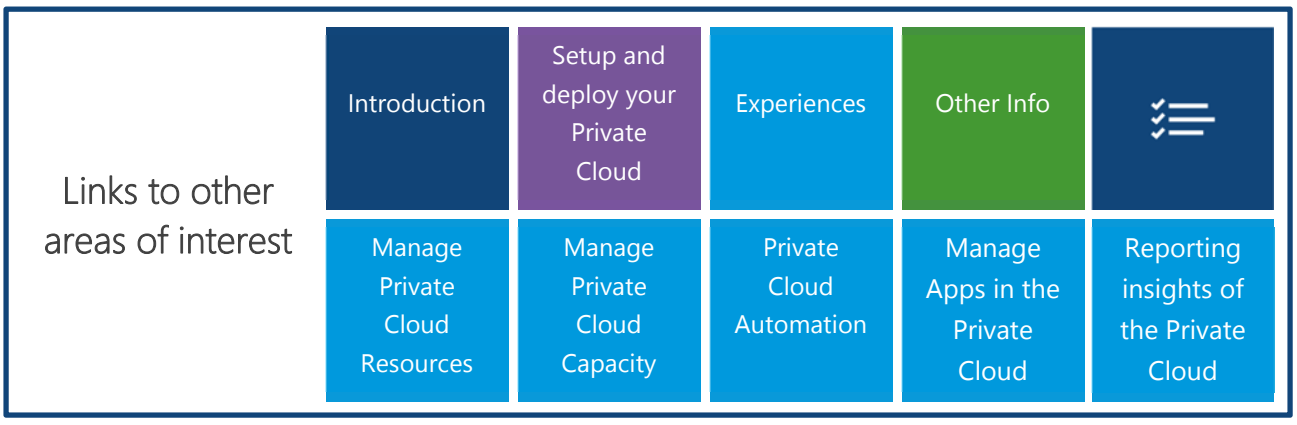


At the end of this scenario we have successfully replicated **Guest01** from **EvalCluster** to **HyperV02**. Hyper-V replication is not a point-in-time snapshot and the data will be constantly replicated from **EvalCluster** to **HyperV02**.

Now that the replica has been configured, Jeff has successfully implemented a robust business continuity solution without having to purchase additional hardware of software. This will ensure the cost of business continuity for Contoso Electronics is kept low.

Synopsis

Jeff has successfully implemented several features that will allow the Contoso Electronics company to continue to work in the event of hardware failure, or if maintenance is required. Previously, work would stop for several hours when Jeff needed to patch his servers or upgrade hardware. Now he can implement improvements and configuration changes to his infrastructure without interrupting the business. This is critical as it allows Jeff to ensure the infrastructure continues to supply the business with always on and efficient supply of computing power to continue to grow and operate.



Experience 2: Manage Private Cloud capacity

Once **Jeff** has a **Private Cloud** in place it's time to take advantage of the features of **Windows Server 2012** and **System Center 2012 SP1** to increase your capacity or manage your workloads dynamically without downtime.

In this Experience we'll step through three scenarios:

- Dynamically expand storage for workloads
- Modifying network resources of VM's
- Using dynamic memory to improve workloads

Scenario: Dynamically expand storage for workloads

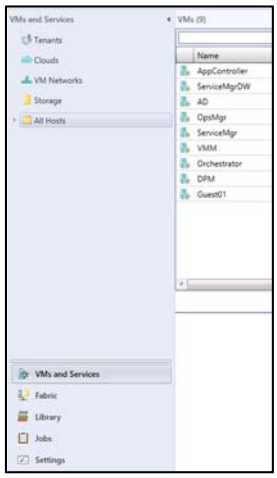
In this scenario we can see how the Jeff can dynamically expand the storage capacity of a server utilizing the Storage Spaces features in Windows Server 2012 without interrupting service.

Debbie has informed Jeff of upcoming projects that will require large amounts of disk capacity for storing SQL databases for a new Sales application that is being developed. Jeff assesses the current capacity and identifies that it will not meet the needs of the business. Jeff also realizes that he does not have enough budget this month to purchase new disk for his storage server.

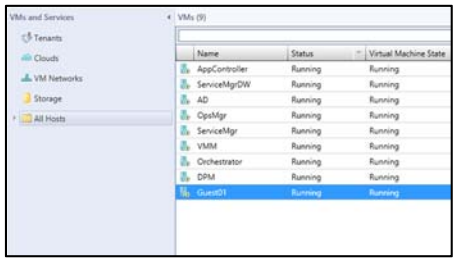
Jeff identifies that he has a pool of different size disks left over from previous projects and hardware upgrades that is still functional, but because they are all different sizes they cannot be included in a typical RAID volume. Jeff decides to take advantage of the Just a Bunch Of Disks (JBOD) features of Server 2012 and present these disks in a pool to the project team for storage. He will also allow for growth on the volume in case the project overruns.

1. Load the Virtual Machine Manager console, select Use current Microsoft Windows session identity and click Connect

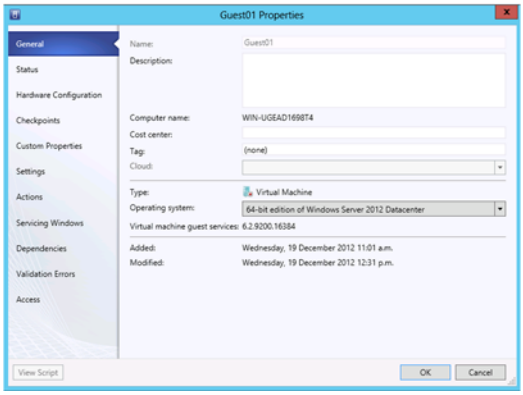
2. Click on VMs and Services and click on All Hosts



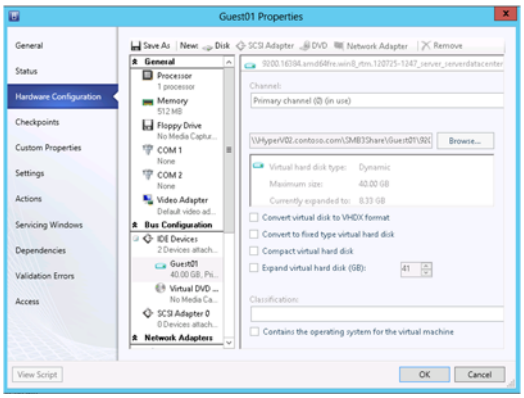
3. Click on **Guest01**



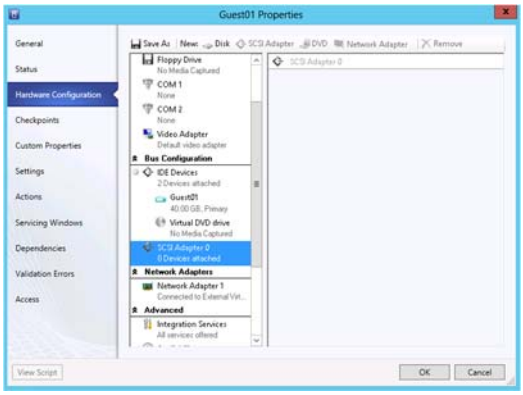
4. Right click **Guest01** and click on **Properties**



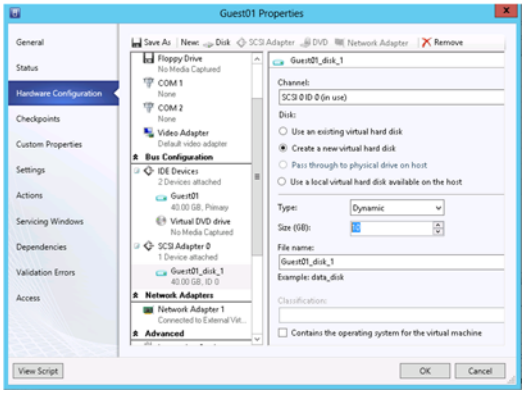
5. Click on Hardware Configuration



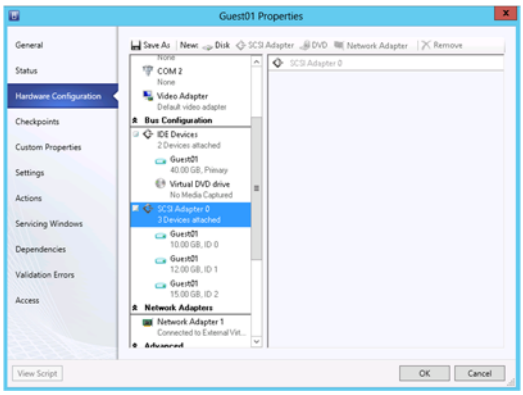
6. We will be adding 3 SCSI disks to **Guest01** as it will allow us to **hot-add storage** without rebooting the guest OS.
7. Scroll down until you can see **SCSI Adapter** under the **Bus Configuration** and select **SCSI Adapter**



8. At the top, click on **New: Disk**, change the size of the disk to **10 GB** and click on **OK**

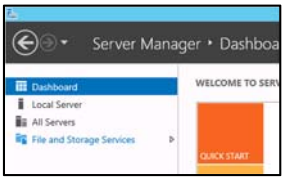


9. Wait for the VM to finish updating, then repeat the above steps to add an additional **2 disks**. These disks will need to be **12GB** and **15GB** in size.

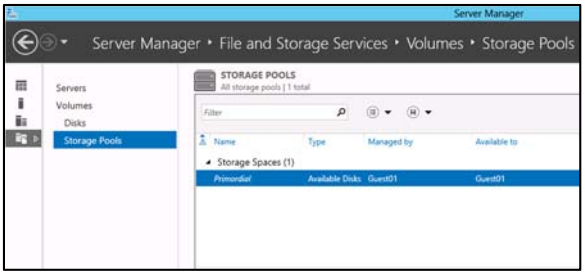


10. At the end of this you will have **three** additional disks assigned to **Guest01**, these will be **10GB**, **12GB**, and **15GB** in size. When we created the disks, VMM will automatically place the created disks in the same location as the current **Guest01** VHDX files. If you look at the **SMB3Share** on **HyperV02** you will notice the newly created VHDX files.

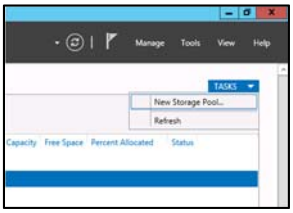
11. Connect to **Guest01** and within Server Manager, click on **File and Storage Services**.



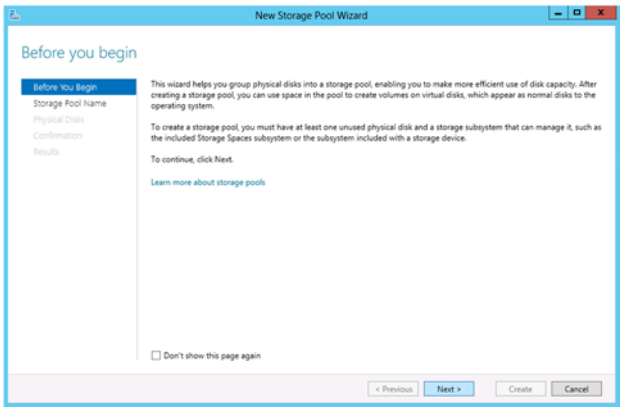
12. Click on Storage Pools



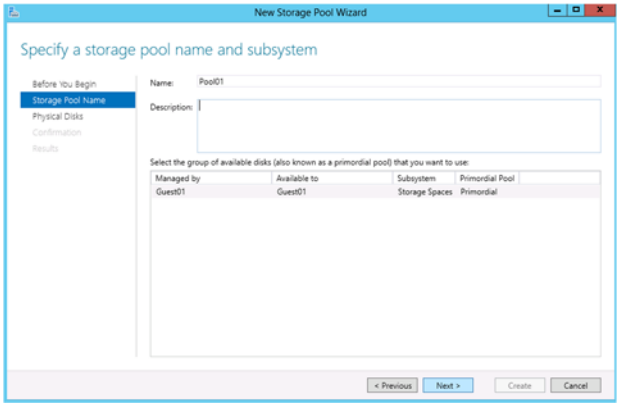
13. In the **Physical Disks** section click on **Tasks** and click **New Storage Pool**



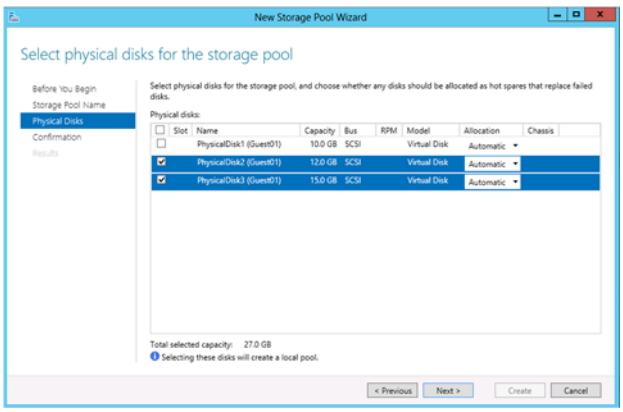
14. In the Before you begin screen click Next



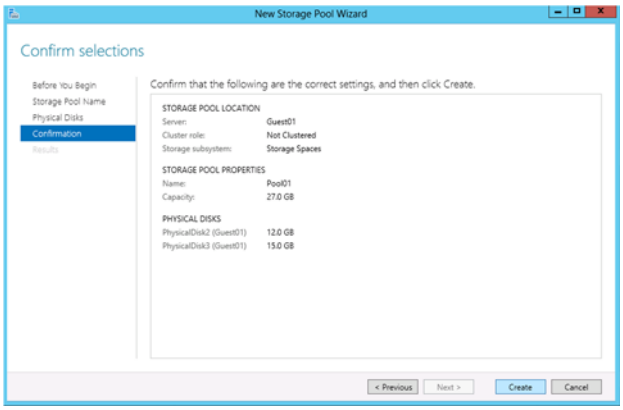
15. In the **Storage Pool Name** screen, type **Pool01** into the **Name** field, click on **Next**



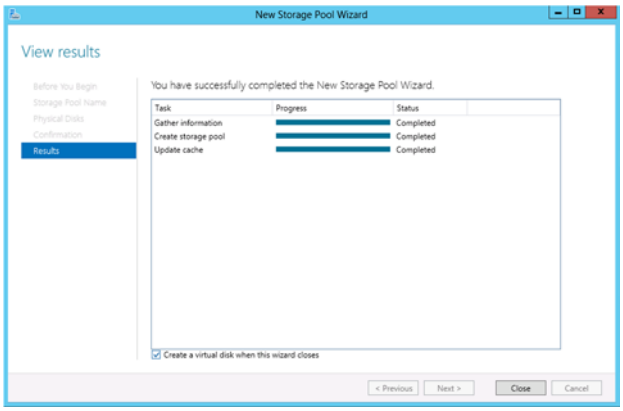
16. In the **Physical Disks** screen tick the **12 GB** and **15GB** disk. Do **NOT** select the **10GB** disk. Click **Next**



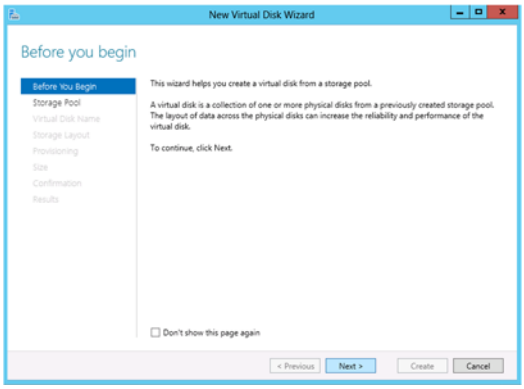
17. Click **Create**



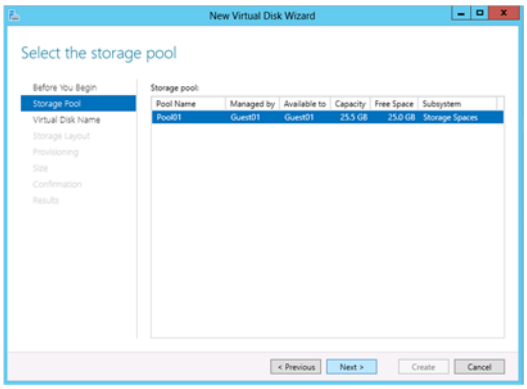
18. Once completed, tick the box that says **Create a virtual disk when this wizard closes** and then click on **Close**



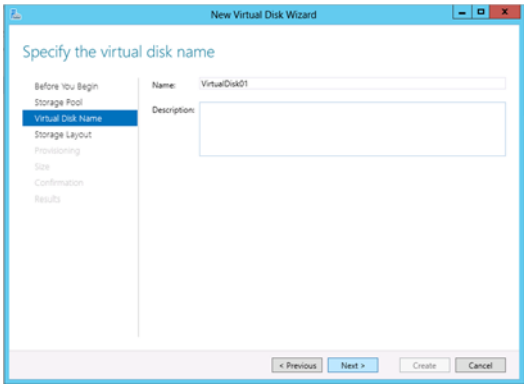
19. The **New Virtual Disk Wizard** will now open. Click on **Next**



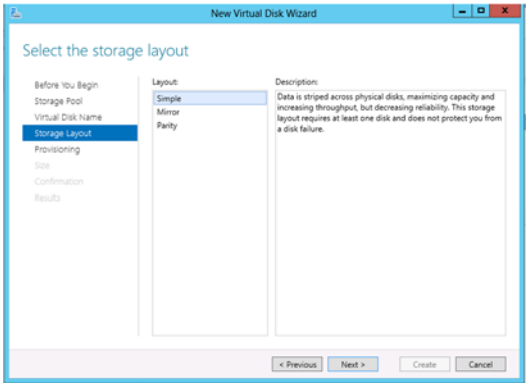
20. Select **Pool01** and click **Next**



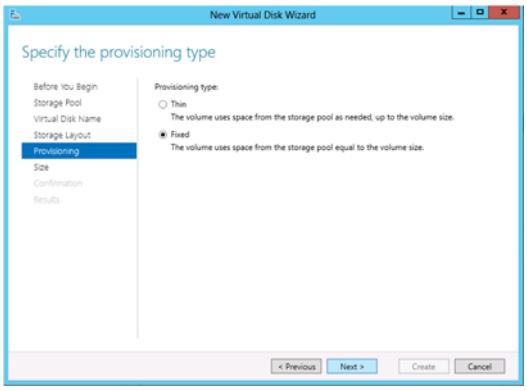
21. Name the virtual disk **VirtualDisk01** and click **Next**



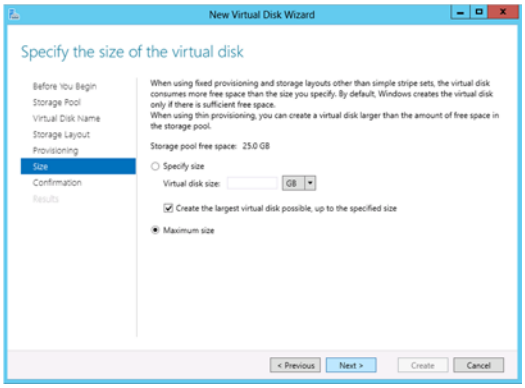
22. Select **Simple** and click **Next**



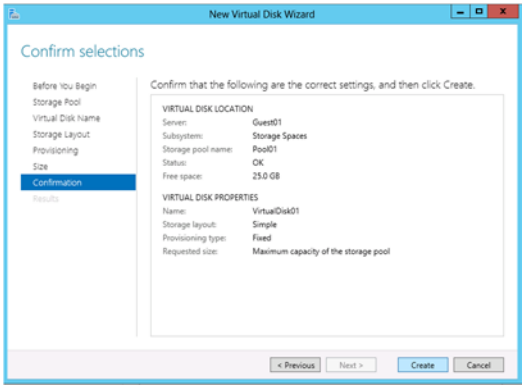
23. Ensure that **Fixed** is selected and click **Next**



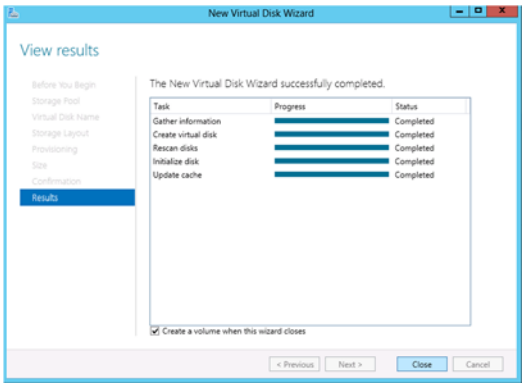
24. Tick the box **Maximum size** and click **Next**



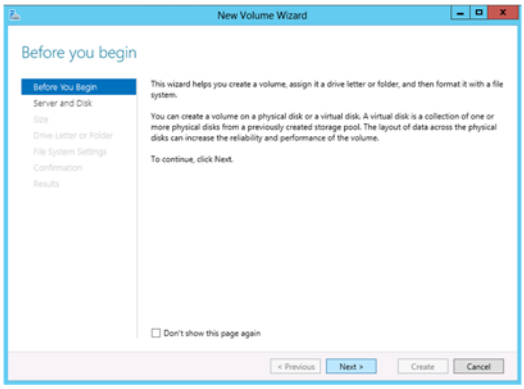
25. Click **Create**



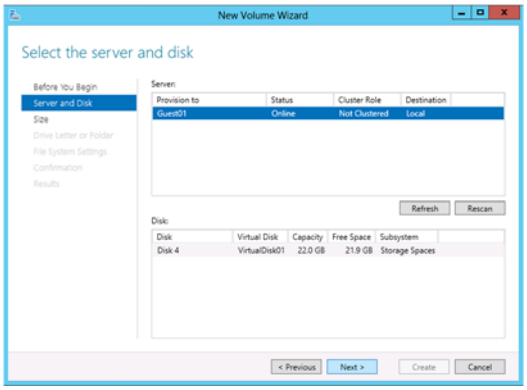
26. Ensure that the Create a Volume when this wizard closes box is ticked, click on Close



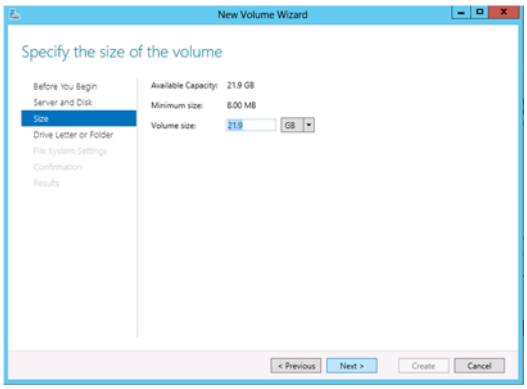
27. The **New Volume Wizard** will now open. Click **Next**



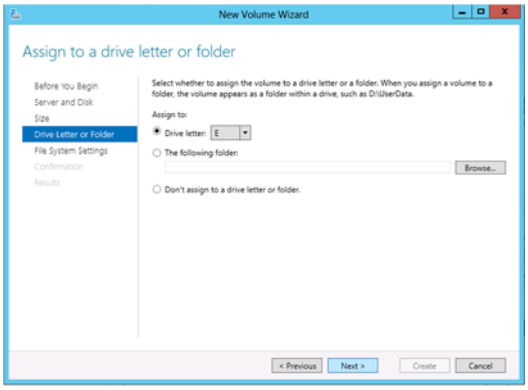
28. Review the settings and click **Next**



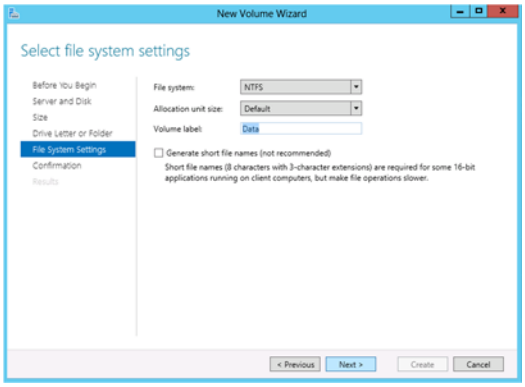
29. Confirm the size of the disk as **21.9GB** and click **Next**



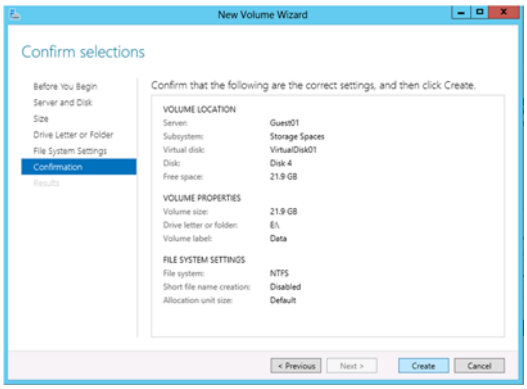
30. Select the drive letter **E:** and click **Next**



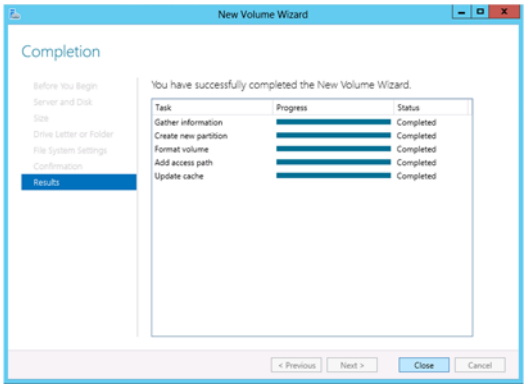
31. Change the **Volume Label** to **Data** and click **Next**



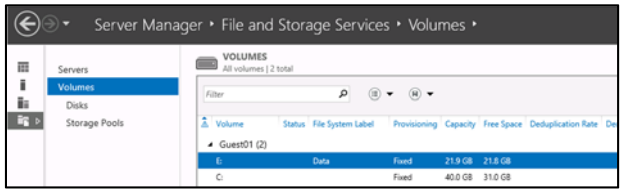
32. Click on **Create**



33. This will now create you a new **Volume**. Click **Close**



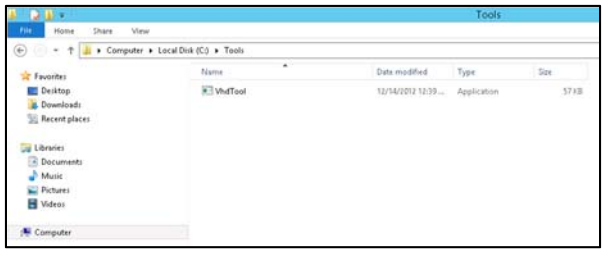
34. Within Server Manager, click on File and Storage Services and click on Volumes



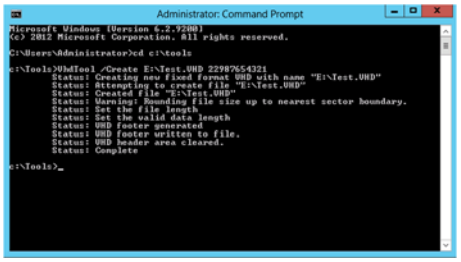
Confirm that your new volume is listed there and the free space is correctly reported. If you did not give your newly created drive the drive letter of E: then take note of the Volume letter here. This information is required for the next step.

35. Copy the VHDtool.exe Application from **\\HyperV02\SMB3Share** to the folder **C:\Tools** on **Guest01**

36. Open a **command prompt** by moving to the bottom left hand corner of the screen, opening the **Start screen** and typing **CMD**. Click on the **Command Prompt** icon. Browse to the folder that VHDtool was downloaded to by typing **cd c:\tools**.

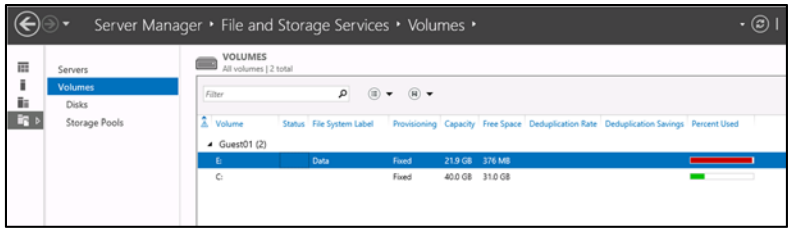


37. Type in the following: **VHDTOOL /Create E:\Test.VHD 22987654321** and press enter. If you created your new drive with a different drive letter then change the above command accordingly.



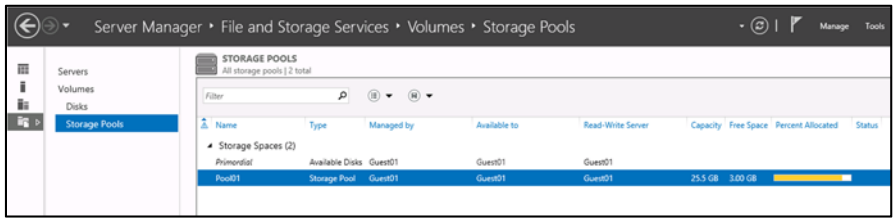
38. You will now have **21.4GB** VHD file on your volume. Close the **command prompt.**

39. Load Server Manager, click on File and Storage Services then Volumes

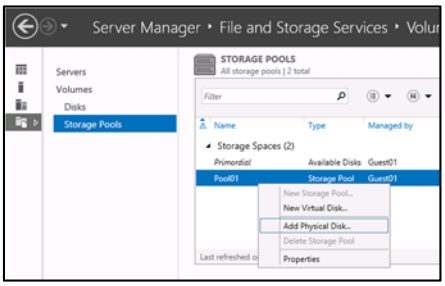


40. Take note of the free space on the **E:** drive and the fact that the **Percent Used** is now in a red **emergency** state

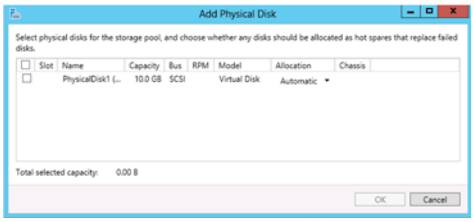
41. Click on Storage Pools



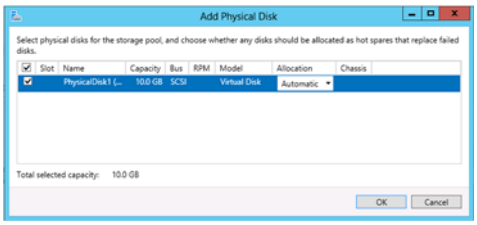
42. Right click on **Pool01** under **Storage Spaces**



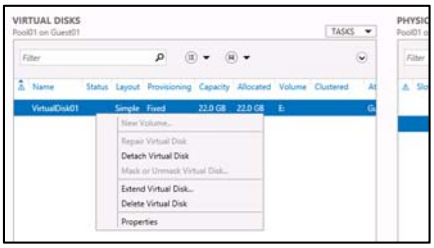
43. Select Add Physical Disk



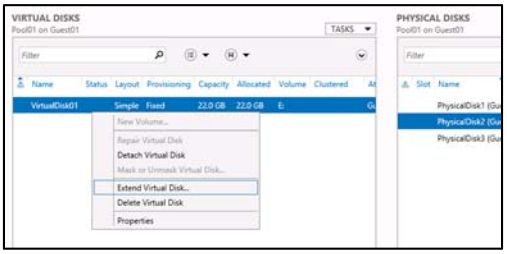
44. **Tick the box** next to the **10GB** disk and click on **OK**



45. Right click on **VirtualDisk01** under the **Virtual Disks** panel

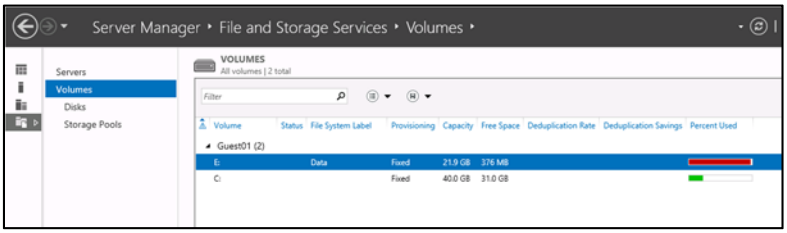


46. Left click on Extend Virtual Disk

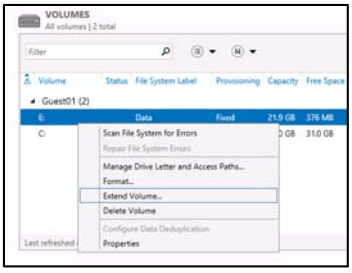


47. Input a new value of **28GB** and click on **OK**

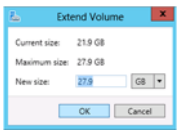
48. Click on **Volumes** in the left hand pane



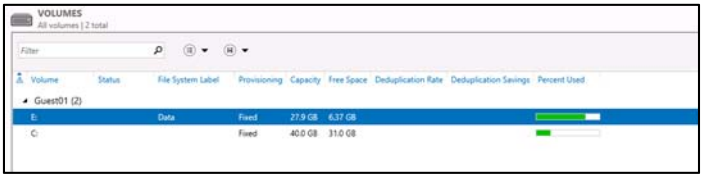
49. Right click the **E:** drive and select **Extend Volume**



50. Input the number listed under **Maximum size** (i.e. **27.9GB**) and click on **OK**



51. Once completed, confirm that the **E:** drive has now additional free space.



At the end of this scenario Jeff has created a single virtual volume that spans multiple physical drives of varying sizes. This allowed him to utilize the disparate disks and combine them into a single pool of storage for the developers to utilize.

He then added additional storage to the pool and extended out logical volume when they over-utilized their assigned capacity.

Important Note

Do not delete the created VHD file as this will be used in the following scenario.

This highlights the ability of administrators to take advantage of disparate disks they may have in different locations to create large pools of storage. JBOD can become an inexpensive way to ensure that environments can extend their disk resource with little new investment in infrastructure.

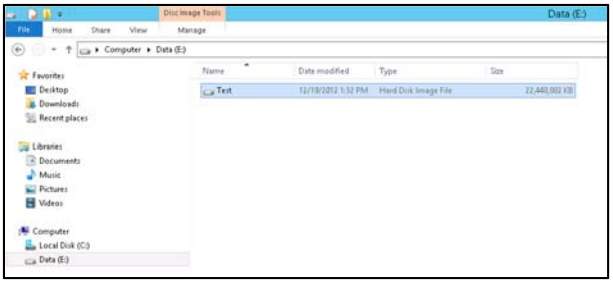
Scenario: Modifying network resources of VM's

Jeff has noticed that a new server that has been implemented into Production that hosts the new sales application is performing at capacity. In fact it appears to be utilizing much of the network resources of the underlying physical infrastructure to the detriment of other virtual servers hosted on the same server.

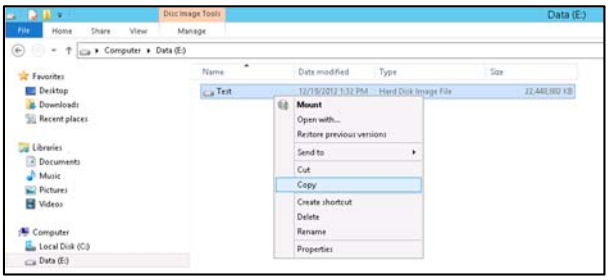
Jeff needs to alter the configuration of the virtual machine without interrupting the service to reduce the network throughput to allow other servers equal access to network resources.

In this example, the network load is generated using a large file copy.

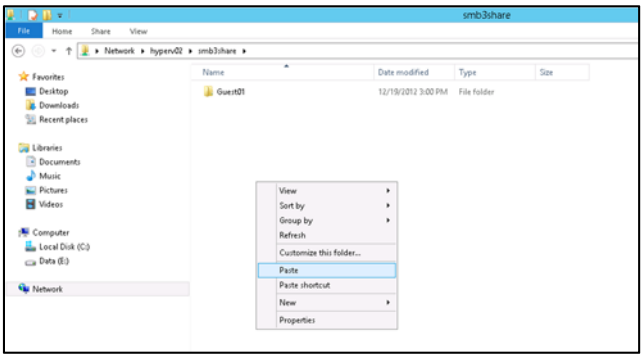
1. Connect to **Guest01**
2. Using **File Explorer** browser to the virtual volume file created in the previous scenario in the **E:** drive labelled **Test**



3. Right click on the file and select **Copy**

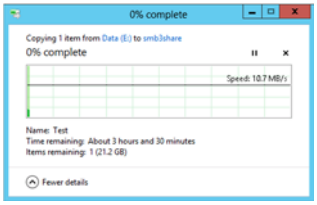


4. Browse to **\\HyperV02\SMB3Share,** right click and select **paste**

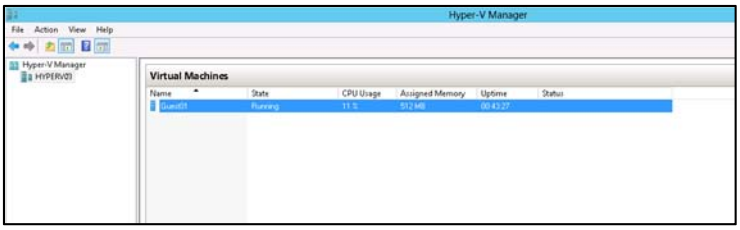


The Windows Copy dialogue box will now open and show you the speed at which you are transferring your file. Click on **More Details** as this will show you a graph that displays the current speed of the transfer.

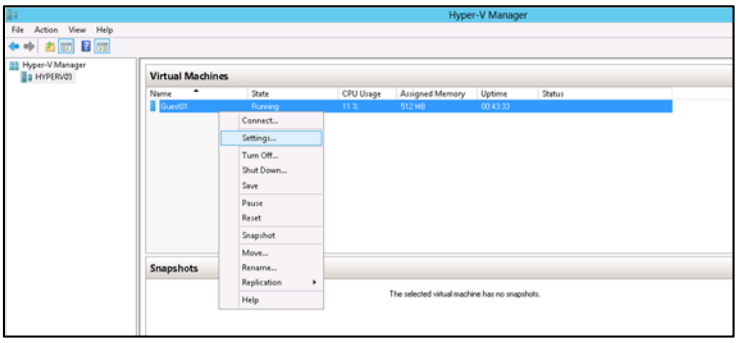
On a 100MB network this should be around 11.4MB/s on a 1000MB network this should be around 110MB/s. Confirm the current speed that the file is copying at and take a note of it.



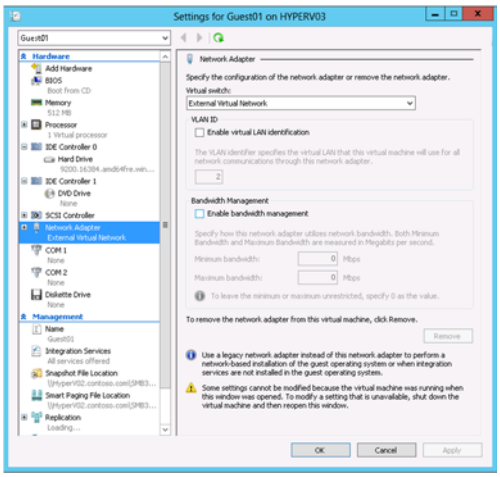
5. With this window still open, open the **HyperV Manager** console on **HyperV03**



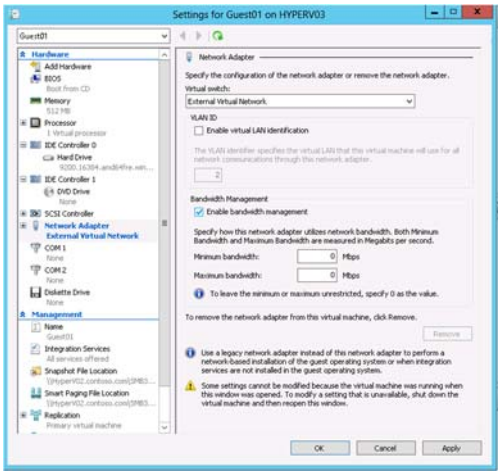
6. Select then **Guest01** virtual machine and right click it then select **Settings**



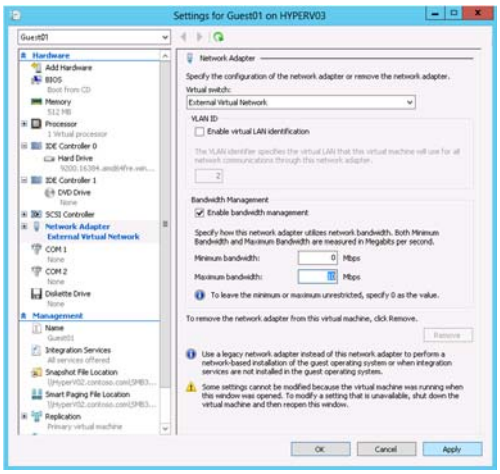
7. Click on Network Adapter



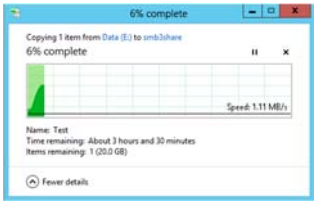
8. Under Bandwidth Management, tick the box Enable bandwidth management



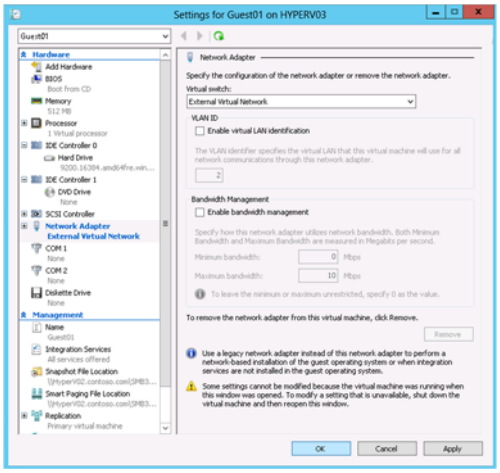
9. In the **Maximum Bandwidth** box type **10**. This will reduce the network to a **1MB/s** network. Click on **Apply**



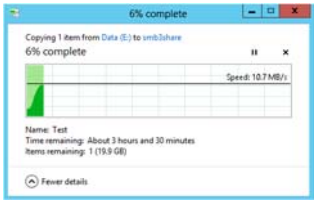
10. Go back to **Guest01** and confirm that the speed of your transfer has been reduced. It should be approximately **1MB/s**



11. Once you have confirmed this, go back to the **Hyper-V Manager** console and remove the tick from the **Enable Bandwidth Management** tick box and click **OK**



12. Once **Bandwidth Management** has been **disabled**, if you monitor the file transfer of **Guest01** you will notice that it will have returned to its original speed.



At the end of this scenario Jeff has successfully throttled the network bandwidth of **Guest01**. This will ensure that one virtual machine in the infrastructure will can be prevented from impacting other virtual machines hosted on the same physical host.

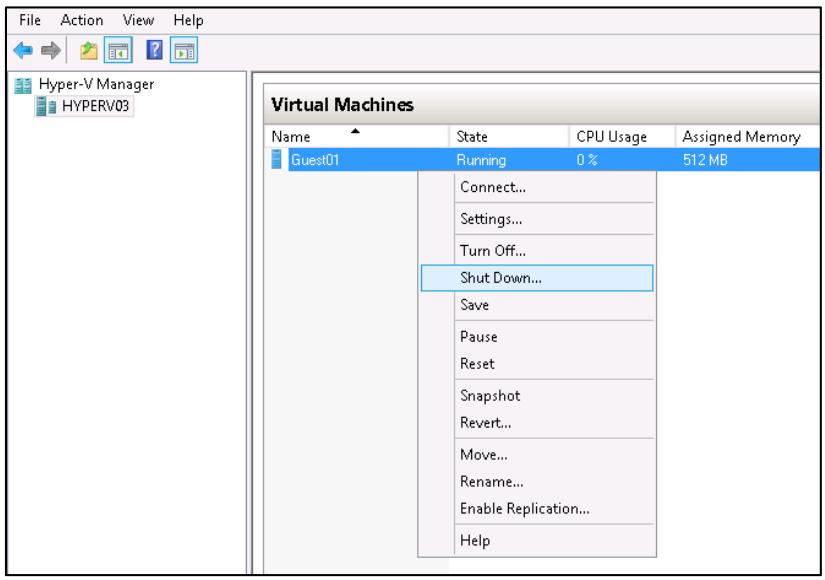
Important Note

The minimum setting is another important feature as it can ensure that an important server can continue to receive a minimum amount of bandwidth in times when all servers hosted on the same infrastructure are under load. This can be utilized to ensure that business critical applications continue to operate at critical times of business flux.

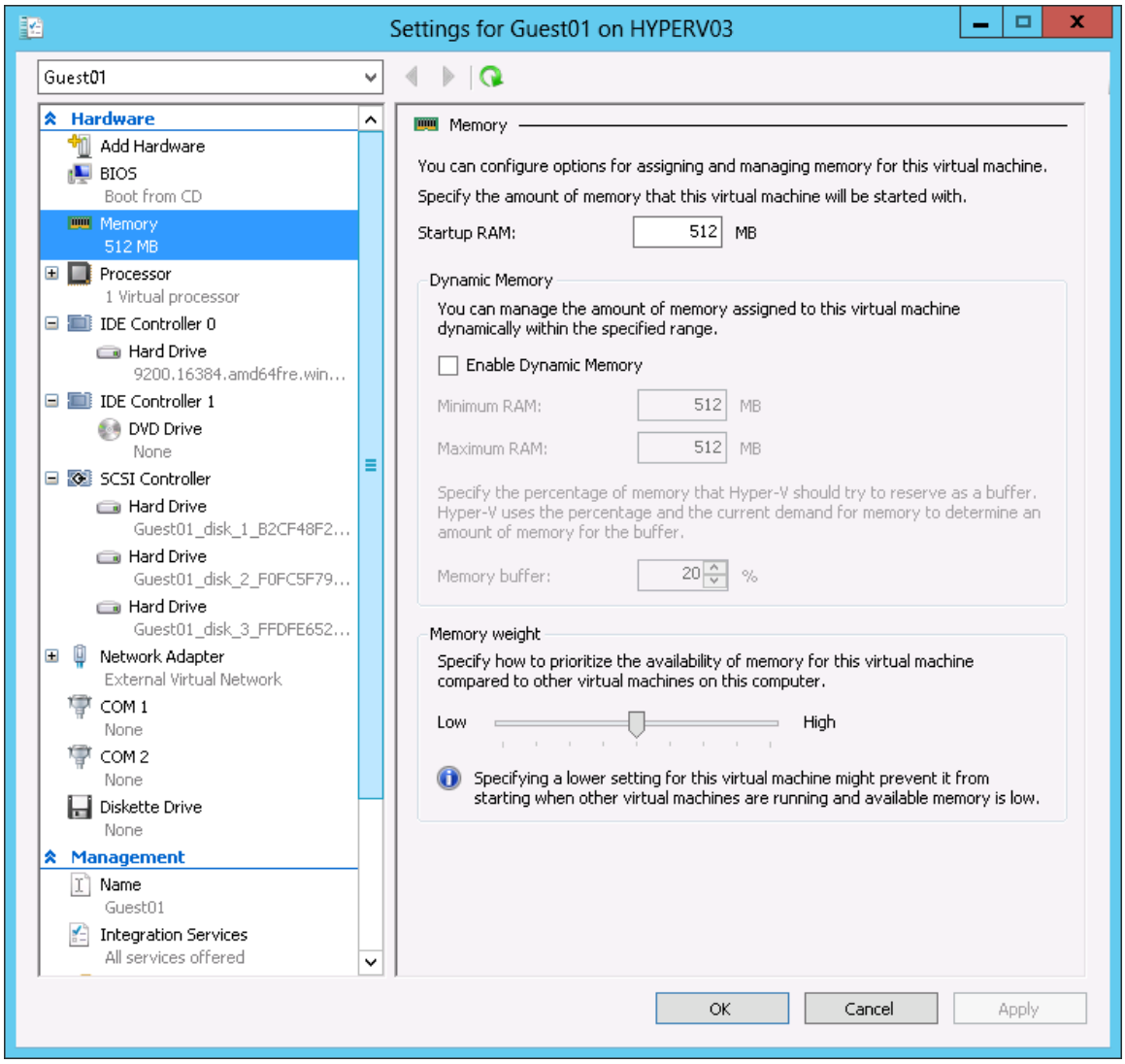
Scenario: Using dynamic memory to improve workloads

Jeff has noticed that at times of load the Sales application in production becomes non-responsive. During one of these periods he investigates the SQL Server and finds that it has consumed all of the available memory. Jeff decides to utilize the Dynamic Memory features of Hyper-V and SQL Server 2012 to allow the server to consume more memory resources as it requires.

1. Log onto the Hyper-V server **HyperV03**, open the **Hyper-V Manager** and Select **Guest01**
2. Right click and select **Shutdown**

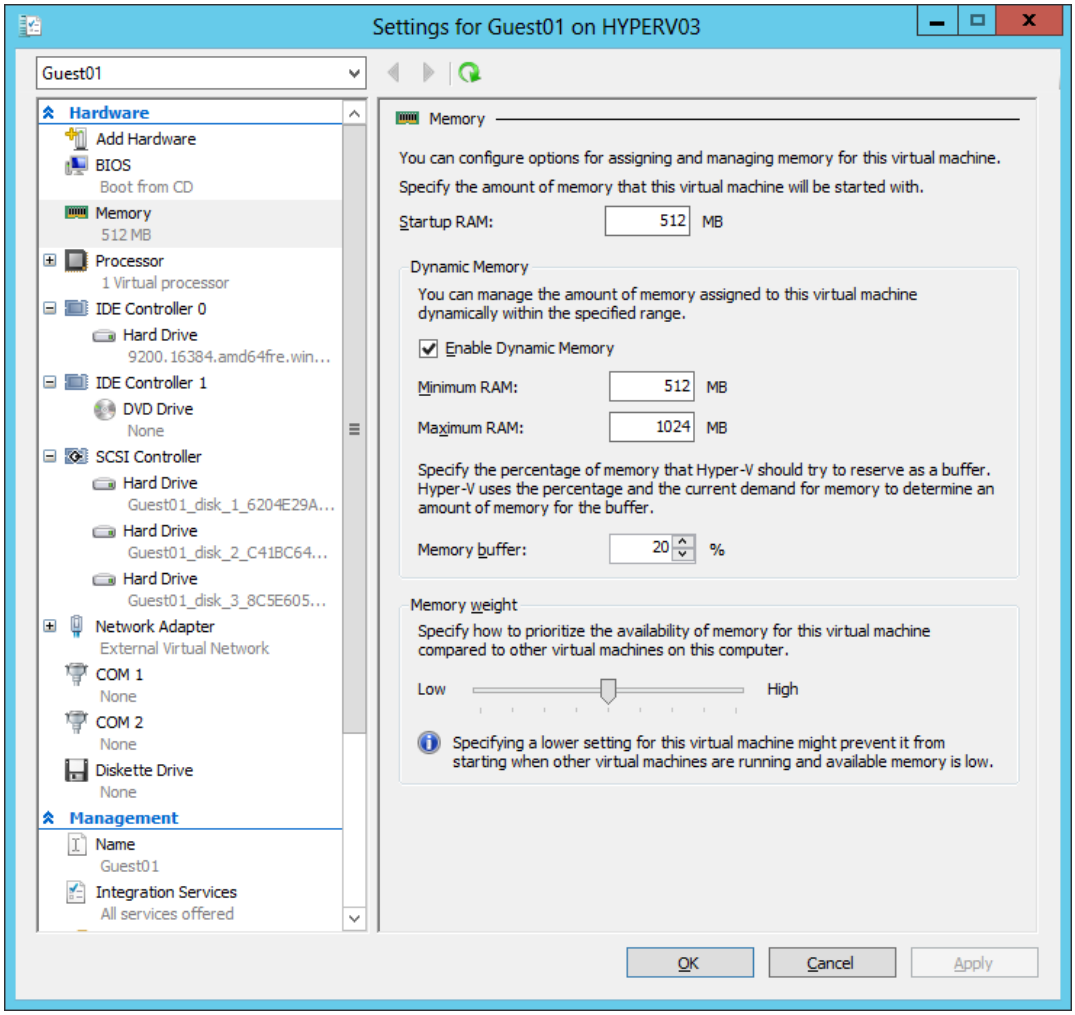


3. Once the server has shutdown, right click and select **Settings** and select **Memory** from the hardware section

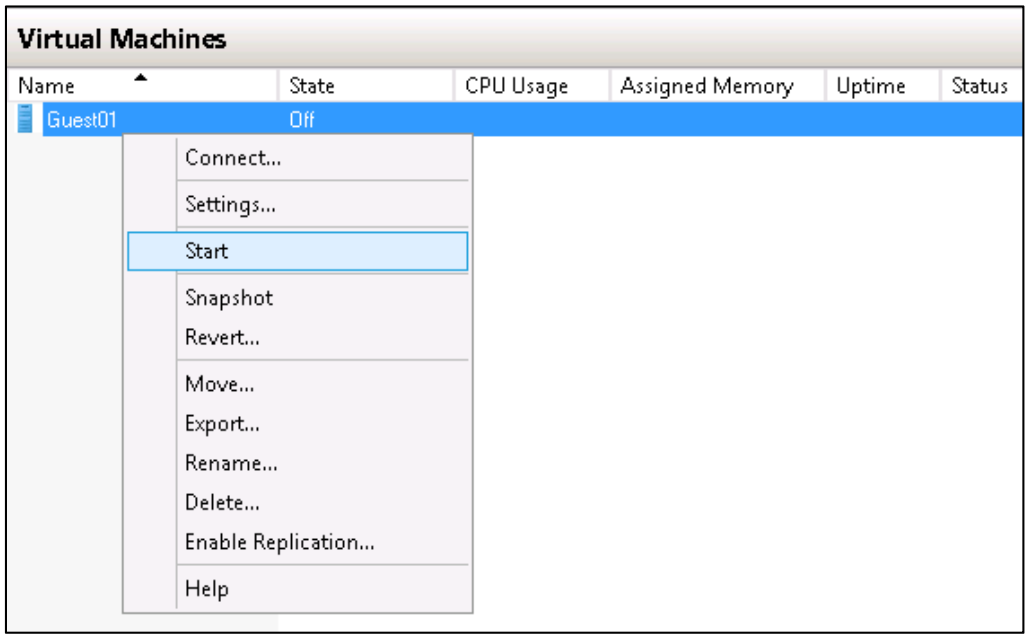


4. Set the following memory configuration and press **OK**

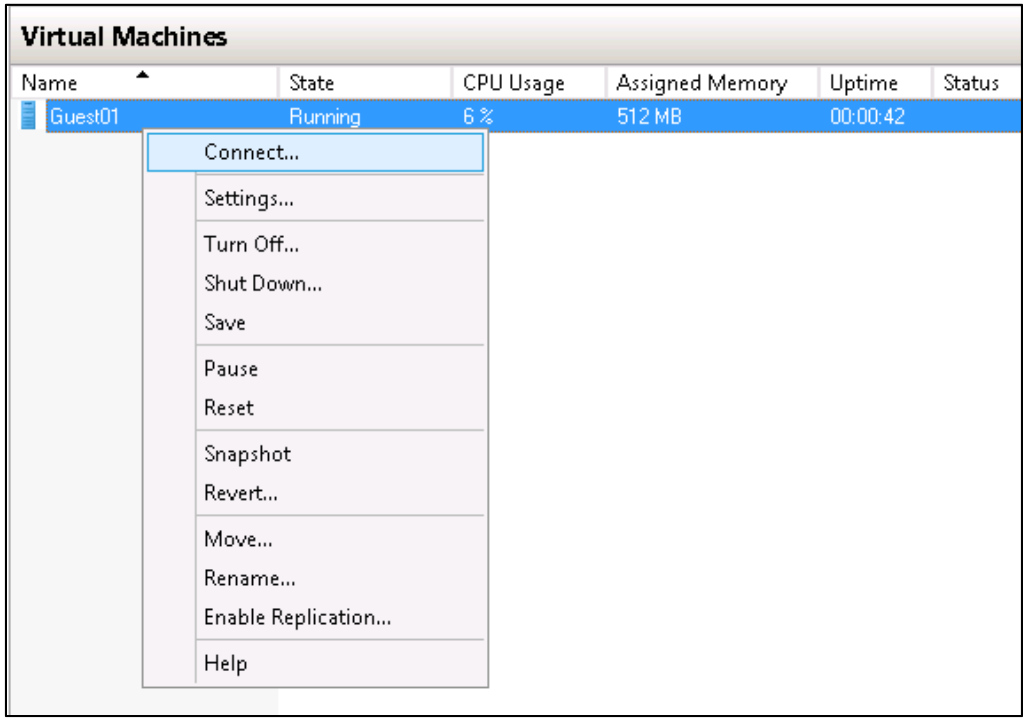
Dynamic Memory	Enabled
Minimum RAM	512MB
Maximum RAM	1024MB
Memory Buffer	20%


Settings for Guest01 on HYPERV03
Guest01
Hardware
Add Hardware
BIOS
Boot from CD
Memory
512 MB
Processor
1 Virtual processor
IDE Controller 0
Hard Drive
9200.16384.amd64fre.win...
IDE Controller 1
DVD Drive
None
SCSI Controller
Hard Drive
Guest01_disk_1_6204E29A...
Hard Drive
Guest01_disk_2_C41BC64...
Hard Drive
Guest01_disk_3_8C5E605...
Network Adapter
External Virtual Network
COM 1
None
COM 2
None
Diskette Drive
None
Management
Name
Guest01
Integration Services
All services offered
Memory
You can configure options for assigning and managing memory for this virtual machine.
Specify the amount of memory that this virtual machine will be started with.
Startup RAM: 512 MB
Dynamic Memory
You can manage the amount of memory assigned to this virtual machine dynamically within the specified range.
Enable Dynamic Memory
Minimum RAM: 512 MB
Maximum RAM: 1024 MB
Specify the percentage of memory that Hyper-V should try to reserve as a buffer. Hyper-V uses the percentage and the current demand for memory to determine an amount of memory for the buffer.
Memory buffer: 20 %
Memory weight
Specify how to prioritize the availability of memory for this virtual machine compared to other virtual machines on this computer.
Low
High
Specifying a lower setting for this virtual machine might prevent it from starting when other virtual machines are running and available memory is low.
OK
Cancel
Apply

5. Select **Guest01,** right click and select **Start**



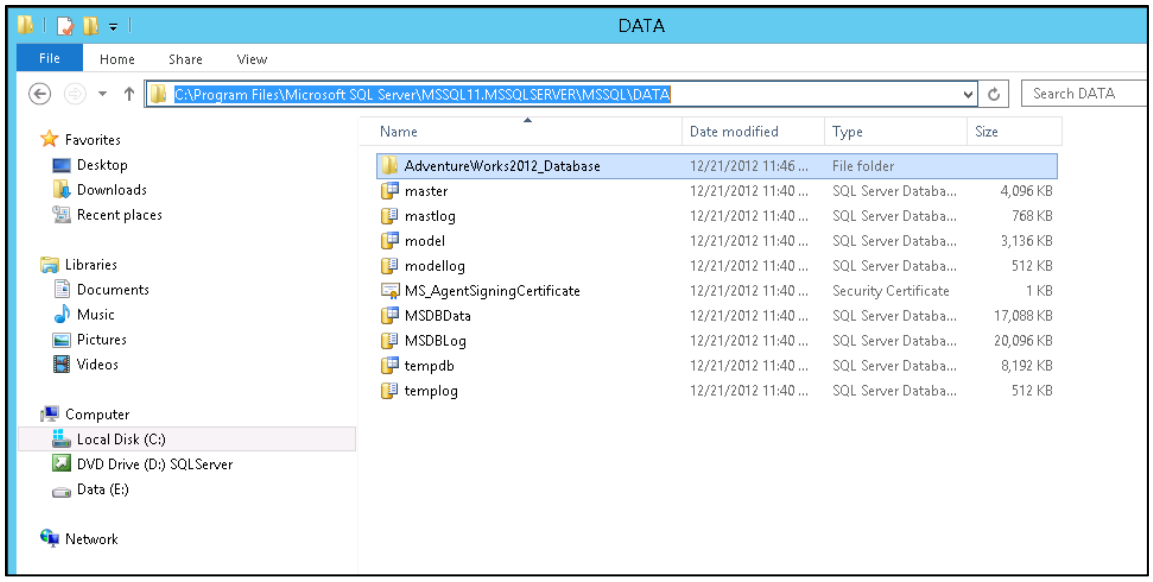
6. Select **Guest01**, right click and select **Connect**



7. Logon as **administrator** with the password **pass@word1**

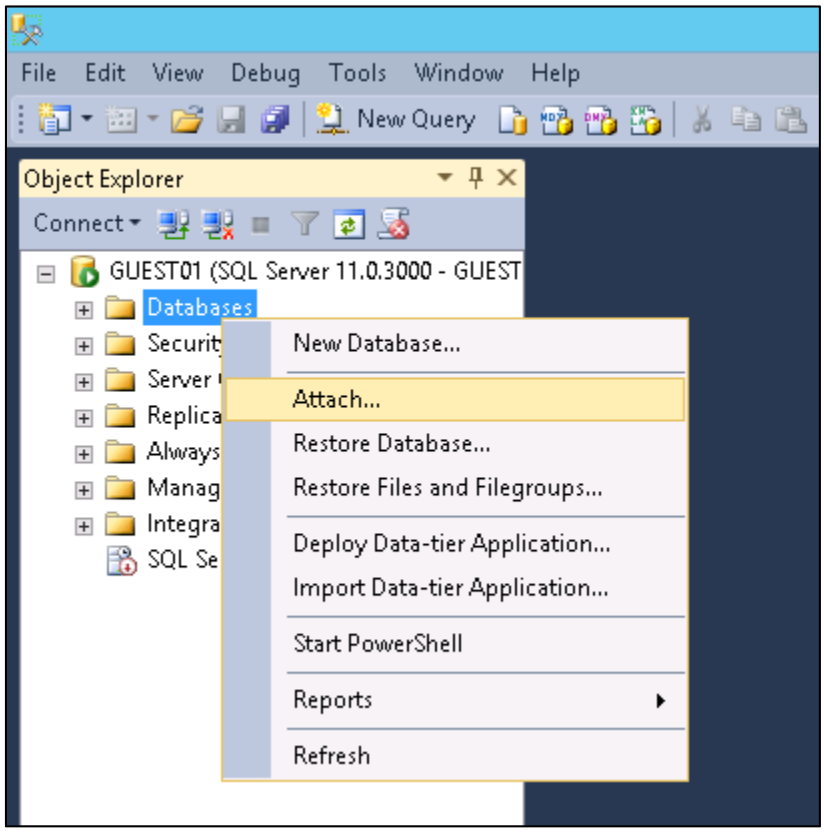
8. Download and extract, the Adventure Works 2012 sample database from codeplex website: http://msftdbprodsamples.codeplex.com/releases/view/93587

9. Copy the downloaded database files to C:\Program Files\Microsoft SQL Server\MSSQL11.MSSQLSERVER\MSSQL\DATA

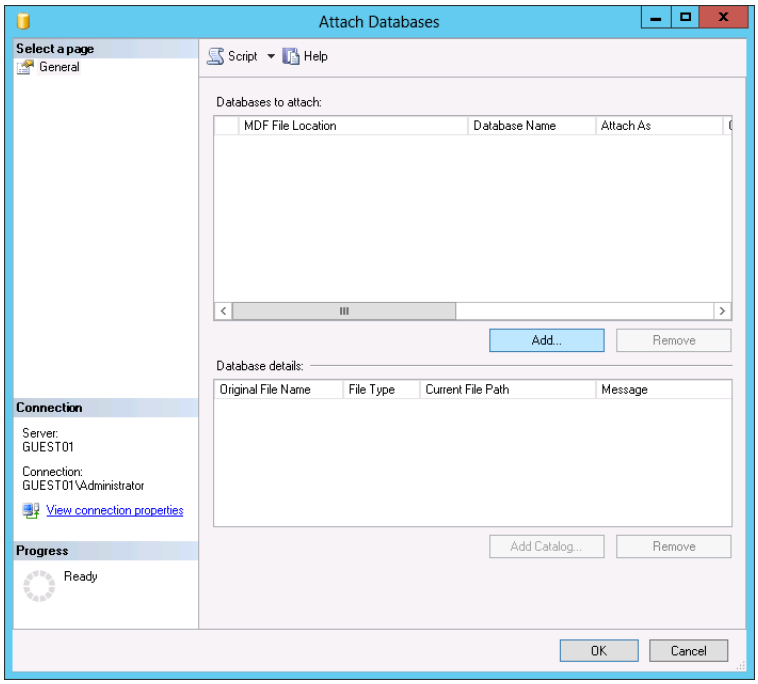


10. Start SQL Server Management Studio and click Connect

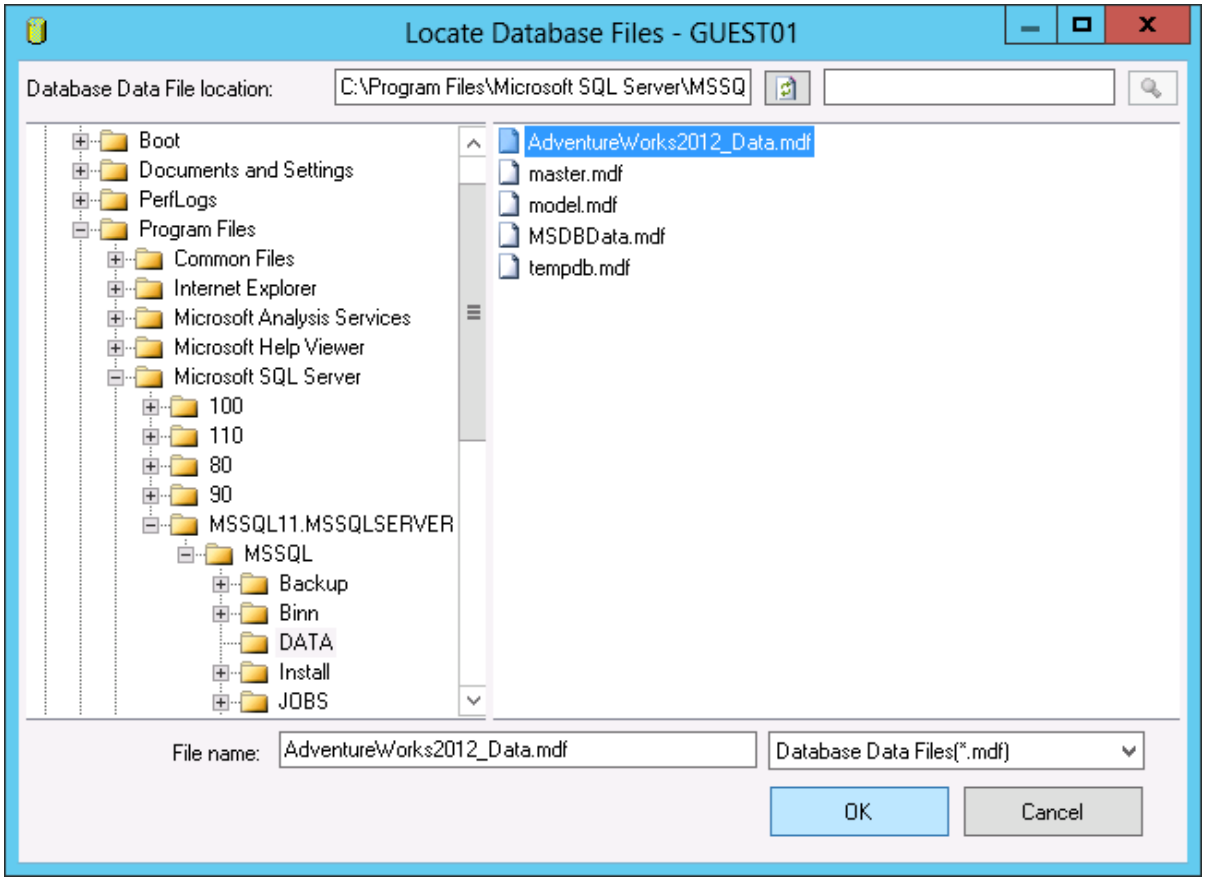
11. Select **Databases**, Right click and select **Attach**



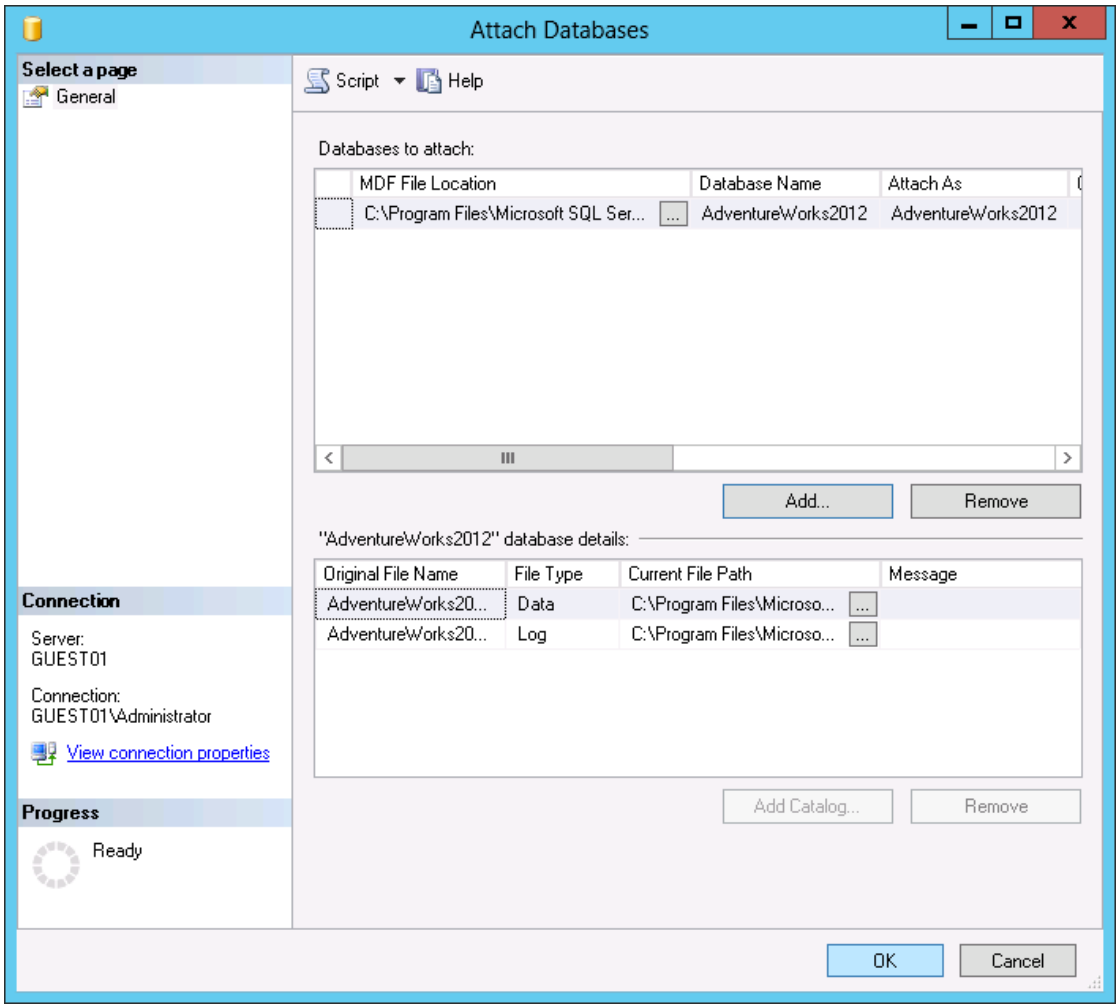
12. Click **Add**



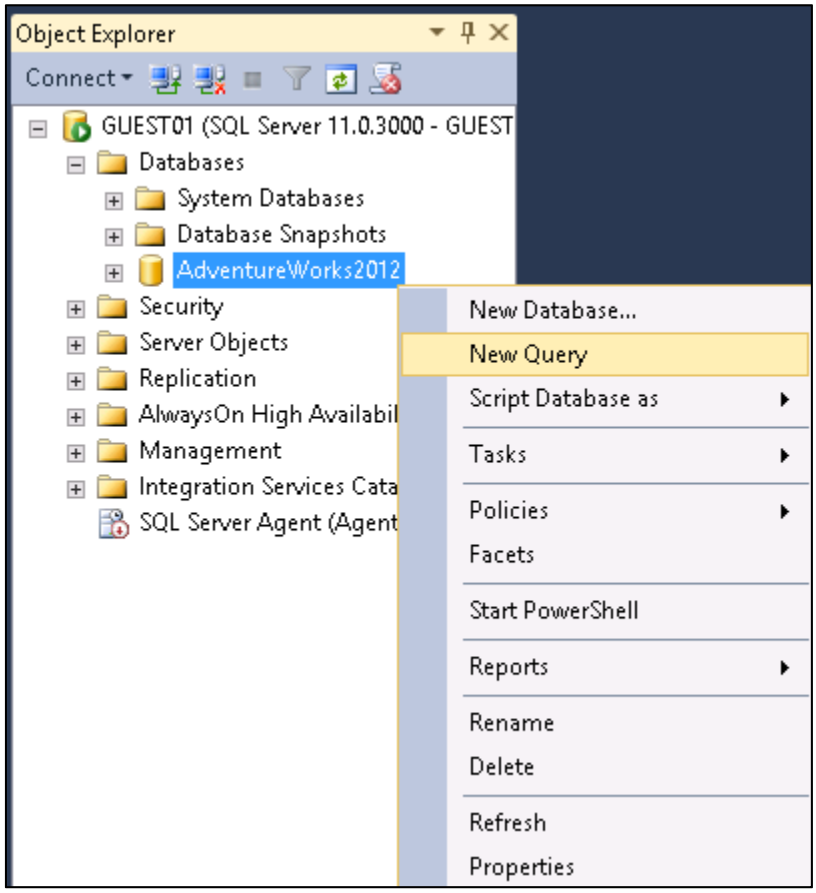
13. Browse to C:\Program Files\Microsoft SQL Server\MSSQL11.MSSQLSERVER\MSSQL\DATA and click OK



14. Click **OK** to attach the **AdventureWorks2012** database



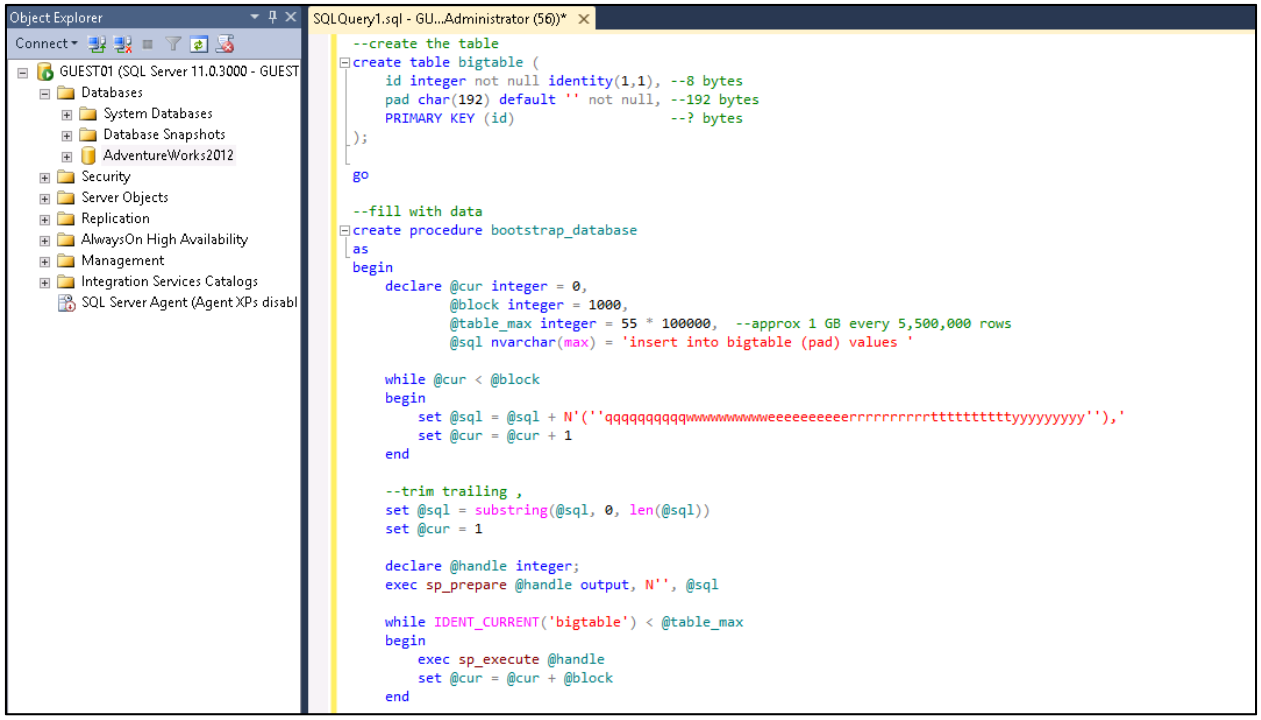
15. Select the **AdventureWorks2012** database by expanding the **Databases** section in **Object Explorer**, right click and choose **New Query**



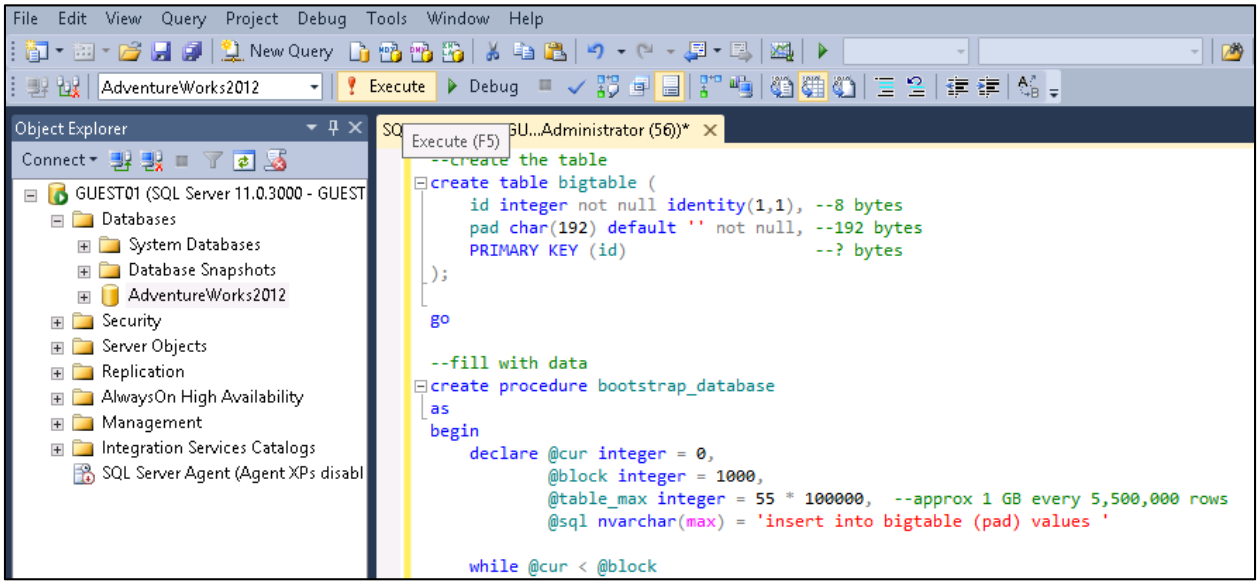
16. Copy and paste the following commands into the query window:

```
--create the table
create table bigtable (
id integer not null identity(1,1), --8 bytes
pad char(192) default '' not null, --192 bytes
PRIMARY KEY (id)                   --? bytes
);

go

--fill with data
create procedure bootstrap_database
as
```

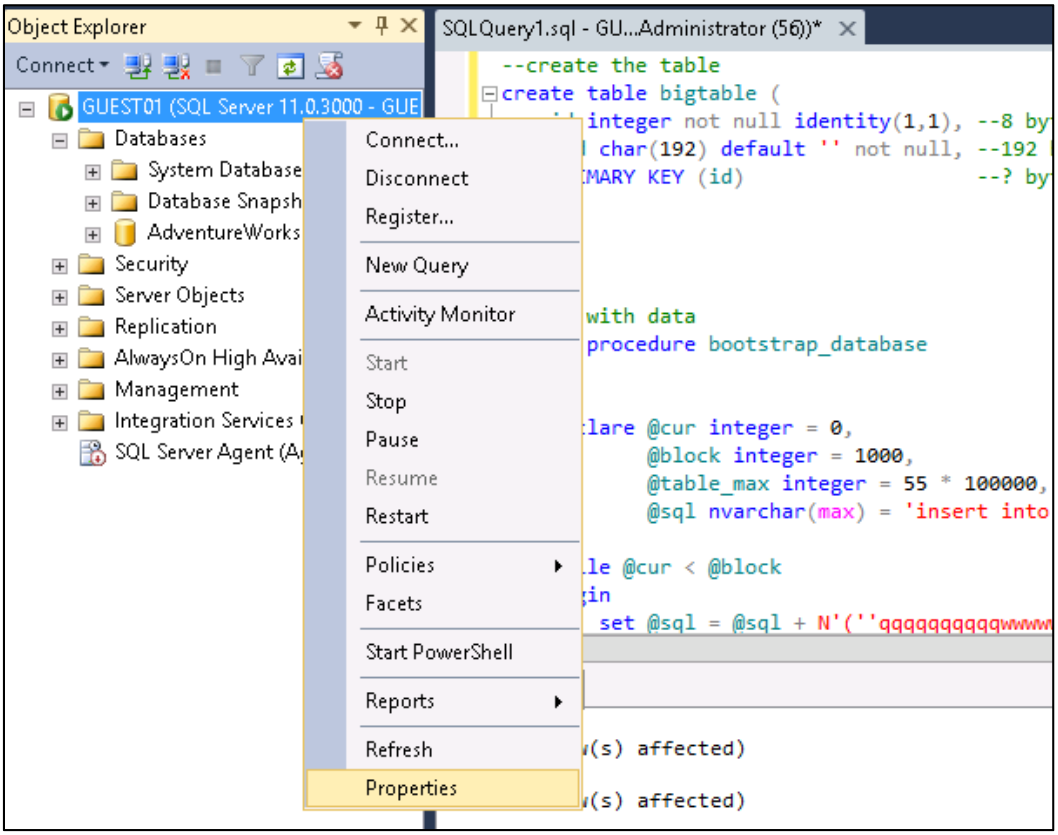
```
begin
declare @cur integer = 0,
@block integer = 1000,
@table_max integer = 55 * 100000,  --approx 1 GB every 5,500,000 rows
@sql nvarchar(max) = 'insert into bigtable (pad) values '

while @cur < @block
begin
set @sql = @sql + N'(''qqqqqqqqqqwwwwwwwwwweeeeeeeeeerrrrrrrrrrttttttttttyyyyyyyyyy''),'
set @cur = @cur + 1
end

--trim trailing ,
set @sql = substring(@sql, 0, len(@sql))
set @cur = 1

declare @handle integer;
exec sp_prepare @handle output, N'', @sql

while IDENT_CURRENT('bigtable') < @table_max
begin
exec sp_execute @handle
set @cur = @cur + @block
end

exec sp_unprepare @handle
end
go

exec bootstrap_database
```



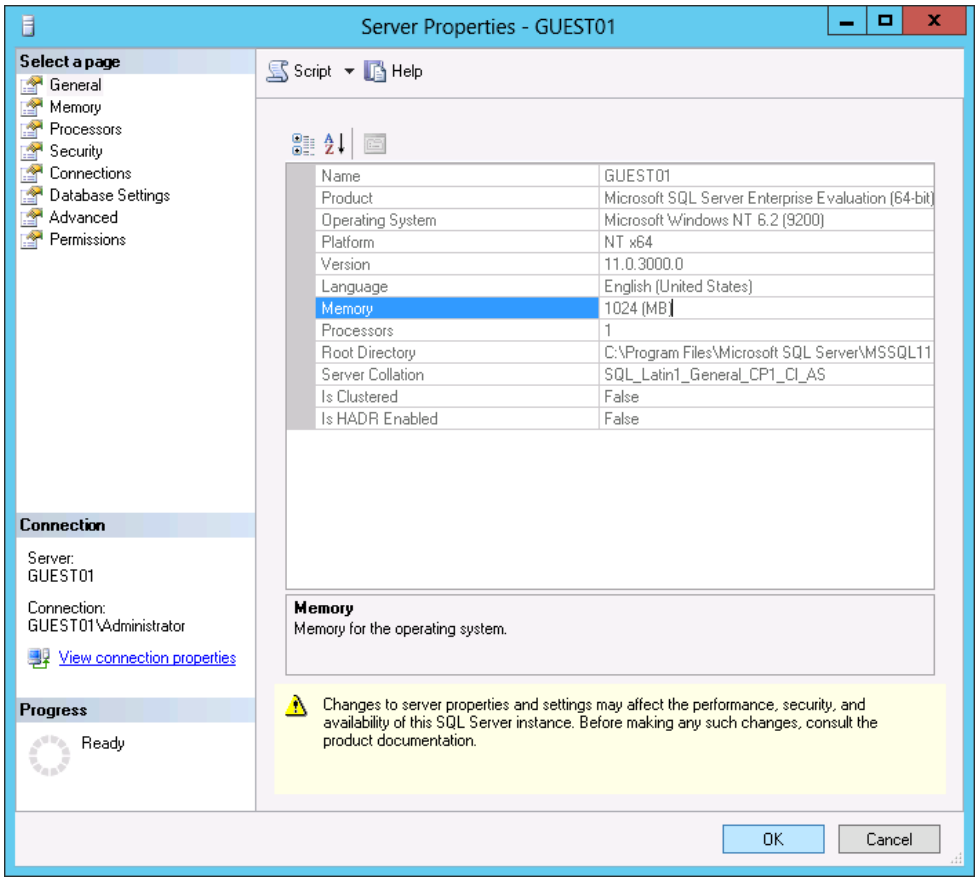
17. Click **Execute** – this will take a few minutes to complete



18. Once the script has completed, right click on **Guest01 (SQL SERVER ……)** from within **Object Explorer** and select **Properties**

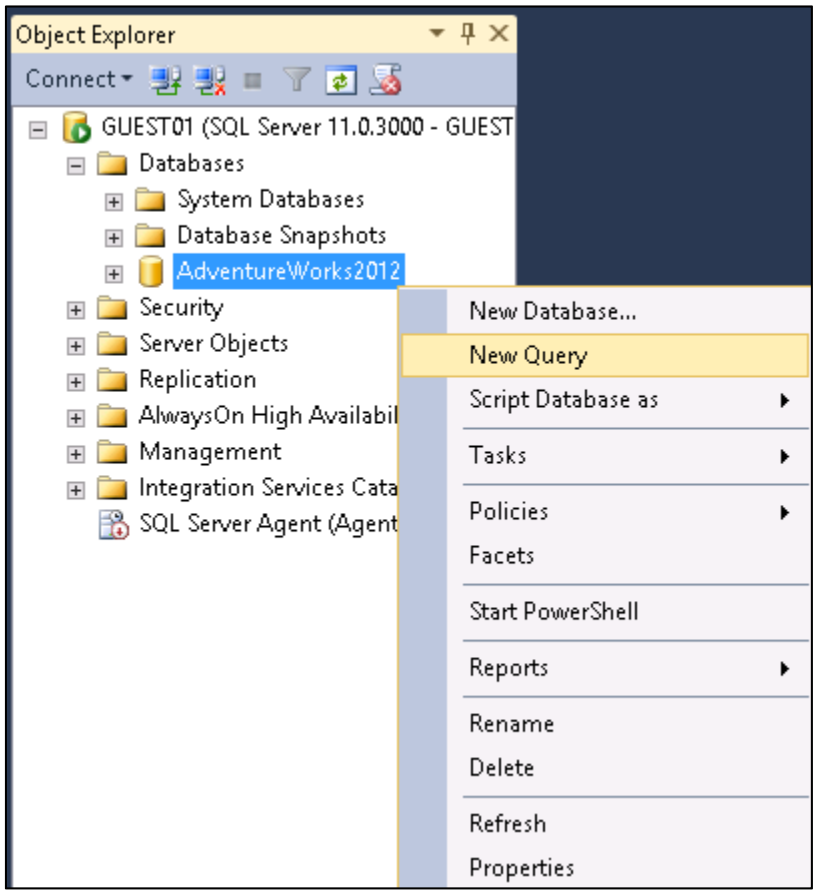


19. Note the **Memory** allocated to the SQL Server instance



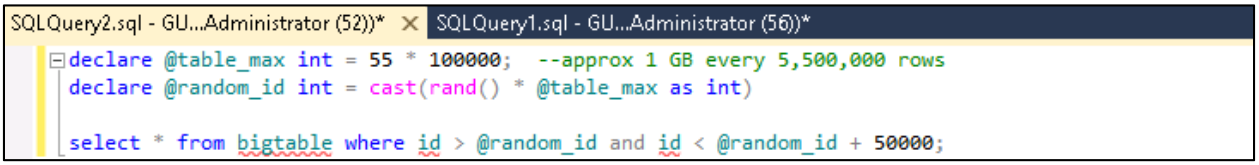
20. Click **OK**

21. Select the **AdventureWorks2012** database again, right click and choose **New Query**

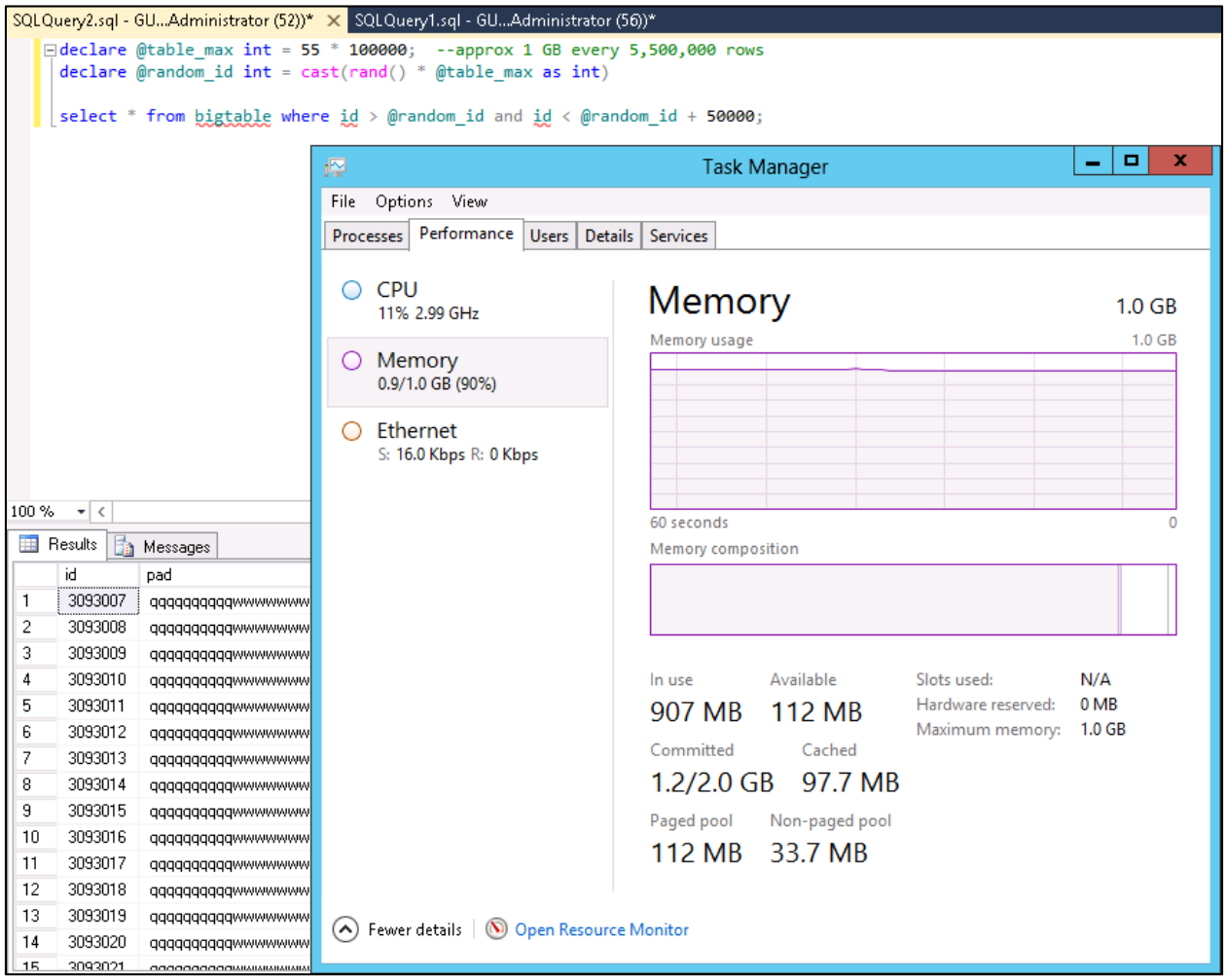


22. Enter the text in the box below into the new **Query** window, then **Execute Query**

```
declare @table_max int = 55 * 100000;  --approx 1 GB every 5,500,000 rows
declare @random_id int = cast(rand() * @table_max as int)

select * from bigtable where id > @random_id and id < @random_id + 50000;
```



23. Monitor the memory usage on **Guest01** with **Task Manager** and opening up the **Performance** tab

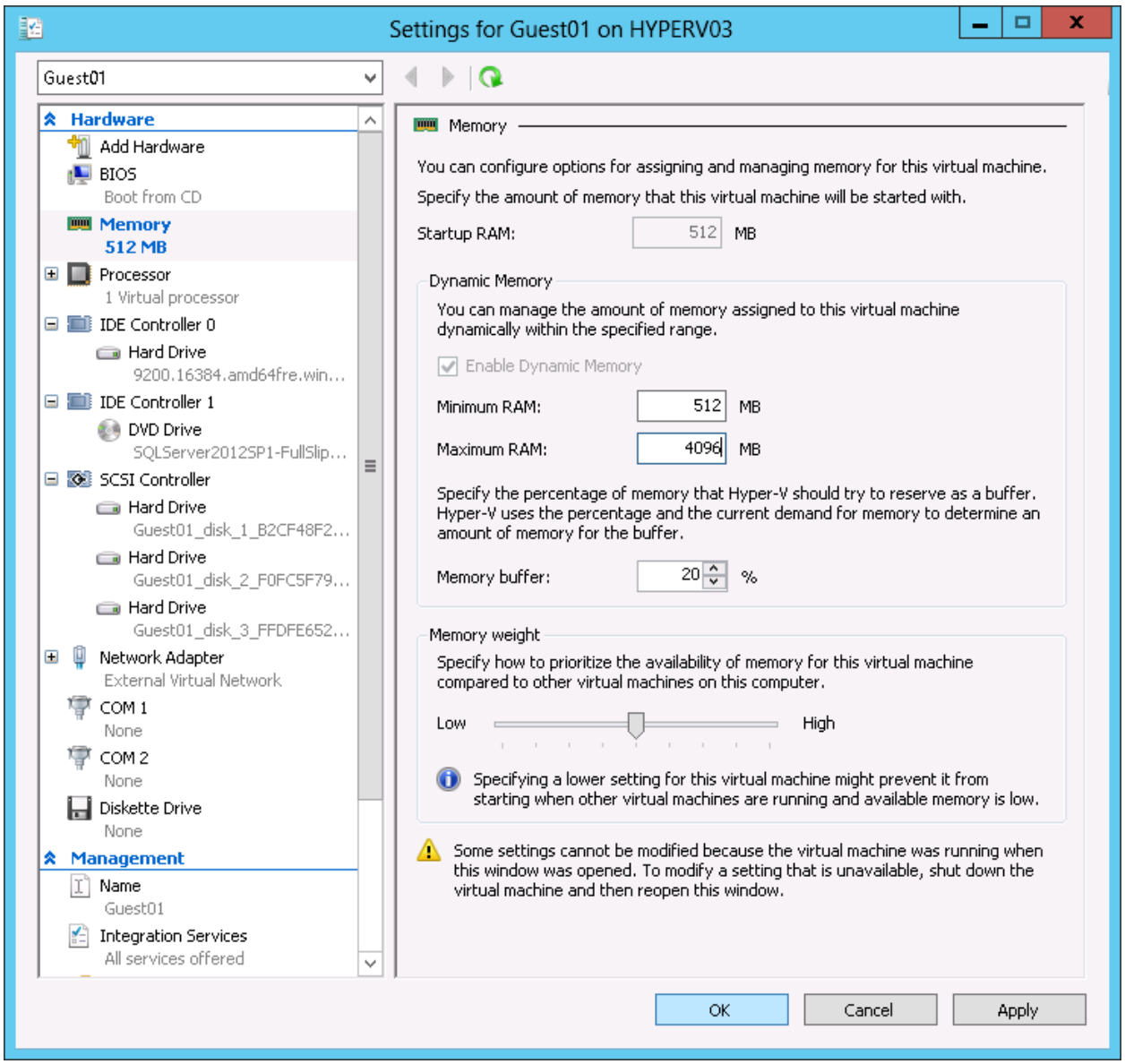


24. Notice that the memory on the server is now consumed by the SQL Server

25. Change to the **Hyper-V Manager** console on **HyperV03**

26. Right click on **Guest01**, click **Settings** and change the memory values as defined below

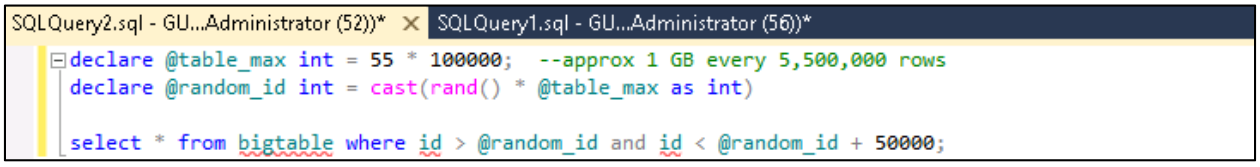
Minimum RAM	512MB
Maximum RAM	4096MB



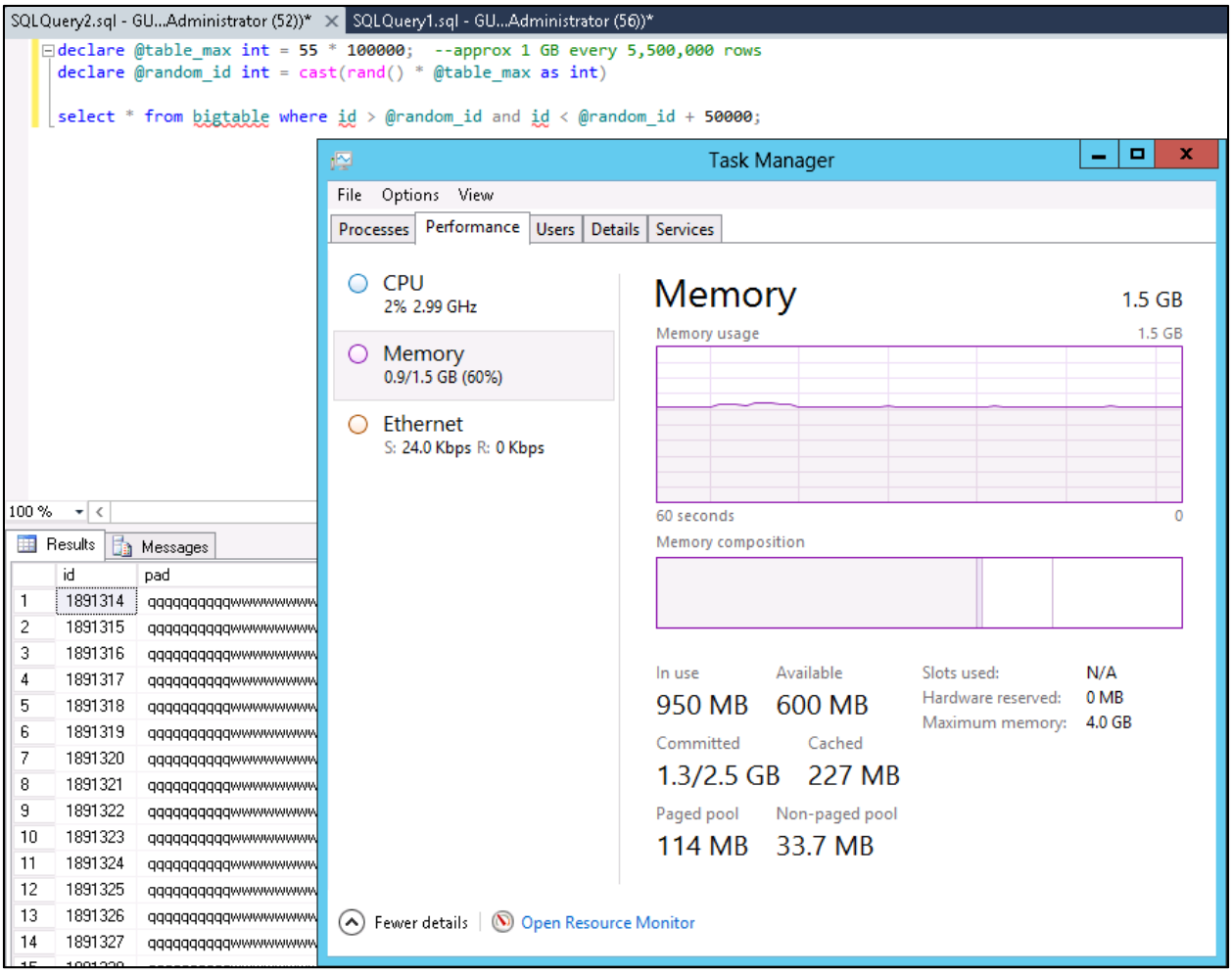
27. Click **OK**

28. Change back to the guest virtual machine **Guest01**

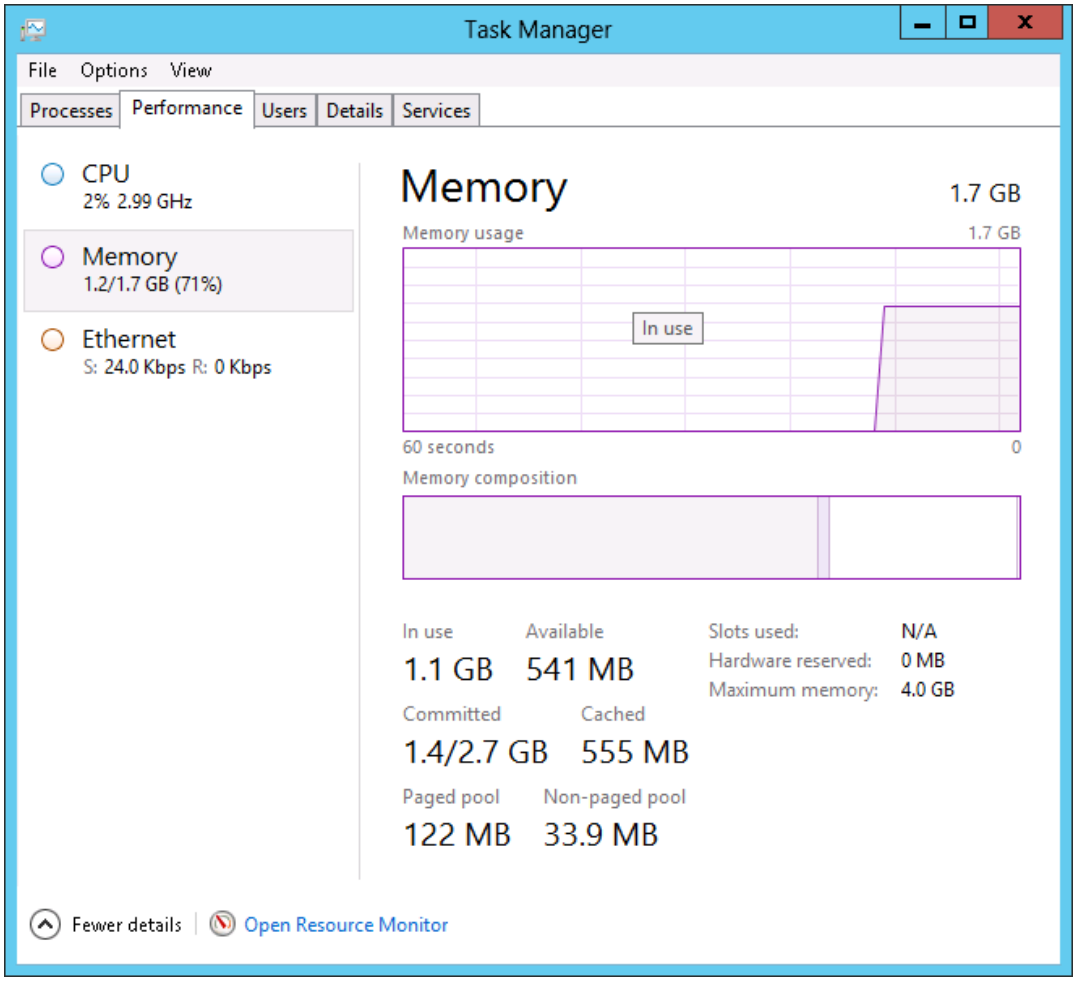
29. Execute the previous SQL Query again still in the **SQL Server Management Studio**



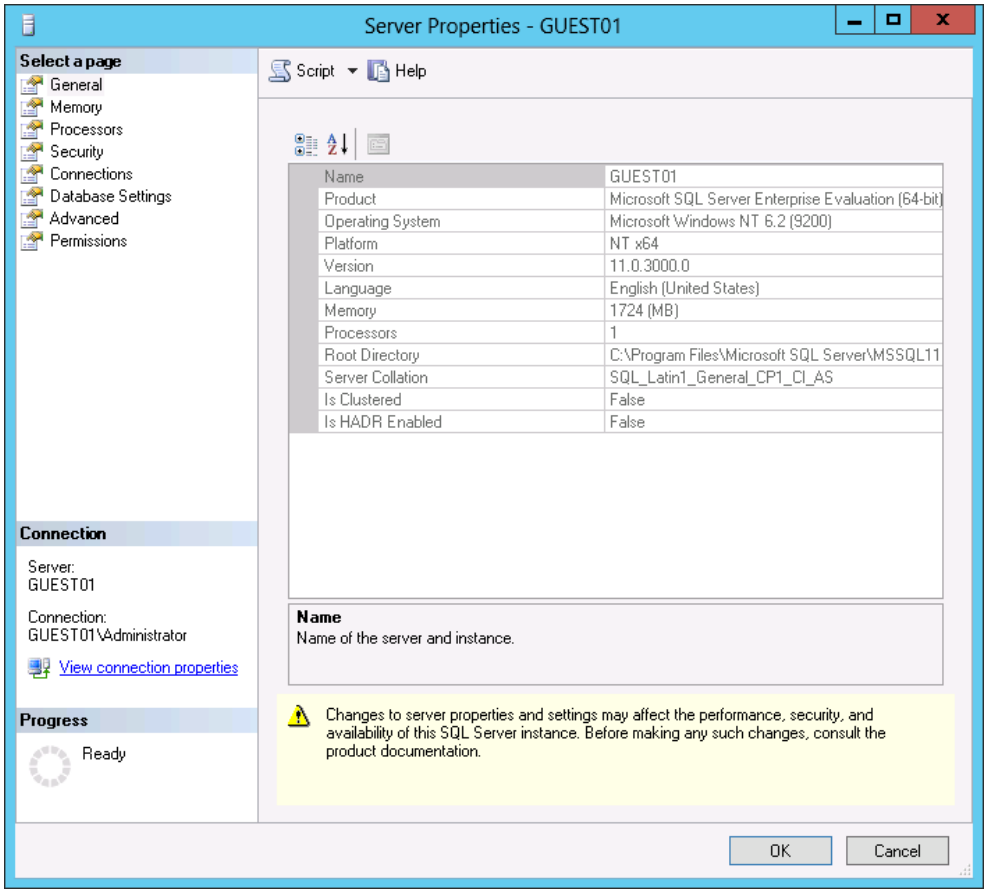
30. Using **Task Manager**, check the servers memory by selecting the **Performance** tab, and selecting **Memory**



31. Notice that the system memory has increased **dynamically**



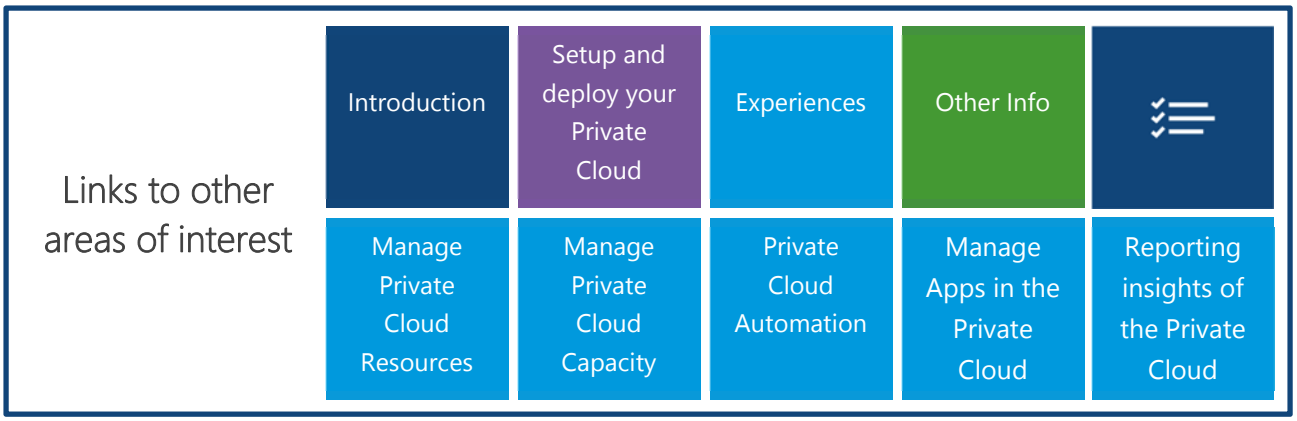
32. Using the SQL Server Management Studio, right click on Guest01 (SQL SERVER .....) and select Properties



Jeff has successfully implemented Dynamic Memory and as it was seen in the scenario, SQL 2012 automatically consumed the memory hot-added to it without stopping the background processing of transactions. This is a major advantage in the ability to respond quickly to performance problems within Server environments where resources are required to be added without interrupting an application. SQL 2012 combined with Server 2012 and Hyper-V Dynamic Memory allows administrators to rectify issues of resource constraint immediately.

Important Note

Note the **Memory** allocated to the SQL Server instance has also increased dynamically, allowing the SQL Server instance to consume the additional memory allocated to the operating system, without taking the server offline.


Links to other areas of interest
Introduction
Setup and deploy your Private Cloud
Experiences
Other Info
Manage Private Cloud Resources
Manage Private Cloud Capacity
Private Cloud Automation
Manage Apps in the Private Cloud
Reporting insights of the Private Cloud

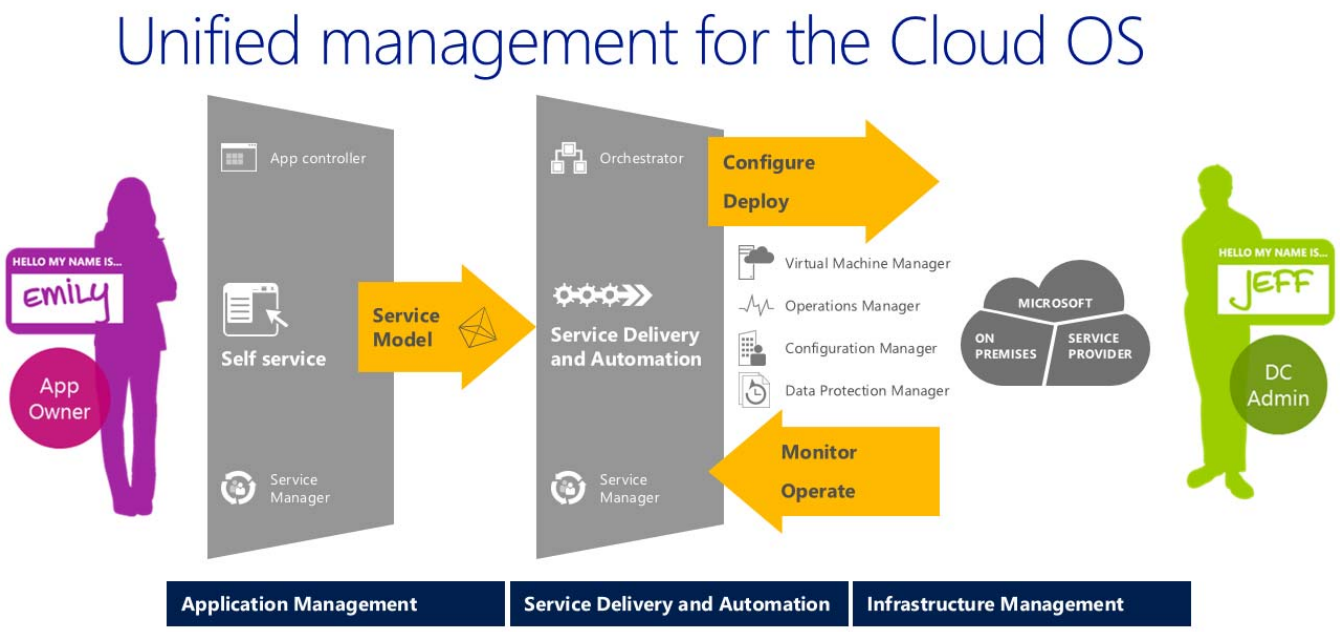
Experience 3: Private Cloud automation

This experience focuses on how to gain operation efficiencies through automation, control through self-service and delegation and delivers a consistent management experience of the infrastructure. Organizations will have a perspective of both the business unit owner to make requests for cloud resources as well as the datacenter admin perspective of building extreme automation to control the environment effectively with little overhead. The key components that are leveraged in this experience will be Service Manager, Orchestrator, App Controller and Virtual Machine Manager.

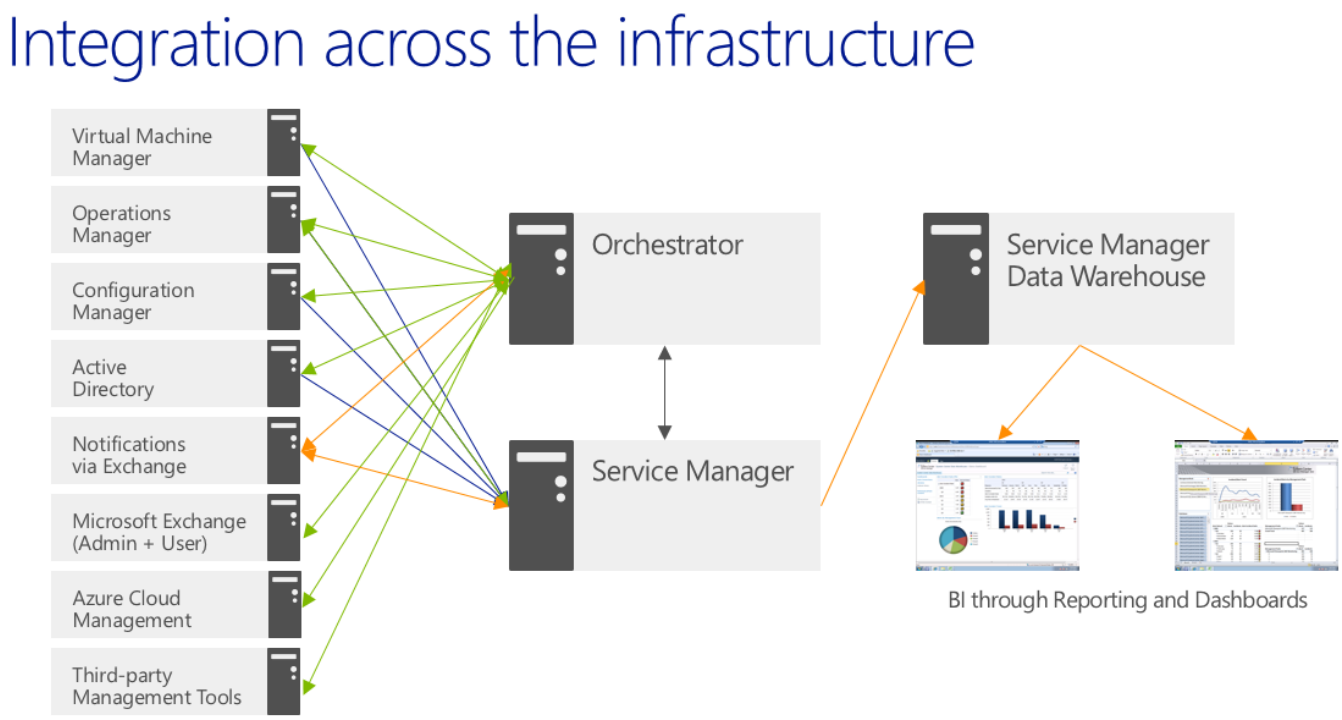
One of the biggest strengths of System Center 2012 is its integrated approach to workflow, knowledge and data management. System Center 2012 has connectors and integration packs that simplify communication between the components that help to provide full life-cycle management of your applications and infrastructure. In this chapter we will start with the overview of how to connect the components together.

Now that Jeff has installed Microsoft System Center 2012 SP1, he is going to set up the connections between the components. Because Jeff's goal is to make sure that Emily's application is monitored, deployed, and upgradable, he needs to ensure that all of the components are working and communicating together. In this chapter, we are going to create all these connections.

Bringing the System Center 2012 Components Together



When we look back at the model of interaction between Emily and Jeff, we see that the Orchestrator and Service Manager components are located in the middle of the model. This is not a coincidence. Service Manager and Orchestrator automate the workflows, the data management required to deliver self-service and efficiently deliver the private cloud SLAs.



In this figure we see how the components work together. We will now describe the different connections and then Jeff will configure them in his environment.

Orchestrator and the Integration Packs

Jeff knows that Orchestrator comes with an entire set of automation activities out of the box. But Orchestrator can easily be extended by using integration packs. An integration pack is a combination of activities that can extend the functionality of Orchestrator to provide more automation capabilities and integration into other third party applications. In this evaluation, Jeff is going to use the Integration Packs for System Center 2012 SP1.

Downloading Integration Packs

The integration packs can be downloaded from the following location:

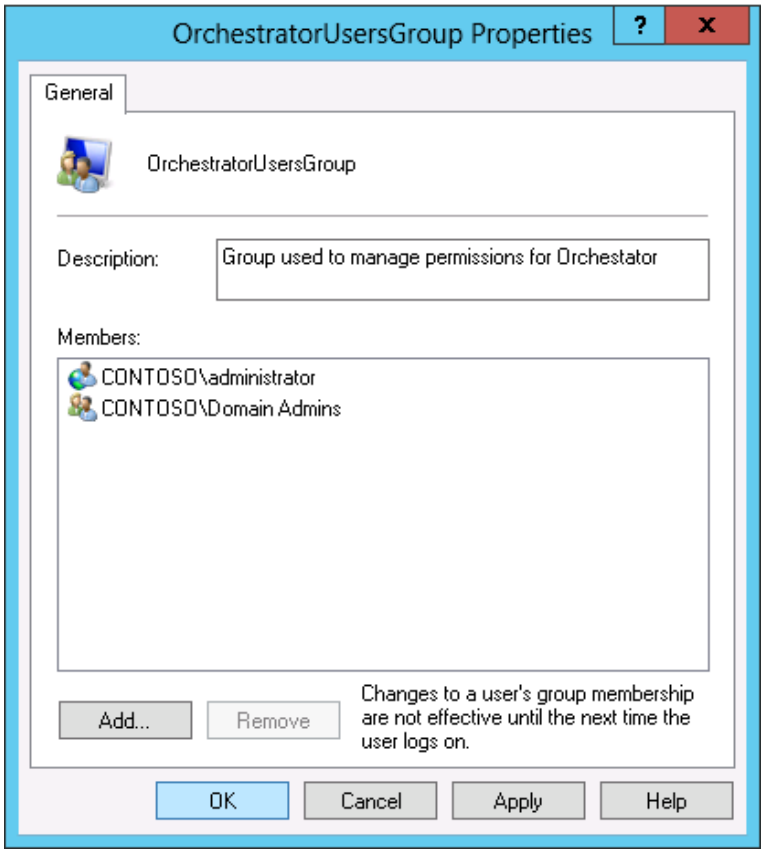
http://www.microsoft.com/en-us/download/details.aspx?id=34611

Registering Integration Packs

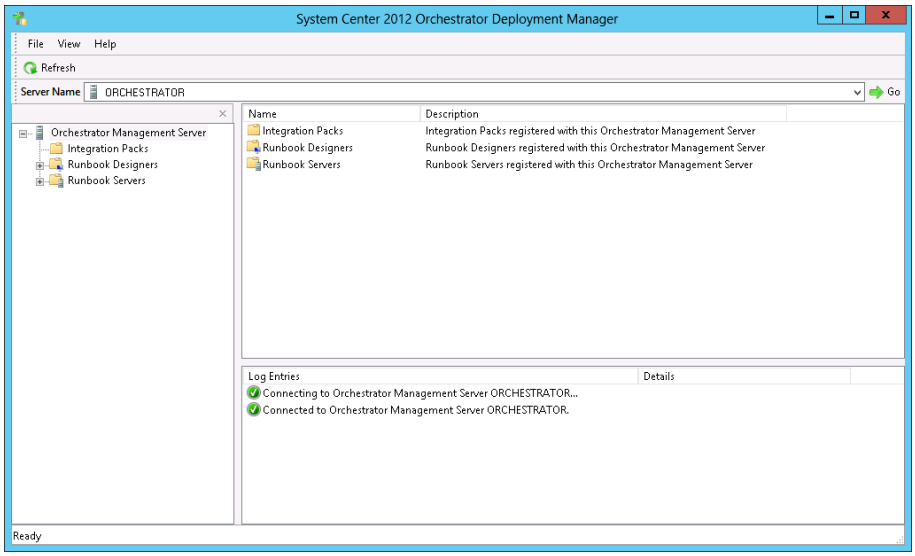
Before we can use an integration pack in Orchestrator, we need to register it. **Jeff** is going to register the **Microsoft Virtual Machine Manager** integration pack.

1. Connect to the **Orchestrator** server as **contoso\administrator** and open **Local Users and Groups**

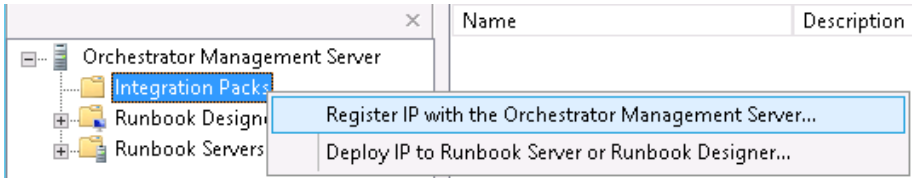
2. Open the Group **OrchestratorUsersGroup** and add **contoso\Domain Admins** to allow Jeff to manage Runbooks



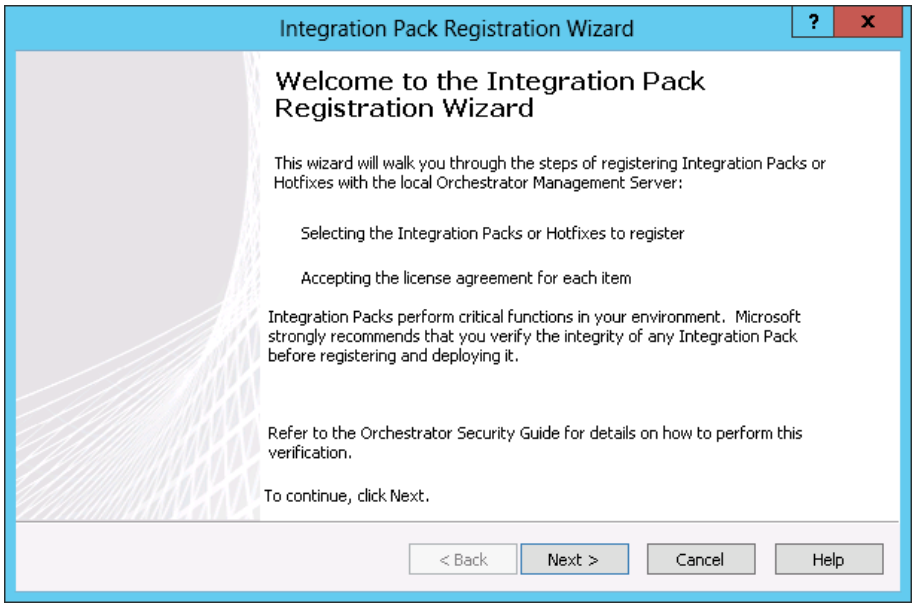
3. Open the **System Center 2012 Orchestrator Deployment Manager**



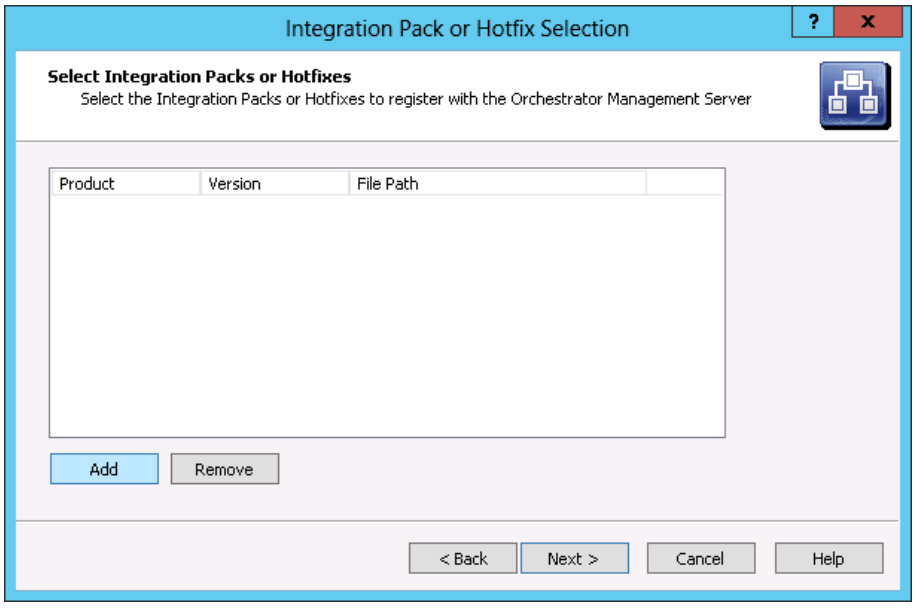
4. Right-click on **Integration Packs** and choose **Register IP with the Orchestrator Management Server...**



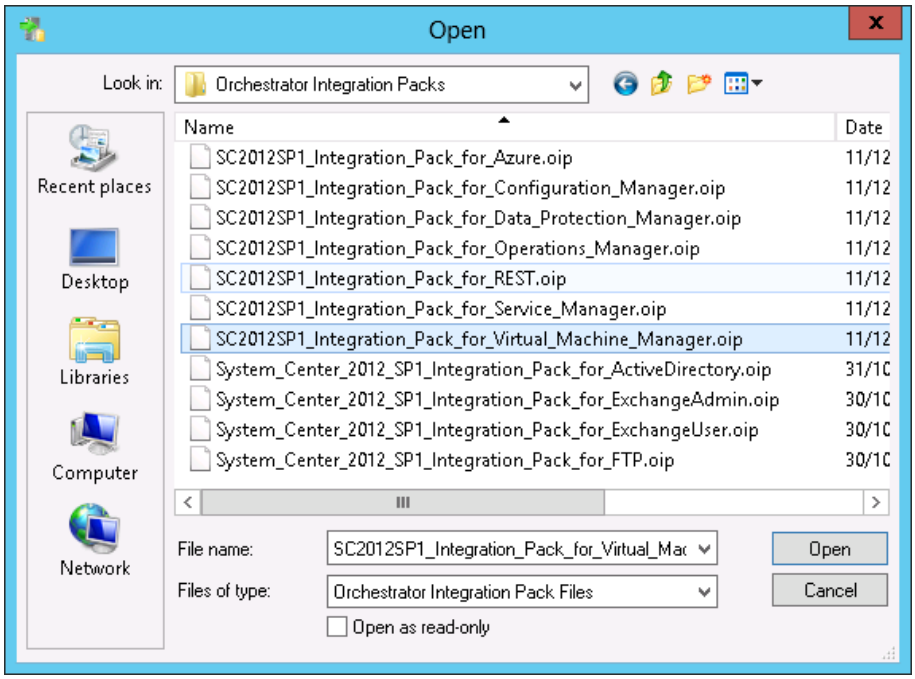
5. On the **Welcome to the Integration Pack Registration Wizard** select **Next**



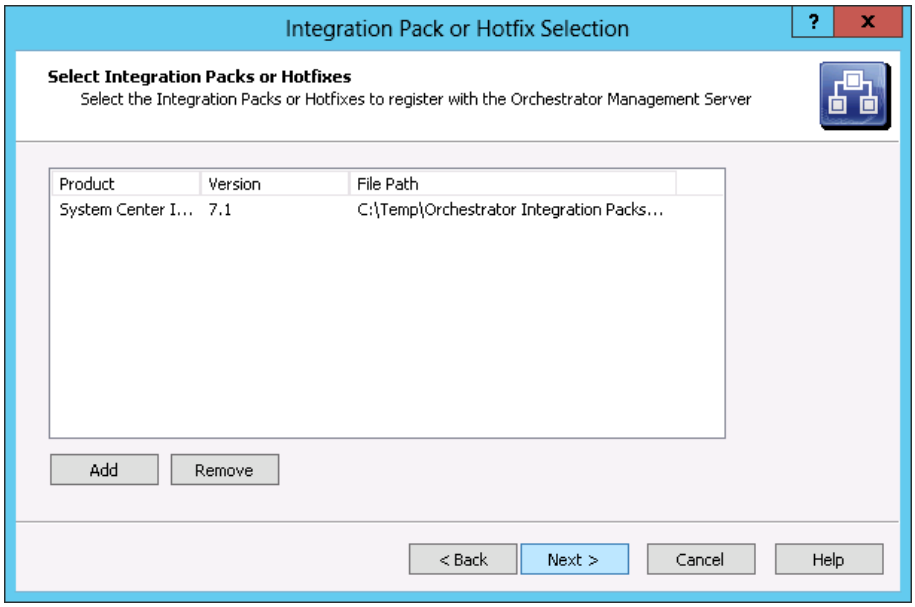
6. Select the **Add** button and browse to the location where you stored your Integration Packs



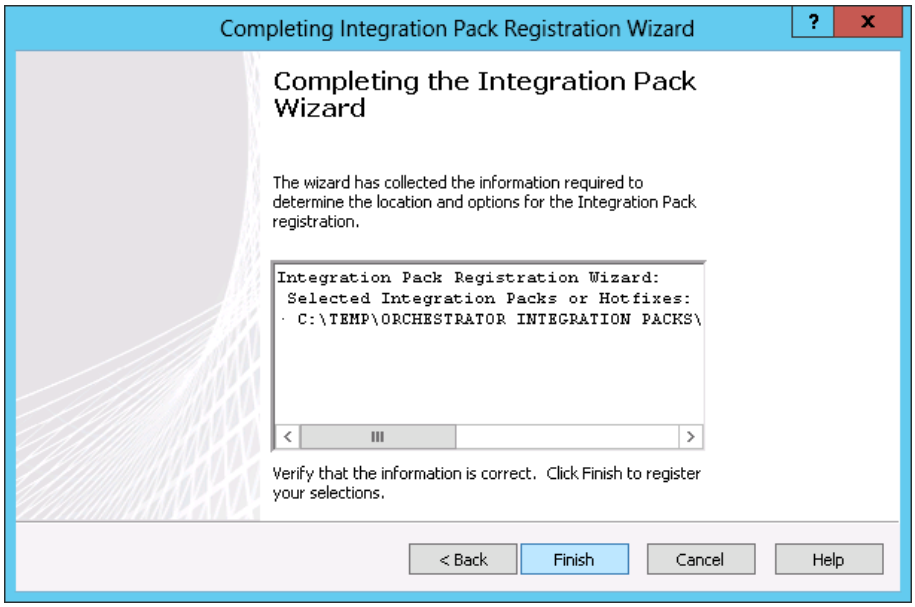
7. Select the **System Center 2012 Virtual Machine Manager Integration Pack** and the select **Open**



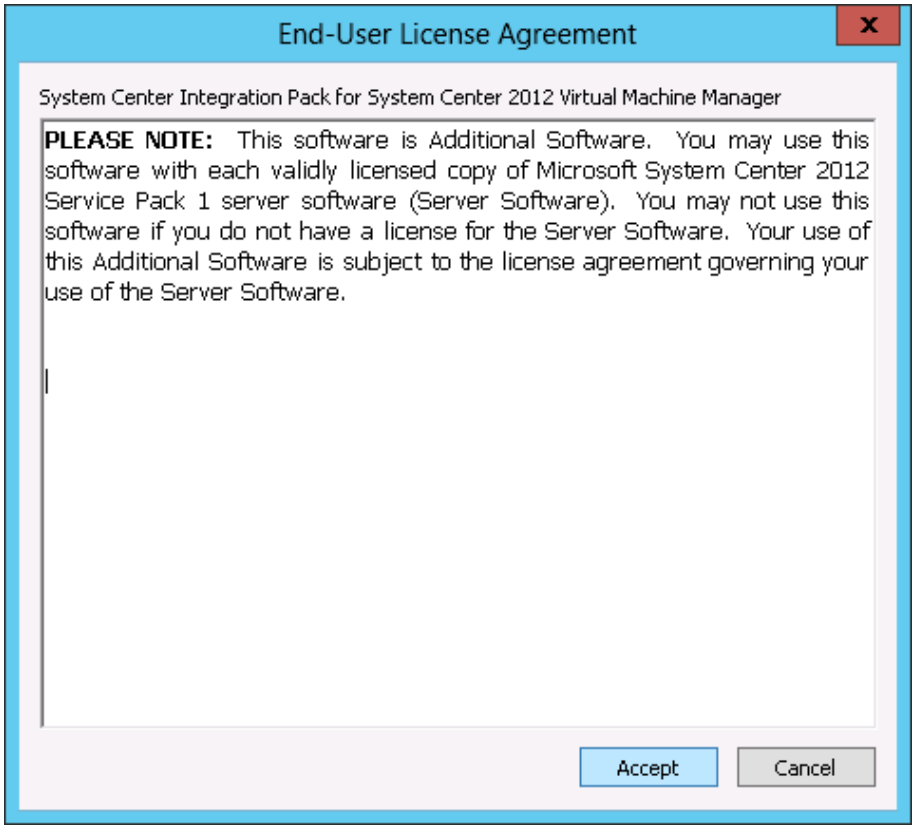
8. Select **Next** when you have added the management pack



9. On the **Completing the Integration Pack** page, select **Finish**



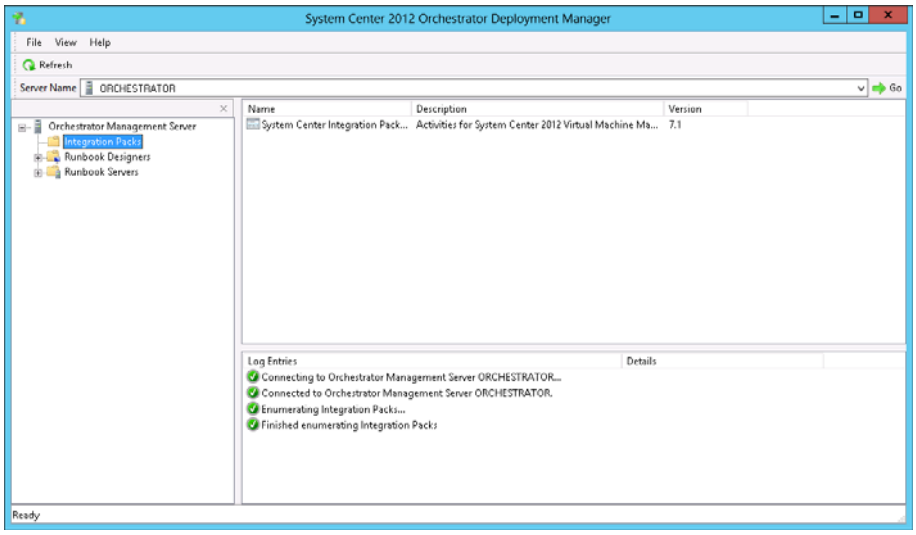
10. Accept the **End-User License Agreement** by selecting **Accept.** You will have to do this multiple times (one EULA per integration pack)



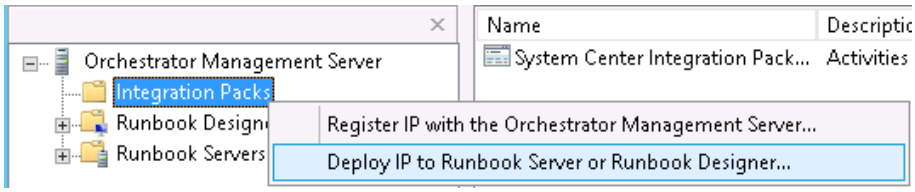
Deploying Integration Packs

Now that Jeff has registered the Integration Packs, he also needs to deploy them to each Runbook server and to each server / workstation with a Runbook designer that will use the integration packs. In our evaluation, Jeff is going to deploy them on the Orchestrator server as this is the server that he is going to use for the evaluation.

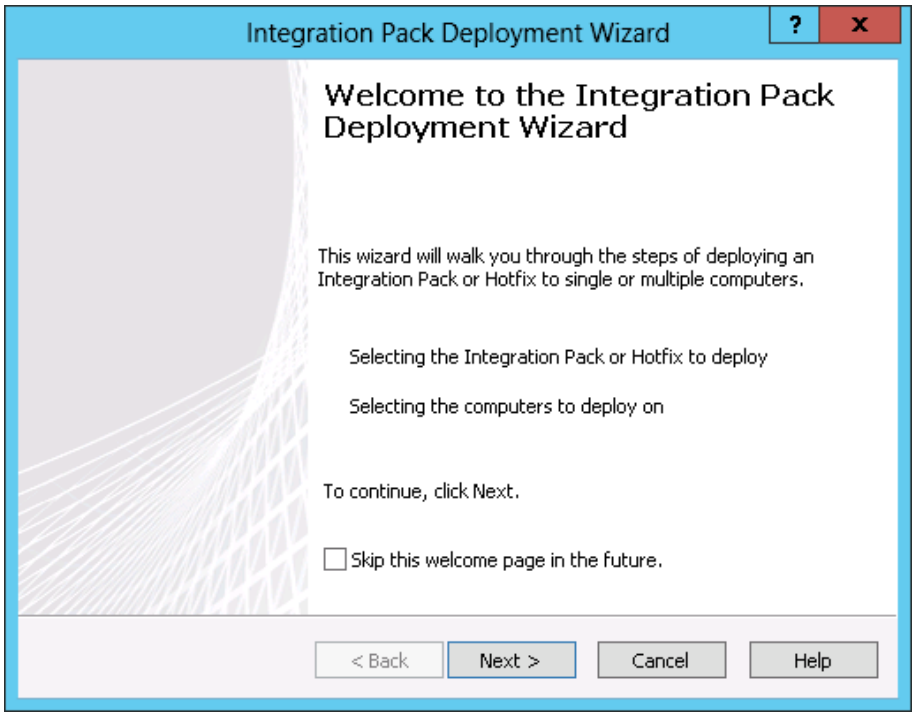
1. Open the **System Center 2012 Orchestrator Deployment Manager**



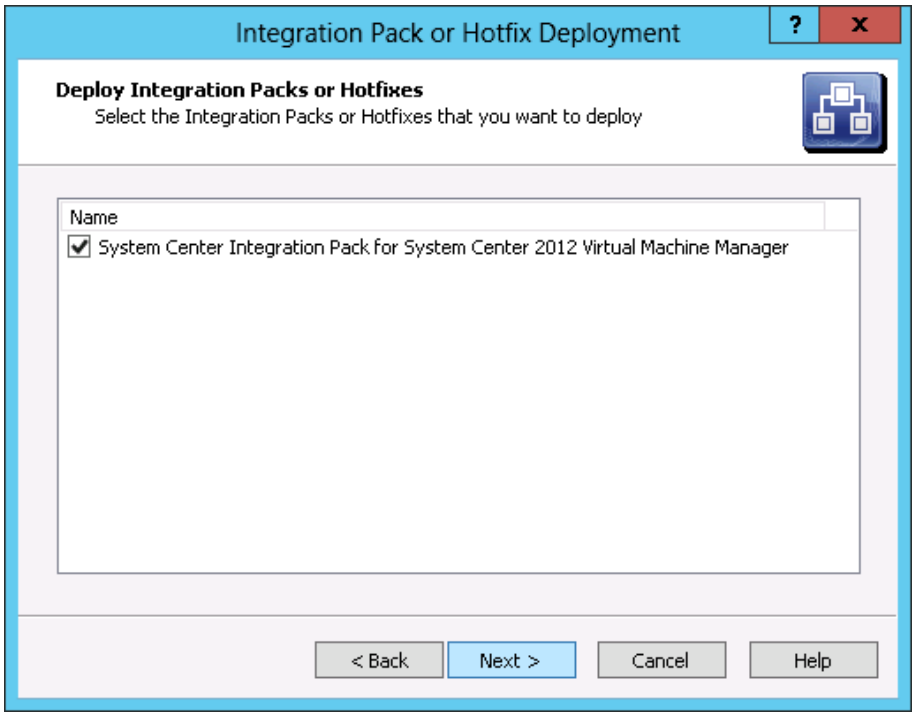
2. Right-click on **Integration Packs** and choose **Deploy IP to Runbook Server or Runbook Designer…**



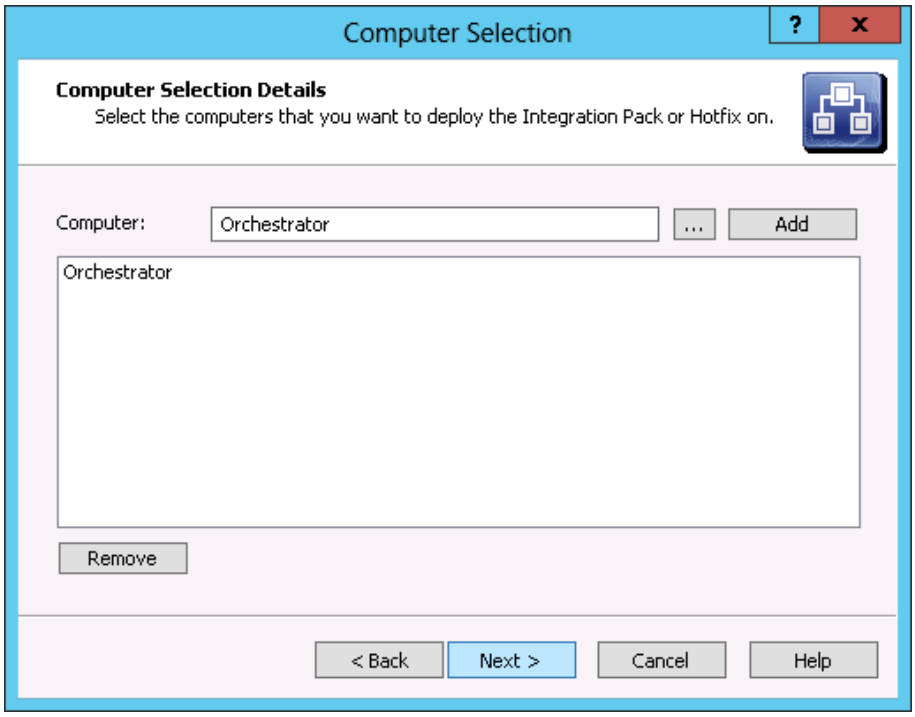
3. On the **Welcome to the Integration Deployment Wizard** select **Next**



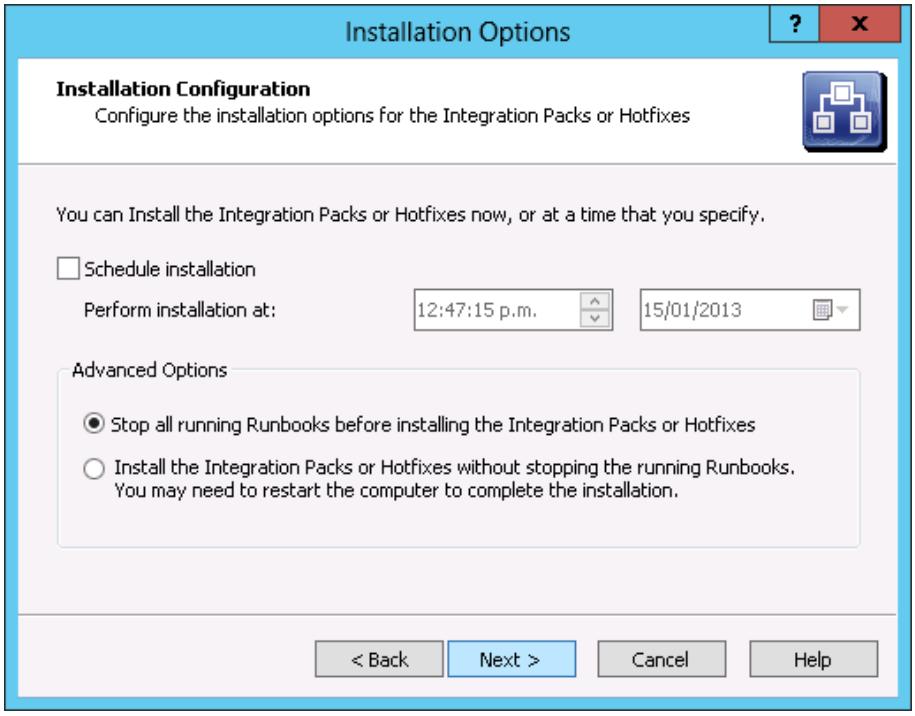
4. On the **Deploy Integration Packs or Hotfixes** page select the integration pack that Jeff registered and then select **Next**
    - System Center Integration Pack for System Center 2012 Virtual Machine Manager



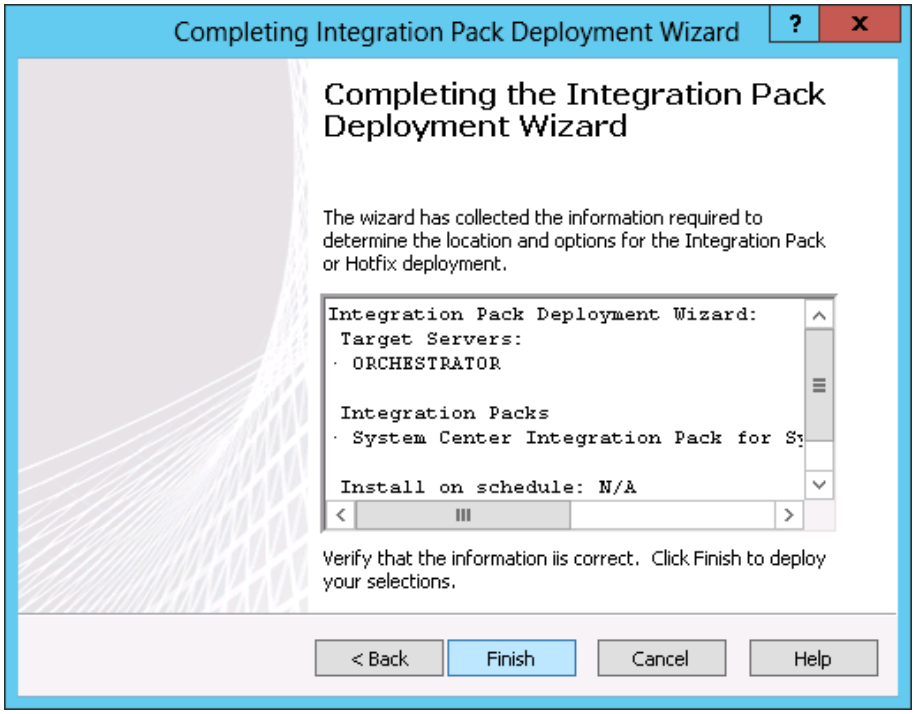
5. On the **Computer Selection Details** page, type in the name of the runbook server (in our case this is **Orchestrator**) and select the **Add** button. Then select **Next**



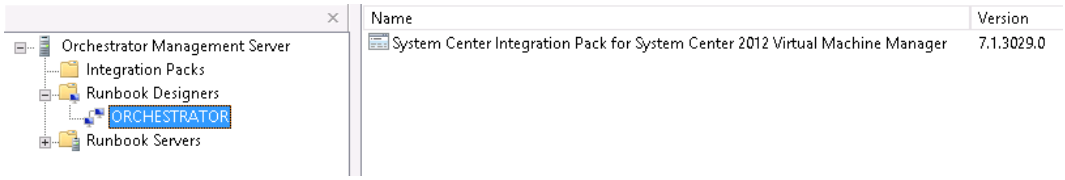
6. On the **Installation Configuration** page, Jeff is going to choose to deploy the **Integration Packs** now. This will stop all the running runbooks on that server. If that would be a problem, then **Jeff** could have chosen to schedule the installation or to install the **Integration Packs** without stopping the runbooks and restart the server afterwards.



7. On the **Completing the Integration Pack Deployment** page, select **Finish**



8. You can verify that the deployment is successful by looking at the **Runbook Designers** folder within the Orchestrator Deployment Manager to see that the **Integration Packs** are there. You can do the same by looking at the **Runbook Servers** and selecting **Orchestrator** to see if the **Integration Packs** are shown.



Configuring the System Center 2012 Virtual Machine Manager Integration Pack

Before Jeff can actually use the Integration Packs, he needs to configure them so that they have a connection to the correct server(s) and that they have a username and password to connect to these servers. In the next steps, Jeff is going to configure the four integration packs that he just registered and deployed.

Before Jeff can configure an integration pack, he needs to make sure that the prerequisites are OK. In the documentation on TechNet, he finds all the information that he needs:

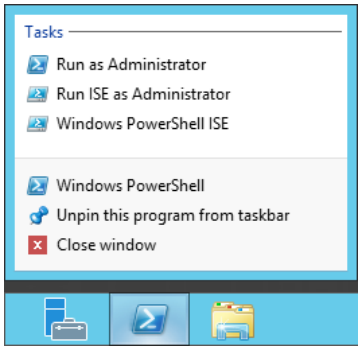
http://technet.microsoft.com/en-us/library/hh295851.aspx

To configure the **System Center 2012 Virtual Machine Manager Integration Pack**, Jeff needs to have the following prerequisites:

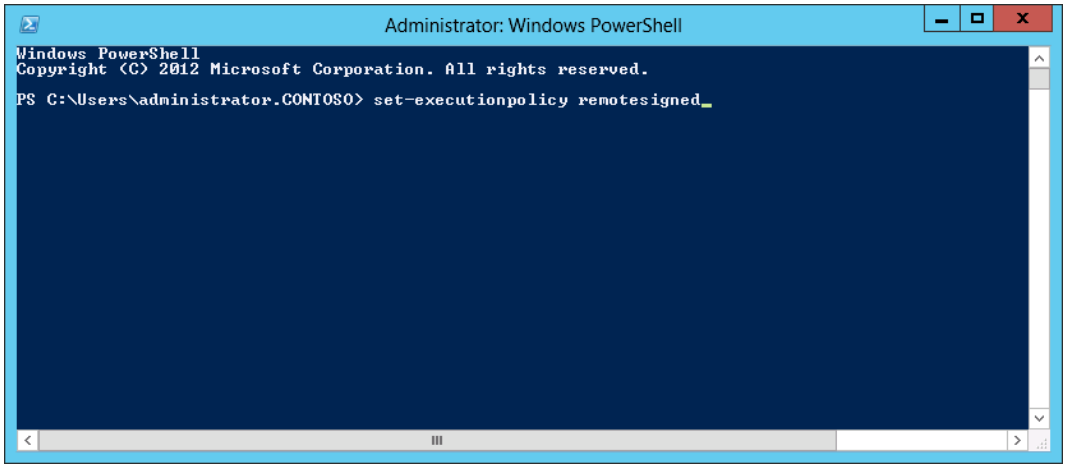
- System Center 2012 – Orchestrator
- System Center 2012 – Virtual Machine Manager (VMM)
- Windows Management Framework (Windows PowerShell 2.0 and WinRM 2.0)

Jeff has deployed the **Orchestrator** server and the **VMM** server in Chapter 3. The **Orchestrator** server is configured on **Windows Server 2008 R2 SP1** so the Windows Management Framework is also covered. Last but not least, **Jeff** needs to change the execution policy of PowerShell. The execution policy in Windows PowerShell determines which scripts must be digitally signed before they will run. By default, the execution policy is set to **Restricted** which prohibits loading any configuration files or running any scripts.

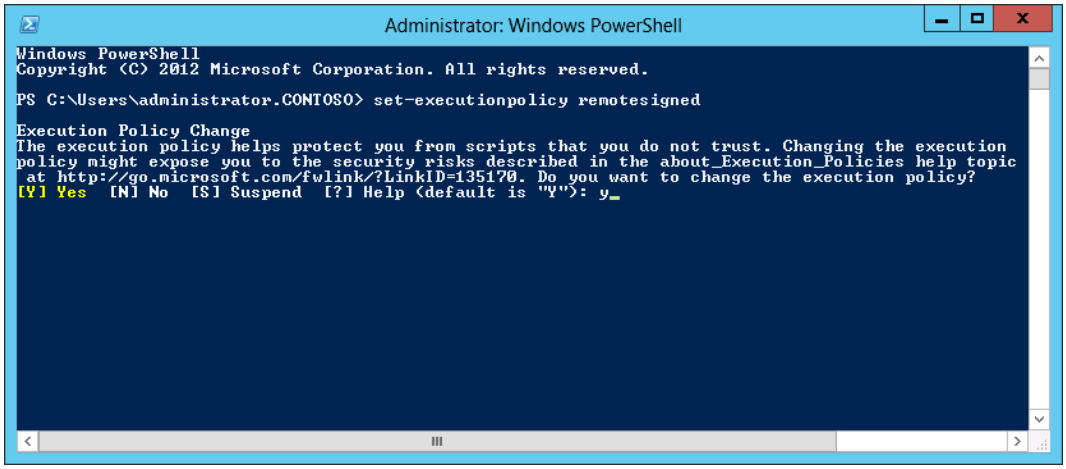
1. Right Click on the **PowerShell** icon and choose **Run As Administrator**



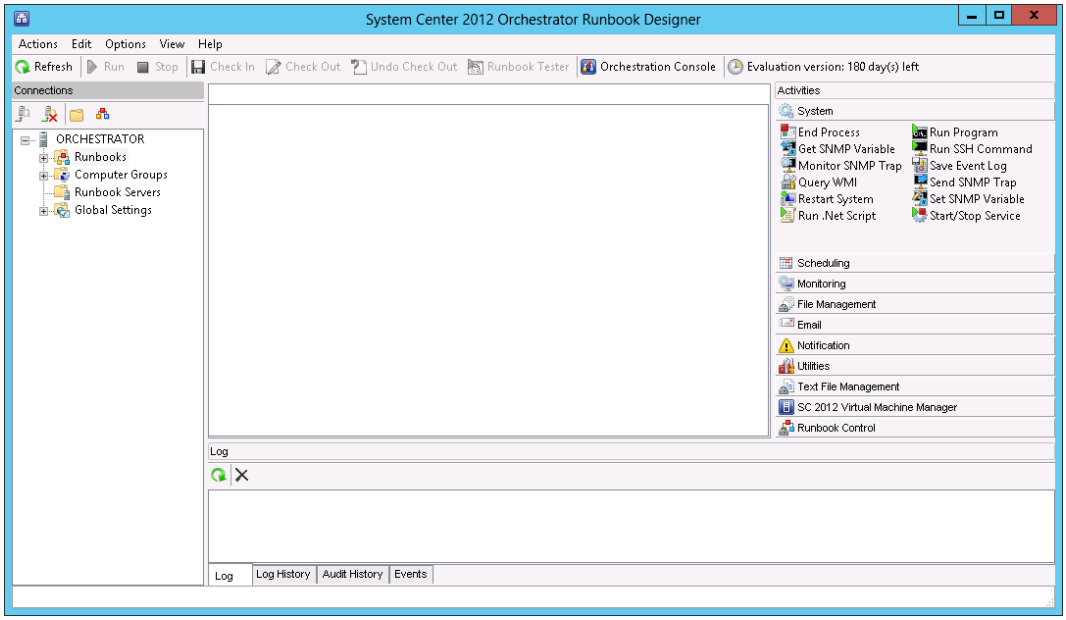
2. In the **PowerShell** window, type **set-executionpolicy remotesigned** and select **Enter**



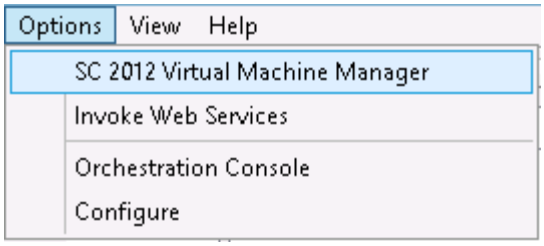
3. Type **Y** when asked



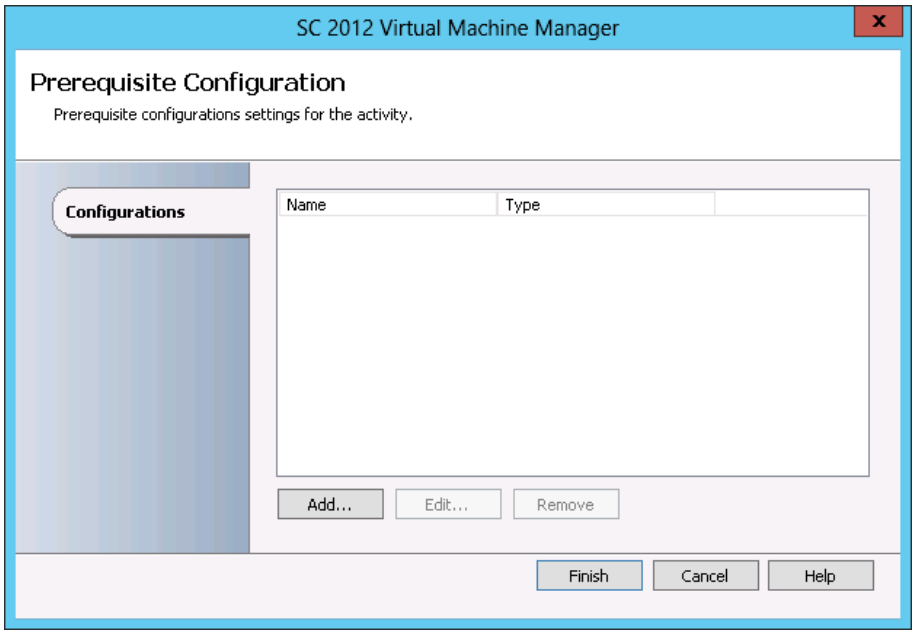
4. Complete steps **1-3** on the server **VMM**
5. Jeff is now ready to connect the Integration Pack to the **Virtual Machine Manager** server. Open **System Center 2012 Orchestrator Runbook Designer**



6. On the menu bar, select **Options** and select **SC 2012 Virtual Machine Manager**

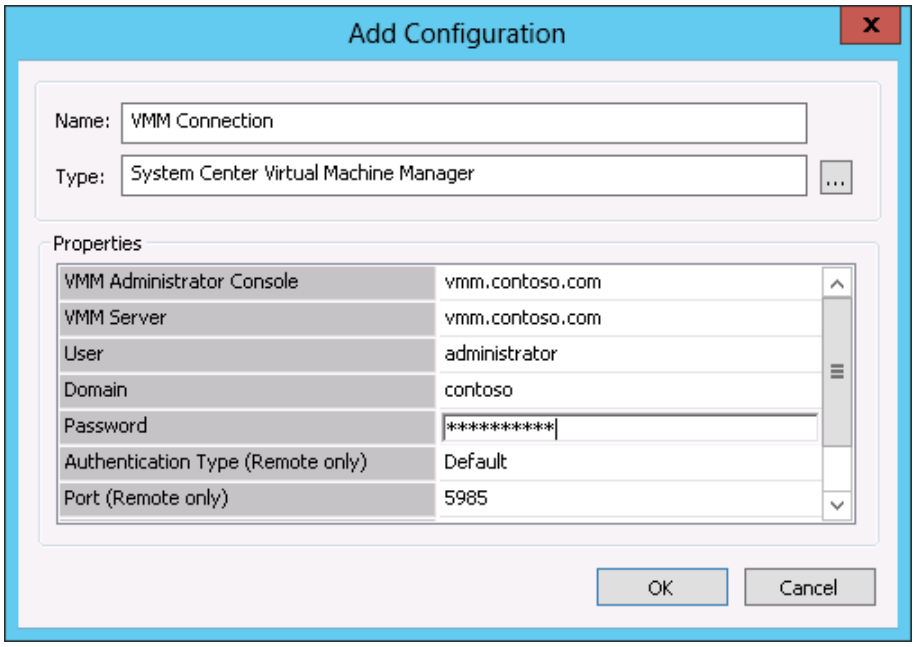


7. On the **Prerequisite Configuration** page, select **Add...**

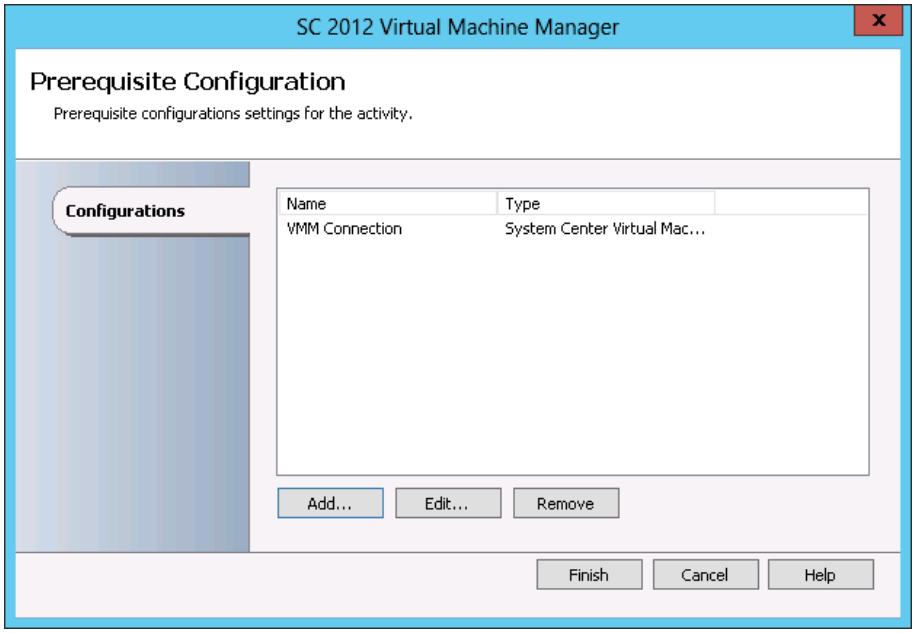


8. On the **Add Configuration** page, type in the **Name** (VMM Connection) and click on the **...** button to add the **Type** (System Center Virtual Machine Manager). Then fill in the following **properties** and then select **OK**

    - **VMM Administrator Console**: vmm.contoso.com
    - **VMM Server**: vmm.contoso.com
    - **User**: administrator
    - **Domain**: contoso
    - **Password**: **********
    - **Authentication Type (Remote only):** Default
    - **Port (Remote only)**: 5985
    - **Use SSL (Remote only)**: False
    - **Cache Session Timeout (Min.)**: 10



9. Select **Finish**



Jeff has now configured the Integration Pack to connect to **Virtual Machine Manager** in his environment.

Service Manager and the Connectors

Jeff has created the connections in **System Center 2012 Orchestrator** using the **Administrator** account. Now he needs to configure the **Connectors** in **System Center 2012 Service Manager**. Connectors in **Service Manager** are used to build the configuration management database, CMDB. The CMDB is created and maintained by importing configuration items (CI's) from Active Directory Domain Services, Operations Manager, Virtual Machine manager, and Configuration Manage. There is also a connector that allows you to automatically import alerts from Operations Manager and you can import data manually as well from a **CSV** file. By having an accurate and easily maintained CMDB, Jeff can accurately create the organizational processes and automated responses required to maintain his private cloud SLAs.

For creating the connectors in Service Manager, Jeff logs on to the Service Manager server as a Domain Administrator

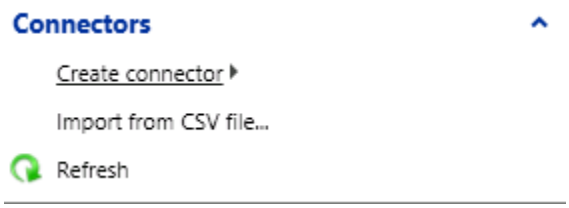
The Active Directory Connector

The Active Directory Connector will import users, groups, printers and computers as configuration items into the Service Manager CMDB database. Those objects then can be used as objects in incidents, requests and so on.

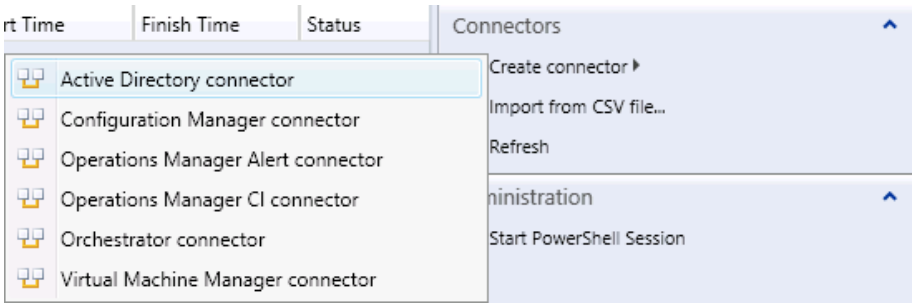
Jeff is going to configure this connector.

1. In the **Service Manager console**, under **Administration**, click on **Connectors**

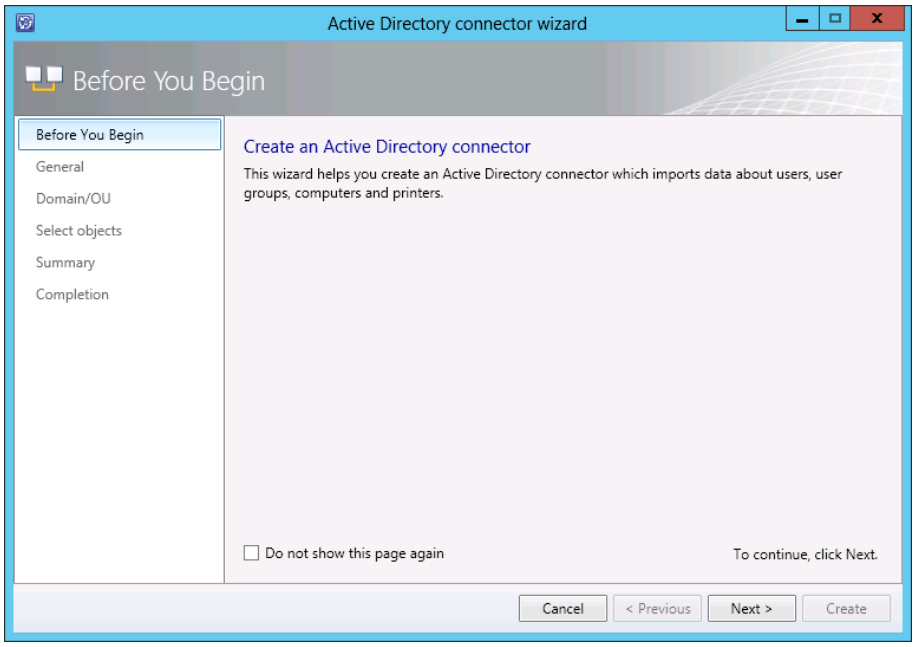
2. In the **Tasks** pane, on the right, select the **Create connector** button



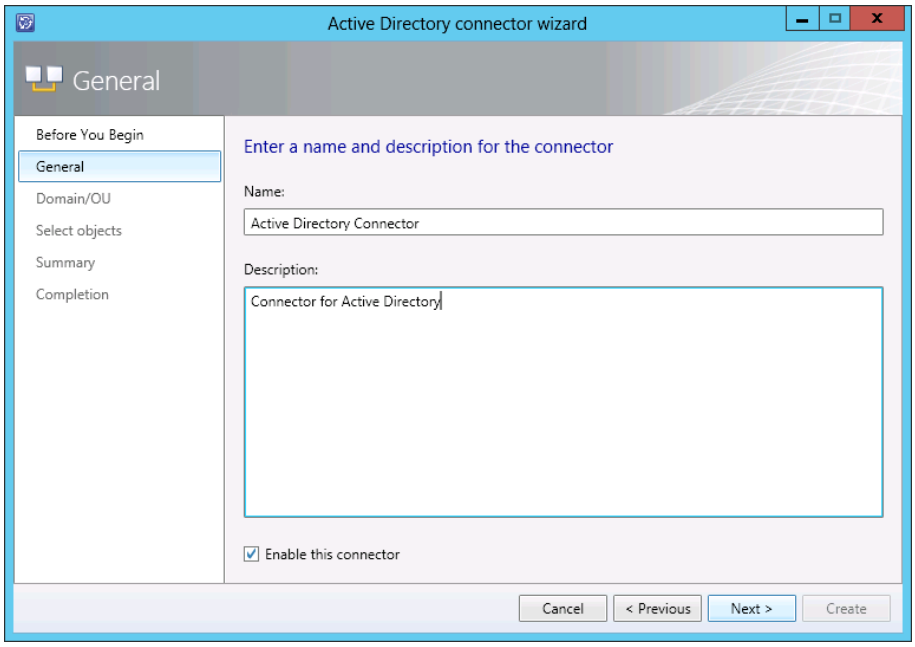
3. Choose **Active Directory connector**



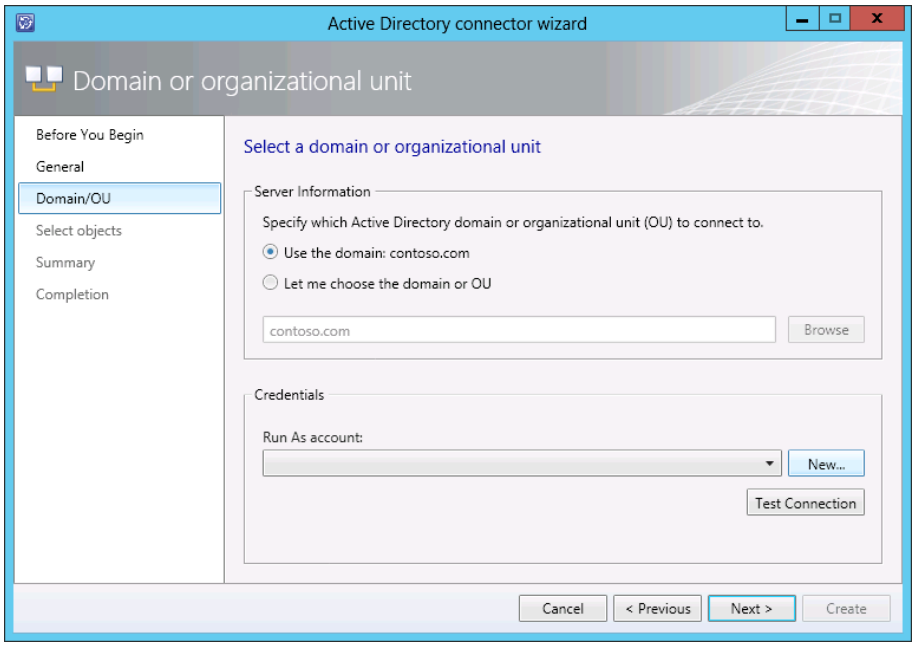
4. On the **Before you begin** page, select **Next**



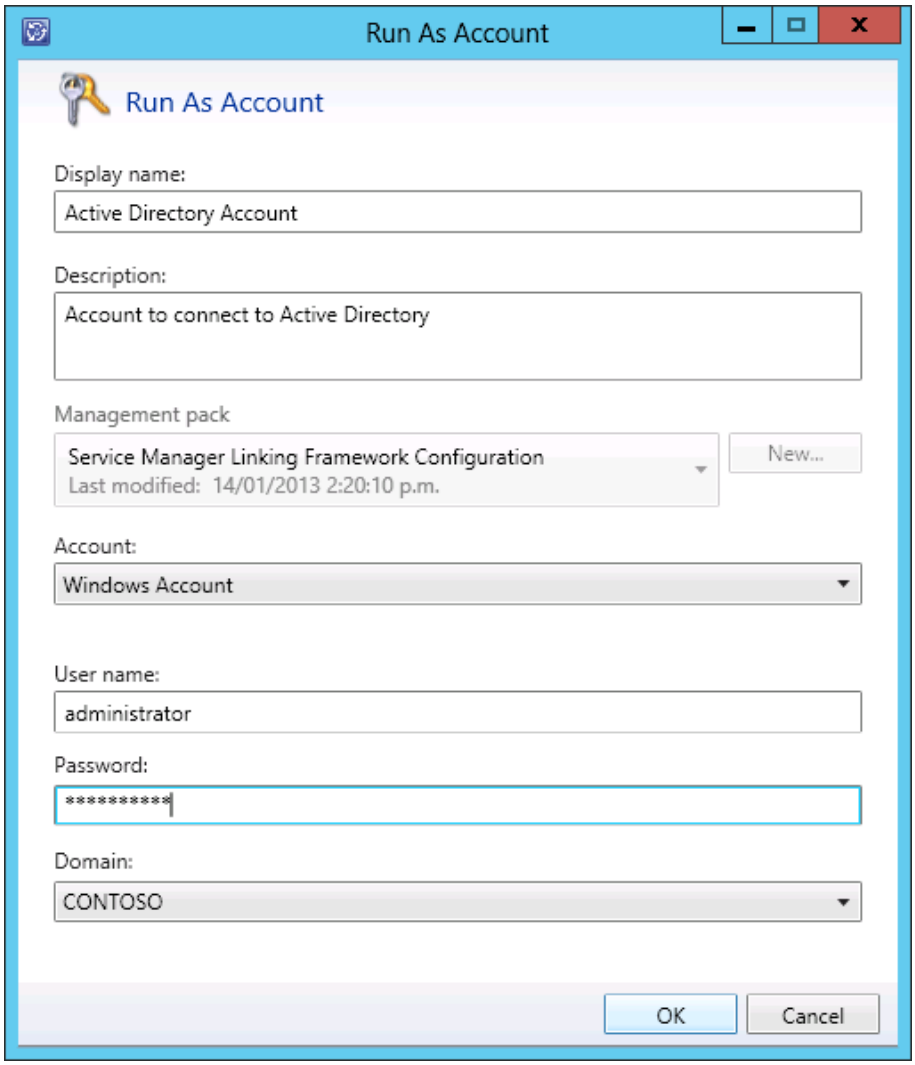
5. On the **General** page, type in the following parameters and select **Next**
    - **Name:** Active Directory Connector
    - **Description:** Connector for Active Directory
    - **Enable this connector:** checked



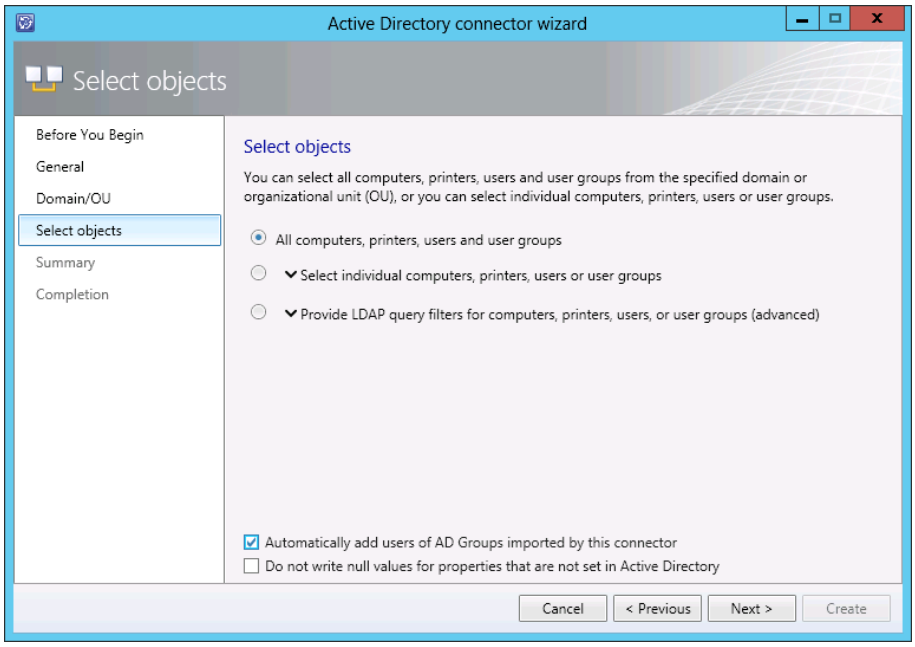
6. On the **Domain or organizational unit** page, select the domain then select **New...** to create a specific account for this connection or choose an already created account. In this case, we are going to create a new one.



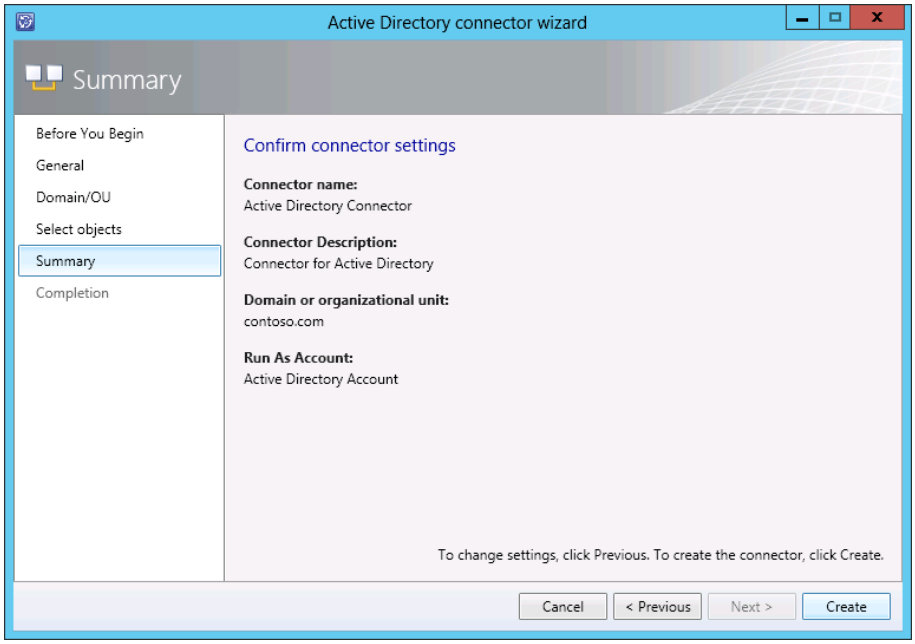
7. On the Run As Account page, fill in the following parameters; after that, when back on the **Domain/OU** page **test the connection** and select **Next**
    - **Name:** Active Directory Account
    - **Description:** Account to connect to Active Directory
    - **Account:** Windows Account
    - **User name:** administrator
    - **Password:** ******
    - **Domain:** contoso



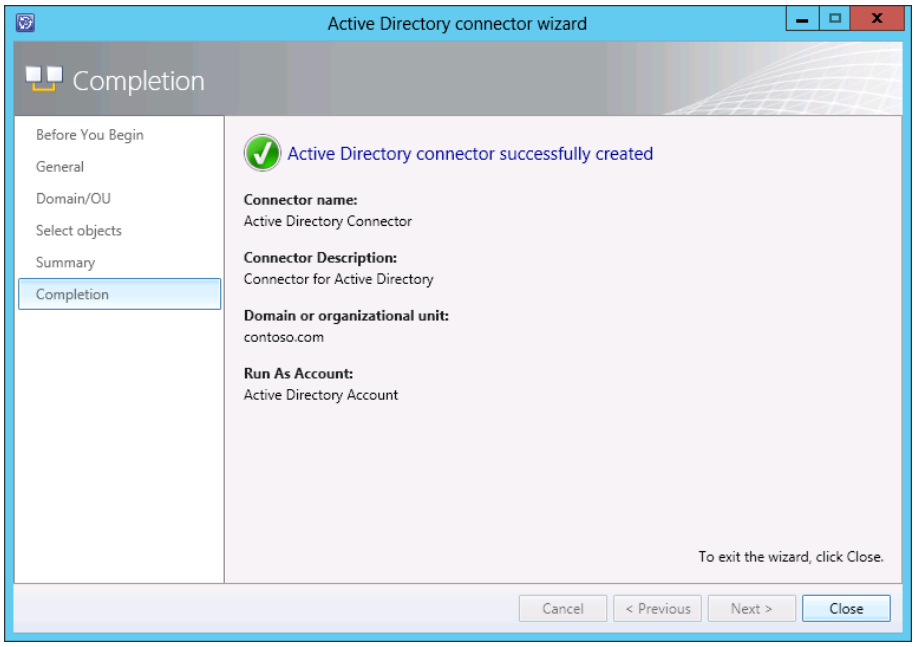
8. On the **Select objects** screen, select **All computers, printers, users and user groups**, tick **Automatically add users of AD Groups imported by this connector**, and select **Next**



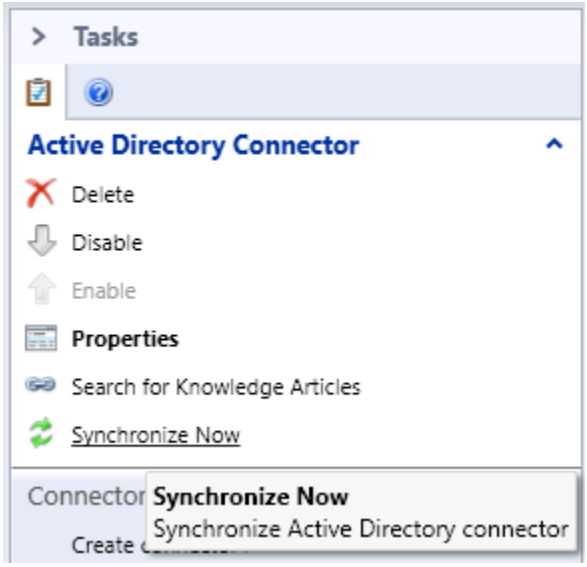
9. On the **Confirm connector settings** screen, review your settings and select **Create** to create the connector



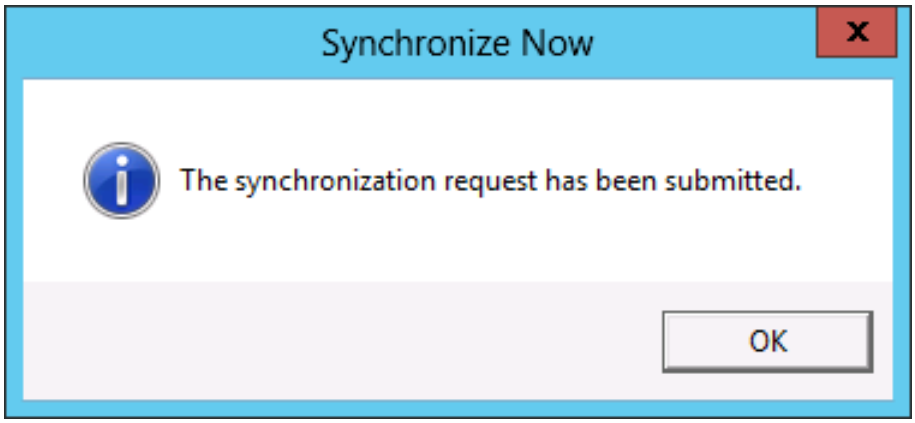
10. On the **Completion** screen, select **Close**



11. From the **Tasks** column, click on **Synchronize now**



12. Click on **OK** to confirm the synchronization request

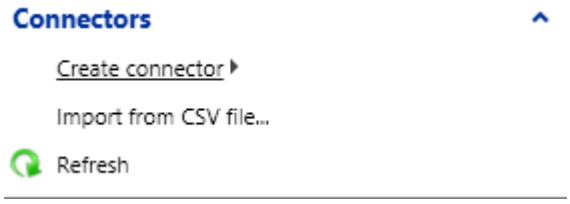


The Orchestrator Connector

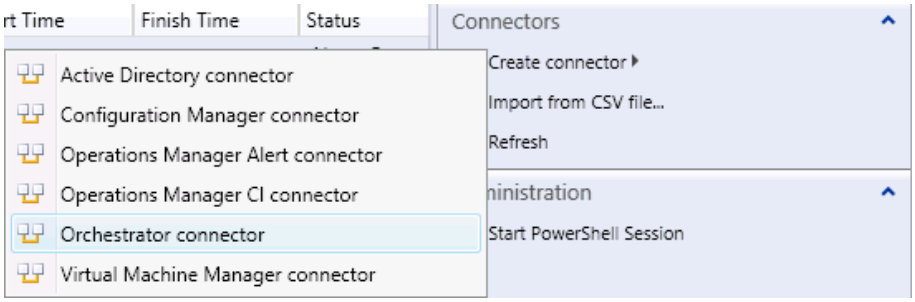
The System Center Orchestrator Connector provides the capability to synchronously invoke runbooks from within Service Manager through the use of workflows.

1. In the **Service Manager** console, under **Administration**, click on **Connectors**

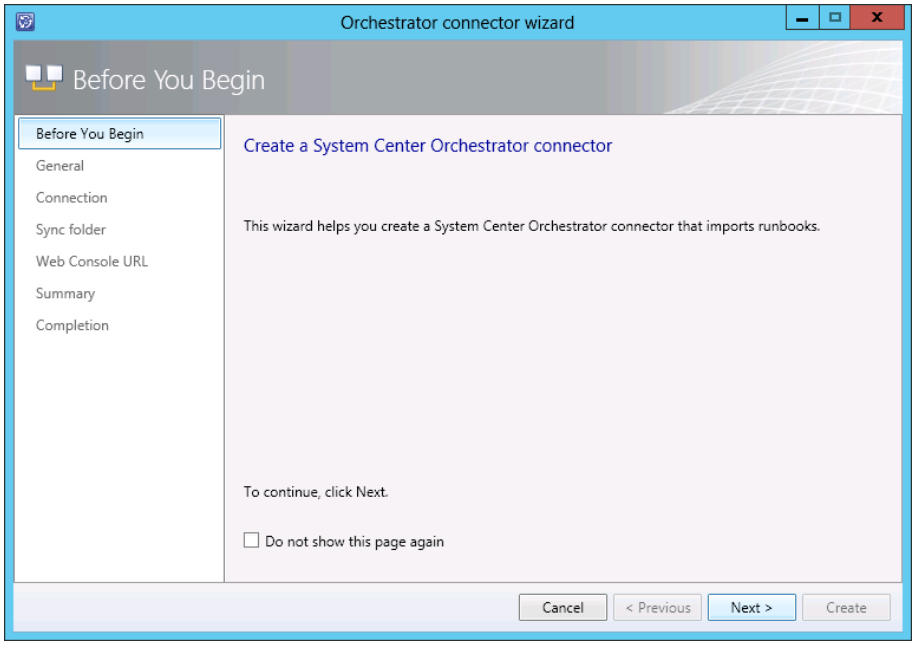
2. In the **Tasks** pane, select the **Create connector** button



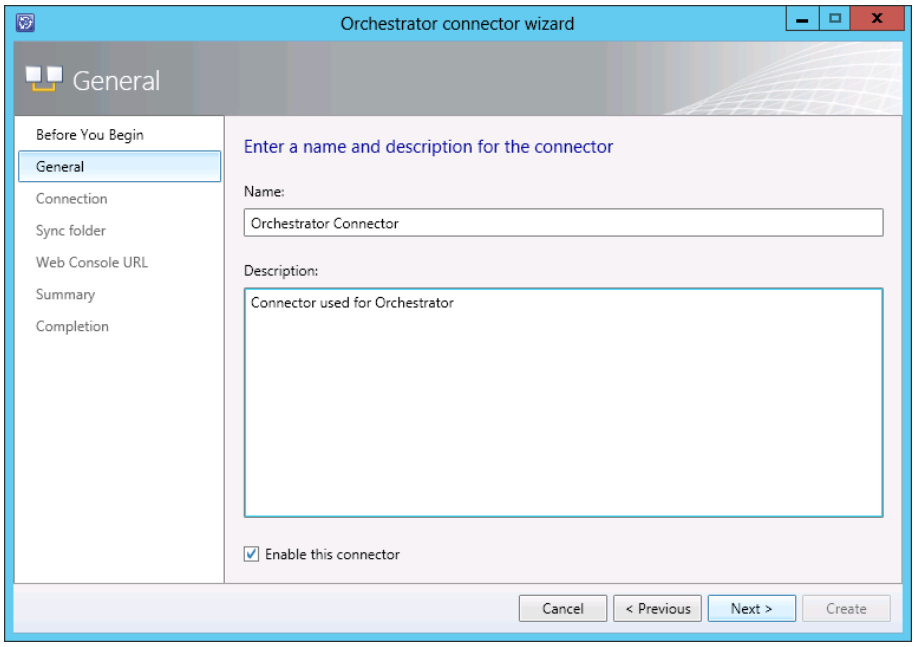
3. Select the **Orchestrator connector** button



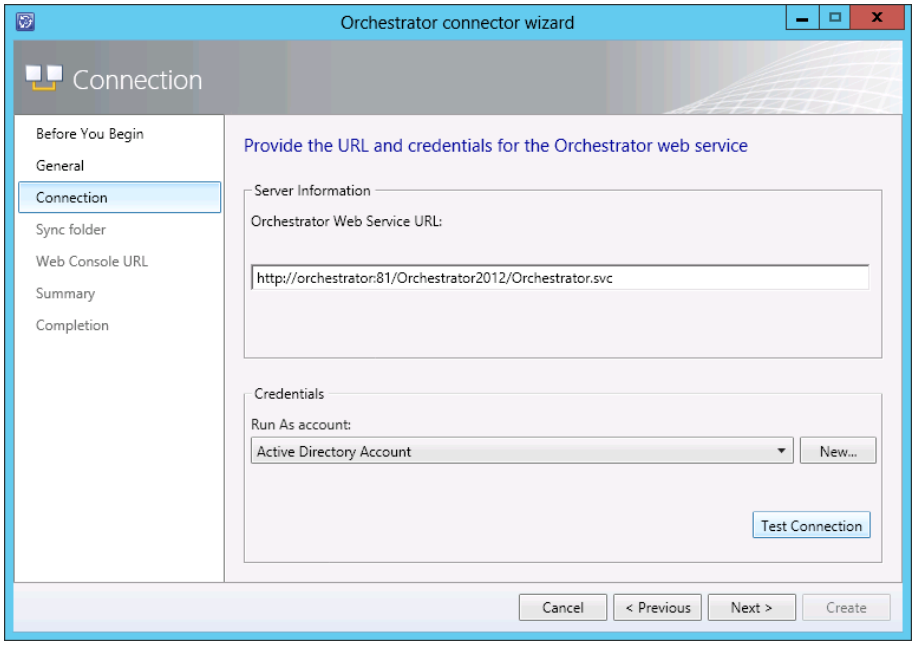
4. On the **Before you Begin** page, select **Next**



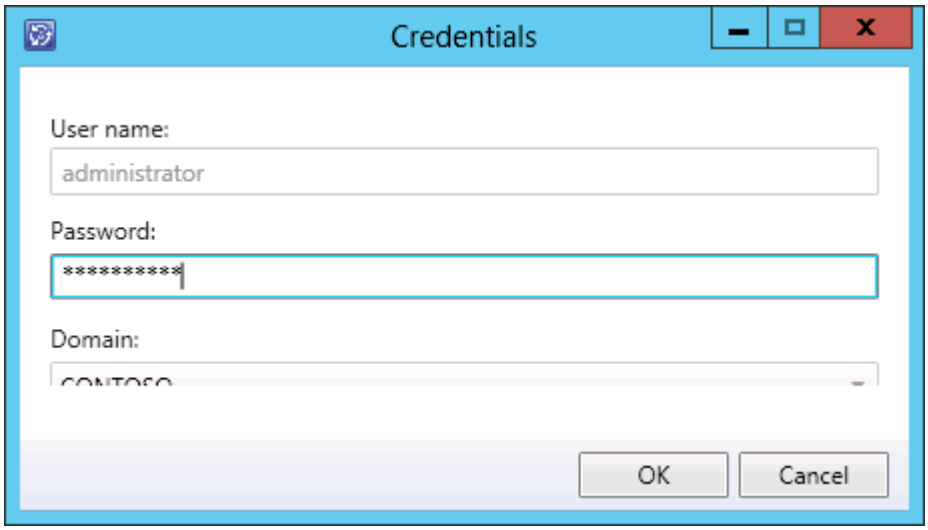
5. In the **General** screen, enter the following parameters and select **Next**
    - **Name:** Orchestrator Connector
    - **Description:** Connector used for Orchestrator
    - **Enable the connector:** checked



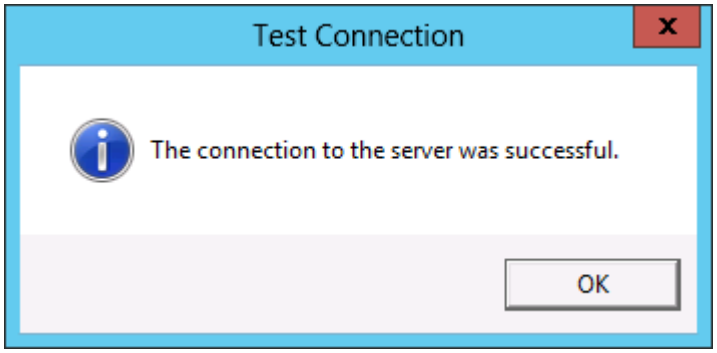
6. On the **Connection** page, fill in the URL for the **Orchestrator Web Service URL**. This is http://servername:port/Orchestrator2012/Orchestrator.svc so in our environment this is http://orchestrator:81/Orchestrator2012/Orchestrator.svc For the **Run As Account** select **Active Directory Account**



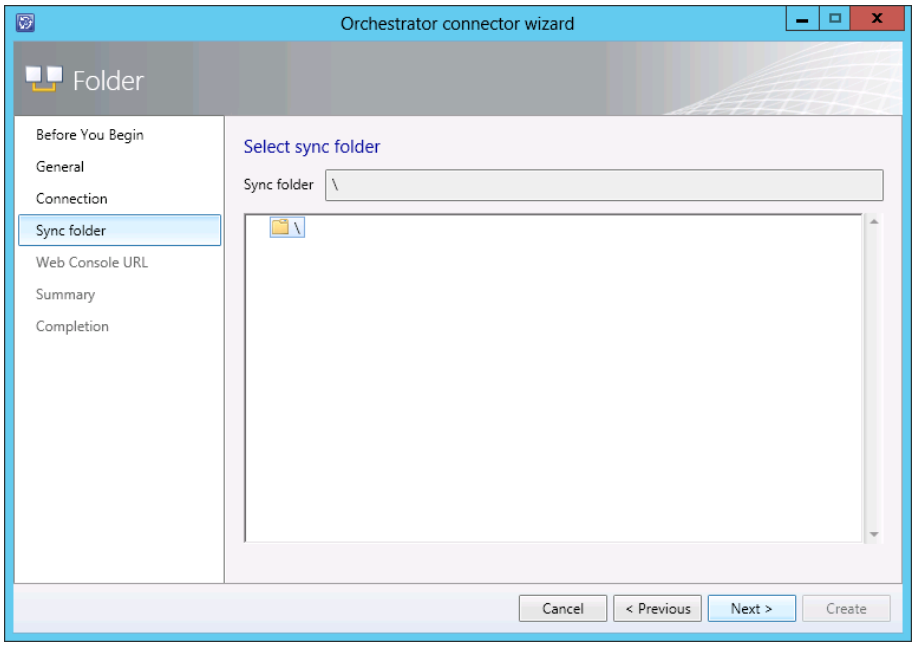
7. For the credentials use the **contoso\administrator** account, then click on **OK**



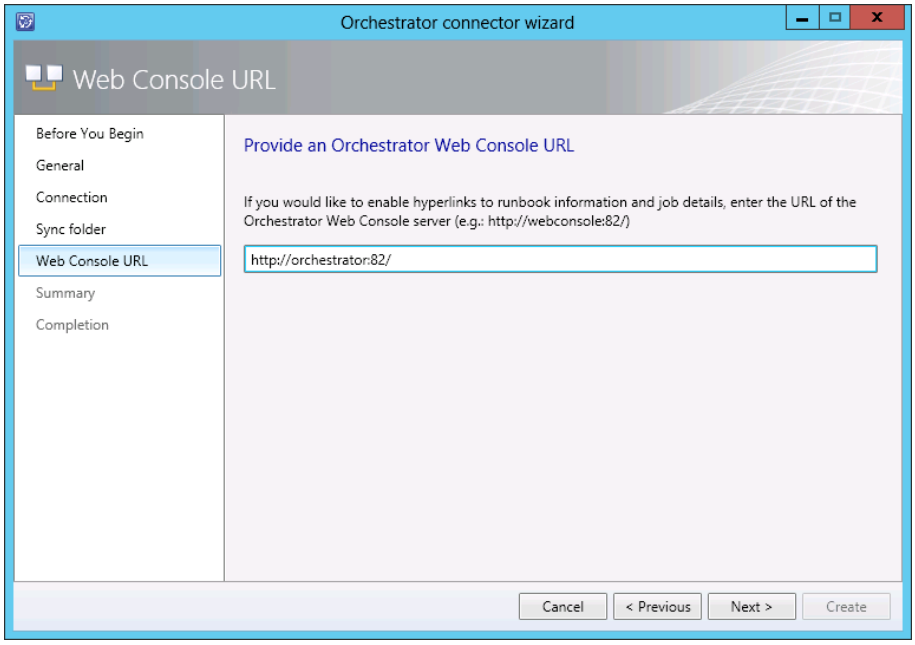
8. Click on **Test Connection** to confirm the credentials, then click **OK**



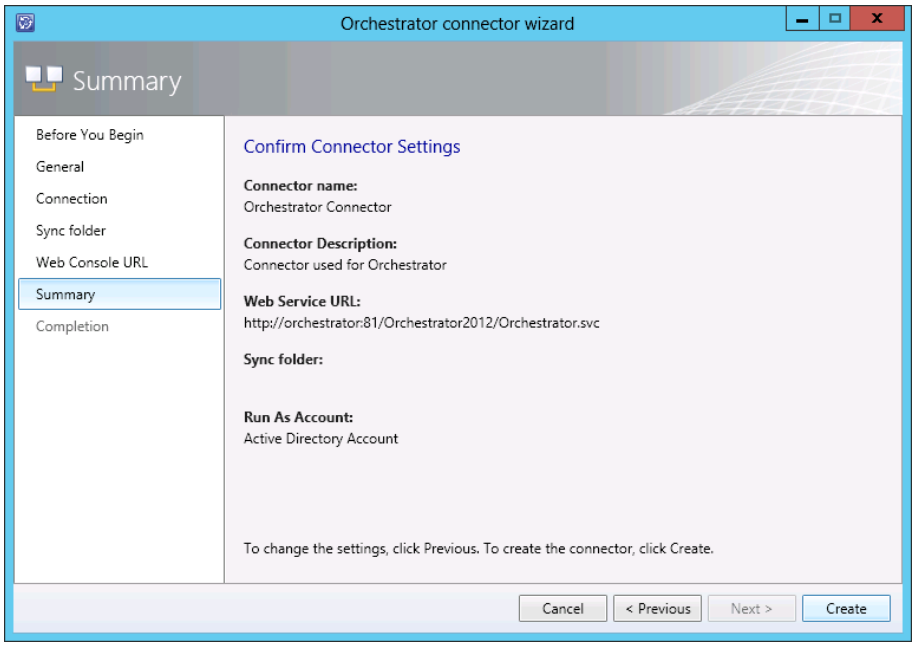
9. On the **Sync folder** page, you can select which folders in **Orchestrator** that you want to import. If you choose the **** folder as Jeff is going to do, then you will import all the runbooks from Orchestrator. Select **Next**



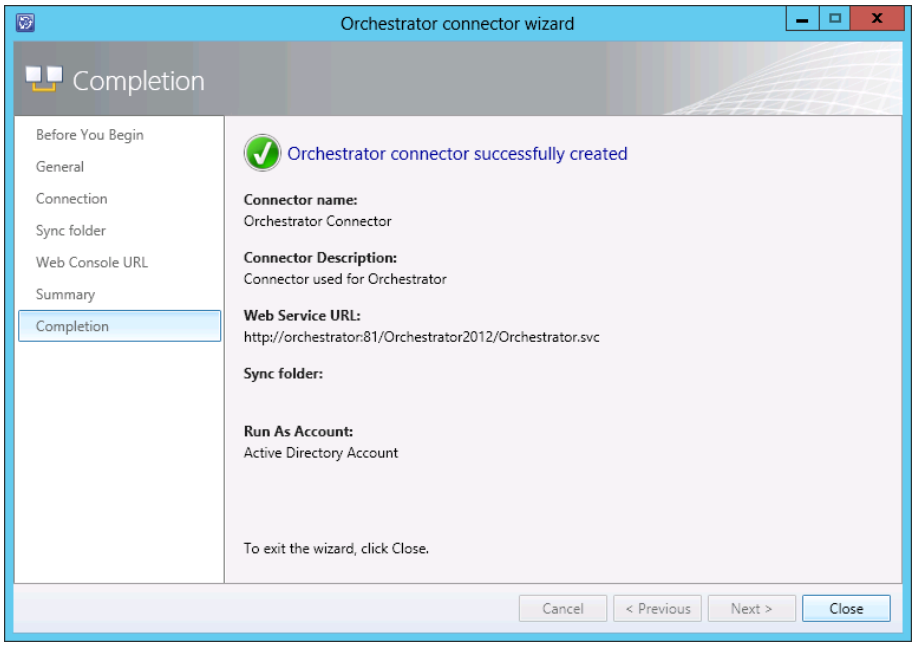
10. On the **Web Console URL** page, enter the URL for the **Web Console** of Orchestrator. This is http://orchestrator:82/(or http://orchestratormanagementserver:port). Select **Next**



11. On the **Summary** page, review your settings and select **Create**



12. On the **Completion** page, select **Close**



Service Manager and the Data Warehouse

After you have deployed the service manager management servers and data warehouse management servers, you need to set up reporting. To do this, you run the Data Warehouse Registration Wizard. This wizard registers the service manager management group with the data warehouse management group. It also deploys management packs from the service manager management server to the data warehouse management server.

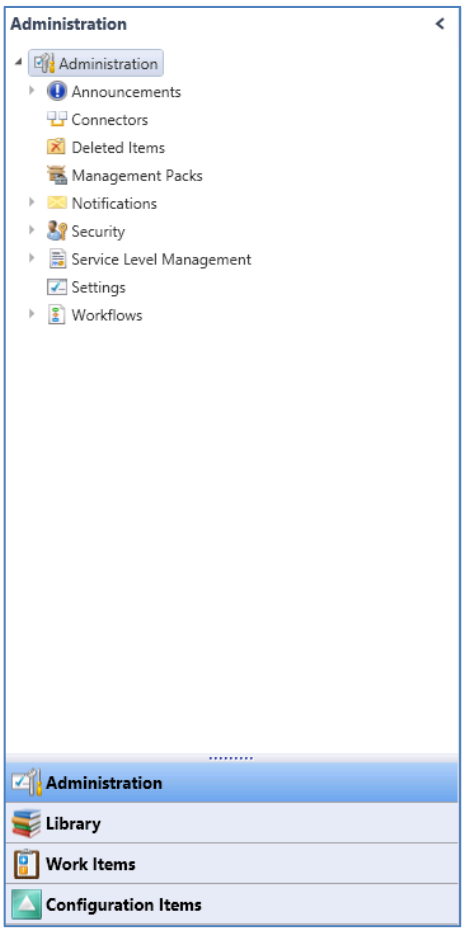
Important Note

This process can take several hours to complete. It is advised that you don't turn off any services during this period. You can work with Service Manager during this time without any problems.

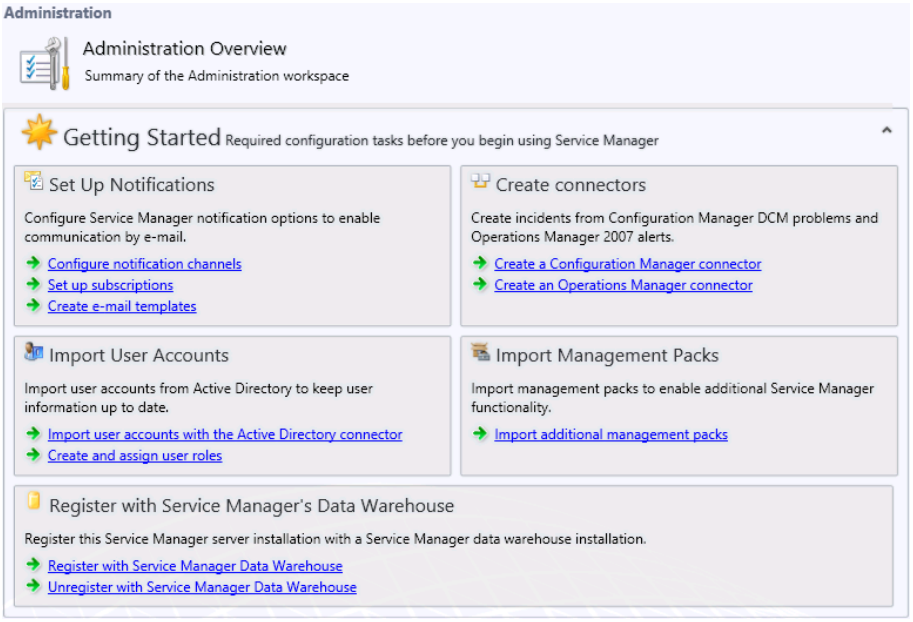
Registering with the Data Warehouse

Jeff is not going to evaluate the reporting now but he sure will want to look into this in a later phase of the evaluation. For now, he is going to register the Service Manager installation with the **Data Warehouse**

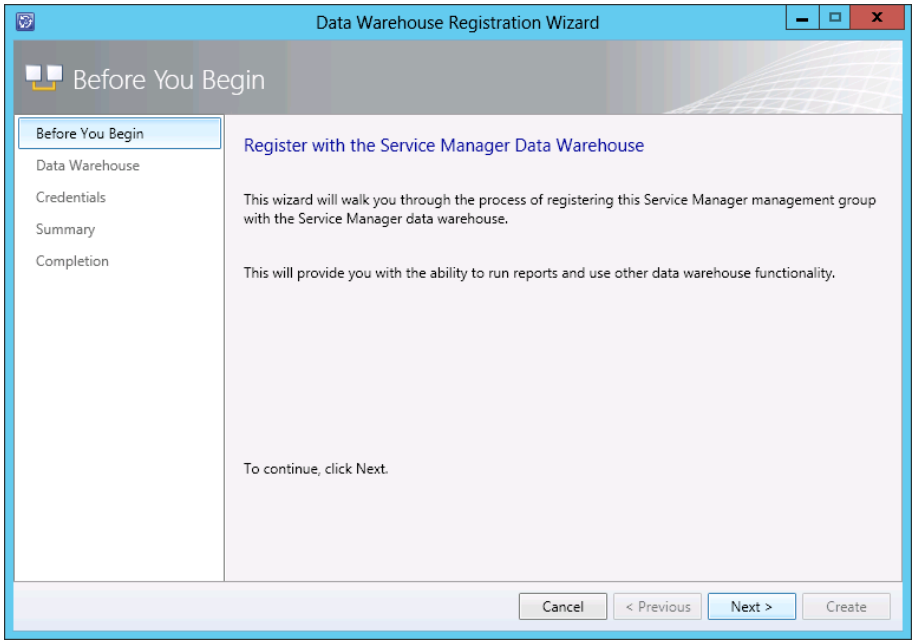
1. In the **System Center 2012 Service Manager** console, go to **Administration**



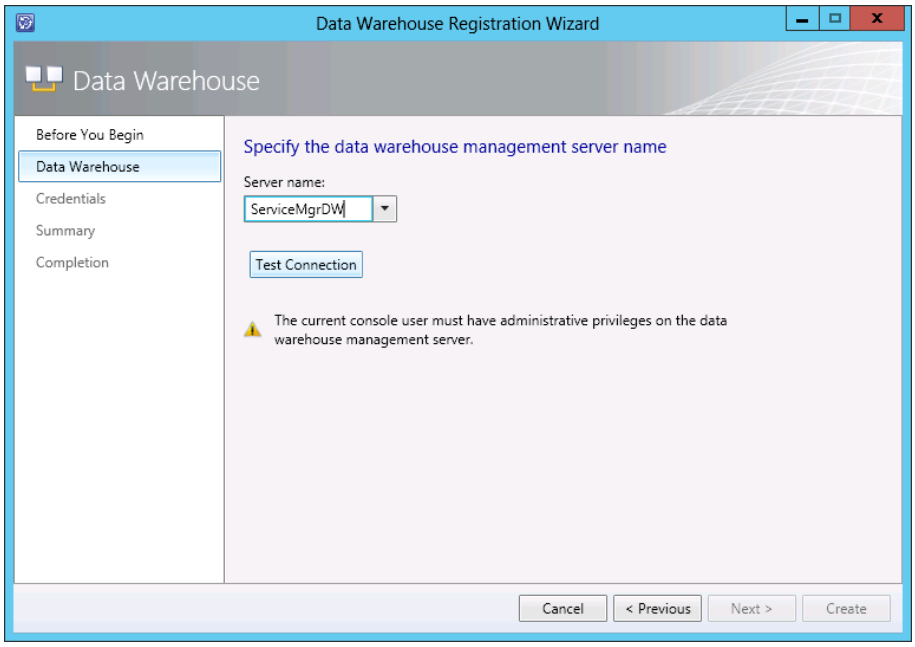
2. On the **Administration Overview** page, click on **Register with Service Manager Data Warehouse**



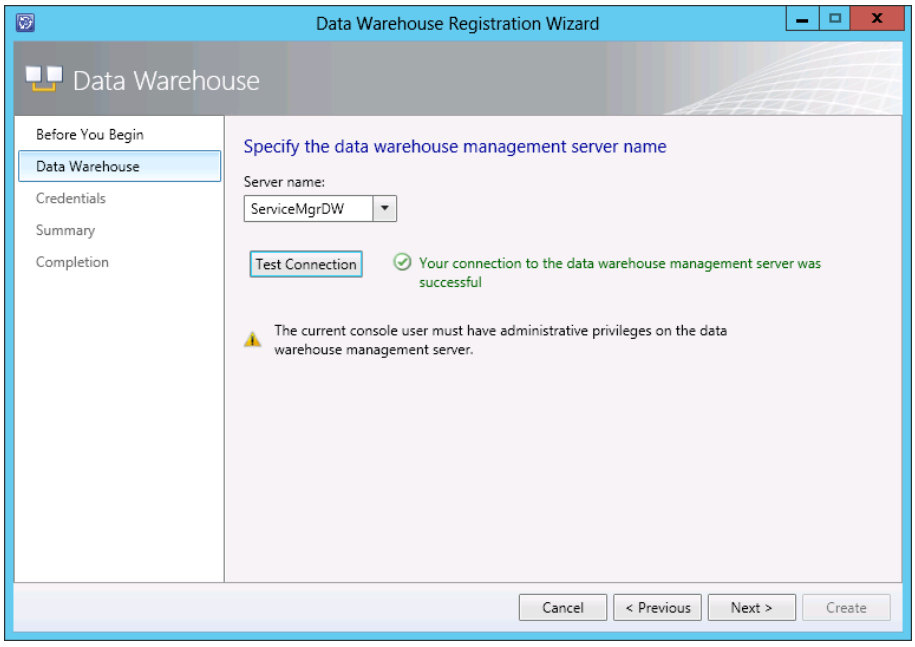
3. On the **Before You Begin** page, select **Next**



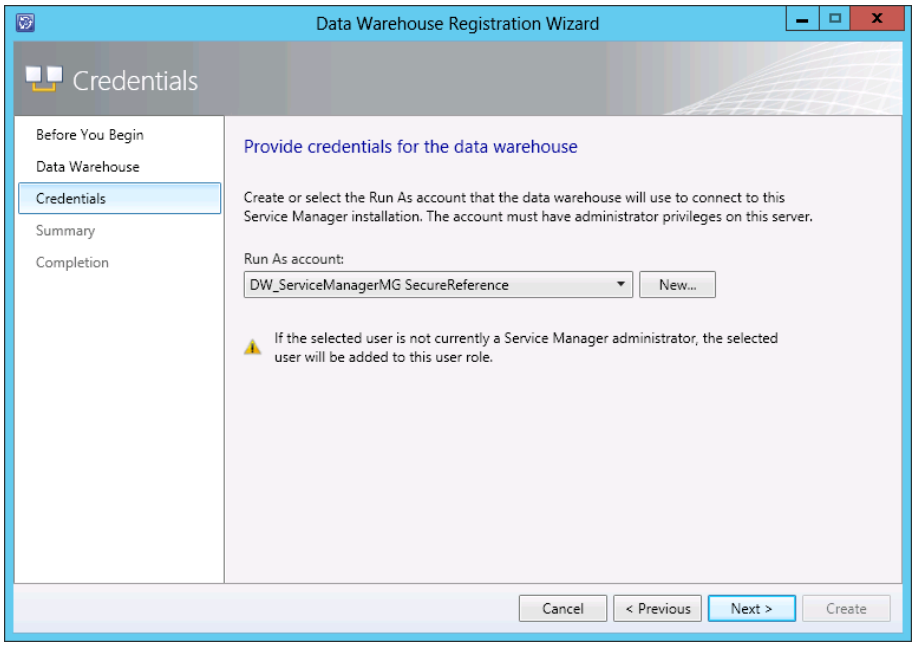
4. On the **Data Warehouse** page, type in the **Server name ServiceMgrDW** and select the **Test Connection** button.



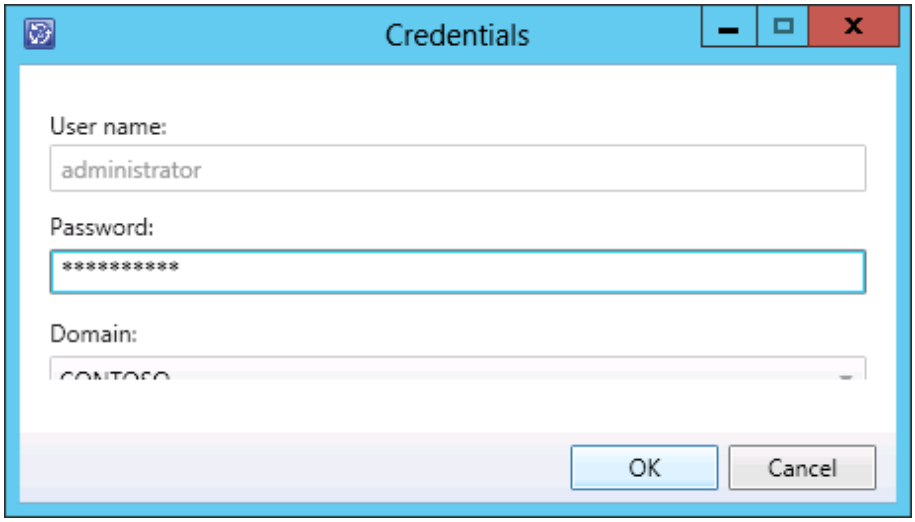
5. If the connection test succeeds, select **Next**



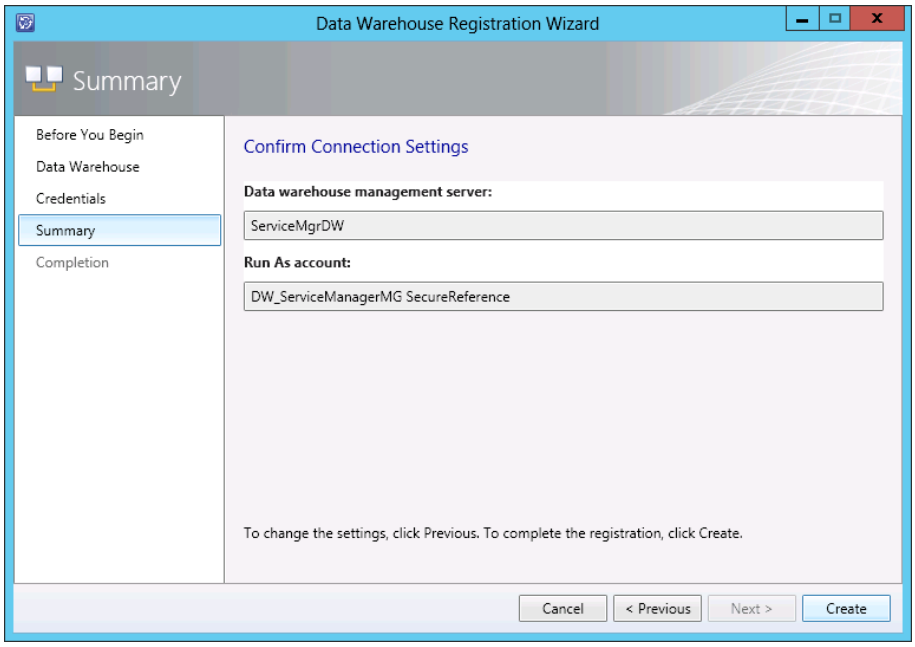
6. On the **Credentials** page, select the **DW_ServiceManagerMG SecureReference Run As Account** and select **Next**



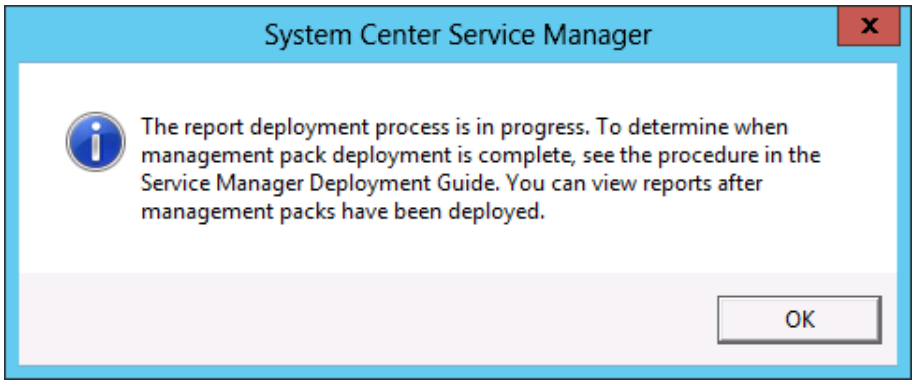
7. The system will request you to enter the **Password** for these credentials. Enter the password and select **OK**



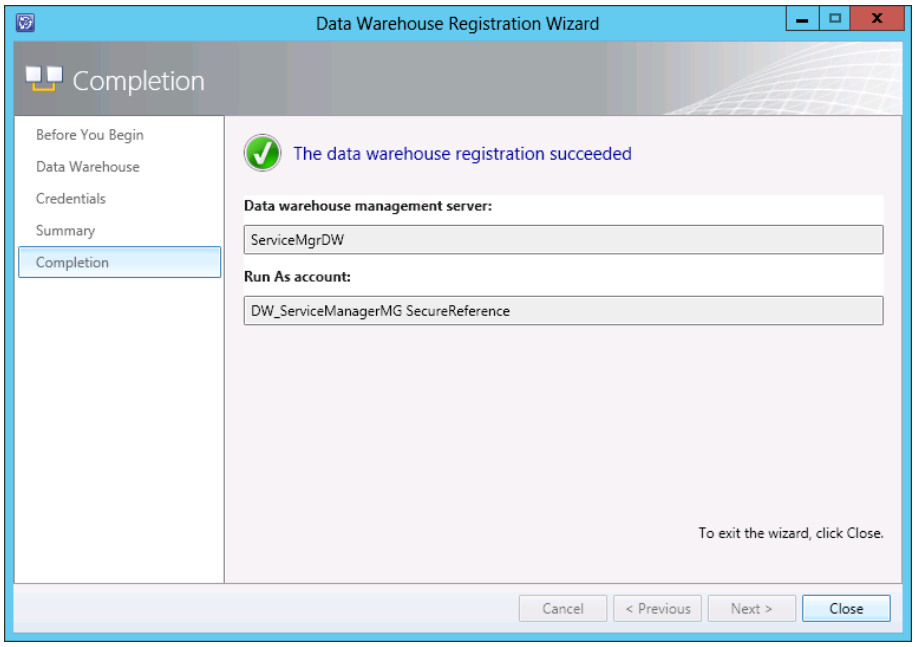
8. On the **Summary** page, review your selections and select **Create**



9. When the **Report Deployment process** popup appears, select **OK**



10. On the **Completion** page, select **Close**

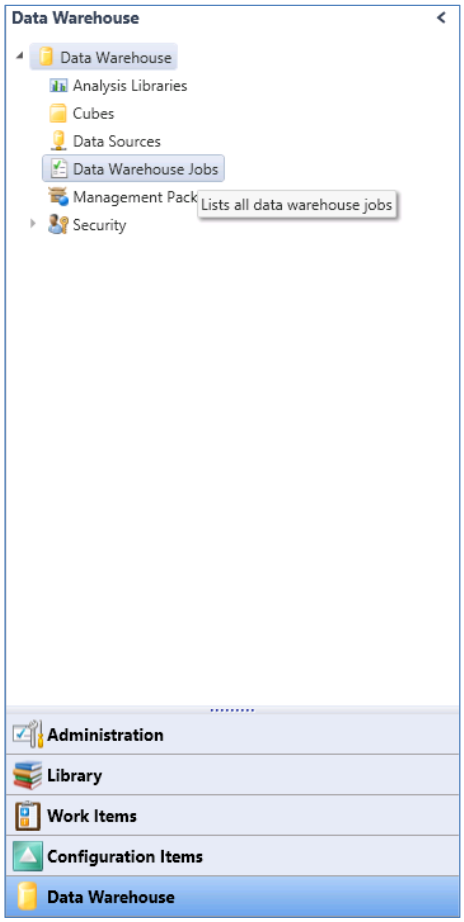


How to determine when the Data Warehouse registration is complete

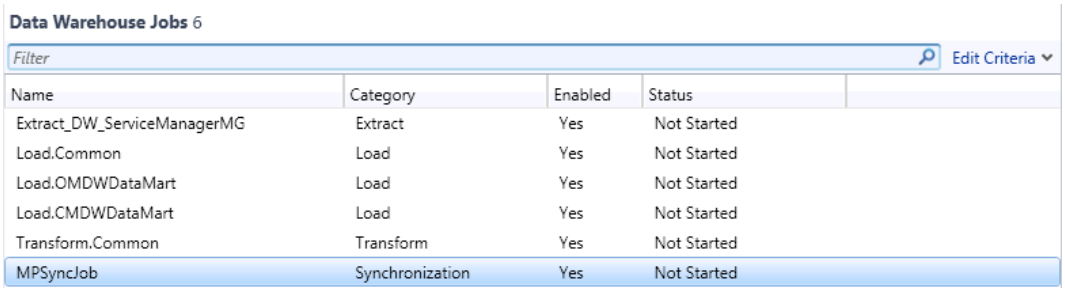
Jeff knows that the synchronization process can take up to a few hours. In the meantime, he will continue with his evaluation. During this time, he also would like to check on the status of the registration process.

Jeff can see if the initial synchronization process is finished by following this procedure:

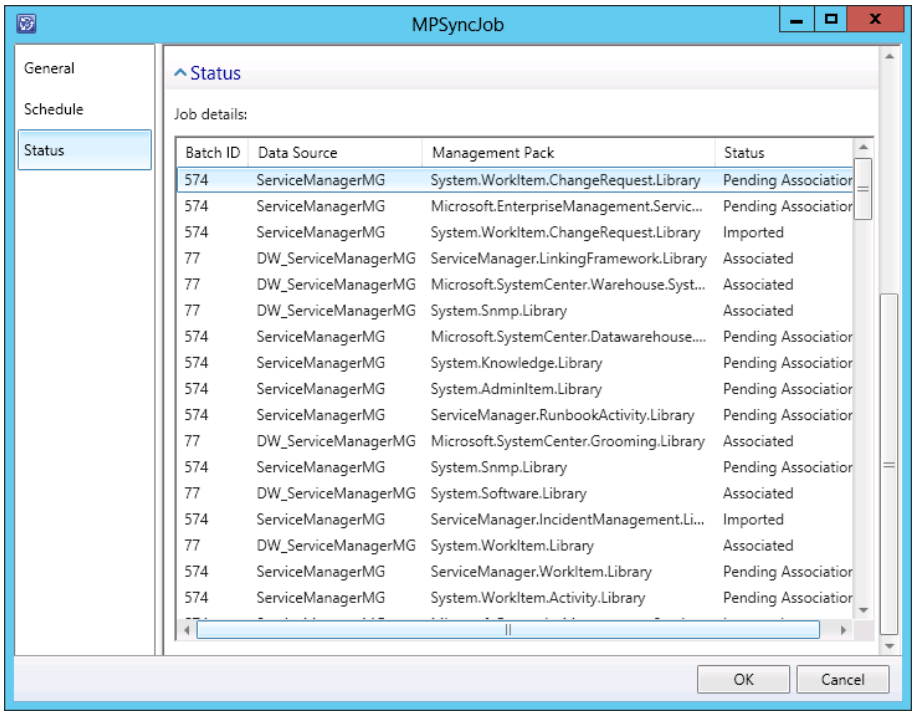
1. In the **System Center 2012 Service Manager** console, go to **Data Warehouse** > **Data Warehouse Jobs**



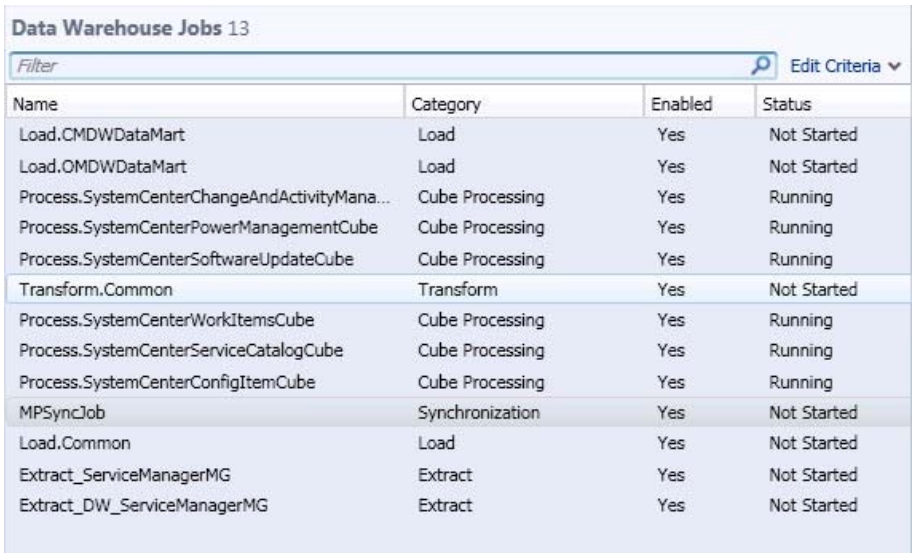
2. In the **Data Warehouse Jobs** pane, double click **MPSyncJob**



3. On the **MPSyncJob details** page, go to **Status** and review the **Job details** list. The management pack deployment process is complete when the status for all of the management packs is **Associated** or **Imported**



4. After the management packs have been deployed (as determined in step 3), make sure that the following five data warehouse jobs appear in the **Data Warehouse Jobs** pane:
    - Extract_<Service Manager management group name>
    - Extract_<data warehouse management group name>
    - Load.Common
    - Transform.Common
    - MPSyncJob



Data Warehouse Jobs 13

Filter — Edit Criteria

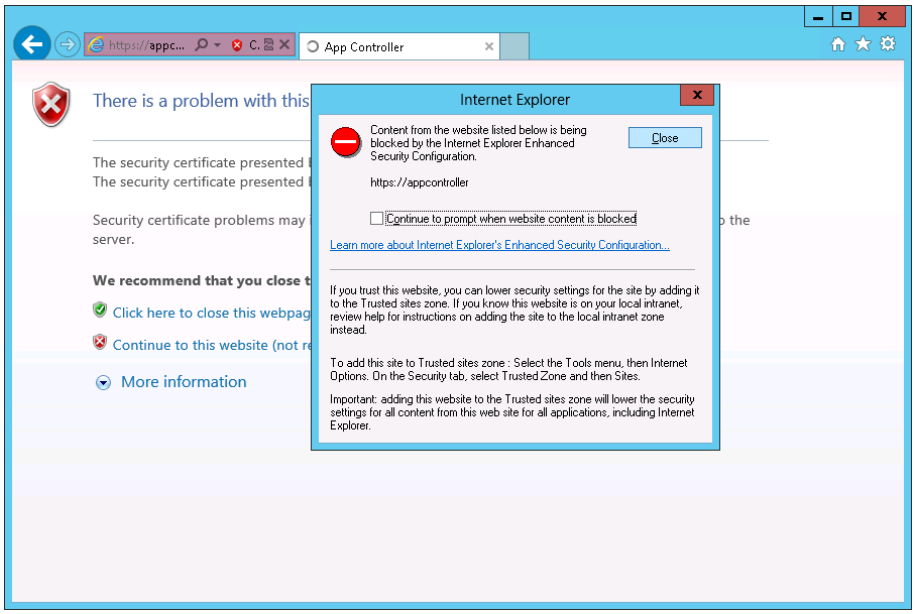
Name	Category	Enabled	Status
Load.CMDWDataMart	Load	Yes	Not Started
Load.OMDWDataMart	Load	Yes	Not Started
Process.SystemCenterChangeAndActivityMana...	Cube Processing	Yes	Running
Process.SystemCenterPowerManagementCube	Cube Processing	Yes	Running
Process.SystemCenterSoftwareUpdateCube	Cube Processing	Yes	Running
Transform.Common	Transform	Yes	Not Started
Process.SystemCenterWorkItemsCube	Cube Processing	Yes	Running
Process.SystemCenterServiceCatalogCube	Cube Processing	Yes	Running
Process.SystemCenterConfigItemCube	Cube Processing	Yes	Running
MPSyncJob	Synchronization	Yes	Not Started
Load.Common	Load	Yes	Not Started
Extract_ServiceManagerMG	Extract	Yes	Not Started
Extract_DW_ServiceManagerMG	Extract	Yes	Not Started

Adding Virtual Machine manager to App Controller

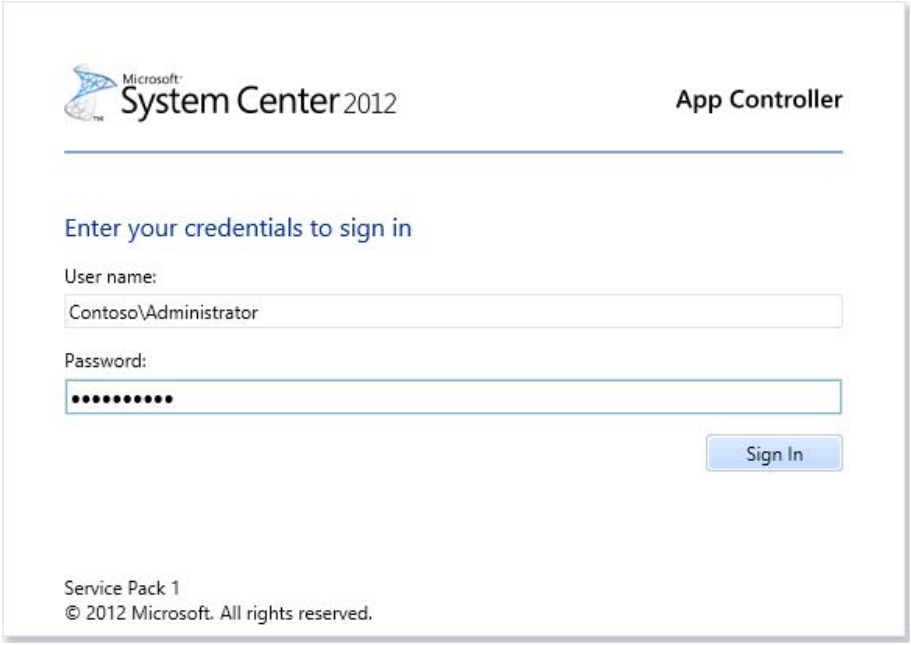
Important Note

Viewing the App Controller portal with Internet Explorer requires the **Silverlight plugin**. In this example, Jeff is running it on the App Controller server itself but you could use another Server / Workstation to connect to the portal without needing to install Silverlight on the App Controller server.

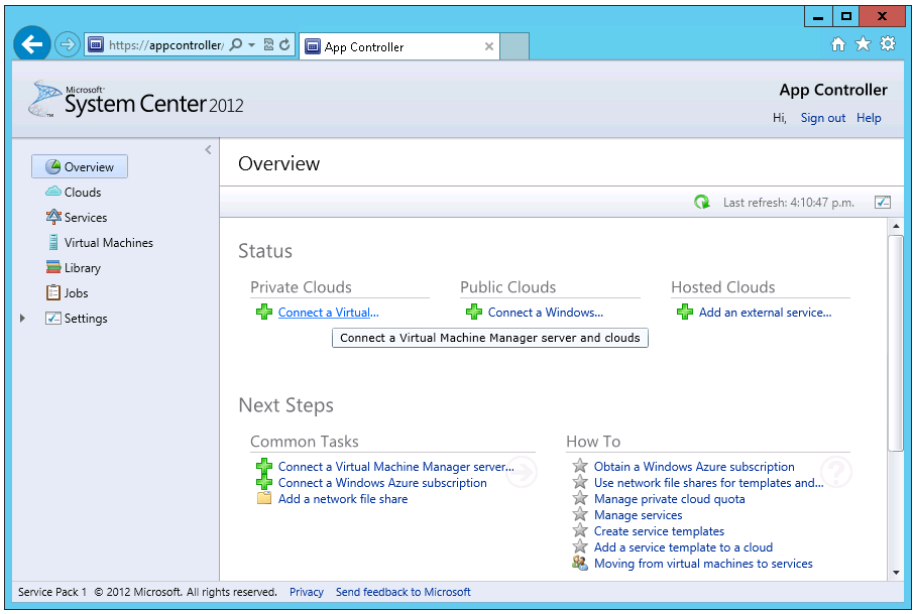
1. Open the App Controller portal by browsing to https://appcontroller/. When prompted with the certificate warning, click **Continue to this website** and add the site to the trusted site list.



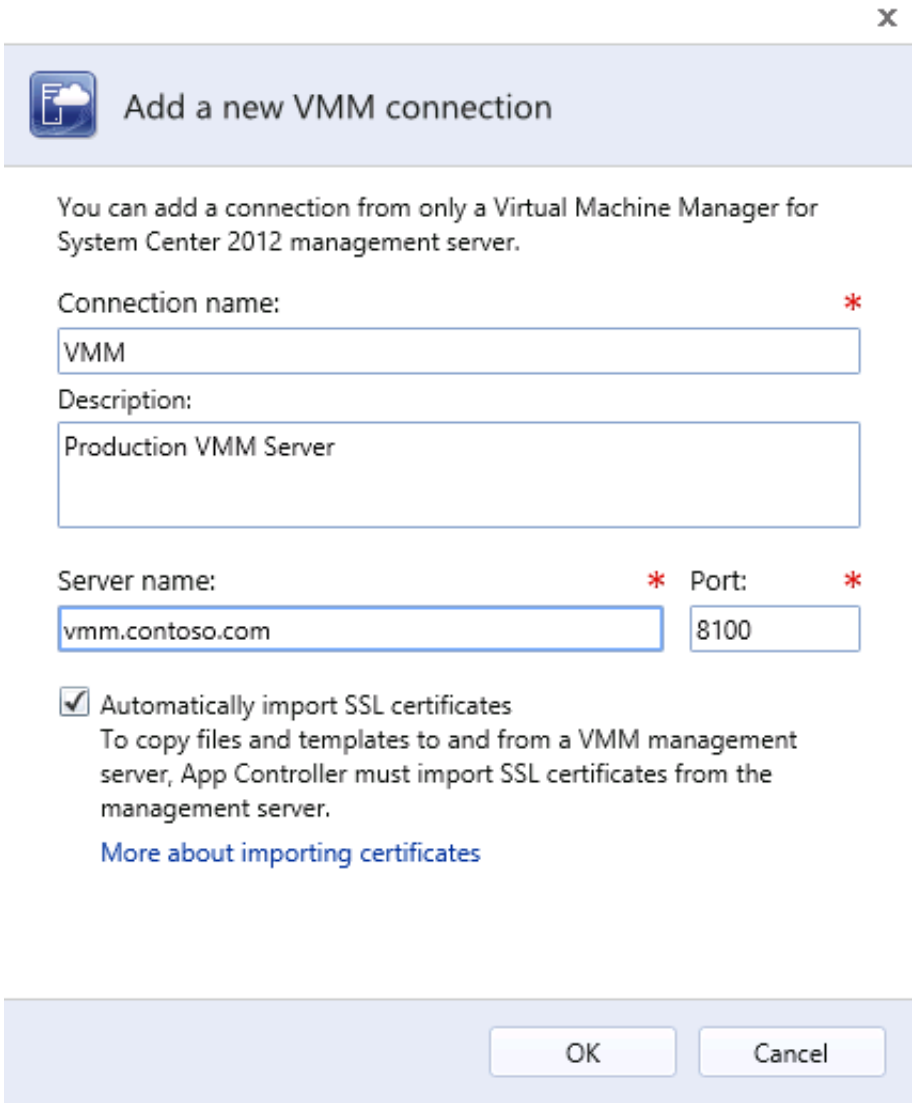
2. At the login screen enter the user name **contoso\Administrator** and relevant password and click Sign In



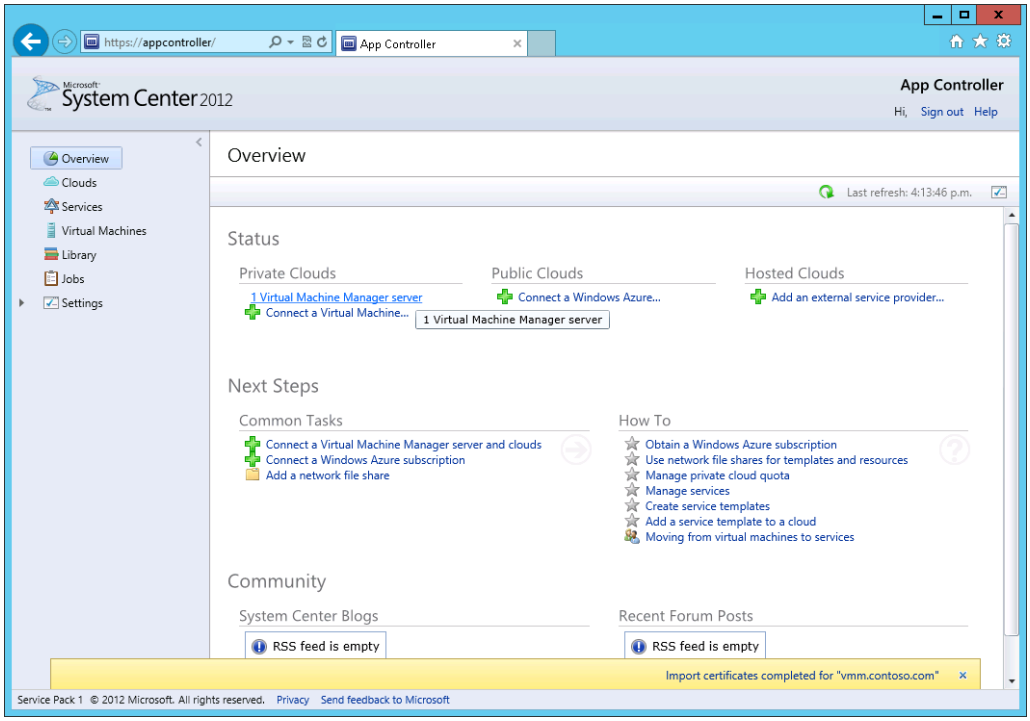
3. On the Overview page, under the Private Clouds heading click **Connect a Virtual Machine Manager server and clouds**



4. On the **Add a new VMM connection** window enter the following information and click **OK**
    - Connection name: VMM
    - Description: Production VMM server
    - Server name: vmm.contoso.com
    - Port: 8100



To confirm the VMM server connection has been created, on the **Overview** page under the **Private Clouds** heading you should now see **1 Virtual Machine Manager server**.



Summary

In this chapter, **Jeff** configured some of the connectors between the System Center 2012 components. He configured the Integration Packs between Orchestrator and Virtual Machine Manager, and linked Operations Manager and Virtual Machine Manager through their connector. He then created the connector between Service Manager and Active Directory, then created this Service Manager Data Warehouse association. Jeff is now ready to start configuring his Self Service Portal.

Scenario: Build extreme automation

In this scenario we are going to explore the automation possibilities of the Microsoft Private Cloud. The primary component of System Center 2012 we will be focusing on will be System Center 2012 Orchestrator.

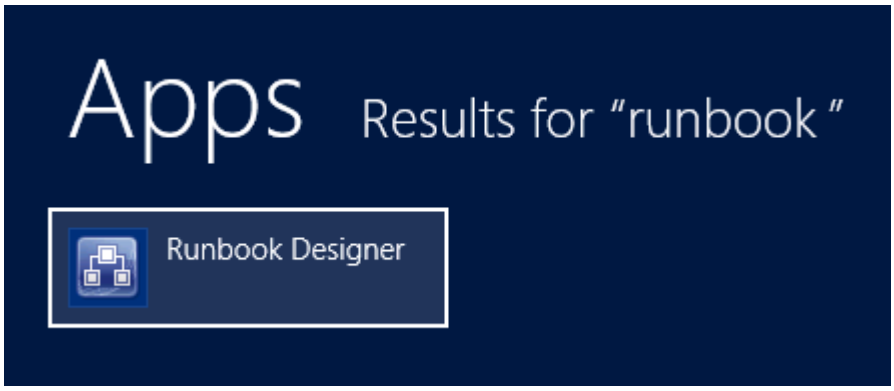
Jeff wants to build some automation into his environment. Initially, Jeff would like to explore some of the automation capabilities available, and may later decide to re-use his new runbooks in his production environment.

Creating the Create Private Cloud runbook

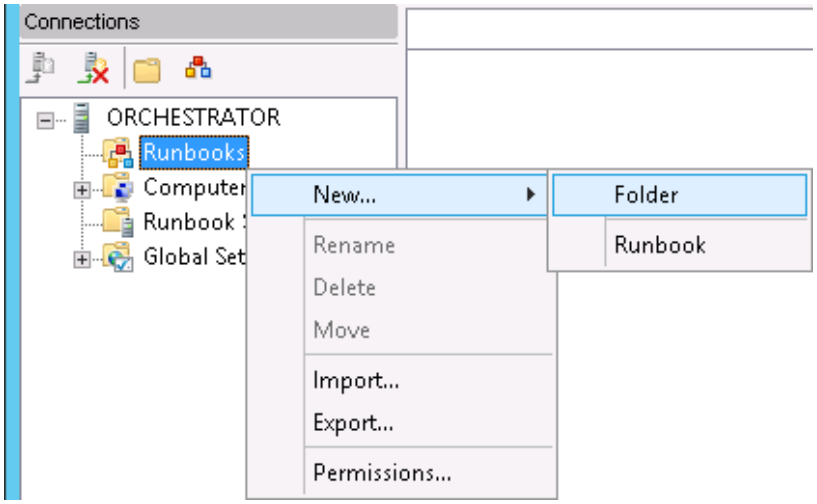
Jeff manages a large number of private clouds that are manually created by him through requests from Emily. Because automation is a key component of the private cloud, Jeff wonders if he can automate this simple task. If he succeeds, Emily's requests will be fulfilled faster, and Jeff can continue to focus on his existing work instead of repeating this simple task. Jeff decides to try to achieve this objective utilizing System Center 2012 Orchestrator.

1. Log on to the Orchestrator server **Orchestrator**

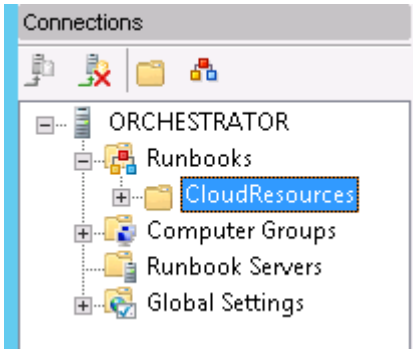
2. Open the **Runbook Designer**



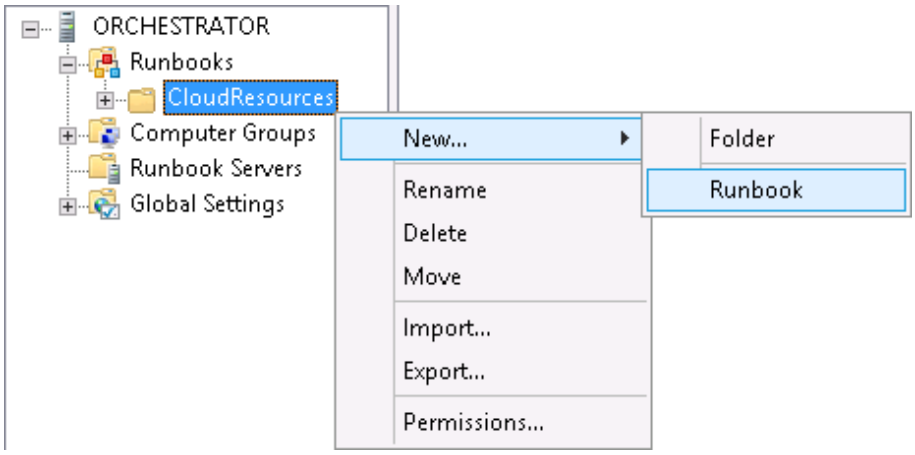
3. On the left, under **connections,** right-click on **Runbooks** and select **New... > Folder**



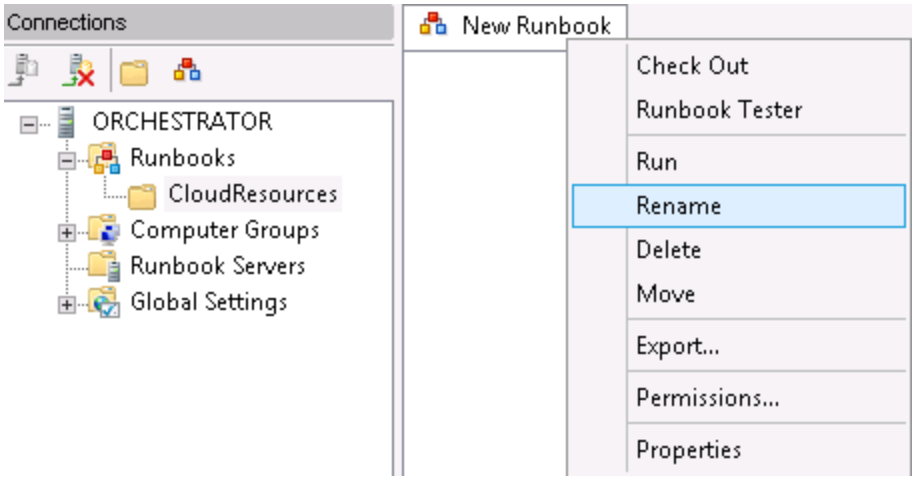
4. Give the new folder a name, for example **CloudResources**



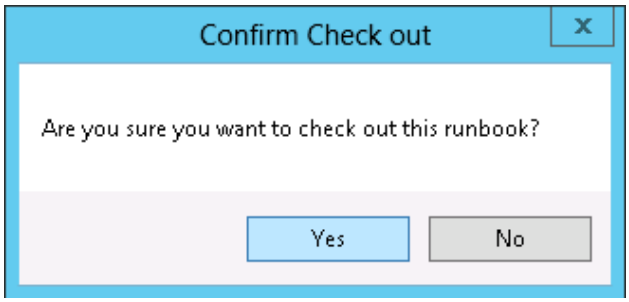
5. On the newly created **CloudResources** folder, right-click and choose **New... > Runbook**



6. On the newly created **Runbook**, right-click and choose **Rename** from the context menu.

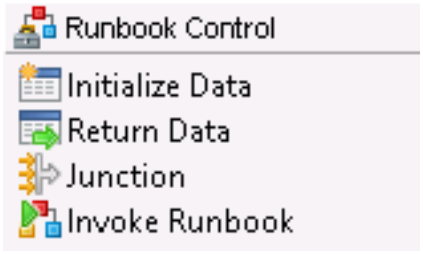


7. When you receive the **Confirm Check out**, select **Yes**

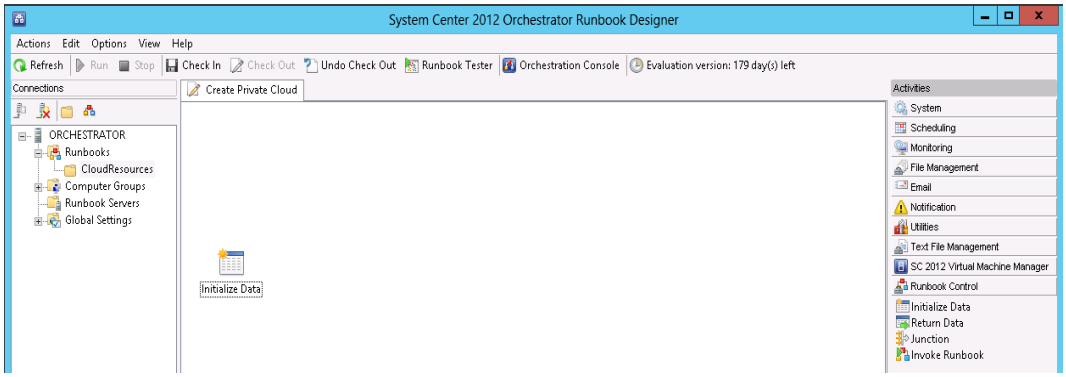


8. To assist in identifying the purpose of the new runbook, change its name to **Create Private Cloud**

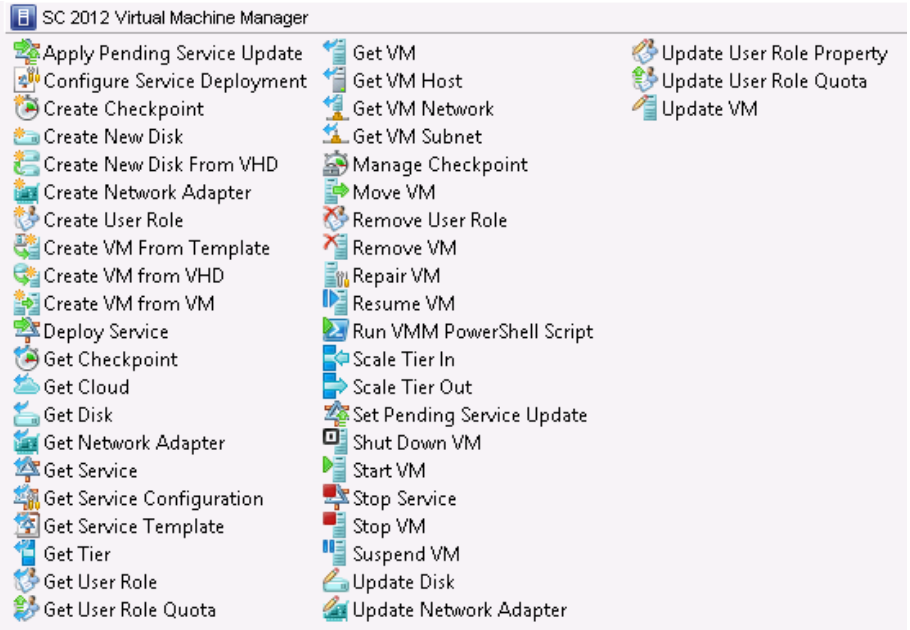
9. From the **activities** pane (right side of the console), select the **Runbook Control** folder, to expose the contained activities.



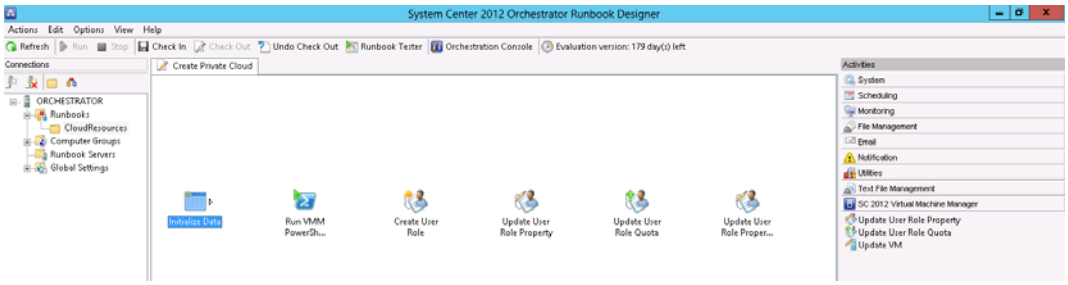
10. Drag the **Initialize Data** activity object to the runbook canvas (as illustrated below)



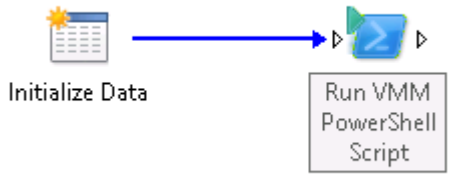
11. Next, select the **SC 2012 Virtual Machine Manager** folder to present its activities



12. Drag the following activities to the runbook canvas as illustrated below:
    - Run VMM PowerShell Script
    - Create User Role
    - Update User Role Property
    - Update User Role Quota
    - Update User Role Property – Becomes **Update User Role Property (2)**



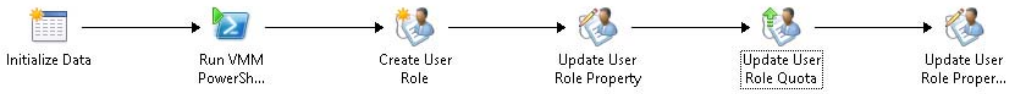
13. Next, we will create three connections in the runbook to wire up the workflow, these connections are commonly referred to as the "pipeline" or "databus", because this is the mechanism utilized to pass information between each activity. On the runbook canvas, move your mouse to the right of the **Initialize Data** activity until your pointer changes to a crosshair. Then, holding down the left mouse button, draw a connection between the **Initialize Data** and the **VMM PowerShell Script** activity.



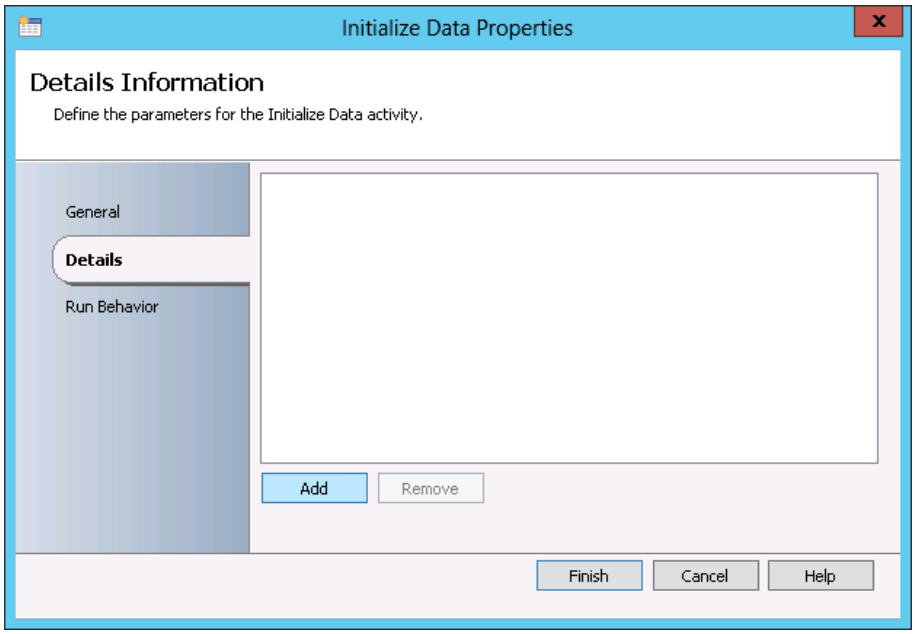
Repeat the process, this time from:

VMM PowerShell Script activity to the **Create User Role** activity
Create User Role activity to the **Update User Role Property** activity
Update User Role Property activity to the **Update User Role Quota activity**.
Update User Role Quota activity to the **Update User Role Property (2)**.

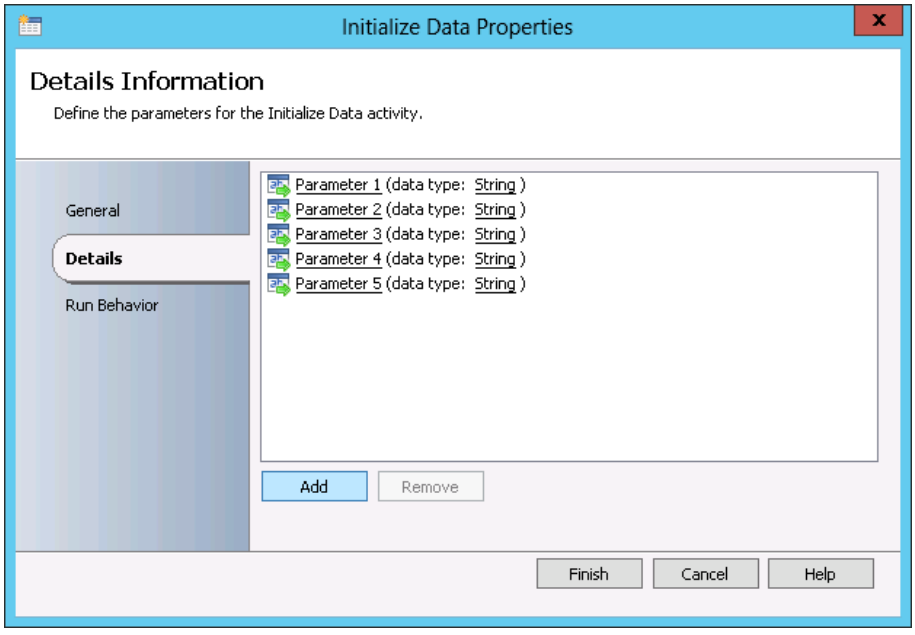
Jeff now has created his first, simple runbook. It will start with the first activity "**Initialize Data**," then proceed to the **VMM PowerShell Script**, then **Create User Role**, then **Update User Role Property**, **Update User Role Quota,** then and finally perform the **Update User Role Property** activity, passing the results of the previous activity over the pipeline, and available for the following activity to utilize.



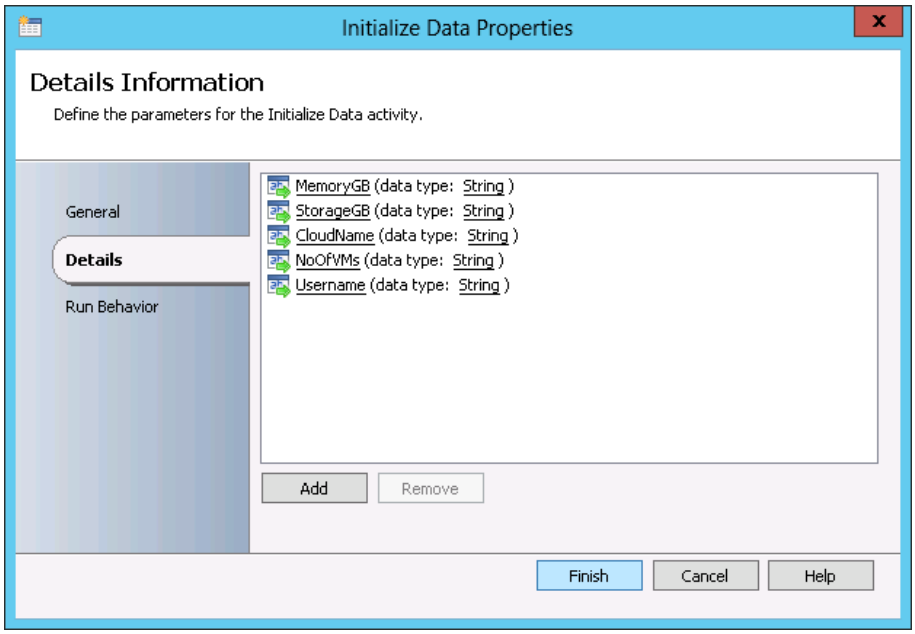
14. Jeff must now configure the activities in the runbook, as currently it has no configured task to perform. To begin, double-click on the **Initialize Data** activity to open the properties dialog box.



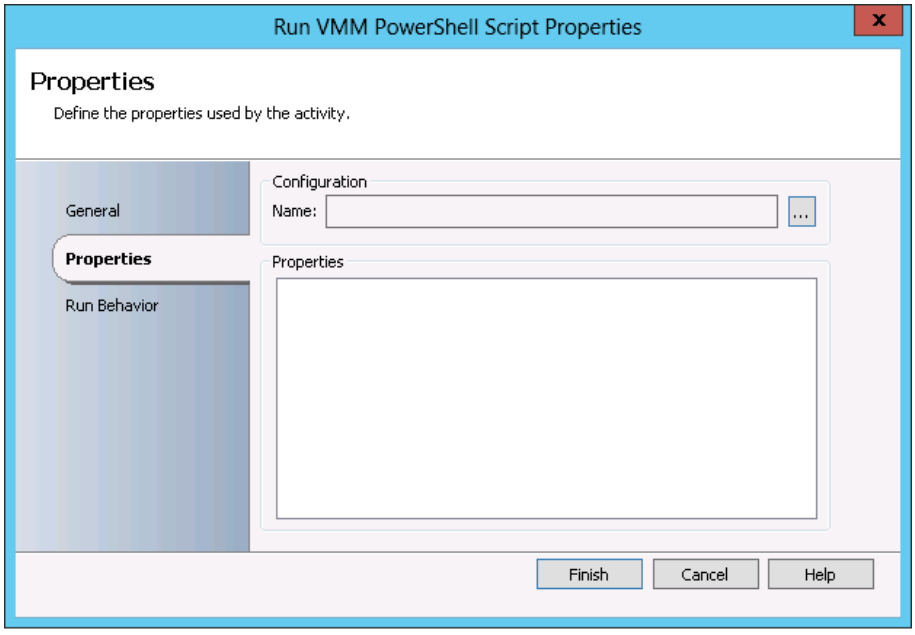
15. On the dialog **Details** page, select the **Add** button five times to add five parameters.



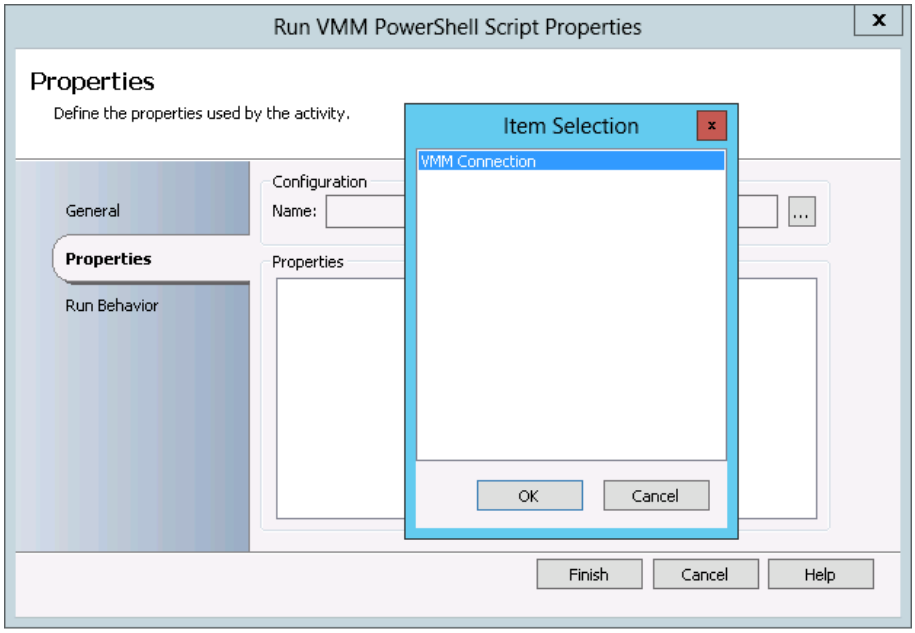
16. Next, select **Parameter 1** and change that name to **MemoryGB**, and then repeat for **Parameter 2, Parameter 3, Parameter 4 and Parameter 5**, changing their respective names to **StorageGB**, **CloudName**, **NoOfVMs** and **Username** . Once complete select **Finish** to close the dialog



17. Back on the canvas, double-click the **Run VMM PowerShell Script** activity.



18. Select the **"..."** button on the right of the **Configuration Name:** and then select the connection (**VMM Connection**) that Jeff created previously

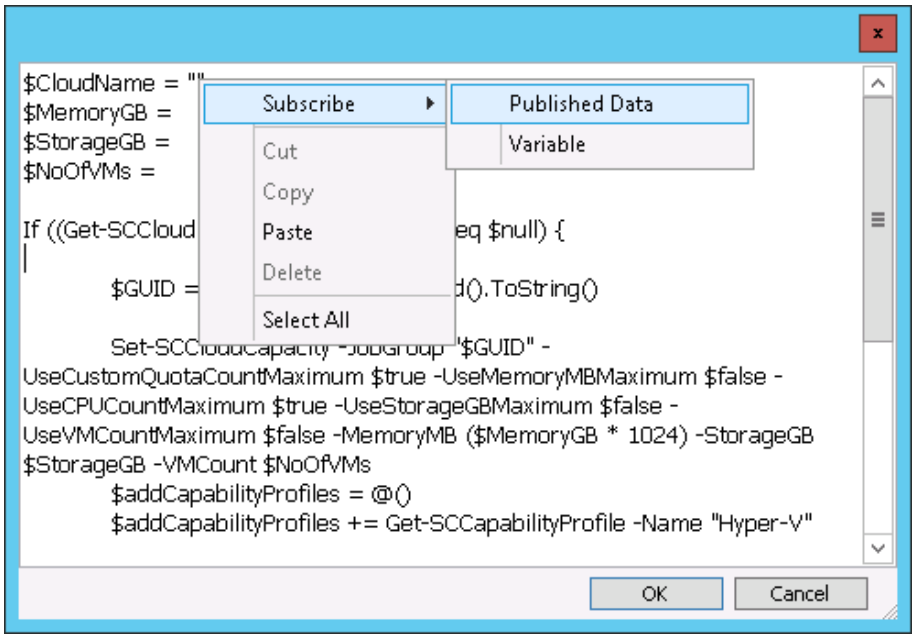


19. In the text field next to **PowerShell Script**, right-click and choose **Expand...** Copy and paste the following PowerShell script.

```
$CloudName = ""

$MemoryGB =

$StorageGB =

$NoOfVMs =


If ((Get-SCCloud -Name "$CloudName") -eq $null) {


$GUID = [System.Guid]::NewGuid().ToString()


Set-SCCloudCapacity -JobGroup "$GUID" -UseCustomQuotaCountMaximum $true -
UseMemoryMBMaximum $false -UseCPUCountMaximum $true -UseStorageGBMaximum
$false -UseVMCountMaximum $false -MemoryMB ($MemoryGB * 1024) -StorageGB
$StorageGB -VMCount $NoOfVMs

$addCapabilityProfiles = @()
```

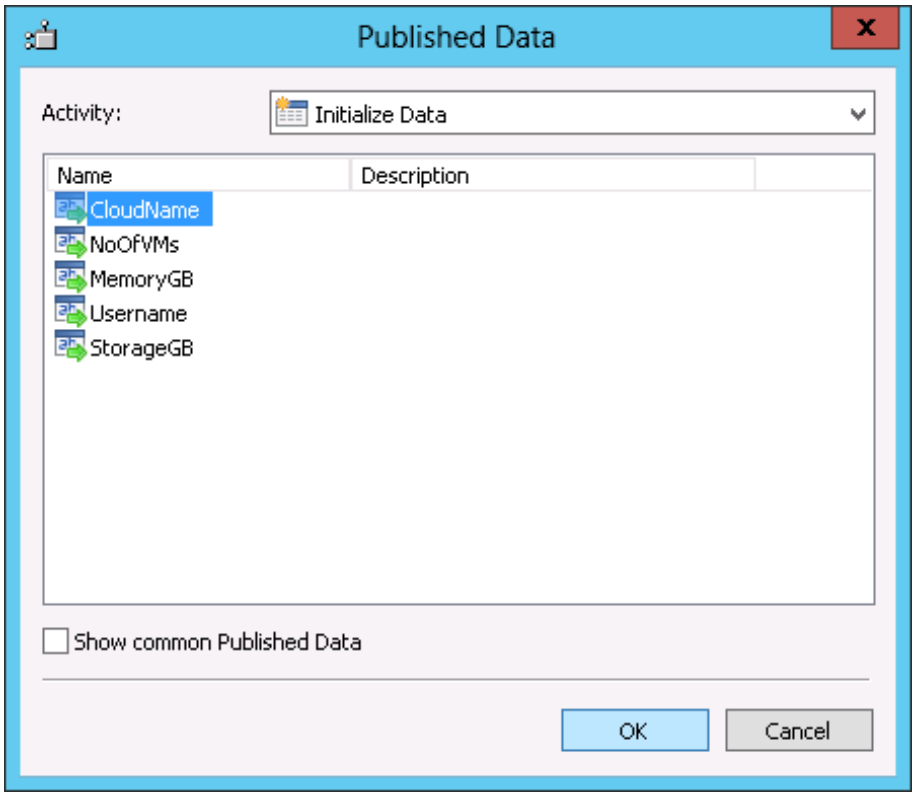
```
$addCapabilityProfiles += Get-SCCapabilityProfile -Name "Hyper-V"


Set-SCCloud -JobGroup "$GUID" -RunAsynchronously -AddCapabilityProfile
$addCapabilityProfiles


$hostGroup = (Get-SCVMHostGroup | where {$_.ParentHostGroup -eq $null})


$hostGroups = @()

$hostGroups += $hostGroup


$Cloud = New-SCCloud -JobGroup "$GUID" -VMHostGroup $hostGroups -Name
$CloudName -Description ""


}
```

20. On the first line of the PowerShell script **($CloudName = “”**) between the quote marks **(“”)** right click and select **Subscribe > Published Data** from the context menu.

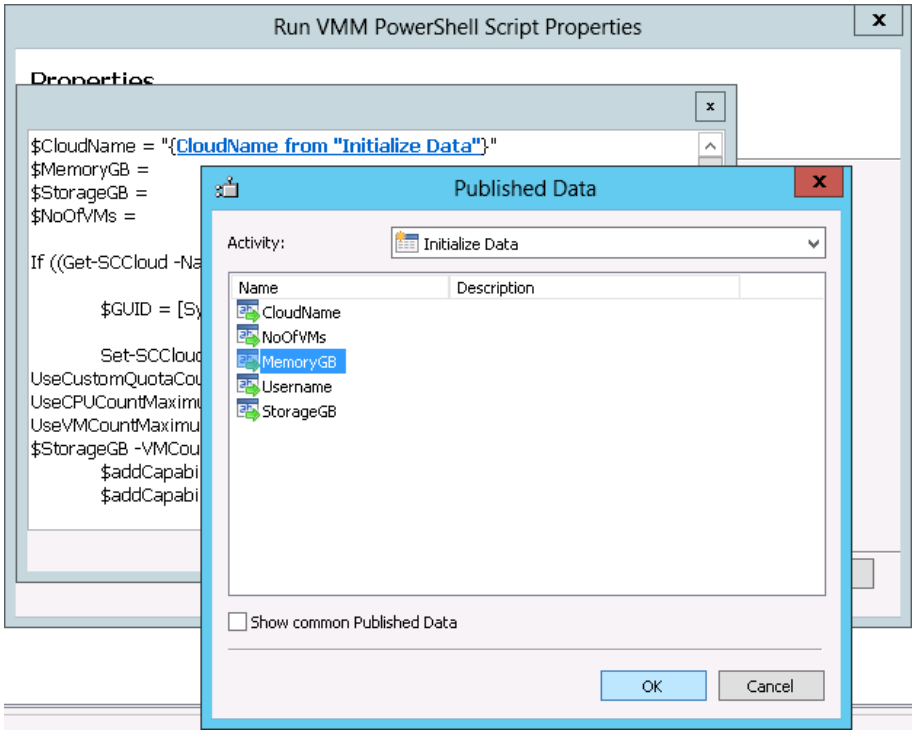


21. In the Activity drop-down, the “Initialize Data” activity should be presented. The “pipeline” or “databus” attributes of “MemoryGB”, “StorageDB”, “CloudName” and “NoOfVMs” which we created earlier are now available to select.

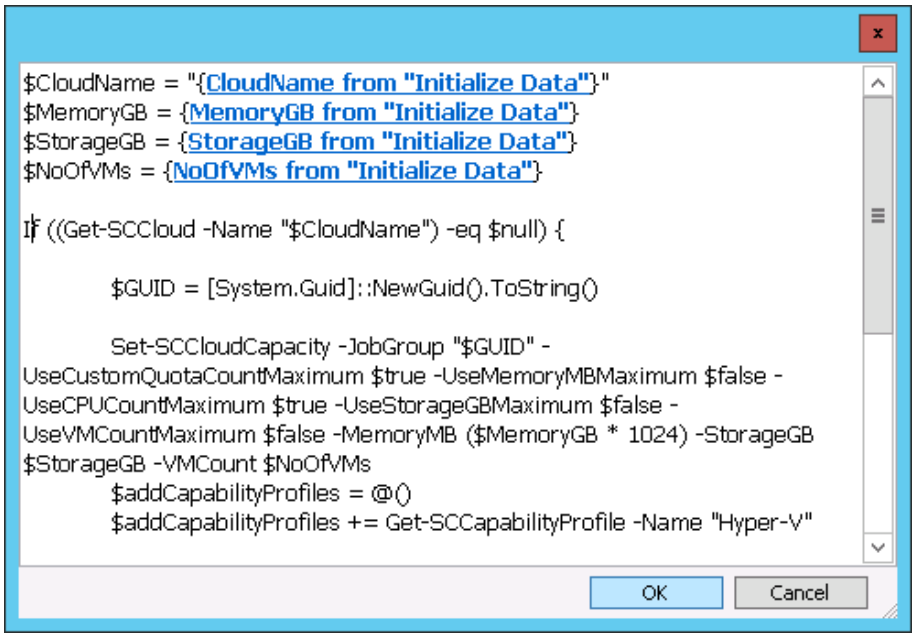
    Select the **CloudName** property and click **OK**



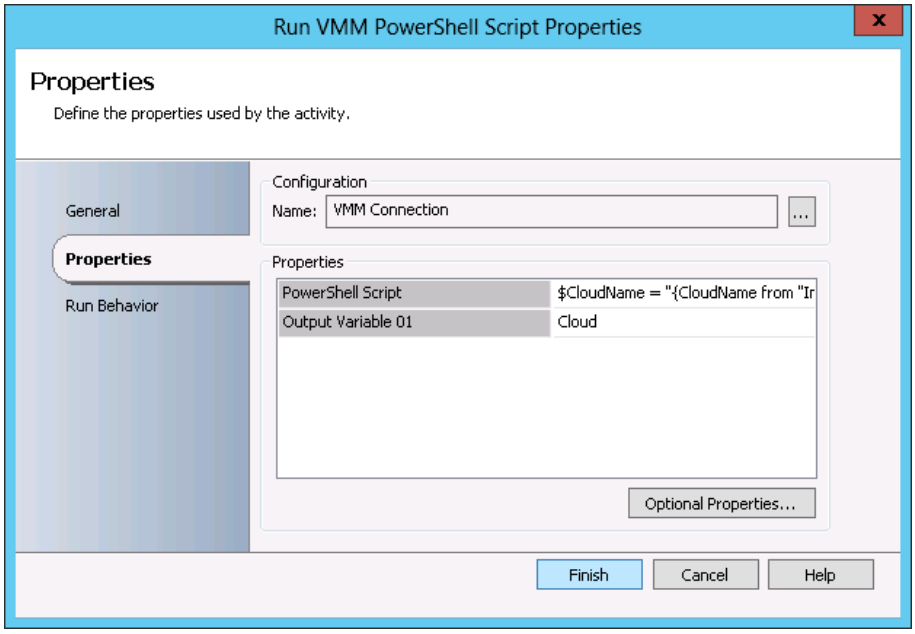
22. For the following PowerShell script lines, repeat the previous steps however right click after the equals sign (=) and select **Subscribe > Published Data** from the context menu and select the associated properties (MemoryGB, StorageGB and NoOfVMs).
    - **$MemoryGB =**
    - **$StorageGB =**
    - **$NoOfVMs =**



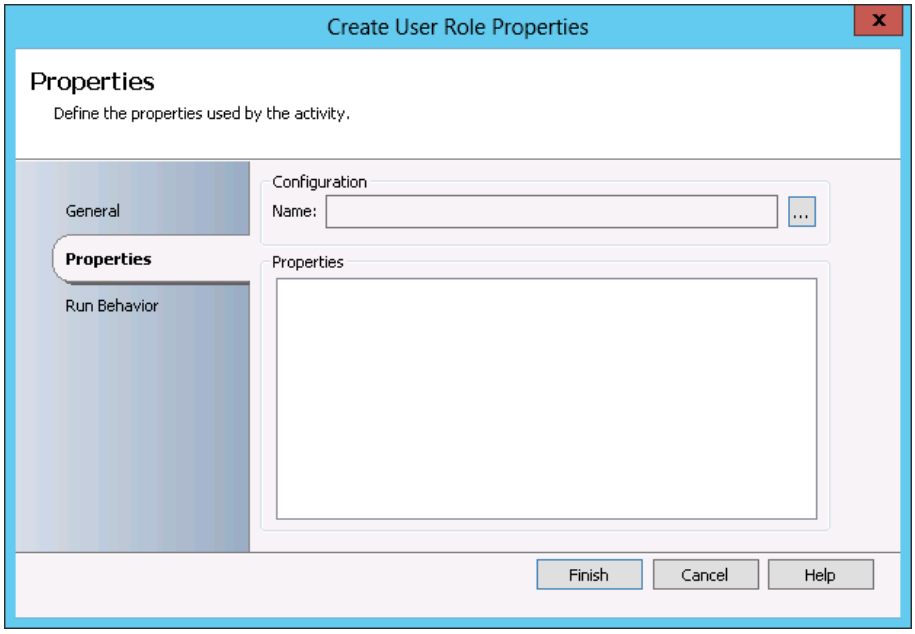
23. Confirm that the first four lines of the PowerShell script are as below and click **OK** and then click **Finish**.



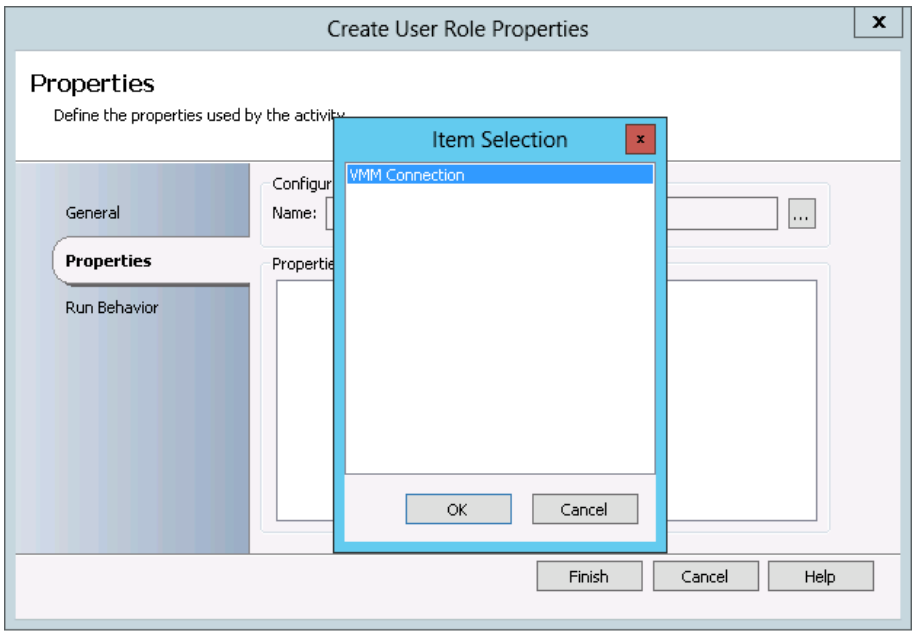
24. In the text field next to **Output Variable 01** click in the text field and type **"Cloud"**. Click **Finish**.



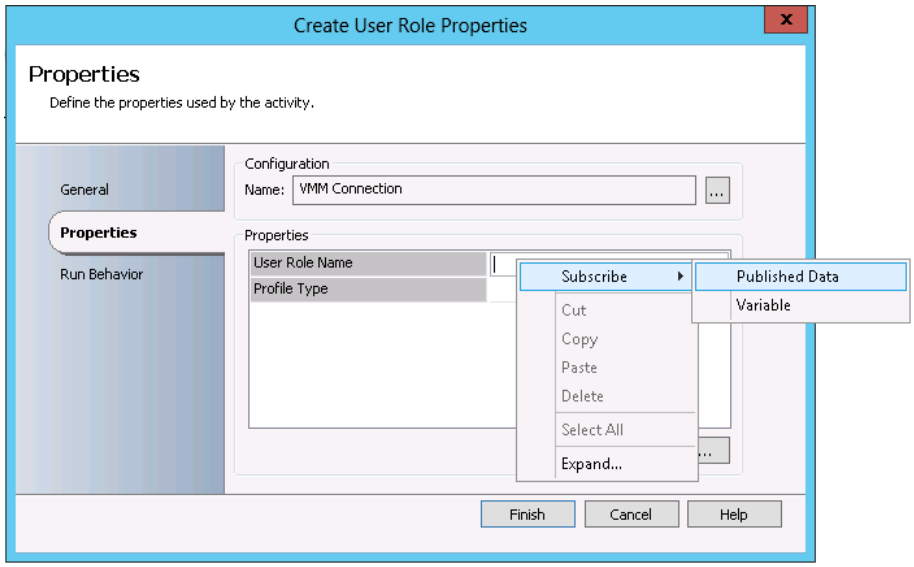
25. Back on the canvas double-click the **Create User Role** activity to open the properties dialog box.



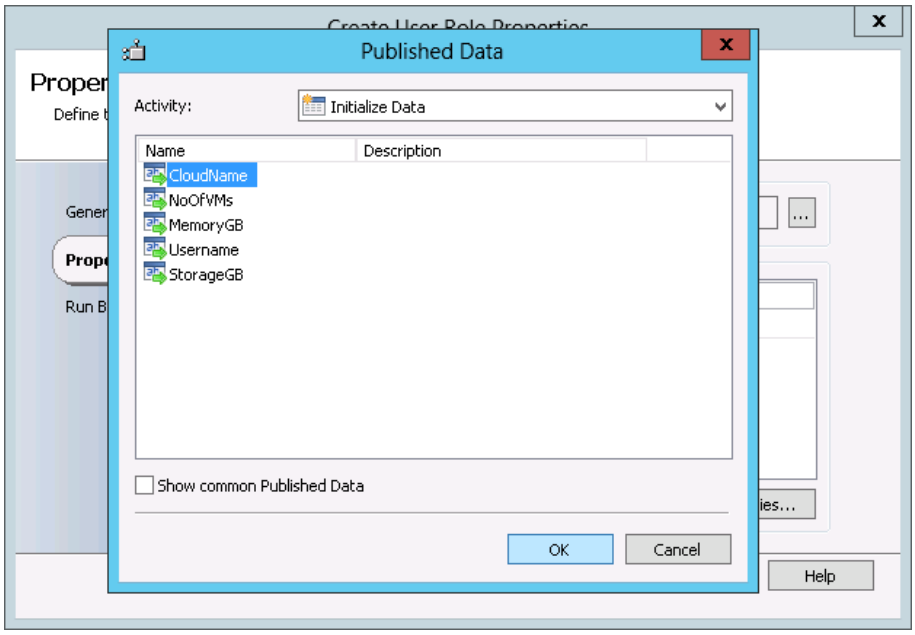
26. Select the **"..."** button on the right of the **Configuration Name:** and then select the connection (**VMM Connection**) that Jeff created earlier.



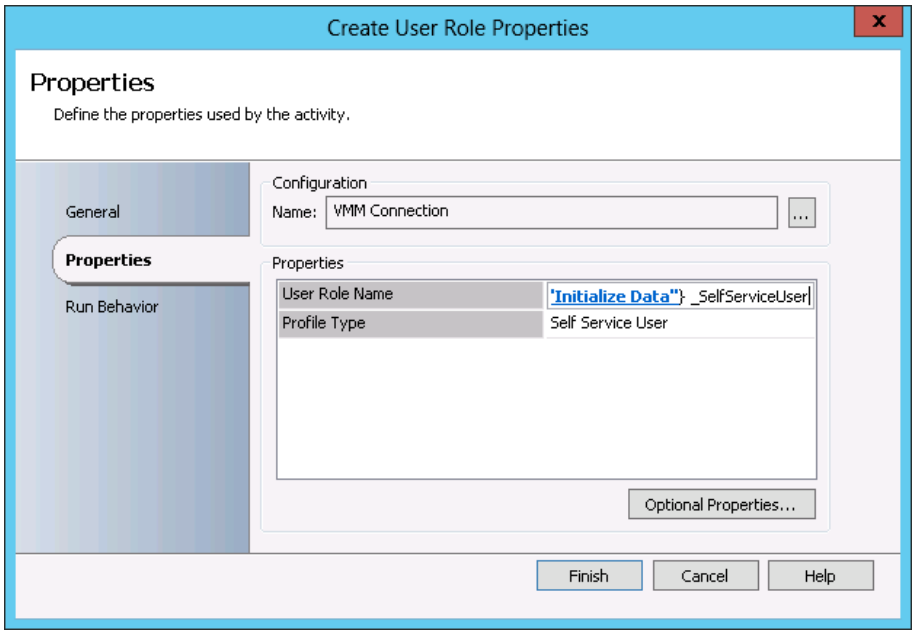
27. In the text field next to **User Role Name**, right click and select **Subscribe > Published Data** from the context menu.



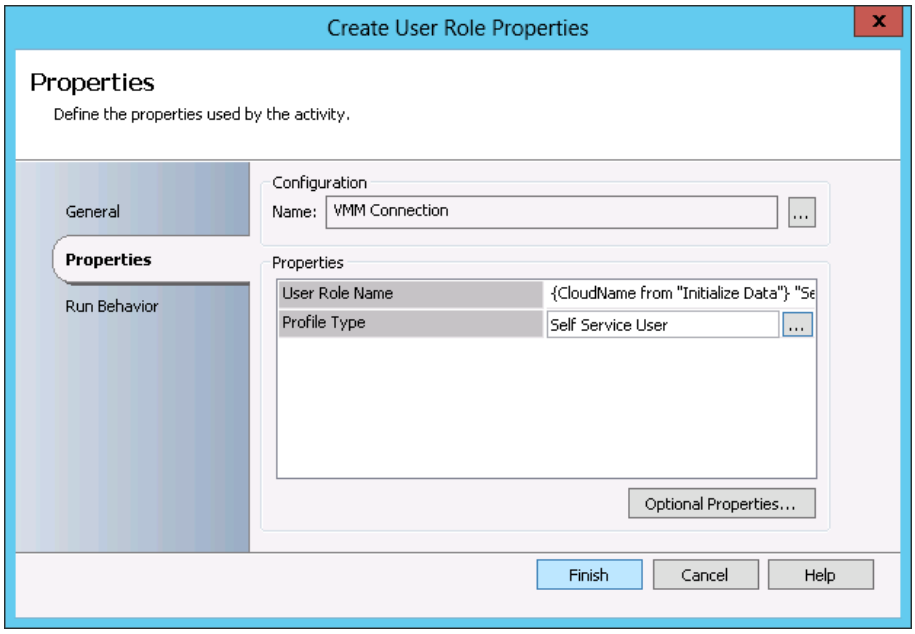
28. In the Activity drop-down, select **"Initialize Data"**. The "pipeline" or "databus" attributes of "MemoryGB", "StorageDB", "CloudName" and "NoOfVMs" which we created earlier are now available to select. Select **CloudName** and click **OK**



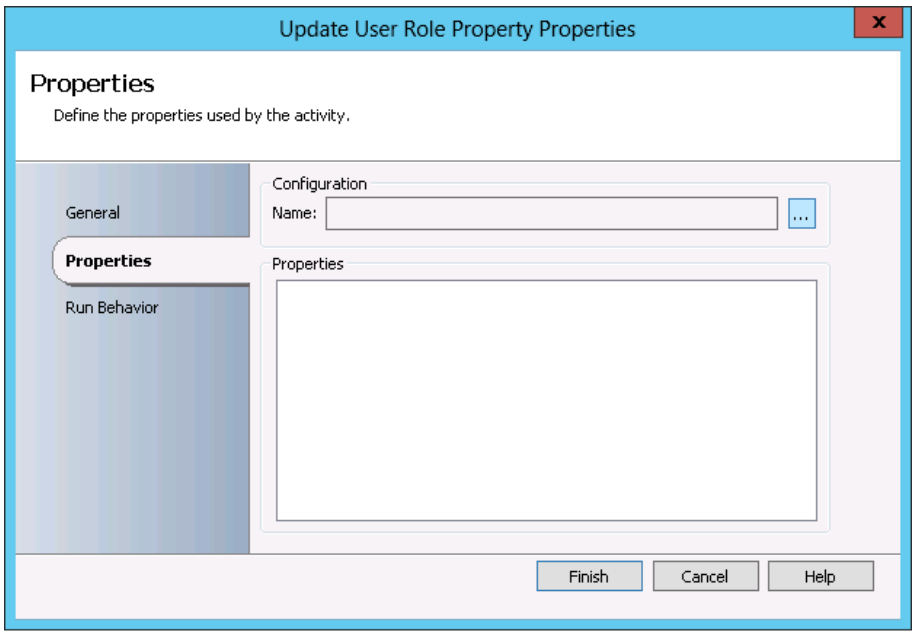
29. At the end of the text box add the text: **_SelfServiceUser**



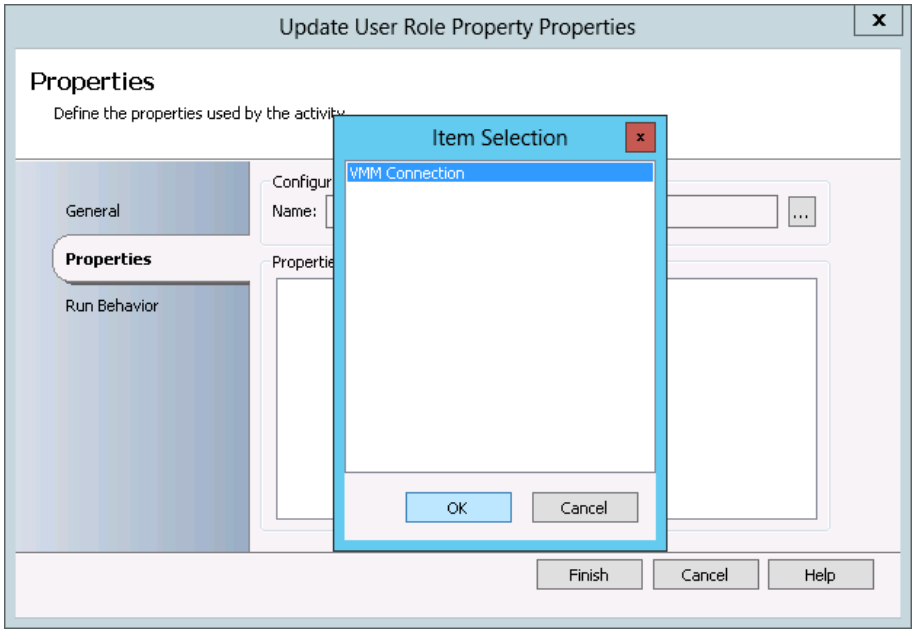
30. In the text field next to **Profile Type** click in the text field and select the **"..."** and select **Self Service User** from the Profile Type window. Click **OK**. Click **Finish.**



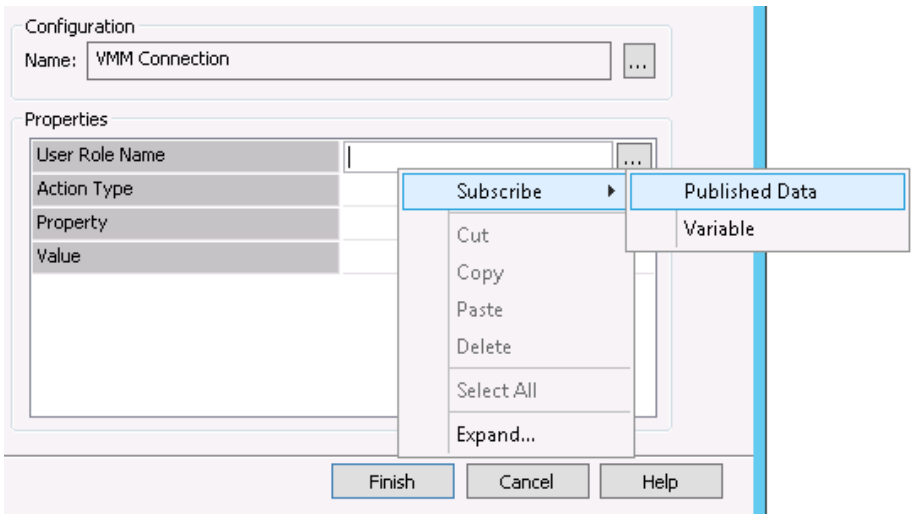
31. Back on the canvas double-click the **Update User Role Property** activity to open the properties dialog box.



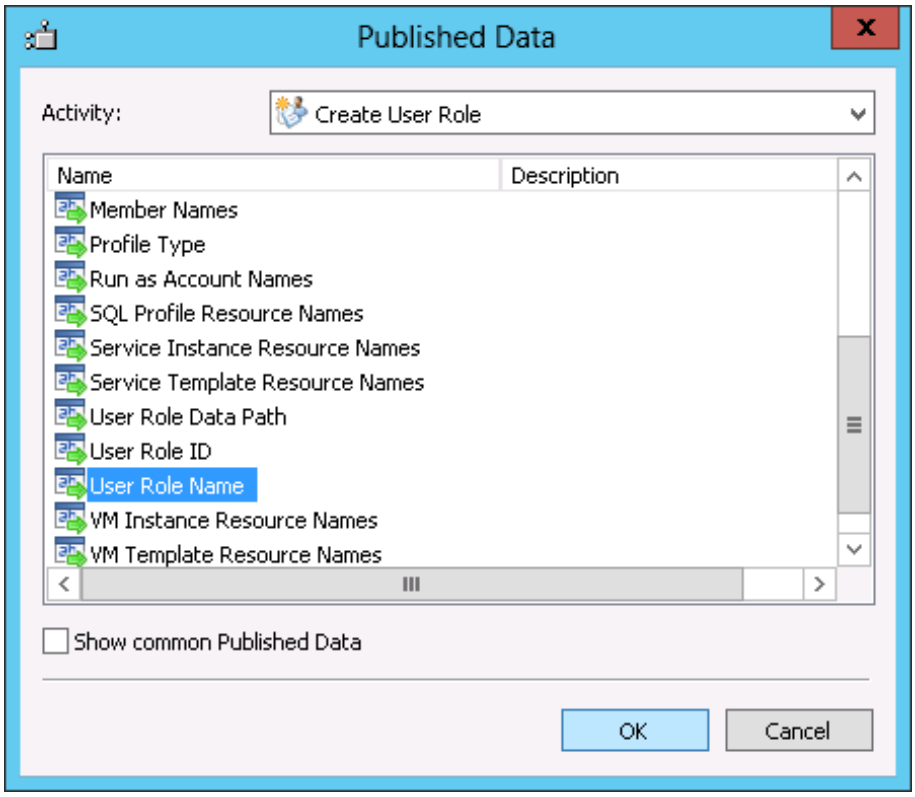
32. Select the "**...**" button on the right of the **Configuration Name:** and then select the connection (**VMM Connection**) that you created earlier in Chapter 4.



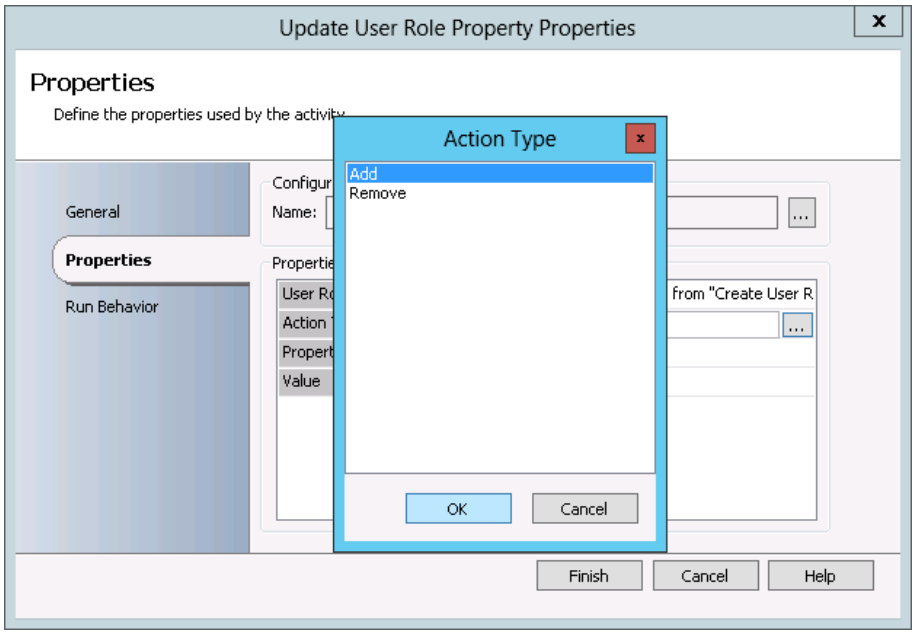
33. In the text field next to **User Role Name**, right click and select **Subscribe > Published Data** from the context menu.



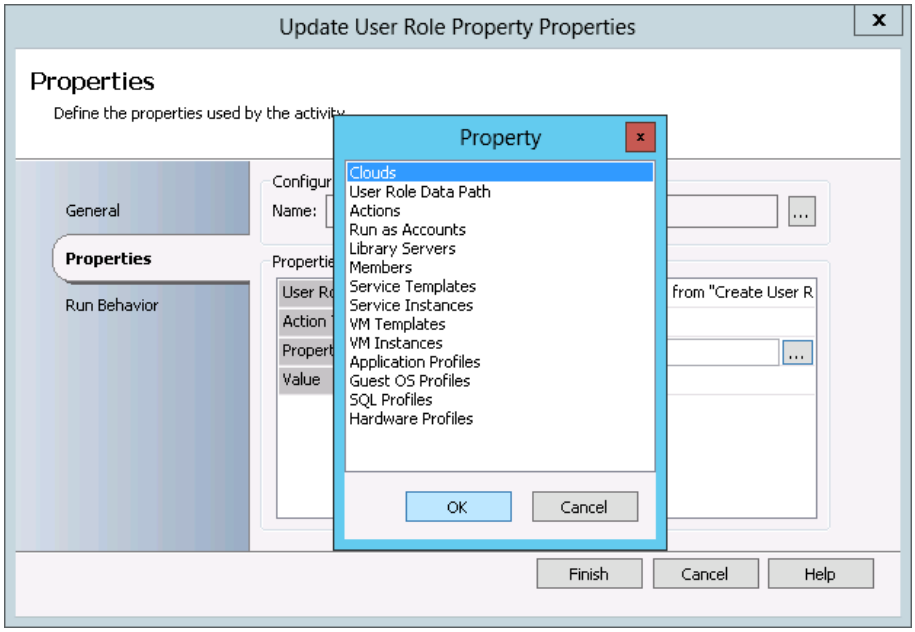
34. In the Activity drop-down, the "Create User Role" activity should be presented. From the list of "databus" attributes select **User Role Name** and click **OK**



35. In the text field next to **Action Type** click in the text field and select the **"..."** and select **Add** from the Action Type window. Click **OK**.

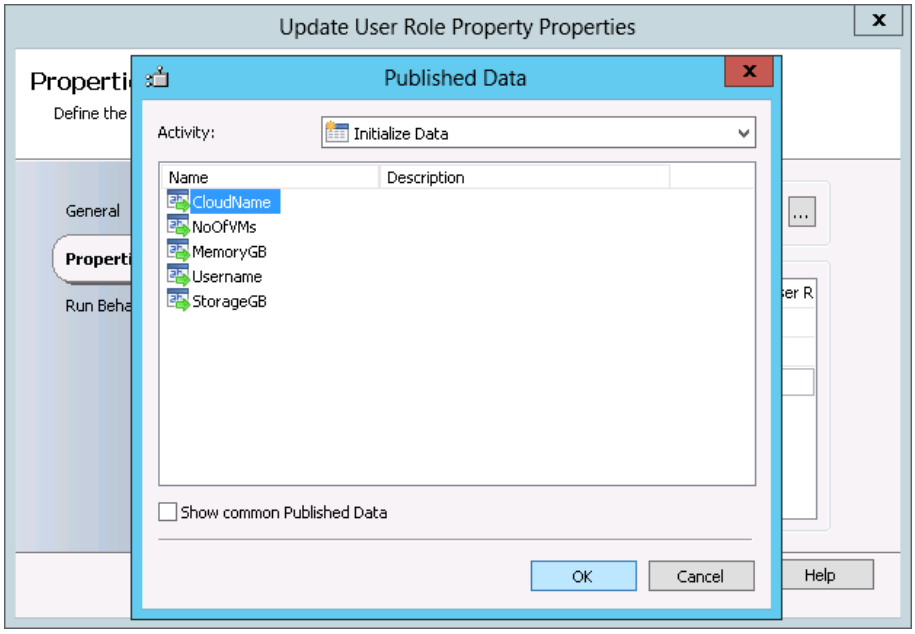


36. In the text field next to **Property** click in the text field and select the "…" and select **Clouds** from the Property window. Click **OK**.

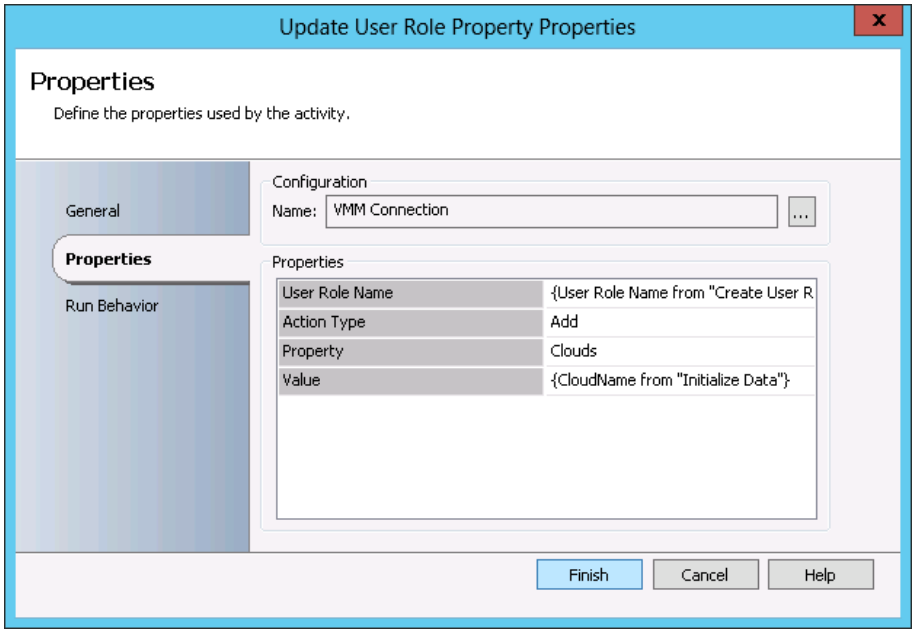


37. In the text field next to **Value**, right click and select **Subscribe > Published Data** from the context menu. In the Activity drop-down, select "**Initialize Data**". . From the list of "databus" attributes select

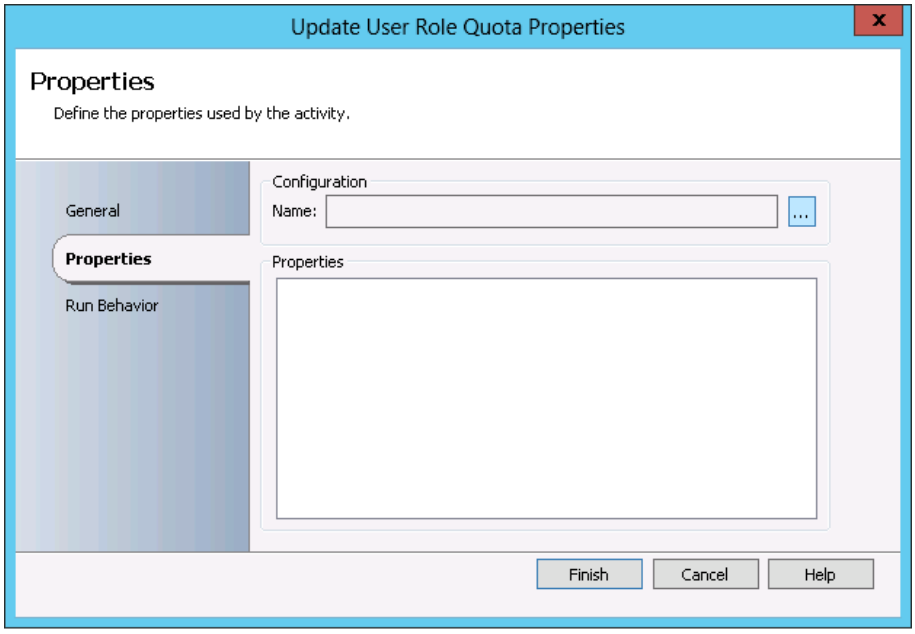
CloudName and click **OK**.



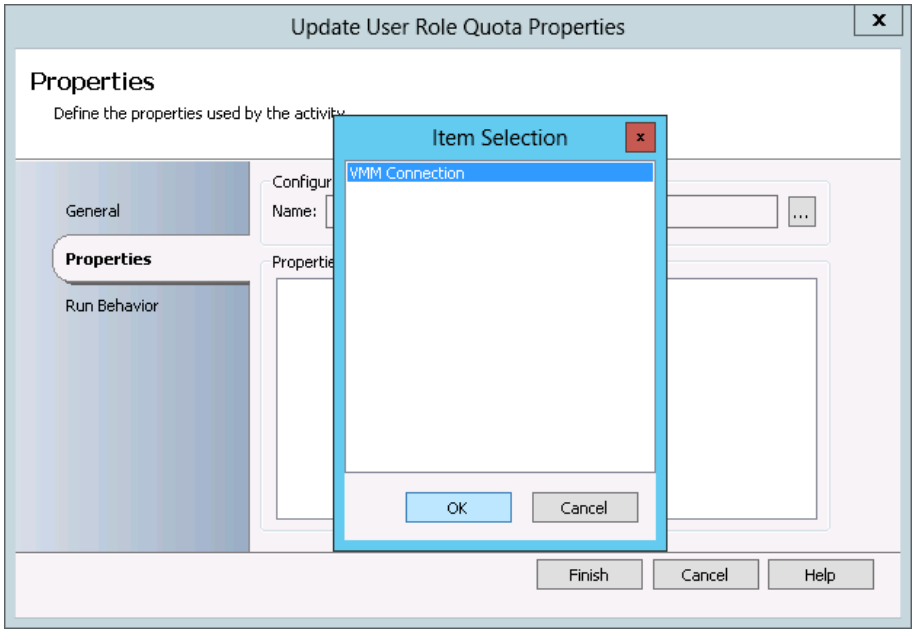
38. Click **Finish**.



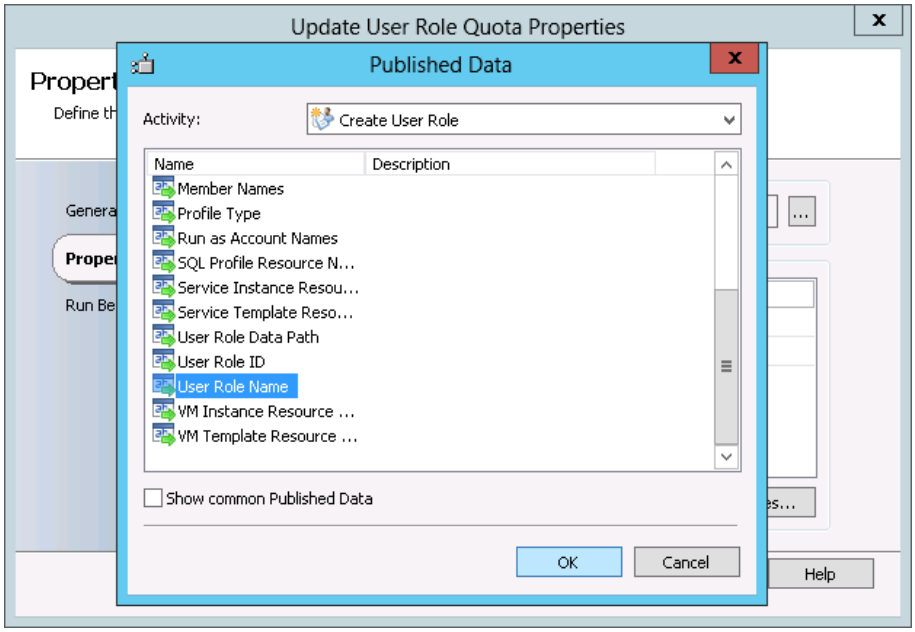
39. Back on the canvas double-click the **Update User Role Quota** activity to open the properties dialog box.



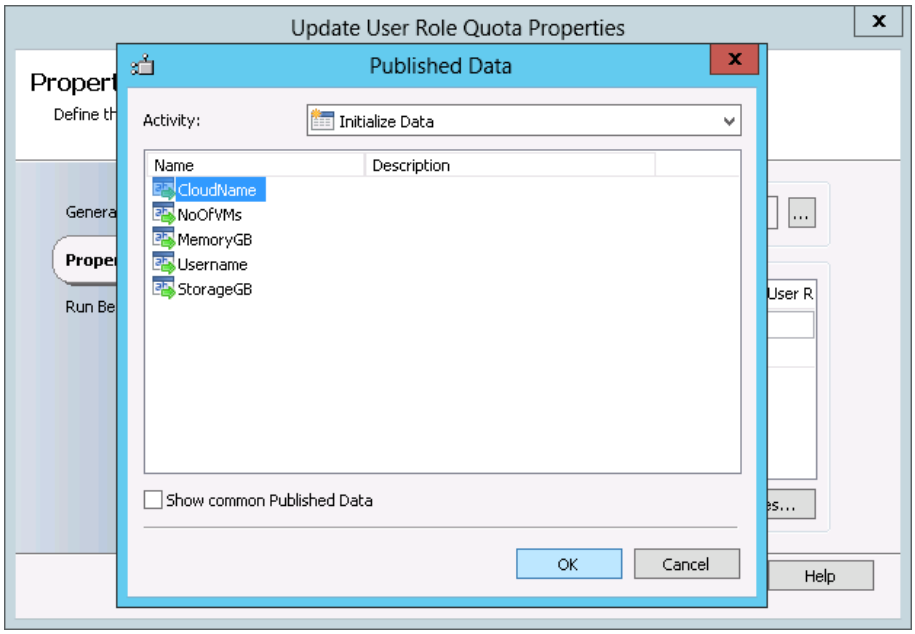
40. Select the **"..."** button on the right of the **Configuration Name:** and then select the connection (**VMM Connection**) that Jeff had created earlier and then click **OK**.



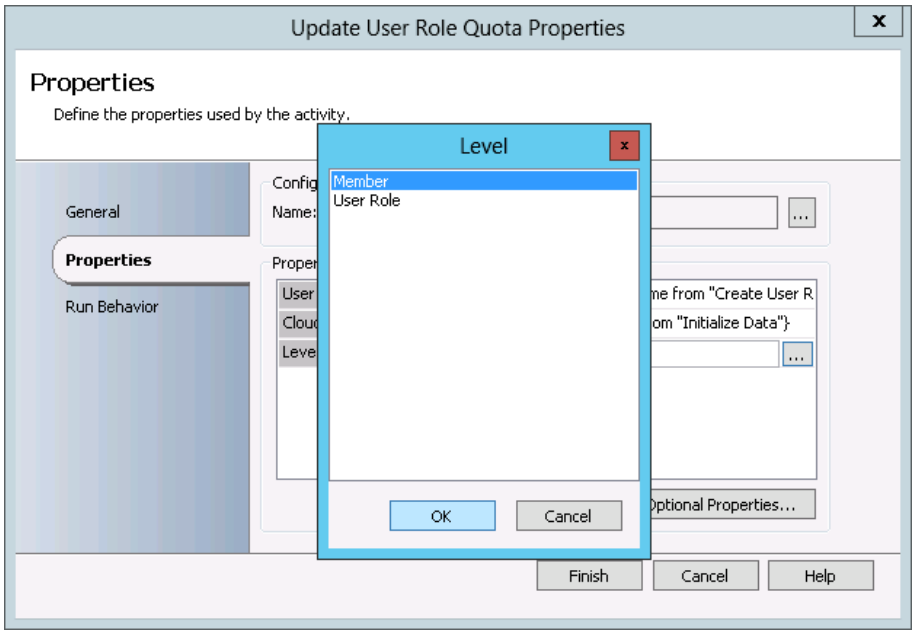
41. In the text field next to **User Role Name**, right click and select **Subscribe > Published Data** from the context menu. In the Activity drop-down, select **"Create User Role"**. From the list of "databus" attributes select **User Role Name** and click **OK**



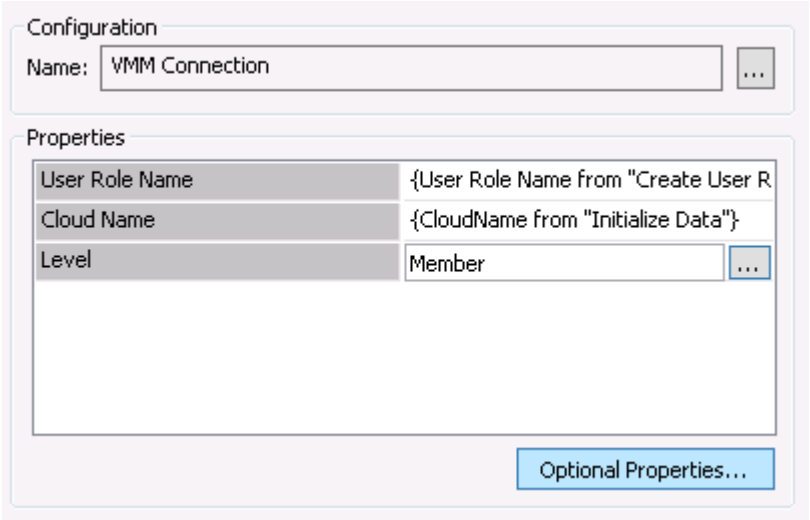
42. In the text field next to **Cloud Name**, right click and select **Subscribe > Published Data** from the context menu. In the Activity drop-down, select **"Initialize Data"**. From the list of "databus" attributes select **Cloud Name** and click **OK**



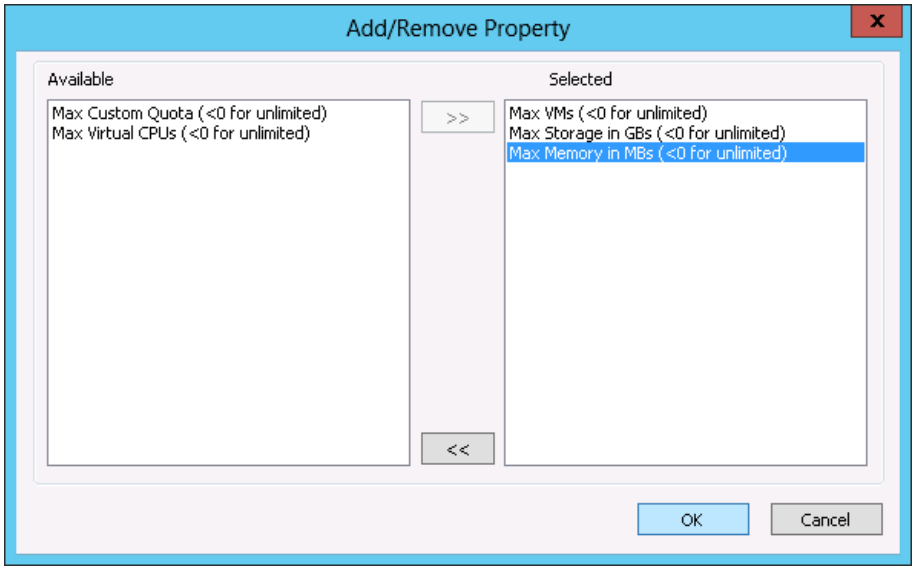
43. In the text field next to **Level** click in the text field and select the **"..."** and select **Member** from the Level window. Click **OK**.



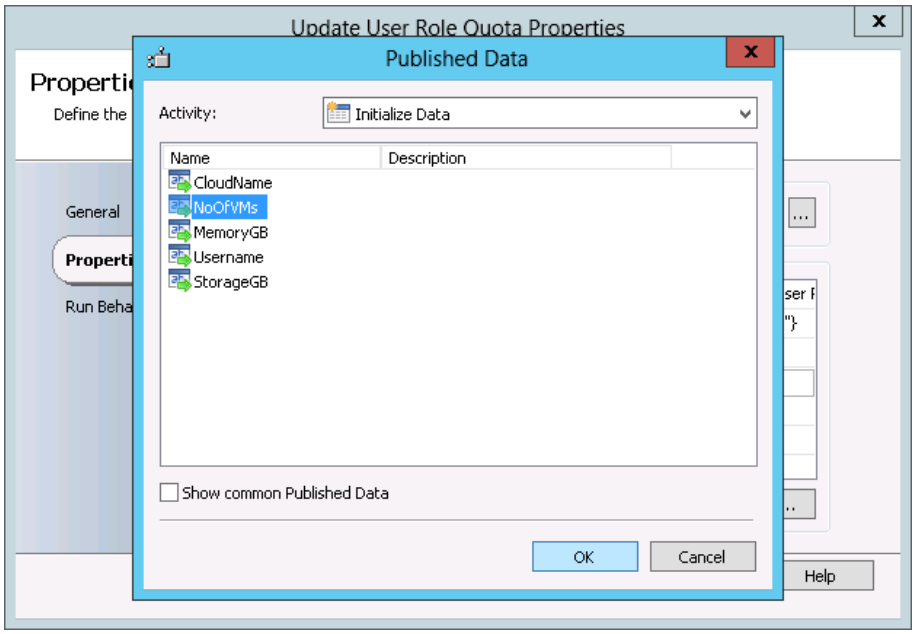
44. Click **Optional Properties...**



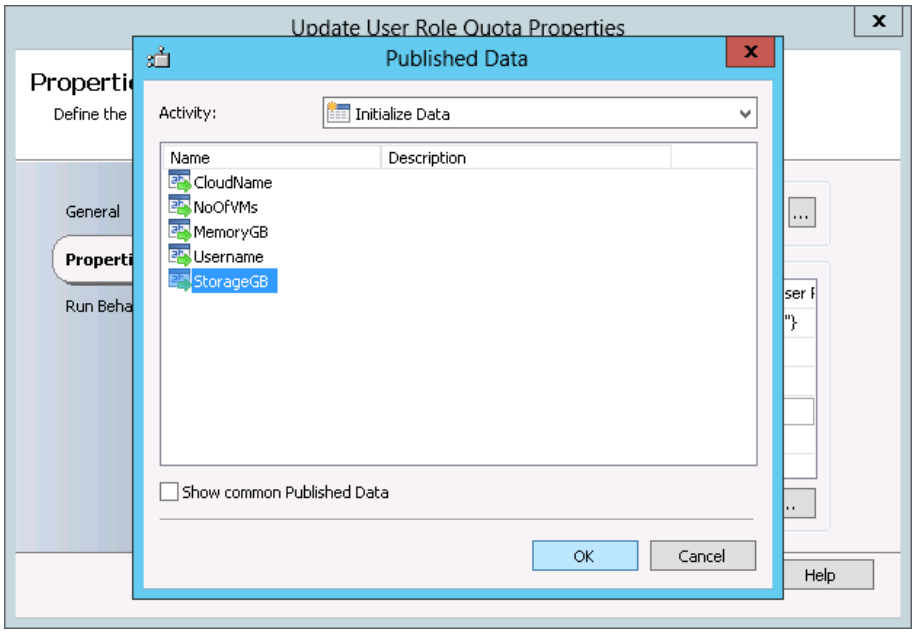
45. From add/remove property add **Max VMs**, **Max Storage in GBs** and **Max Memory in MBs** and click **OK**



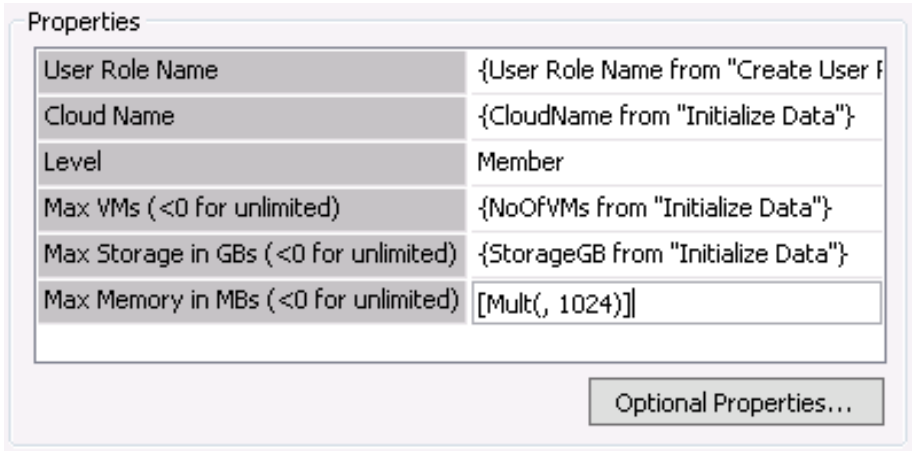
46. In the text field next to **Max VMs (<0 for unlimited)**, right click and select **Subscribe > Published Data** from the context menu. In the Activity drop-down, select **"Initialize Data"**. From the list of "databus" attributes select **NoOfVMs** and click **OK**



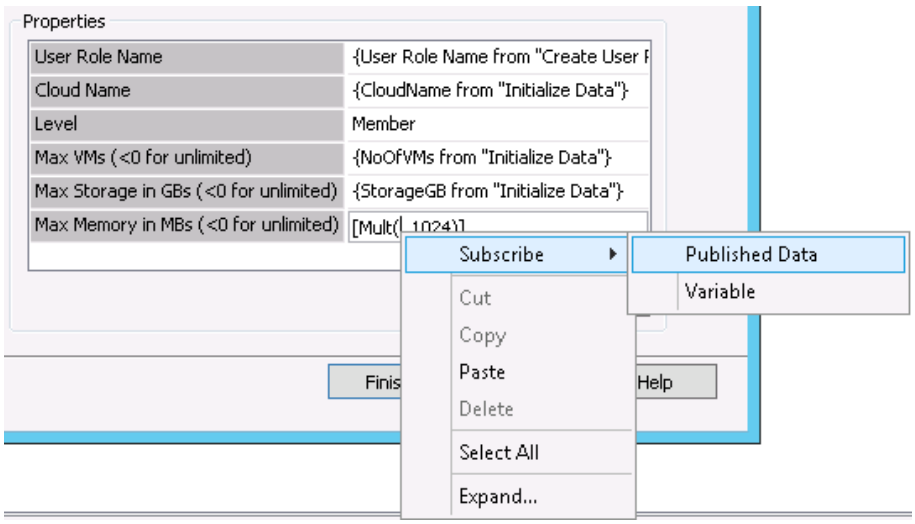
47. In the text field next to **Max Storage in GBs (<0 for unlimited)**, right click and select **Subscribe > Published Data** from the context menu. In the Activity drop-down, select **"Initialize Data"**. From the list of "databus" attributes select **StorageGB** and click **OK**



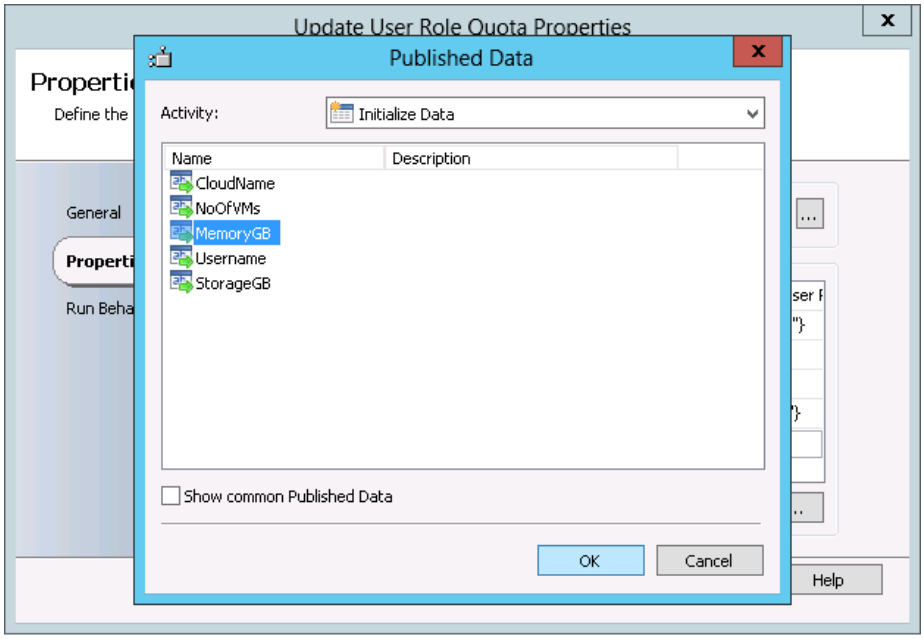
48. In the text field next to **Max Memory in MBs (<0 for unlimited)** type **[Mult(, 1024)]**



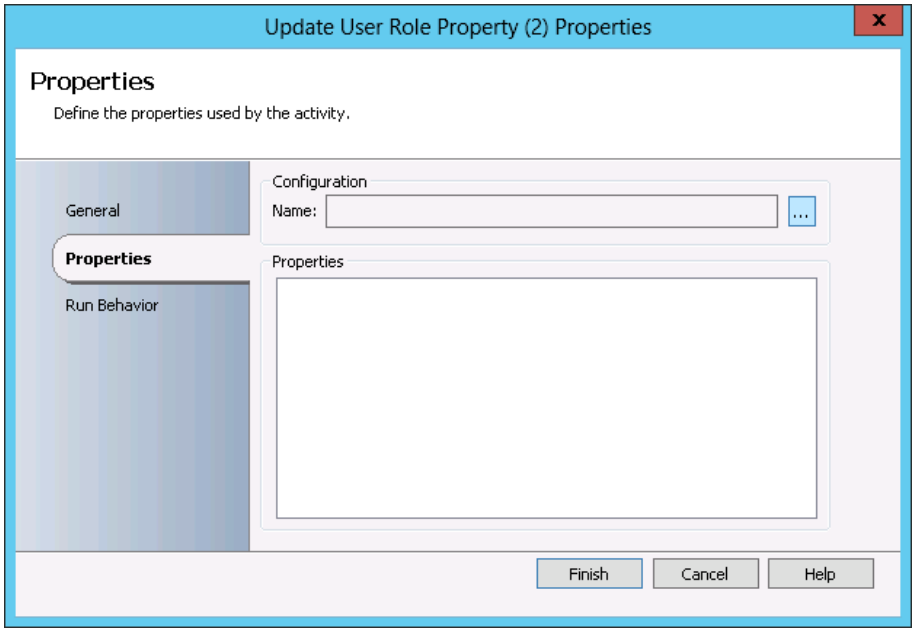
49. Right click between the left parenthesis (**(**) and comma (,)and select **Subscribe > Published Data** from the context menu.



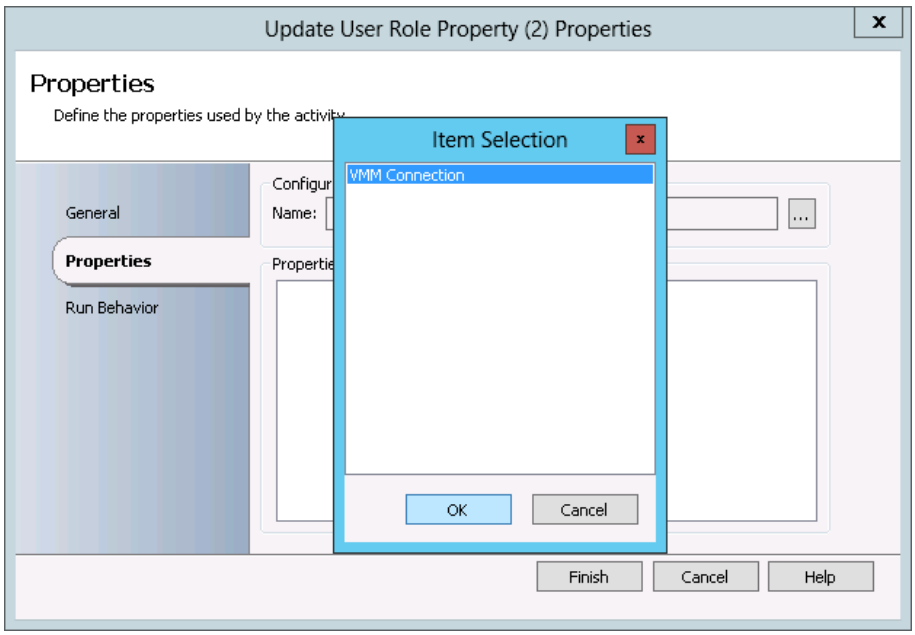
50. In the Activity drop-down, select **"Initialize Data"**. From the list of "databus" attributes select **MemoryGB** and click **OK.** Then click **Finish**



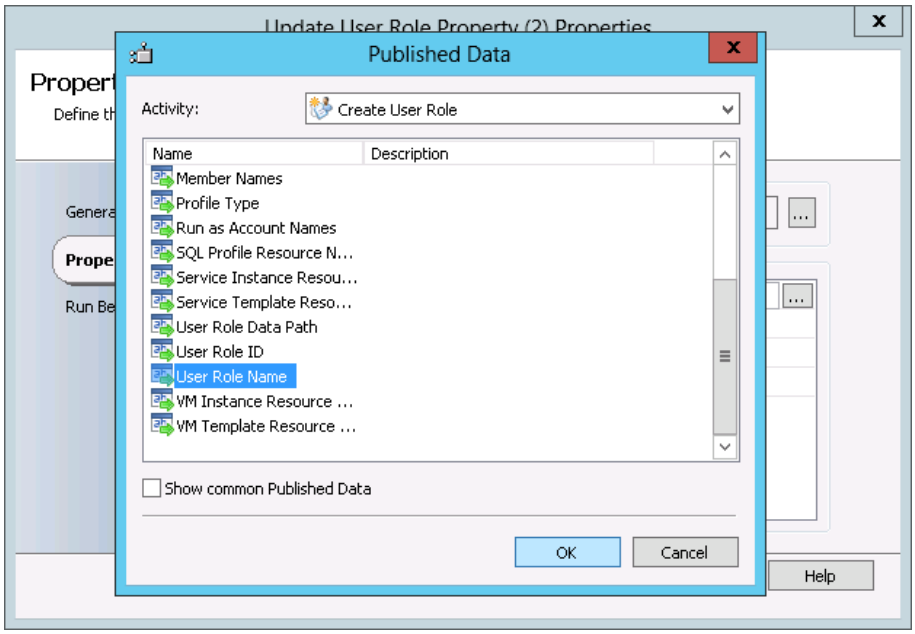
51. Back on the canvas double-click the **Update User Role Property (2)** activity to open the properties dialog box.



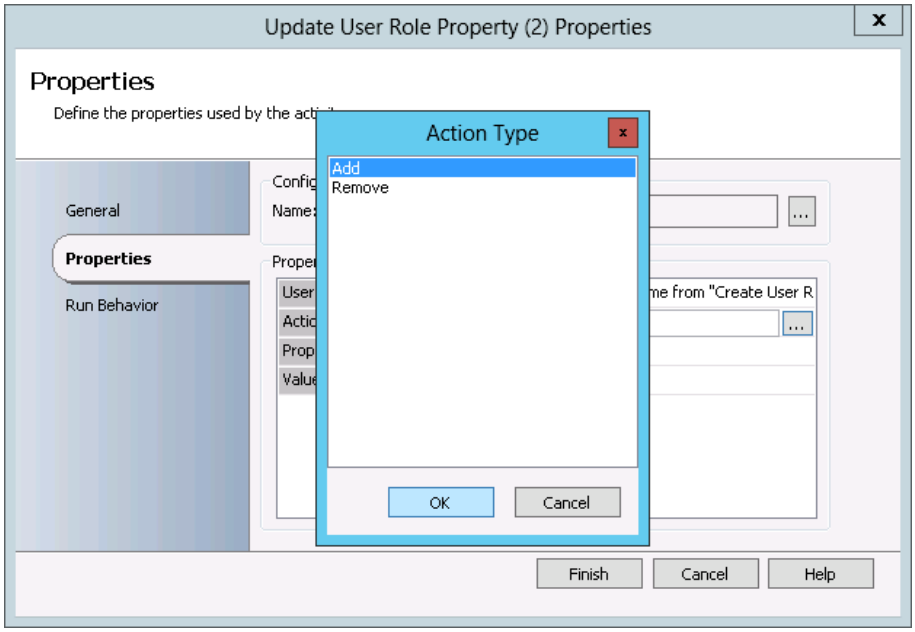
52. Select the **"..."** button on the right of the **Configuration Name:** and then select the connection (**VMM Connection**) that you created earlier in Chapter 4.



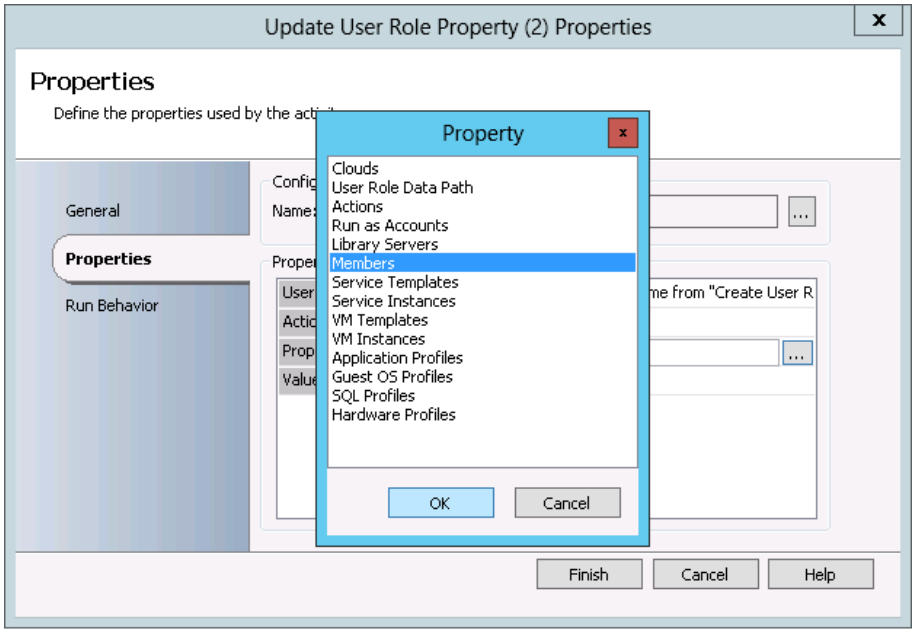
53. In the text field next to **User Role Name**, right click and select **Subscribe > Published Data** from the context menu. In the Activity drop-down, select **"Create User Role"**. From the list of "databus" attributes select **User Role Name** and click **OK**



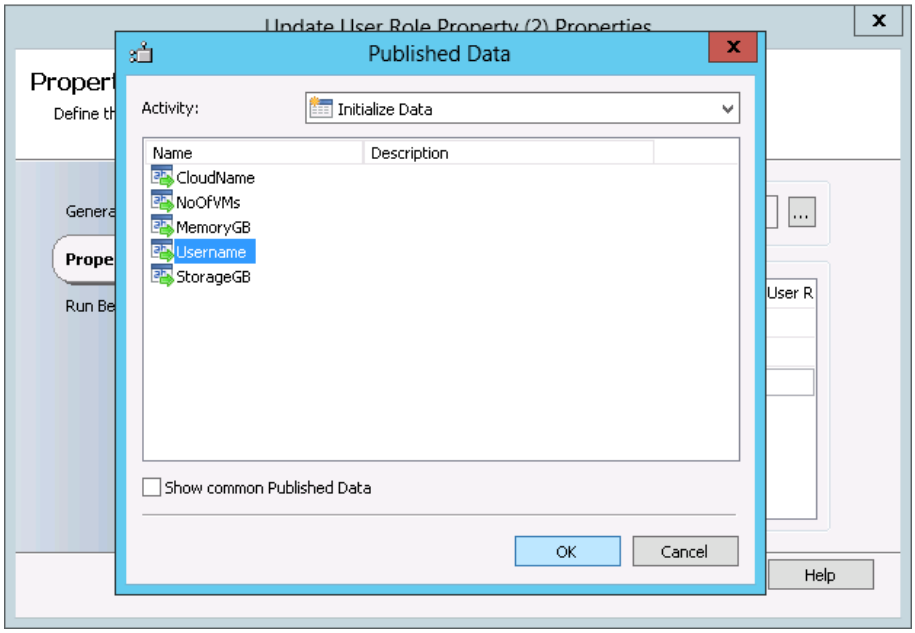
54. In the text field next to **Action Type** click in the text field and select the **"..."** and select **Add** from the Action Type window. Click **OK**.



55. In the text field next to **Property** click in the text field and select the **"..."** and select **Members** from the Property window. Click **OK**.



56. In the text field next to **Value**, right click and select **Subscribe > Published Data** from the context menu. In the Activity drop-down, select **"Initialize Data"**. From the list of "databus" attributes select **Username** and click **OK**. Click **Finish**



Testing the Runbook

Jeff has now created his first runbook. The only thing he now has to do is start the runbook, provide the four parameters and his job should be automated, saving him time in the future. But before he relaxes, he must now check to see if the runbook actually works. For that, he is going to use the **Runbook Tester** which is integrated into the **Orchestrator Runbook Designer** we have already been utilizing. For his test, Jeff wants to create a new cloud called "XYZ" and assign the following resources:

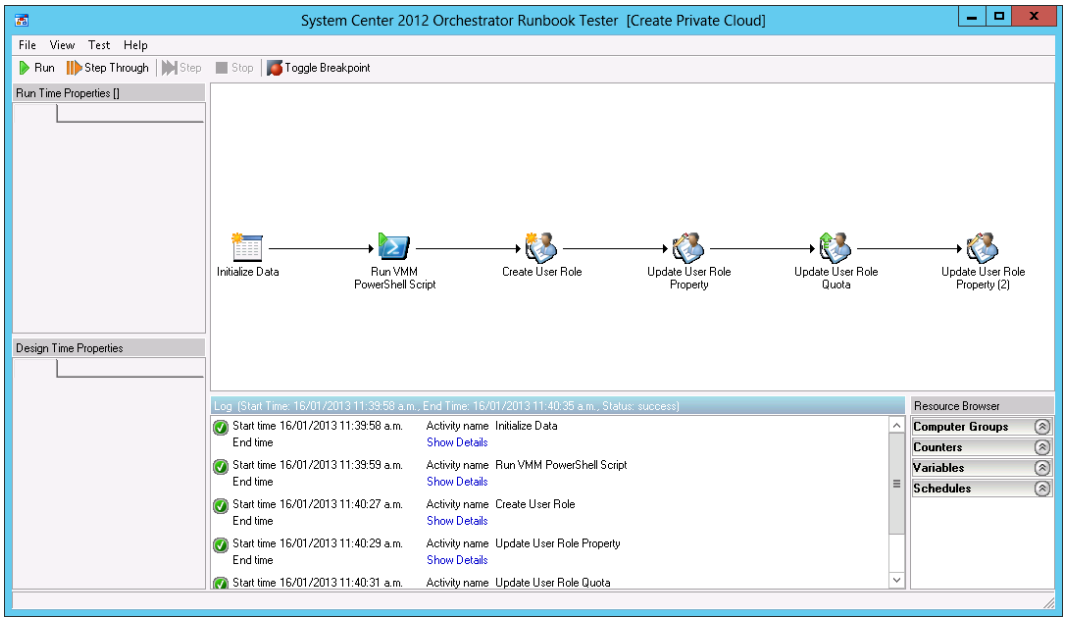
16GB of Memory

200GB of Storage

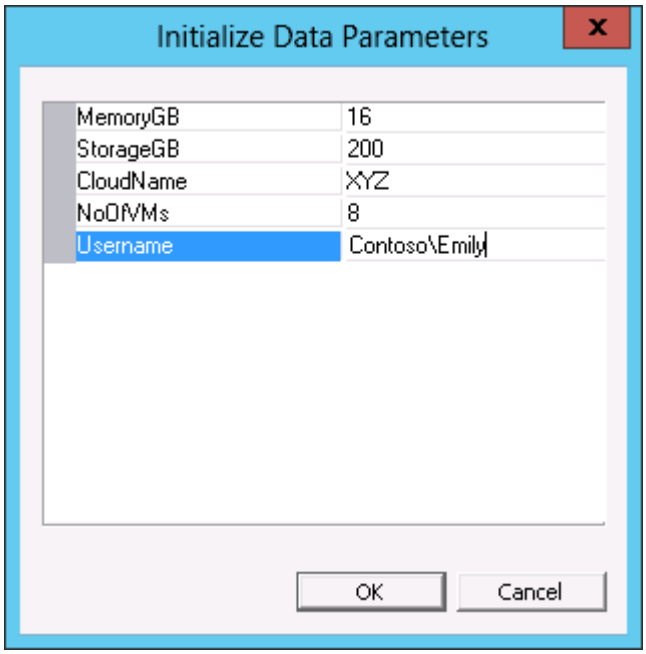
Allow a maximum of 8 virtual machines to be created in the cloud.

1. Select the **Runbook Tester** in the menu bar

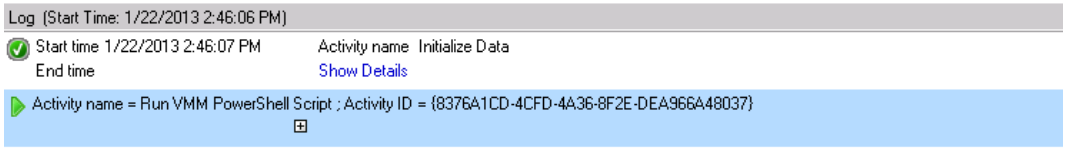
2. The **System Center 2012 Orchestrator Runbook Tester** window will now be presented, select the **Run** button to test the runbook.



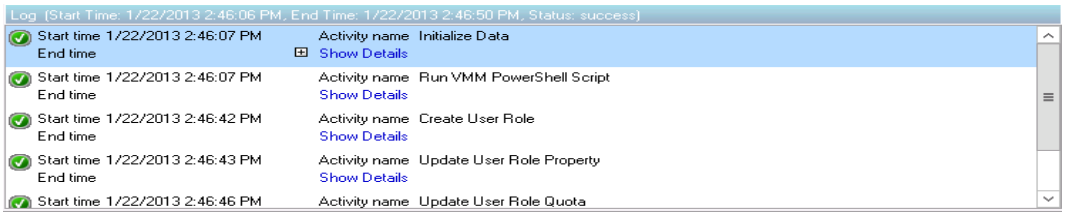
3. The **Initialize Data Parameters** dialog will be presented; here we will assign the **MemoryGB** as 16, **StorageGB** of 200, **CloudName** as XYZ, **NoOfVMs** as 8 and **Username** of **contoso\Emily**. Then select **OK**



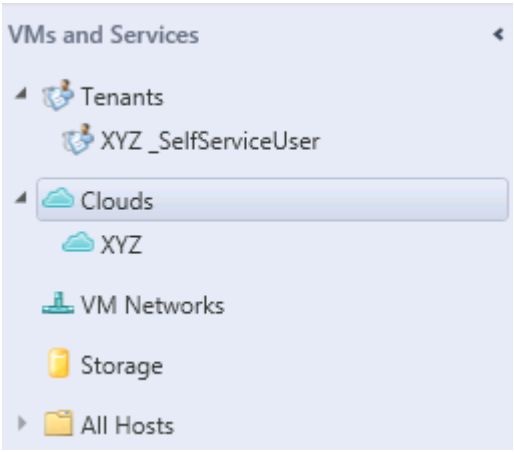
4. Using the **Log** pane, Jeff can observe the runbook activities which are currently active, or have completed. By selecting the **Show Details** link, or expanding the **+** icon, Jeff can view the results of each activity as it processes within the workflow.



5. When all activities are complete, Jeff will see that the job status now reads, "success"



6. Back in the **Virtual Machine Manager Console**, Jeff should now also see that the Cloud called **XYZ** cloud has been created.



Check In the Runbook

Prior to utilizing our completed runbook, it must be "checked in" again, using the Orchestrator Runbook Designer. If the runbook has not been checked in, it will not be replicated to Service Manager in the next chapter.

1. Open the **Runbook Designer**
2. Locate our **Runbook** which we named **Create Private Cloud**
3. Select the **Check In** button from the menu bar

Scenario: Simplified Self Service

Overview

In the previous scenario, Jeff created a runbook with the objective of optimizing his time not addressing repetitive tasks. Jeff read about the Self-Service potential of the private cloud and plans to embrace automation even further. Currently, Jeff must still start his new runbook and provide the private cloud details each time he receives a request ticket. Unfortunately, many times he simply only receives an email request, eager to automate the complete process, Jeff plans to present Emily and her team a simple web page to request the change, letting Service Manager and Orchestrator do the rest.

Prerequisites

Before Jeff begins to create the Self-Service portal, he will need to first source all the required components for the project. For his evaluation implementation, Jeff is going to use the **SharePoint** server to install all the required components. As SharePoint Foundation does not support Windows Server 2012, the **SharePoint** server will be installed with the **Windows Server 2008R2** operating system. In production, Jeff will follow the recommend guidelines, utilizing additional virtual machines to distribute the roles and service loads. The **Windows Server 2008 R2** evaluation can be downloaded from

http://www.microsoft.com/en-us/download/details.aspx?id=11093

The initial installation that Jeff needs is **SharePoint Foundation 2010**. The Self-Service portal is a collection of components that live on top of SharePoint. In production, he will embed those components on his production SharePoint farm, while for the evaluation; he is going to utilize the free SharePoint Foundation sever that he downloads from http://www.microsoft.com/download/en/details.aspx?id=5970

Prior to beginning the installation of **SharePoint Foundation 2010**, Jeff must download and install the following prerequisites specifically for **SharePoint**

SharePoint Foundation 2010 prerequisites:

Prerequisite	URL
Windows Identity Foundation:	http://www.microsoft.com/download/en/details.aspx?id=17331
Microsoft Sync Framework Runtime v1.0 (x64):	http://www.microsoft.com/download/en/details.aspx?id=15391
Microsoft Chart Controls for Microsoft .NET Framework 3.5:	http://www.microsoft.com/download/en/details.aspx?id=14422
Microsoft Filter Pack 2.0:	http://www.microsoft.com/download/en/details.aspx?id=17062
Microsoft Server Speech Platform Runtime (x64):	http://www.microsoft.com/download/en/details.aspx?id=16789

Microsoft Server Speech Recognition Language - TELE(en-US):	http://go.microsoft.com/fwlink/?LinkID=166371
SQL 2008 R2 Reporting Services SharePoint 2010 Add-in:	http://www.microsoft.com/download/en/details.aspx?id=622
Microsoft Sync Framework Runtime v1.0 (x64):	http://go.microsoft.com/fwlink/?LinkID=160382
Microsoft® SQL Server® 2008 R2 Native Client	http://www.microsoft.com/en-us/download/details.aspx?id=16978
Microsoft SQL Server 2008 Analysis Services ADOMMD.NET	http://go.microsoft.com/fwlink/?LinkID=160390
Microsoft .Net Framework 4.0	http://go.microsoft.com/fwlink/p/?LinkID=232304

Next, Jeff proceeds with the installation of **SharePoint Foundation 2010** as a **Standalone server** with the default settings

After the installation, he then runs the **SharePoint Configuration Wizard** (default settings)

Once Jeff has installed SharePoint Foundation 2010 he will need to download and install SP1 from the Microsoft website. This is to add compatibility for SQL 2012. It can be downloaded from the following location

http://www.microsoft.com/en-us/download/details.aspx?id=26640

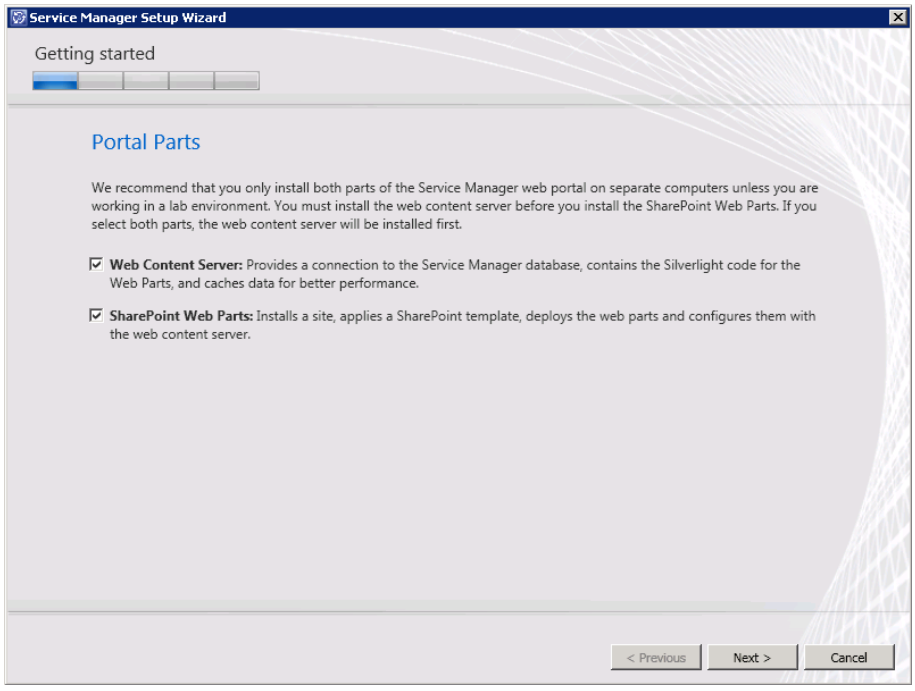
Installing the Self-Service Portal

Now that all the prerequisites have installed, Jeff can start installing the **System Center 2012 Service Manager web portal.**

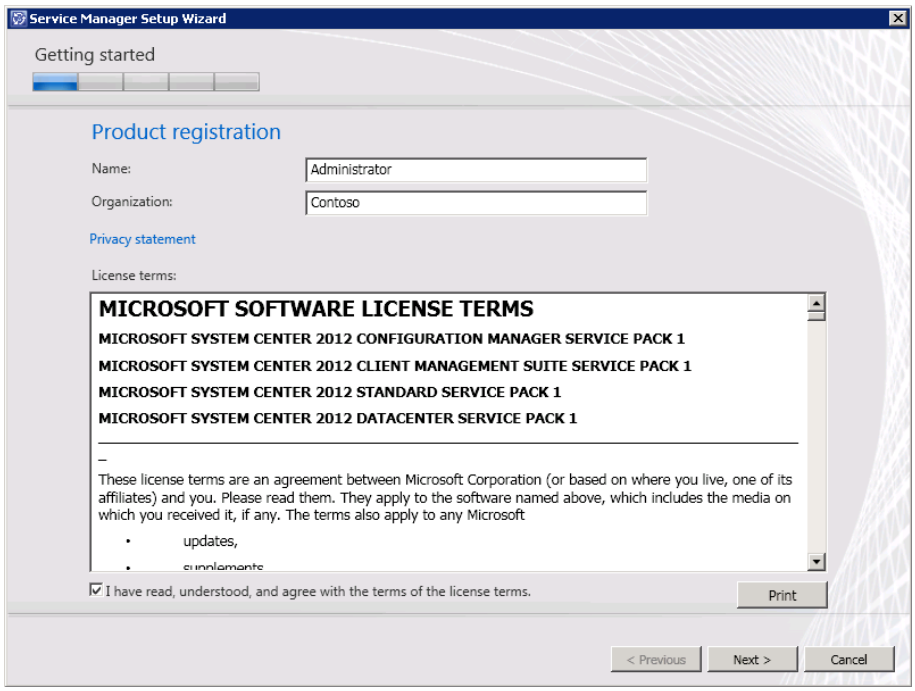
1. Still working on the **SharePoint** computer, Run **setup.exe** from the **System Center Service Manager** installation media

2. On the **Service Manager** splash screen, select **Service Manager web portal**

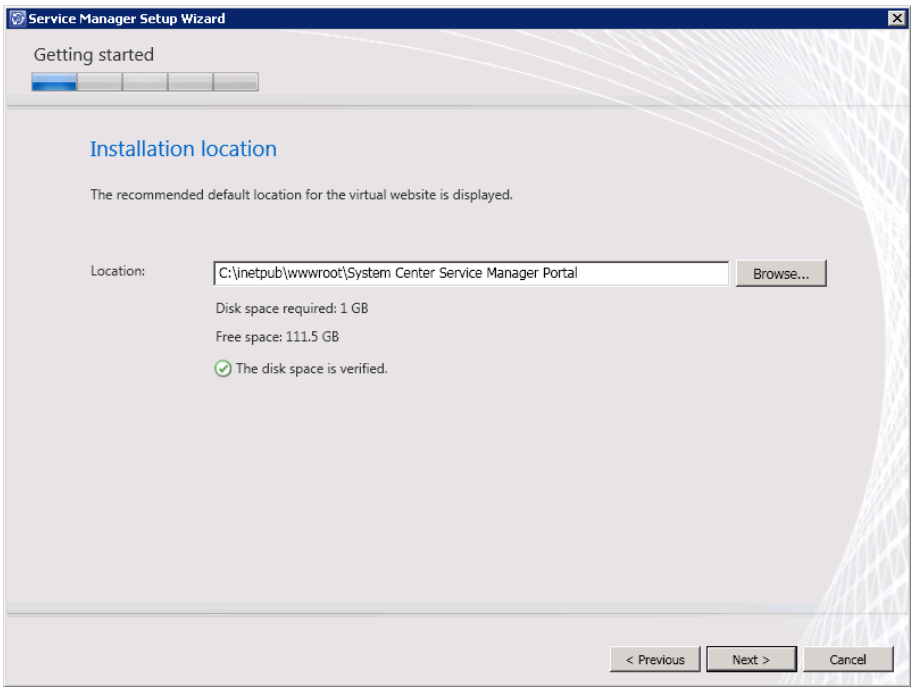
3. On the **Portal Parts** screen, select both the **Web Content Server** and **SharePoint Web Parts** and select **Next**



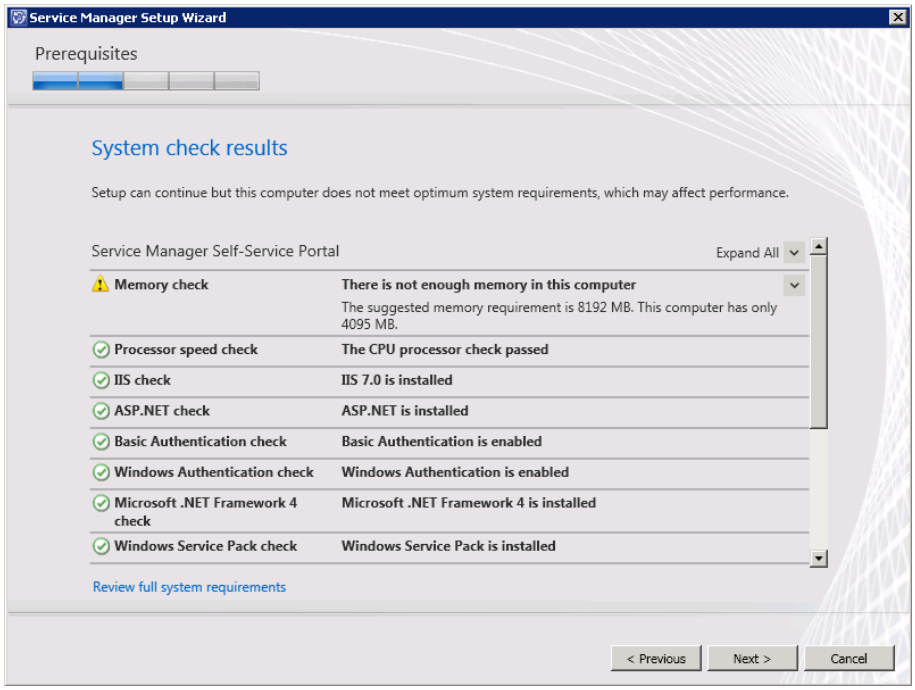
4. On the **Product registration** page, fill in a **Name** and **Organization**, select **I have read, understood, and agree with the terms of the license terms** and select **Next**



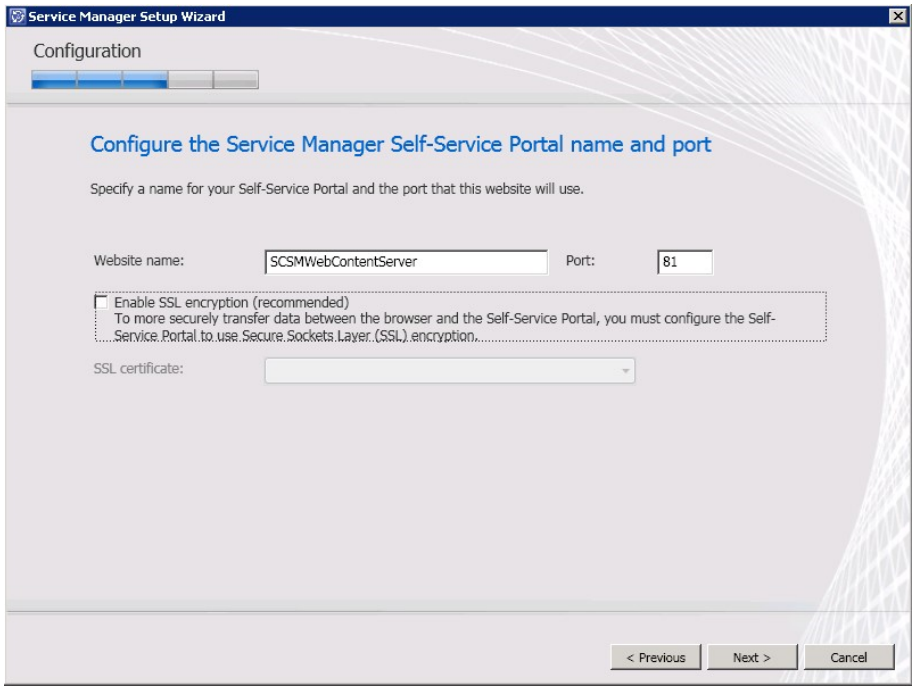
5. On the **Installation location** page, change the installation location or leave it default and select **Next**



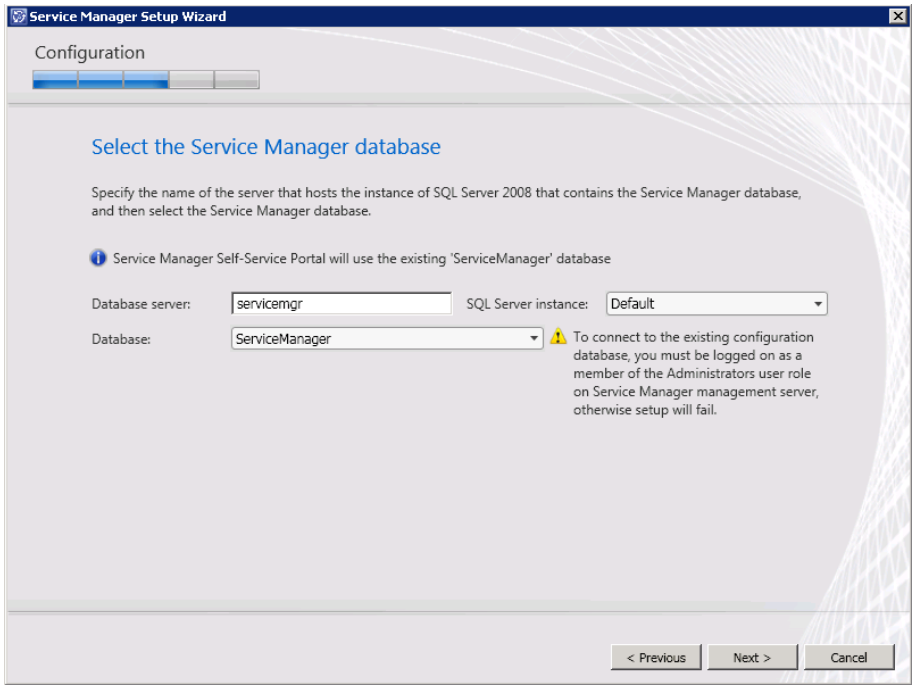
6. On the **System check results**, when you have passed the system requirements, select **Next**



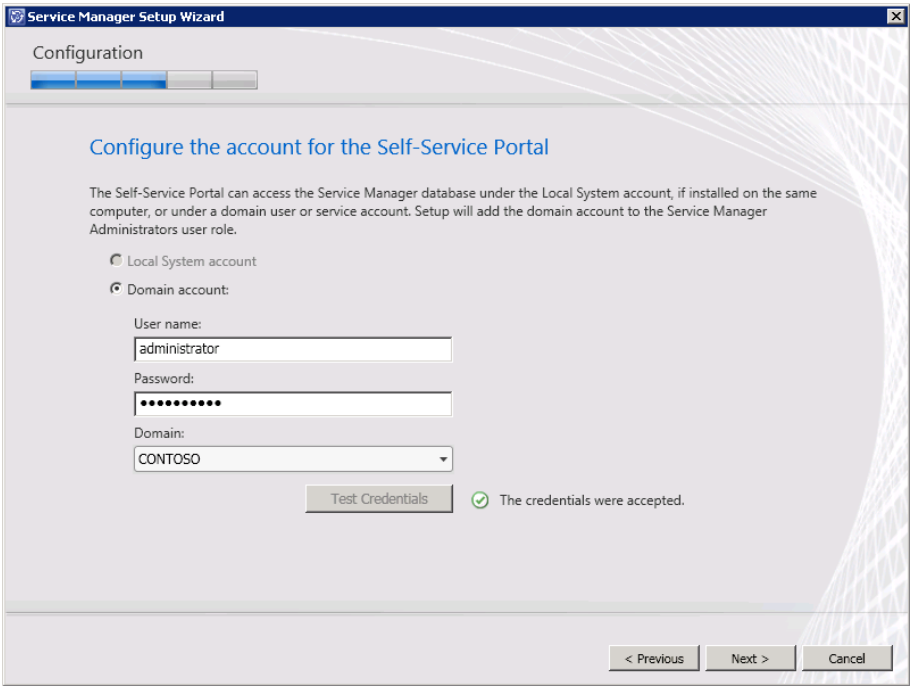
7. On the Configure the Service Manager Self-Service Portal name and port, uncheck the **Enable SSL encryption** and make sure that the **Port** is 81. Select **Next** to continue



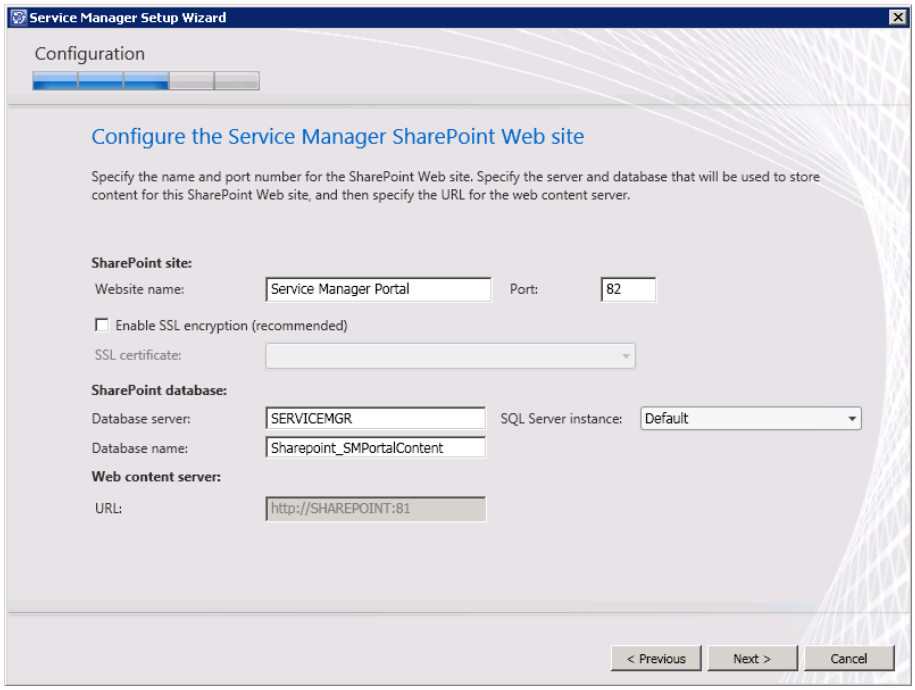
8. On the **Select the Service Manager database** page, in the **Database server** text field enter **servicemgr**, and set the **SQL Server instance** dropdown to **Default.** Finally, in the **Database** dropdown select **ServiceManager**. Then select **Next**



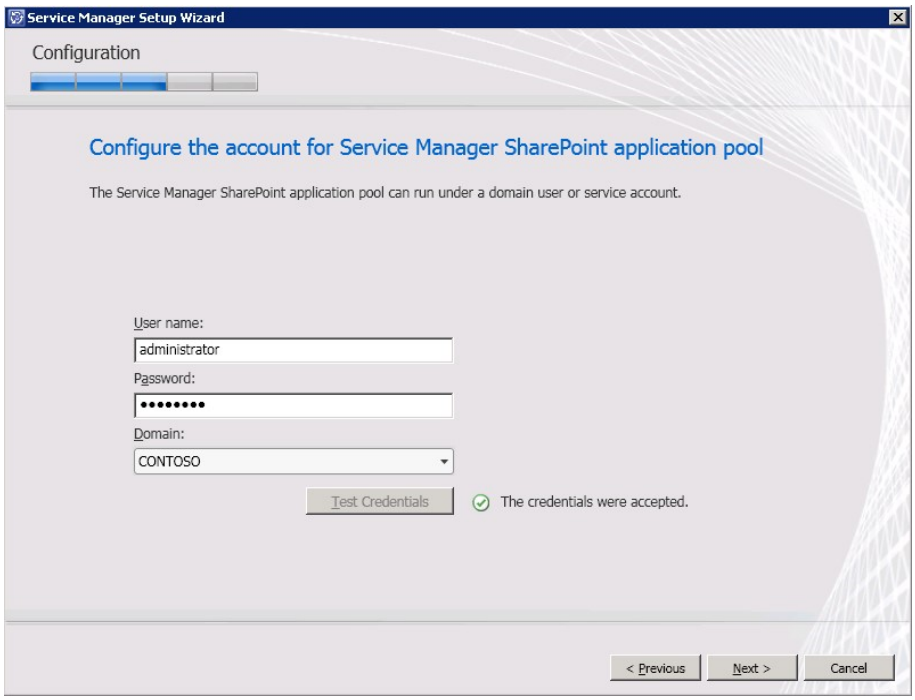
9. On the **Configure the account for the Self-Service Portal** page, select the **Domain account** option; provide the **User name** as **administrator**, provide the relevant **password**, and then set the **domain** to **contoso**. Select **Test the credentials**. When the test succeeds, select **Next**



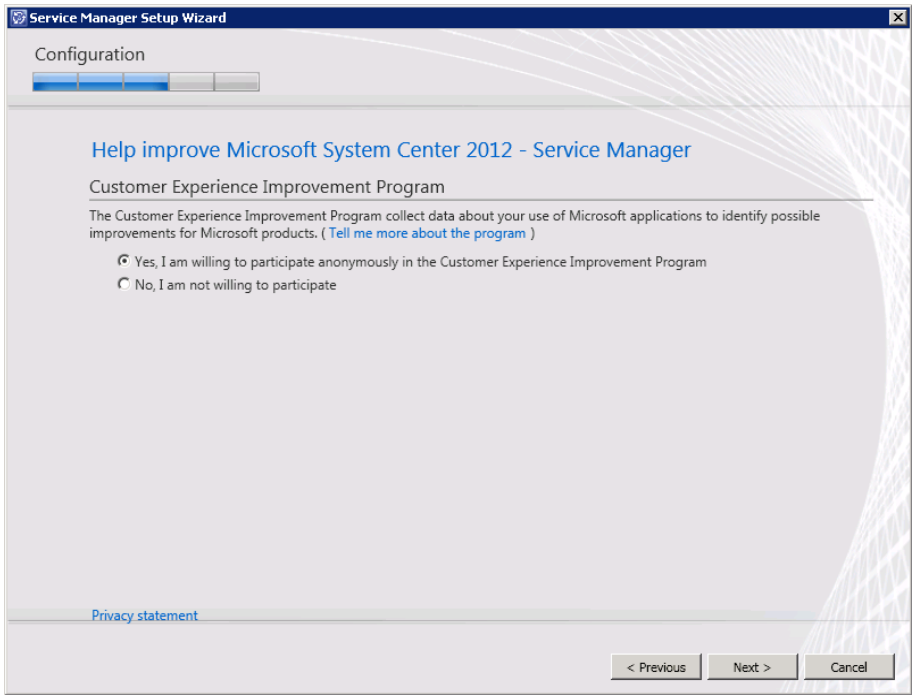
10. On the **Configure the Service Manager SharePoint Web site**, uncheck the **Enable SSL encryption** and set the port to **82**. Set the **Database server** as **ServiceMgr**, the **SQL Server instance** drop down to **Default**, and the **Database name** to read **SharePoint_SMPortalContent**. Then select **Next**



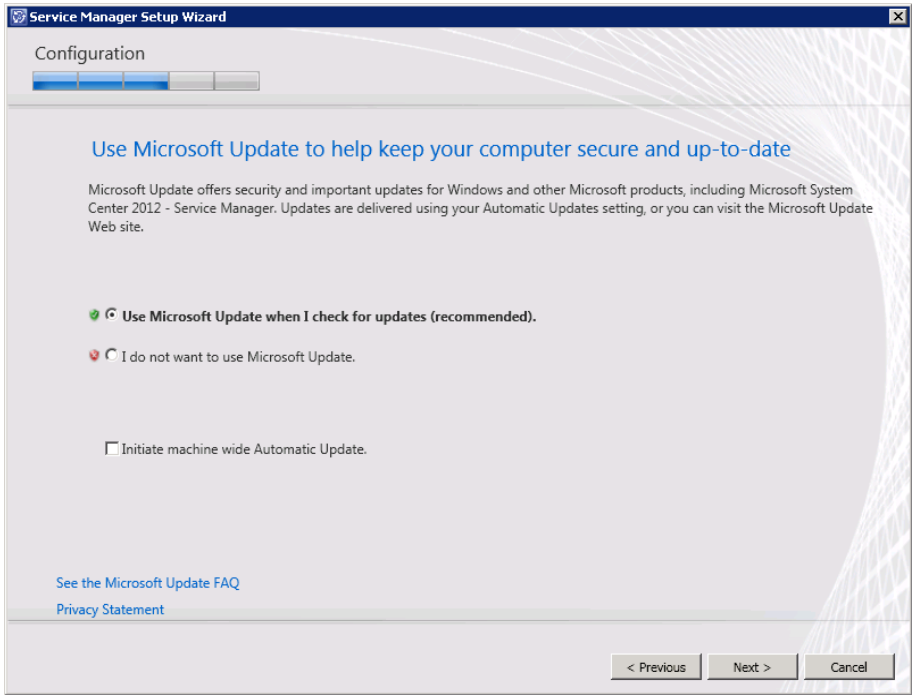
11. On the **Configure the account for Service Manager SharePoint application pool**, provide the **User name** as **administrator**, provide the relevant **password**, and then set the **domain** to **contoso**. Select **Test the** credentials and when the test succeeds, select **Next**



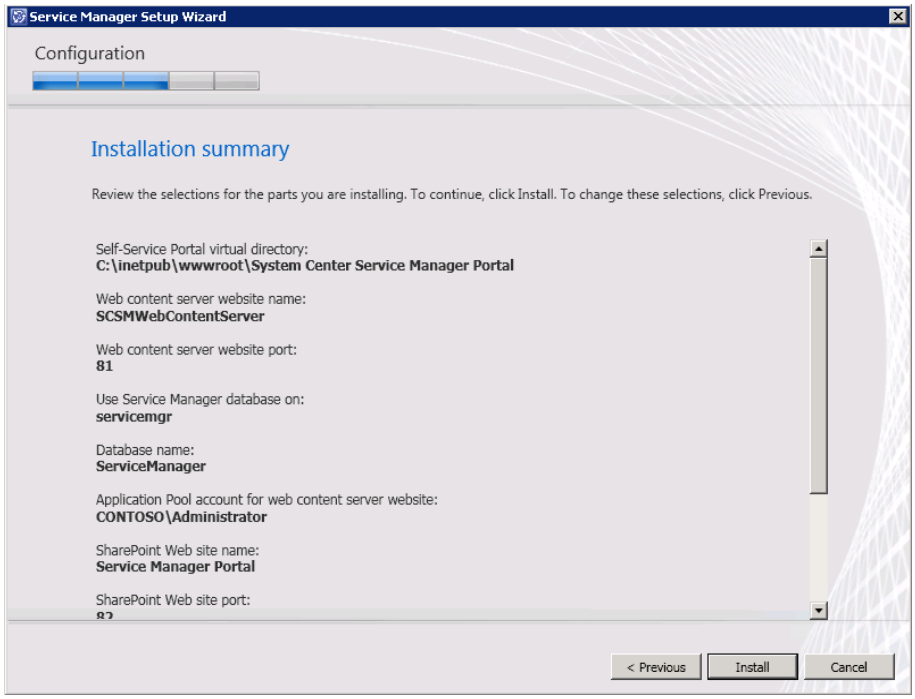
12. On the **Help improve Microsoft System Center 2012 – Service Manager** choose if you want to participate anonymously in the **Customer Experience Improvement Program** or not, then select **Next**



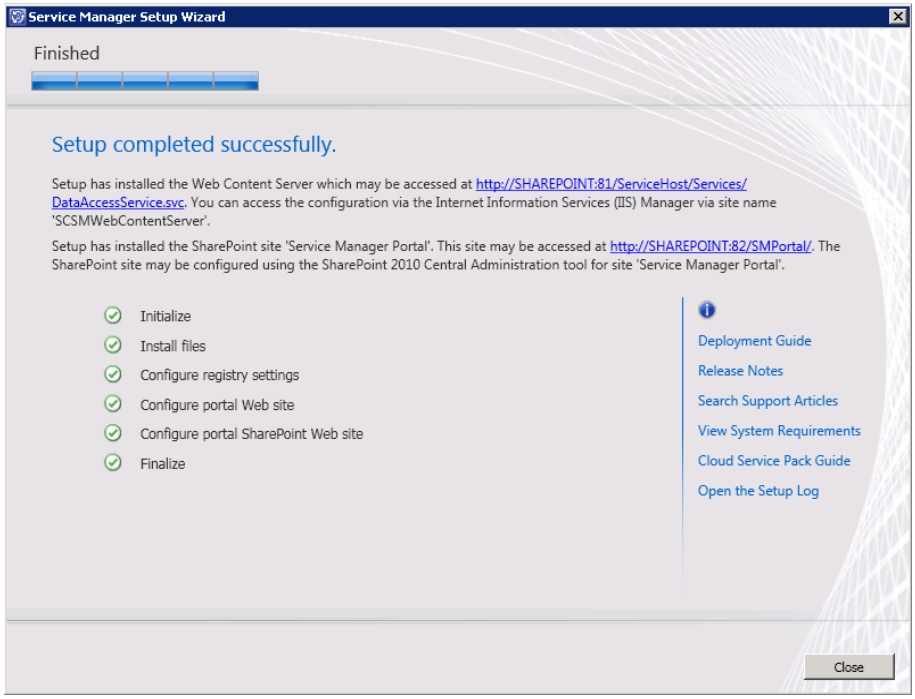
13. On the **Use Microsoft Update to help keep your computer secure and up-to-date** page, select if you want to use **Microsoft Update**, and select **Next**



14. On the **Installation summary**, review your settings and select **Install**



15. On the **Setup completed successfully** page, write down the **URL (http://sharepoint:82/SMPortal)** of your newly created portal as presented in the results text. Finally you may select **Close** to complete the exercise.



Creating a Runbook

Jeff now considers the steps which he needs to repeat in order to deliver a service to his end-users, some of these will not require to be repeated each time, so the list will act as a guide for him as he proceeds to publishing his new service offerings on the portal.

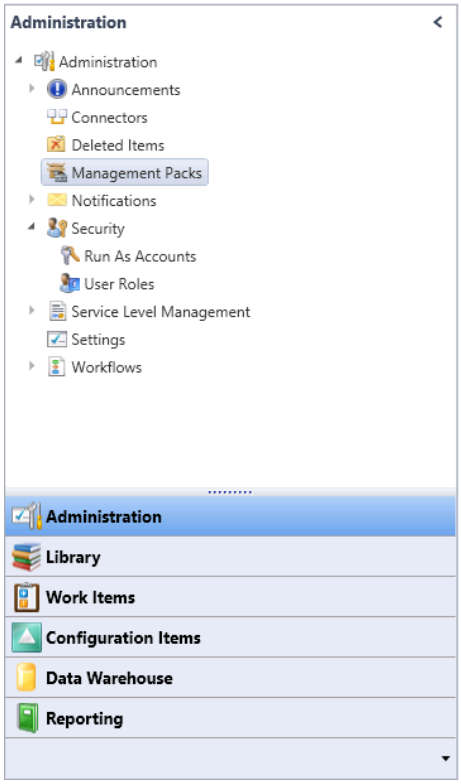
- Pre-Work
- Create a Runbook in Orchestrator (already covered in this chapter)
- Create the Orchestrator Connector in Service Manager (already covered in this chapter)
- Synchronize Orchestrator with Service Manager
- Create a Runbook Automation Activity template
- Create a Service Request template
- Create a Service Offering
- Create a Request Offering
- Add the Request Offering to the Service Offering

As this is the first time Jeff has encountered this new process, he decides that he will attempt to complete each of these steps now so he understands the process, and later when he is ready to offer additional services he can opt to exclude the optional steps if he so wishes.

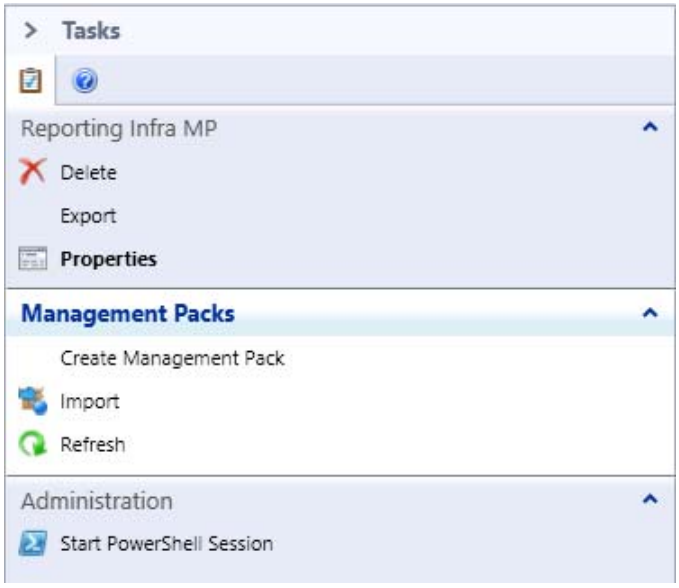
Pre-Work

Prior to Jeff beginning the work of offering new services, he first will create a Management Pack. This management pack will be used to store all the customizations which he will create as he implements his offerings. An advantage of this approach is that he can undo his changes by deleting the management pack, but also he can export the management pack from his evaluation environment, and simply import it onto another environment, for example Production at any time in the future. This enables Jeff to have a good test first approach for delivering new services.

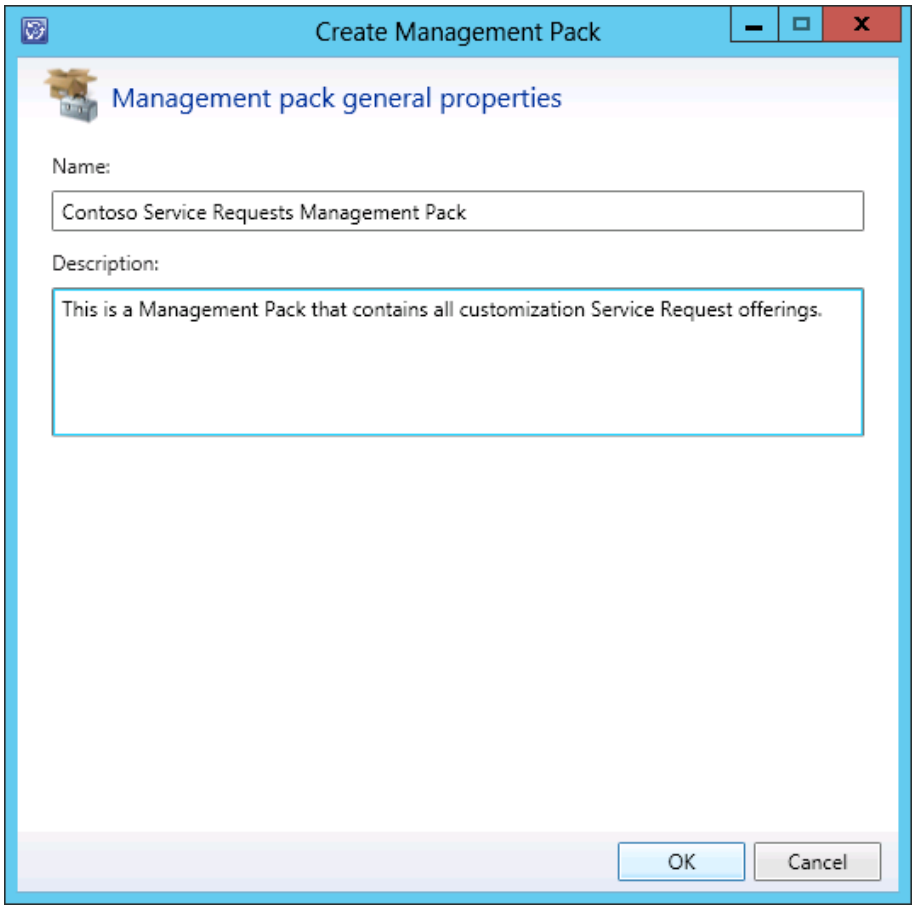
1. In the **System Center 2012 Service Manager console**, with the **Administration** workspace selected, expand the navigation tree to select **Administration** > **Management Packs**



2. The context of the **Tasks** Pane will update, and now we can choose **Create Management Pack**



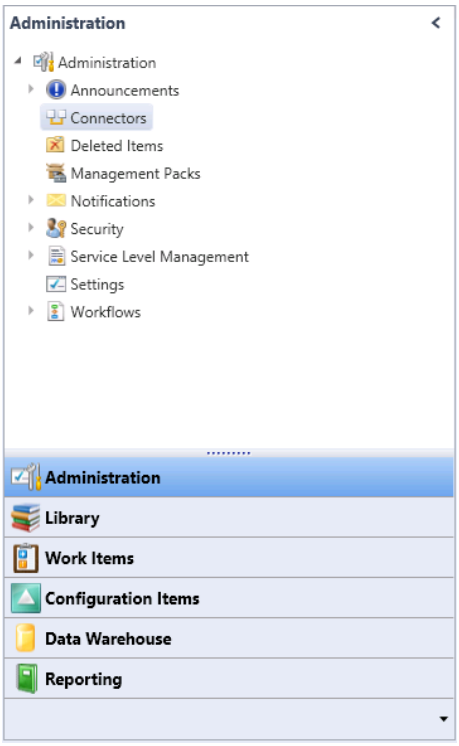
3. On the **Create Management Pack** dialog, in the **Name** text box enter **CONTOSO Service Requests Management Pack,** and in the **Description** text box enter: **This is a Management Pack that contains all customization Service Request offerings**. Then select **OK**



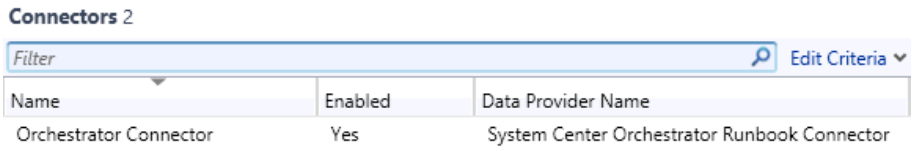
Synchronize Orchestrator with Service Manager

As we created a runbook in the previous chapter he must now synchronize this new information with Service Manager, otherwise the new runbook will be unavailable for us to utilize in Service Manager.

1. In the **System Center 2012 Service Manager** console, with the **Administration** workspace selected, expand the navigation tree to select **Administration** > **Connectors**.

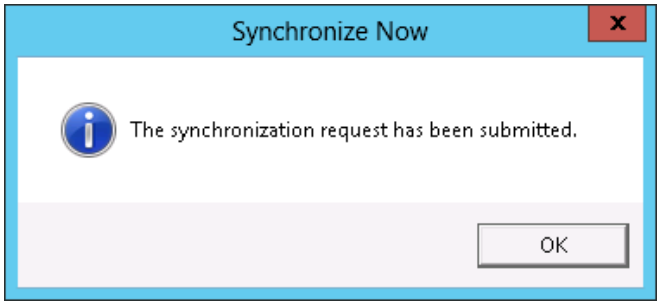


2. Select the previously created **Orchestrator Connector** connection



3. The context of the **Tasks** pane will update, and now we can choose **Synchronize Now**

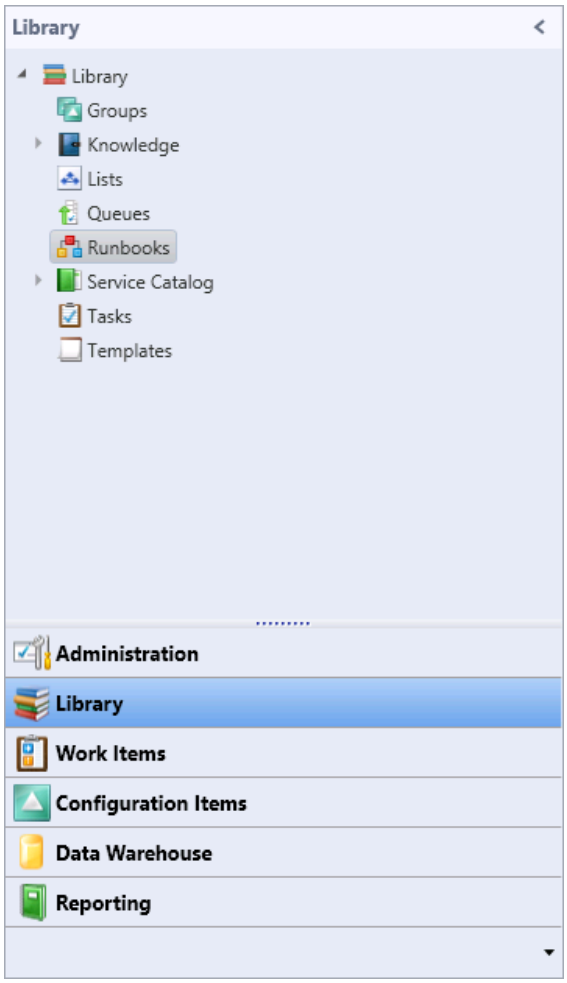
4. Select **OK** on the **Synchronize Now** page and then refresh the page until the synchronization has finished.



Create Runbook Automation Activity Template

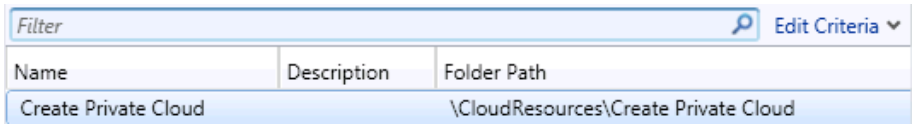
Next, Jeff needs to prepare **Service Manager** to offer the runbook that he created in Chapter 9 as available for use in his templates. Once this is complete Jeff can then utilize the runbook in any of his service templates, including requests and incidents. Jeff also has the option to take advantage of the template to define some parameters now, so they do not need to be redefined every time the runbook is required.

1. In the **System Center 2012 Service Manager** console, with the **Library** workspace selected, expand the navigation tree to select **Runbooks**.



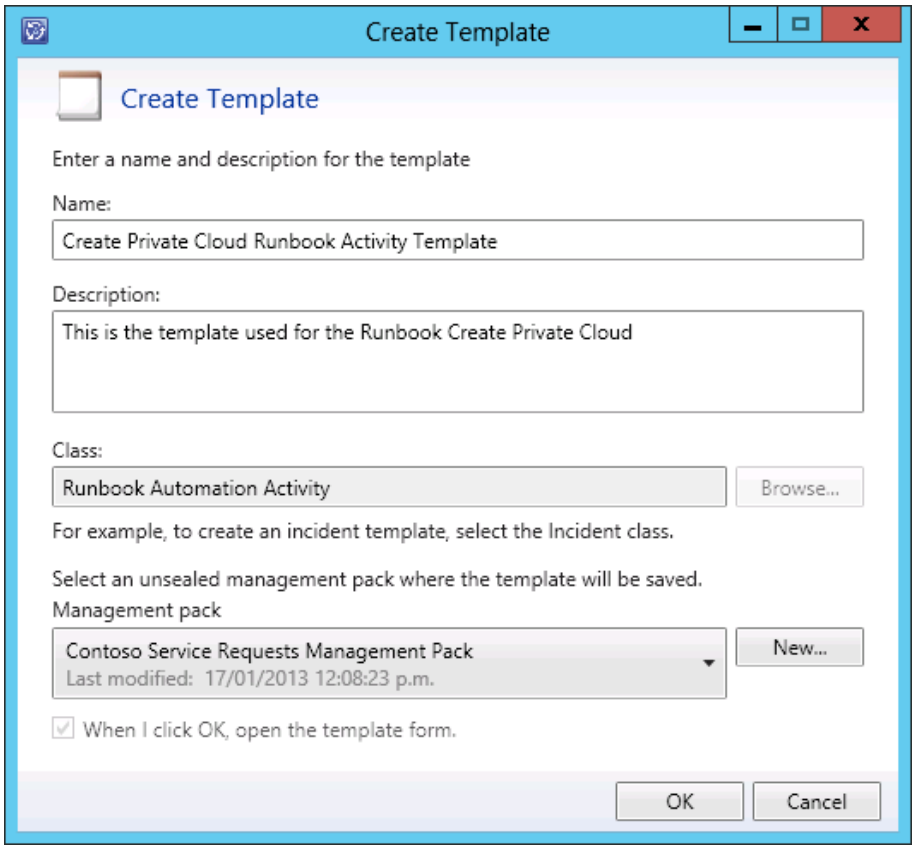
2. Select the Runbook from the **Runbooks** pane that you created in Chapter 9 called **Create Private Cloud**
    - *If you don't see the **Create Private Cloud** runbook in the **Runbooks** pane, then run the following SQL statement on the Orchestrator server:*

```
USE Orchestrator
TRUNCATE TABLE
[Microsoft.SystemCenter.Orchestrator.Internal].AuthorizationCache
```

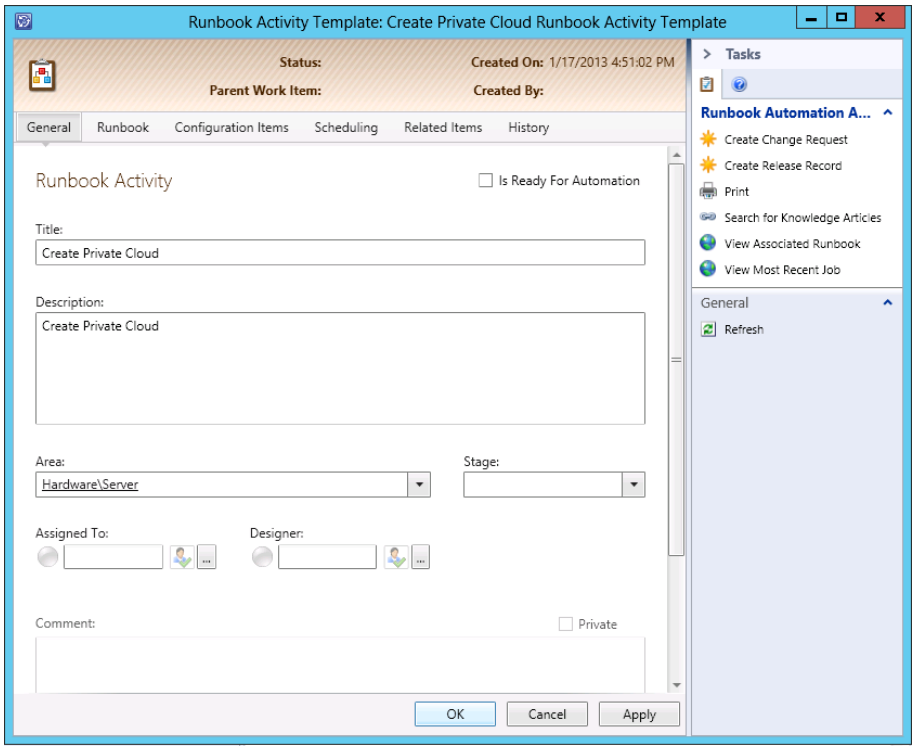


3. The context of the **Tasks** pane will update, and now we can choose **Create Runbook Automation Activity Template**.

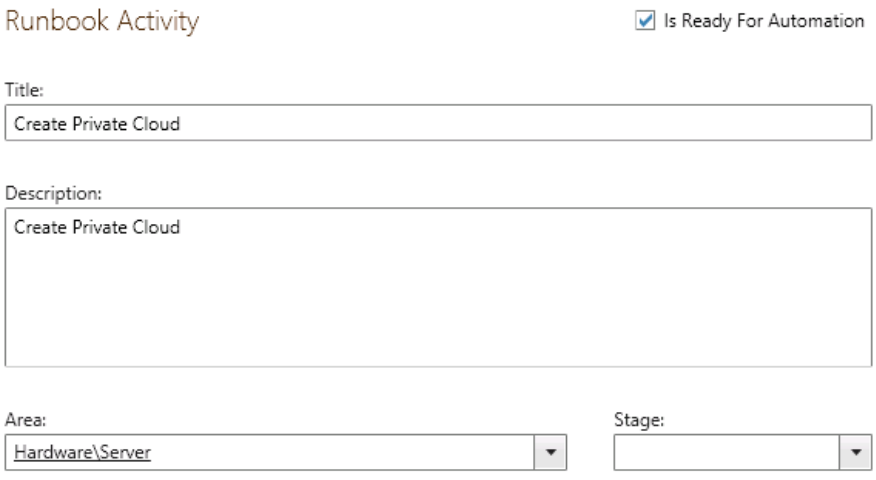
4. Fill in the following data on the **Create Template** page. When done, select **OK** to open the template editor
    - **Name**: Create Private Cloud Runbook Activity Template
    - **Description**: This is the template used for the Runbook Create Private Cloud
    - **Class**: Runbook Automation Activity
    - **Management Pack**: CONTOSO Service Requests Management Pack



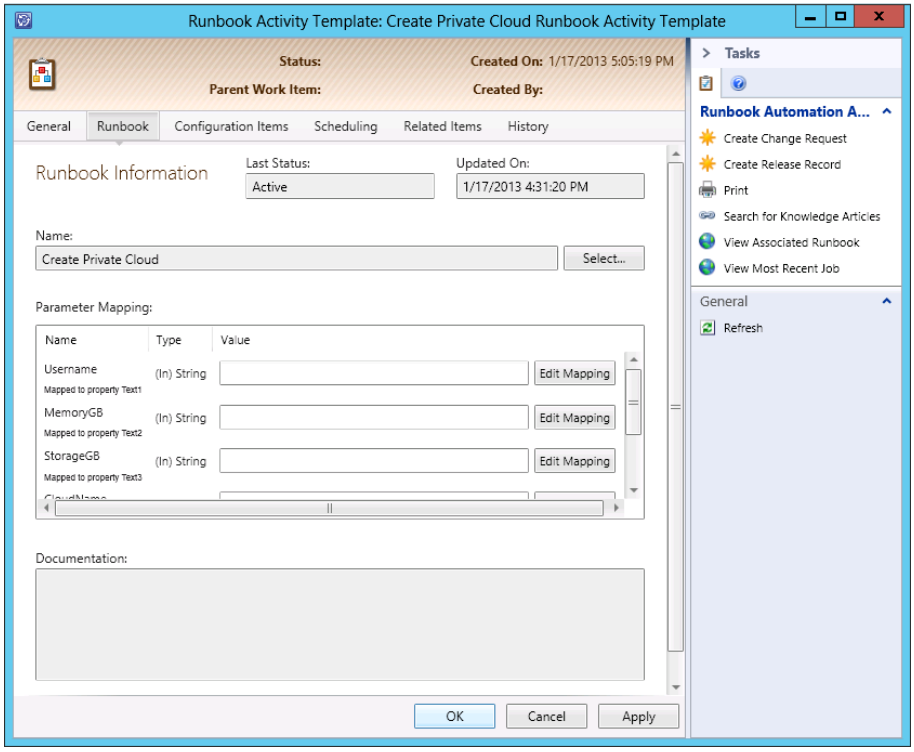
5. Now you can change the parameters for the runbook activity. Those parameters will be automatically filled in when a new activity is created in **Service Manager.**
    - **Title**: Create Private Cloud
    - **Description**: Create Private Cloud
    - **Area**: Hardware\Server



6. Ensure the option **Is Ready for Automation** box is checked; otherwise you cannot use this runbook as an automated activity. Finally close and save the dialog changes, click **Apply**



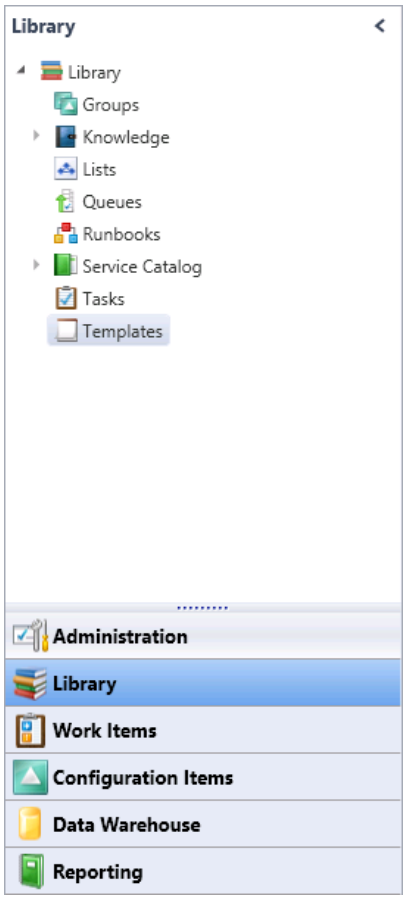
7. Click the **Runbook** tab, here we define the Runbook that is associated with this Activity Template. Select **SELECT**, make sure **Create Private Cloud** is highlighted then click on **OK**. Click on **OK** when returned to the **Runbook Activity Template** form.



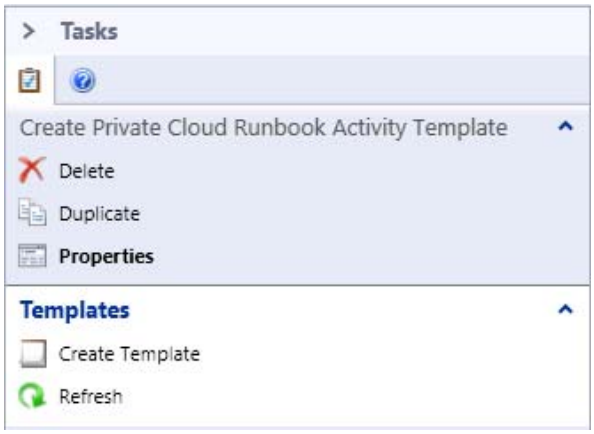
Create Service Request Template

When the end user connects to the Self Service portal to submit a new request, Service Manager is presenting a Service Request offering; these offering are based on the Service Request Template, and can be customized to auto-populate some of the fields in the request. Additionally, the Service Request Template allows Jeff to define which workflow will be carried out when this request offering is implemented. In this example the main activity Jeff is concerned with is his recently created runbook activity.

1. In the **System Center 2012 Service Manager** console, with the **Library** workspace selected, expand the navigation tree to select **Templates**



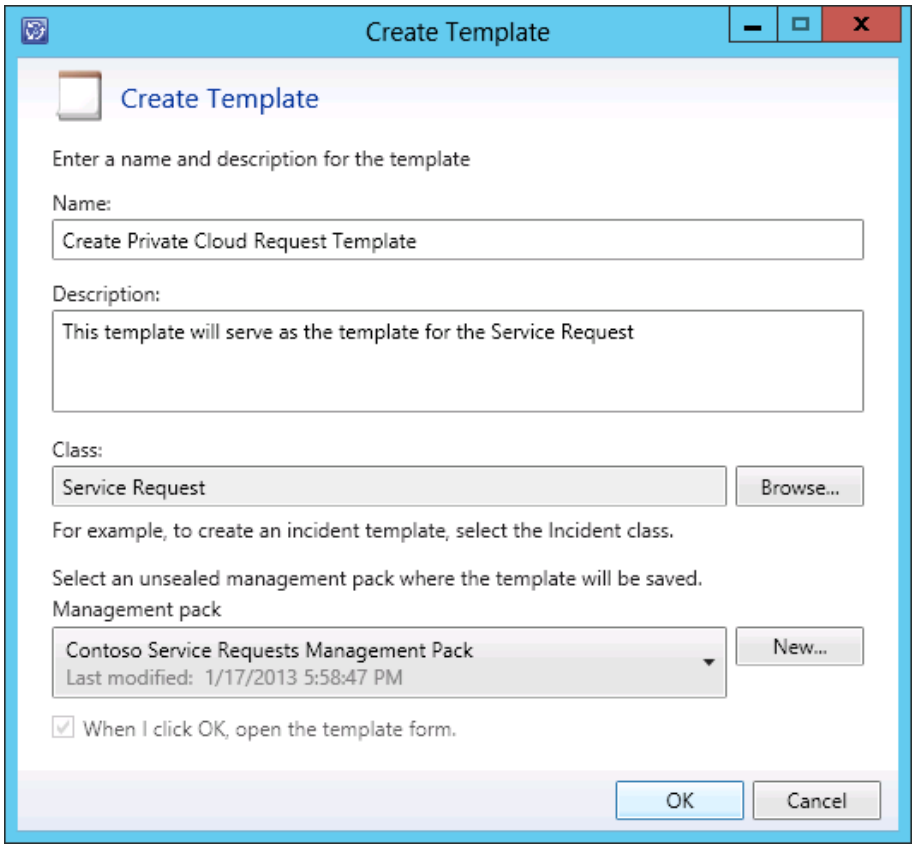
2. The context of the **Tasks** pane will update, and now we can choose **Create Template**.



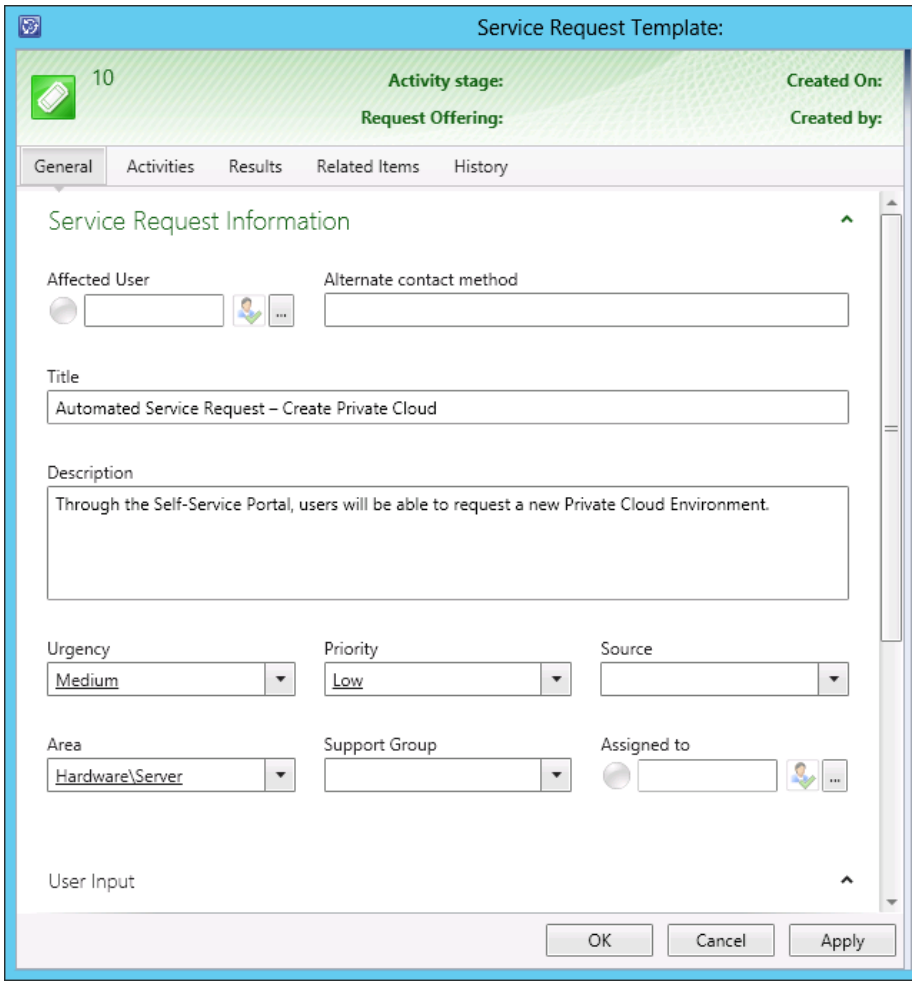
3. On the **Create Template** page fill in the following data:
    - **Name**: Create Private Cloud Request Template

- **Description**: This template will serve as the template for the Service Request
- **Class**: Service Request

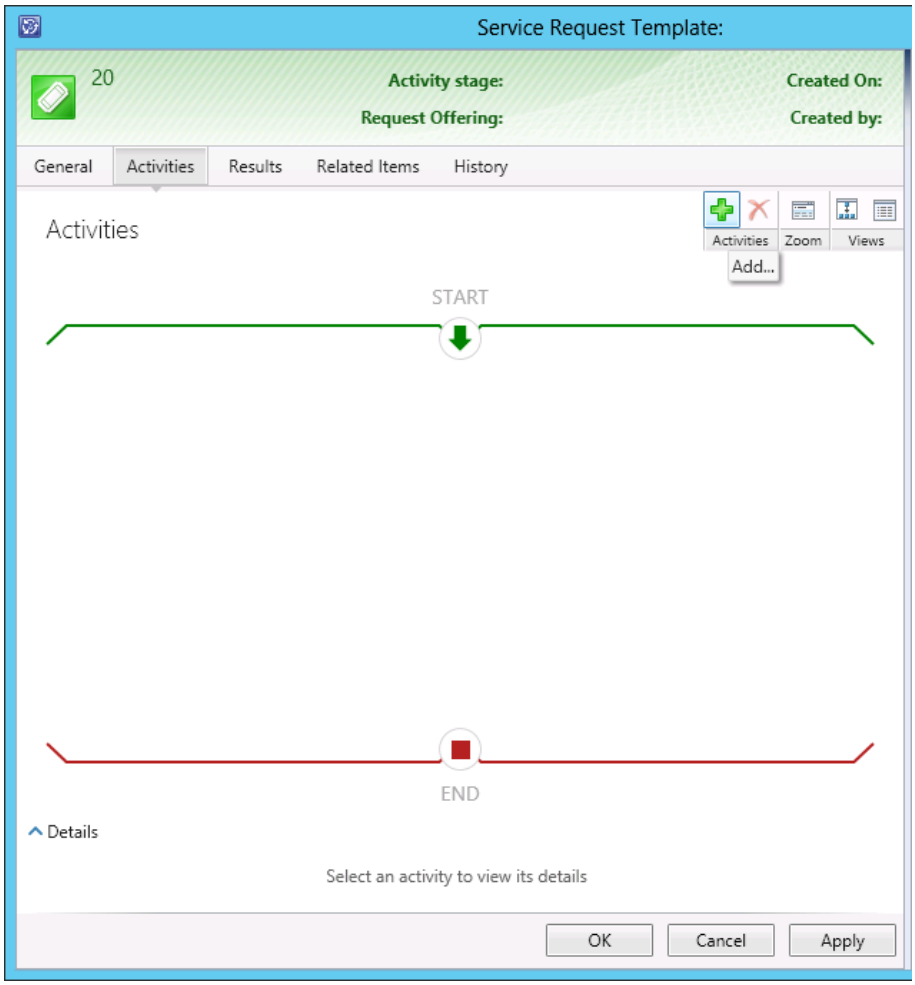
Our management pack called **CONTOSO Service Requests** should still be selected, and we can Select **OK** to open the Template Editor



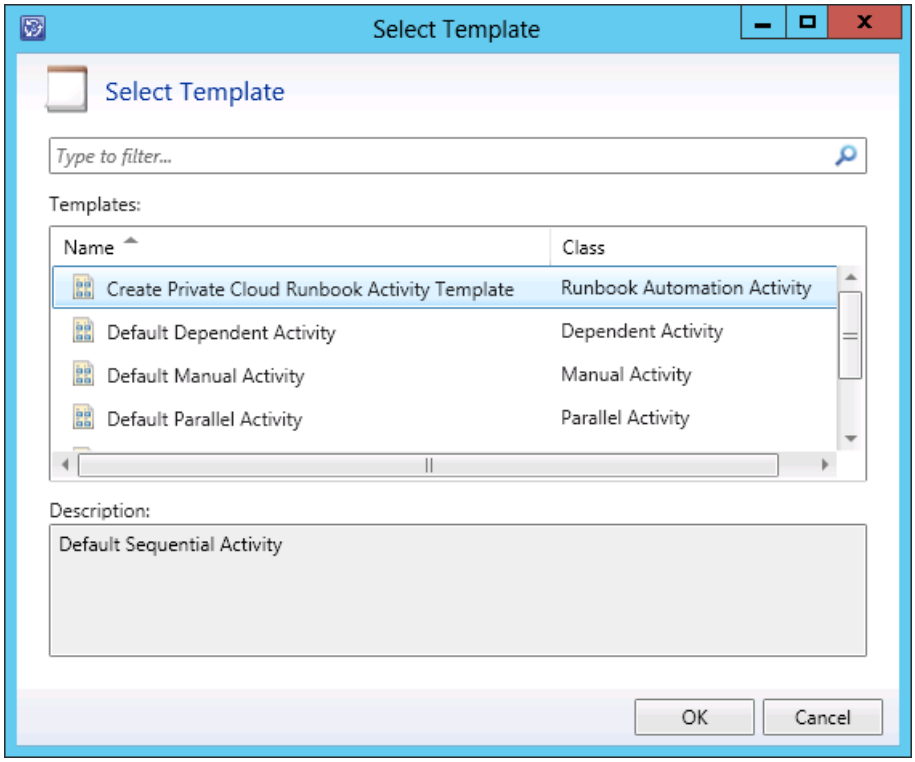
4. Once the form is presented, we will be placed on the **General** tab, where we can customize the parameters which we would like to be automatically filled for each new service request created in **Service Manager** based on this template
    - **Title**: Automated Service Request – Create Private Cloud
    - **Description**: Through the Self-Service Portal, users will be able to request a new Private Cloud Environment.
    - **Urgency**: Medium
    - **Priority**: Low
    - **Area**: Hardware\Server



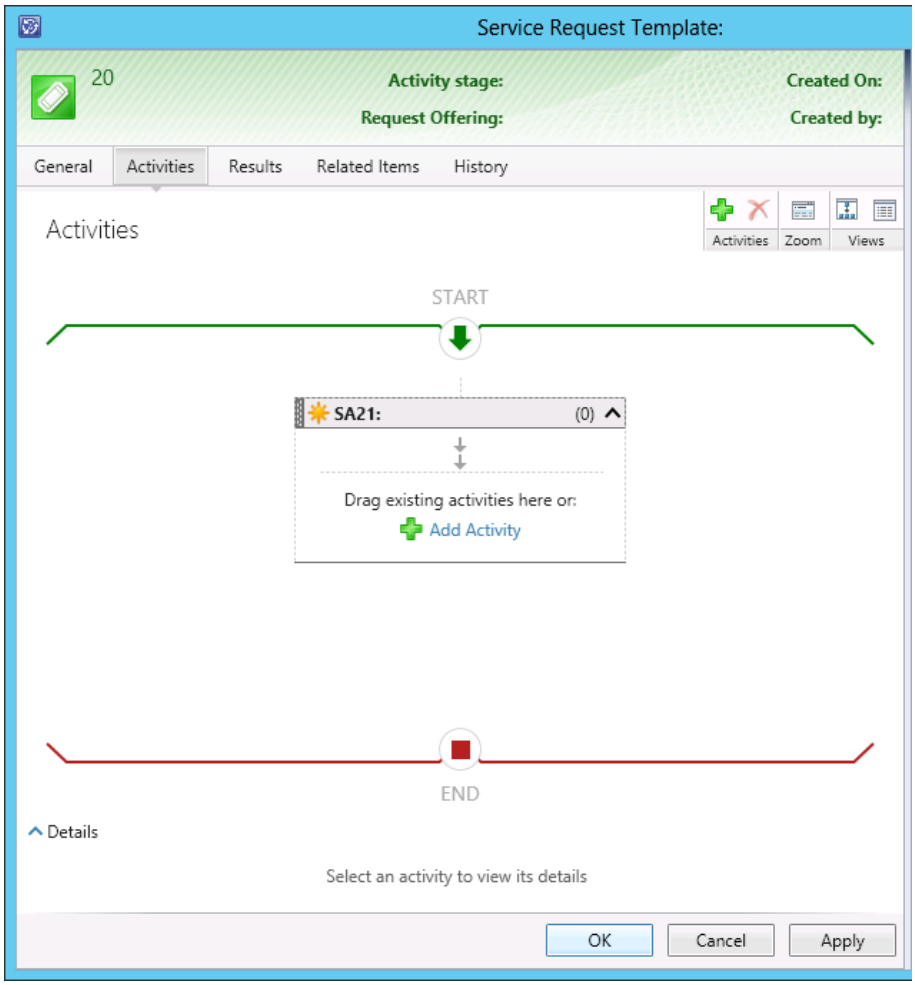
5. We can then proceed on to the **Activities** tab. Here we can define the workflow of activities which should be executed for this service request. Examples here may be a Review Activity when we would first prefer a manager to approve the request prior to moving on with the next step in the workflow. Jeff has decided that, for his initial validation, to add the Runbook Activity he created earlier so as to understand how this really works. He proceeds by selecting the  button.



6. In the **Select Template** dialog, choose the **Runbook Automation Activity** that we created a little earlier and called **Create Private Cloud Runbook Activity**. Then select **OK**



7. As Jeff does not plan to add any additional activities for now, just select **Apply** and **OK** to complete creating the Service Request template



Create a Service Offering category

Now that Jeff has created both his templates, he can finally proceed to create the offerings that will appear on the Self-Service portal. Jeff has browsed through some information about this process and learned that there are two types of offerings.

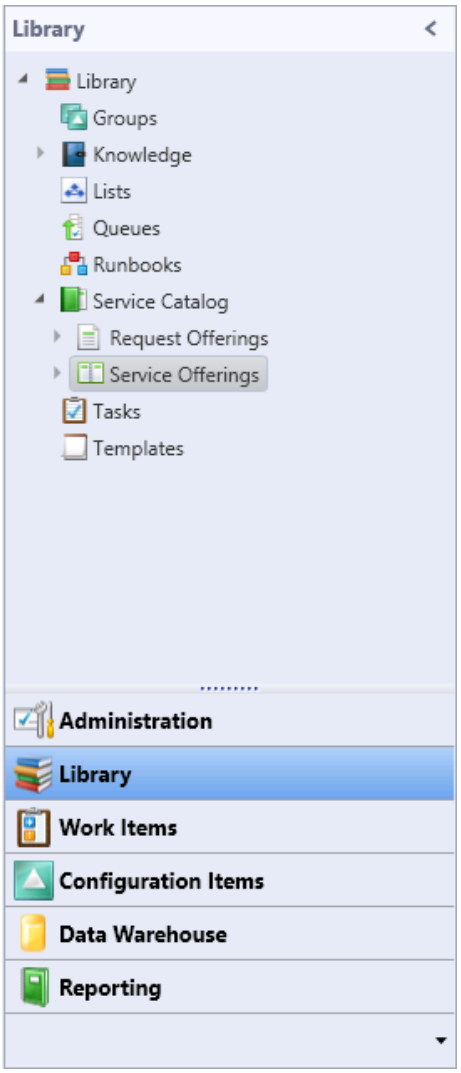
A) **Service Offering**, these are really nothing more than a collection of different **Request Offerings**.

B) **Request Offerings** are the actual offering which end-users are presented, and are normally customized with specific questions relevant to the request.

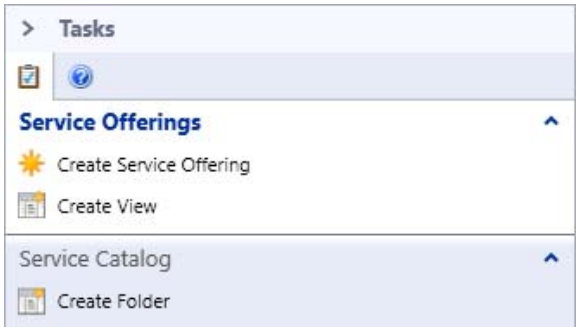
Create a Service Offering

In this example Jeff will be creating the offerings called **Emily's Cloud Offerings**. Since a **Service Offering** is a collection of different **Request Offerings**, Jeff will add all offerings specific for Emily's Cloud under this Service Offering.

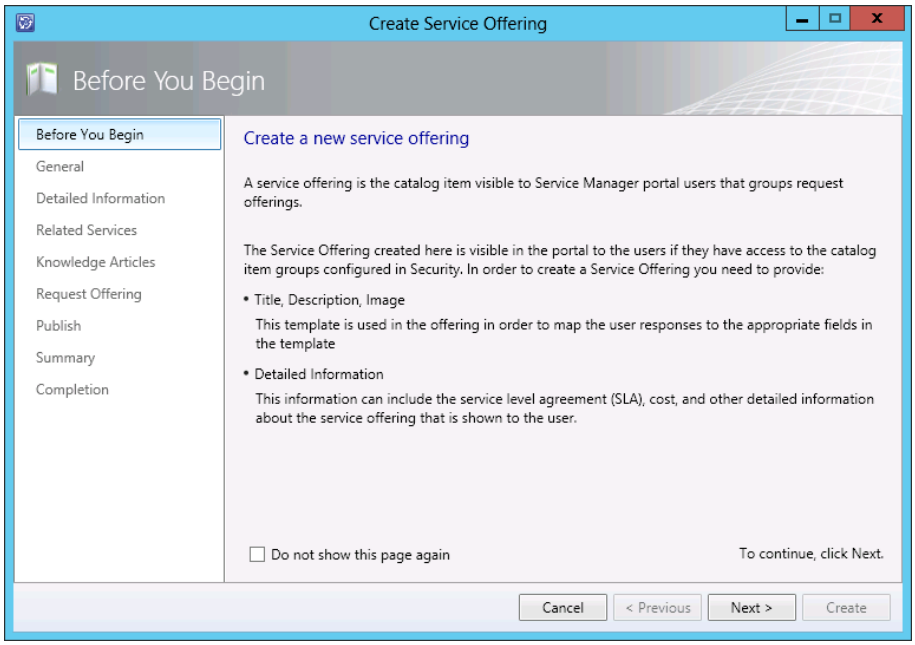
1. In the **System Center 2012 Service Manager** console, with the **Library** workspace selected, expand the navigation tree to select **Service Catalog** > **Service Offerings**.



2. The context of the **Tasks** pane will update, and now we can choose **Create Service Offering**

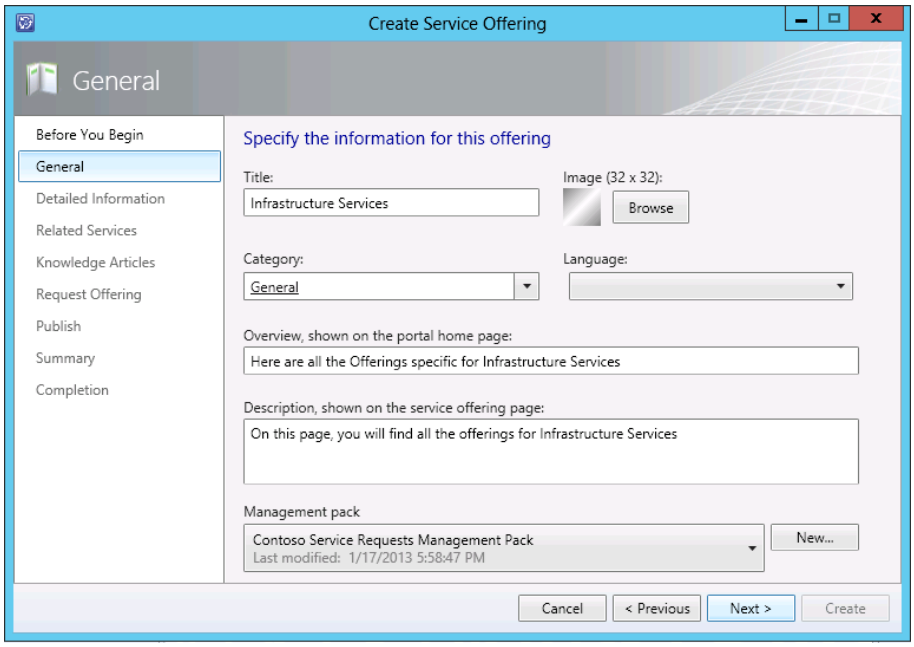


3. The **Create a new service offering wizard** will be presented, select **Next** to begin

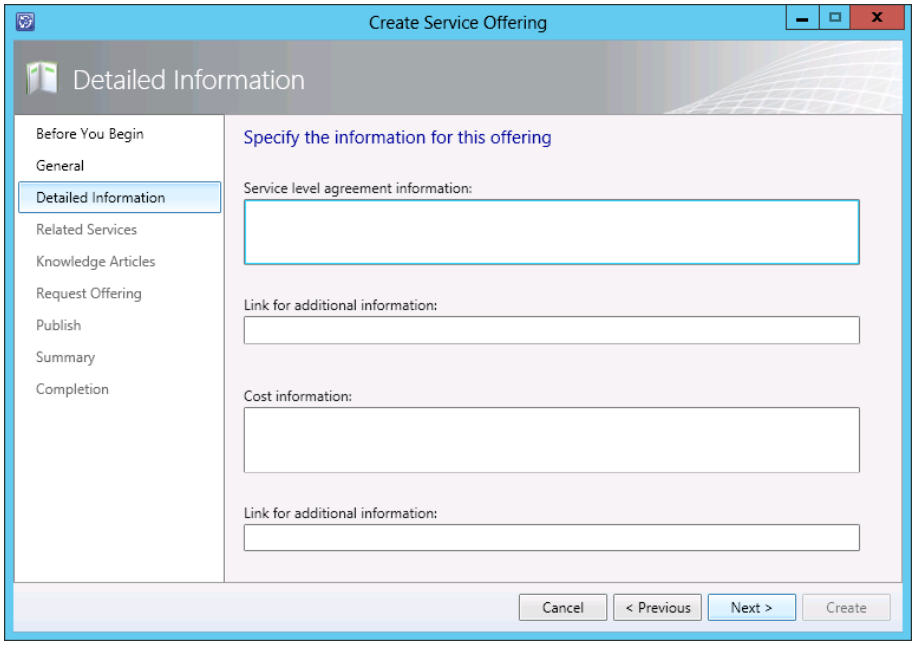


4. On the **Specify the information for this offering** page fill in or select the following information:
    - **Title:** Infrastructure Services
    - **Category:** General
    - **Language:** Leave empty!
    - **Overview:** Here are all the Offerings specific for Infrastructure Services
    - **Description:** On this page, you will find all the offerings for Infrastructure Services
    - **Management Pack:** CONTOSO Service Requests Management Pack

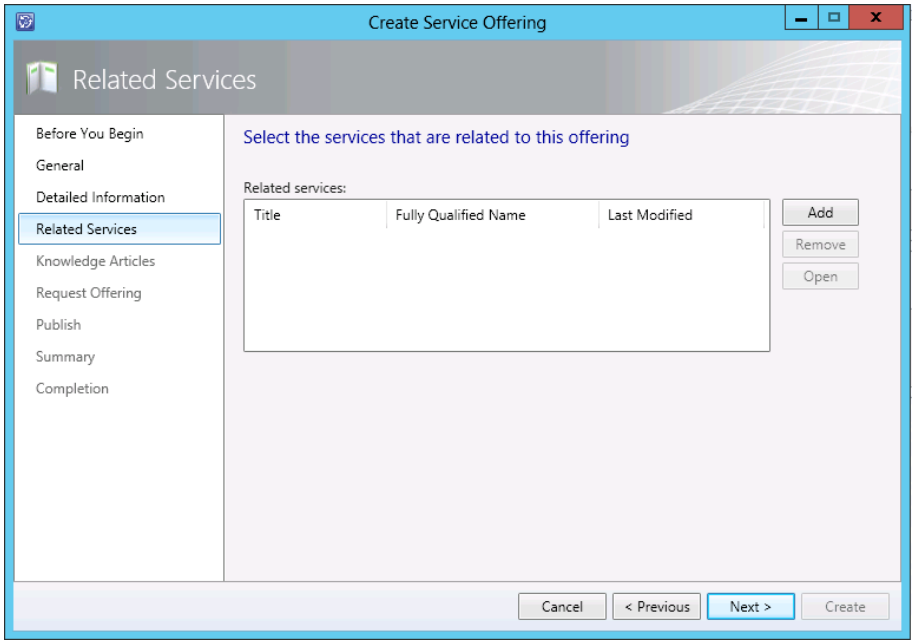
    Select **Next**



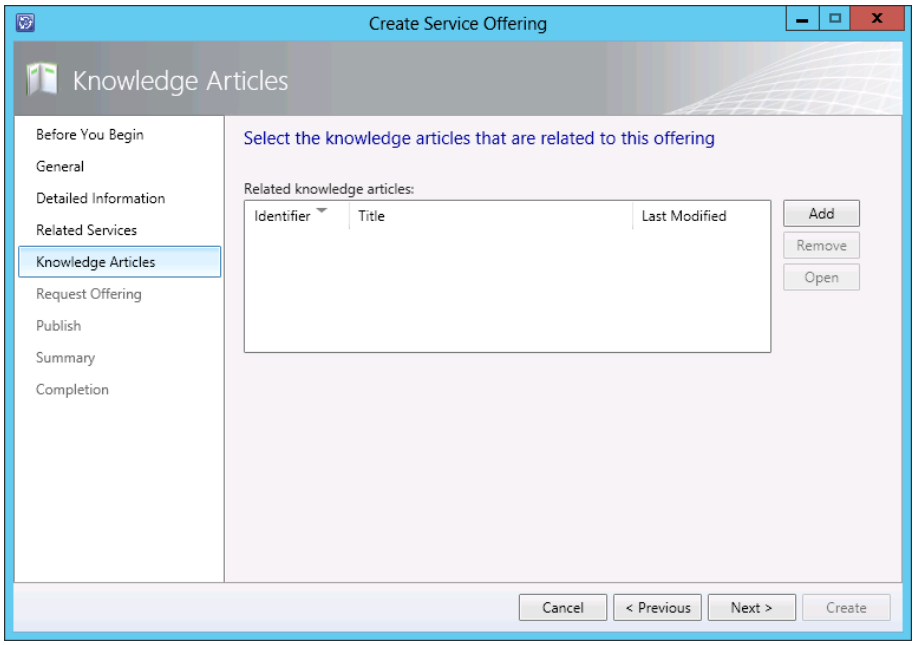
5. The **Detailed Information** page enables Jeff to provide both **Service level agreement** (and a related URL), as well as **Cost information** (and a related URL), which are associated with the service offering. Currently Jeff has not figured out what these might be in his evaluation, but will review these before going into production, for now, just select **Next**



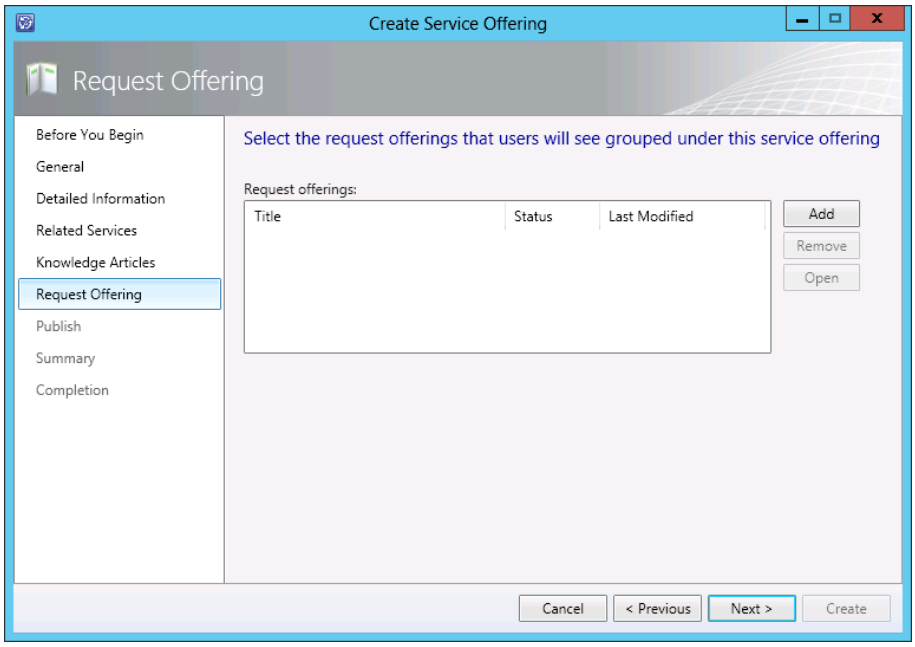
6. Click **Next**



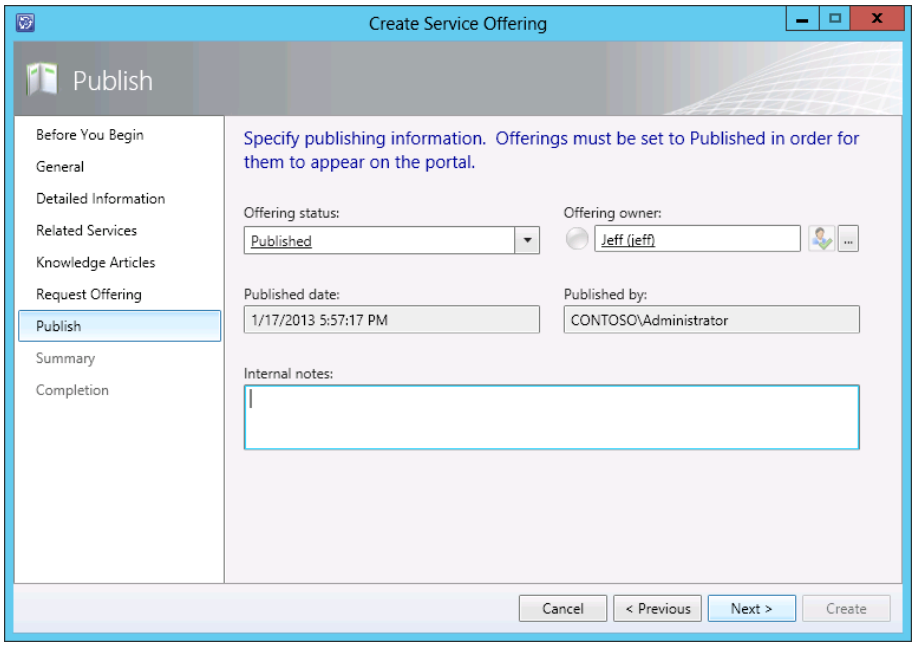
7. Click **Next**



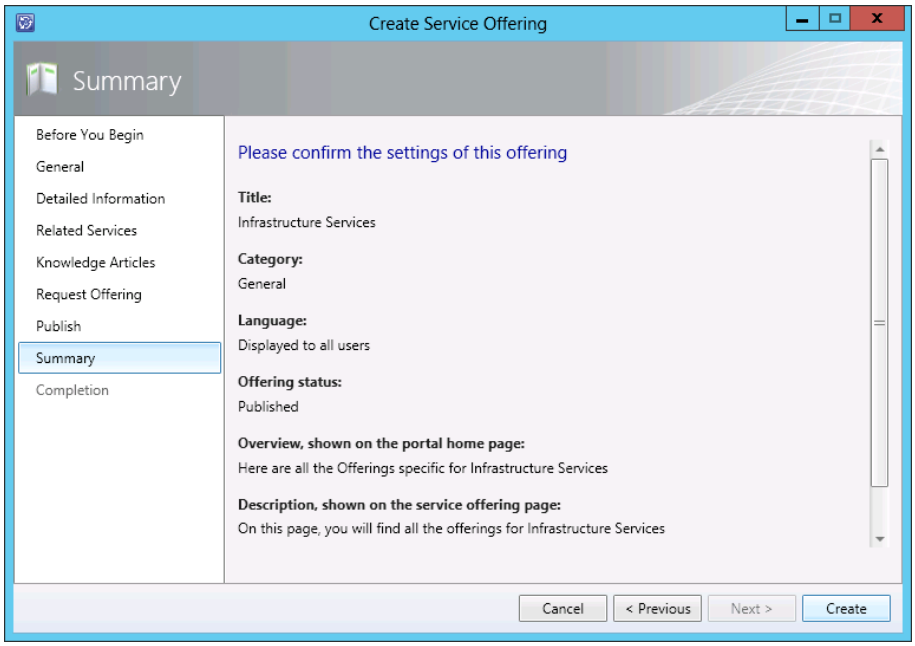
8. On the **Request Offering** page, we have the ability to add the request offerings that users will see grouped under this service offering. Because we have not yet created a request offering, we will leave this empty for now and select **Next**



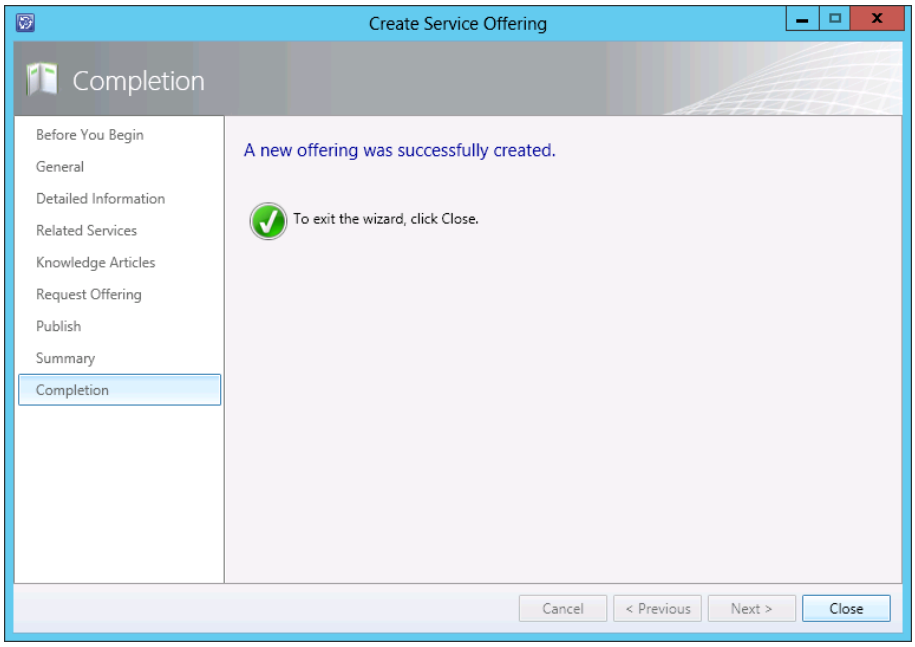
9. On the **Specify publishing information** page, we will set the **Offering status** to **Published** and set the **Offering owner** to **Jeff.** And again we can select **Next**



10. Finally, we can review the choices from the **Summary** page, before we select **Create**



11. When the offering has been created, select **Close**. Please note that we will not see this service offering presented in the portal until we associate at least one request offering to the service

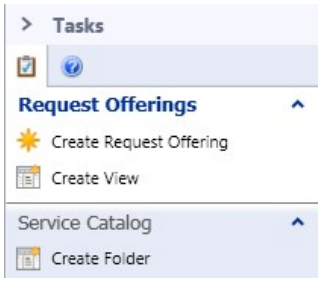


Create Request Offering

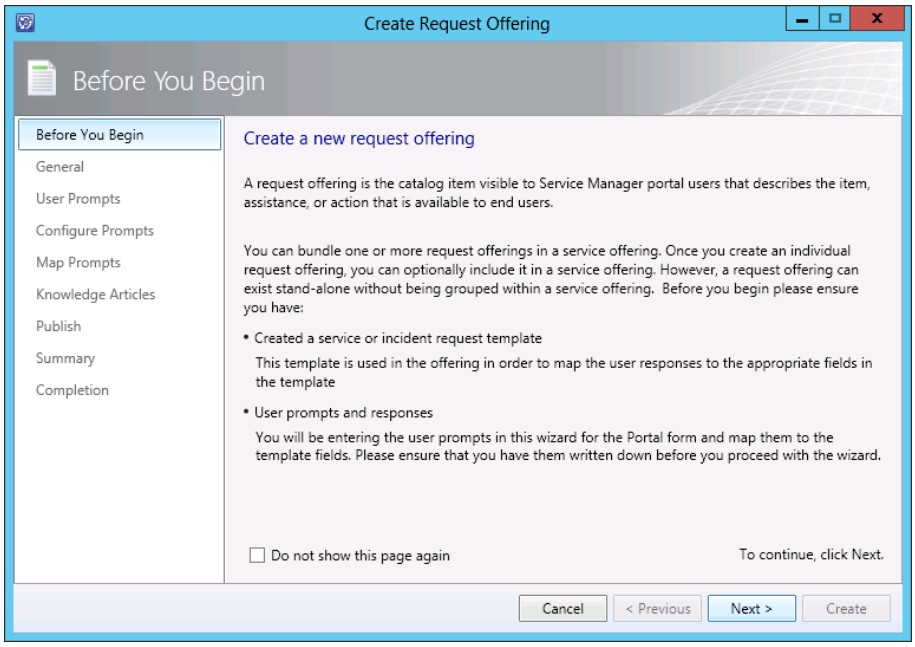
Jeff just finalized his **Service Offering** and is now eager to publish his first **Request Offering**. As Jeff required just some specific information for this runbook, instead of manually extracting the information from a generic request, and transposing this into his runbook, he is instead going to customize the offering specifically prompting the user for the information which is relevant to this specific request.

1. In the System Center 2012 Service Manager console, with the **Library** workspace selected, expand the navigation tree to select **Service Catalog** > **Request Offerings**

2. The context of the **Tasks** Pane will update, and now we can choose **Create Request Offering**



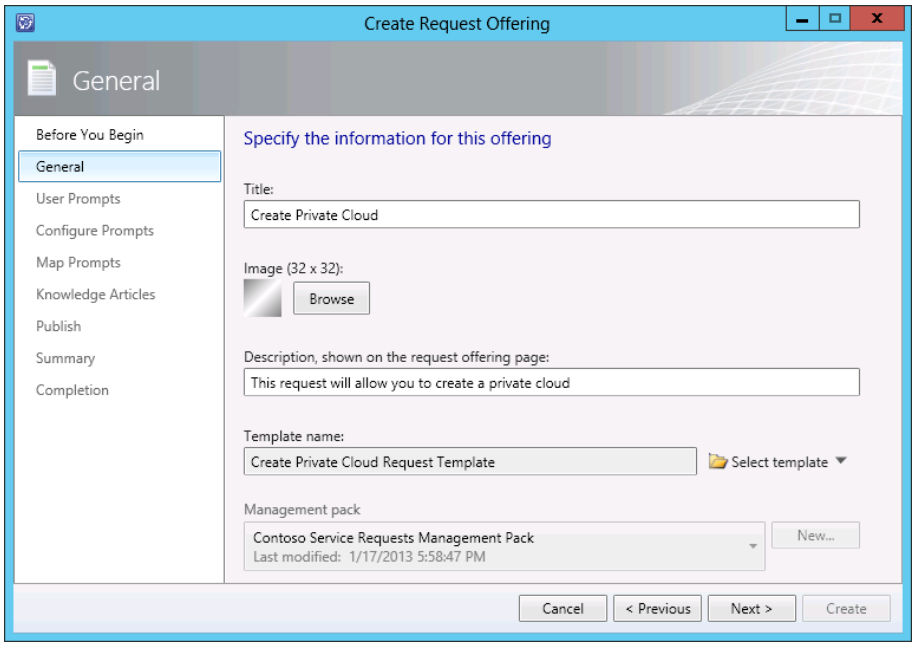
3. The **Create Request Offering** wizard will be presented, select **Next** to begin



4. On the **Specify the information** page, fill in the following information:
    - **Title:** Create Private Cloud
    - **Description:** This request will allow you to create a private cloud
    - **Template name:** Create Private Cloud Request Template

Jeff now understands why he created the template earlier, as every request that will be made through the portal will be created in Service Manager utilizing this template.

- **Contoso Service Request Management Pack** Select **Next**



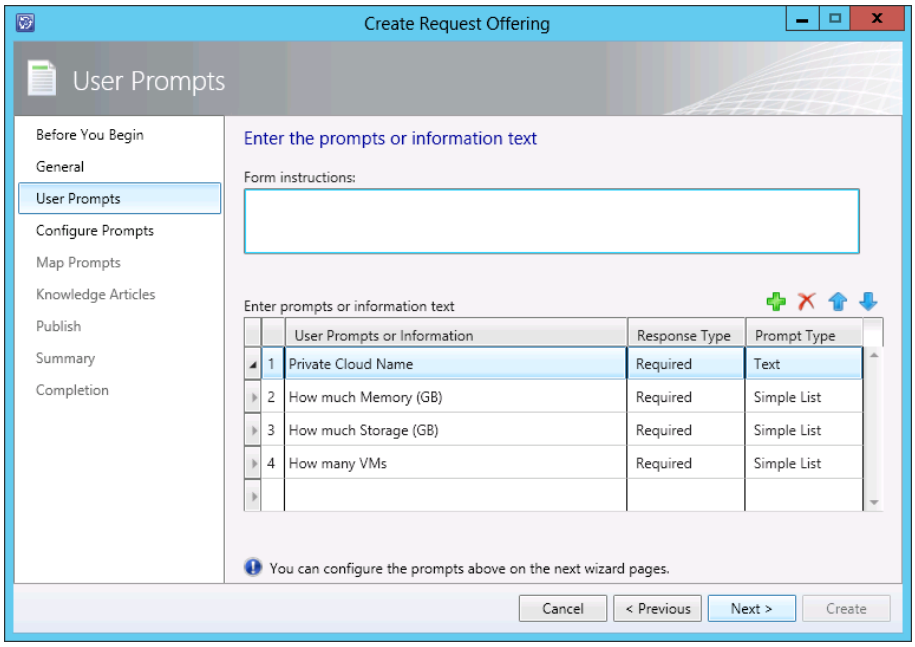
The **User Prompts** page enables Jeff to define what questions should be presented to his end users when they select this request offering, these can be marked as mandatory or optional for the user to provide.

(For more information about the different prompt types: http://blogs.technet.com/b/servicemanager/archive/2011/11/08/request-offering-wizard-overview.aspx)

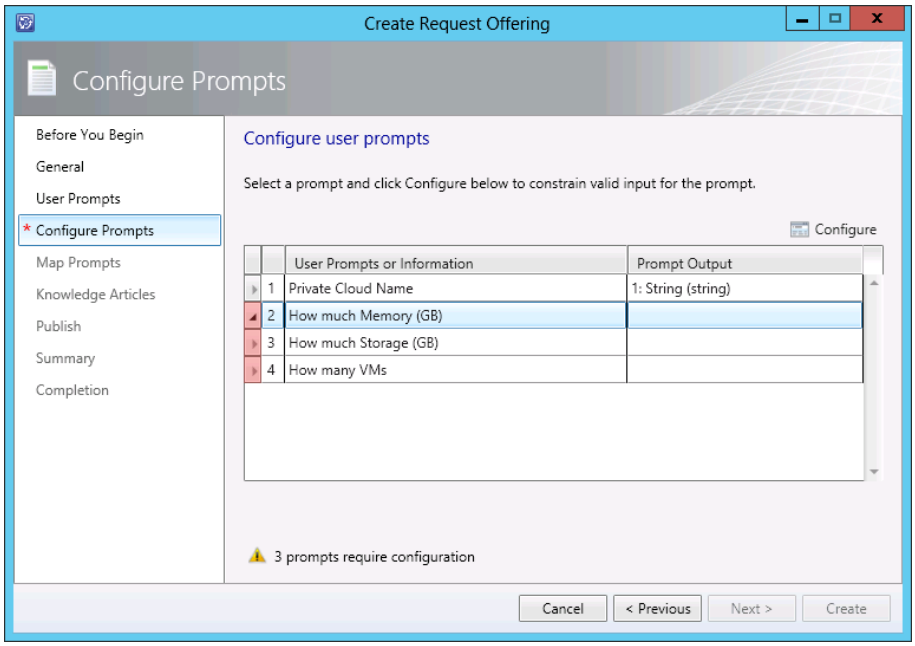
For Jeff's example there are four questions which he requires to be answered, so appends these to the form

- **Private Cloud Name** > Required > Text
- **How much Memory (GB)** > Required > Simple List
- **How much Storage (GB)** > Required > Simple List
- **How many VMs** > Required > Simple List

Select **Next**



5. On the **Configure Prompts** page, we will need to configure the **List** user prompts Select the **How much Memory (GB)** prompt and click the Configure button.



6. On the Configure prompt, add the following items, these will represent the predefined units of memory (in GB) that can be configured for a new Private Cloud.

- **8**
- **16**
- **32**
- **64**

Click **OK**

7. Repeat the previous step for the **How much Storage (GB)** and **How many VMs** prompts populating the lists with the following values:
   How much Storage (GB)
   - **64**
   - **128**
   - **256**
   - **512**

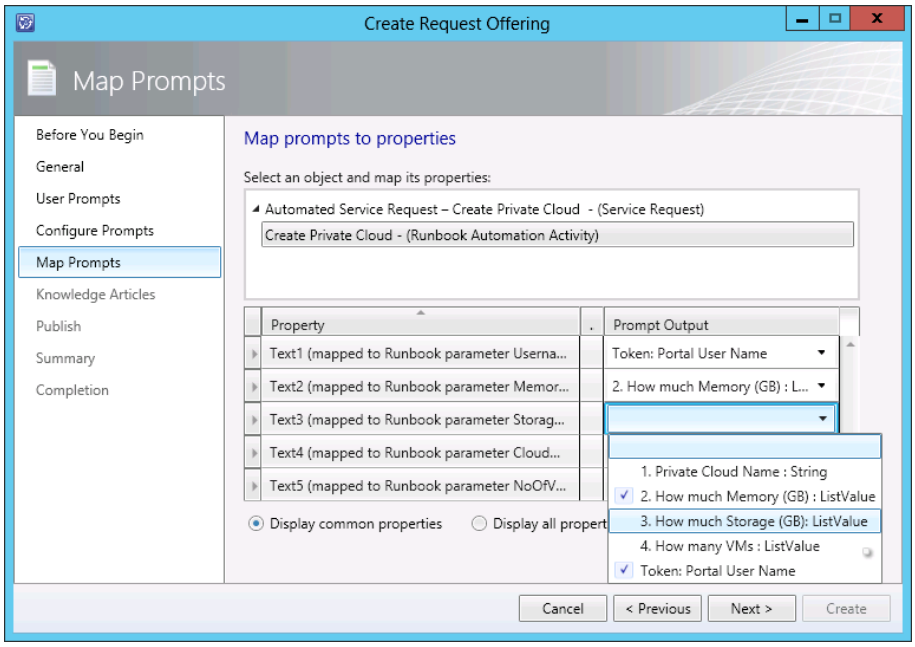
   How many VMs
   - **4**
   - **8**
   - **16**
   - **32**

Click **Next**

8. On the **Map prompts** page, we are going to map the user's responses to fields so that we can process them in Service Manager. Jeff plans to use all of the prompts to be passed as parameters to his runbook activity so that Orchestrator knows what we would like to take action on. To do this, we simply select the **Runbook Automation Activity** we created earlier called **Create Private Cloud**
   Now, we can map all of the prompt fields to the parameters of the activity by widening the **Property** field to see the full entry, and then mapping the matching option in the **Prompt Output** field. For example, for the **Property** that is **mapped to Runbook parameter StorageGB**, select the **Prompt Output** option **How much Storage (GB).**

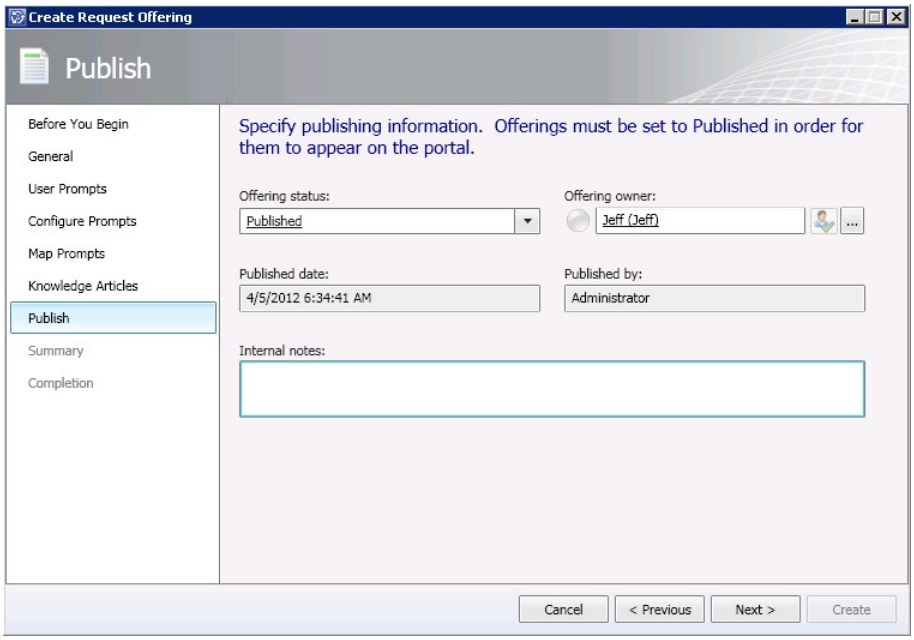
   Additionally we need to pass the user name of the person creating the service request to the runbook as a parameter, to do this map the **Token: Portal User Name** to the **Text** property that is **mapped to Runbook parameter Username**.

   Proceed by selecting **Next**

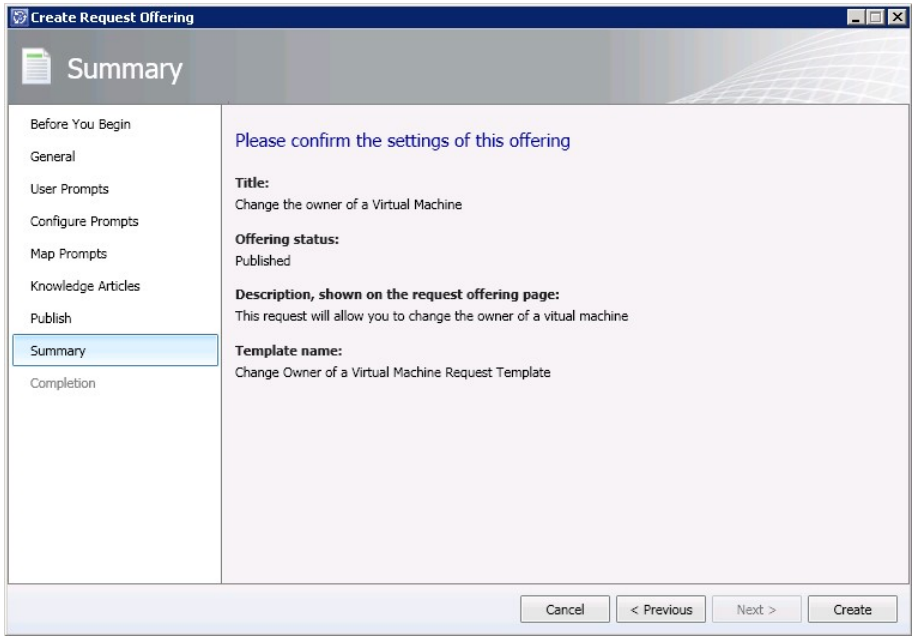


9. Click **Next** on the **Knowledge Articles**

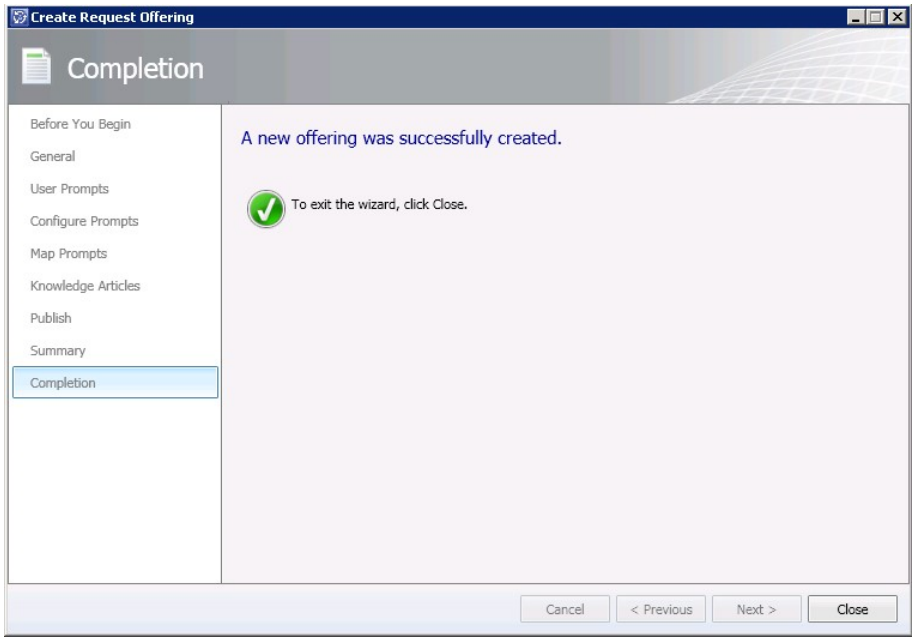
10. On the **Publish page**, change the **Offering** status to **Published** and change the **Offering** owner to **Jeff**. Select **Next**



11. On the **Summary** page, review your settings and then select **Create**



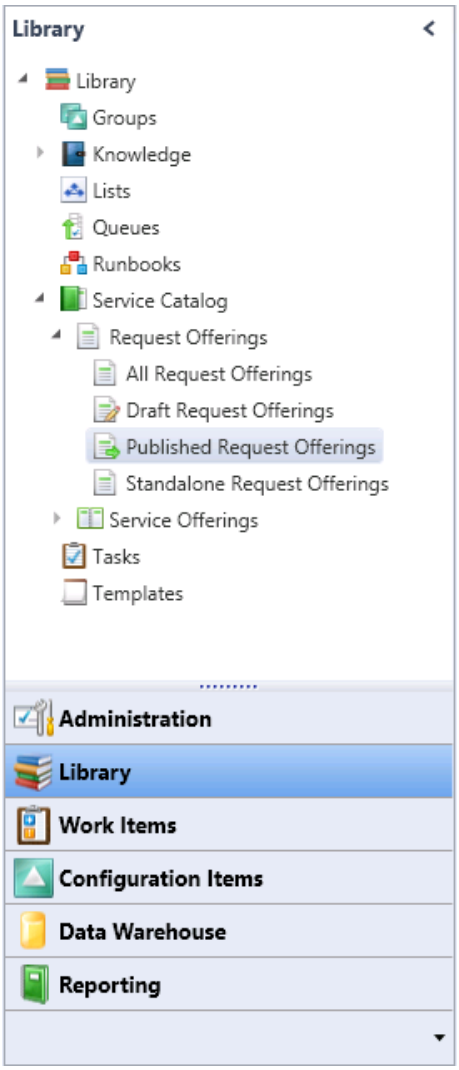
12. Once the **Completion** page is presented we can select **Close**



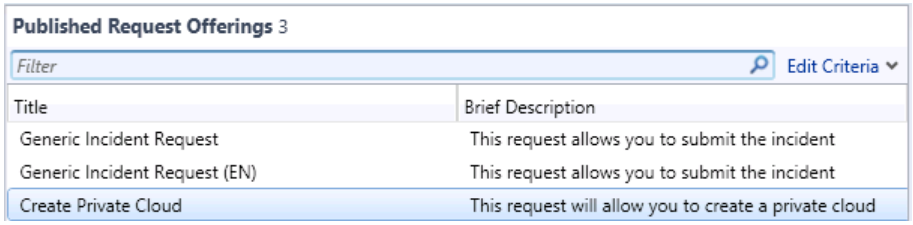
Add the Request Offering to the Service Offering

Jeff now has to add this **Request Offering** to the **Service Offering** that we created earlier. Then we will be able to see the offering on the **Self-Service** portal.

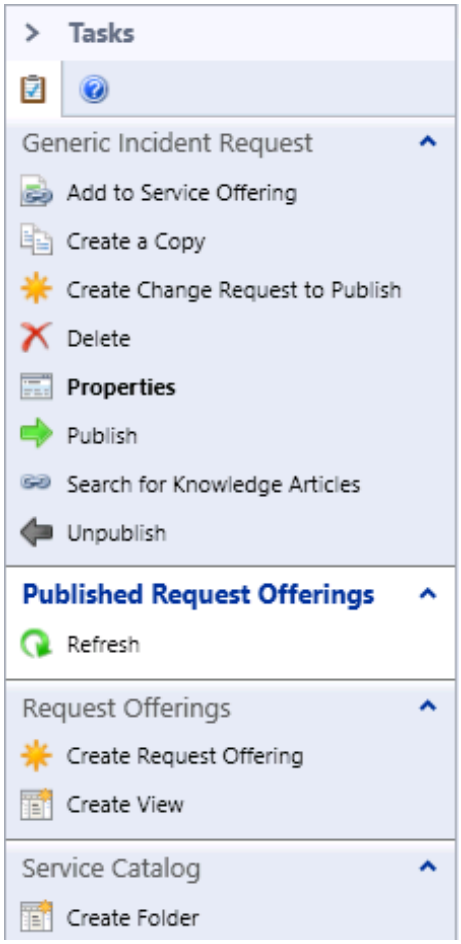
1. In the **System Center 2012 Service Manager** console, with the **Library** workspace selected, expand the navigation tree to select **Service Catalog** > **Request offerings** > **Published Request Offerings**



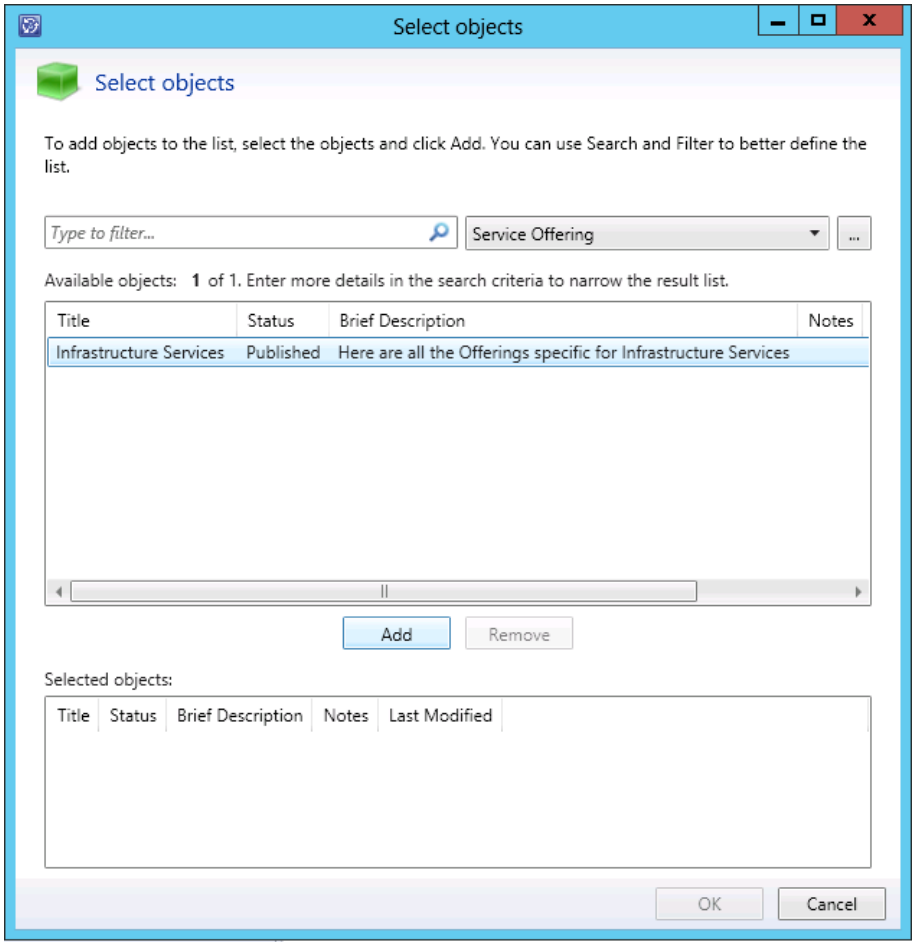
2. Select the **Create Private Cloud** offering from the presented list



3. The context of the **Tasks** pane will update, and now we can choose **Add to Service Offering**



4. Choose **Infrastructure Services** and select **Add**. Then select **OK**

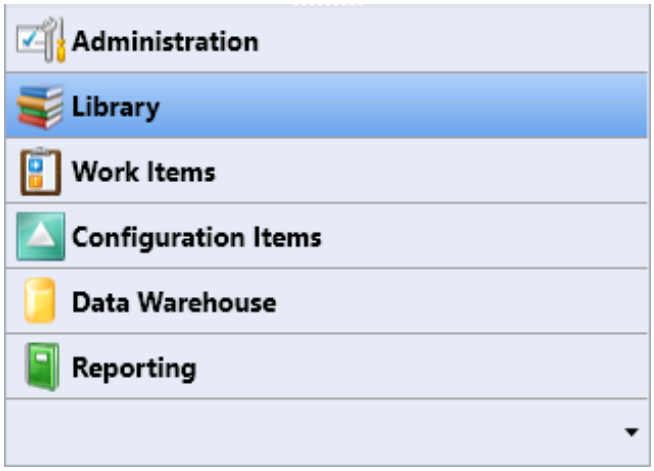


Create Service Catalog Groups

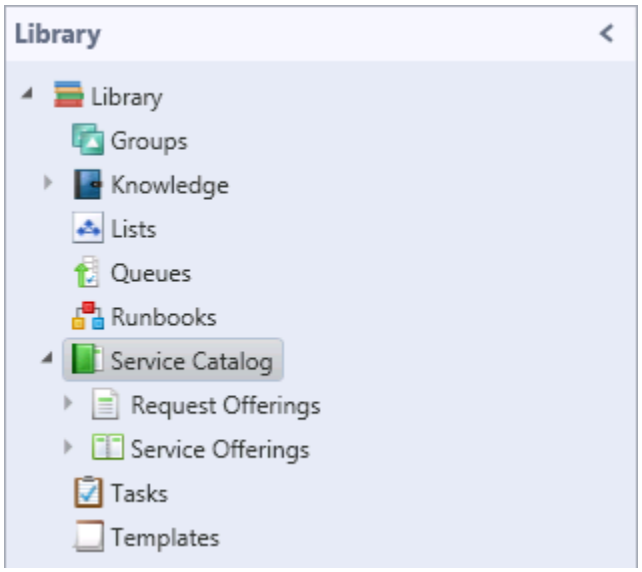
Jeff creates a **Service Catalog Group** that be will used for defining role based access to specific Service Request Offerings from the Self-Service Portal. For example Jeff may wish to only show HR related Service Request Offerings to a specific HR AD group - he does this by created **Service Catalog Group** and selecting only those specific **Service Request Offerings** and create a **User Role** which is assigned to the **Service Group Catalog** and an associated AD group. This section will be broken down into two sections

FYI: User Roles in Service Manager are cumulative.

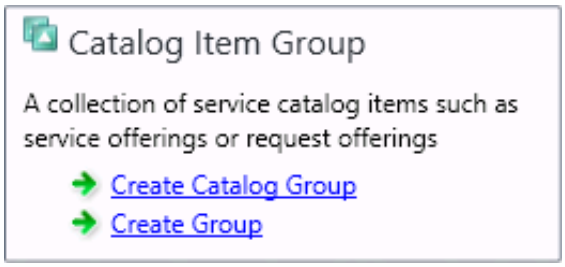
1. From the **System Center Service Manager** console click **Library**



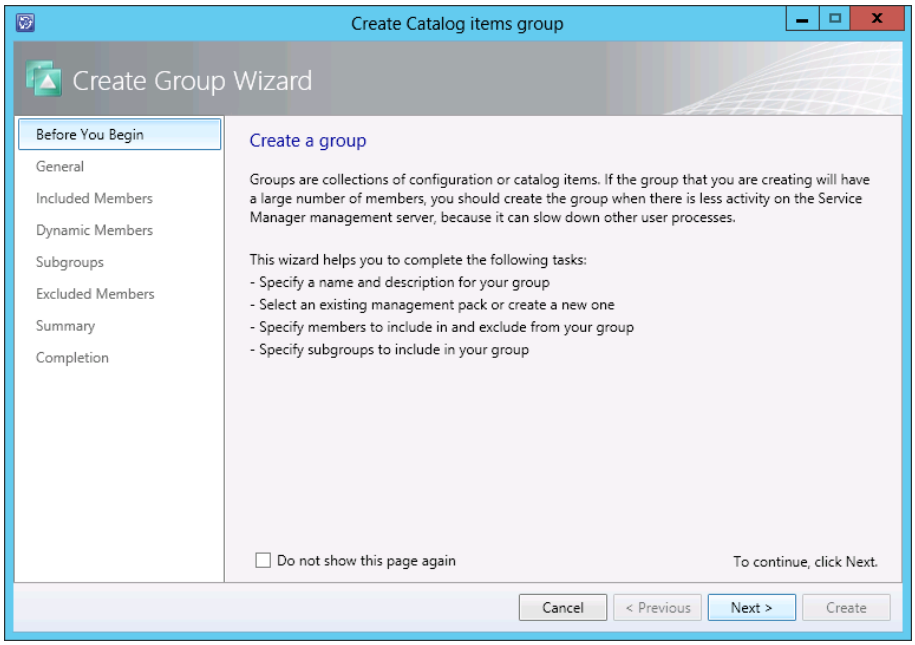
2. Then select **Service Catalog**



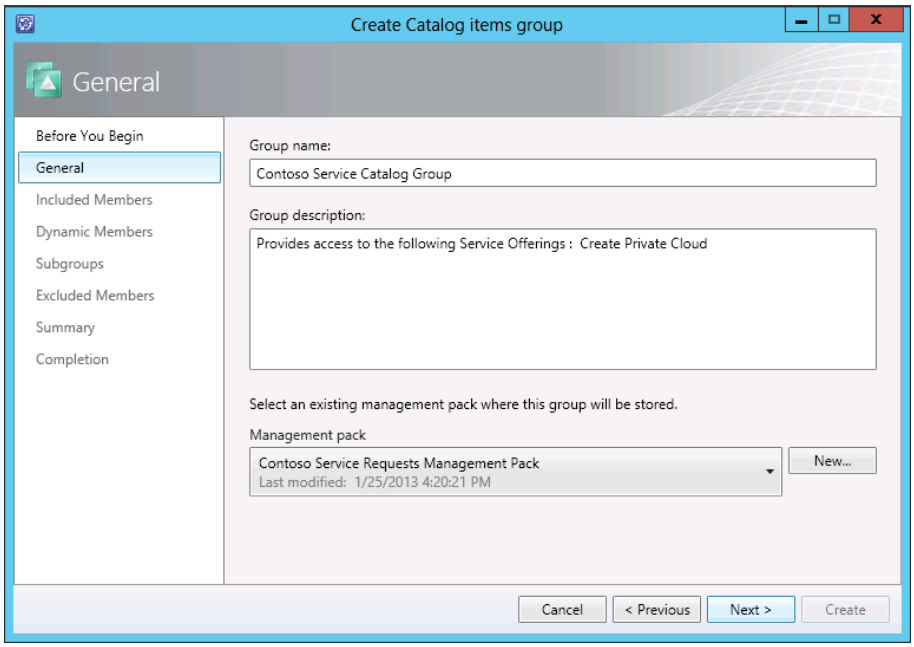
3. And in the right-hand side column, click **Create Catalog Group**



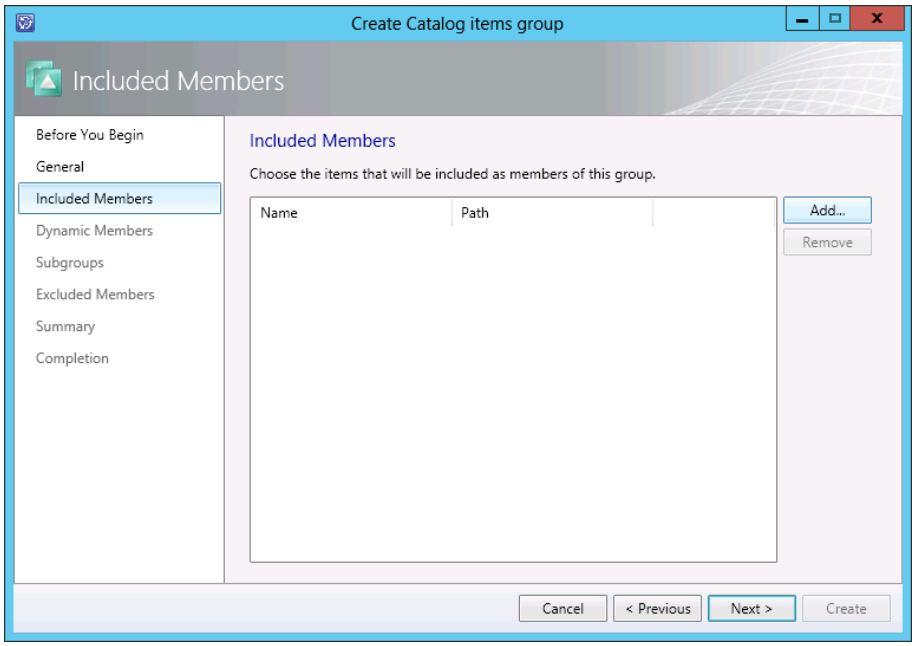
4. From the **Create Group Wizard** click **Next**



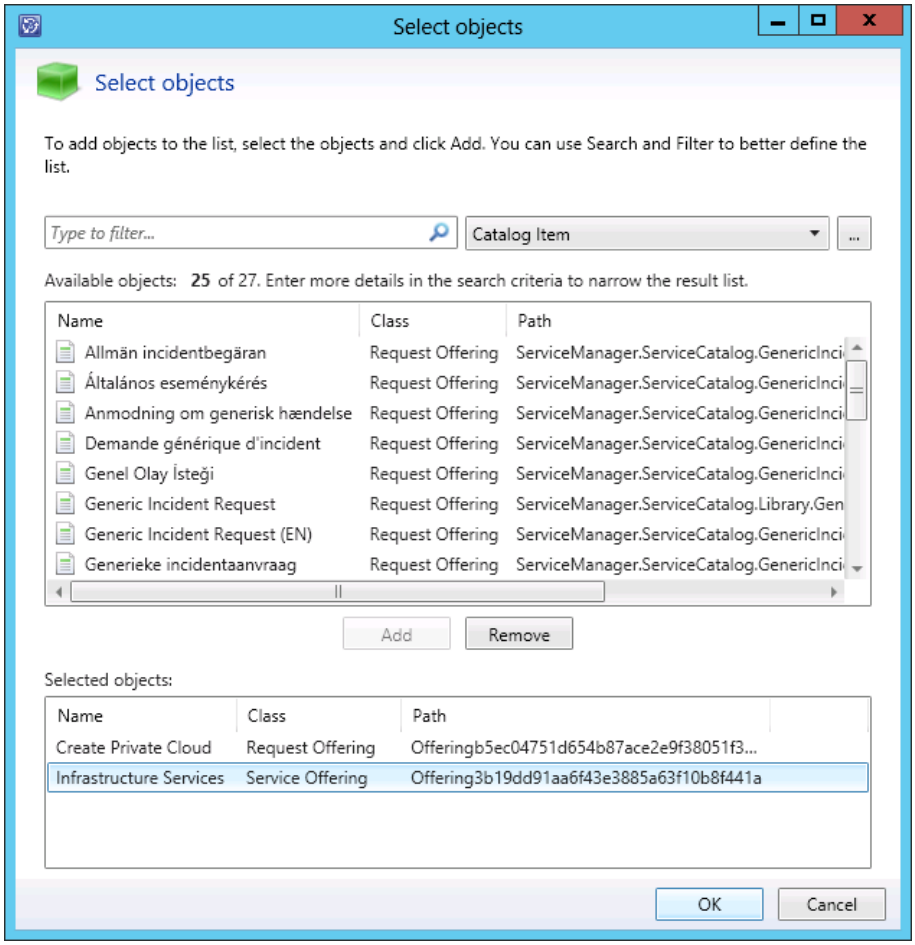
5. Next Jeff creates a Catalog Group called **Contoso Service Catalog Group** and adds all Service Offerings & Request Offerings to this group
    - **Group name** Contoso Service Catalog Group
    - **Group description** Provides access to the following Service Offerings: Create Private Cloud
    - **Management pack** Contoso Service & Request Offerings



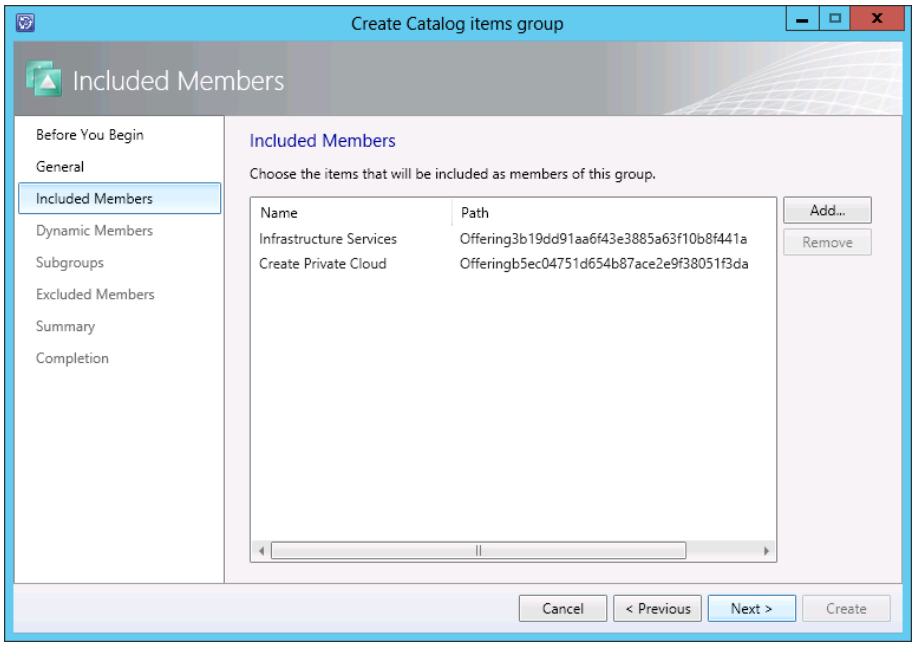
6. From the **Included Members** screen click on **Add...**



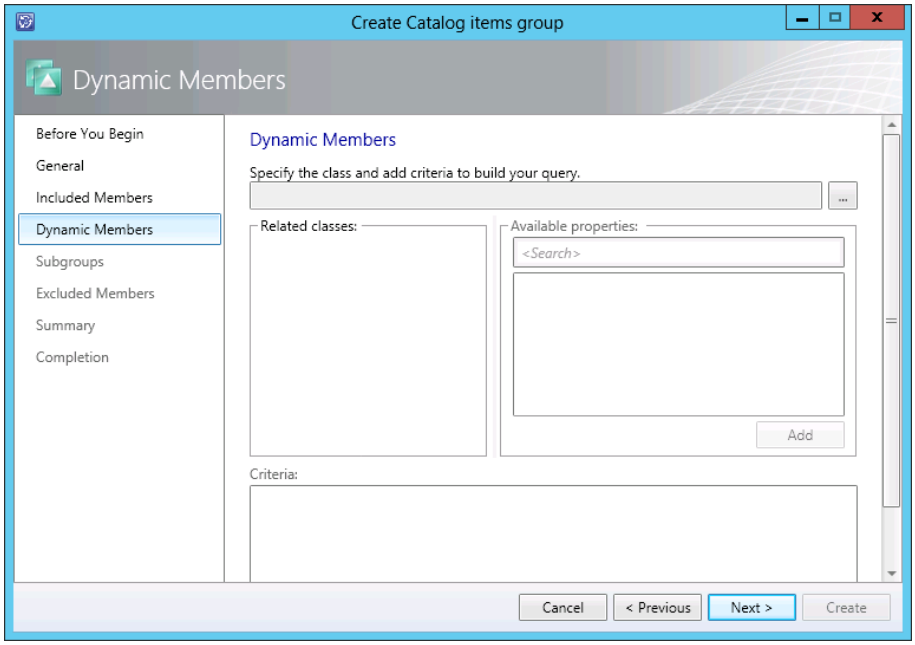
7. Select **Create Private Cloud** and **Infrastructure Services**, click **OK**



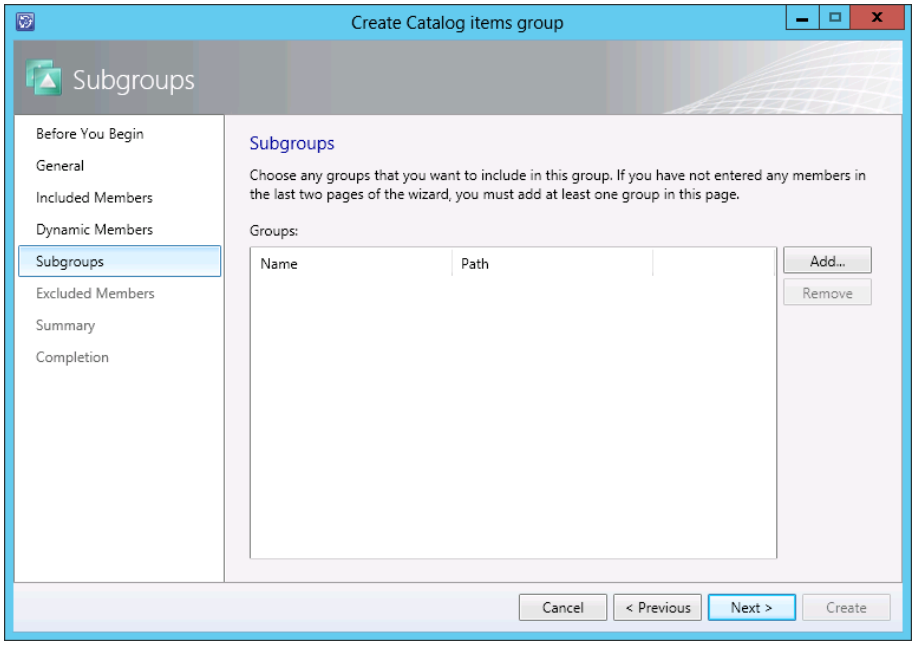
8. Confirm **Create Private Cloud** and **Infrastructure Services** are shown then click **Next**



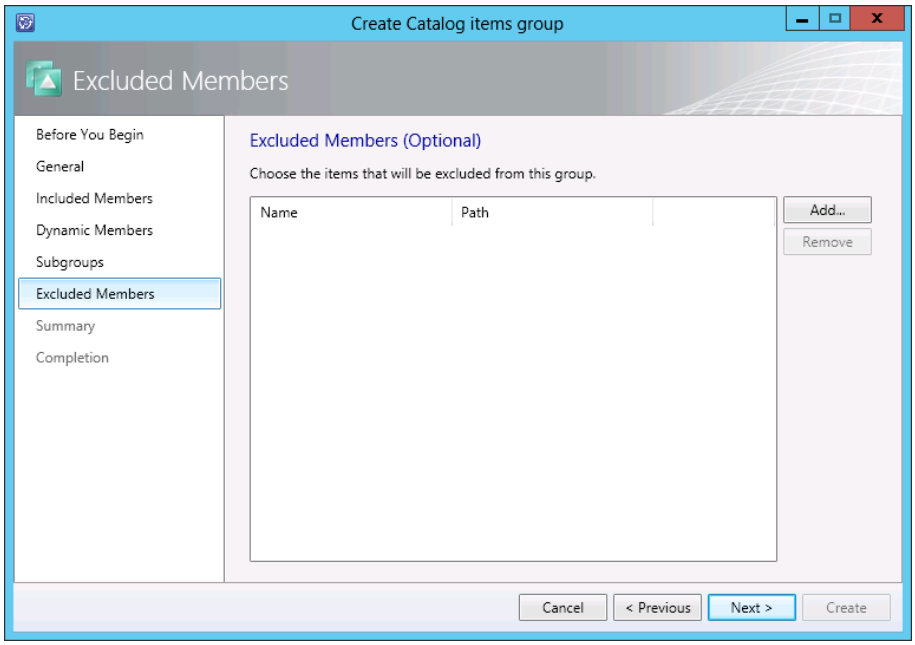
9. For **Dynamic Members** leave all as default and select **Next**



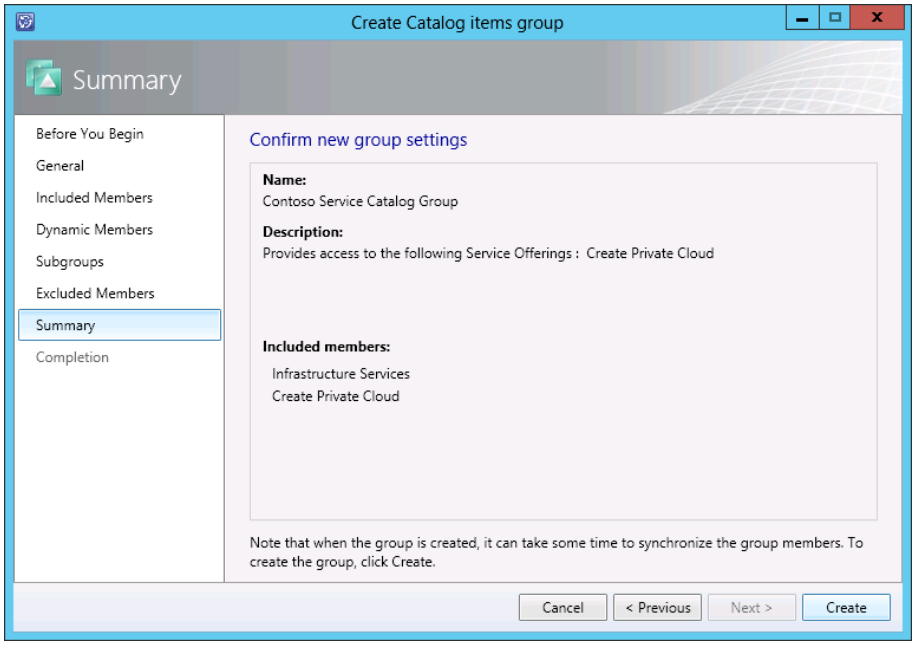
10. For **Subgroups** leave all as default and select **Next**



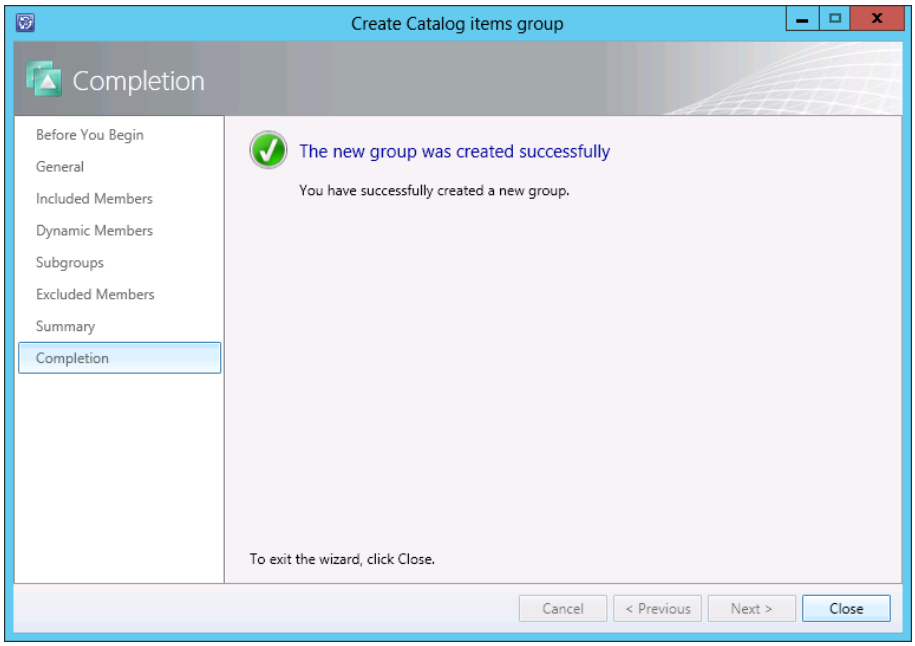
11. For **Excluded Members** leave all as default and select **Next**



12. On the **Summary** Screen click **Create**



13. Click **Close** after the group is created

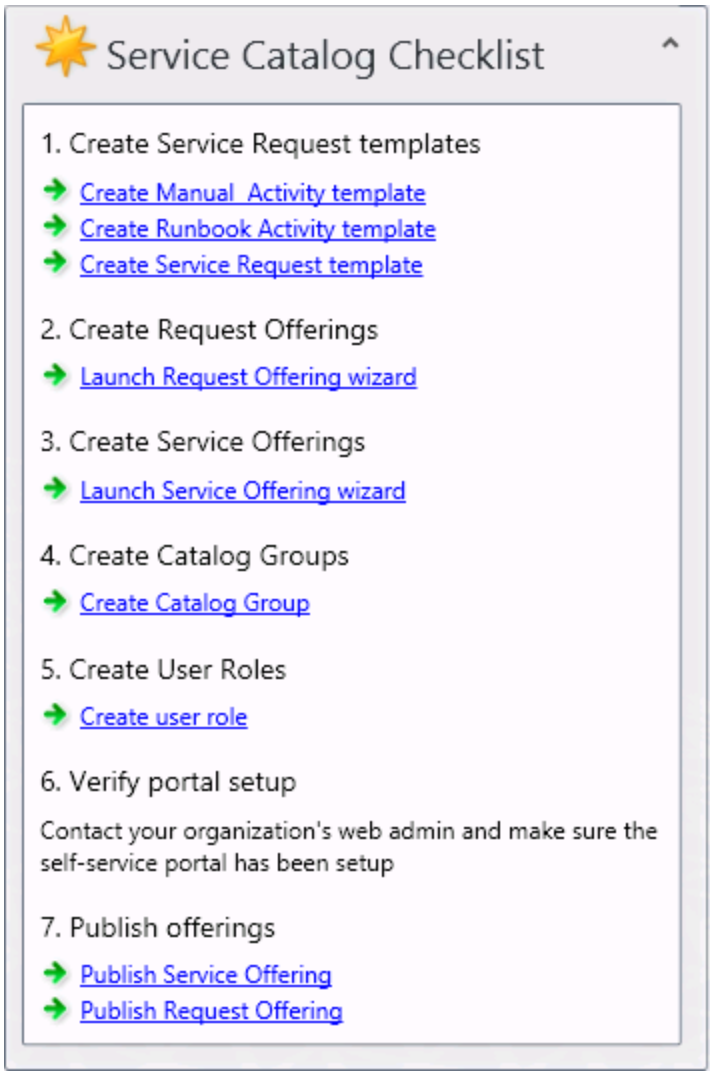


Create User Role

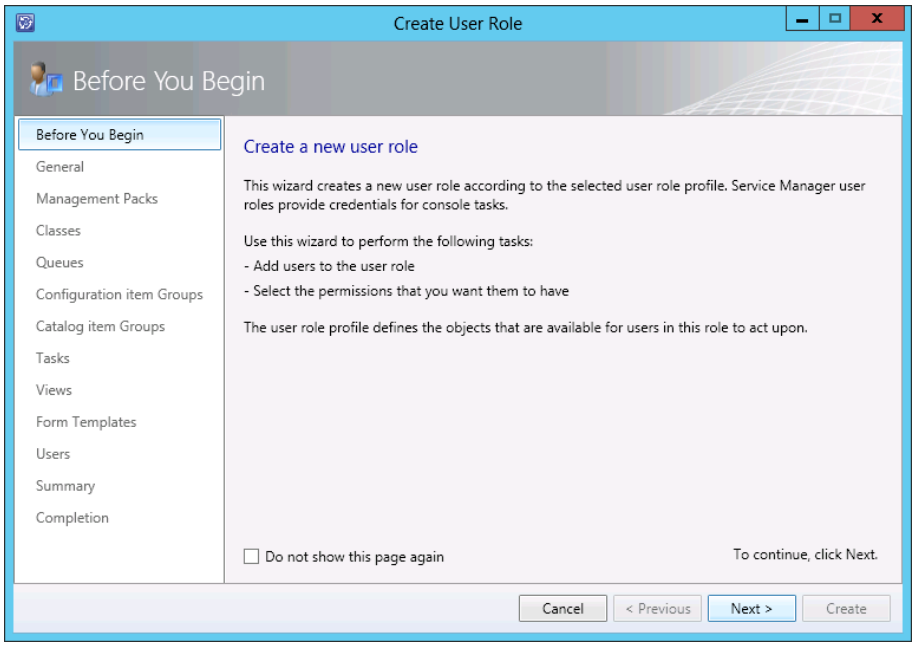
In System Center 2012 – Service Manager SP1, the security rights that allow users to access or update information are defined in a user role profile. A user role profile is a named collection of access rights, and it usually corresponds to an employee's business responsibilities. Each user role profile controls access to such artifacts as knowledge articles, work items (incidents, change requests), authoring, administration, and other credentials. Think of user role profiles as defining what you are allowed to do.

Jeff will now grant Emily access to the relevant parts of Self-Service Portal so she can submit a new service request.

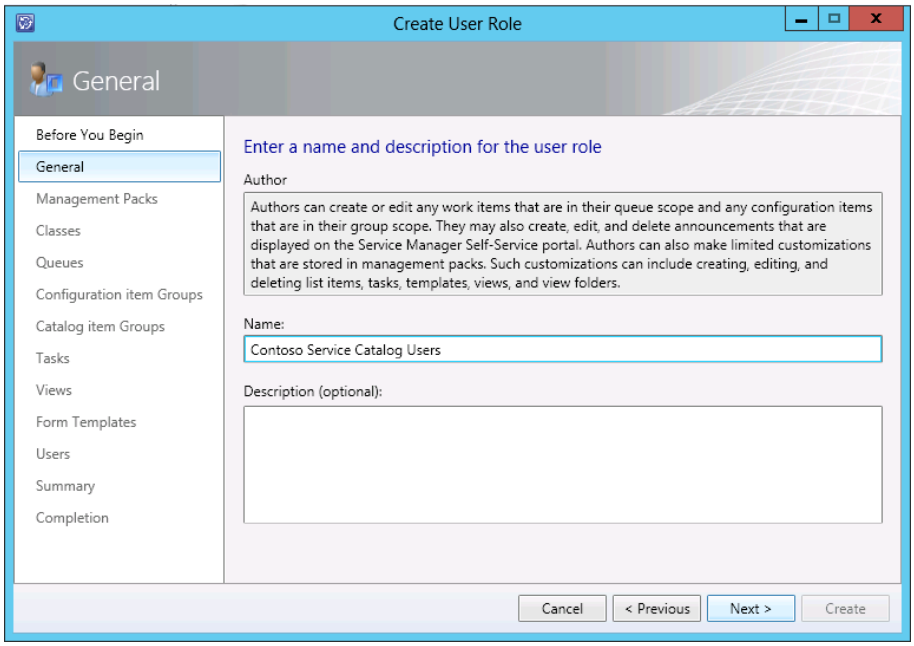
1. On the **System Center Service Manager** console, click **Library** then click **Create User Role**



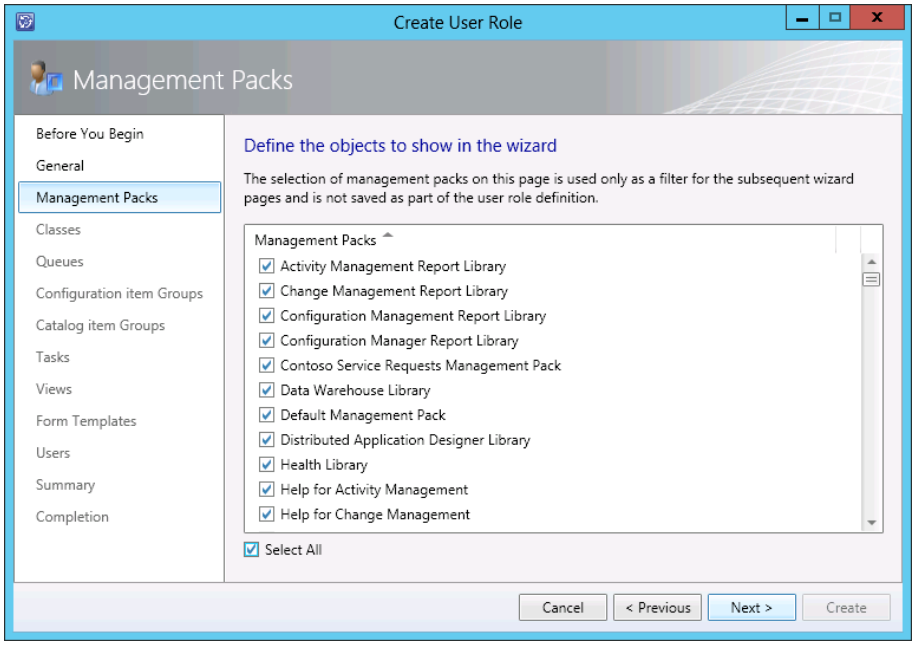
2. On the **Before You Begin** screen click on **Next**



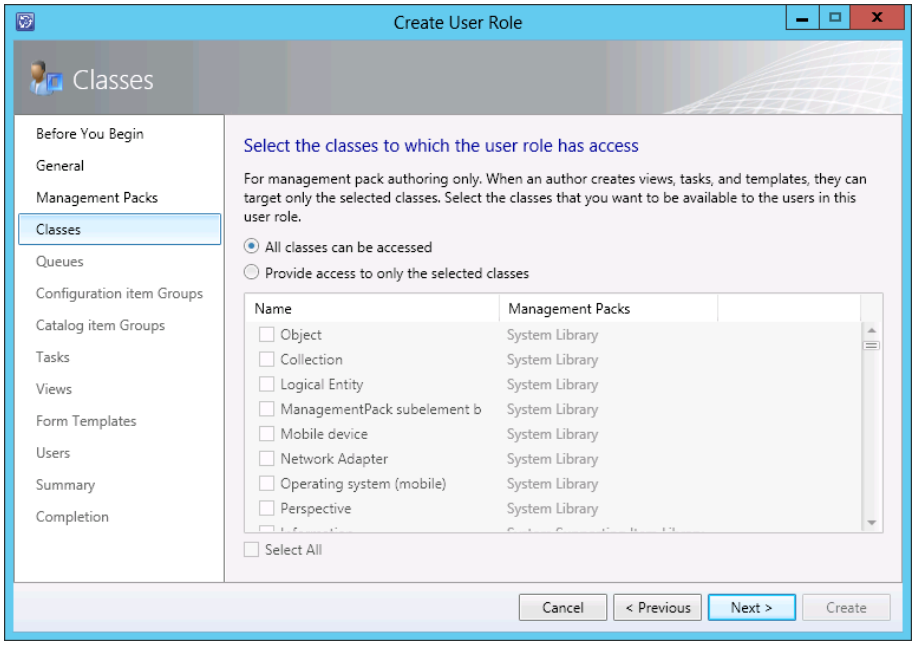
3. Input the name **Contoso Service Catalog Users**, click **Next**



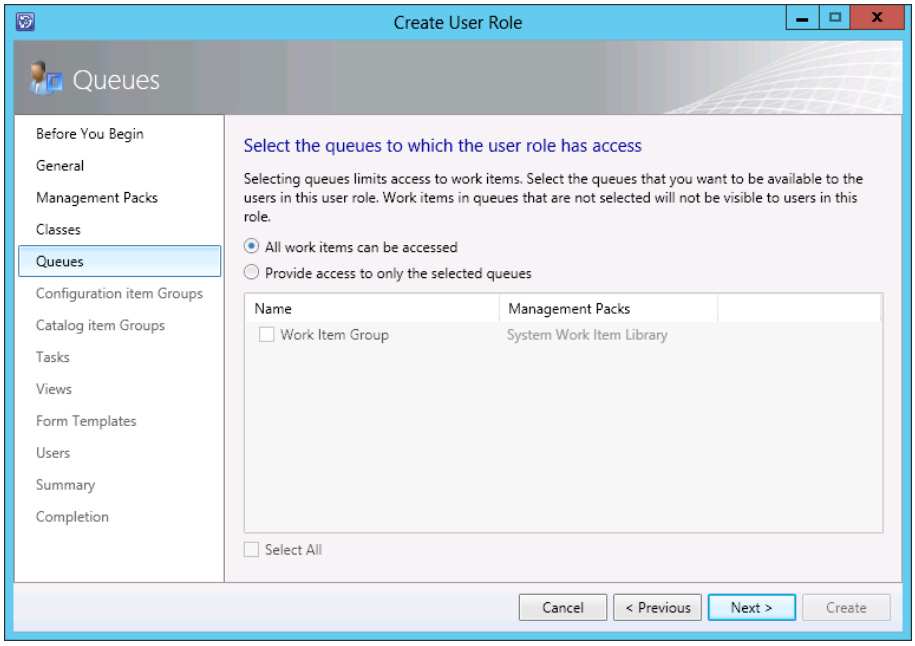
4. In the **Management Packs** screen check the **Select All** box and click **Next**



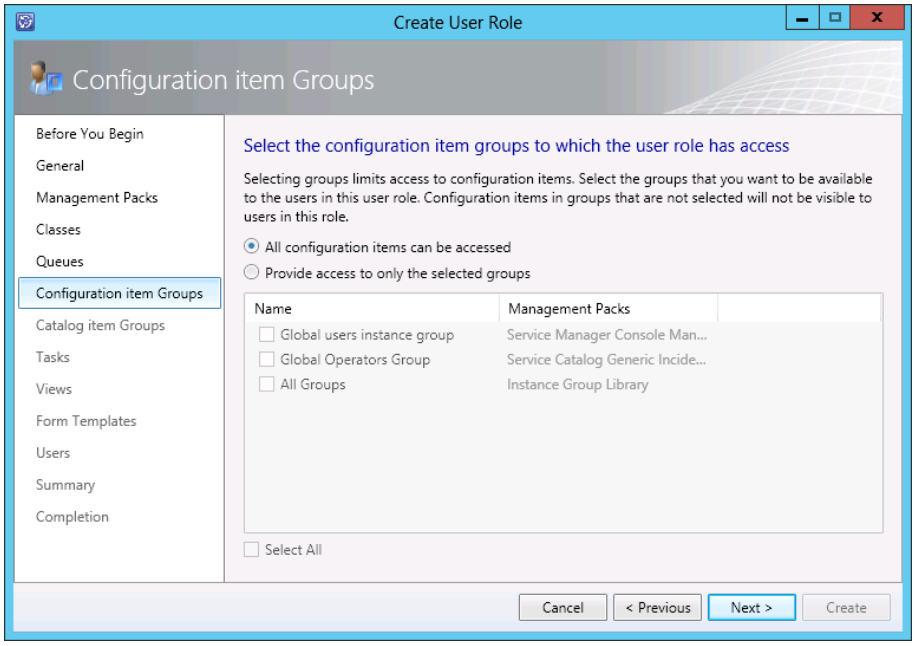
5. Leave the **Classes** as default and click **Next**



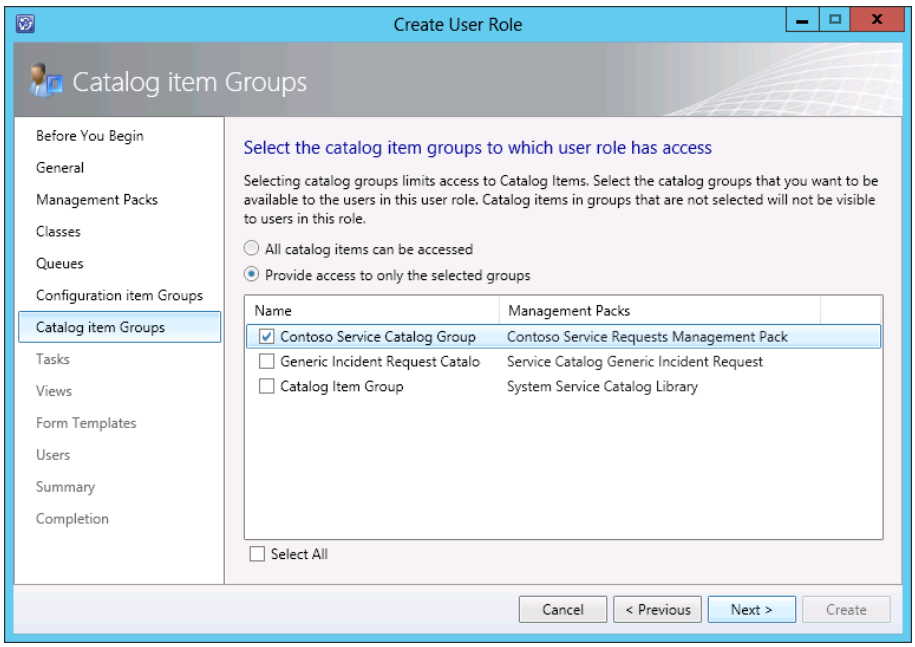
6. Tick **All** for the **Queues** followed by **Next**



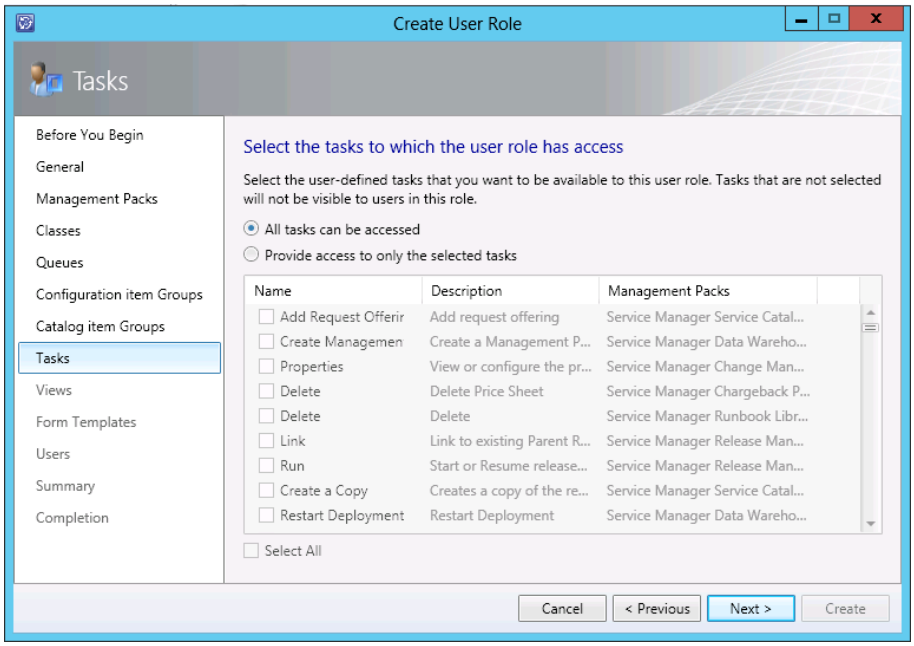
7. Tick **All** for the **Configuration item Groups** followed by **Next**



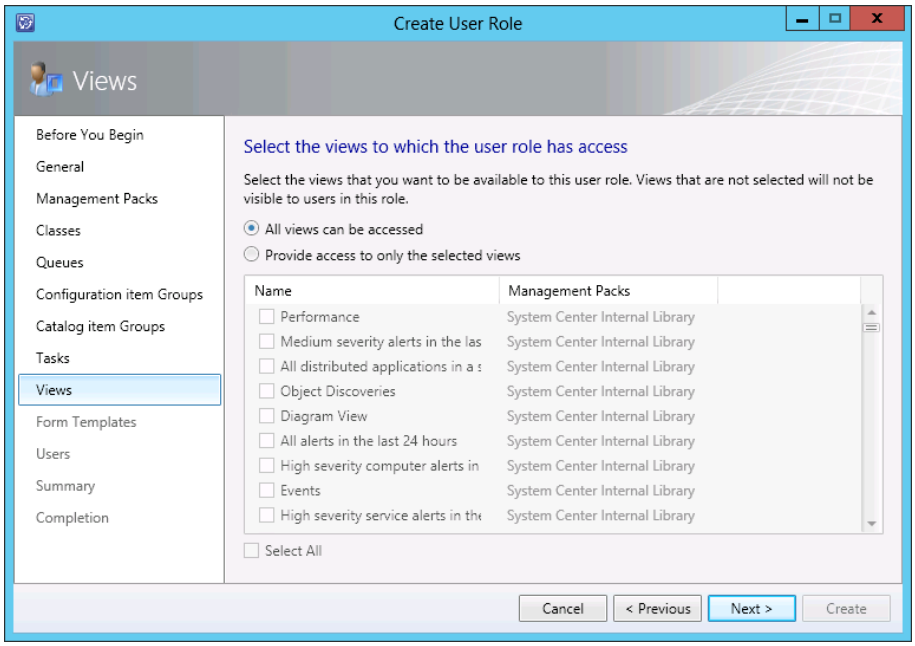
8. On the **Catalog item Groups** check, **Provide access to only the selected group**: **Contoso Service Catalog Group** and click **Next**



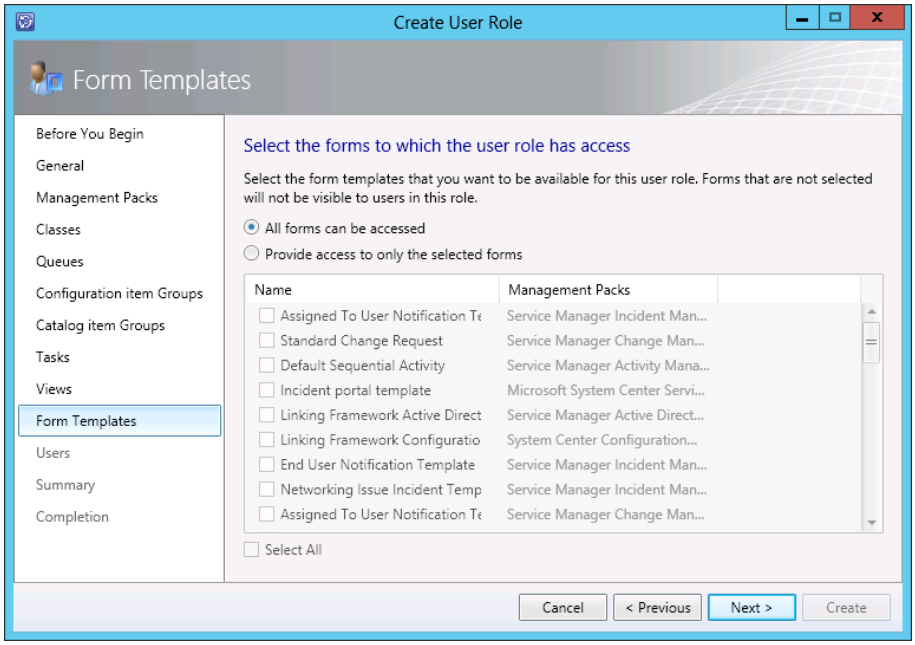
9. Tick **All** for the **Tasks** followed by **Next**



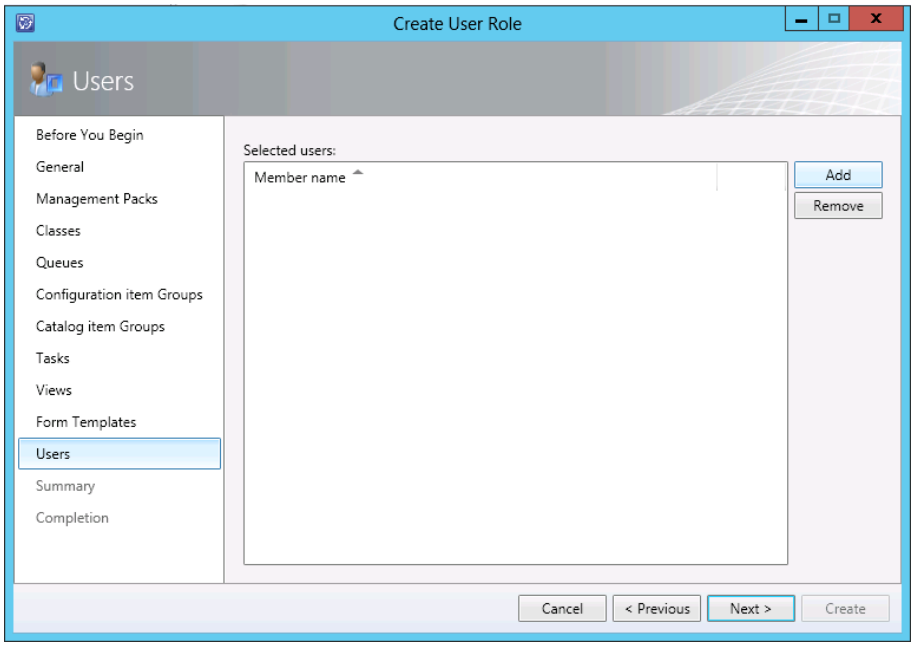
10. Tick **All** for the **Views** followed by **Next**



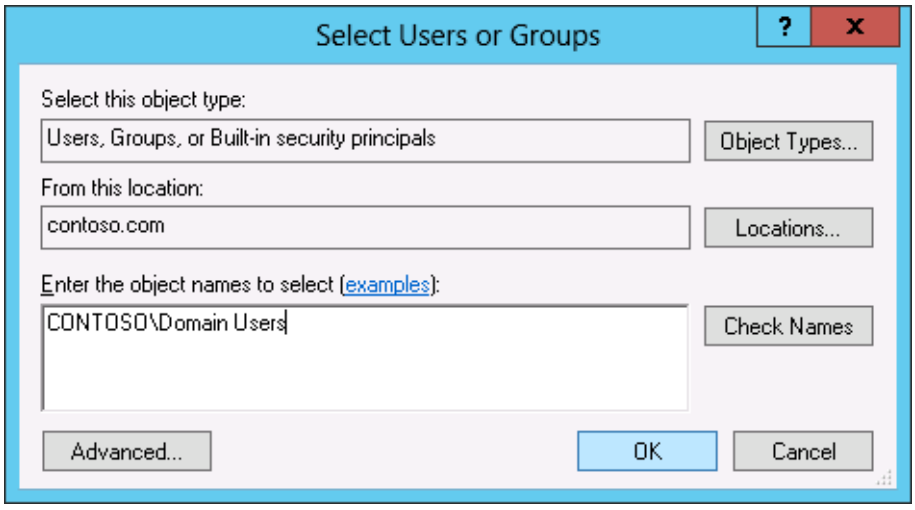
11. Tick **All** for the **Form Templates** followed by **Next**



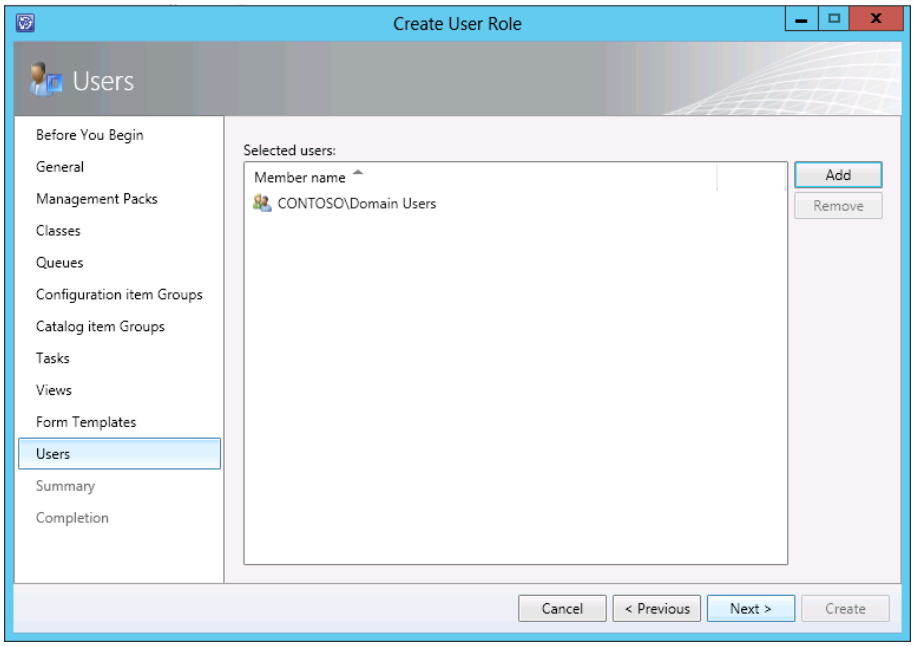
12. For **Users** click **Add**



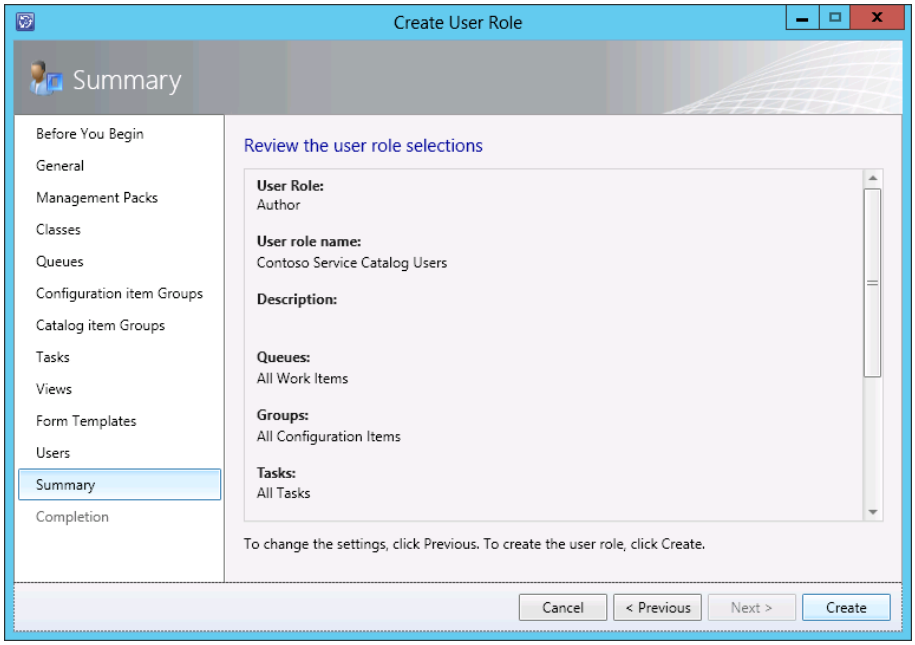
13. Type **contoso\domain users** followed by **OK**



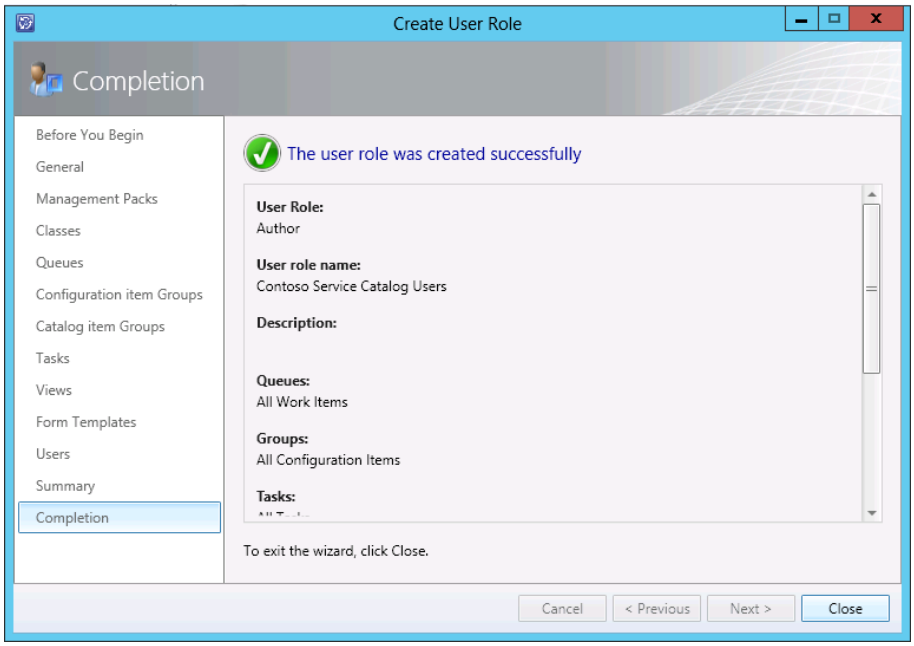
14. Confirm **CONTOSO\Domain Users** is displayed then click **Next**



15. On the **Summary** screen click on **Create**



16. Once this has been completed click on **Close**



Running through the Service Request

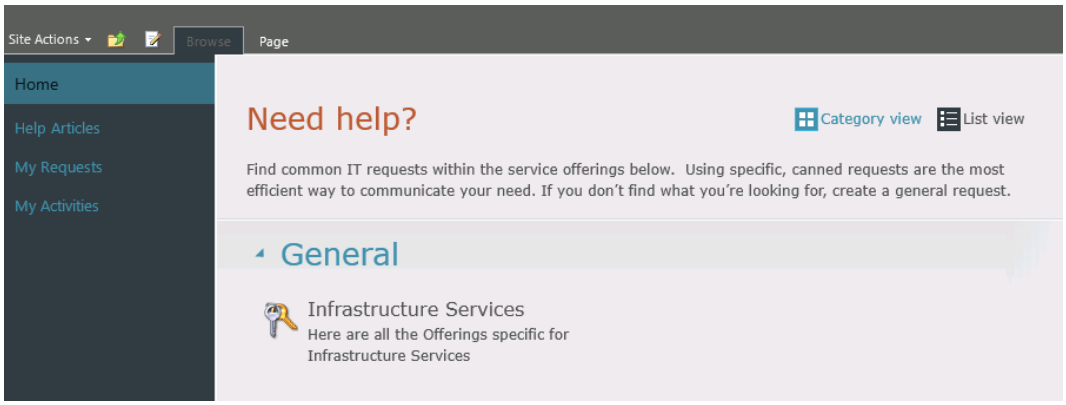
Prior to Jeff releasing this new Service Offering to Emily, he first plans to verify the experience which will be presented to the users and that the interactions with Orchestrator and Virtual Machine Manager do indeed work as planned. For his test scenario, Jeff will create a test Private Cloud.

Important Note

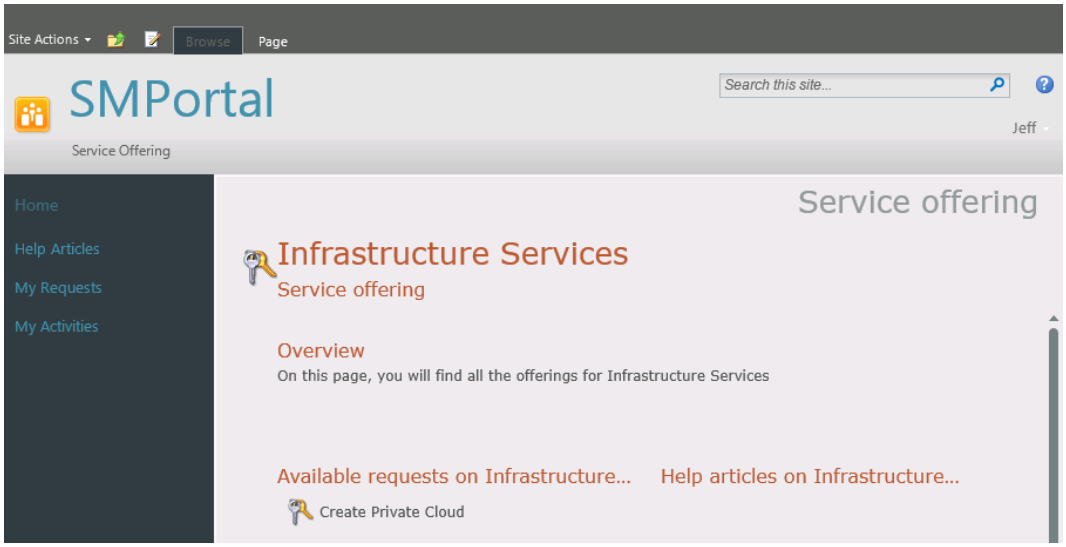
Viewing the App Controller portal with Internet Explorer requires the **Silverlight plugin**. In this example, Jeff is running it on the Service Manager server itself but you could use another Server / Workstation to connect to the portal without needing to install Silverlight on the Service Manager server.

Jeff decides to test this with his own account. He logs onto the Service Manager server as himself (**contoso\jeff**).

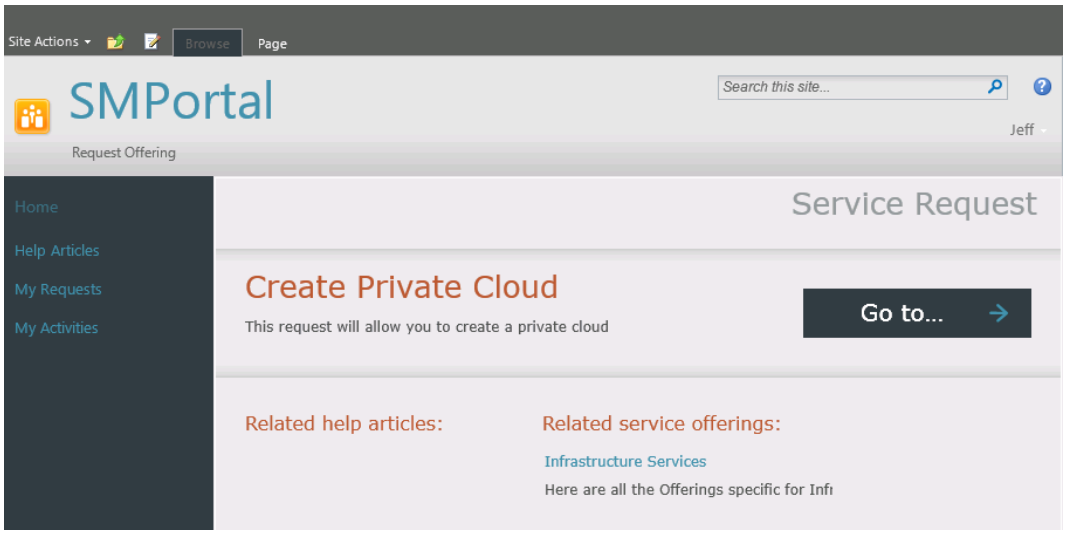
1. Opens the Service Manager portal by browsing to **http://Sharepoint:82/SMPortal** After a few moments the portal will be presented, select **Infrastructure Services** from the category **General**



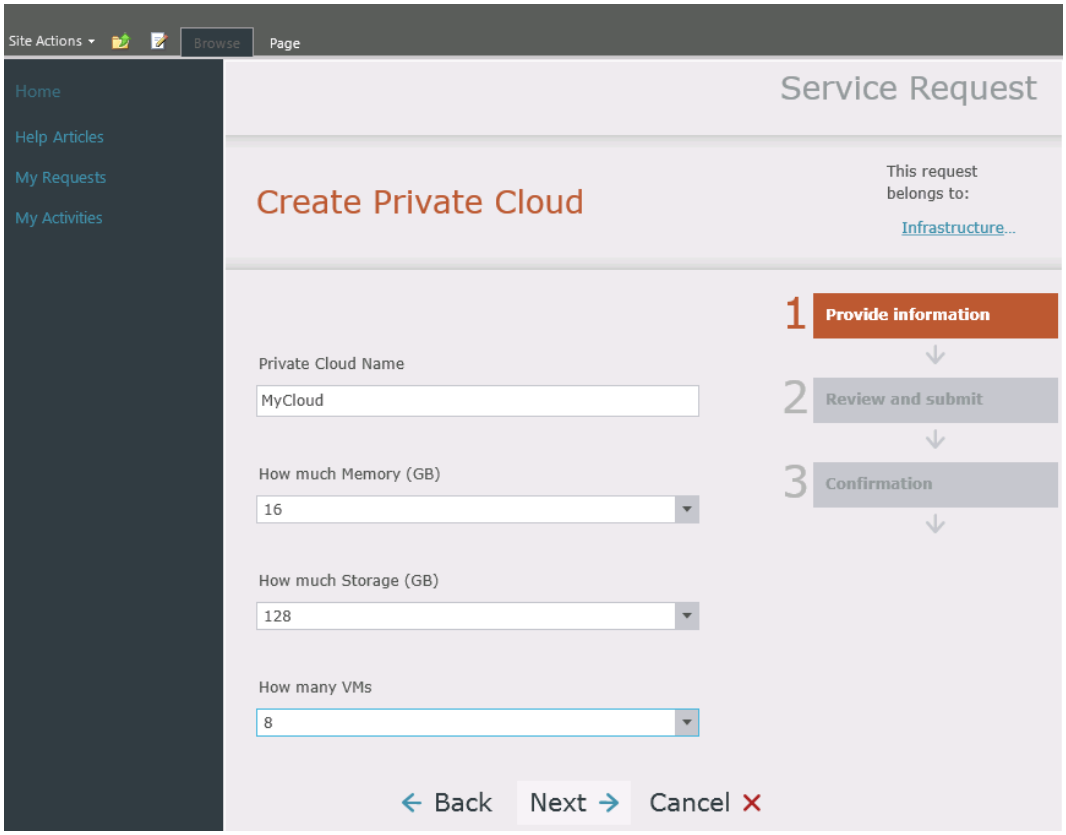
2. As Jeff only has a single Request offering created and published, this will be presented on the **Infrastructure Services** page. Here Jeff will select this offering **Create Private Cloud**



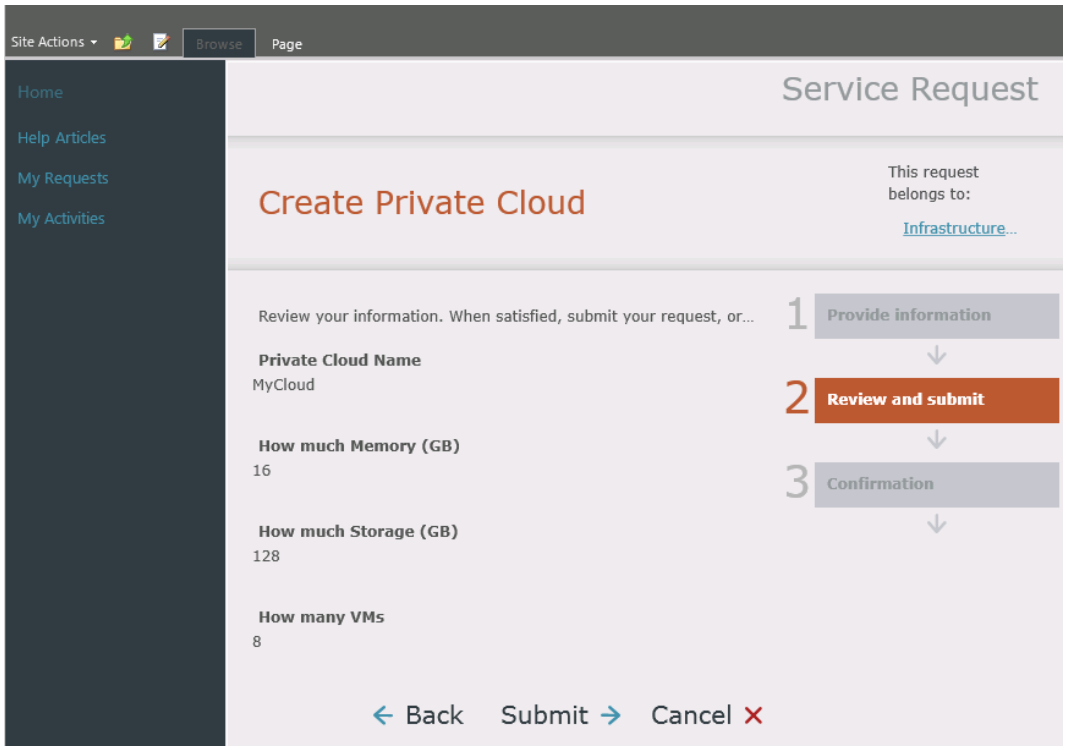
3. On the Request offering page, select **Go to request form**



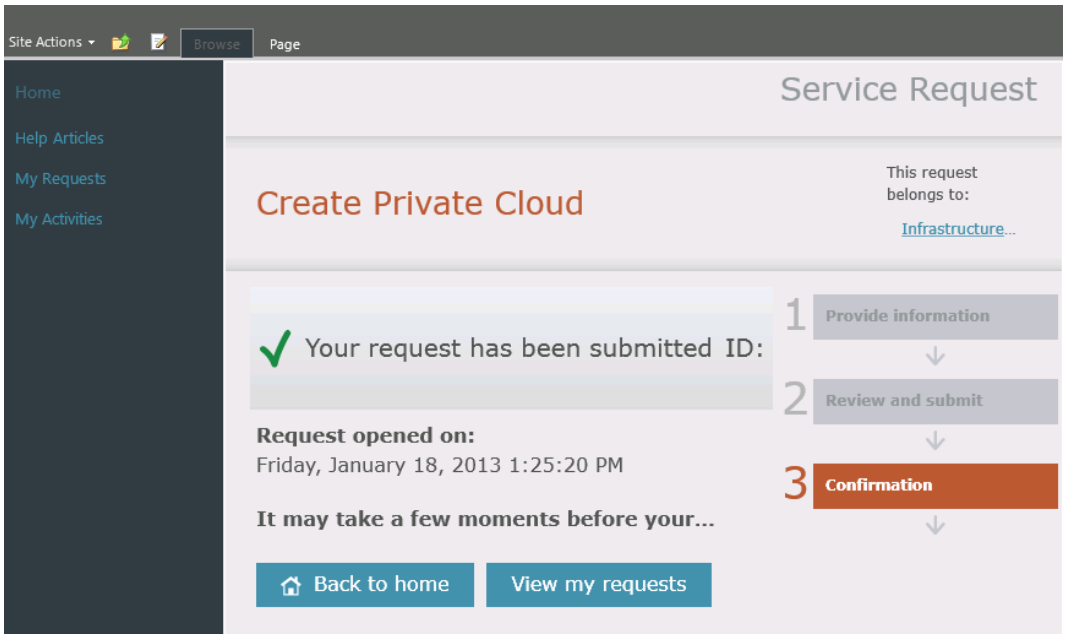
4. Our four user prompts which we created on the request form are now presented, and we can proceed to fill these out as follows. In the **Private Cloud Name** text box enter **MyCloud**, from the **How much Memory (GB)** list select **16**, from the **How much Storage (GB)** list select **128**, and from the **How many VMs** list select **8**. Then we can select **Next**



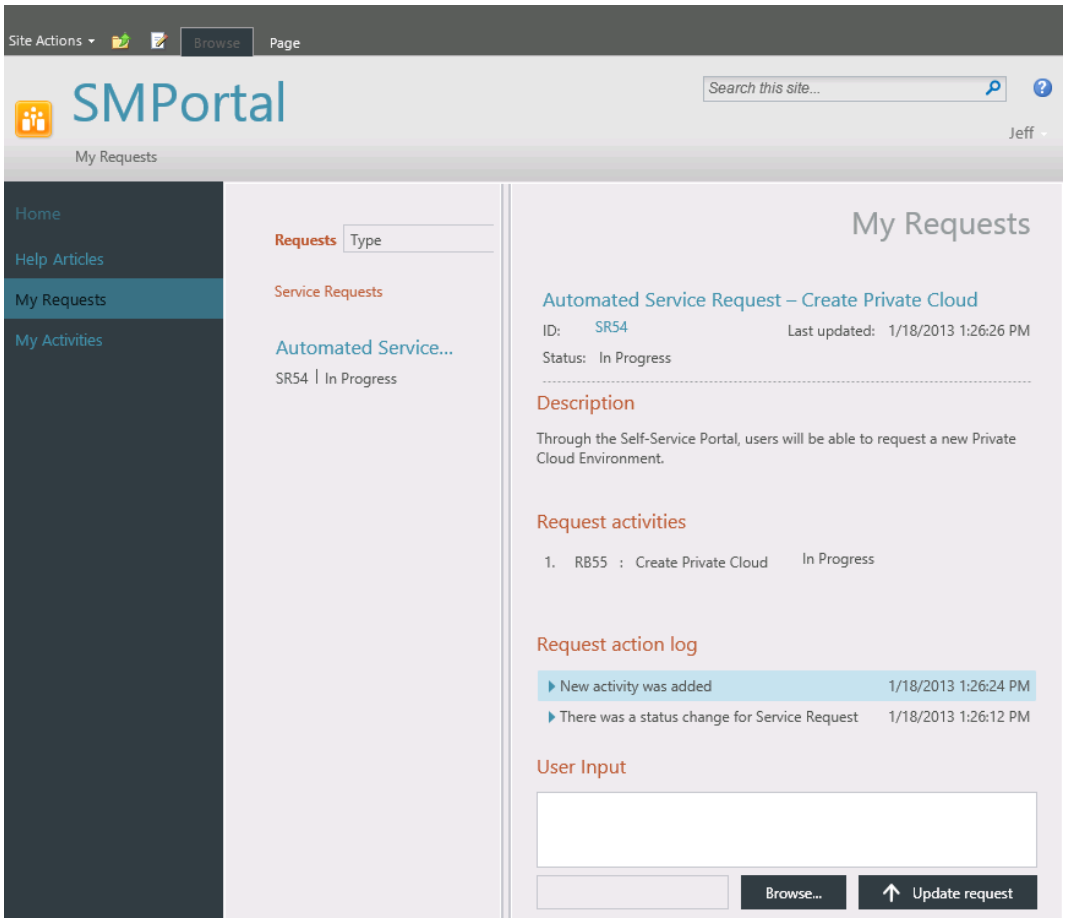
5. On the **Review and Submit** page, we can double check our information prior to selecting **Submit**



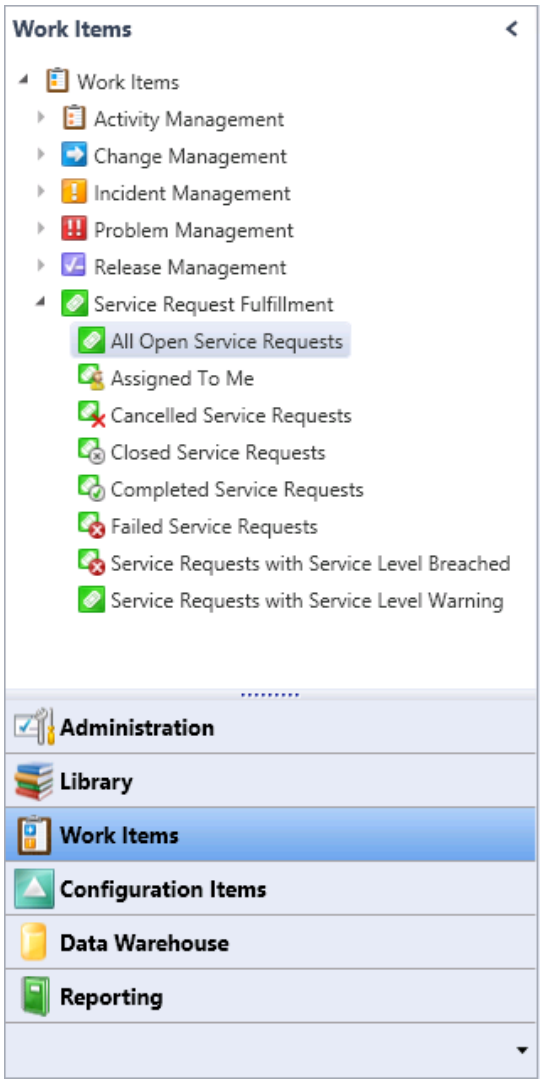
6. The **Confirmation** page will then be presented where we now have the option to select **View my requests**



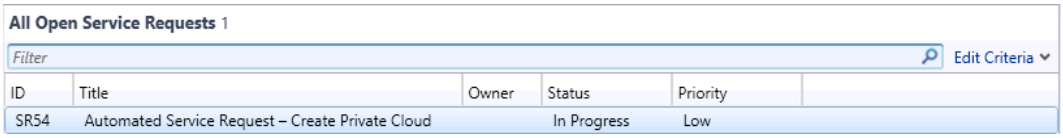
7. After Selecting the **View My Requests** button, we can observe that our request is currently in progress.



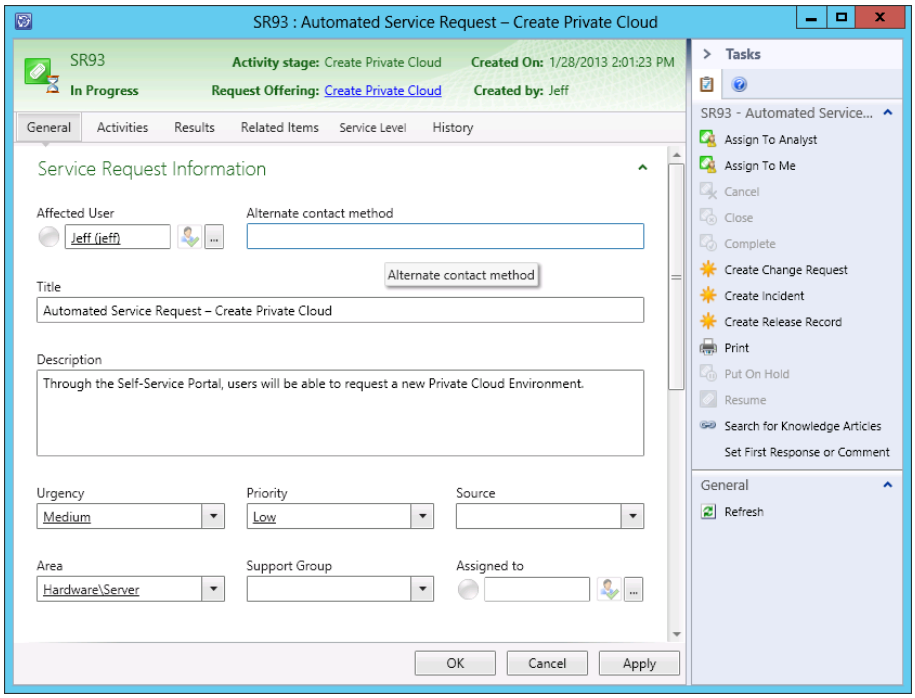
8. Switching back to the **Service Manager Console,** Jeff can see the newly created service request by ensuring the **Work Item** workspace selected, and expanding the navigation tree to select **Service Request Fulfillment** > **All Open Service Requests**



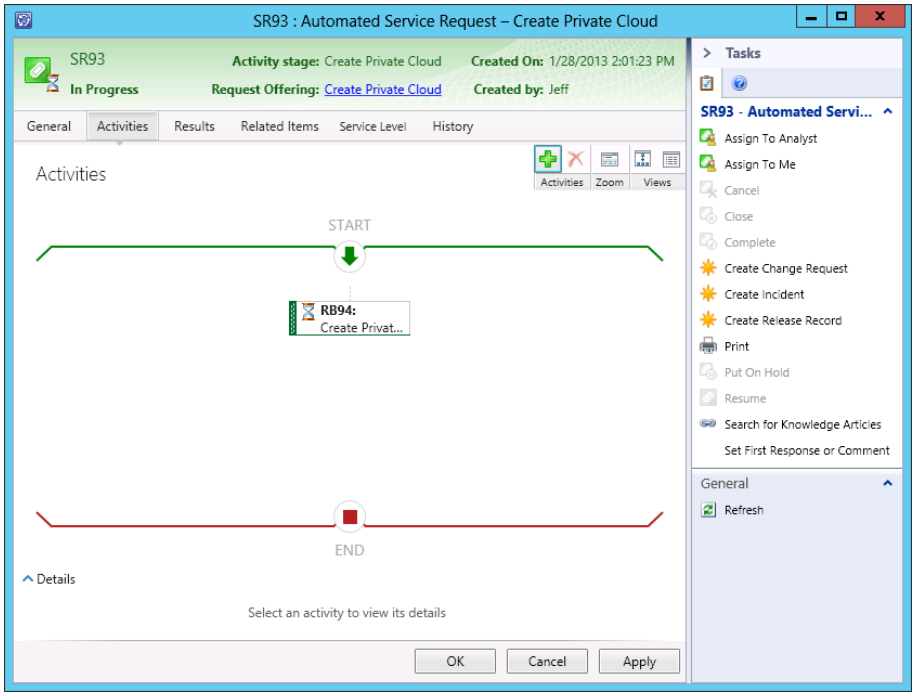
9. In the **All Open Service Requests**, the ticket created is now presented, matching the ID which was reported on the **My Requests** page in the portal. We can now double-click this request to **open** it



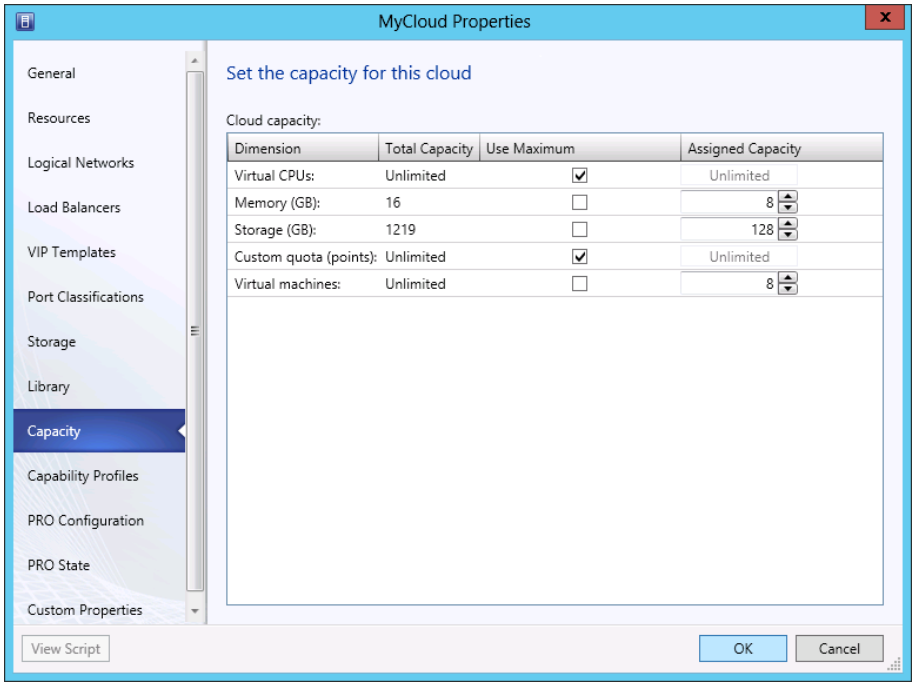
10. On the **general** page we can see that all the fields which we defined in the template are now completed as we provided



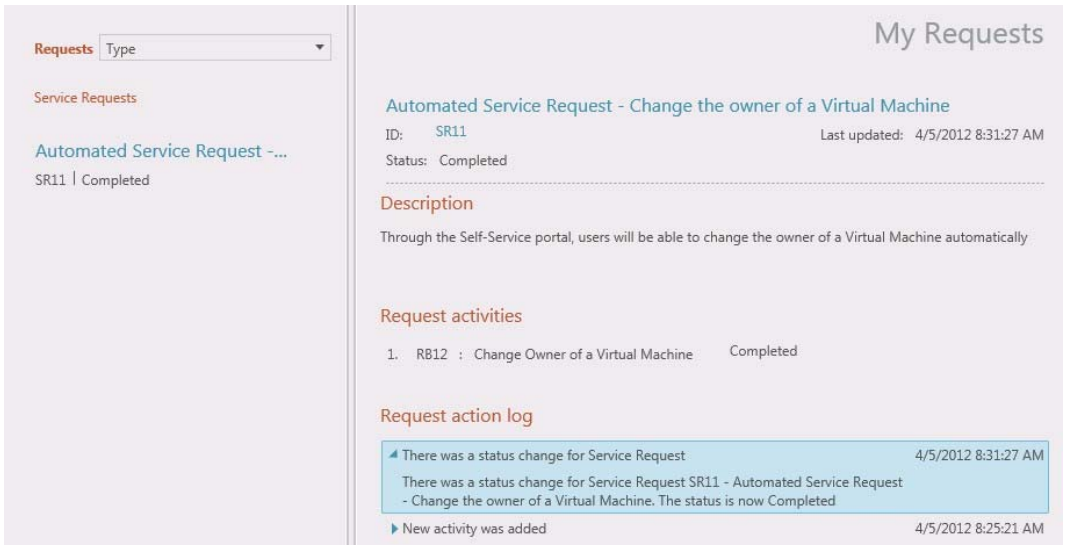
11. When **Jeff** selects the **Activities** tab we can see that the Runbook activity is already in progress.



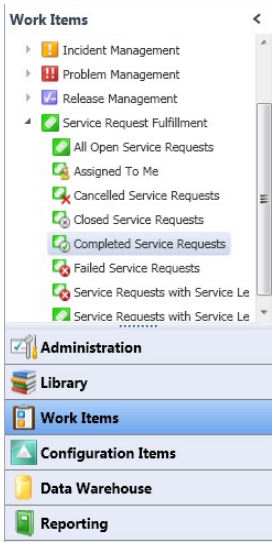
12. Switching over to the **Virtual Machine Manager console**, Jeff can see a new cloud called **MyCloud** has been created, exploring the properties of the new cloud will show the allocated resources requested.



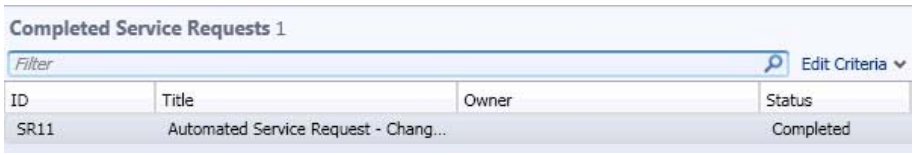
13. Switching back again to the Self-Service Portal, Jeff refreshes the request to see that its status is now also updated, and marked as complete**.**



14. In the Service Manager console, ensuring the **Work Item** workspace is selected, and expanding the navigation tree to this time select **Service Request Fulfilment** > **Completed Service Requests**



15. Jeff can now see his **Service Request** is indeed presented and its status updated to read **Completed**.

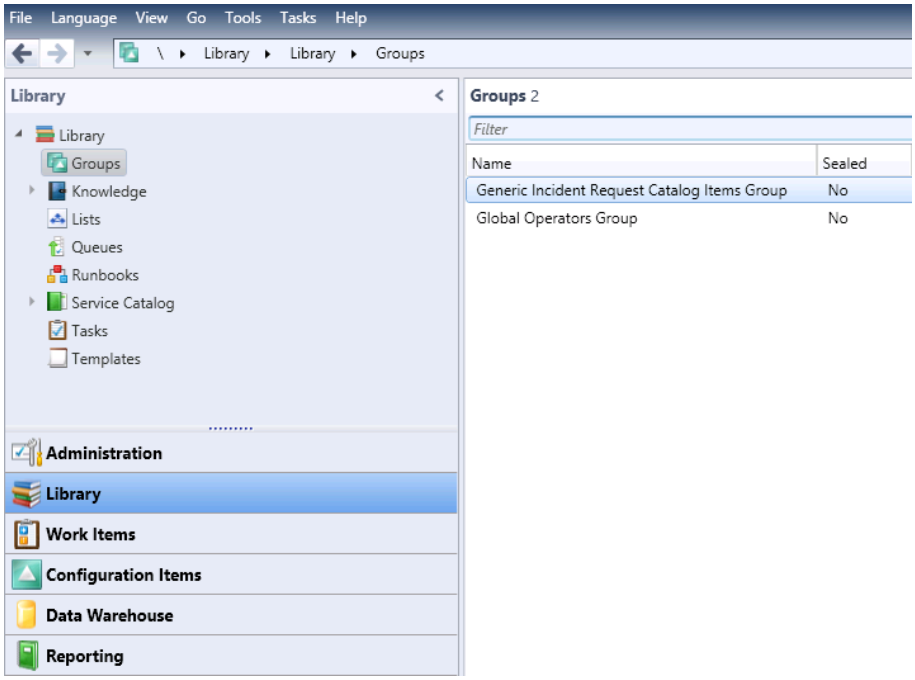


Using the Service Request in Production

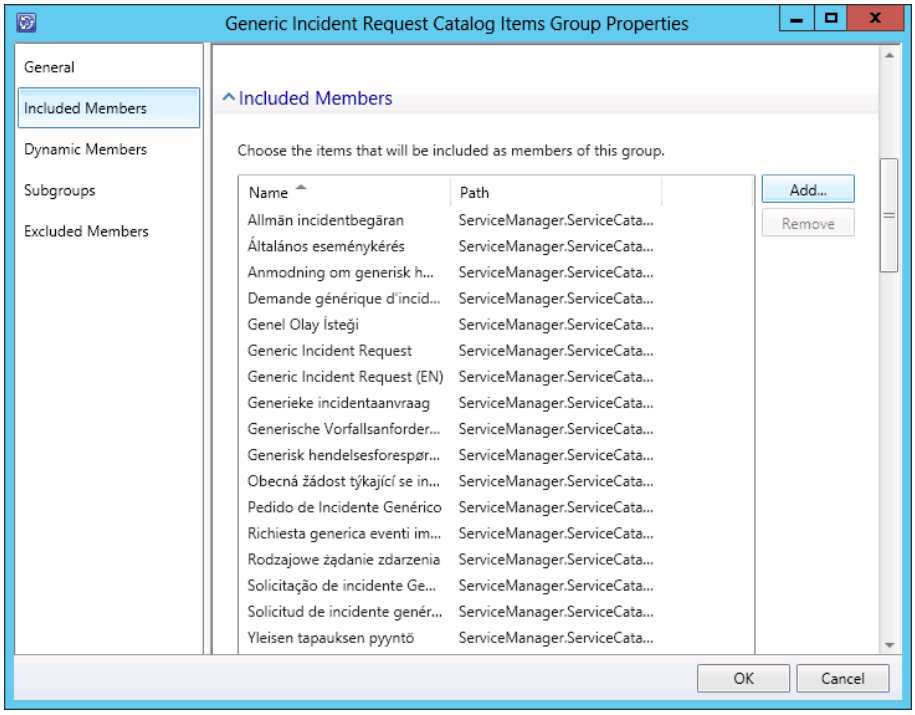
Now that Jeff has verified that the automated service request process for creating a private cloud is working as expected he notifies Emily so that she can now create a new development environment using the Self Service Portal.

Before **Emily** can access the Self-Service Portal **Jeff** will need to add the **Create Private Cloud** request offering, and the **Infrastructure Services** service offering to the **Generic Incident Request Catalog Items Group**.

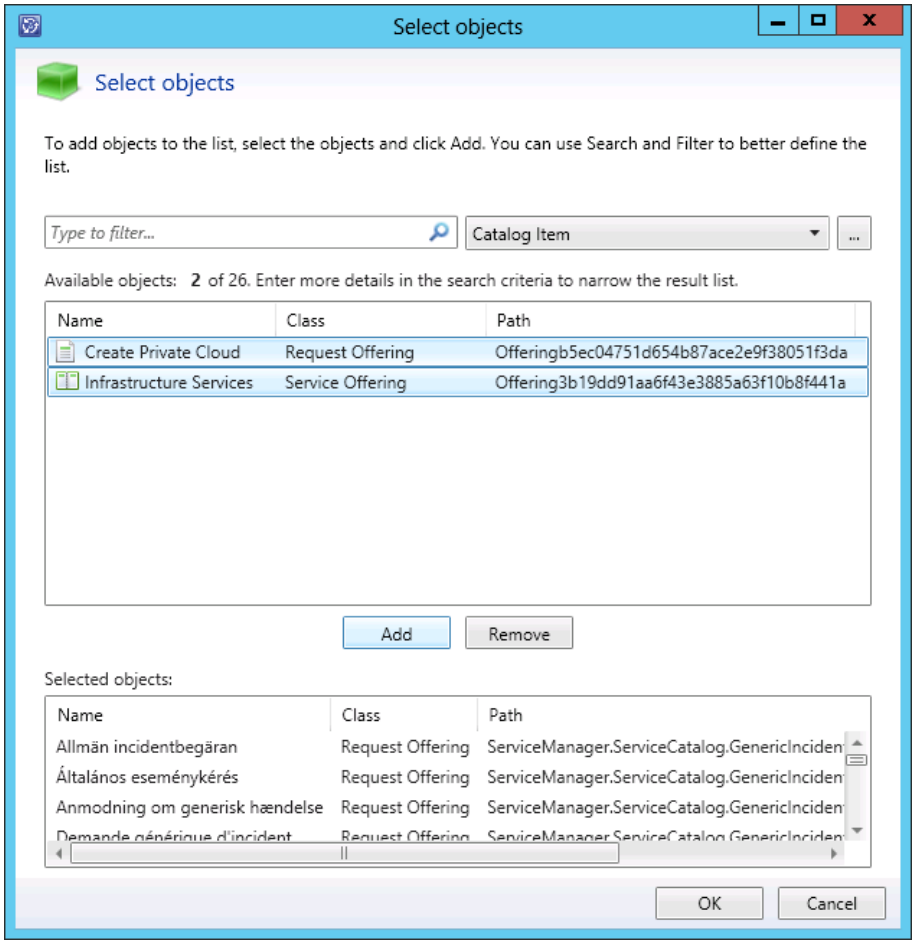
1. On the Service manager console on **ServiceMgr**, **Choose Library** and click on **Groups,** select **Generic Incident Request Catalog Items Group,** then in the **Tasks** pane, click on **Properties**



2. Click **Included Members**, followed by **Add**



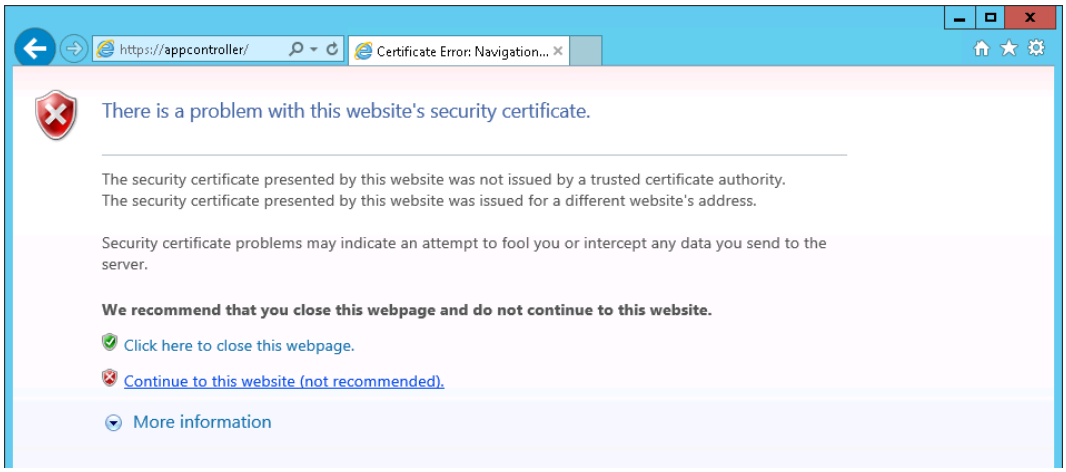
3. Add **Create Private Cloud**, and **Infrastructure Services,** then click **OK**, followed by **OK** on the **Generic Incident Request Catalog Items Group** properties screen**.**



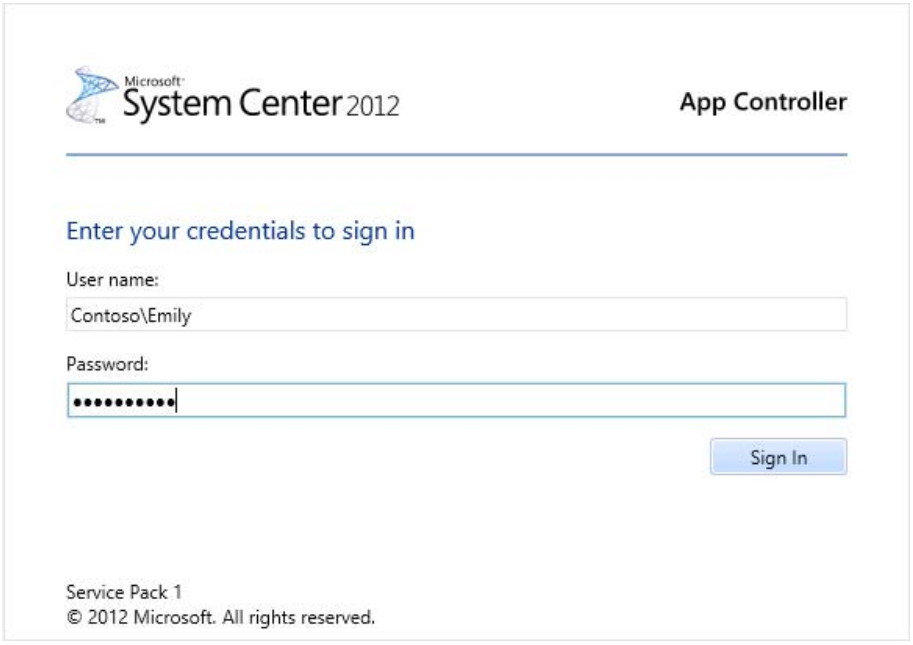
Testing the Self-Service Portal with a user

Jeff is now ready for **Emily** to connect to the **Self-Service Portal** and create her own private cloud.

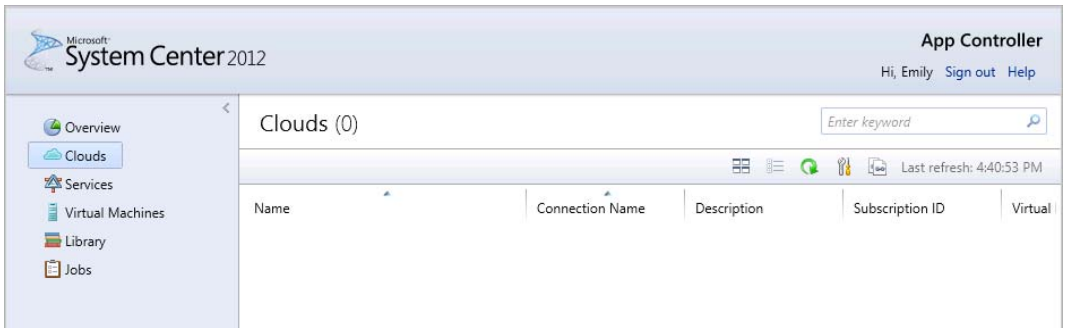
1. Open the App Controller portal by browsing to https://appcontroller . When prompted with the certificate warning, click **Continue to this website** and add the site to the trusted site list.



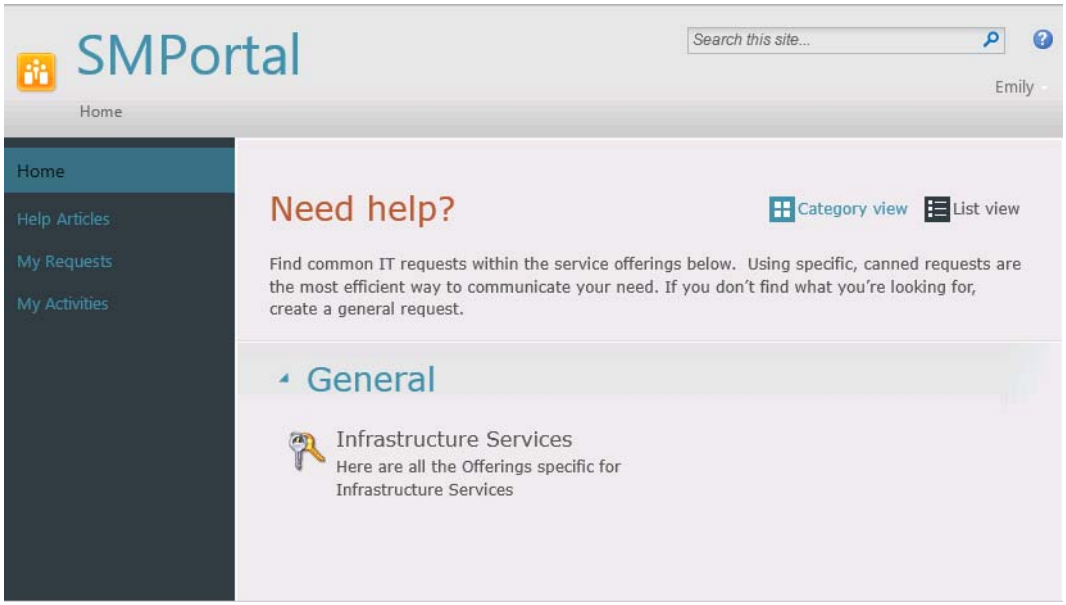
2. At the login screen enter the user name **contoso\emily** and relevant password and click **Sign In**



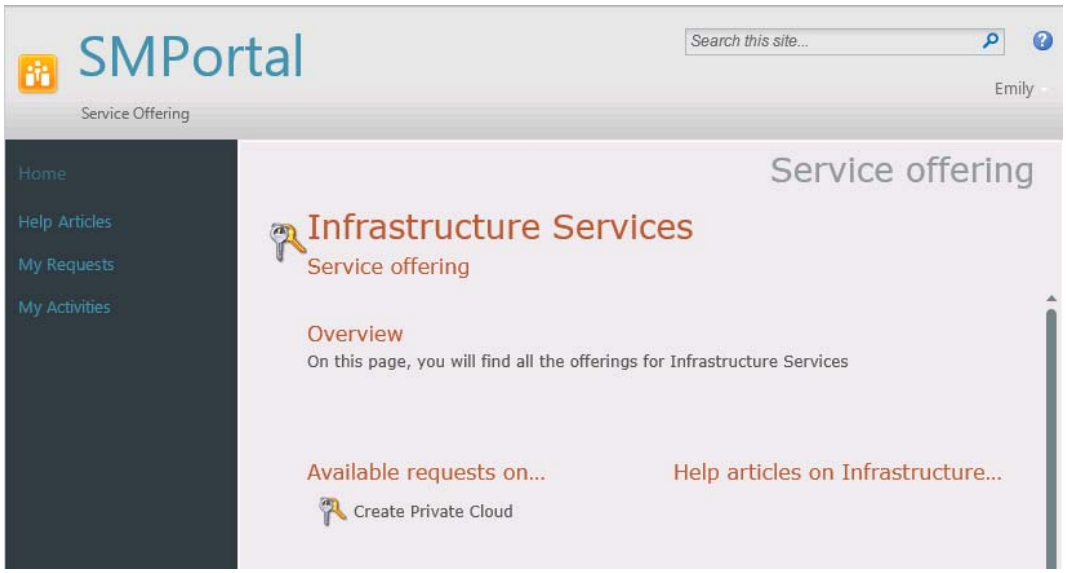
3. Click **Clouds** from the navigation pane, Emily will see that there are no Clouds available for her.



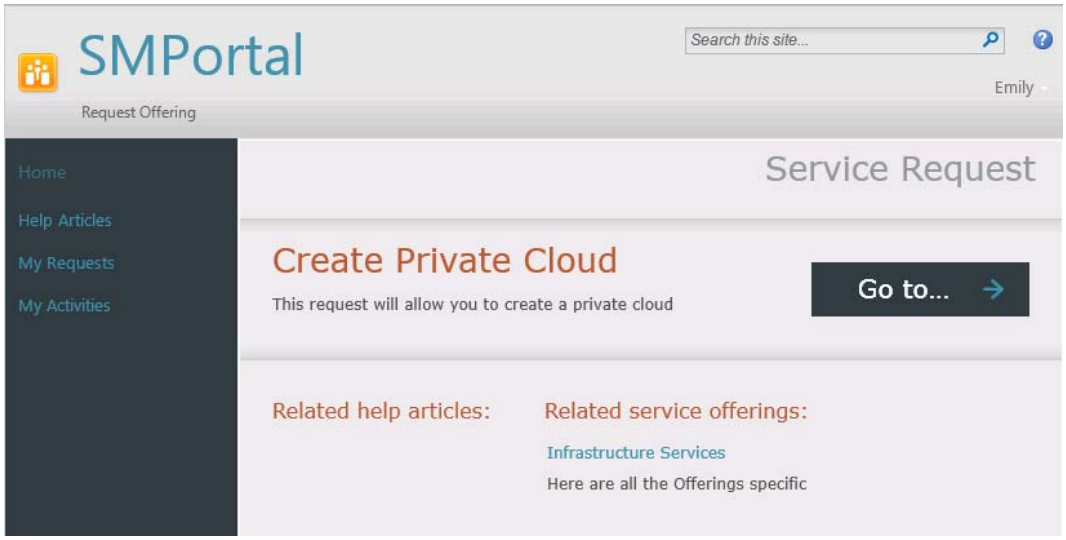
4. Open **http://Sharepoint:82/SMPortal** and login as Emily. After a few moments the portal will be presented, and Emily can select **Infrastructure Services** from the category **General**



5. Select the **Create Private Cloud** offering

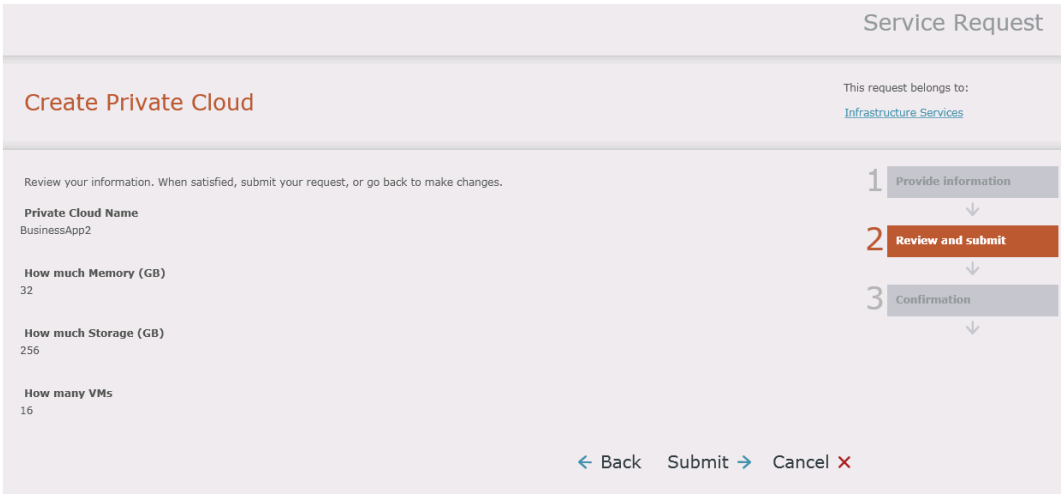


6. On the Request offering page, Create Private Cloud select to **Go to request form**

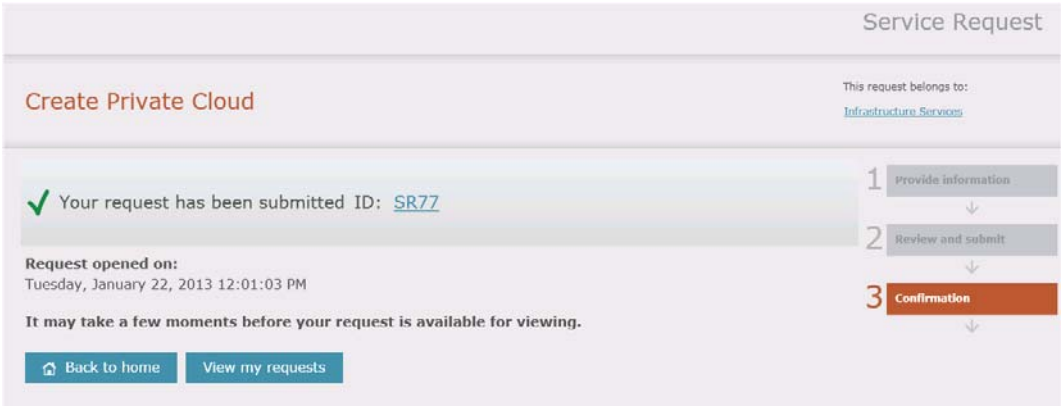


7. Our four user prompts which we created on the request form are now presented, and we can proceed to fill these out as follows. In the **Private Cloud Name** text box enter **BusinessApp2**, from the **How much Memory (GB)** list select **32**, from the **How much Storage (GB)** list select **256**, and from the **How many VMs** list select 16. Then we can select **Next**

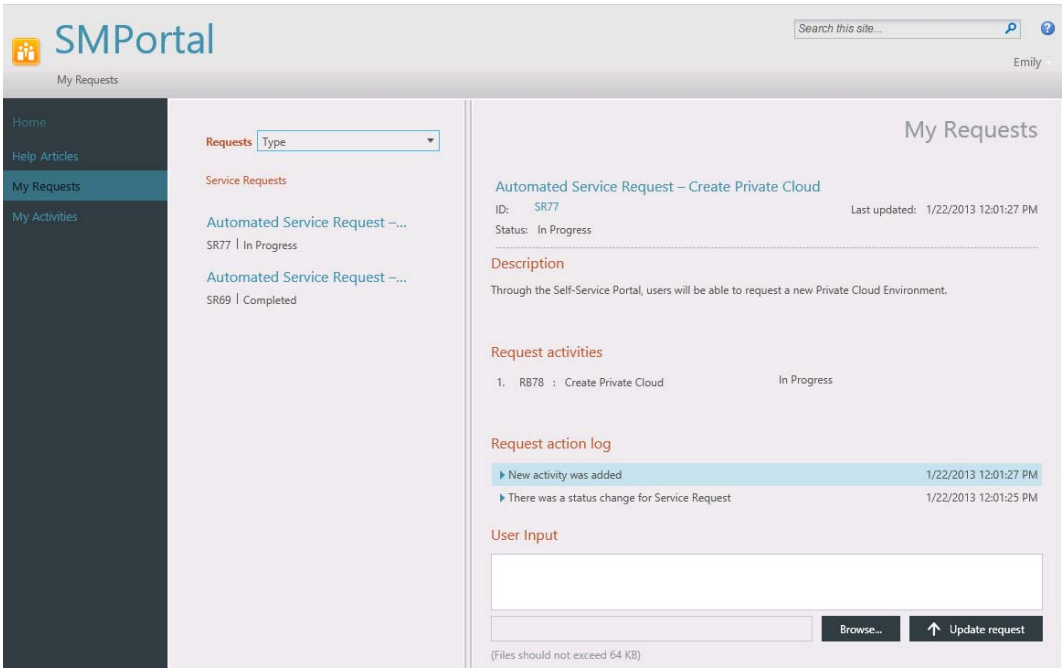
8. On the **Review and Submit** page, we can double check our information prior to selecting **Submit**



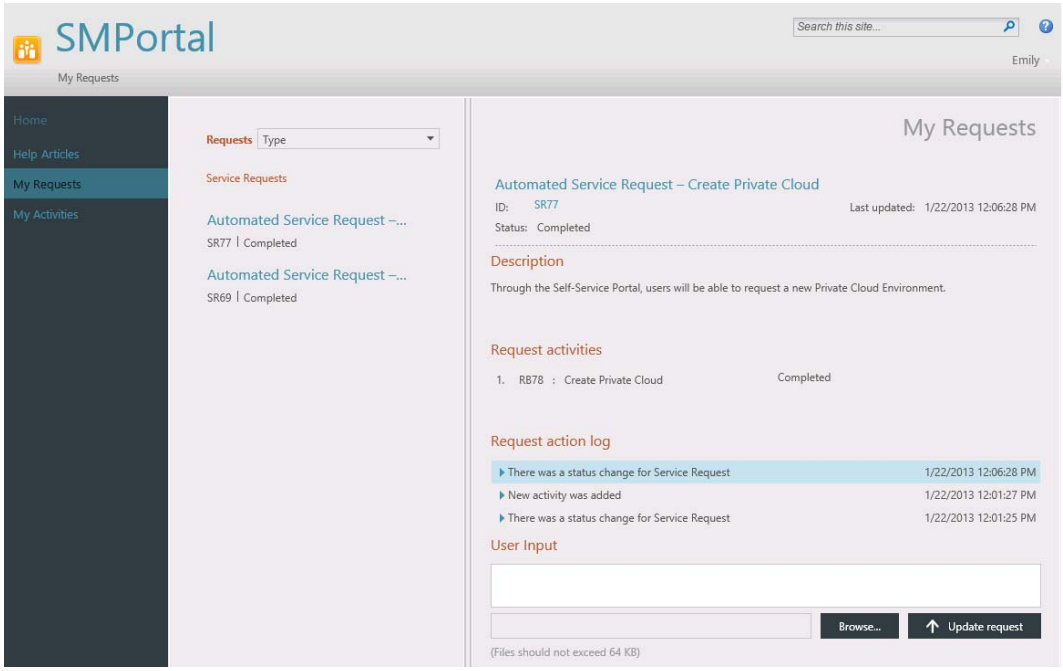
9. The **Confirmation** page will then be presented where we now have the option to select **View my requests**



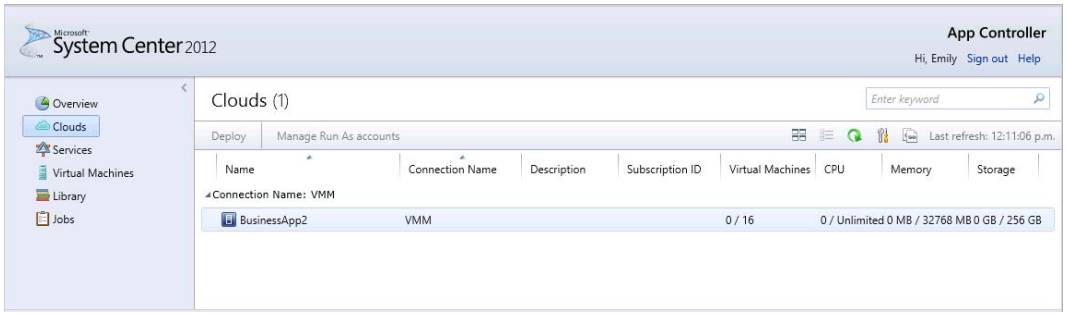
10. Selecting the **View My Requests** button, we can observe that our request is currently in progress.



11. After a few minutes Emily refreshes the request to see that its status is now also updated, and marked as complete.



12. Emily returns to the App Controller portal and clicks the **Refresh** button from the toolbar in the Clouds pane. Emily sees the **BusinessApp2** cloud, and clicking the **Show items as cards** from the toolbar displays the available resources for the private cloud.



Scenario: Provision storage to increase cloud capacity

Jeff decides that he would like to implement a flexible and vendor neutral storage solution that he can use for future provisioning of private clouds. Using the SMI-S provider that is included Virtual Machine Manager, Jeff is able to connect to an iSCSI target and provision his storage as and when the need arises.

Install SMI-S target Server on HyperV02

Prerequisites

- Install the update described in Microsoft KB article 27558246:

  http://www.microsoft.com/en-us/download/details.aspx?id=34851
  This update contains WMI-related changes to iSCSI Target Server that improve VMM discovery performance.

- Install the SMI-S provider, using the iSCSI Target SMI-S Provider Setup wizard.

  The SMI-S provider is included with the System Center 2012 Virtual Machine Manager SP1 installation. You can find the setup file on the installation CD under the path:**\amd64\Setup\msi\iSCSITargetSMISProvider.msi**, or alternatively on the VMM server under **\Program Files\Microsoft System Center 2012\Virtual Machine Manager\setup\msi\iSCSITargetProv\iSCSITargetSMISProvider.msi**. Complete the wizard to install the SMI-S provider on the iSCSI Target Server computer.

1. Load the PowerShell console on **HyperV02** and run the following command:

```
Add-WindowsFeature FS-IscsiTarget-Server
```

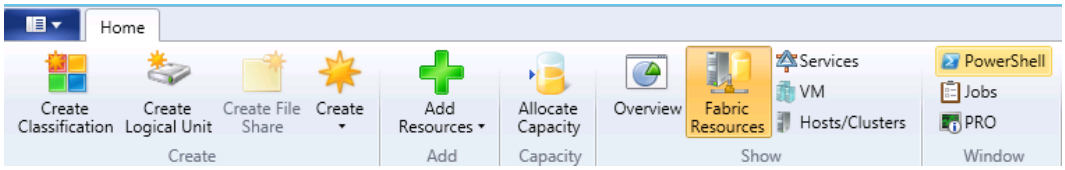
This will install the **iSCSI Target feature** to HyperV02. A reboot will not be required.

```
PS C:\Users\Administrator> Add-WindowsFeature fs-iscsitarget-server

Success Restart Needed Exit Code      Feature Result
------- -------------- ---------      --------------
True    No             NoChangeNeeded {}

PS C:\Users\Administrator> _
```

Add a storage provider on HyperV02

Jeff has just configured HyperV02 as an iSCSI target, he will use this target as the storage destination for his private cloud solution. Once the iSCSI target has been configured, Virtual Machine Manager can be connected to this target via SMI-S. In this scenario the iSCSI target is HyperV02, alternatively this could be any other SMI-S compatible storage solution.

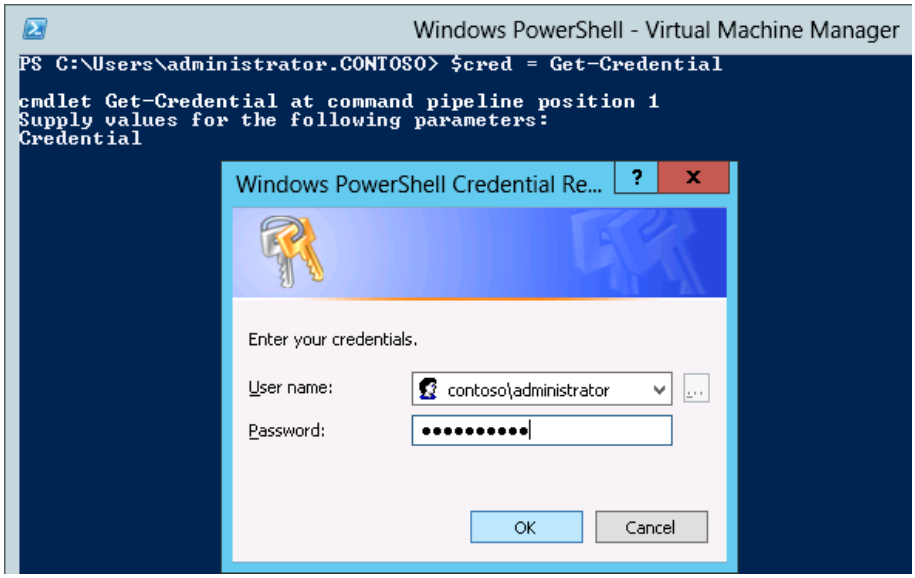
1. From the server **VMM**, open the **VMM Management Console** as an administrator, and then run **PowerShell** from the console.



2. Type the following command into the console:

```
$cred = Get-Credential
contoso\administrator
```

*Note that any account that is part of the Local Administrators group is sufficient.



3. Create a RunAs account in VMM as follows:

```
$runas = New-SCRunAsAccount –Name “iSCSIRunas” –Credential $cred
```

```
$runas = New-SCRunAsAccount -Name "iSCSIRunas" -Credential $cred
```

4. Add the storage provider as follows:

```
Add-SCStorageProvider -Name "Microsoft iSCSI Target Provider" -
RunAsAccount $Runas -ComputerName "HyperV02" –AddSmisWmiProvider
```

```
Add-SCStorageProvider -Name "Microsoft iSCSI Target Provider" -RunAsAccount $Runas -ComputerName "HyperV02" -AddSmisWmiProvider
```

View storage properties

1. Review the storage array attributes as follows:

```
$array = Get-SCStorageArray –Name “HyperV02”
$array.StoragePools
```

2. This final command will display the list of iSCSI Targets configured on HyperV02. Jeff notices that the results correlate with the logical drives on **HyperV02**, he takes note of an iSCSI target name that maps to a logical drive with at least 50GB of space free. In this example Jeff will use the iSCSI target: “**iSCSITarget: HyperV02: D:**”

```
PS C:\Users\administrator.CONTOSO> $array = Get-SCStorageArray -Name "HyperV02"
PS C:\Users\administrator.CONTOSO> $array.StoragePools
```

Add pools from iSCSI Target Server for VMM Management

1. Jeff will populate the following PowerShell command with the iSCSI Target that he determined in the previous step:

```
$pool = Get-SCStoragePool –Name “iSCSITarget: HyperV02: D:”
```

```
$pool = Get-SCStoragePool -Name "iSCSITarget: HyperV02: D:
```

2. Jeff then creates a new storage classification called “Gold”:

```
$class = New-SCStorageClassification –Name “Gold”
```

```
$class = New-SCStorageClassification -Name "Gold"
```

3. He then adds the Storage Pool to VMM:

```
Set-SCStorageArray –AddStoragePoolToManagement $pool –StorageArray
$pool.StorageArray –StorageClassification $class
```

```
Set-SCStorageArray -AddStoragePoolToManagement $pool -StorageArray $pool.StorageArray -StorageClassification $class
```

4. Finally he allocates the newly created storage pool to the “All Hosts” host group

```
Set-SCStoragePool -StoragePool $pool -AddVMHostGroup (Get-
SCVMHostGroup -Name "All Hosts")
```

```
Set-SCStoragePool -StoragePool $pool -AddVMHostGroup (Get-SCVMHostGroup -Name "All Hosts")
```

Install Multipath-IO on HyperV03 and HyperV04

Before Jeff can connect to the iSCSI target on **HyperV02** he will need to install the **Multipath-IO** feature on **HyperV03** and **HyperV04**:

1. From **HyperV03**, he opens PowerShell with elevated privileges and runs the command:

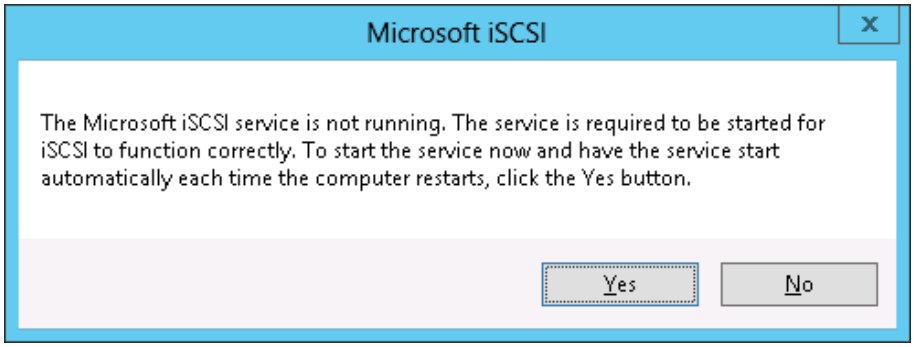
```
Add-WindowsFeature multipath-io
```

```
PS C:\Users\administrator.CONTOSO> Add-WindowsFeature multipath-io

Success Restart Needed Exit Code      Feature Result
------- -------------- ---------      --------------
True    No             Success        {Multipath I/O}
WARNING: Windows automatic updating is not enabled. To ensure that your newly-installed role or feature
is automatically updated, turn on Windows Update.
```

2. Jeff repeats the above step on **HyperV04**

```
PS C:\Users\administrator.CONTOSO> Add-WindowsFeature multipath-io

Success Restart Needed Exit Code      Feature Result
------- -------------- ---------      --------------
True    No             Success        {Multipath I/O}
WARNING: Windows automatic updating is not enabled. To ensure that your newly-installed role or feature
is automatically updated, turn on Windows Update.
```

3. Jeff also needs to run the **iSCSI initiator** on both **HyperV03** and **HyperV04** for discovery of the iSCSI target. Jeff is prompted that the service is currently not running by default and click **Yes** to start the required service.



Create a LUN

Jeff now needs to create a LUN which will be presented via the iSCSI Target.

1. Run the following command from Powershell on the server **VMM**:

```
$LUN = New-SCStorageLogicalUnit –Name “iSCSI1” –StoragePool $pool –
DiskSizeMB 50000
```

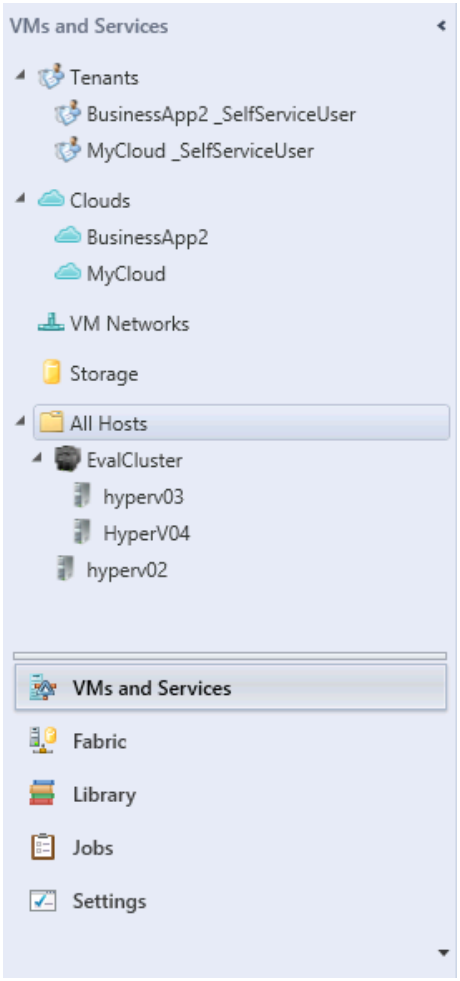
```
$LUN = New-SCStorageLogicalUnit -Name "iSCSI1" -StoragePool $pool -DiskSizeMB 50000
```

2. He then allocates the LUN to the **All Hosts** host group as follows:

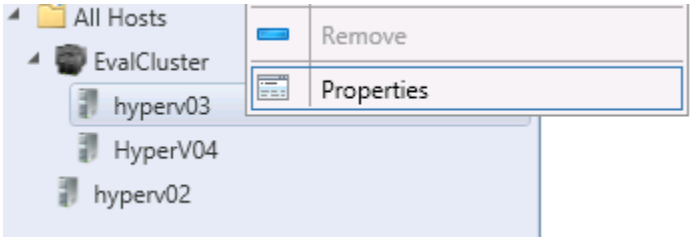
```
Set-SCStorageLogicalUnit –StorageLogicalUnit $LUN –VMHostGroup
(Get-SCVMHostGroup –Name “All Hosts”)
```

```
Set-SCStorageLogicalUnit -StorageLogicalUnit $LUN -VMHostGroup (Get-SCVMHostGroup -Name "All Hosts")
```

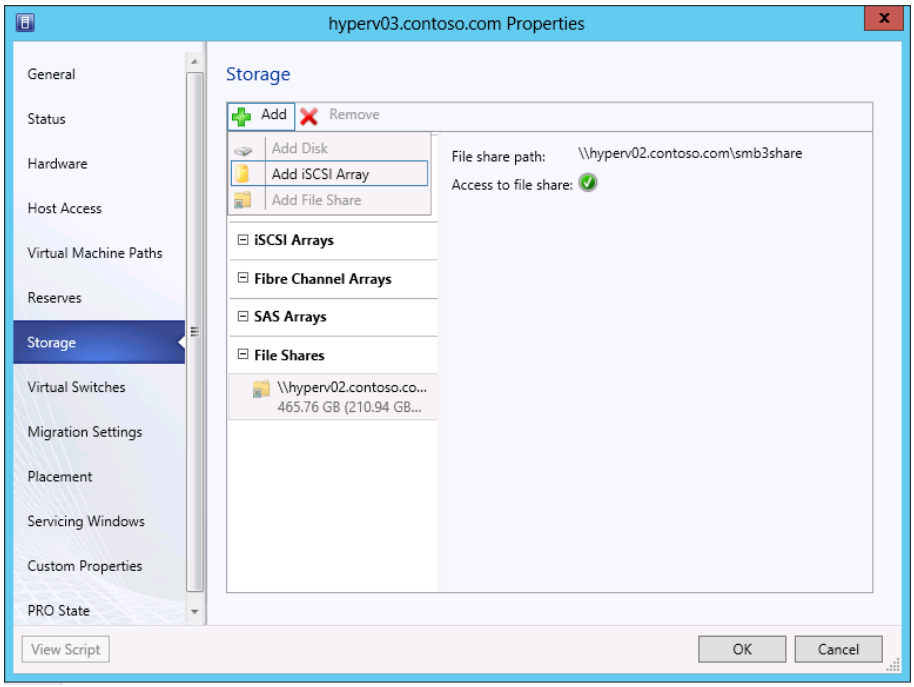
3. Jeff then goes into the Virtual Machine Manager console on **VMM** and clicks on the **VMs and Services** button and expands **All Hosts** and then **Evalcluster**.



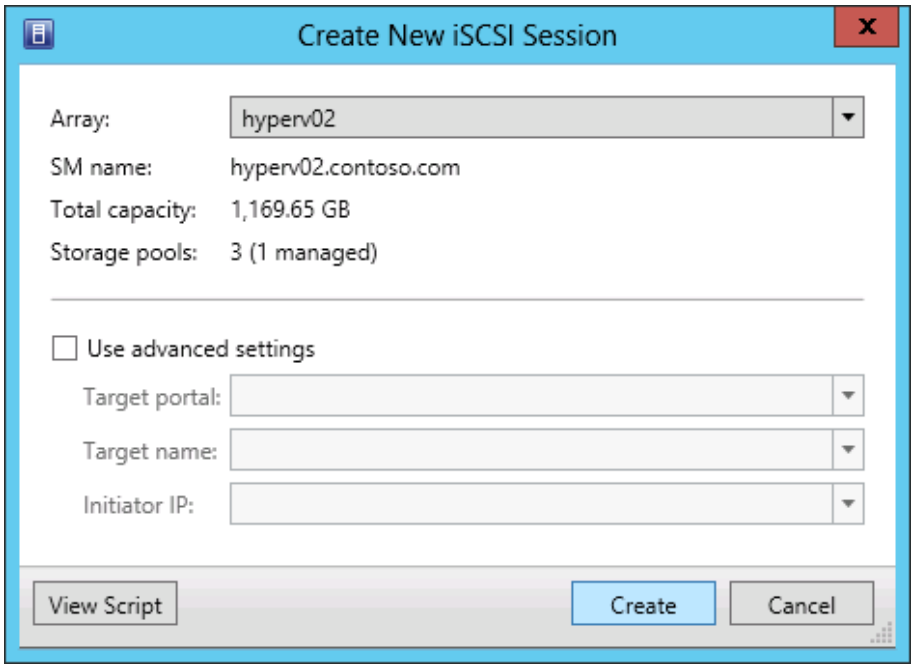
4. Jeff then Right clicks on the **HyperV03** server and clicks on **Properties**



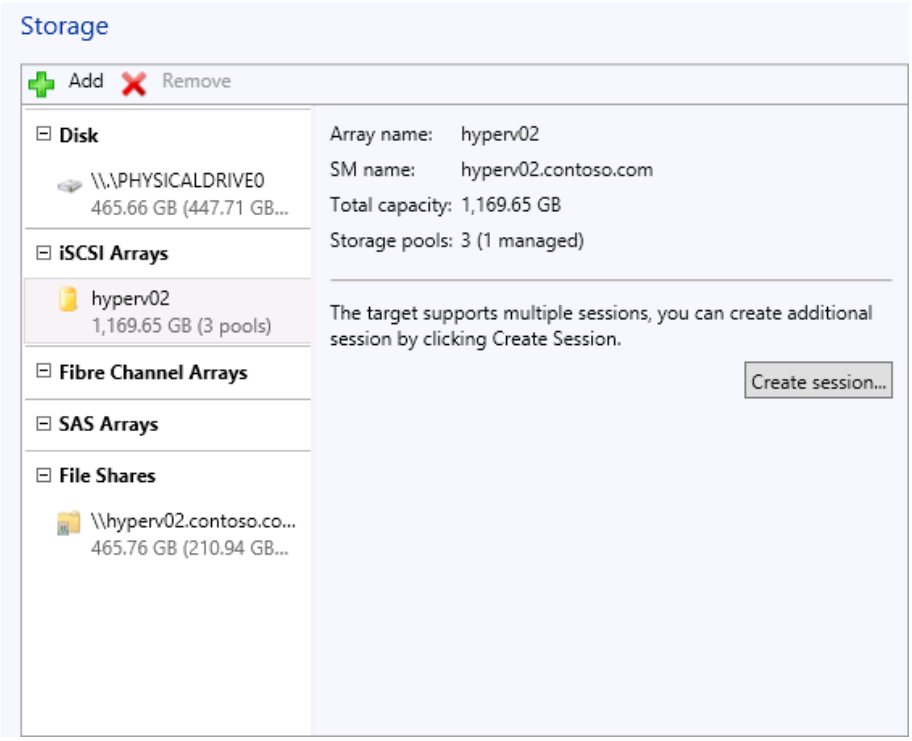
5. Choose the **Storage tab**, click **Add** at the top and choose **Add iSCSI Array**.



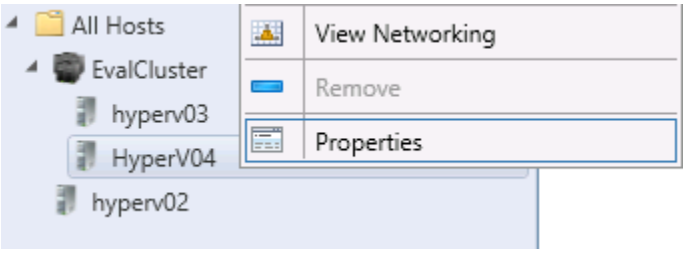
6. On the **Create New iSCSI Session** page, Jeff clicks on the drop down on the array and chooses **HyperV02** from the selection, and then clicks **Create**



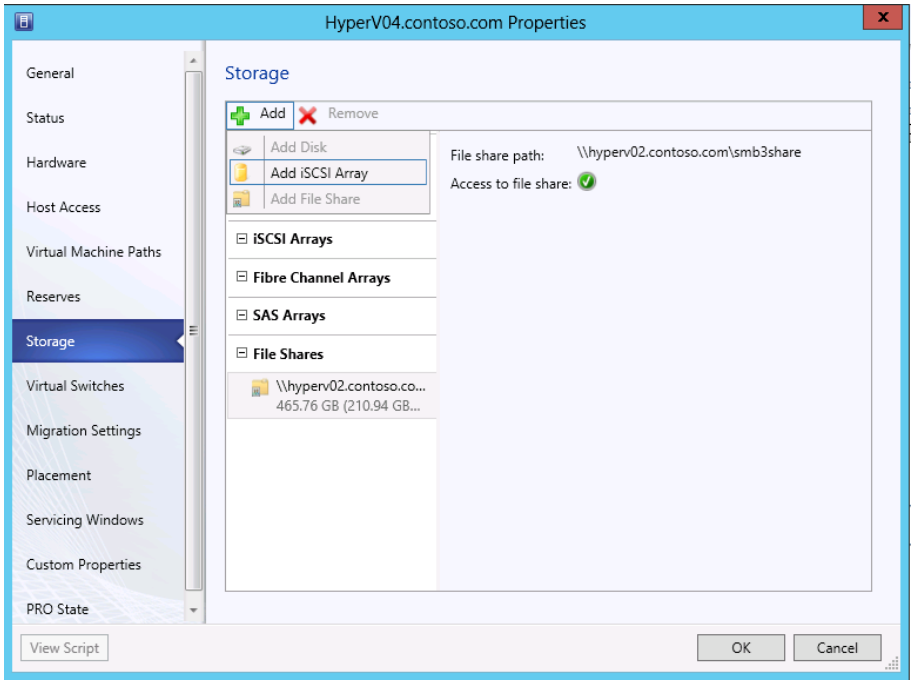
7. Jeff now sees that **HyperV03** now has an iSCSI session against the configured HyperV02 iSCSI Array. Click **OK.**



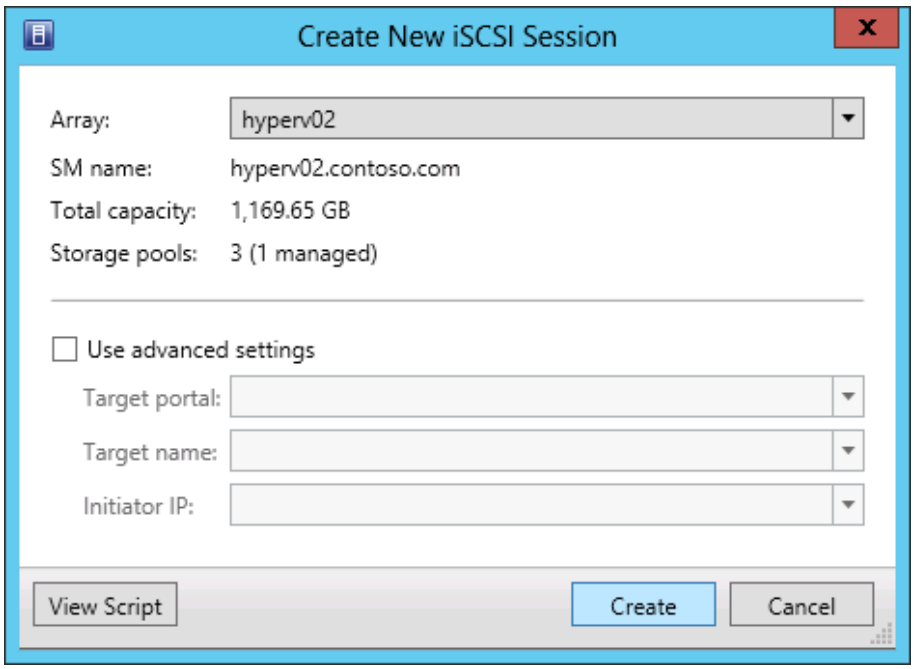
8. Jeff then Right clicks on the **HyperV04** server and clicks on **Properties**



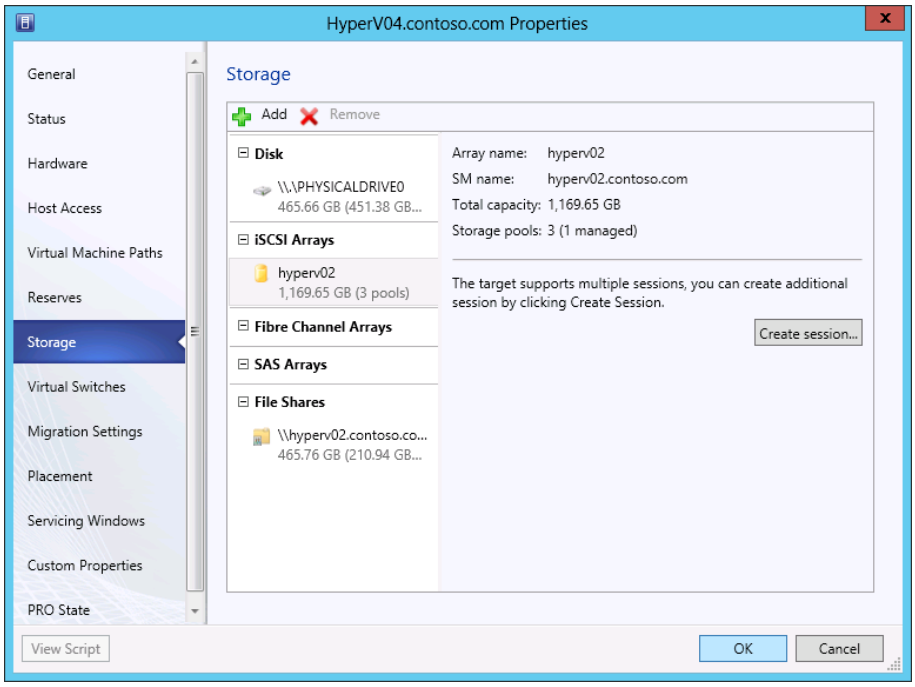
9. Choose the **Storage** tab, click **Add** at the top and choose **Add iSCSI Array**.



10. On the **Create New iSCSI Session** page, Jeff clicks on the drop down on the array and chooses **HyperV02** from the selection, and then clicks **Create**



11. Jeff sees that **HyperV04** now has an iSCSI session against the configured HyperV02 iSCSI Target.

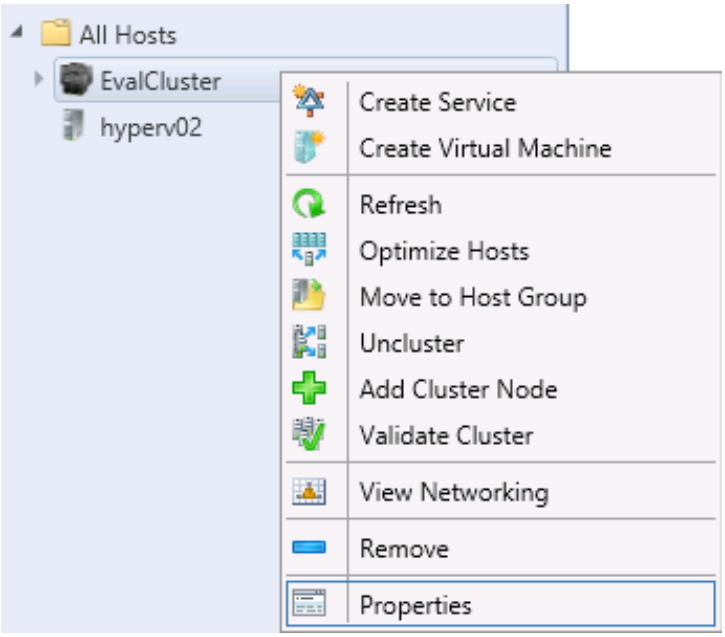


12. Next Jeff needs to create a **LUN** for the Cluster "**EvalCluster**"

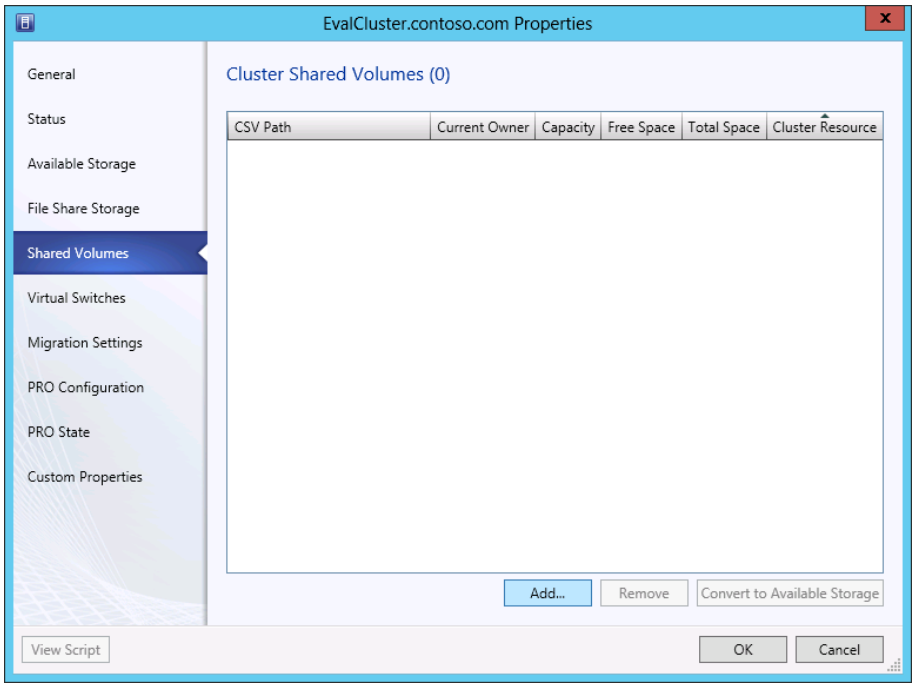
```
$host01 = Get-SCVMHostCluster –Name EvalCluster
Register-SCStorageLogicalUnit –StorageLogicalUnit $LUN –
VMHostCluster $host01
```

```
$host01 = Get-SCVMHostCluster -Name EvalCluster

Register-SCStorageLogicalUnit -StorageLogicalUnit $LUN -VMHostCluster $host01
```

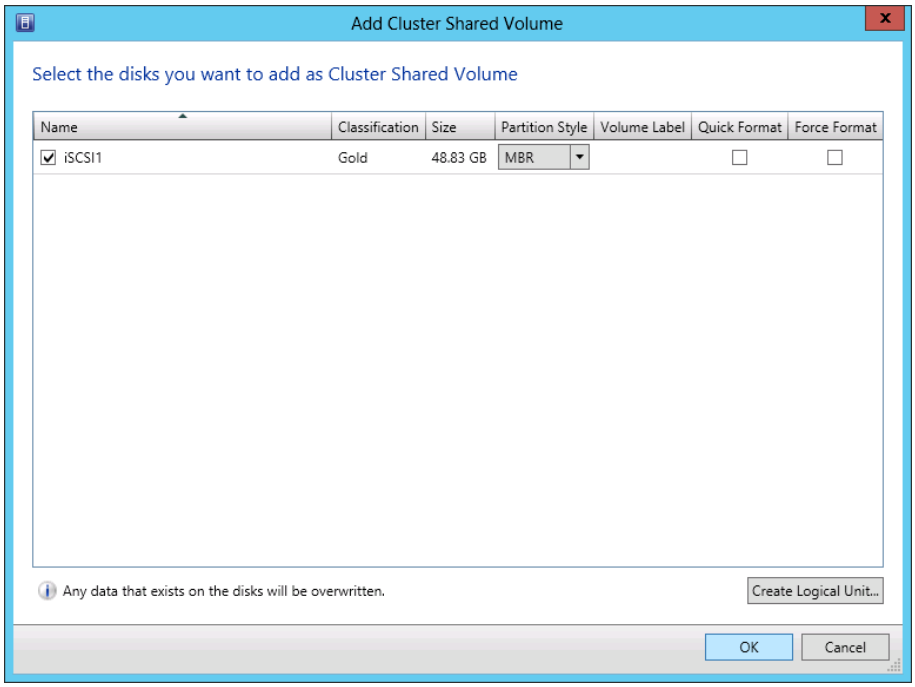
13. Jeff now runs **Virtual Machine Manager console** and from **VMs and Services** expands the **All Hosts**. He then right clicks on **EvalCluster** and chooses properties.



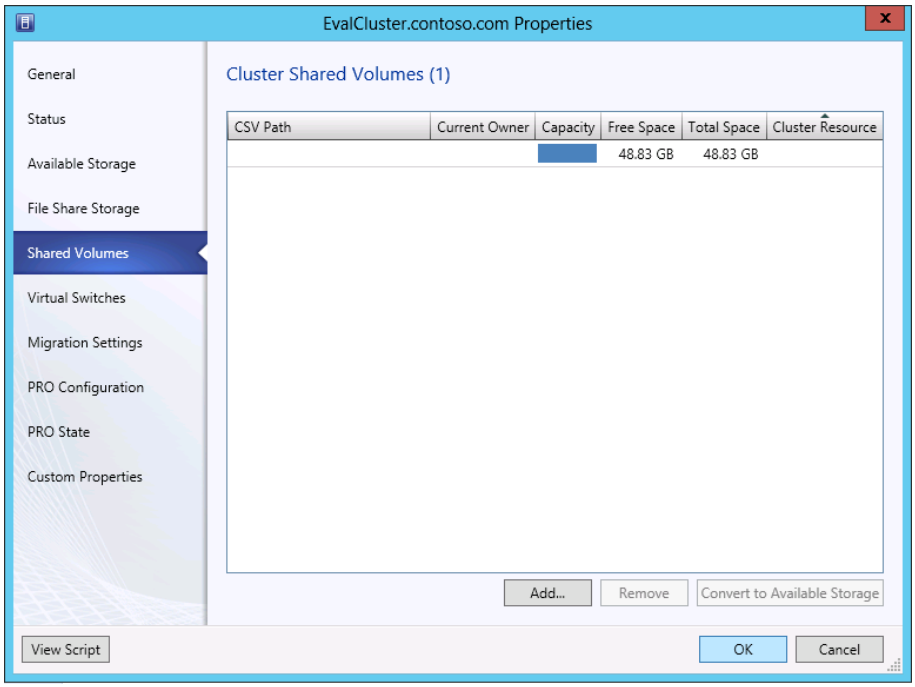
14. From the **EvalCluster** properties, click on **Shared Volumes** and then click **Add**



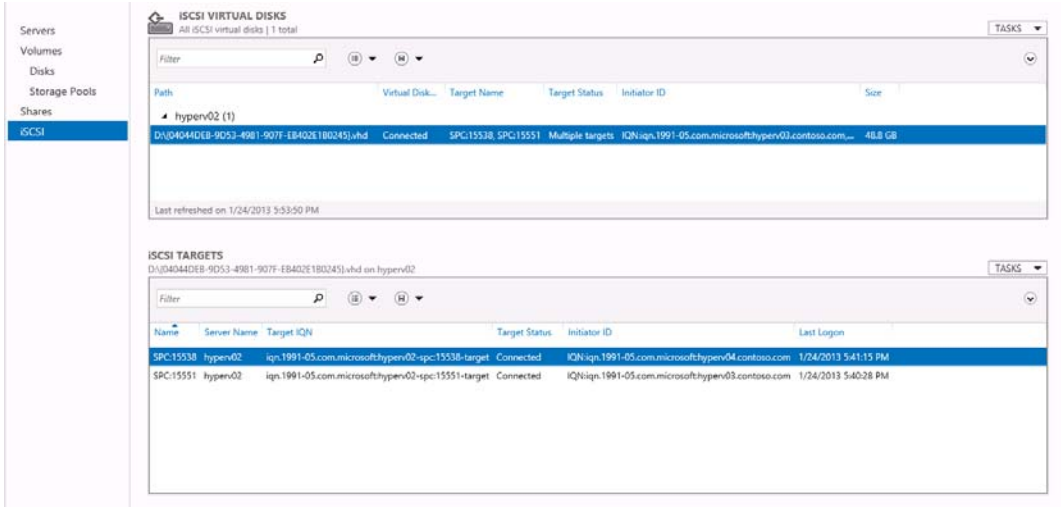
15. Tick on the **iSCSI1** target we created earlier to add the cluster shared volume and click **OK**.



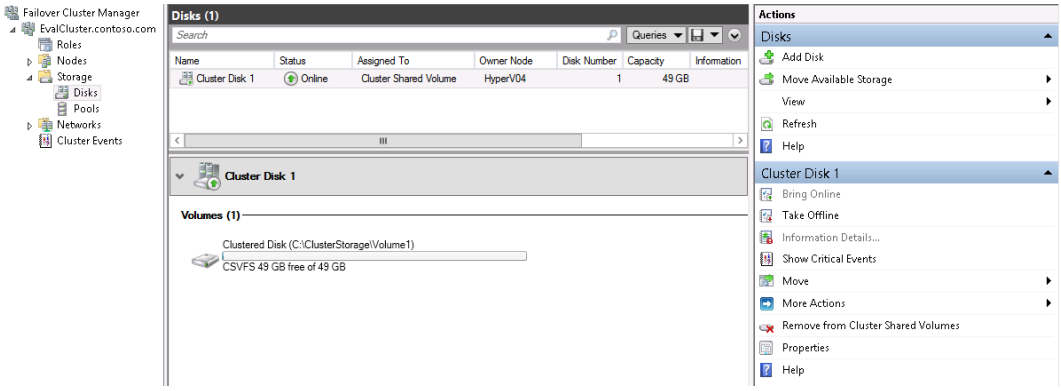
16. We now see that the iSCSI target has been added as a shared volume to the cluster, press **OK** now to exit.



17. Jeff now checks Server Manager on **HyperV02**. He opens **File and Storage Services** and checks the **iSCSI** status. We can see the **iSCSI VHD** we created and also the **EvalCluster** servers attached to the storage currently.



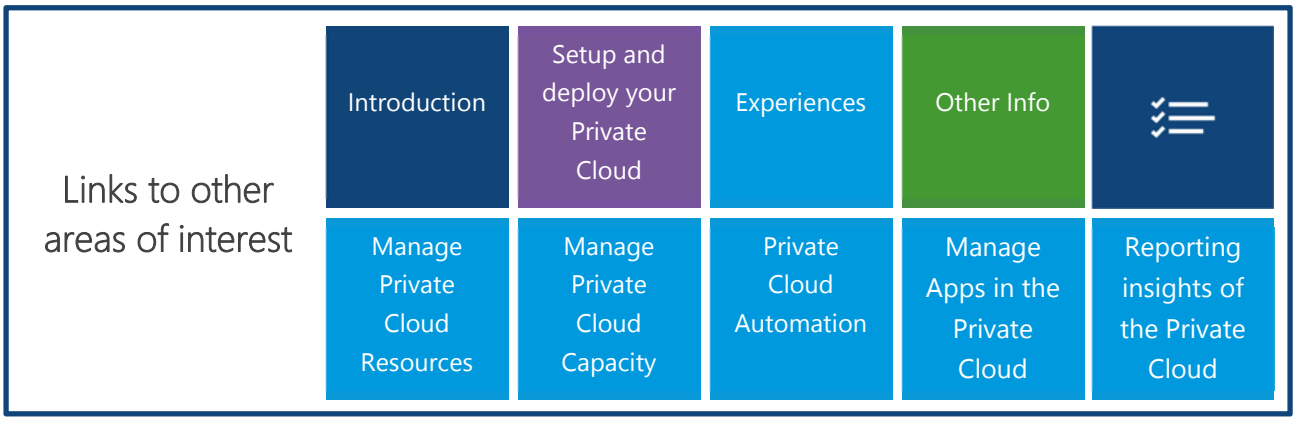
18. Jeff now navigates to either **HyperV03** or **HyperV04** and opens up **Failover Cluster Manager**. We can see that the Clustered Shared Volume is now available to all servers in the cluster. The change may not be reflected immediately and the disk may take 5 minutes to appear in Failover Cluster Manager.



19. If we navigate to **C:\ClusterStorage** we can access and use the **iSCSI** shared disk created on the **HyperV02** server.

Summary

At the end of this scenario Jeff has implemented a storage solution that he knows will grow with the business's storage requirements.



Experience 4: Manage apps in the Private Cloud

This experience will show how System Center 2012 SP1 improves datacenter service levels with dynamic reporting and monitoring to provide deep application insight. Ultimately, organizations will be able to improve root-cause analysis processes while reducing the mean time to service restoration. You can now leverage Operations Manager and Global Service Monitoring to deliver better application support. The scenarios to follow will visualize how Operations Manager, Visual Studio and Virtual Machine Manager provides seamless integration and simplifies IT oversight of the infrastructure and applications.

Scenario: Deep application performance insight

Jeff has a customer facing .Net web application used for Sales called Contoso Electronics. This application has been developed internally and the development team have asked for feedback on how the application has been performing. Jeff wishes to leverage the features of System Center to closely monitor the website and associated systems, and provide detailed feedback to the developers.

Jeff also notes that along with many of his other applications he spends a large amount of time delving into logs, server management consoles and performance monitor logs to try and find the reason an application is performing poorly. Most of the evidence provided by the end user is anecdotal and non-specific as they have no personal insight into the different layers of an application. Jeff decides to implement System Center 2012 Operations Manager with Application Performance Monitoring (APM) and begin collecting precise data on the activity of the application.

The business has also dictated to Jeff that the Sales Application is extremely important and must be available 99% of the time. Within the terms of this Service Level Agreement it is defined that poor performance is also considered to be not meeting the terms of the SLA.

Prerequisites:

These prerequisite steps will deploy the Sales Application to the GUEST01 server already present in the lab environment. This will give a target for the Operations Manager agent to discover, inventory and monitor.

- **Guest01** added to the **Contoso** domain. Refer to the appendix information on how to achieve this
- **Microsoft SQL 2012** installed on **Guest01** in mixed mode, with the **sa** account enabled and a password of **pass@word1**
- **IIS** installed on **Guest01**
- Enable the following features under **.NETFramework 4.5 Features** > **WCF Services**
    - HTTP Activation
    - TCP Activation
- **Memory** on **Guest01** will need to be configured accordingly
    - Startup RAM **1024**
    - Dynamic RAM **Enabled**
    - Minimum RAM **1024**
    - Maximum RAM **2048**

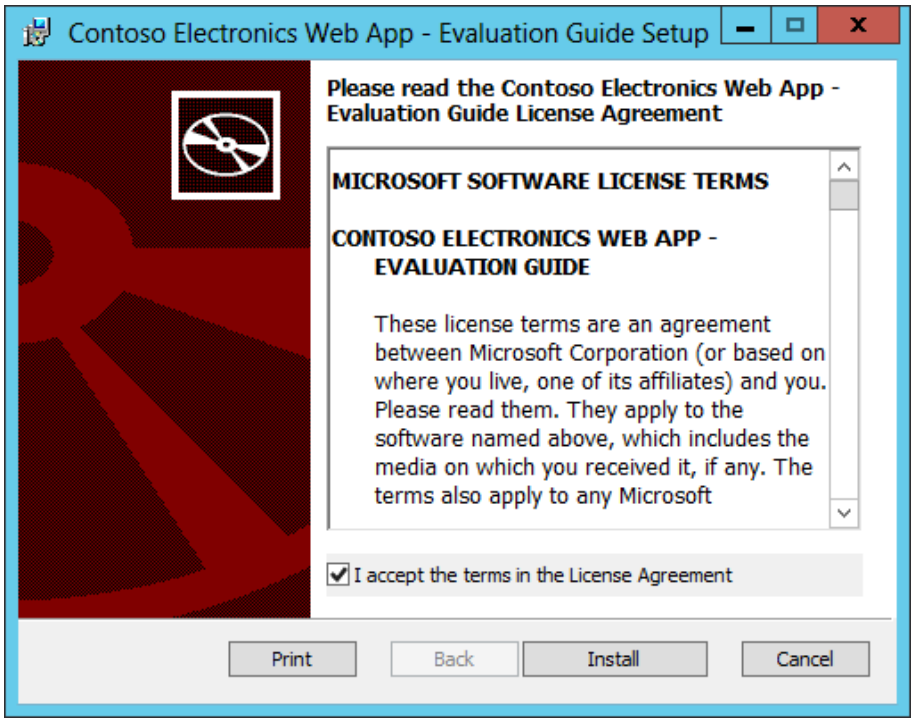
- **Guest01** will require **2 vCPU's**

Download the **Contoso Electronics Web App** source and extract to \\HyperV02\SMB3Share\SalesApp. The source is located here **http://go.microsoft.com/?linkid=9825975**

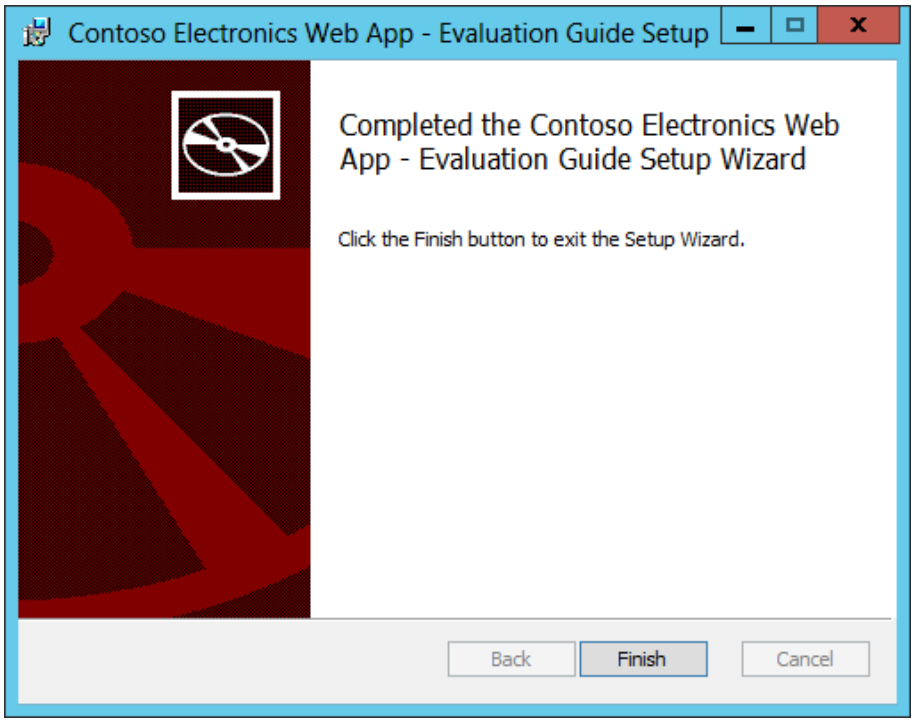
Disable **IE Enhanced Security Configuration** via the **Server Manager Console**

Install the Contoso Electronics Web App sales application

1. After downloading the Contoso Electronics Web App from **http://go.microsoft.com/?linkid=9825975**, you now install it. Run the file, check **I accept the terms in the License Agreement**, then click **Install**

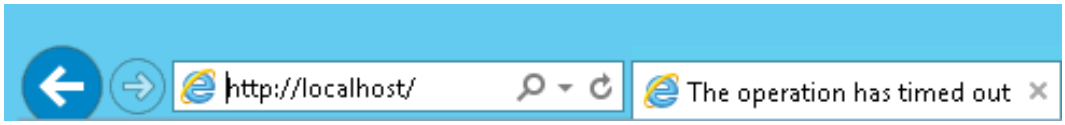


2. Click **Finish** to complete the install

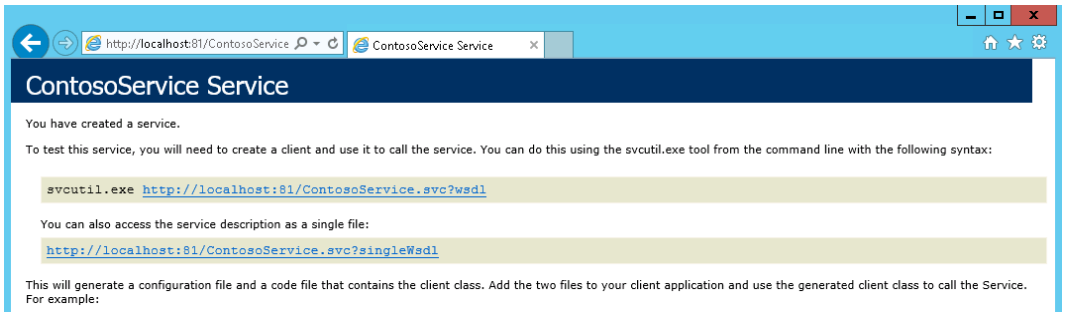


3. Open **Internet Explorer** and browse to **http://localhost**

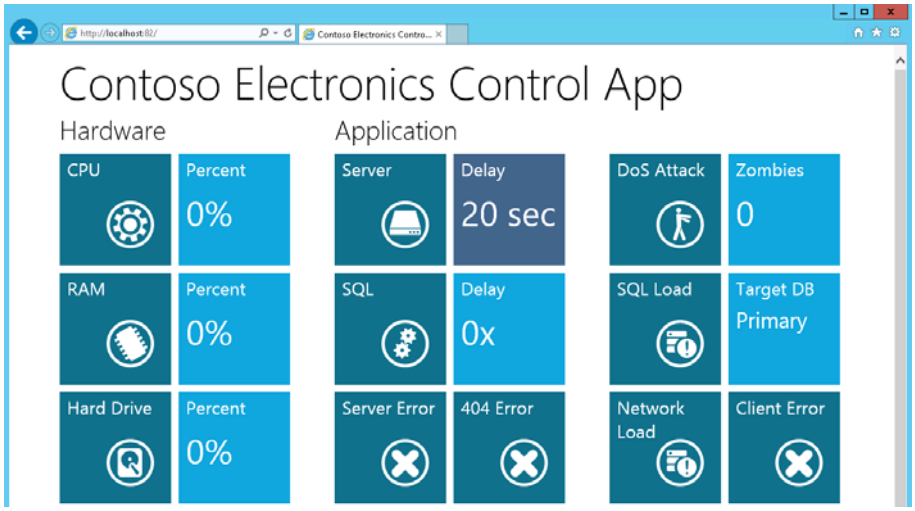
4. The website will now create the database, this can take approximately 2 minutes to complete. When you get an error page with the title "The Operation has timed out", refresh the page.



5. Open **http://localhost:81** and click on **ContosoService.svc** and confirm that the website displays as below



6. Open **http://localhost:82** and confirm that you receive the service console as displayed below



Install the Operations Manager Agent to Guest01

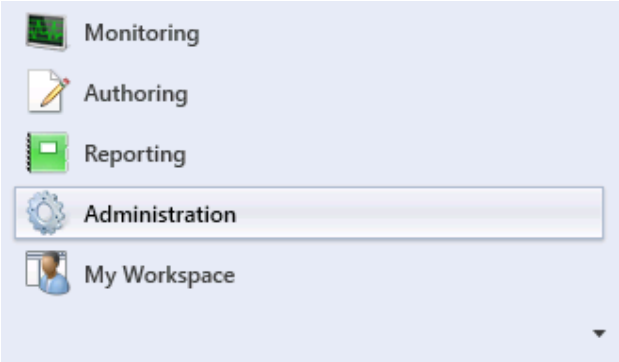
Now that the application exists in the domain, Jeff must deploy an Operations Manager agent to the server.

Log onto the server **OpsMgr** as **contoso\administrator** and load the **Operations Manager console**

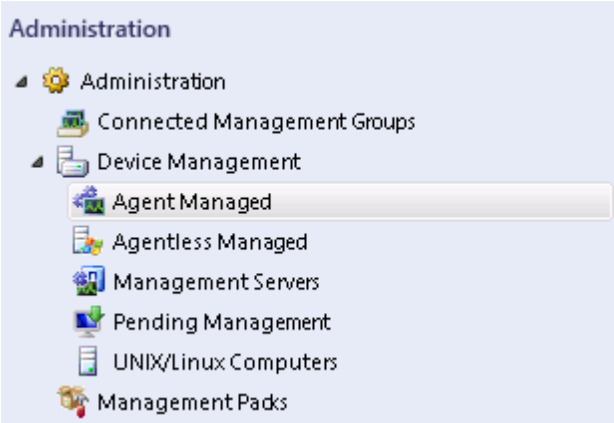
1. Click on **Administration**

Important Note

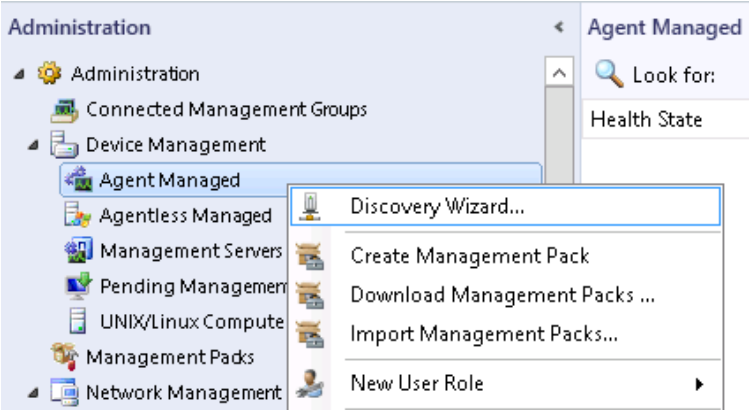
To monitor a website using APM the server does not need to be a member of the domain. It must have the Operations Manager agent, but it may exist in a DMZ or in an untrusted domain. This is especially important when dealing with customer facing websites. For more information refer to TechNet http://technet.microsoft.com/en-us/library/hh212856.aspx



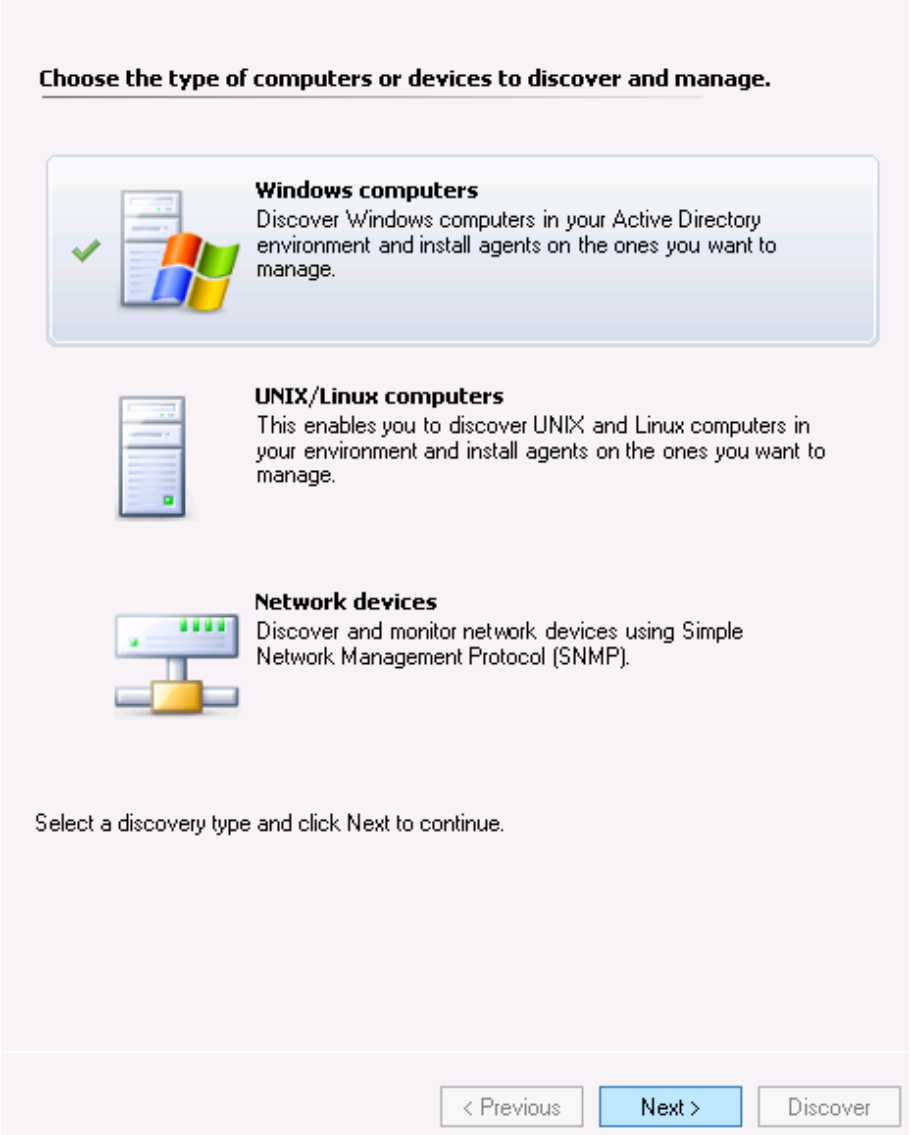
2. Click on **Agent Managed**



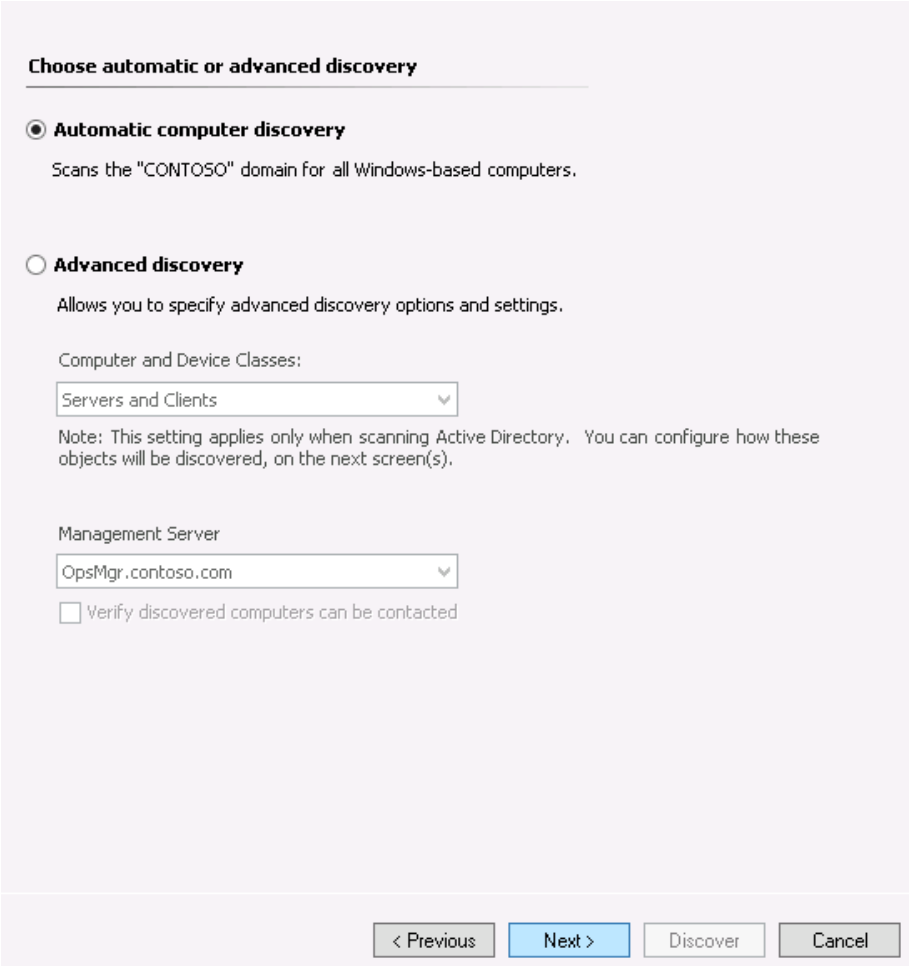
3. Right click **Agent Managed** and select **Discovery Wizard…**



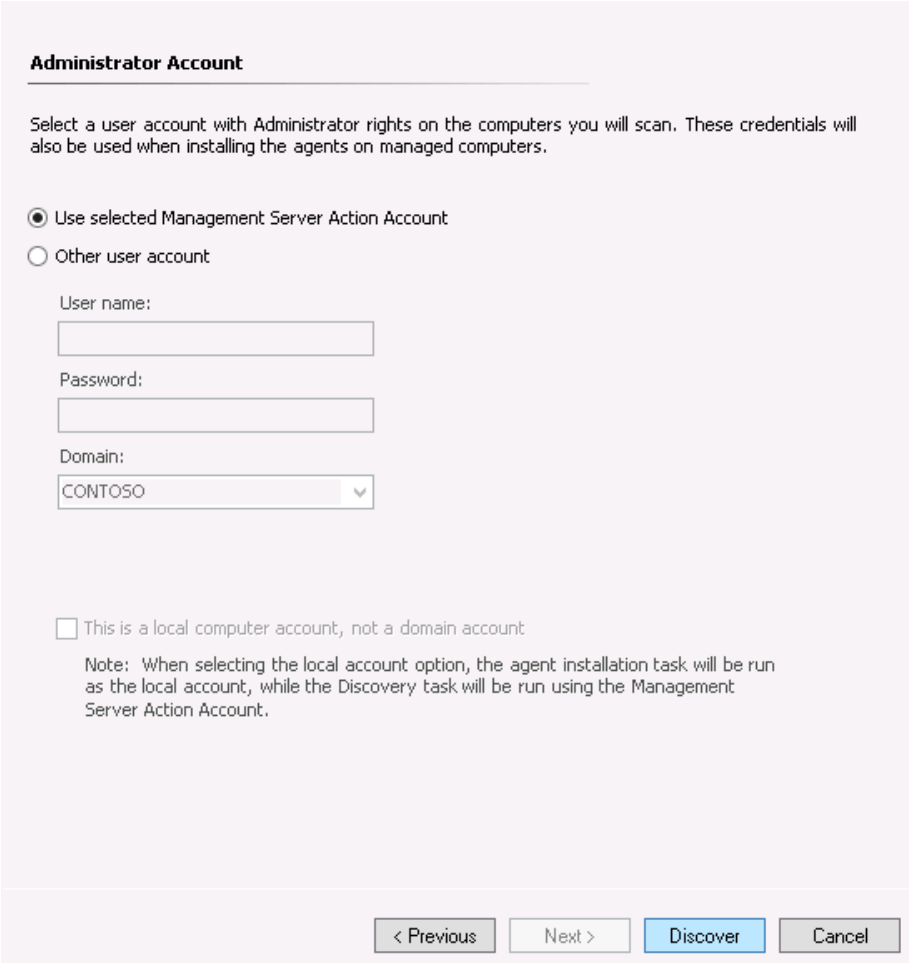
4. In **Computer and Device Management Wizard** select **Windows Computers** and click **Next**



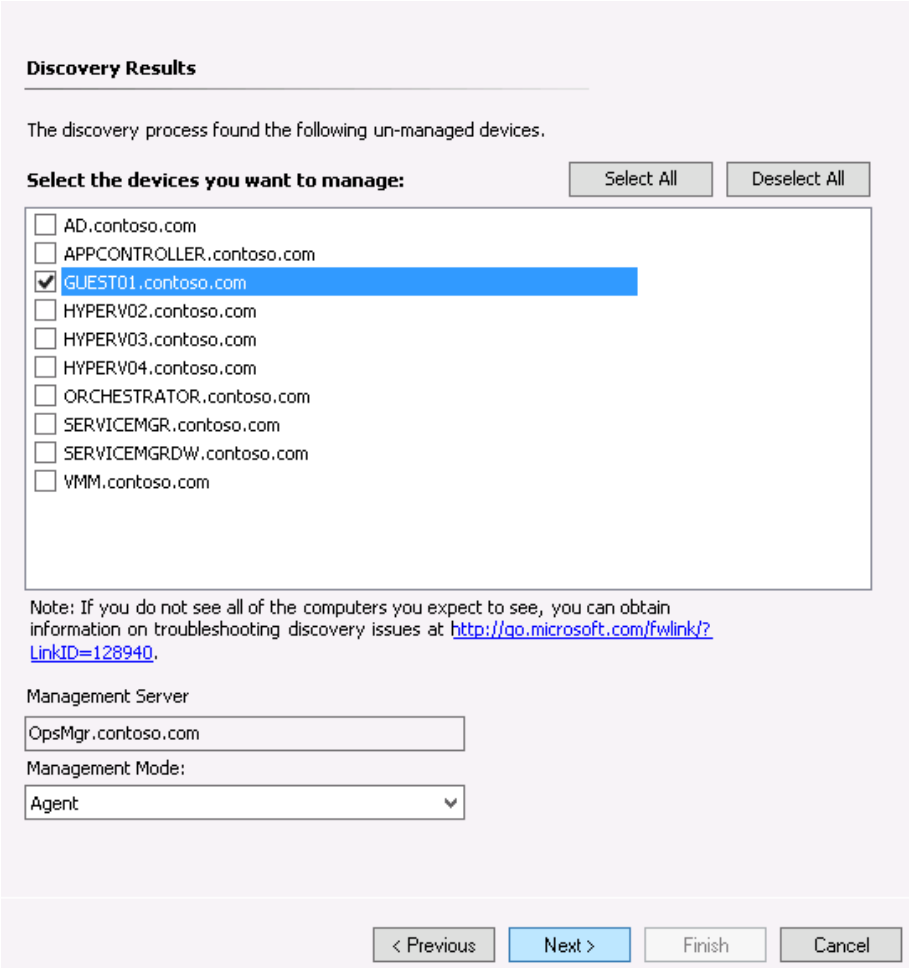
5. Tick **Automatic Computer Discovery** then click **Next**



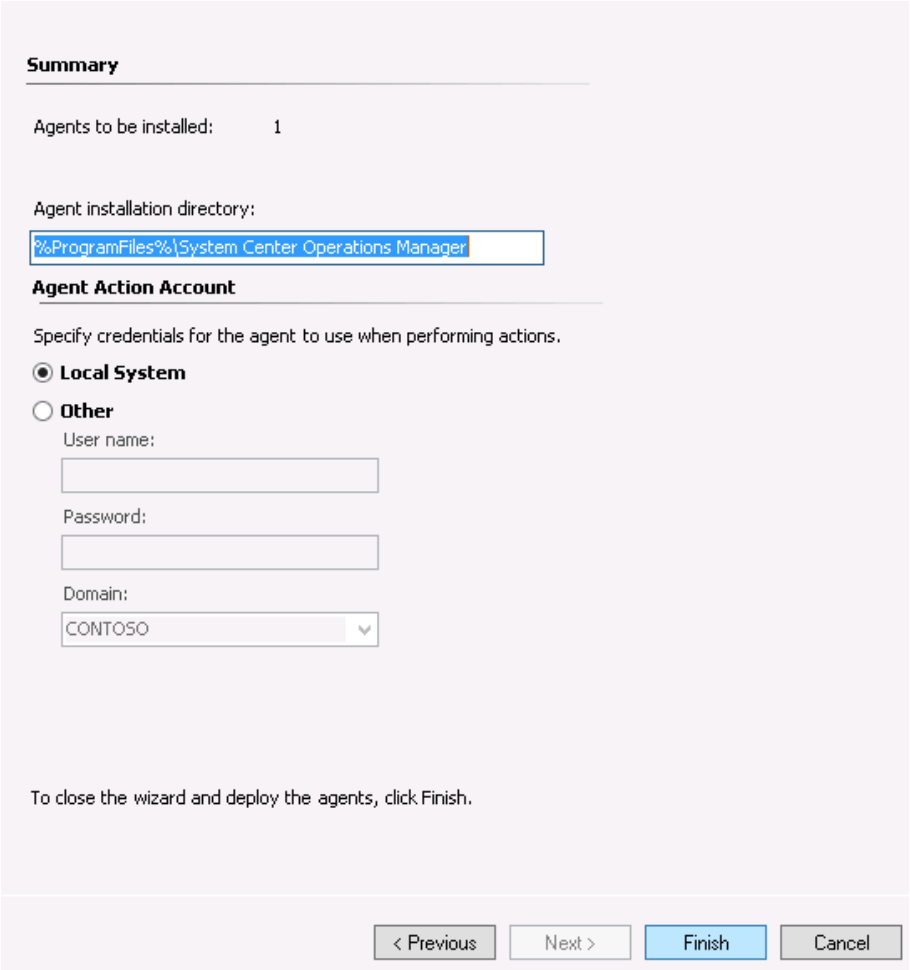
6. Tick **Use Select Management Server Action Account** and click **Discover.** Discovery can take up to 2 minutes to complete.



7. Tick the computer **Guest01.contoso.com** then click **Next**



8. On the **Summary Page** leave the defaults and click **Finish**



Import the Management Packs to Operations Manager

Operations Manager uses specific management packs for information on how to monitor everything in the enterprise from Network Devices to specific software such as SQL 2012 or an operating system like Windows. These management packs are maintained and updated in an online repository accessible via the Operation Manager console. Many other vendors also release their own management packs to allow the Operations Manager agent to get more granular understanding of devices and applications in the enterprise.

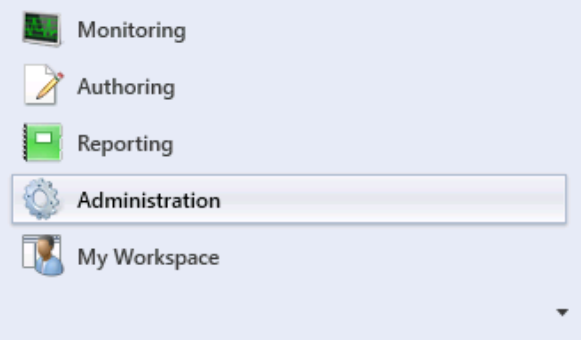
To allow Jeff to monitor the Sales Application from front end to back end database he must import specific management packs for each of the technologies involved in the process such as SQL 2012, IIS 8 and Windows Server 2012.

Important Note

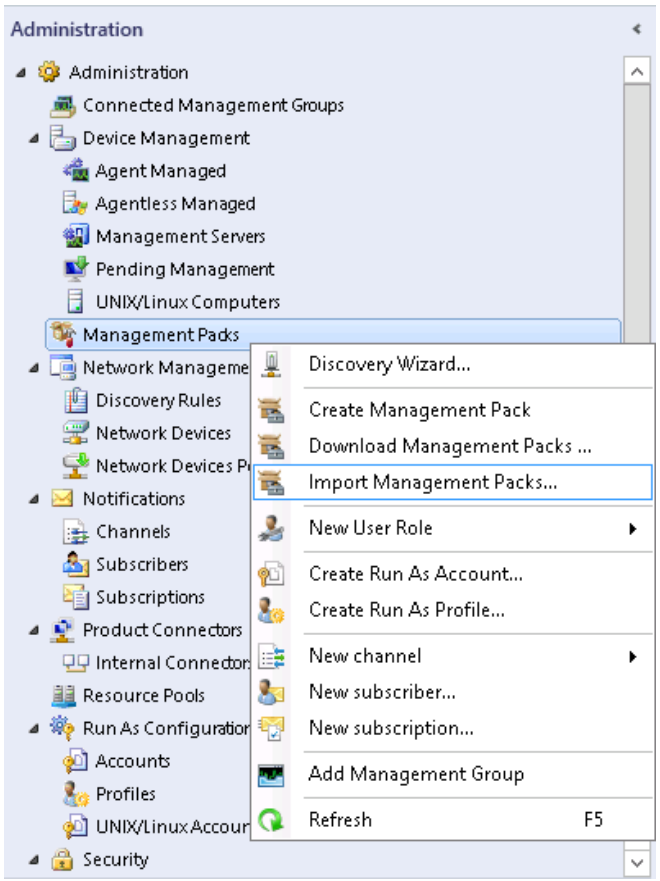
A minimum of management packs are installed by default to allow Operations Manager administrators to only monitor and receive alerts from a small base number of services. This alerting and monitoring can be further fine-tuned through the use of Overrides to ensure only the services and applications that require monitoring are targeted, rather than receiving a storm of alerts from every device on the network. See TechNet for further information http://technet.microsoft.com/en-us/library/hh230704.aspx

1. Log onto the server **OpsMgr** as **contoso\administrator** and load the **Operations Manager console**

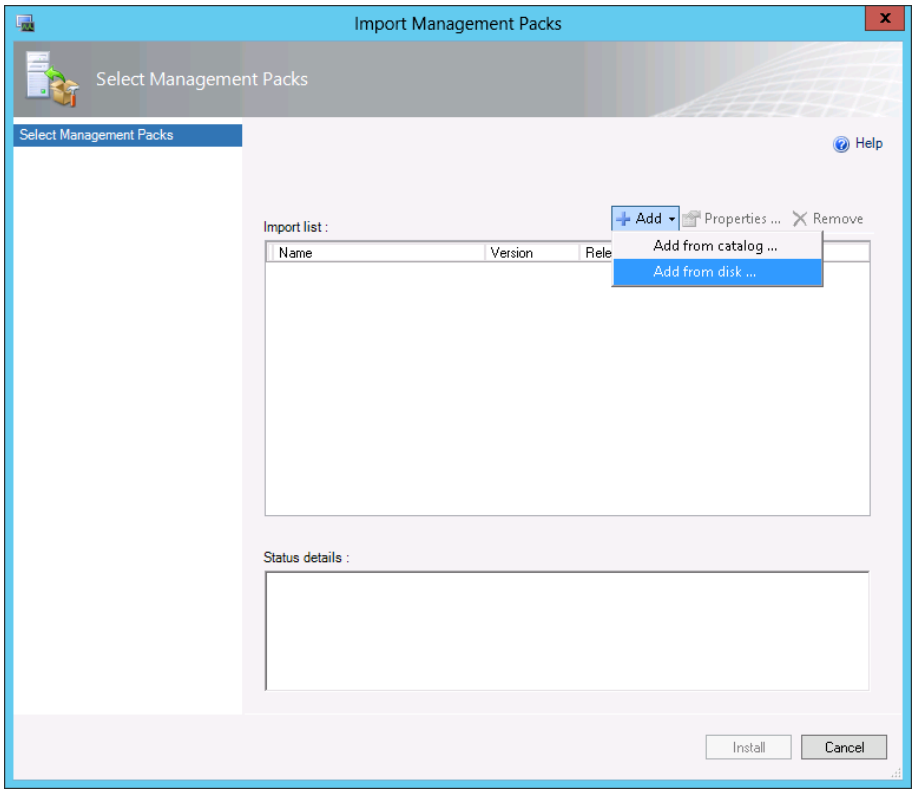
2. Click on **Administration**



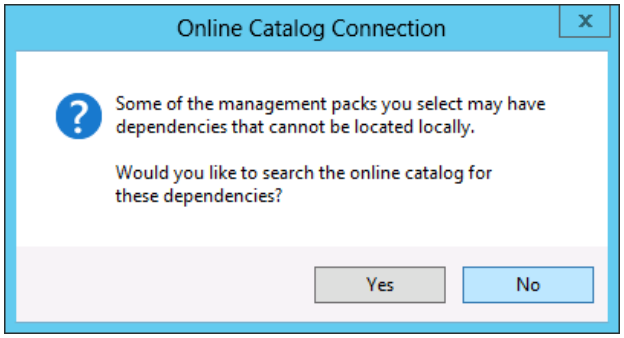
3. Right click **Management Packs** select **Import Management Packs**



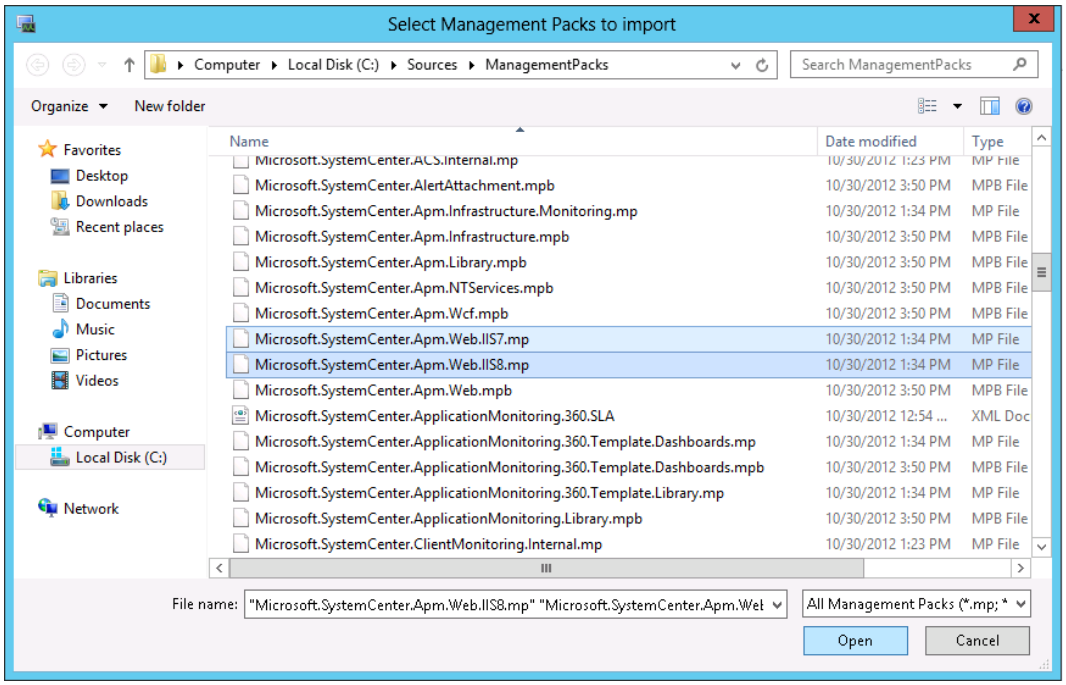
4. Click **Add**, then select **Add from disk ...**



5. When prompted to check the online catalog for further dependencies choose **No**

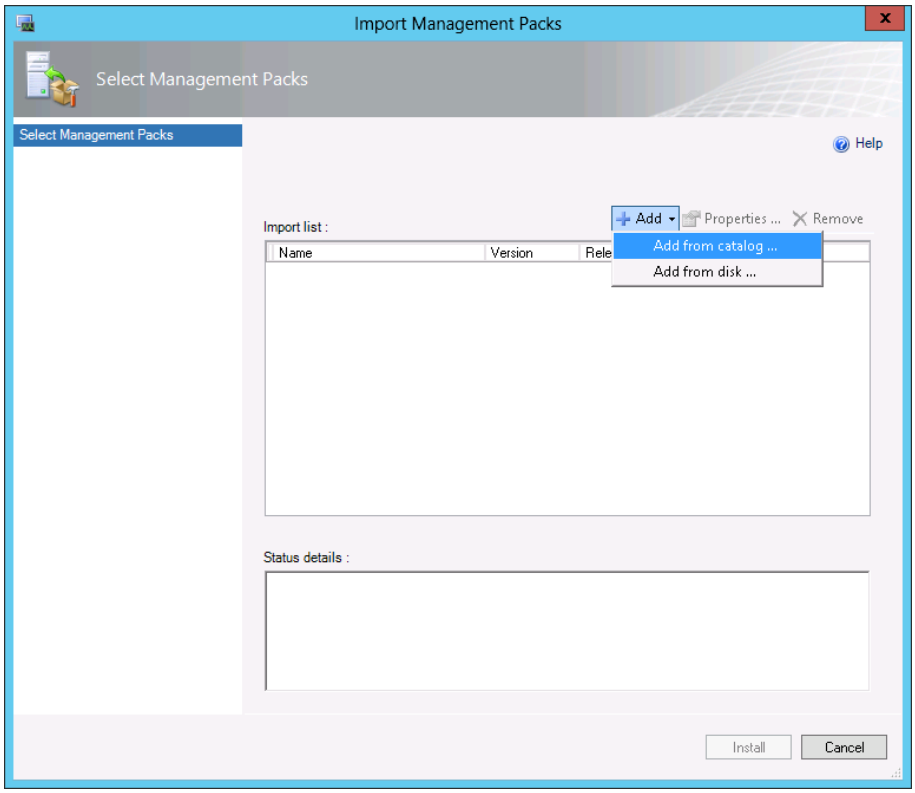


6. Browse to the System Center Service Manager installation media. In this example we are using **C:\Sources\Management Packs**. Select the following packs and then click **Open**
    - Microsoft.System.Center.Apm.Web.IIS7.mp
    - Microsoft.System.Center.Apm.Web.IIS8.mp



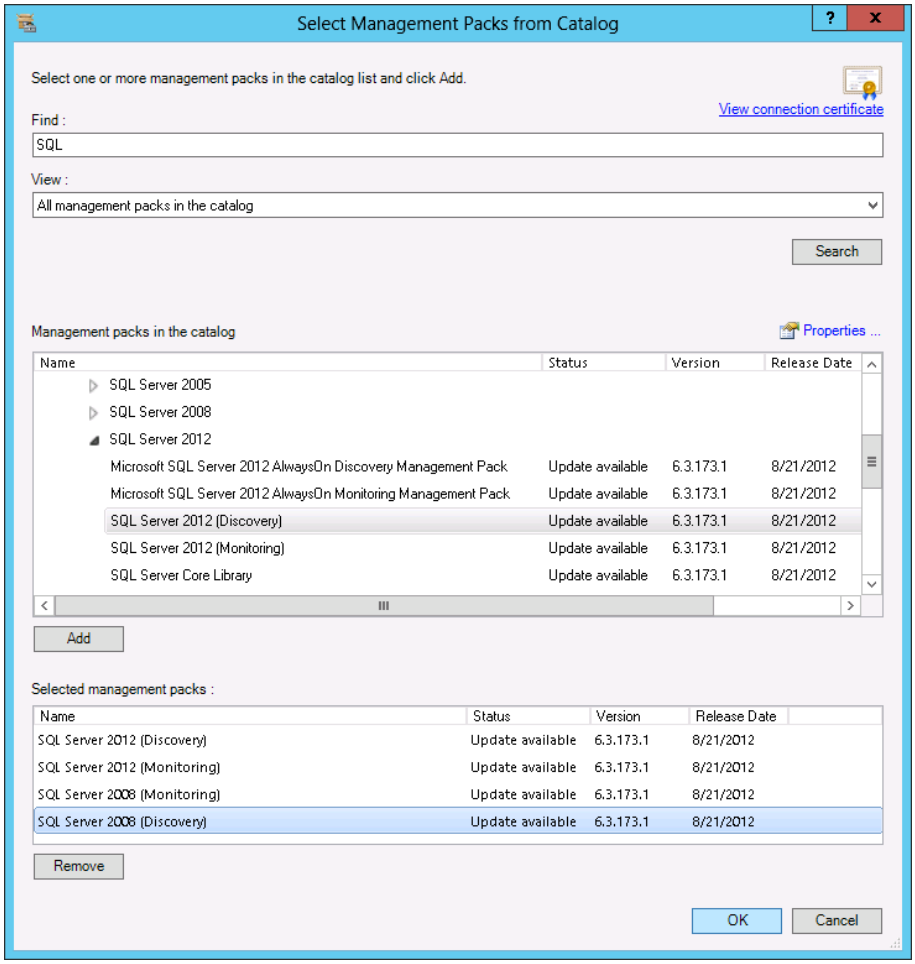
7. The following management packs will also need to be installed. These are not located on the System Center Service Manager installation media, these have to be downloaded from the **online catalog.** An internet connection will be required for these packs to be downloaded:
    - SQL Server 2008 (Discovery)
    - SQL Server 2008 (Monitoring)
    - SQL Server 2012 (Discovery)
    - SQL Server 2012 (Monitoring)
    - Windows Server 2008 Internet Information Services 7
    - Windows Server 2012 Internet Information Services 8
    - Windows Server 2008 Operating System (Discovery)
    - Windows Server 2008 Operating System (Monitoring)
    - Windows Server 2012 Operating System (Discovery)
    - Windows Server 2012 Operating System (Monitoring)

8. Click on **Add** followed by **Add from Catalog**



9. Wait for the following dialogue box to disappear:

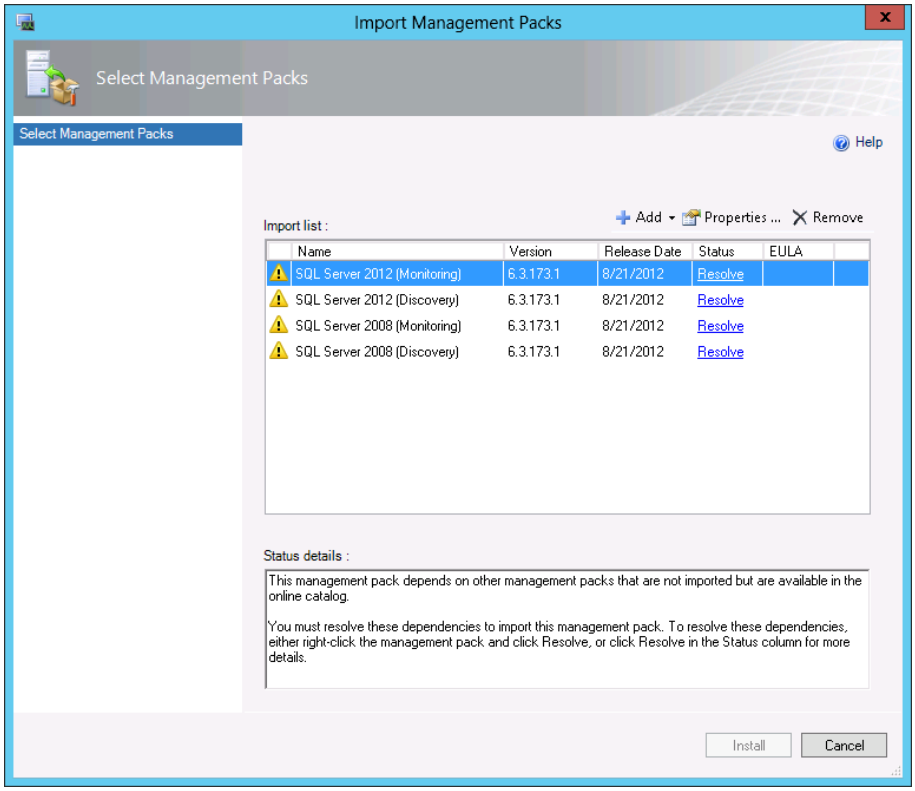
10. Type **SQL** into the **Find** query and hit search. You can now expand the results in the management packs in the catalog and find the required **SQL Server 2012 (Discovery)** and click on **Add**.



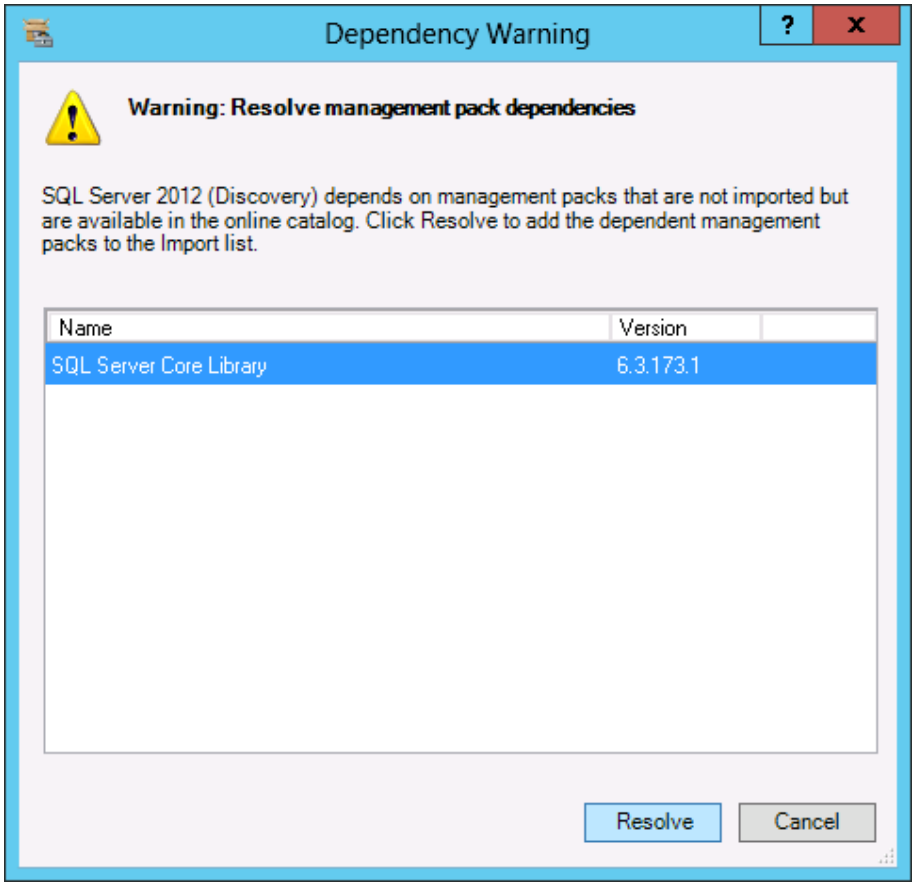
11. Repeat Step 10 and add the other Management packs:
    - SQL Server 2008 (Discovery)
    - SQL Server 2008 (Monitoring)
    - SQL Server 2012 (Discovery)
    - SQL Server 2012 (Monitoring)
    - Windows Server 2008 Internet Information Services 7
    - Windows Server 2012 Internet Information Services 8
    - Windows Server 2008 Operating System (Discovery)
    - Windows Server 2008 Operating System (Monitoring)
    - Windows Server 2012 Operating System (Discovery)
    - Windows Server 2012 Operating System (Monitoring)

    Click **OK** once completed

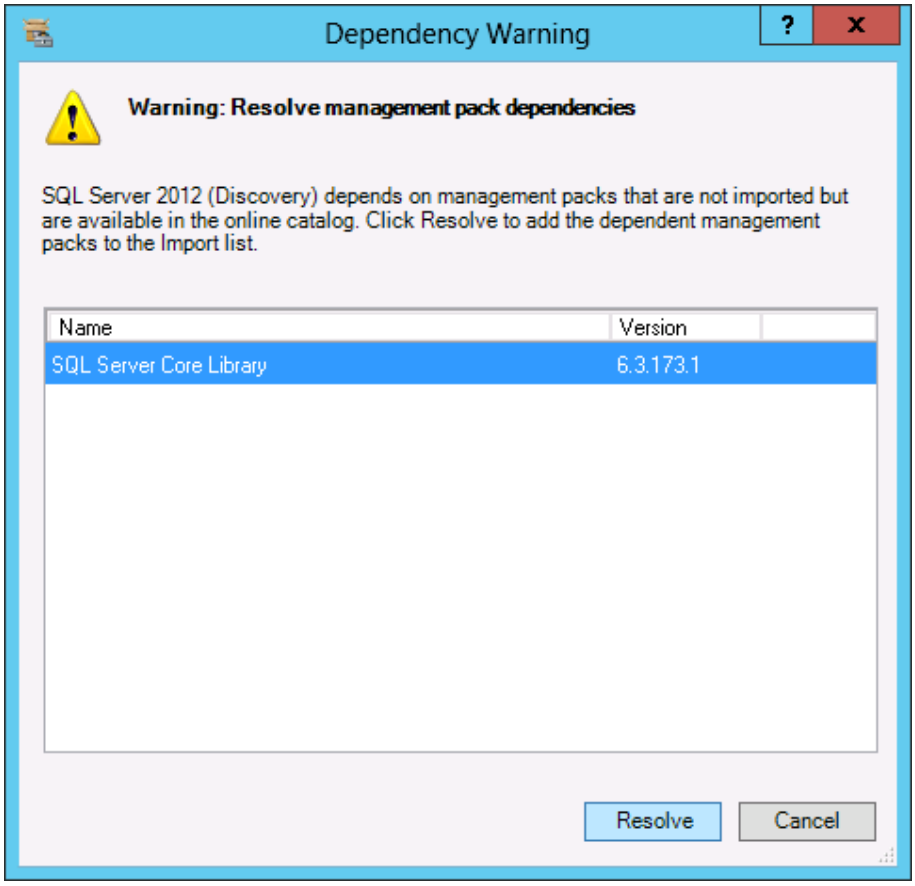
12. Once the Management Packs have been added you will be returned to the Import Wizard.



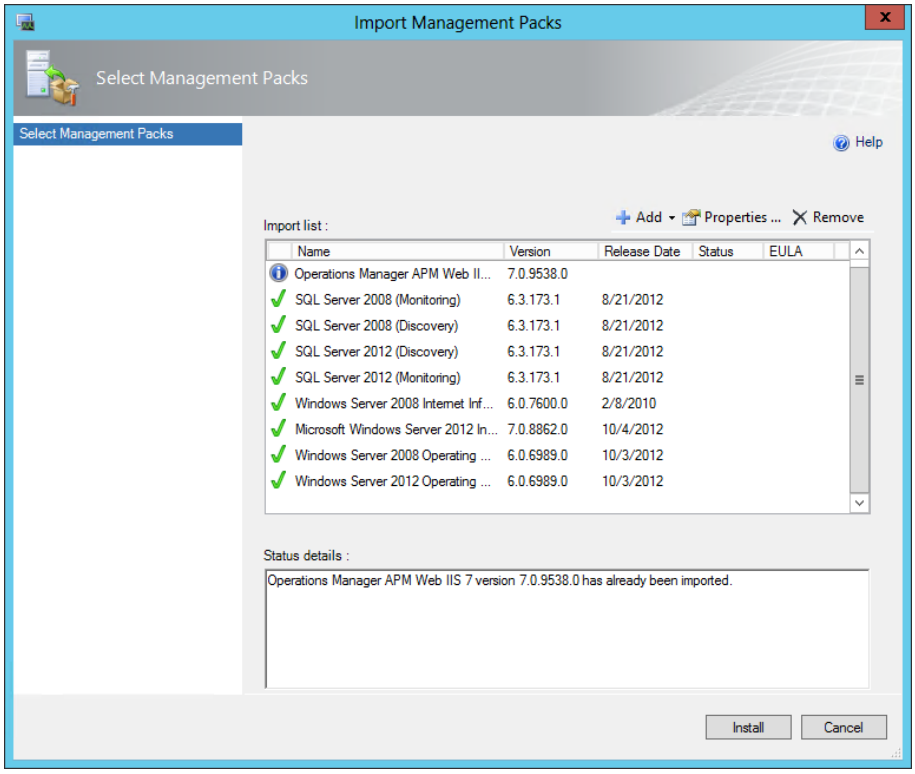
13. A number of the packs will have dependency issues, click on **Resolve** to resolve these issues.



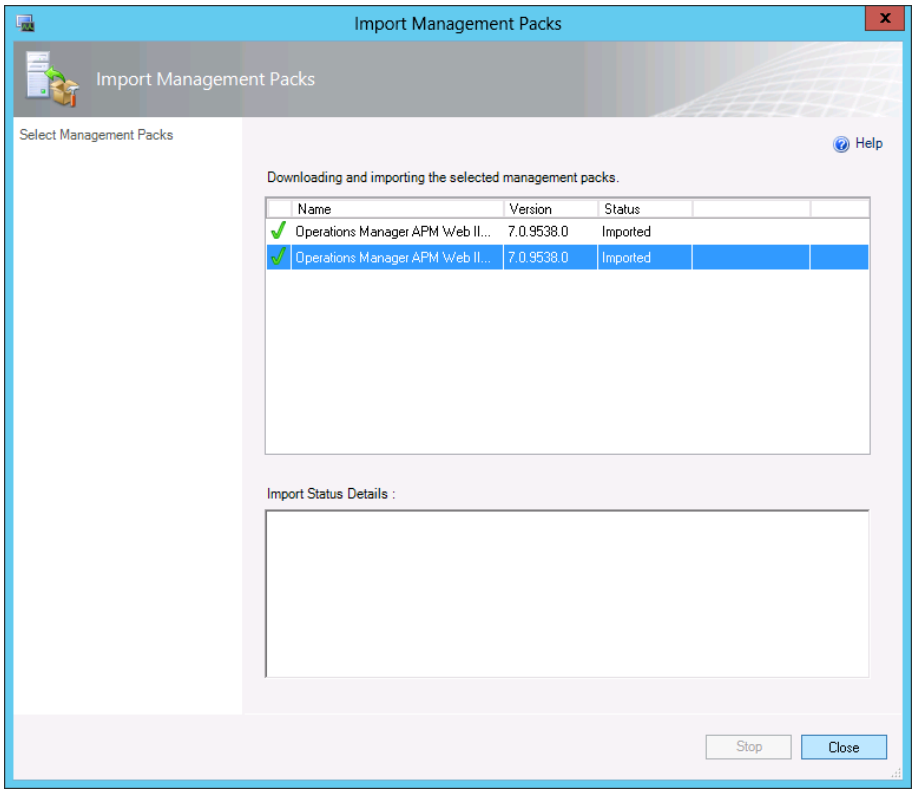
14. Click **Resolve** to fix any dependency errors seen.



15. Click on **Install** to complete the installation of the Management Packs



16. Once the import has completed, click on **Close**



Create the .Net Application Monitor

Now that Jeff has imported the correct management packs, these will be disseminated to the agents on the monitored servers when they next check in. If the software or application that these management packs apply to exists on the server it will be inventoried and the data returned to the management server. In this case, all IIS websites and SQL Databases will be inventoried.

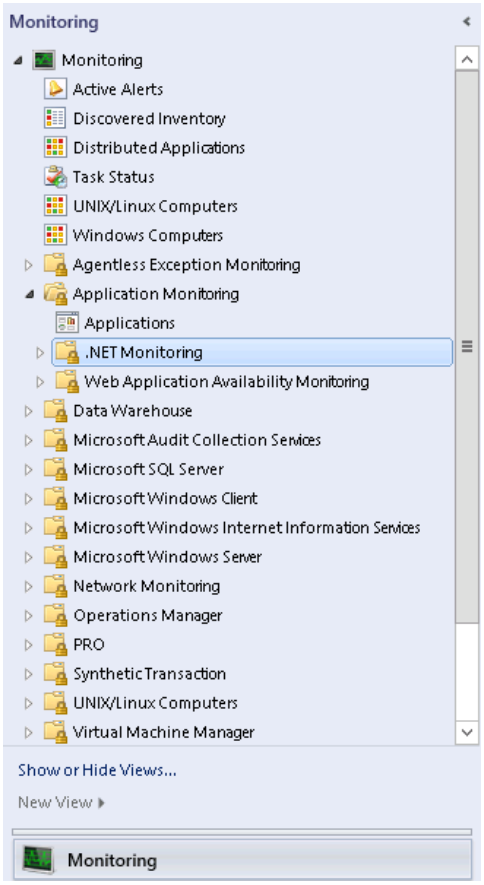
Once the inventory has been completed and the data returned to the Operations Manager server, Jeff will be able to select specific websites and databases to monitor. In this case he will create a specific .Net application monitor that will invoke APM to delve deep into the .Net code.

During this set up Jeff will configure the APM agent to monitor two things – the web service and the front end web site. For the front end website he will also enable the Client Side monitoring which will allow APM to inject code into the webpage and return error and performance metrics on how the client sees the final webpage.

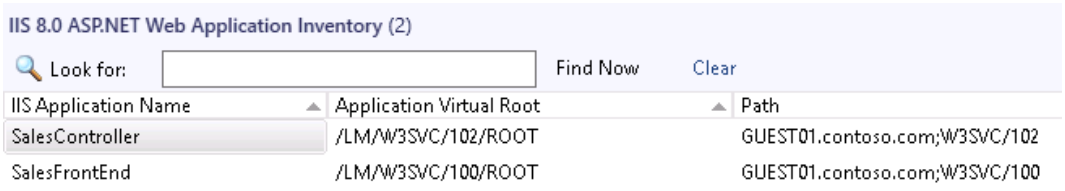
Important Note

APM can be utilised to monitor Java web applications and .Net based Windows Services as well as .Net based web applications.

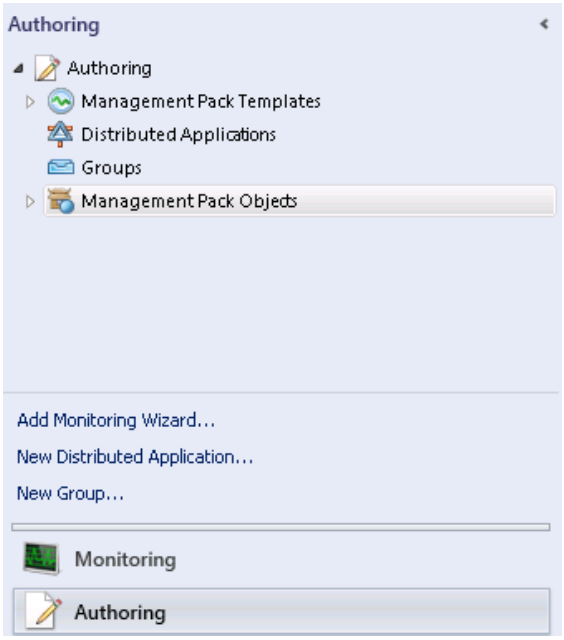
1. Once imported, click on **Monitoring** and browse to **Application Monitoring**, **.Net Monitoring**



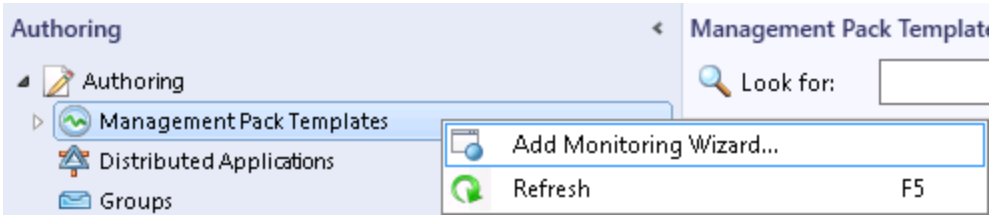
2. Click on **IIS 8.0 ASP.NET Web Application Inventory** and wait for the **Sales Application** to appear in this pane. This can take up to 15 minutes to appear



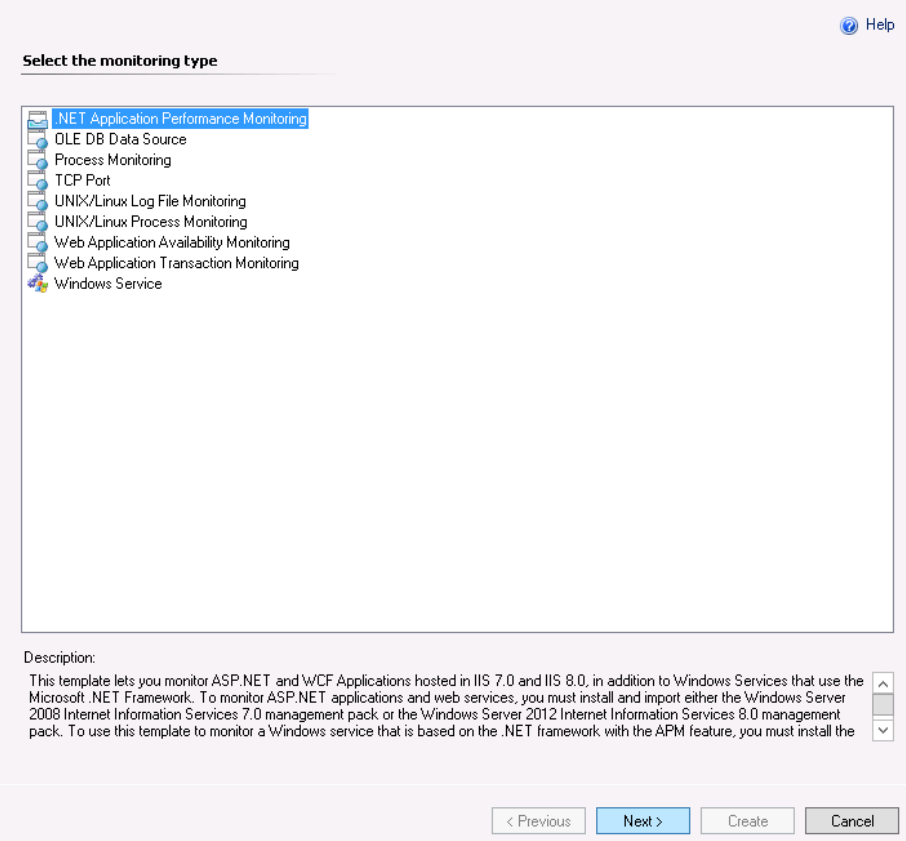
3. Once it has appeared, click on **Authoring**



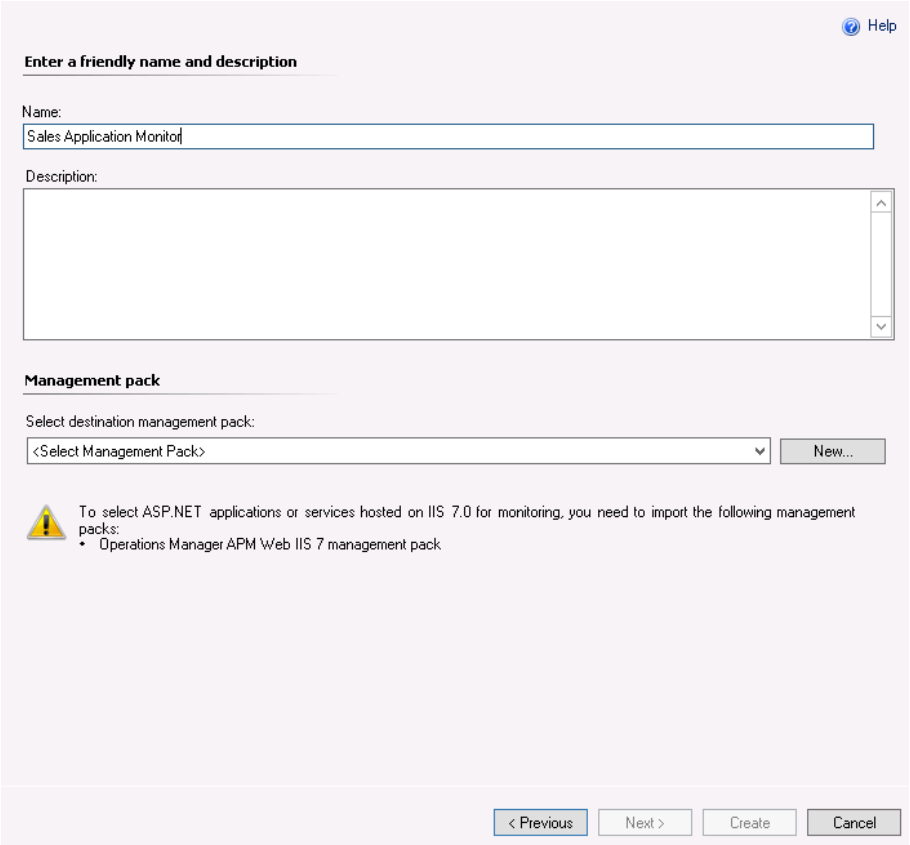
4. Right click **Management Pack Templates** and select **Add Monitoring Wizard**



5. Select **.Net Application Performance Monitoring** and click **Next**

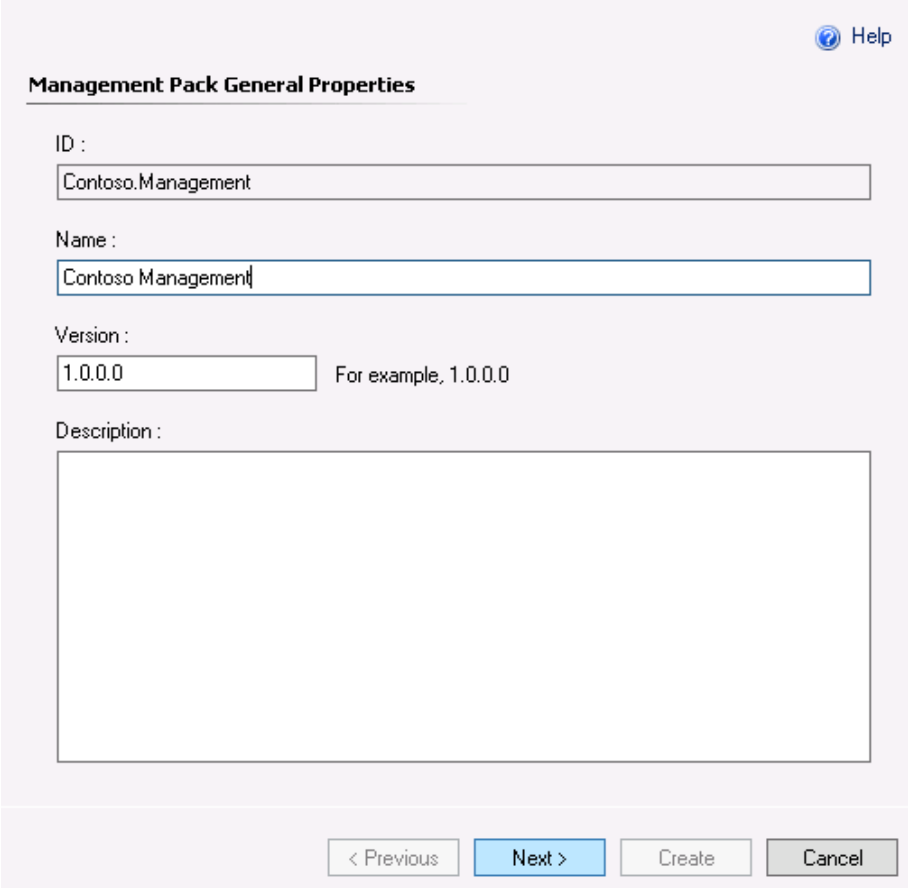


6. In the **Name** field type **Sales Application Monitor**

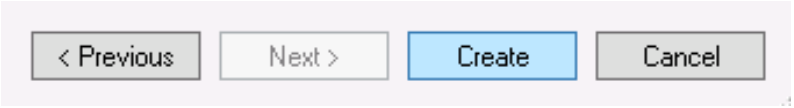


7. Click on the **New** button next to the **Destination Management Pack** field

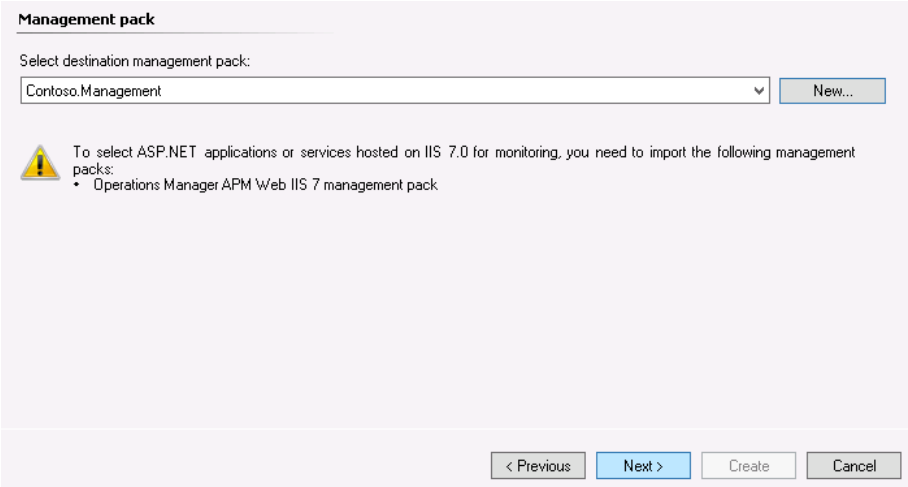
8. Enter **Contoso Management** in the **Name** field, click **Next**



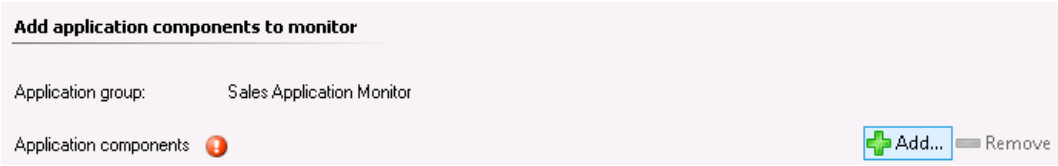
9. In **Knowledge Article** window that opens, click **Create**



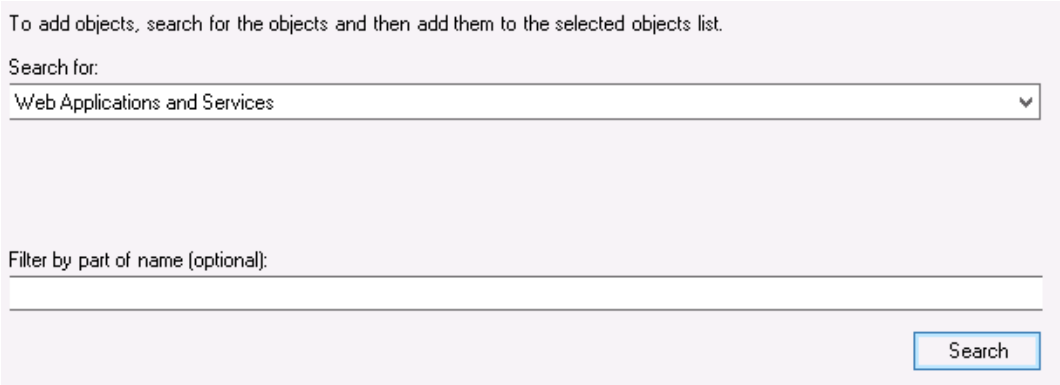
10. This should now be selected in the destination management pack field. Click **Next**



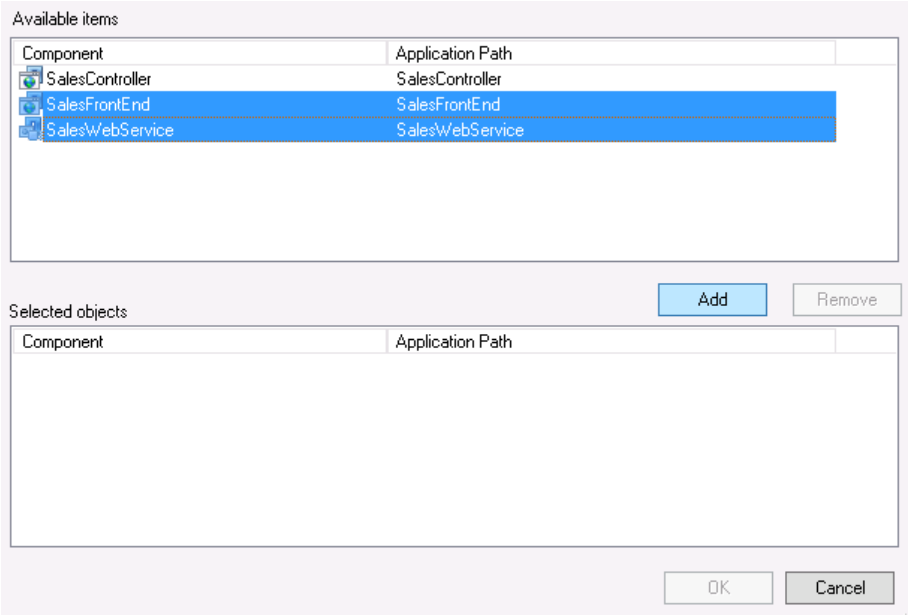
11. Click **Add**



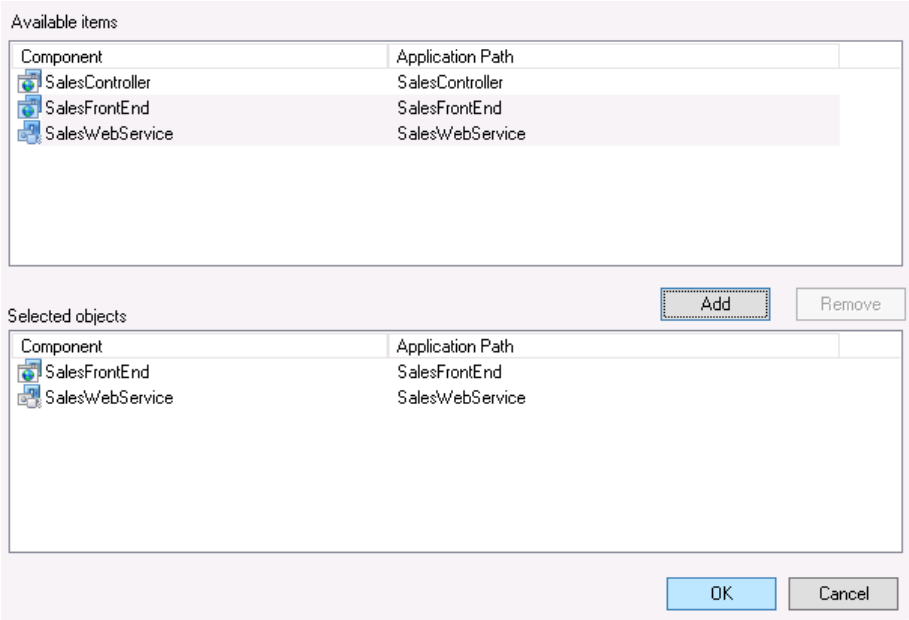
12. Select **Web Applications and Services** and click **Search**



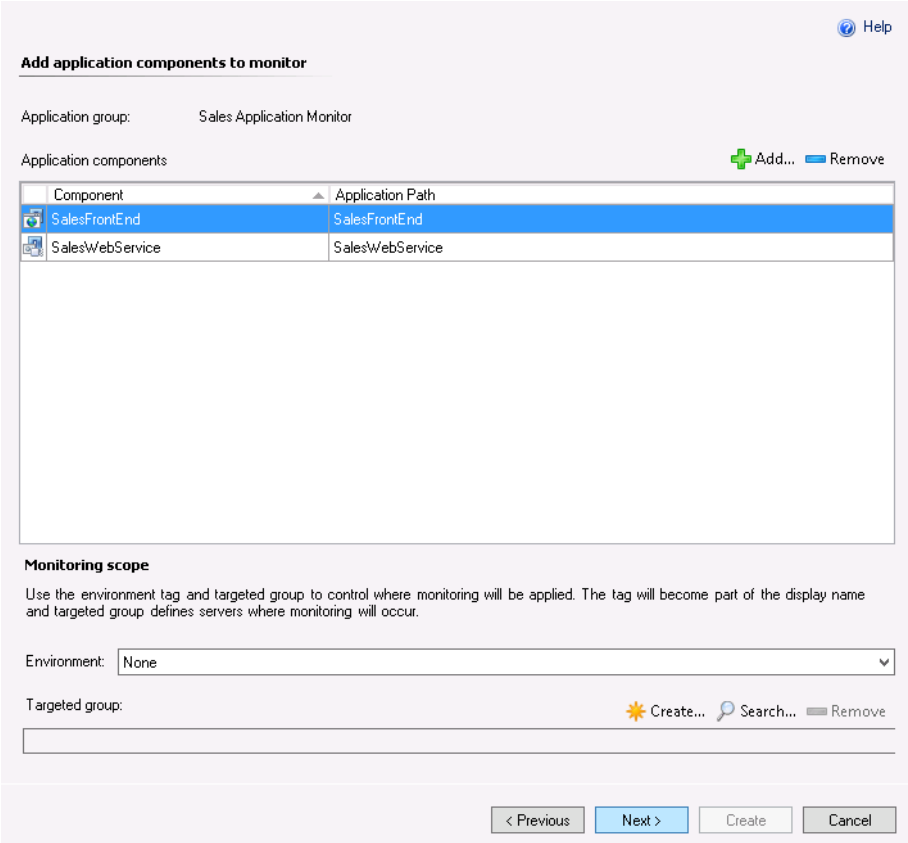
13. Click and highlight **SalesFrontEnd** and **SalesWebService** and click **Add**



14. Click **OK**



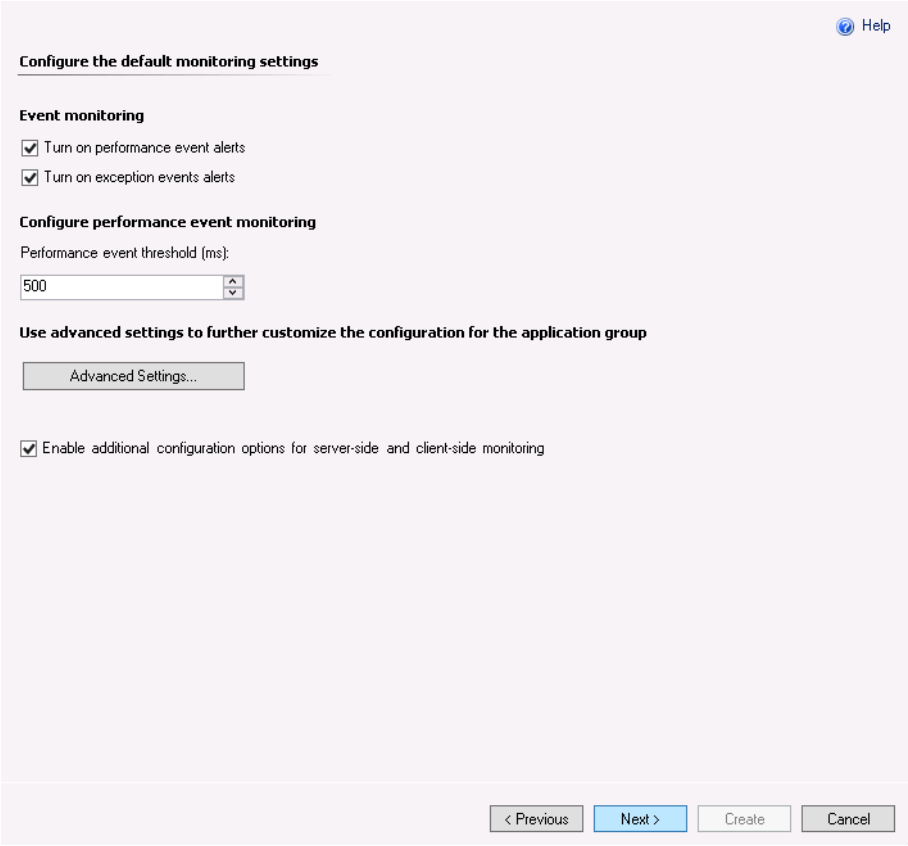
15. Click **Next**



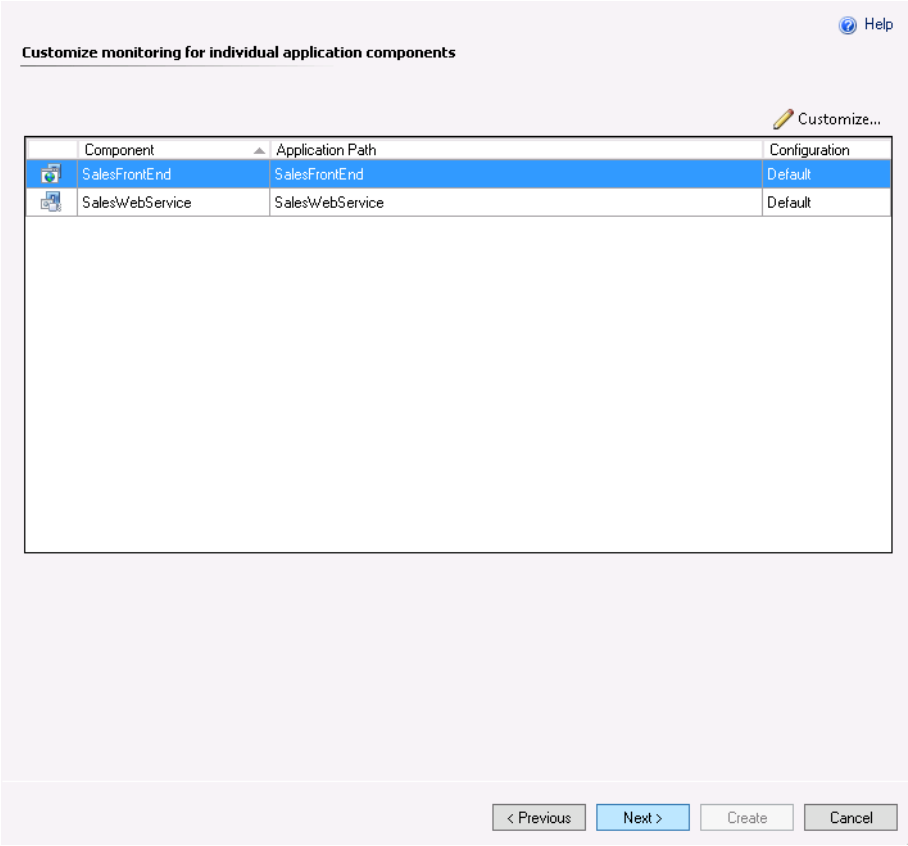
16. Change the **Performance Event** threshold to **500ms** and ensure the box **Enable Additional Configuration Options** is ticked and then click **Next**

Important Note

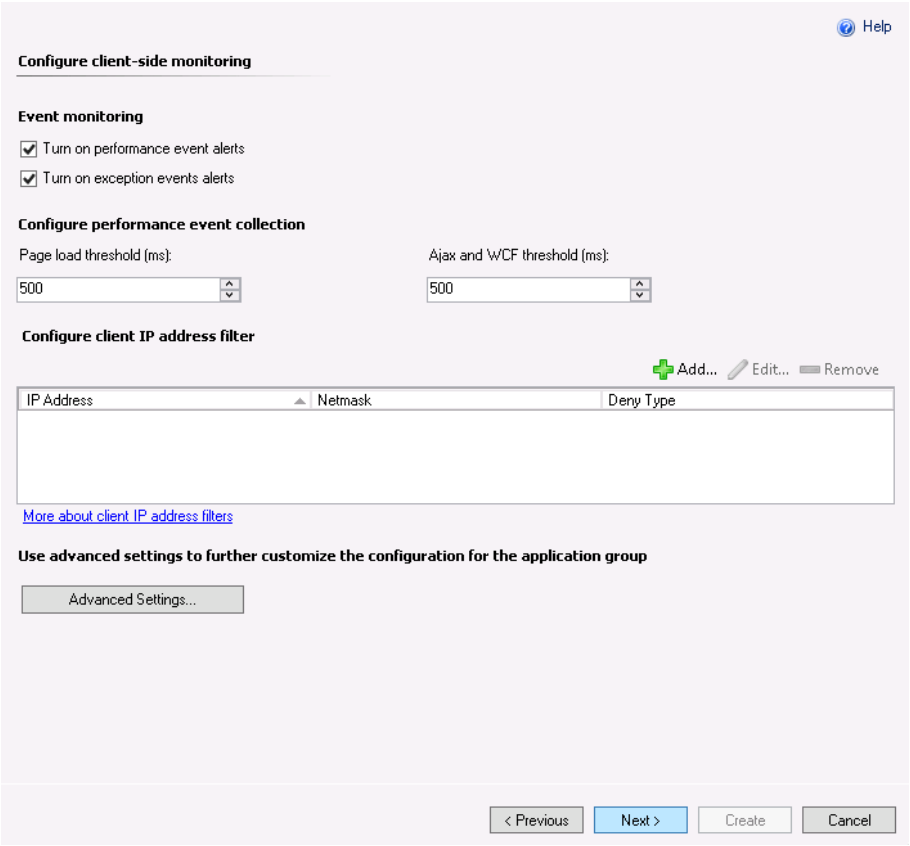
This threshold is purposefully set very low in this monitor to ensure alerts are generated. In a production situation this would require discussion with the business, tuning and testing to find the best fit for your environment.



17. Click **Next**



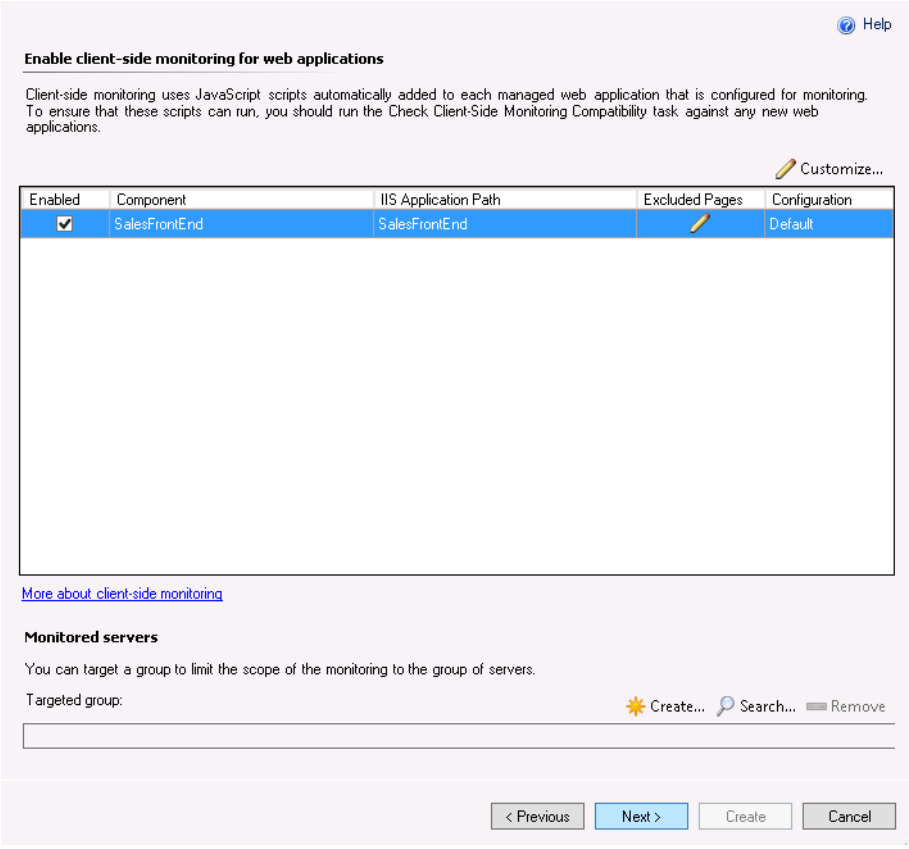
18. Tick the turn on **Exception Events Alerts** box, change the **Page Load** and **Ajax** thresholds to **500ms** and click **Remove** twice to clear the IP Address filter and click **Next**



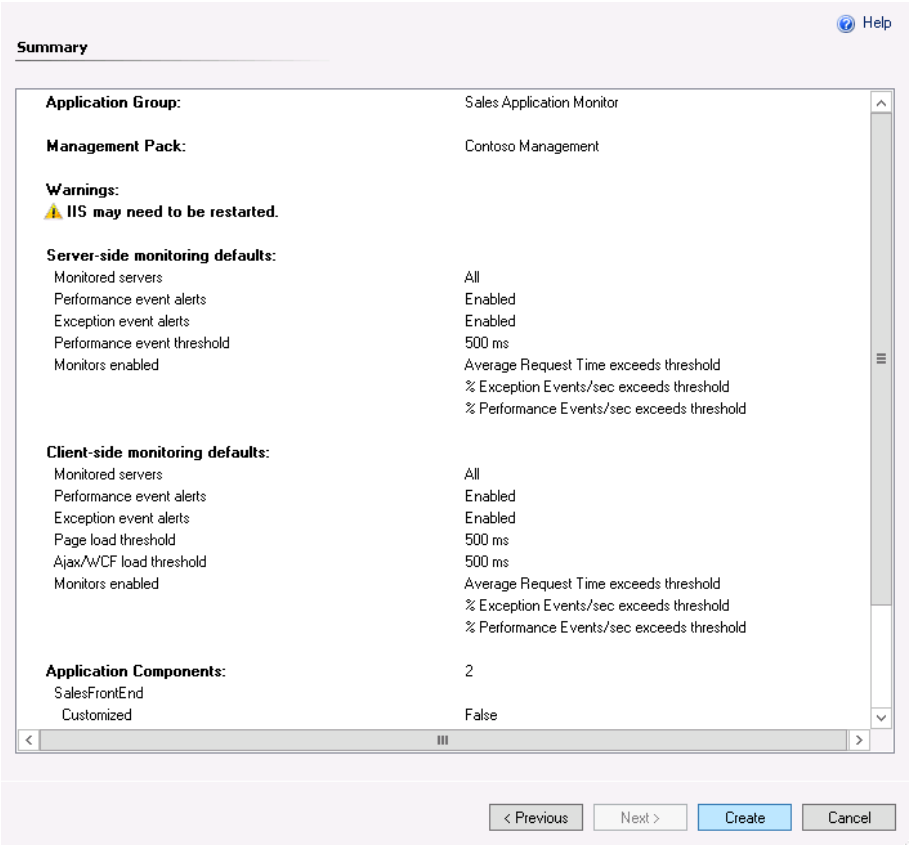
Important Note

The subnets are removed in this scenario to ensure all client web requests have APM injected code in the return. In a production implementation this would be tuned to a small percentage of web clients depending on factors such as the number of web requests, network configuration and the location of clients.

19. Tick the **Enabled** box next to **SalesFrontEnd** and click **Next**

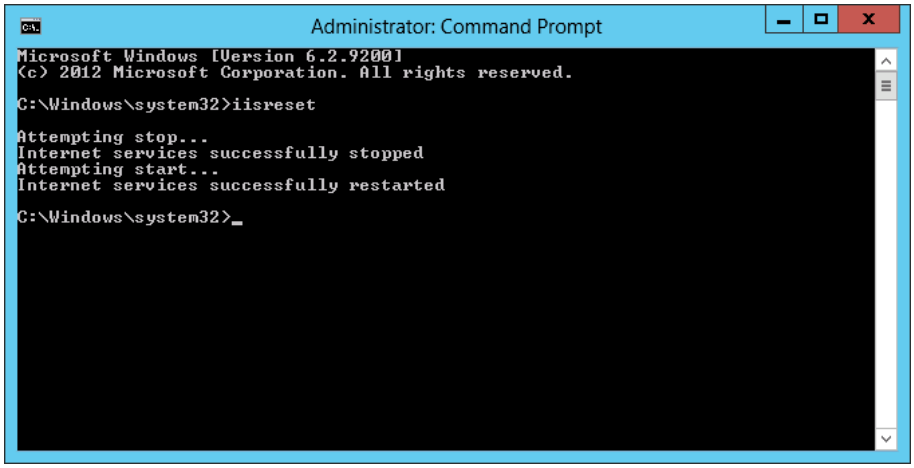


20. Click **Create**



21. Back on **Guest01** wait for **10 minutes** then open an administrative command prompt and type the command

```
Iisreset
```



Important Note

The IIS reset is required as the APM agent begins in a disabled state before a .Net monitor is configured. When the agent detects that a .Net monitor for a website hosted in the server has been enabled, it starts the APM agent, creates Application Pools and injects itself into IIS and the website. At this point IIS must be restarted to enable the new configuration to be loaded.

Create a Web Application Availability Monitor

Now that Jeff has created a .Net Application monitor, he also needs to monitor the website to ensure it is responding within a specific timeframe. This is accomplished by creating Availability Monitors which use Operations Manager agents to run requests against specific websites. This means that any server with an Operations Manager agent can be used to monitor a website. This can allow Jeff to build up a map of how a website is responding from a perspective of any of his managed sites within the enterprise.

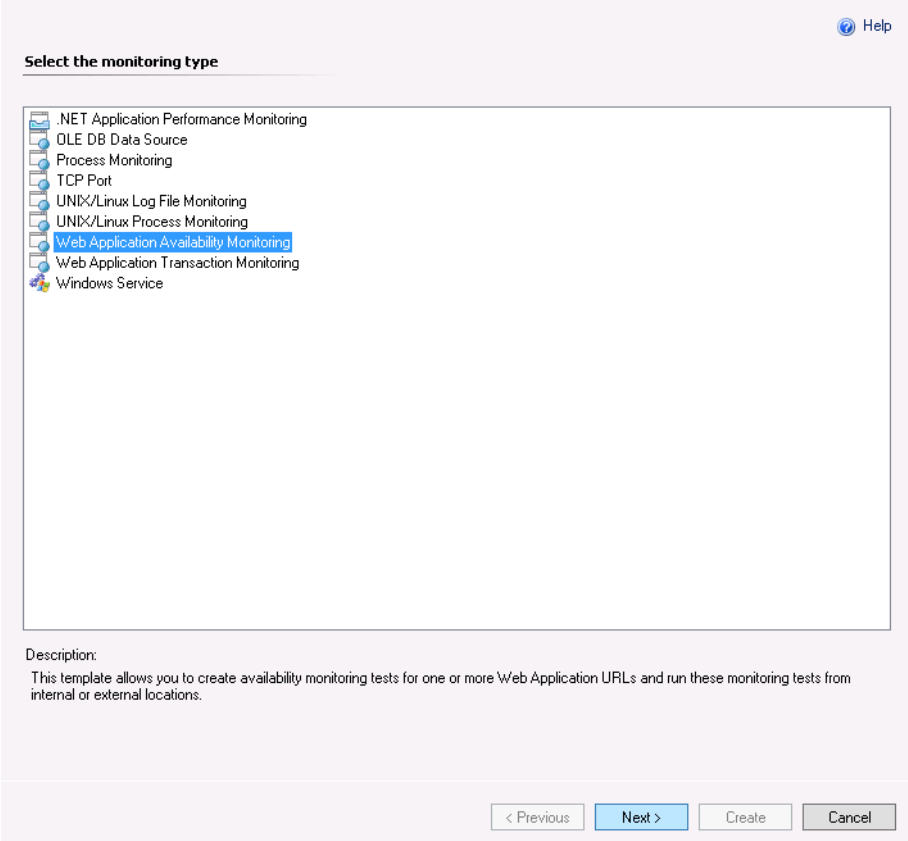
Important Note

This can also be extended to the point of view of an external source also. Using Global Service Monitor (GSM) availability monitors can be created to report on your websites from multiple countries around the globe. This gives a true 360 degree perspective of your web applications and adds even more granularity to the triage process. For more information see http://www.microsoft.com/en-us/server-cloud/system-center/global-service-monitor.aspx

1. In the **Operations console** on **OpsMgr** click on **Authoring**

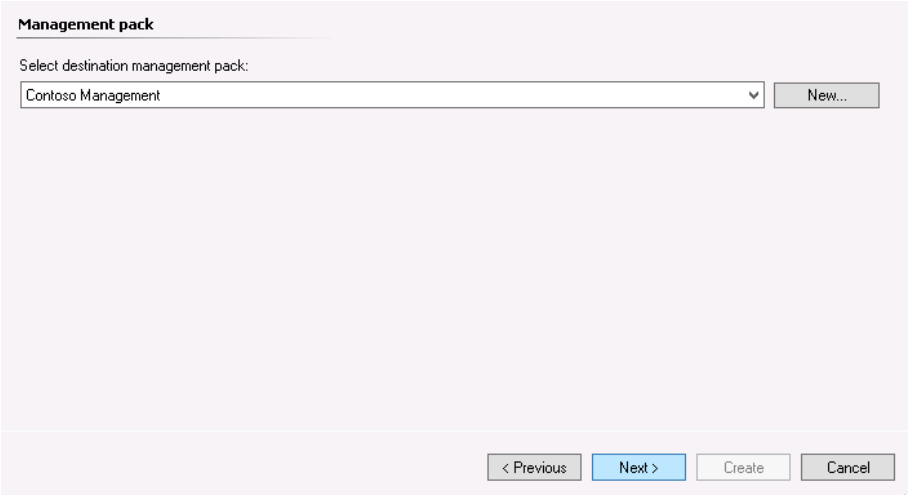
2. Right click **Management Pack Templates** and select **Add Monitoring Wizard**

3. At **Select Monitoring Type** click on **Web Application Availability Monitoring** and press **Next**

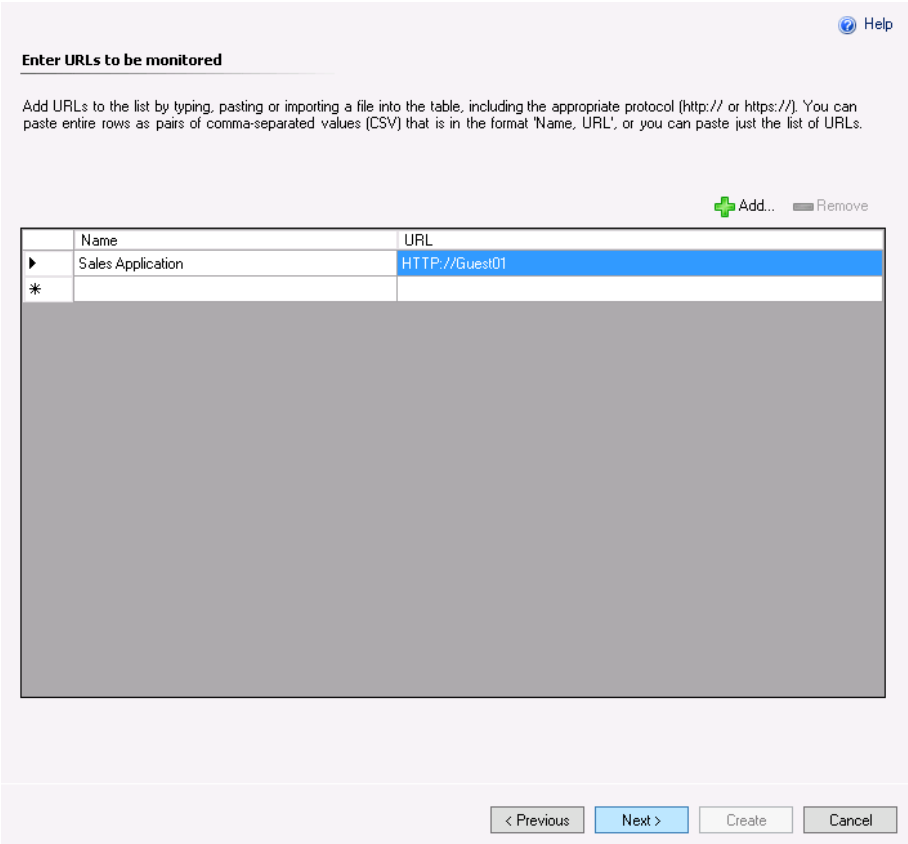


4. In the **Name** field type **Sales Application Availability**

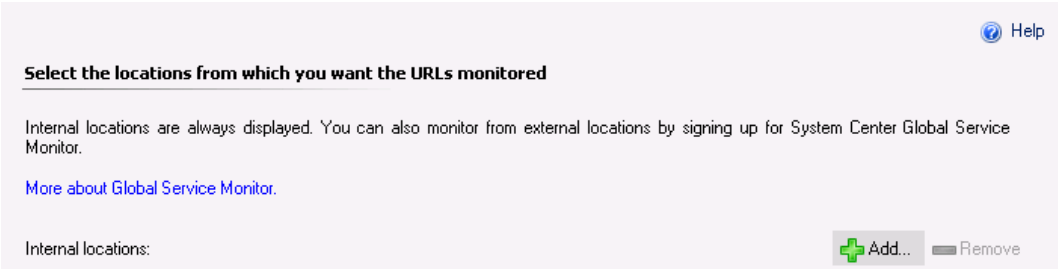
5. Select **Contoso Management** in the **Management Pack** drop down box and click **Next**



6. In the **Name** field type **Sales Application** and in the **URL** field type **HTTP://Guest01** and click **Next**

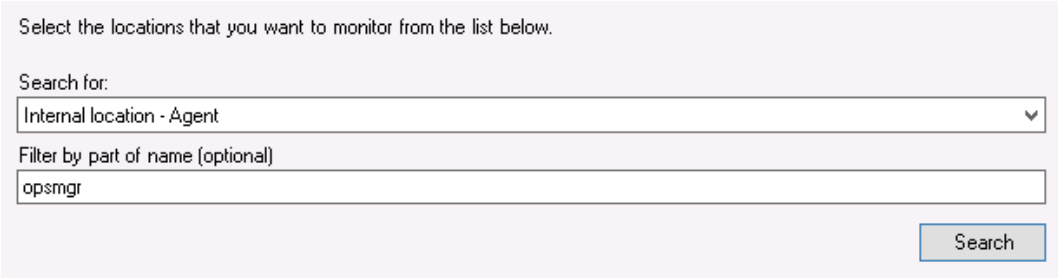


7. Click **Add**

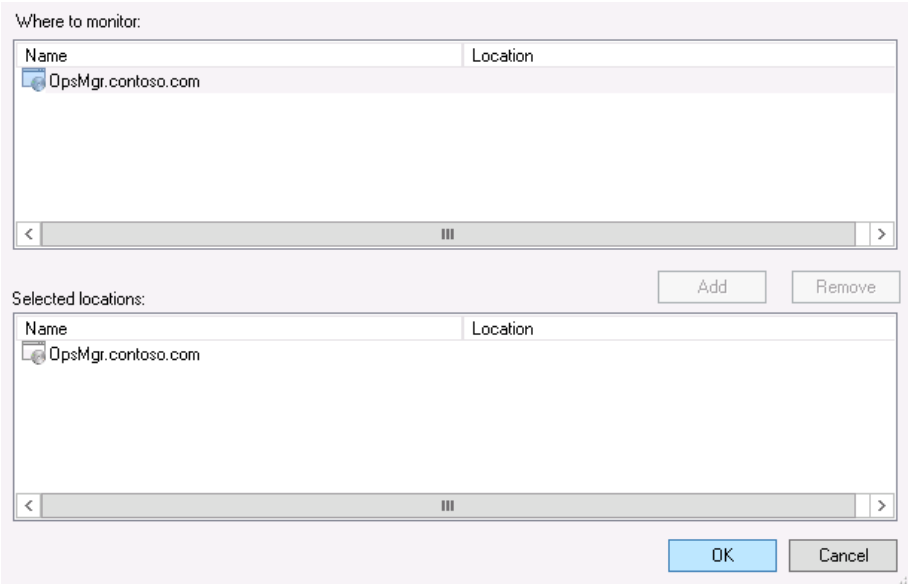


> **Important Note**
>
> At this point if GSM is enabled, you may select external servers in one of many global locations to run your web monitors from.

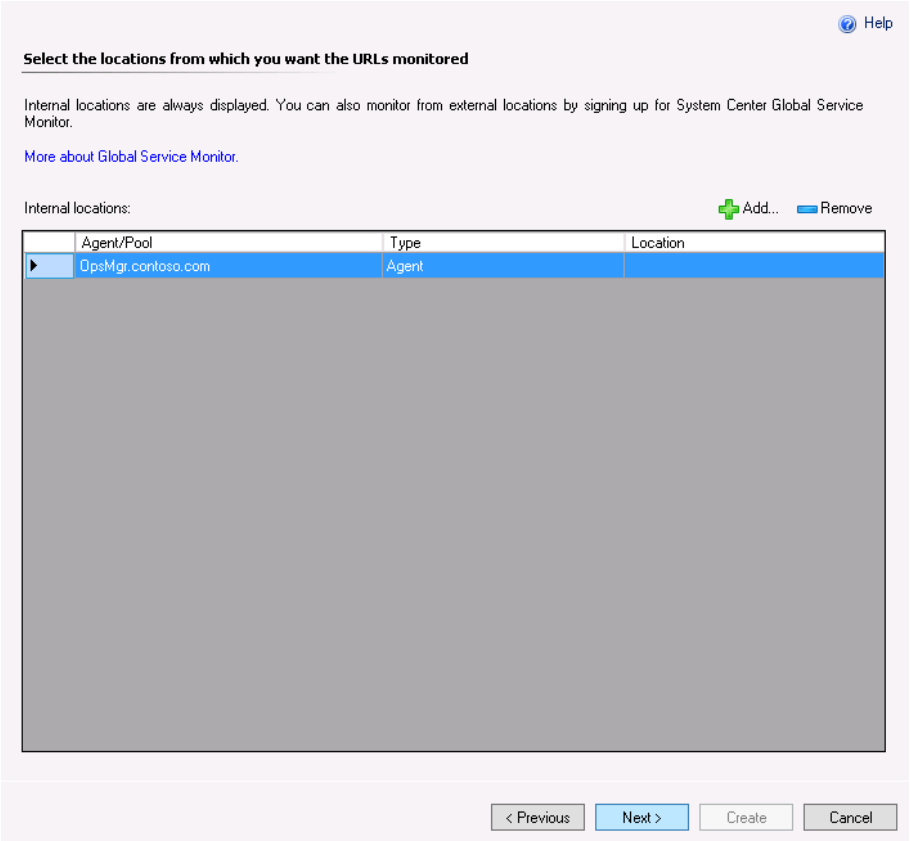
8. Type **OpsMgr** in the Filter area and click **Search**



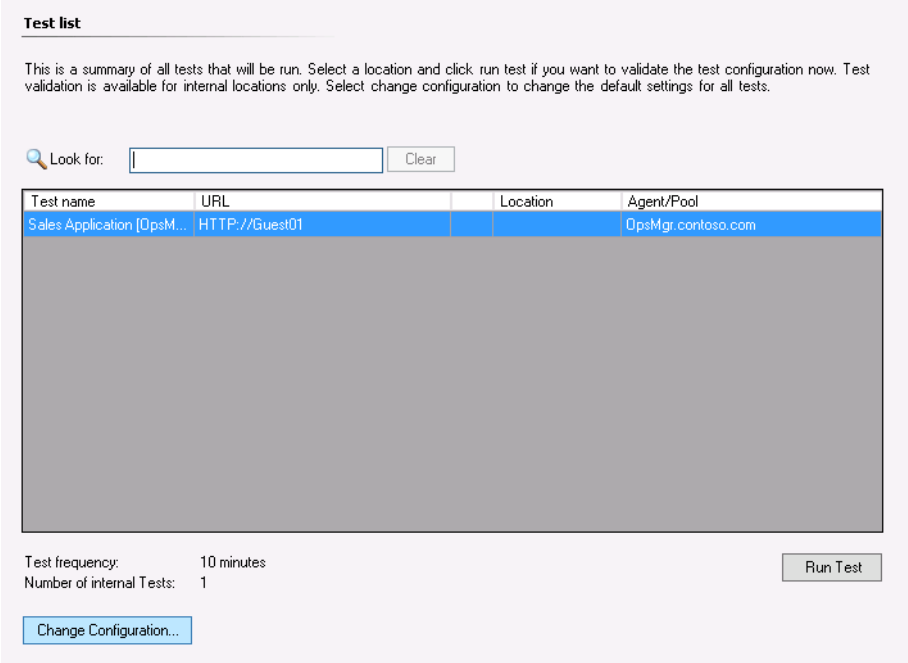
9. Click on **OpsMgr.contoso.com**, click **Add** and then **OK**



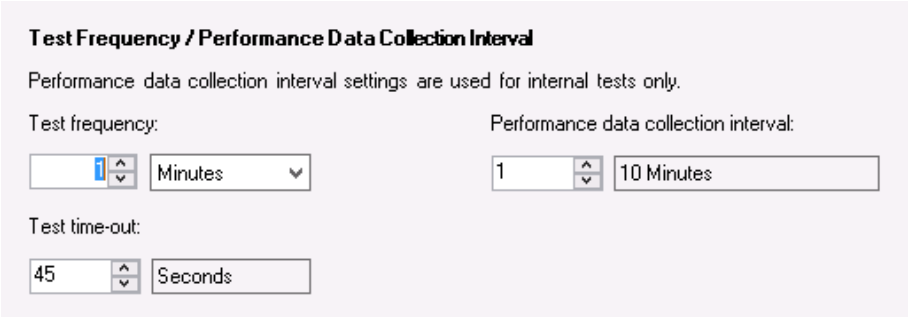
10. Click **Next**



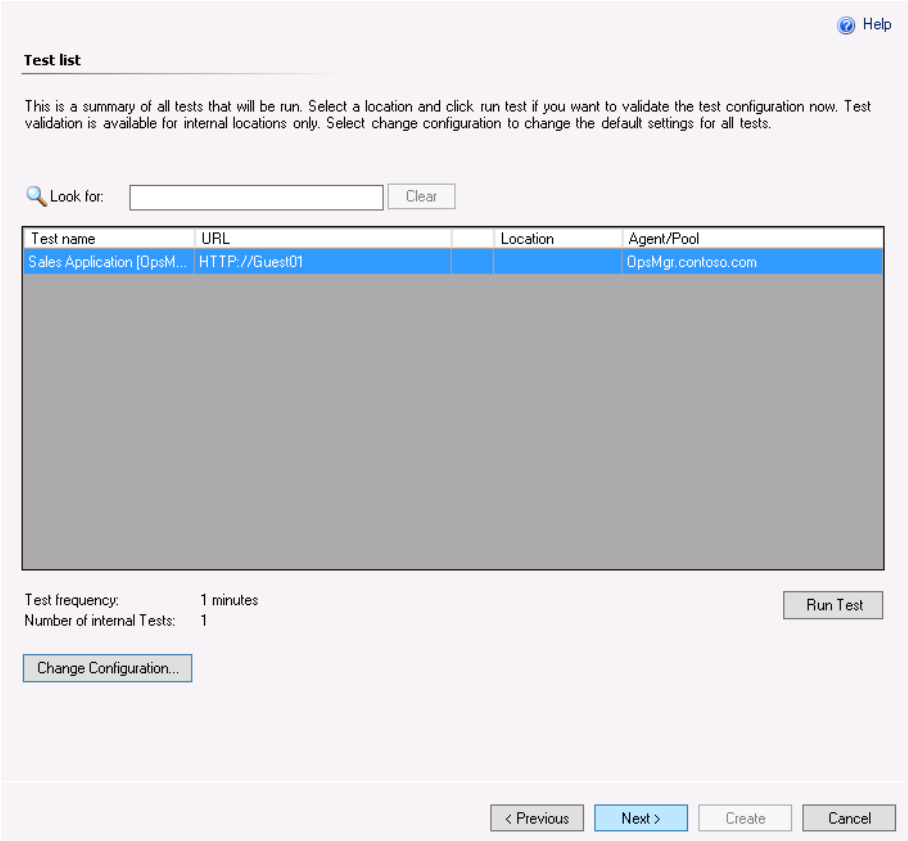
11. Click **Change Configuration** from the **Test List**



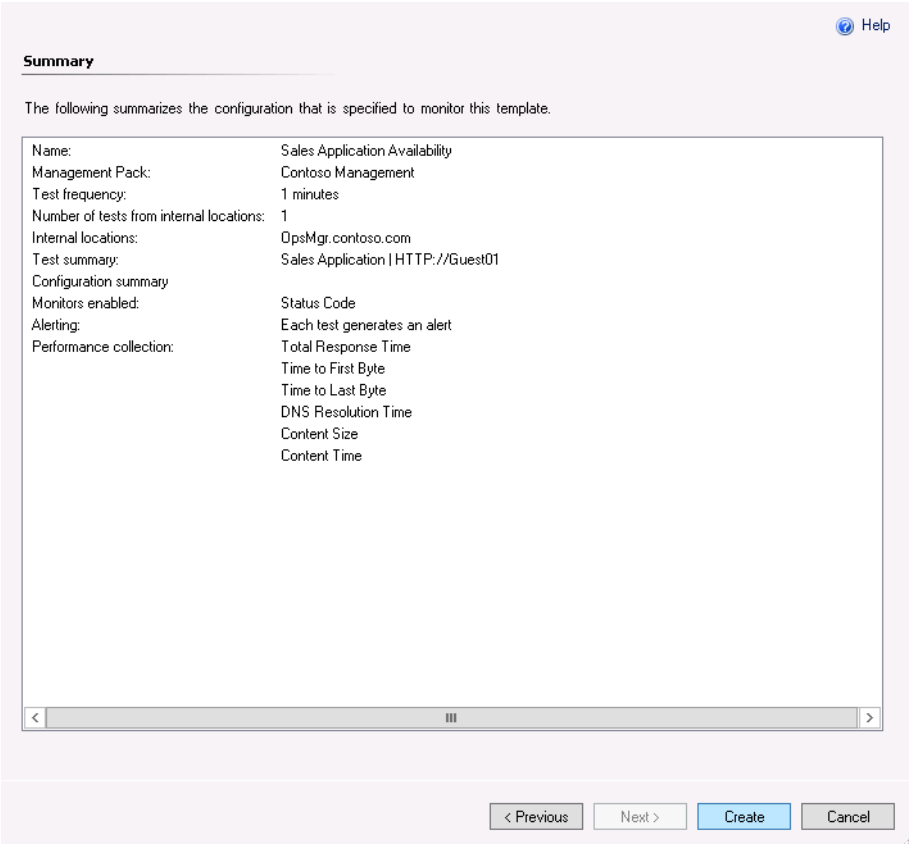
12. Change the **Test Frequency** to **1 minutes** and click **OK**



13. Click **Next**



14. On the **Summary** page click **Create**



Create a Distributed Application

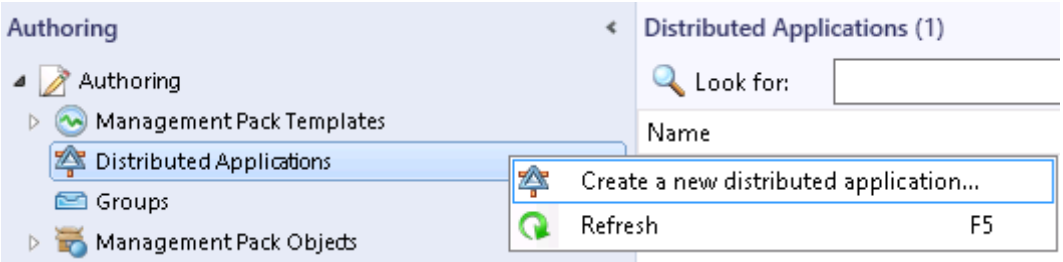
Now that Jeff has deployed his .Net Application monitor and Availability monitor he can start to draw a picture of the status of the Sales Application from front end to back end. This can be easily compiled using the Distributed Application monitor which can be utilised to create a single point of view of a service comprised of all of the individual components that the service relies upon.

Important Note

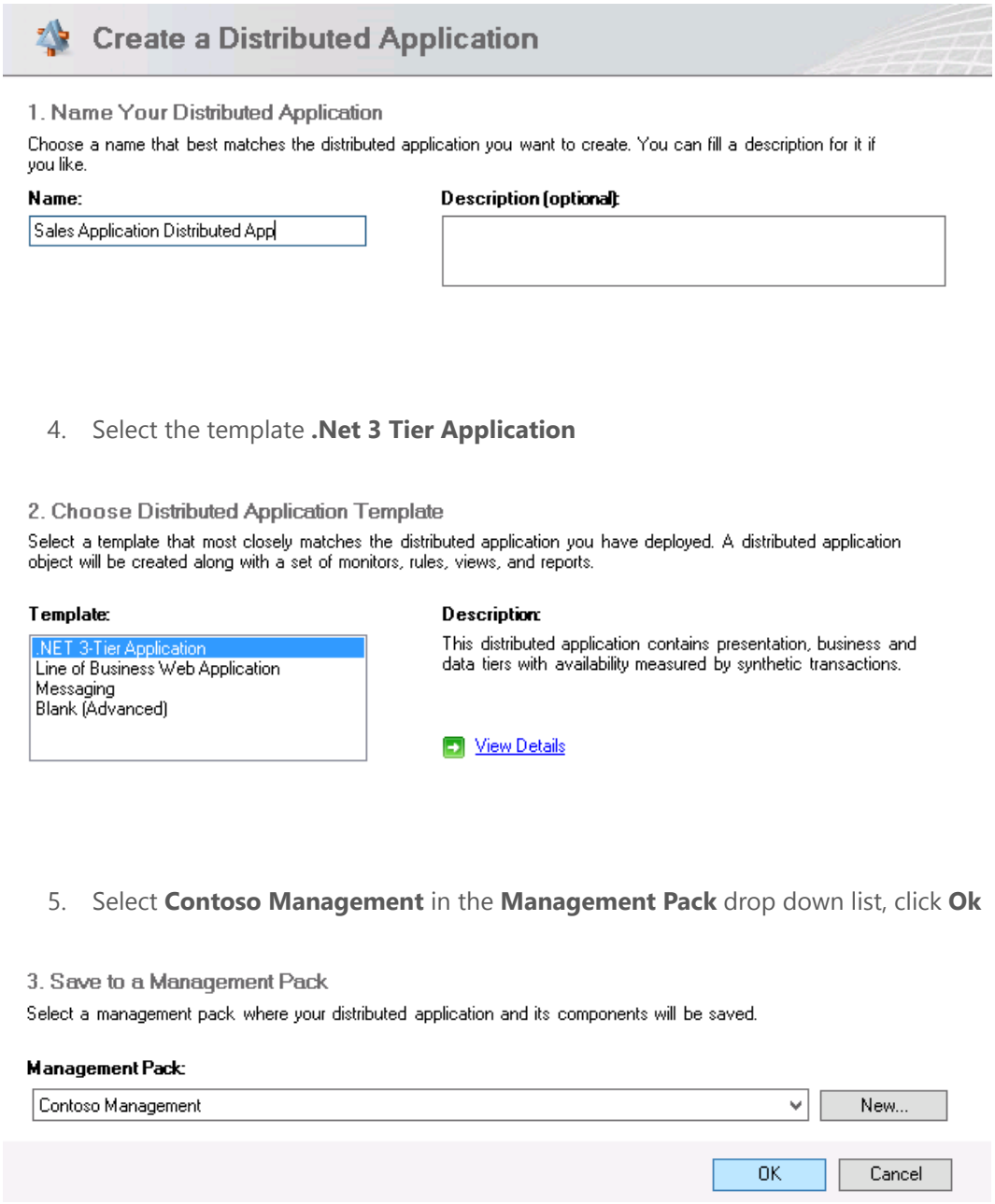
All devices and software that a specific service relies upon can be collected into a single monitor. For instance, a standard monitor for an Email application may report on database status, connector state and mail flow performance. Using a distributed application we can add in network infrastructure, internet access, other vendors mail filters, inter site connectivity and much more. This gives an overall status of the entire Email Service rather than a specific Mail Server status. This more closely aligns with an end user view of a service. For instance, an end user may be unable to access their email, but it may be a network device preventing them from doing so.

1. In the **Operations console** click on **Authoring**

2. Right click **Distributed Applications** and select **Create a New Distributed Application**



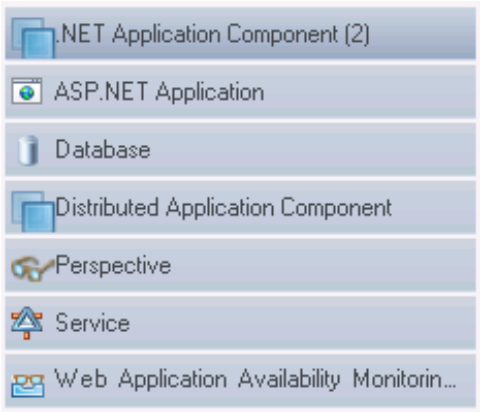
3. In the **Name** field type **Sales Application Distributed App**



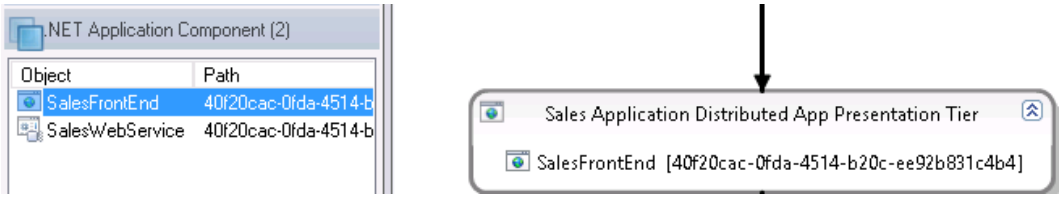
4. Select the template **.Net 3 Tier Application**

5. Select **Contoso Management** in the **Management Pack** drop down list, click **Ok**

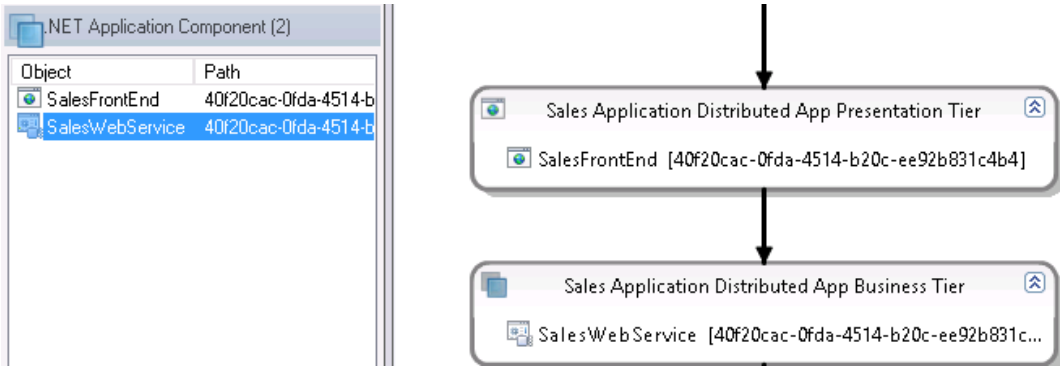
6. In the designer, click on **.Net Application Component** on the left



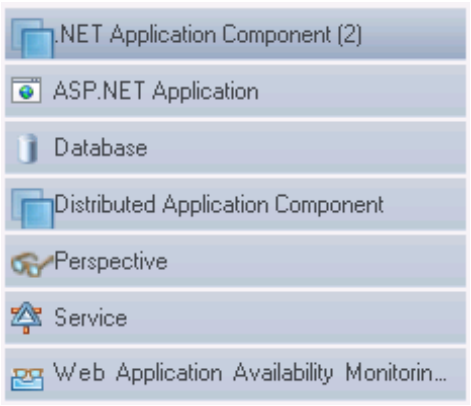
7. Drag **SalesFrontEnd** to the object in the diagram labelled **Sales Application Distributed App Presentation Tier**



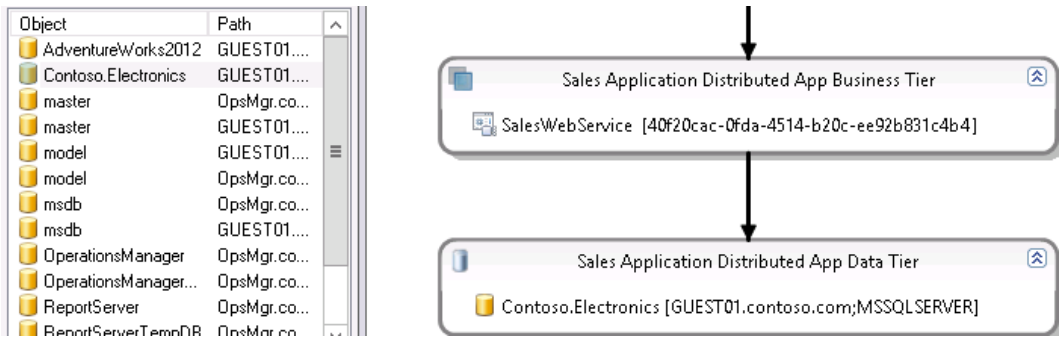
8. Drag **SalesWebService** to the **Sales Application Distributed App Business Tier**



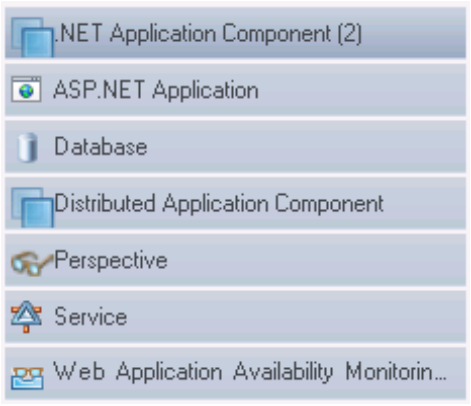
9. Click on **Database** on the left



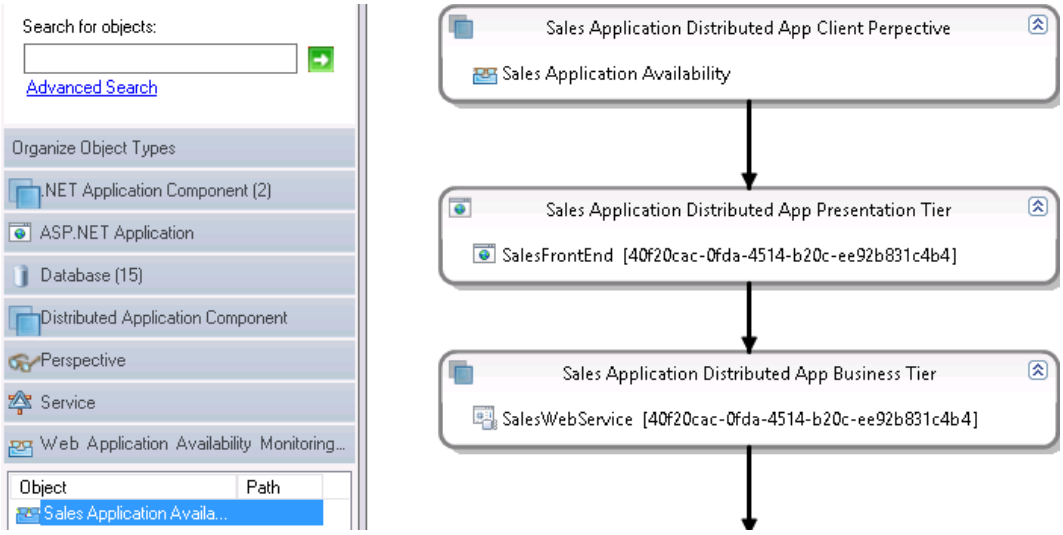
10. Drag the object **Contoso.Electronics on GUEST01** to the **Sales Application Distributed App Data Tier**



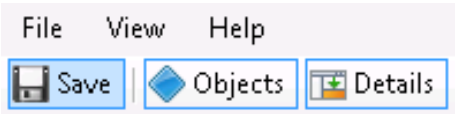
11. Click on **Web Application Availability Monitoring**



12. Drag the object **Sales Application Availability** to the **Sales Application Distributed App Client Perspective**



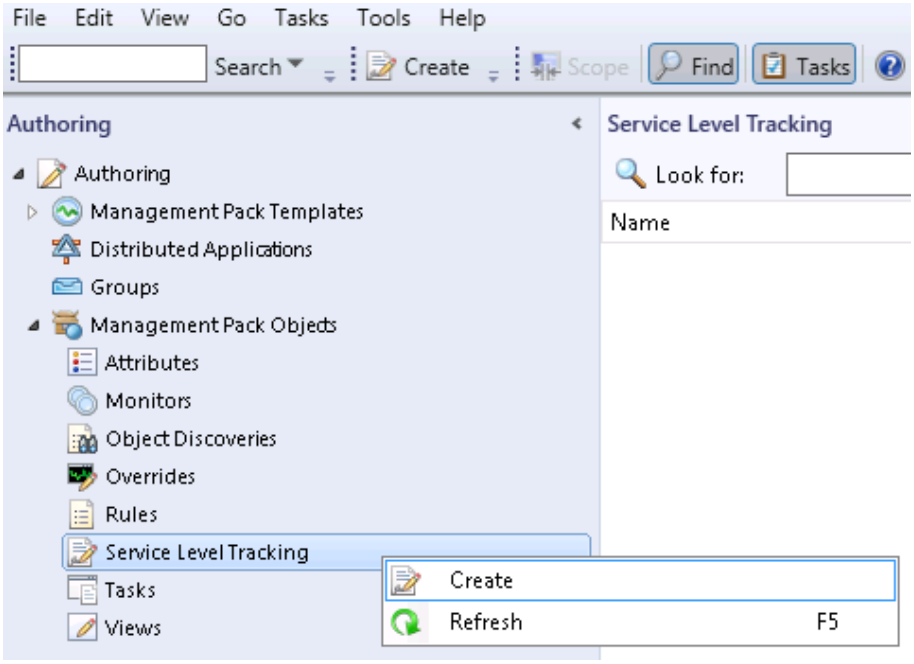
13. Click the **Save** button in the toolbar and then once saved, close the designer



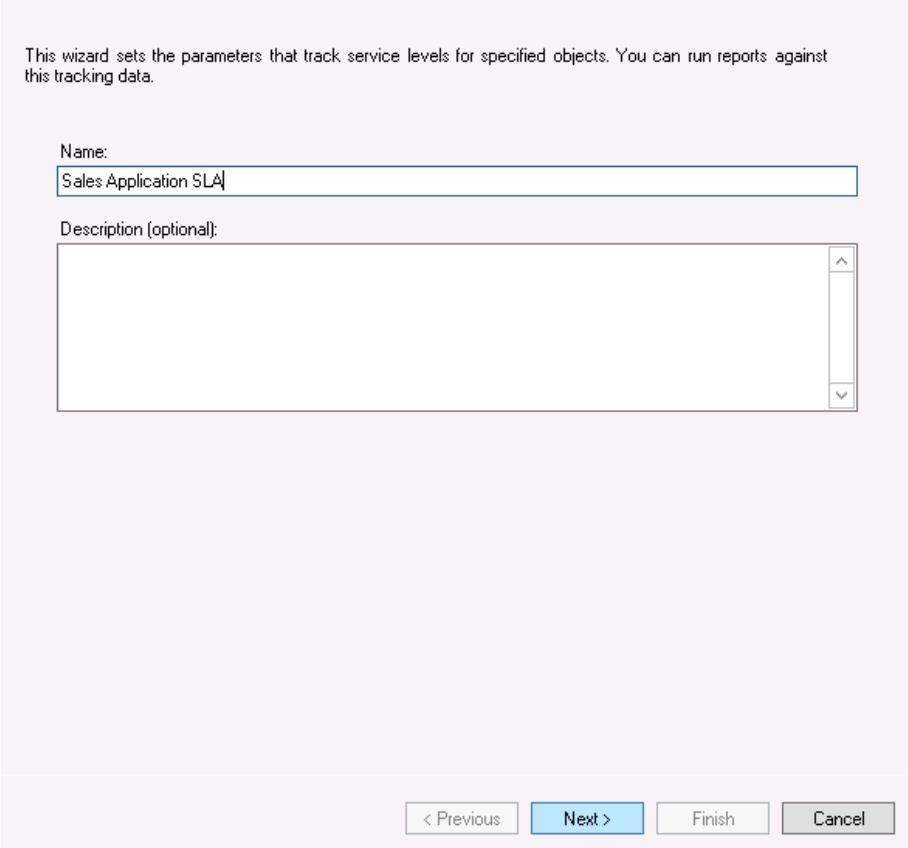
Create a Service Level Tracking object

Now that Jeff has created an overall picture of the Sales Application service, he can apply an SLA to that service. As decided previously, the business has decreed that the web application must be available 99% of the time so he creates an SLA that matches that requirement.

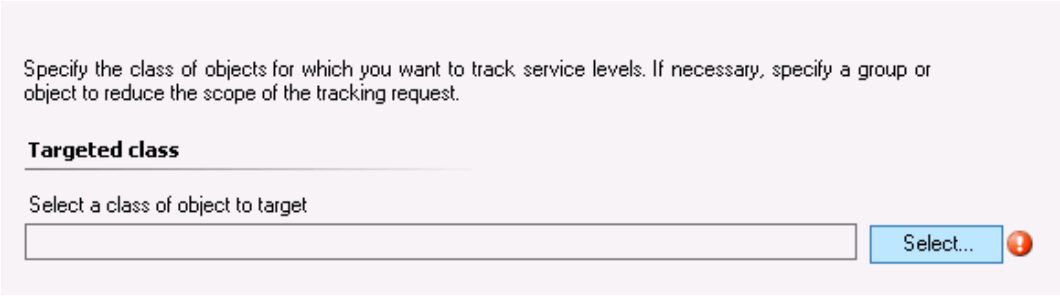
1. Expand **Management Pack Objects** and right click **Service Level Tracking** and select **Create**



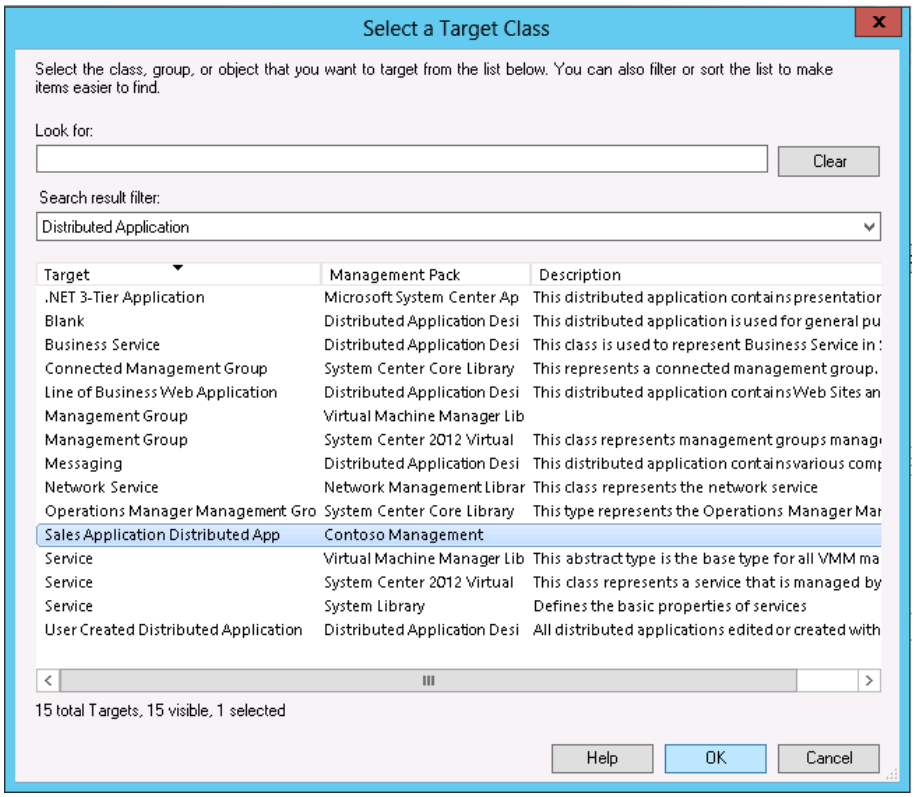
2. In the **Name** field type **Sales Application SLA** and click **Next**



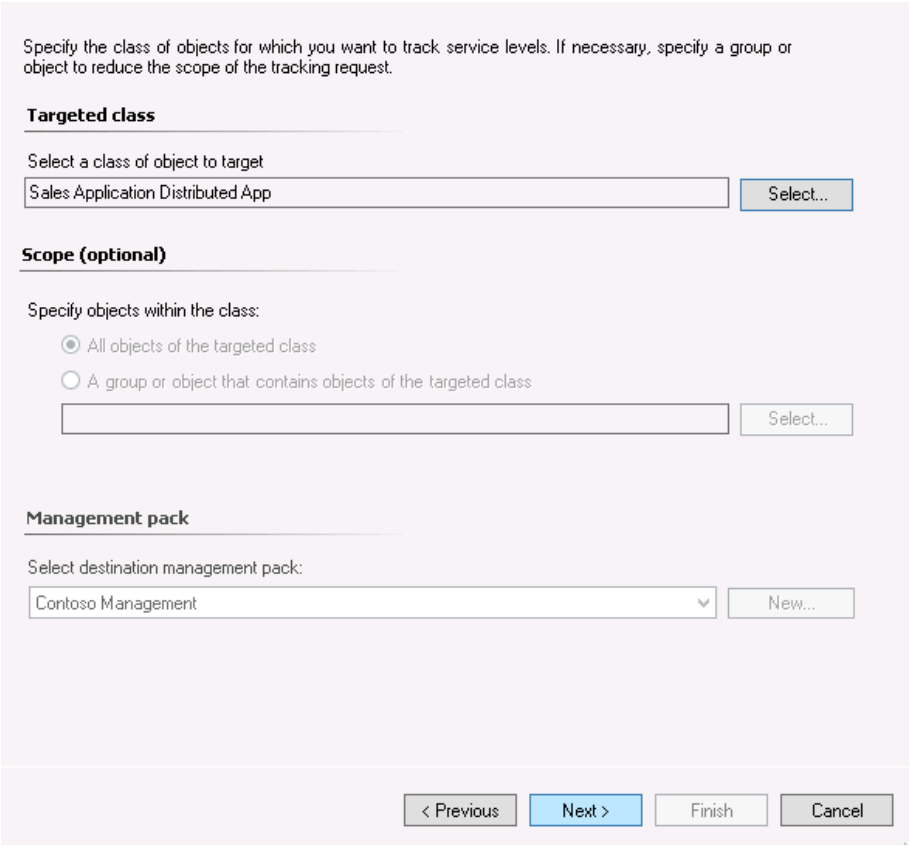
3. In the **Select a Class** field, click **Select**



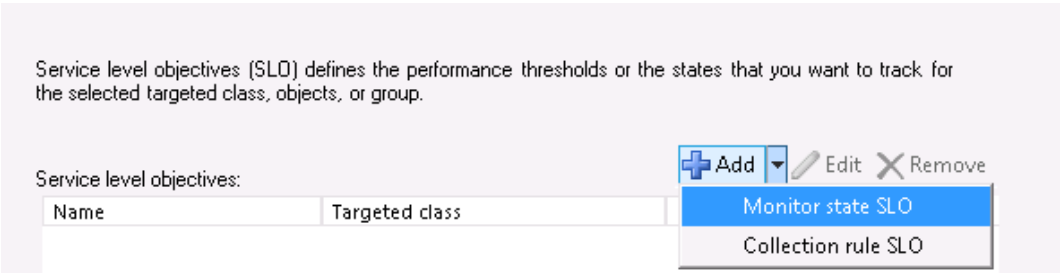
4. Click on the item in the list labelled **Sales Application Distributed App** and click **OK**


Select a Target Class
Select the class, group, or object that you want to target from the list below. You can also filter or sort the list to make items easier to find.
Look for:
Clear
Search result filter:
Distributed Application
Target
Management Pack
Description
.NET 3-Tier Application
Microsoft System Center Ap
This distributed application contains presentatior
Blank
Distributed Application Desi
This distributed application is used for general pu
Business Service
Distributed Application Desi
This class is used to represent Business Service in
Connected Management Group
System Center Core Library
This represents a connected management group.
Line of Business Web Application
Distributed Application Desi
This distributed application contains Web Sites an
Management Group
Virtual Machine Manager Lib
Management Group
System Center 2012 Virtual
This class represents management groups manag
Messaging
Distributed Application Desi
This distributed application contains various comp
Network Service
Network Management Librar
This class represents the network service
Operations Manager Management Gro
System Center Core Library
This type represents the Operations Manager Mar
Sales Application Distributed App
Contoso Management
Service
Virtual Machine Manager Lib
This abstract type is the base type for all VMM ma
Service
System Center 2012 Virtual
This class represents a service that is managed by
Service
System Library
Defines the basic properties of services
User Created Distributed Application
Distributed Application Desi
All distributed applications edited or created with
15 total Targets, 15 visible, 1 selected
Help
OK
Cancel

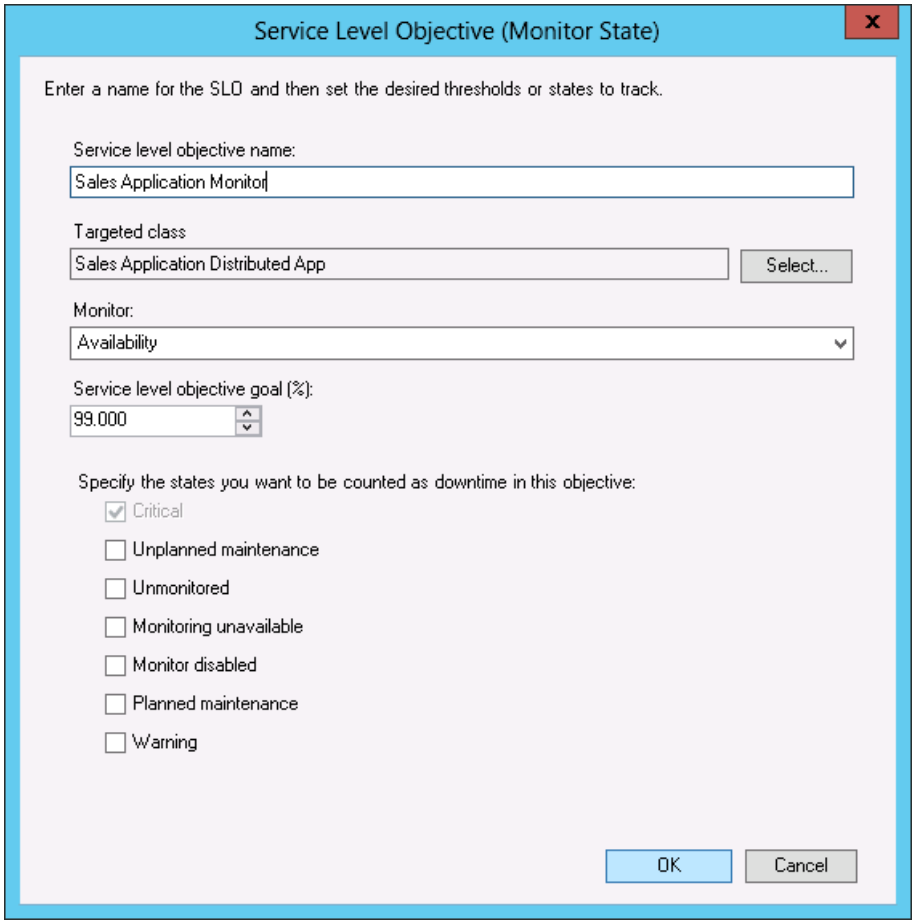
5. Click **Next**



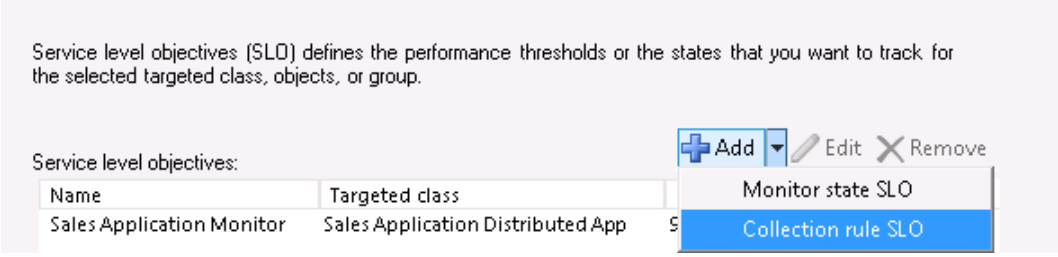
6. Click **Add** and select **Monitor state SLO**



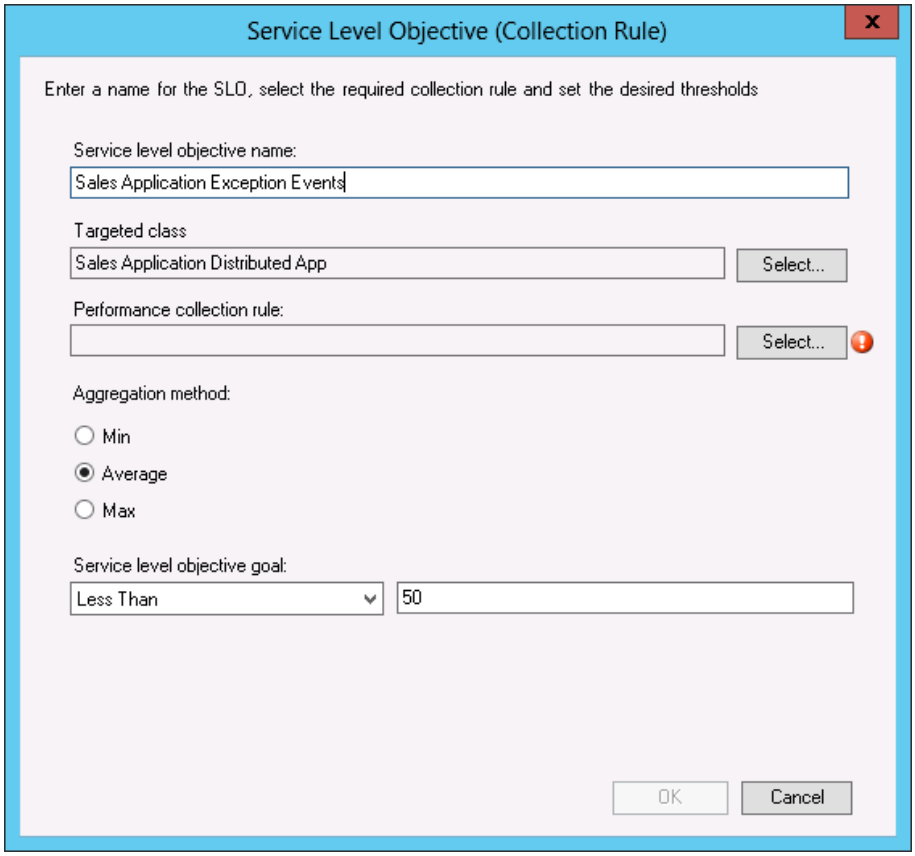
7. In the **Service level objective name** field type **Sales Application Monitor** and click **OK**



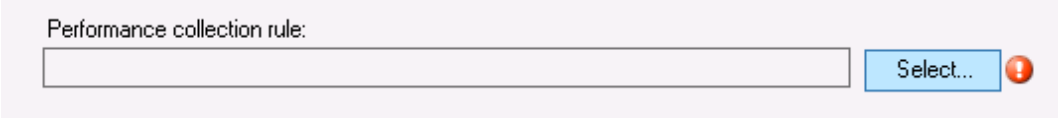
8. Click **Add** and select **Collection Rule SLO**



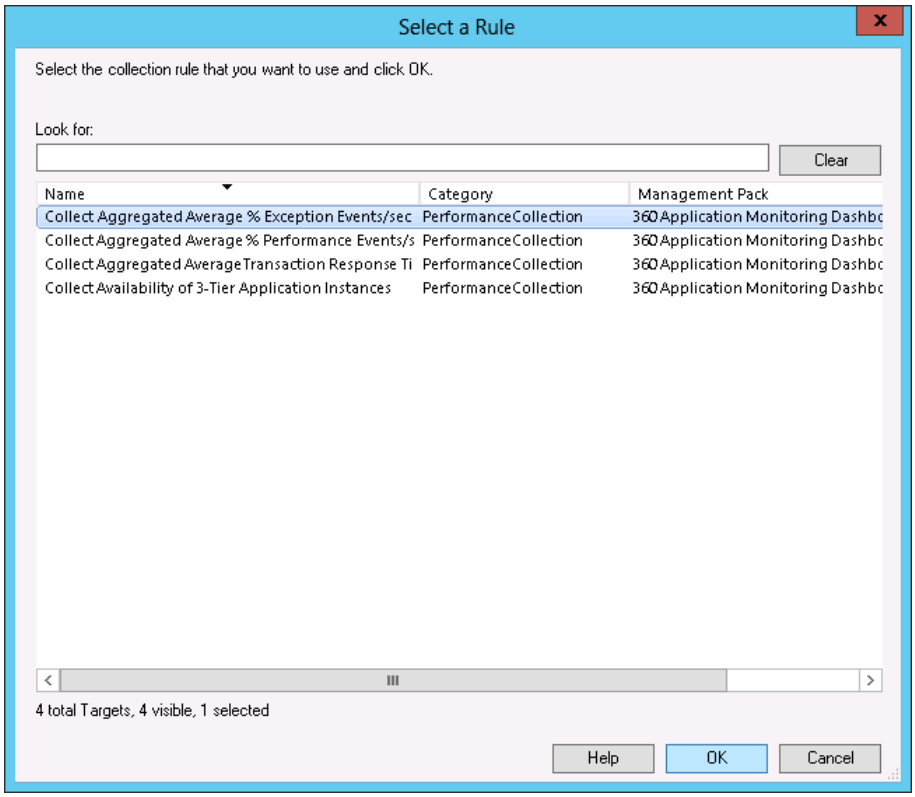
9. In the **Service Level Objective** name field type **Sales Application Exception Events**



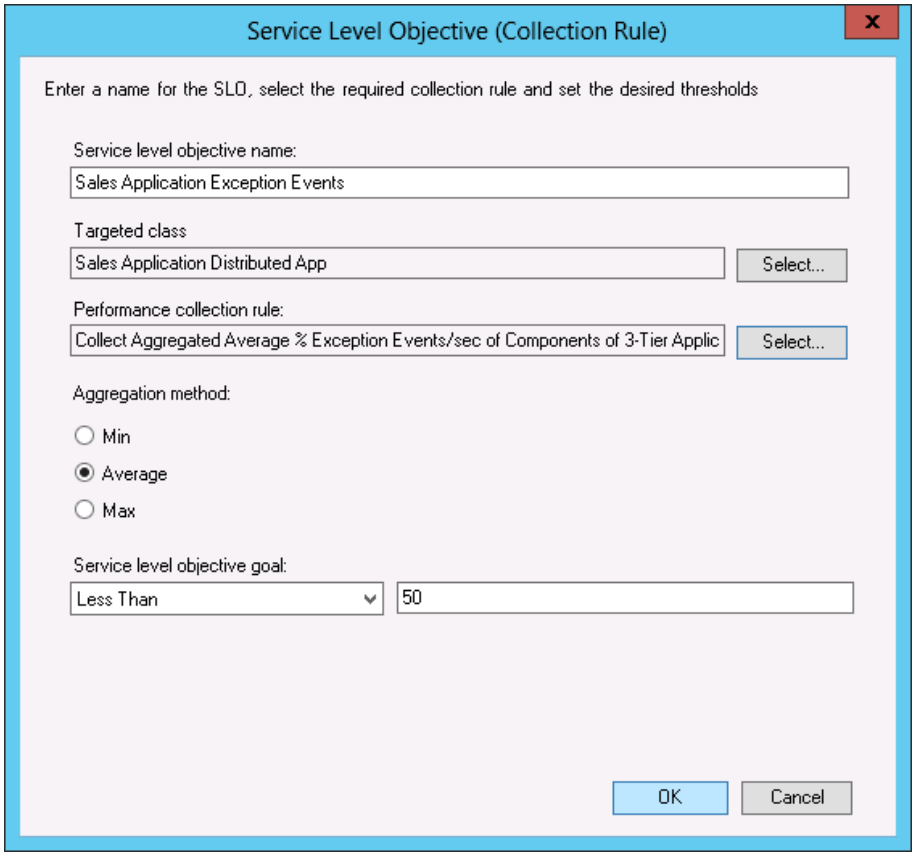
10. In the **Performance Collection** rule, click **Select**



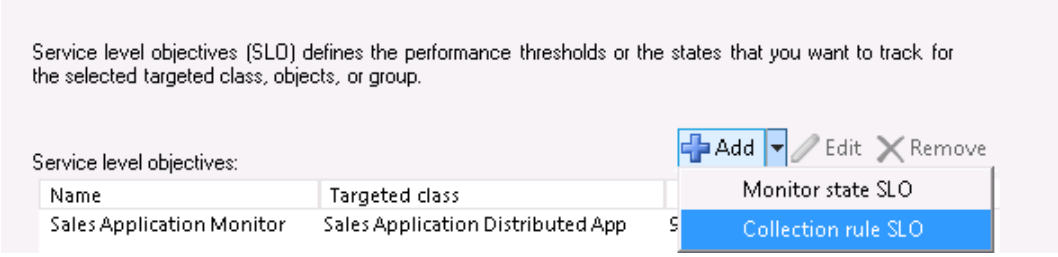
11. Click on **Collected Aggregated Average % Exception Events/Sec of Components of 3-Tier Application Instances** and click **OK**


Select a Rule
Select the collection rule that you want to use and click OK.
Look for:
Clear
Name
Category
Management Pack
Collect Aggregated Average % Exception Events/sec
PerformanceCollection
360 Application Monitoring Dashbc
Collect Aggregated Average % Performance Events/s
PerformanceCollection
360 Application Monitoring Dashbc
Collect Aggregated Average Transaction Response Ti
PerformanceCollection
360 Application Monitoring Dashbc
Collect Availability of 3-Tier Application Instances
PerformanceCollection
360 Application Monitoring Dashbc
4 total Targets, 4 visible, 1 selected
Help
OK
Cancel

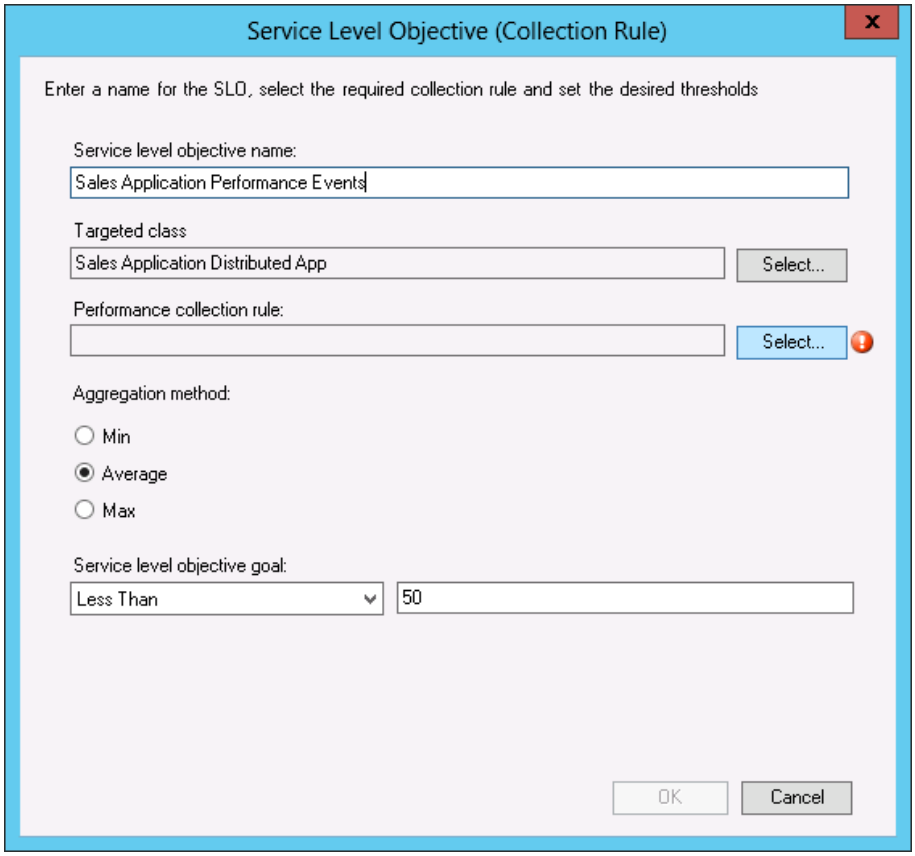
12. Click **OK**



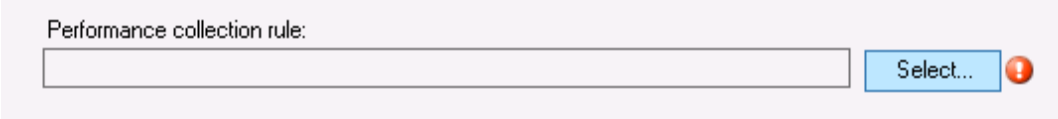
13. Click **Add** and select **Collection Rule SLO**



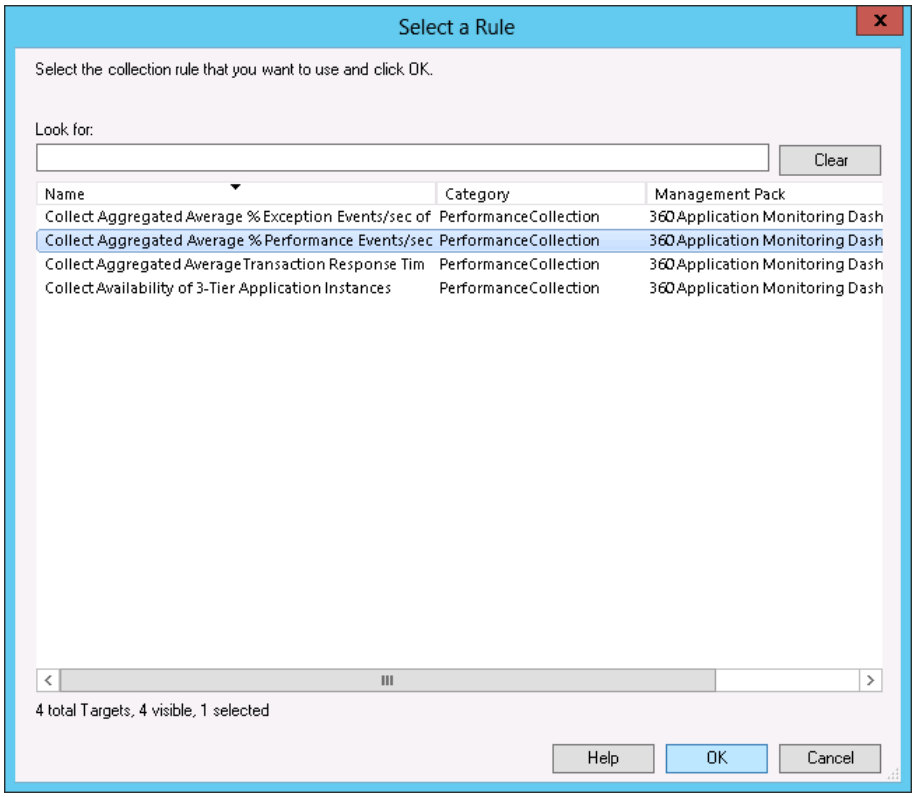
14. In the **Service Level Objective** name field type **Sales Application Performance Events**



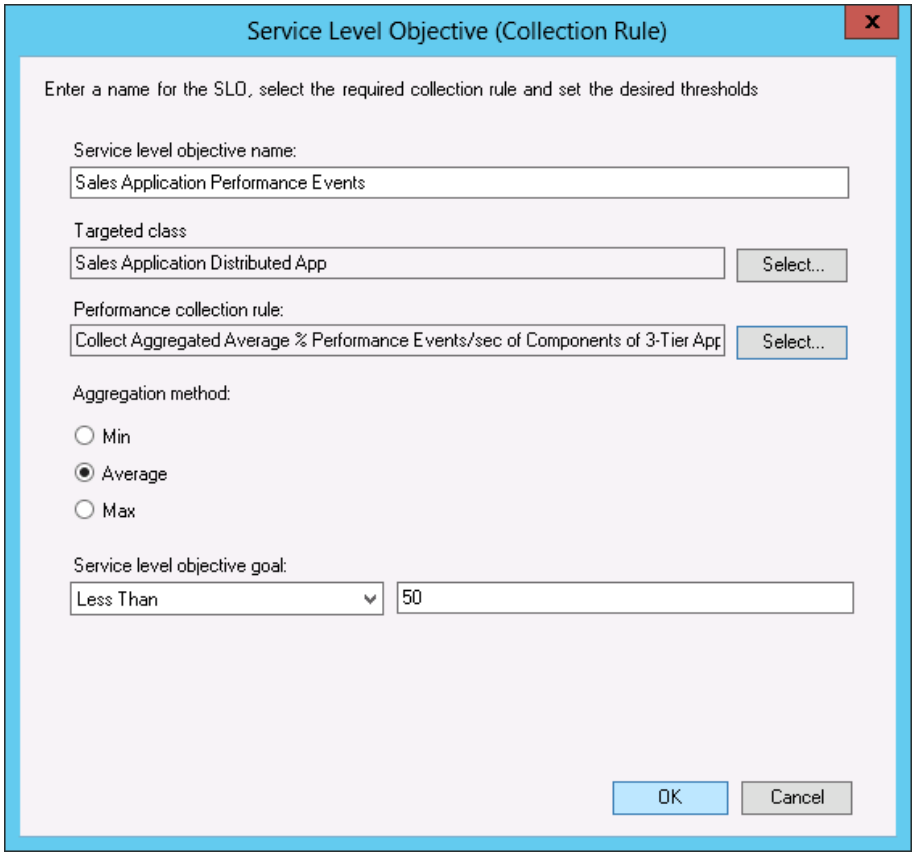
15. In the **Performance Collection** rule, click **Select**



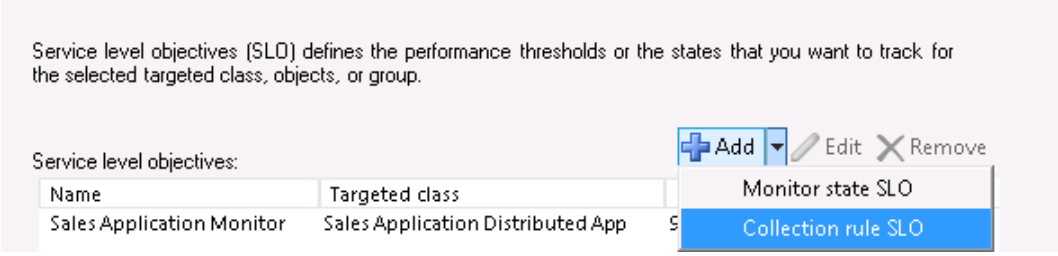
16. Click on **Collected Aggregated Average % Performance Events/sec of Components of 3-Tier Application Instances** and click **OK**


Select a Rule
Select the collection rule that you want to use and click OK.
Look for:
Clear
Name
Category
Management Pack
Collect Aggregated Average % Exception Events/sec of
PerformanceCollection
360 Application Monitoring Dash
Collect Aggregated Average % Performance Events/sec
PerformanceCollection
360 Application Monitoring Dash
Collect Aggregated Average Transaction Response Tim
PerformanceCollection
360 Application Monitoring Dash
Collect Availability of 3-Tier Application Instances
PerformanceCollection
360 Application Monitoring Dash
4 total Targets, 4 visible, 1 selected
Help
OK
Cancel

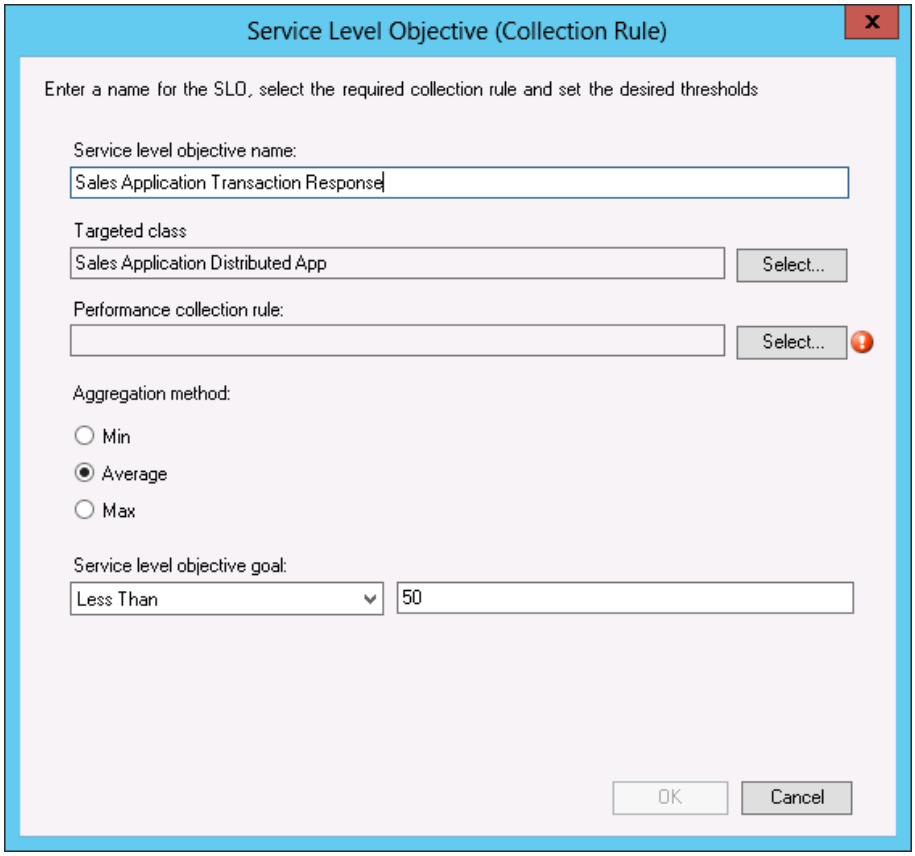
17. Click **OK**



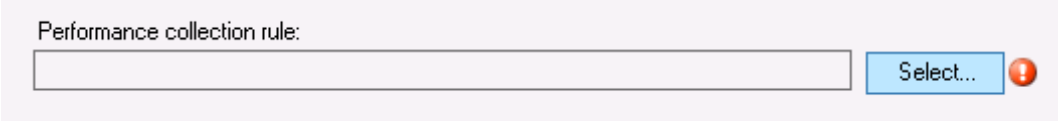
18. Click **Add** and select **Collection Rule SLO**



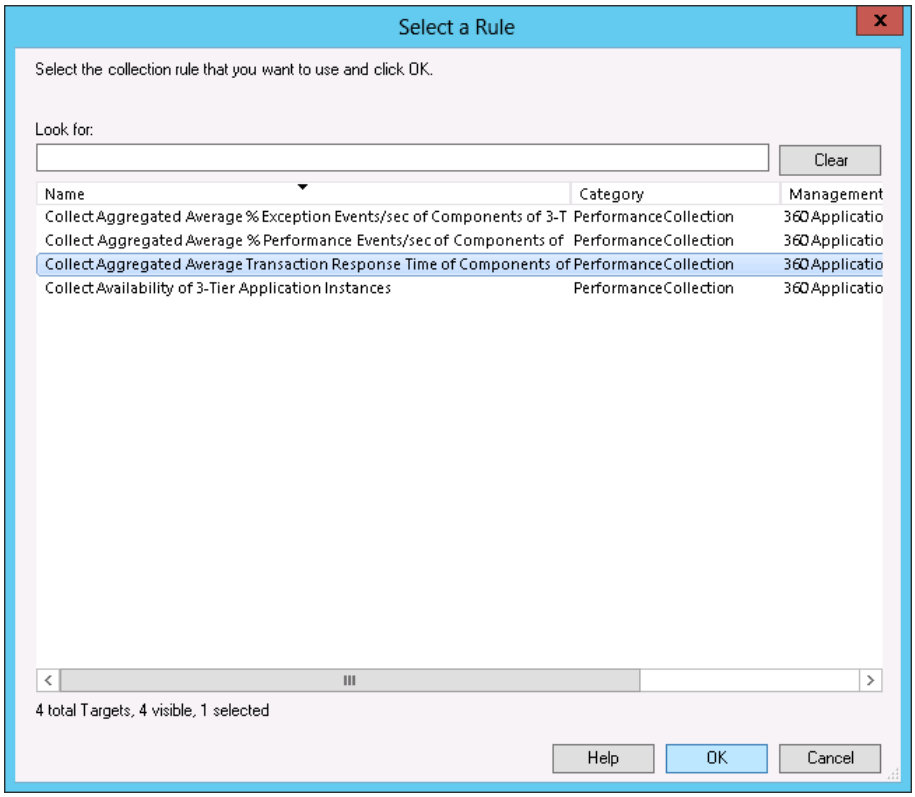
19. In the **Service Level Objective** name field type **Sales Application Transaction Response**



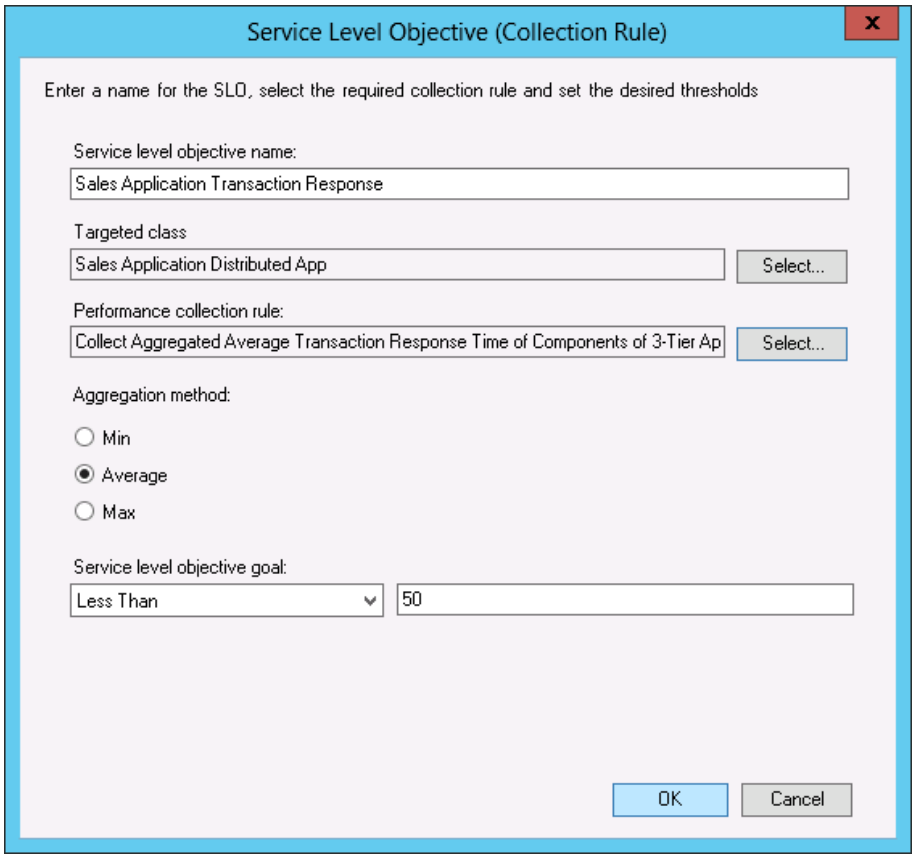
20. In the **Performance Collection** rule, click **Select**



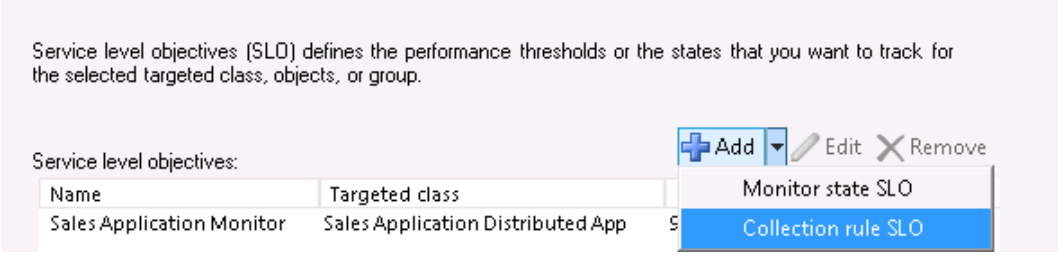
21. Click on **Collected Aggregated Average Transaction Response Time** and click **OK**


Select a Rule
Select the collection rule that you want to use and click OK.
Look for:
Clear
Name
Category
Management
Collect Aggregated Average % Exception Events/sec of Components of 3-T
PerformanceCollection
360.Applicatio
Collect Aggregated Average % Performance Events/sec of Components of
PerformanceCollection
360.Applicatio
Collect Aggregated Average Transaction Response Time of Components of
PerformanceCollection
360.Applicatio
Collect Availability of 3-Tier Application Instances
PerformanceCollection
360.Applicatio
4 total Targets, 4 visible, 1 selected
Help
OK
Cancel

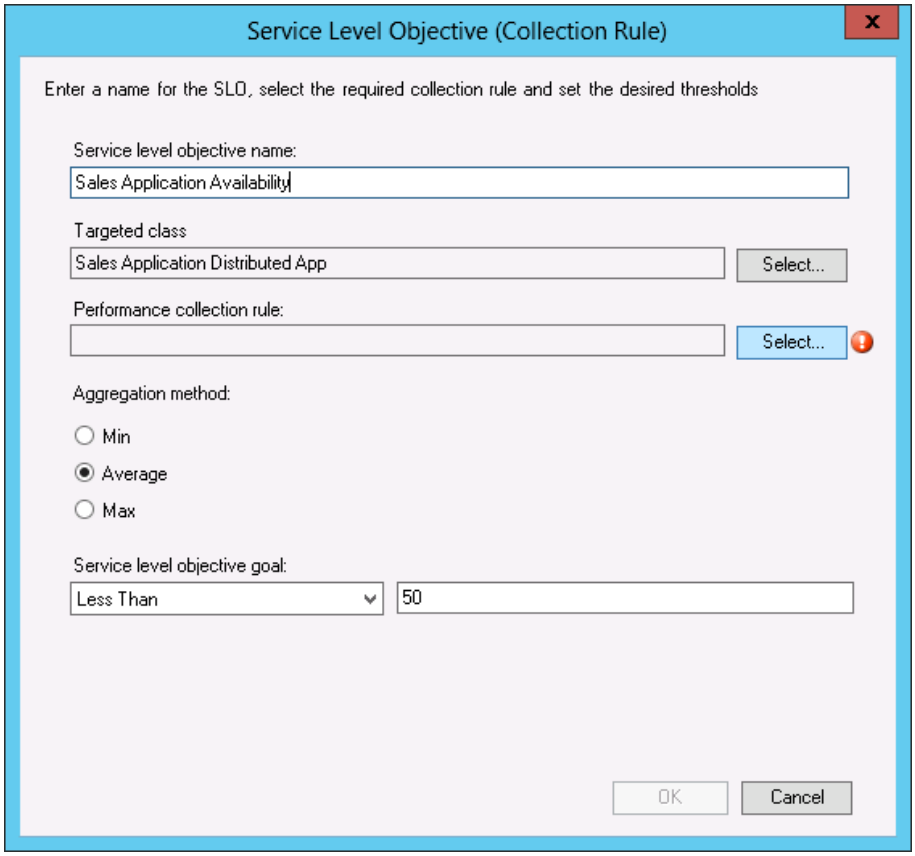
22. Click **OK**



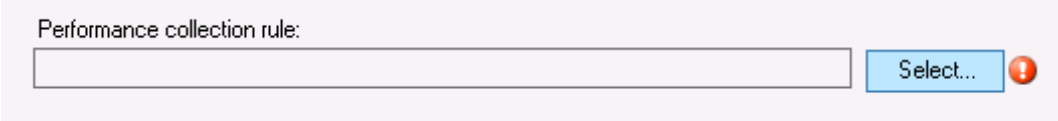
23. Click **Add** and select **Collection Rule**



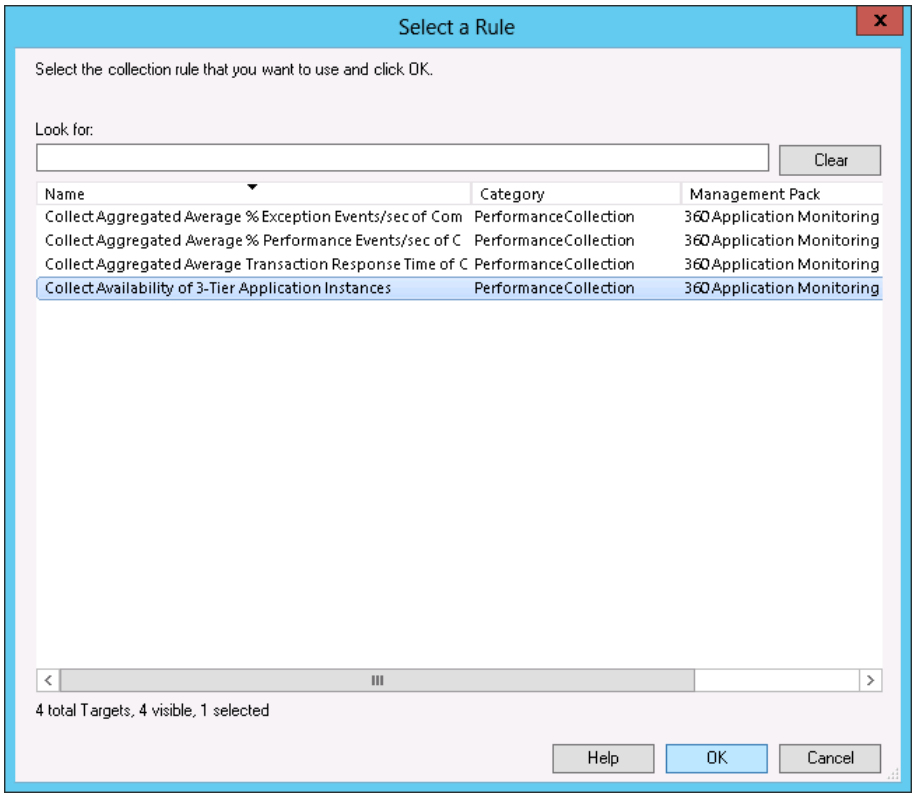
24. In the **Service Level** Objective name field type **Sales Application Availability**



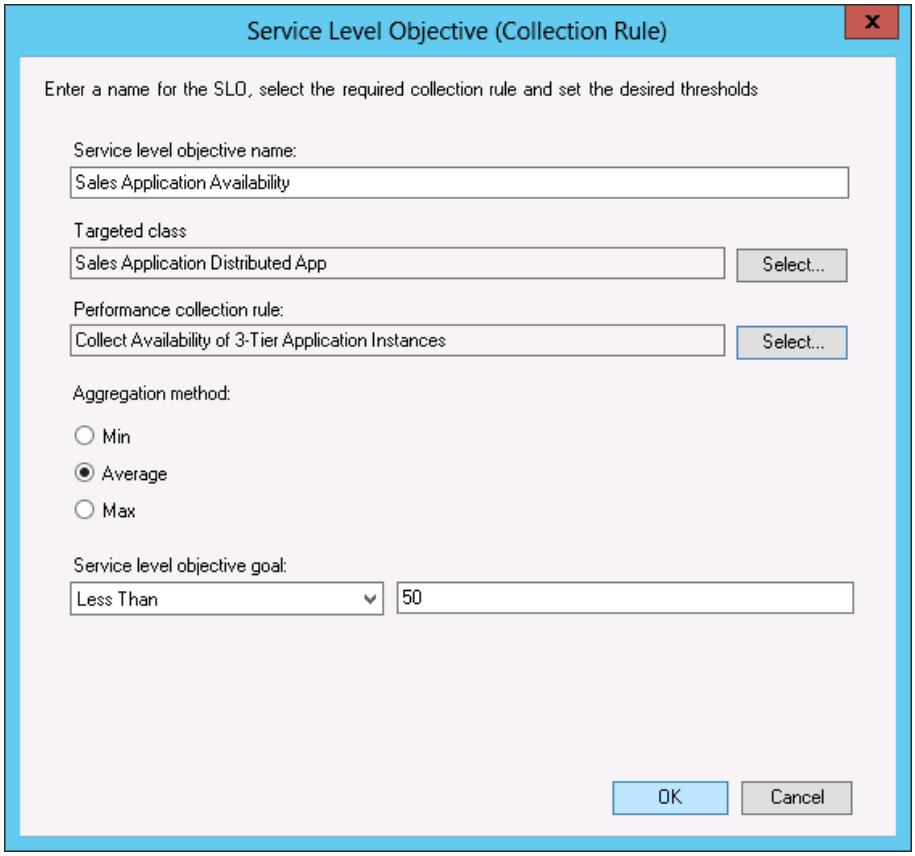
25. In the **Performance Collection** rule, click **Select**



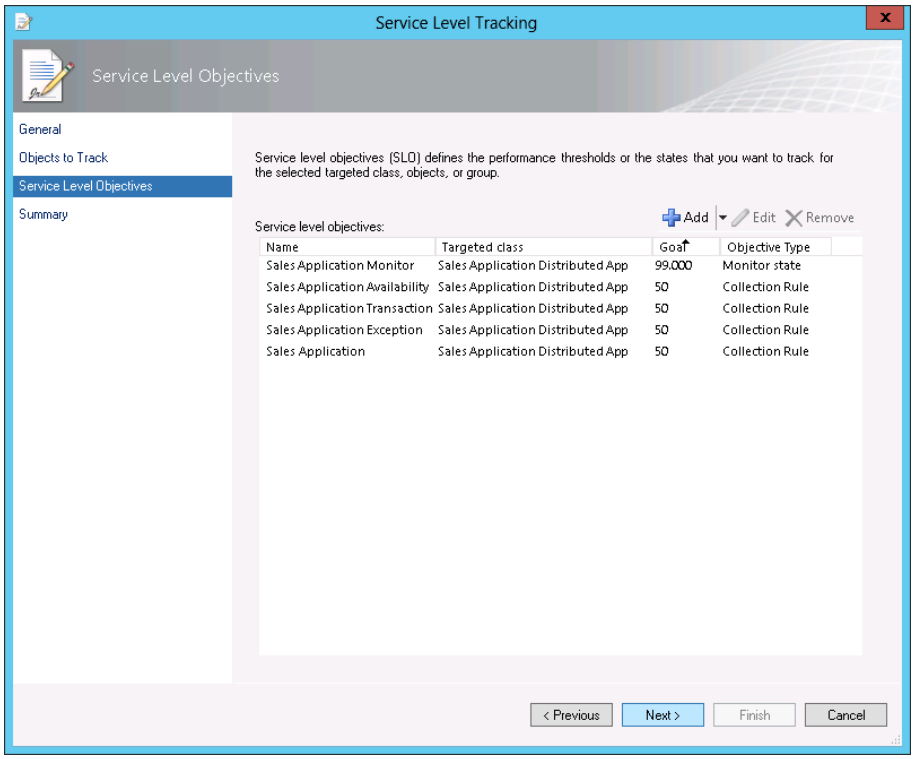
26. Click on **Collect Availability of 3 Tier Application Instances** and click **OK**


Select a Rule
Select the collection rule that you want to use and click OK.
Look for:
Clear
Name
Category
Management Pack
Collect Aggregated Average % Exception Events/sec of Com
PerformanceCollection
360.Application Monitoring
Collect Aggregated Average % Performance Events/sec of C
PerformanceCollection
360.Application Monitoring
Collect Aggregated Average Transaction Response Time of C
PerformanceCollection
360.Application Monitoring
Collect Availability of 3-Tier Application Instances
PerformanceCollection
360.Application Monitoring
4 total Targets, 4 visible, 1 selected
Help
OK
Cancel

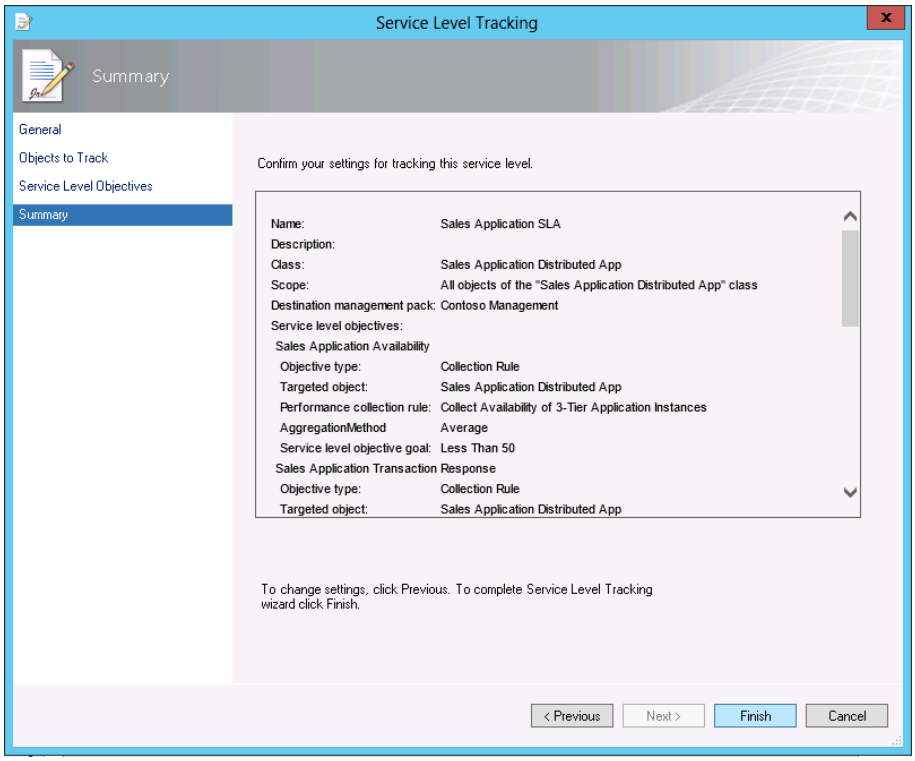
27. Click **OK**



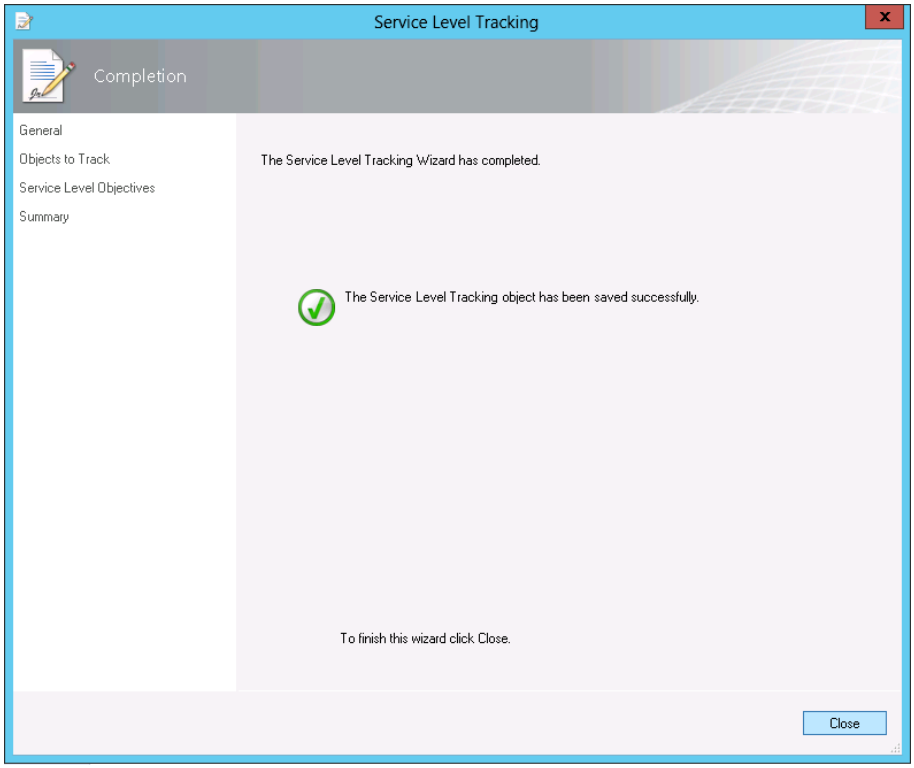
28. Click **Next**



29. Click **Finish**



30. Click **Close**



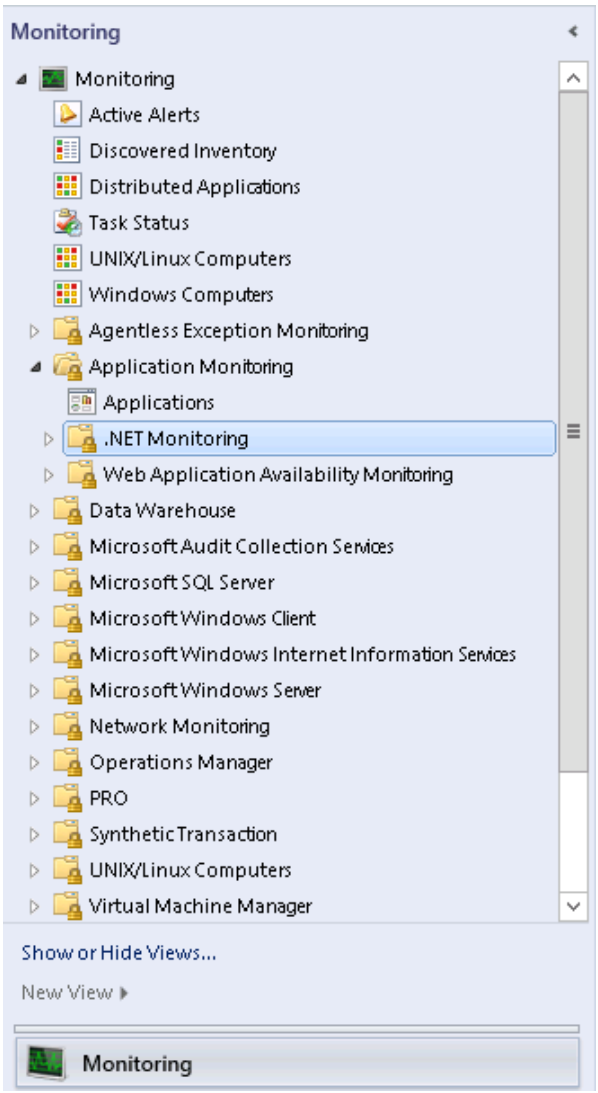
Create an SLA Dashboard

Once the SLA has been created, Jeff can create a dashboard that can show the status of the SLA. This will allow managers and support staff to see at a glance, the status of their Service Level Agreements.

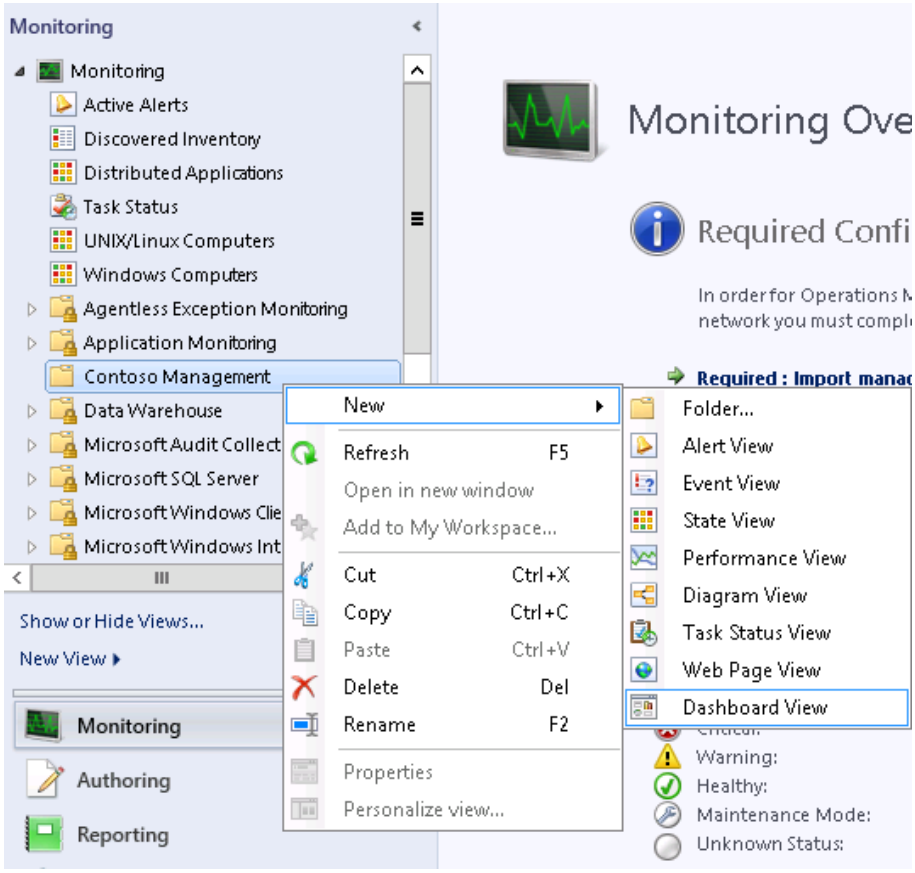
Important Note

Dashboards and their components can be published to SharePoint Portals in the organisation to allow for wider distribution of SLA Status information.

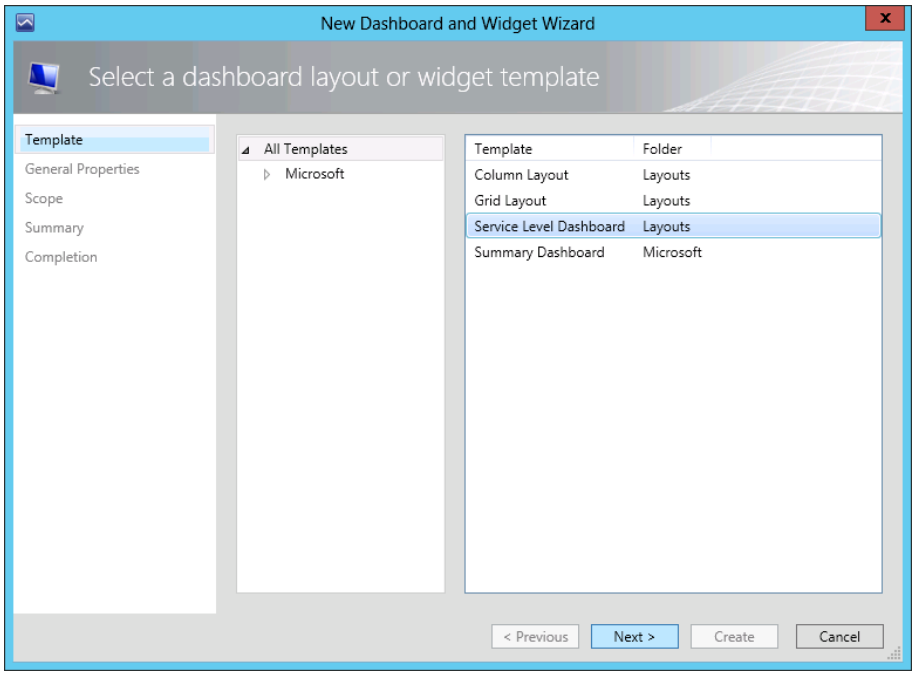
1. In the **Operations Manager** console, click on **Monitoring**



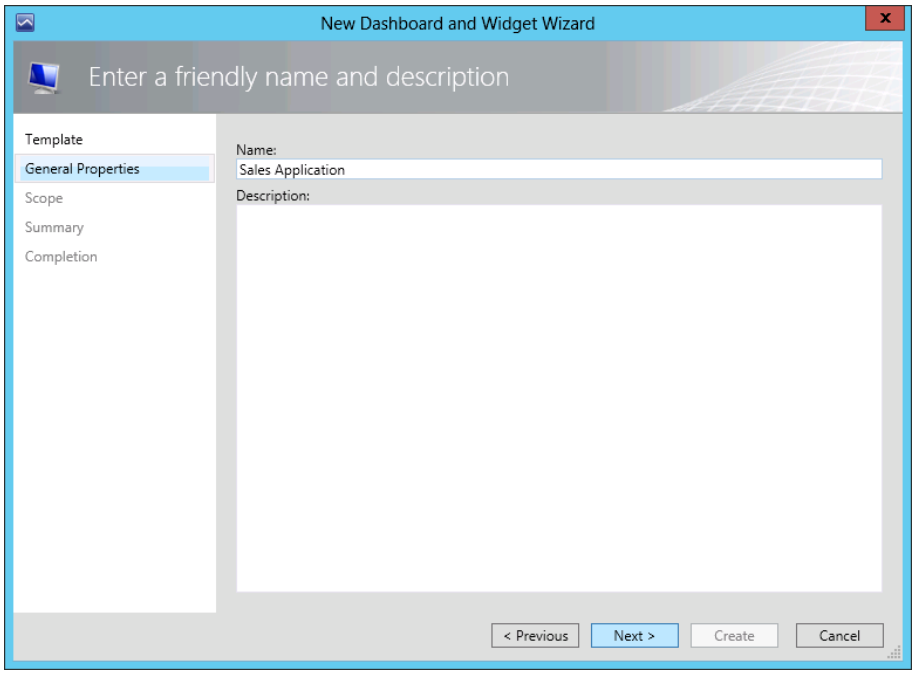
2. Right click on **Contoso Management** and select **New** then select **Dashboard View**



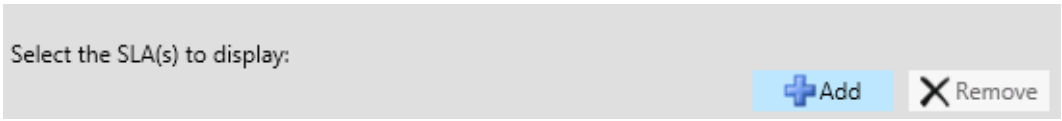
3. Select **Service Level Dashboard** and click **Next**



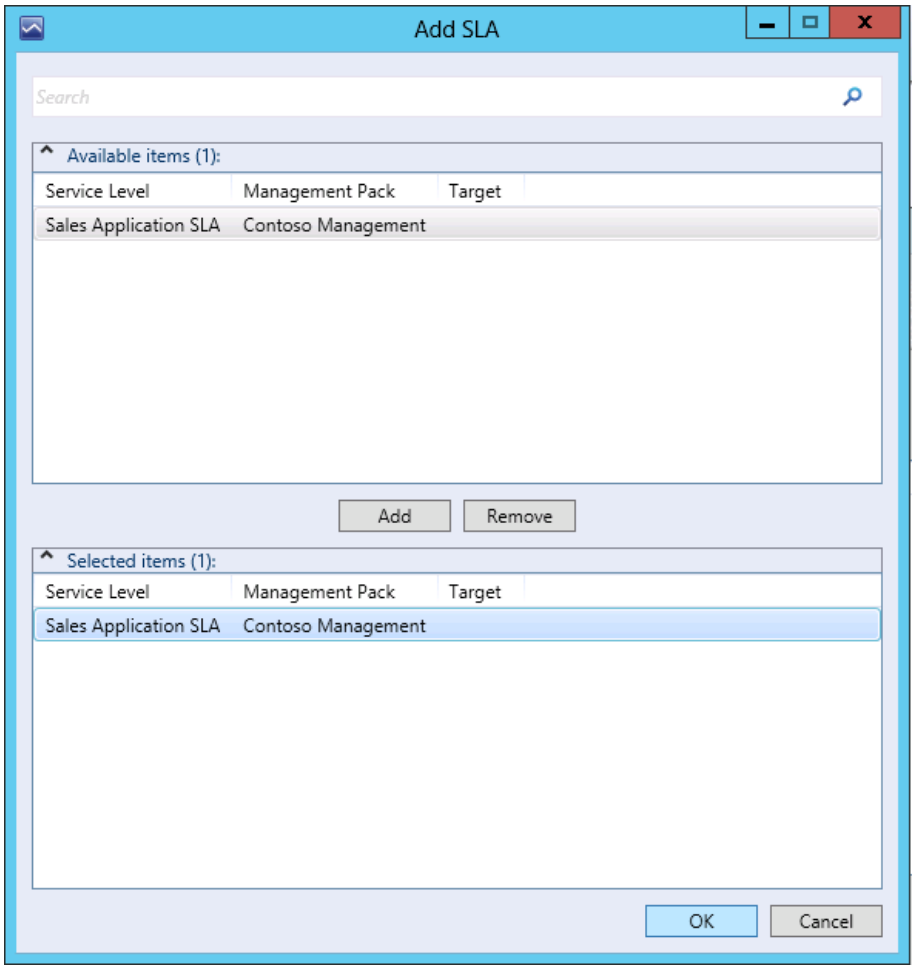
4. In the **Name** field type **Sales Application** and click **Next**



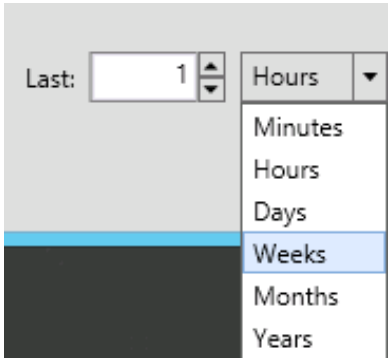
5. Click **Add**



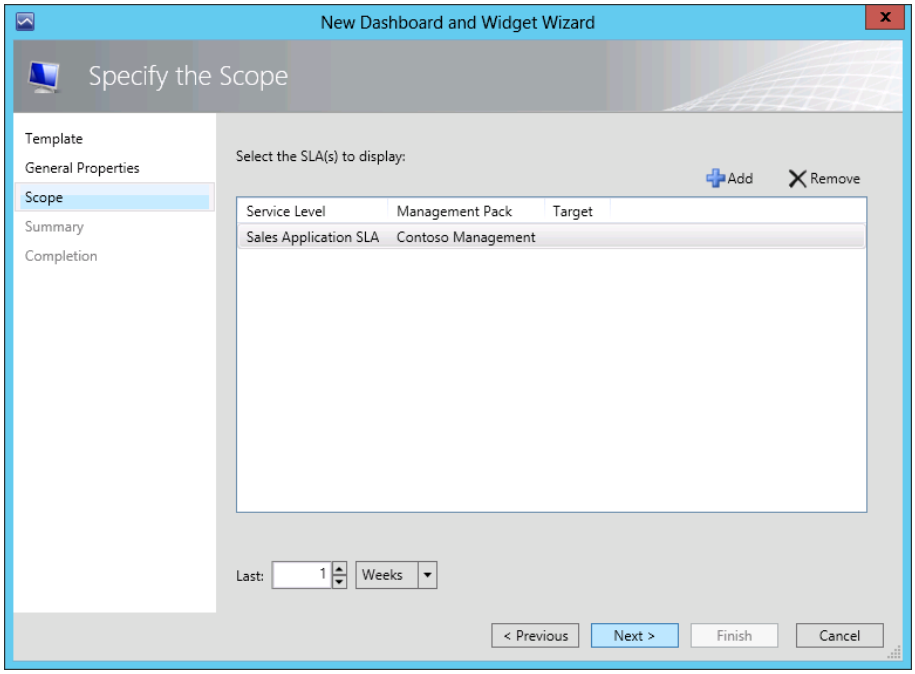
6. In the available items click on **Sales Application SLA** and click **Add** and click **OK**



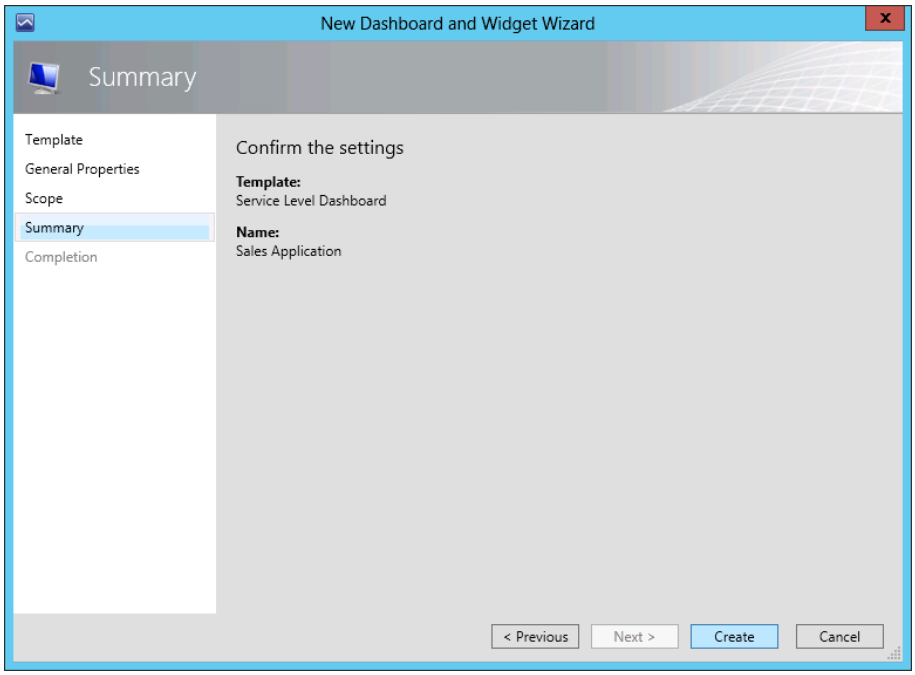
7. Change time period at the bottom to **1 weeks**



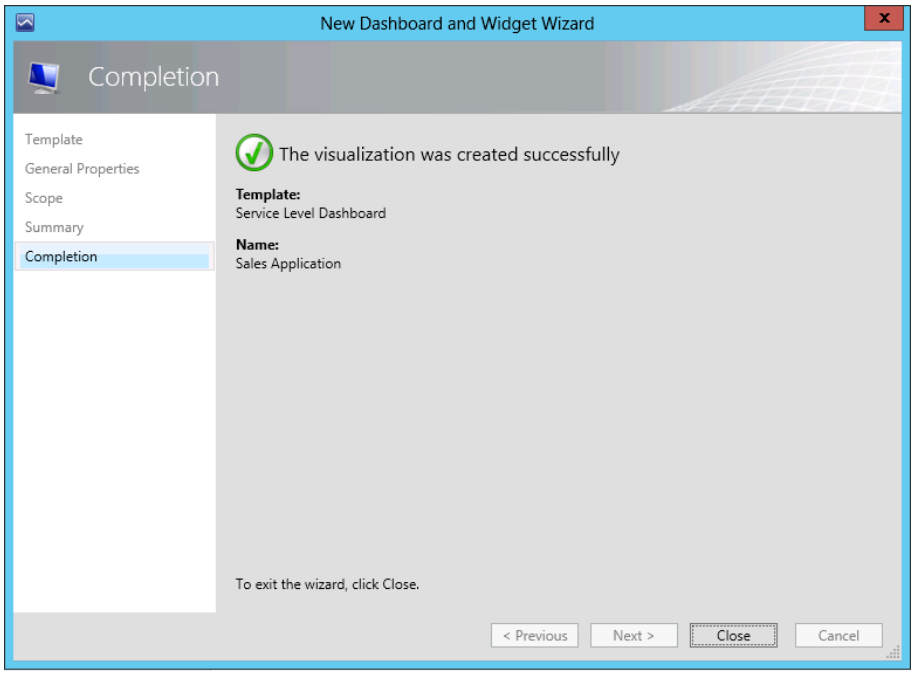
8. Click **Next**



9. Click **Create**



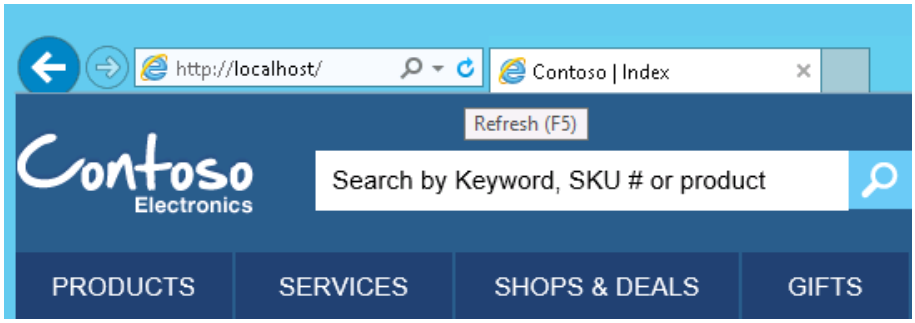
10. Click **Close**



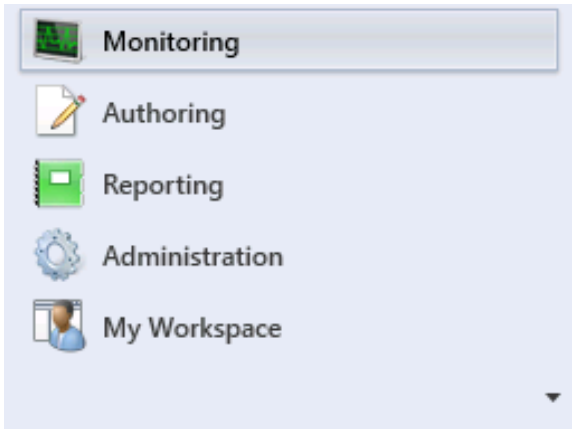
Generating the load and analysing the data

1. On **Guest01** open Internet Explorer and browse to **http://localhost**

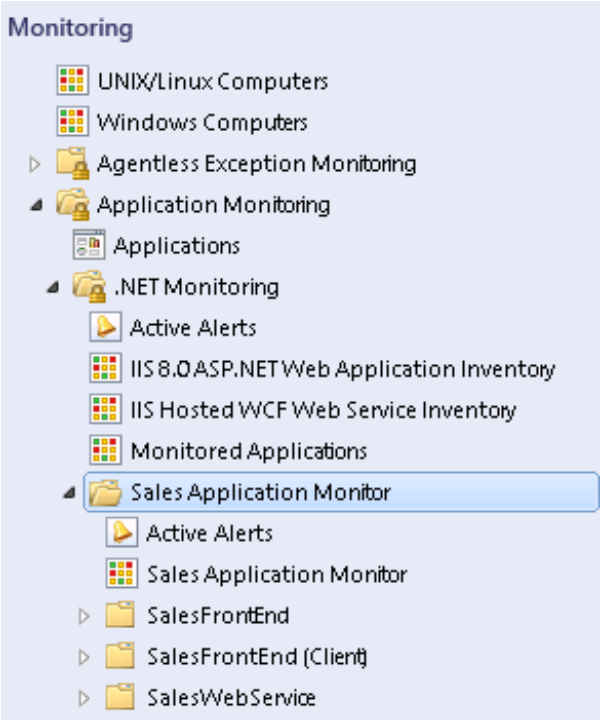
2. Refresh the page several times (press the **F5** key)



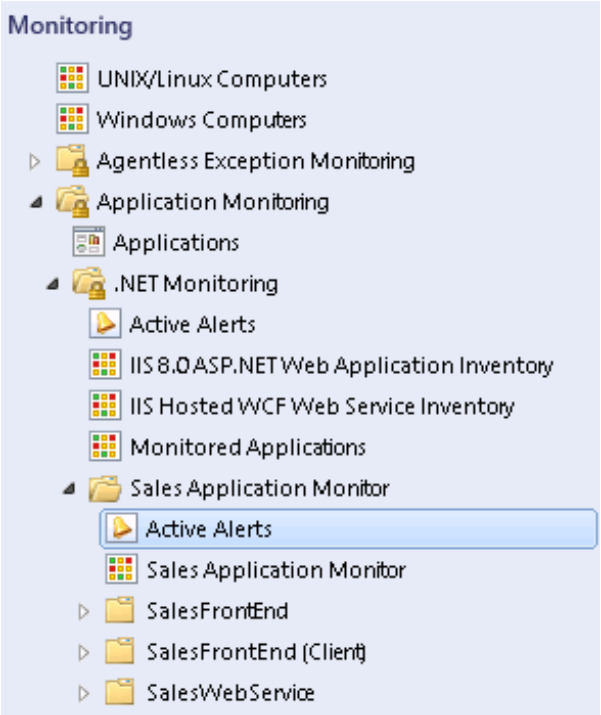
3. In the **Operations Manager** console, click on **Monitoring**



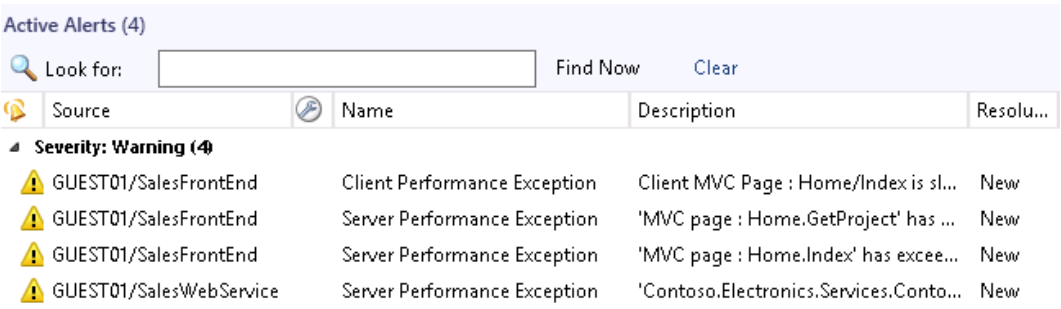
4. Expand **Application Monitoring**, **.Net Monitoring and Sales Application Monitor**



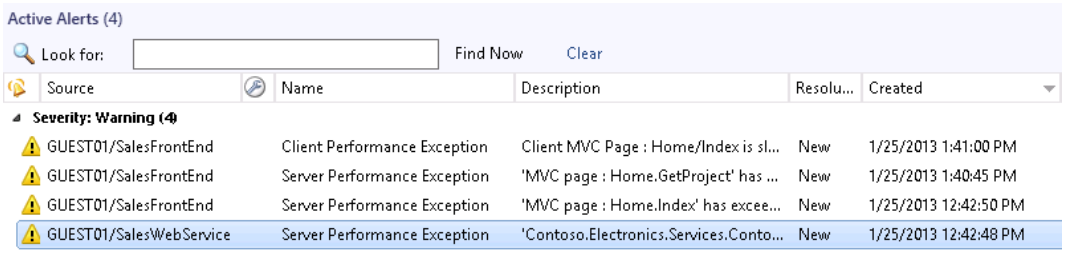
5. Click on **Active Alerts**



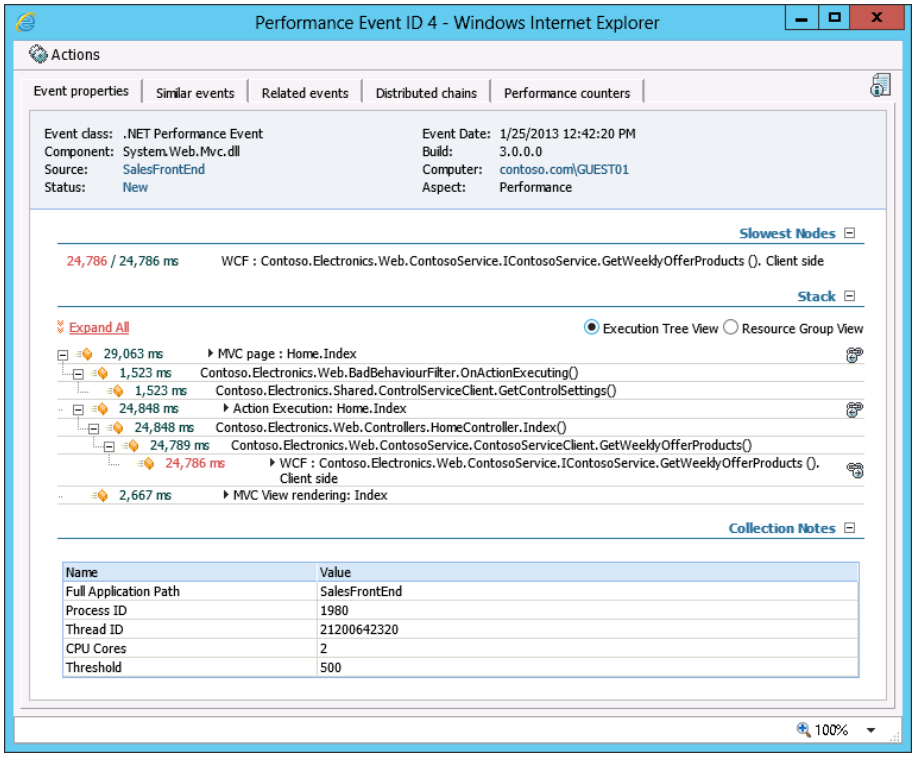
6. There should be several alerts available in the console



7. Click on one that is labelled:
   - Name - Server Performance Exception
   - Description - Contoso.Electronics.Web.ContosoService.IContosoService.GetWeeklyOfferProducts



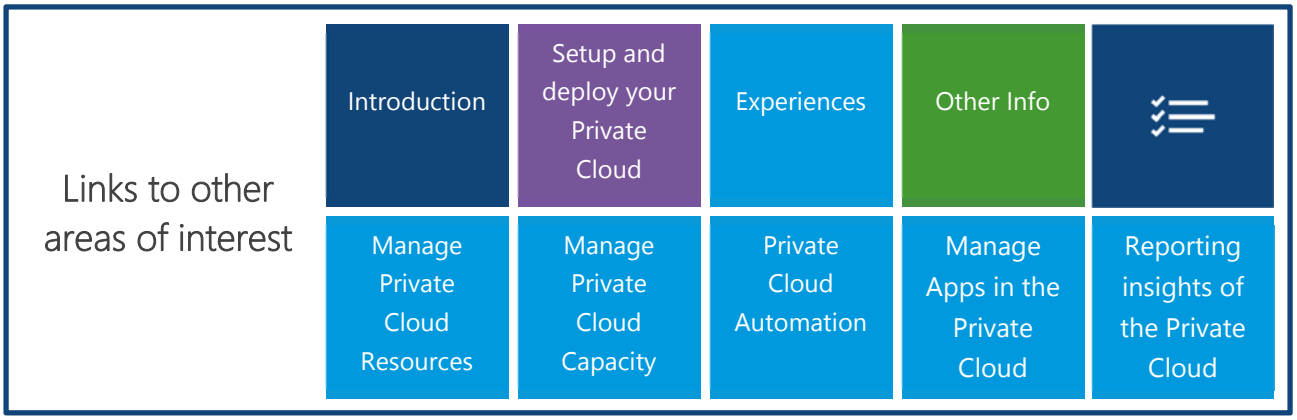
8. In the **Alert Details** pane below, in the **Alert Description** click on the URL beginning with **http://opsmgr/AppDiagnostics. IE ESC** may have to be disabled to view the site correctly.



With this information it is easy for Jeff to start dissecting any alerts with the web application and discover the component or components that are causing an issue. Using the alert above as an example, the WCF web service took 24 seconds to respond to the web request that came through from the front end web site. Jeff can pass that information along with all the associated data to the developers to allow them to start looking at the code.

Important Note

Another feature of Operations Manager is the ability to integrate with (TFS). This feature allows Operations Manager users to assign incoming alerts to Engineering. Operations Manager then transfers the data into TFS Projects ready for developers to investigate. Intellitrace logs can also be gathered and transferred in this fashion to allow for in-depth logging and investigation.



Experience 5: Reporting insights of the Private Cloud

This experience walks organizations through setting up reporting to delivery virtually no downtime with better business continuity and higher SLA. IT is able to provide deep insight into infrastructure performance and utilization via Chargeback which drives lower operational costs. Operations Manager, Virtual Machine Manager and Service Manager will be used in the scenarios in this experience.

Scenario: Chargeback reporting for visibility into usage

Connecting System Center 2012 Virtual Machine Manager with System Center 2012 Operations Manager allows you to monitor the health and availability of the virtual machines and hosts that VMM manages. You can also monitor the VMM Management server, the VMM database server, library servers, and VMM Self-Service Portal web servers, and see diagram views of the virtualized environment through the Operations console in Operations Manager. As services are deployed in Virtual Machine Manager, a corresponding Distributed Application will be created in Operations Manager. Also, through the connections between Virtual Machine Manager and Operations Manager, you can view reports on the health and utilization of your private cloud infrastructure.

Jeff wants to make sure that he has those views. Because Virtual Machine Manager will be crucial in his private cloud, he needs to make sure that every component is healthy.

Before he can start connecting Virtual Machine Manager and Operations Manager he needs to verify some prerequisites:

Credentials

- An account which is a member of the Operations Manager Administrators user role
- An account which is a member of the VMM Administrator user role

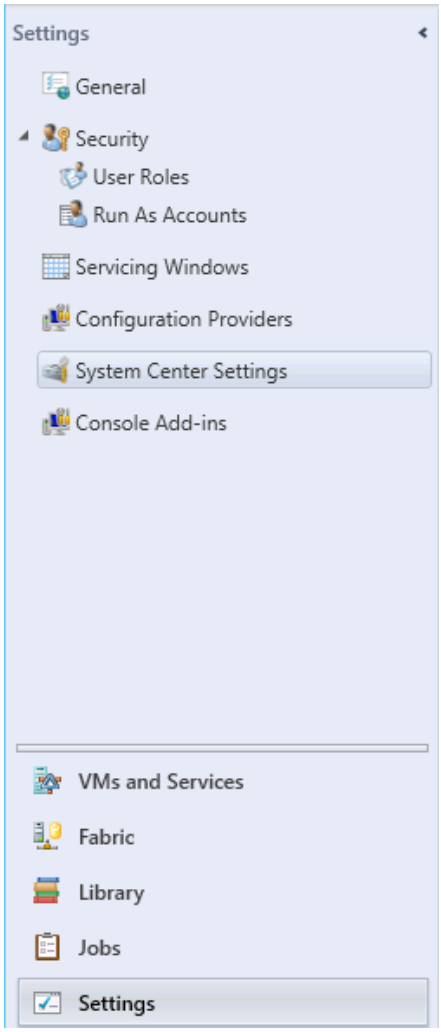
Management Packs

- SQL Server Core Library version 6.0.5000.0 or later (http://www.microsoft.com/download/en/details.aspx?id=10631)
- Windows Server Internet Information Services Library version 6.0.5000.0 or later
- Windows Server Internet Information Services 2003 version 6.0.5000.0 or later
- Windows Server 2008 Internet Information Services 7 version 6.0.6539.0 or later(http://www.microsoft.com/download/en/details.aspx?id=9815)
- Operations Manager console installed on the VMM Server

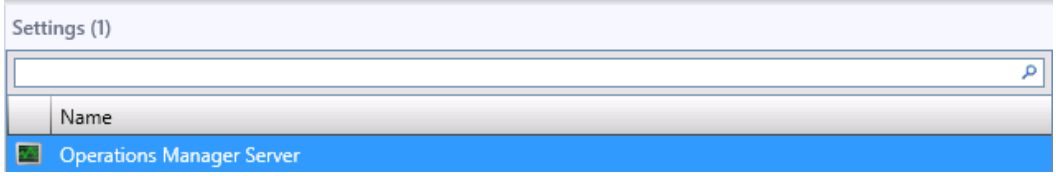
Since this is an evaluation, he can use the **administrator** role for the credentials.

Jeff performs these procedures first and then starts with the connecting procedure.

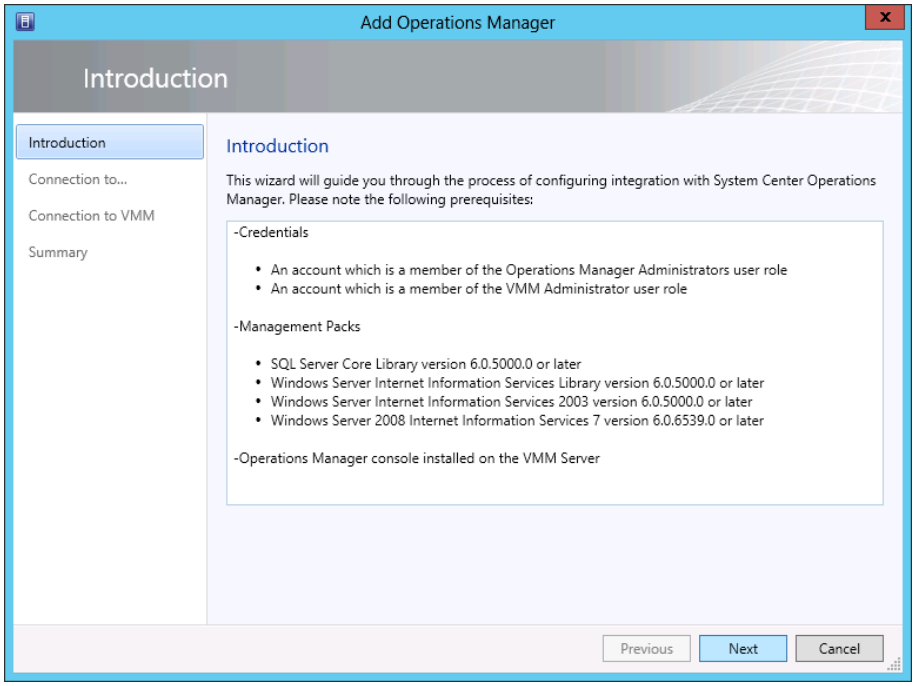
1. From the **VMM** Server open up the **System Center Virtual Machine Manager** console and click on the **Settings** workspace. Jeff then clicks on **System Center Settings**



2. In the **Settings** pane, double click on **Operations Manager Server**

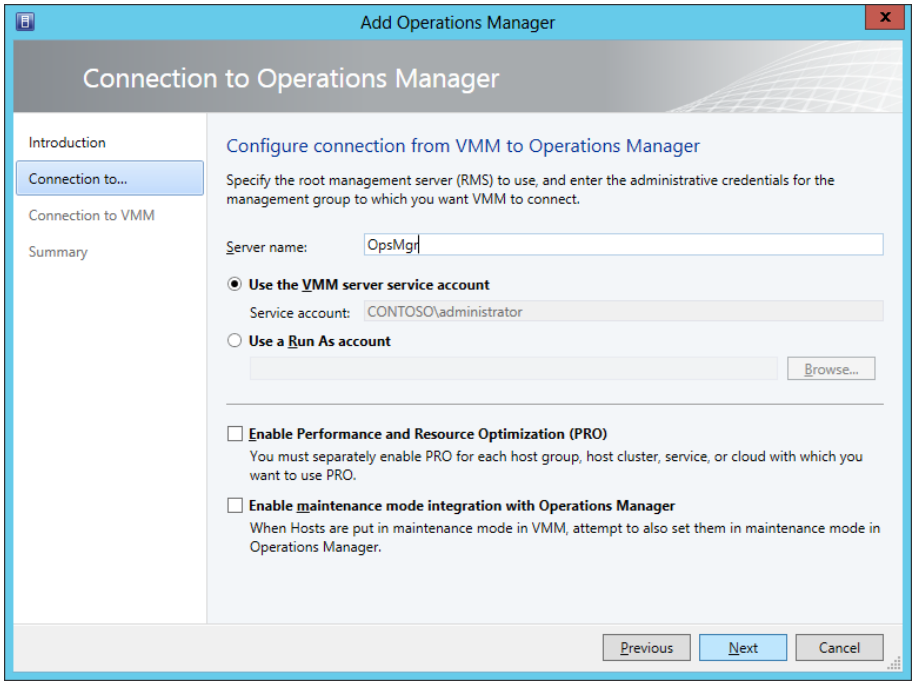


3. From the **Introduction** page, select **Next**

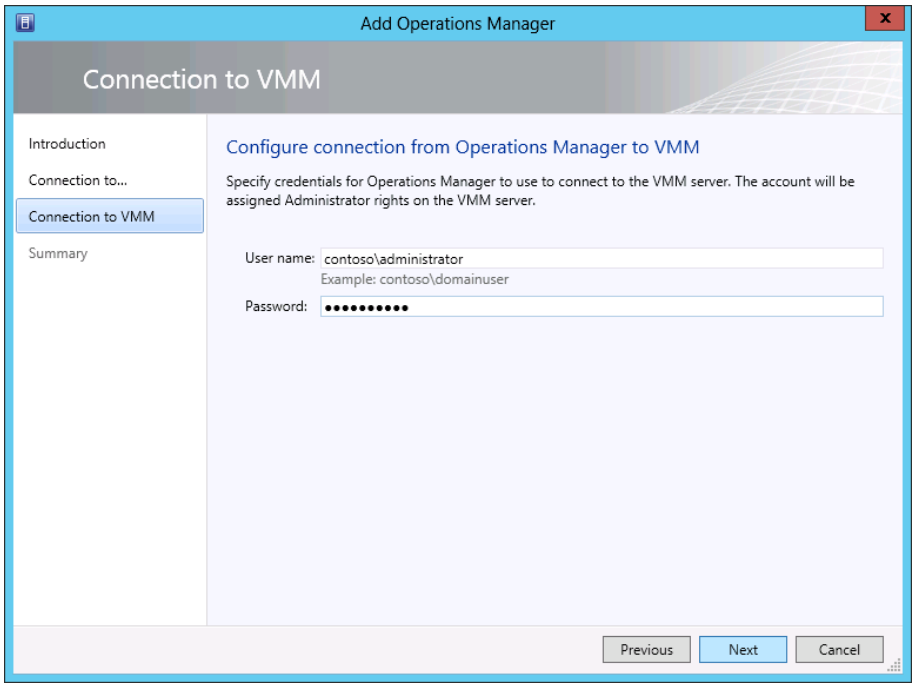


4. On the **Connection to Operations Manager** page, enter the following information and then select **Next**
   - **Server name:** OpsMgr
   - **Use the VMM server service account:** Enabled
   - **Enable Performance and Resource Optimization (PRO):** unchecked
   - **Enable maintenance mode integration with Operations Manager:** unchecked

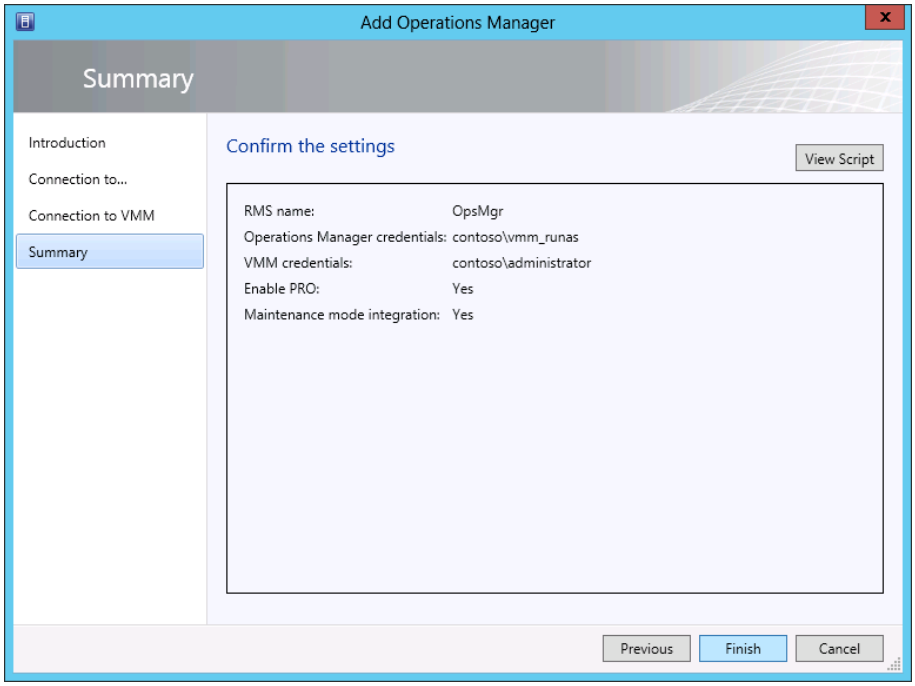
   *Note: If the **VMM server service account** is **LocalSystem**, then tick the **Use a Run As account** instead and select **VMM_RunAs_Contoso***



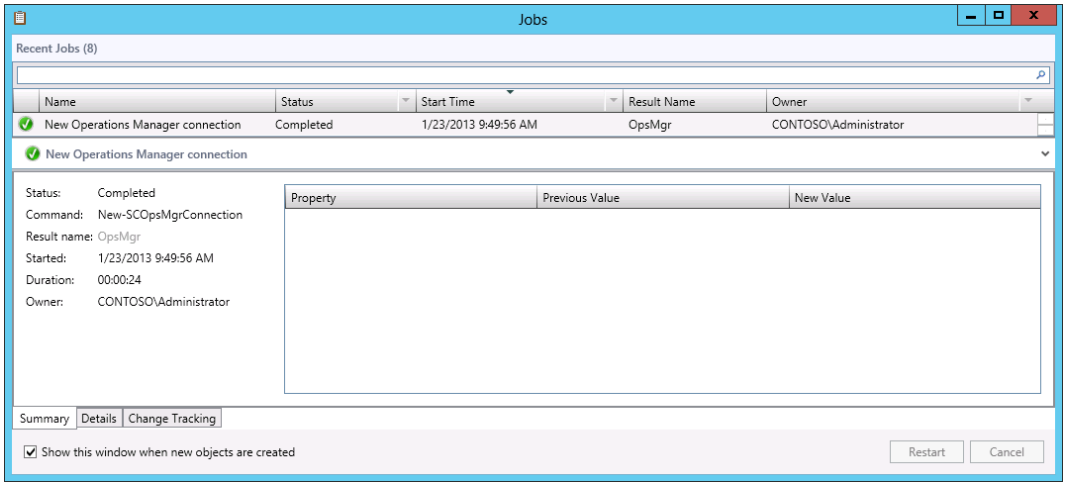
5. On the **Connection to VMM** page, fill in the following data and select **Next**
   - **User name:** contoso\administrator
   - **Password:** ******



6. On the **Summary** page, review your settings and select **Finish**



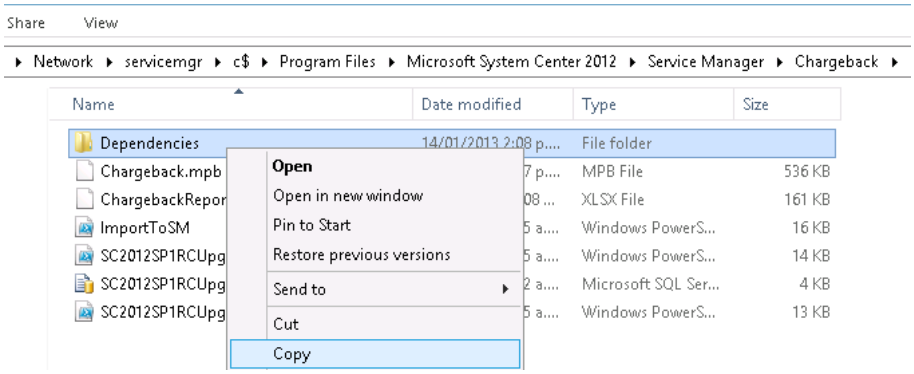
7. A job window will open where you can see if the job succeeds or fails



Install the chargeback report files on the Operations Manager management server

1. Log on to the **Operations Manager** management server as **contoso\Administrator**

2. In the Chargeback folder (located on the ServiceMgr server \\servicemgr\c$\program files\Microsoft System Center 2012\Service Manager\Chargeback\ ) **copy** the subfolder named **Dependencies** from the Service Manager Management server to the Operations Manager management server. In this example **Dependencies** has been copied to the desktop.



3. On the Operations Manager management server, start Windows PowerShell and then navigate to the **Dependencies** folder. Using the same example as above, the following command will work:

```
cd c:\users\administrator.contoso\desktop\dependencies
```

```
Windows PowerShell
Copyright (C) 2012 Microsoft Corporation. All rights reserved.

PS C:\Users\administrator.CONTOSO> cd c:\users\administrator.contoso\desktop\dependencies
```

4. If you have not already set execution policy to **remotesigned**, then type the following command, and then press ENTER:

```
Set-ExecutionPolicy –force RemoteSigned
```

```
Windows PowerShell
Copyright (C) 2012 Microsoft Corporation. All rights reserved.

PS C:\Users\administrator.CONTOSO> Set-ExecutionPolicy -force RemoteSigned
```

5. Type the following command, and then press ENTER to run the PowerShell script that imports chargeback management packs and that add chargeback functionality to Operations Manager:

```
.\ImportToOM.ps1
```

```
PS C:\Users\administrator.CONTOSO> .\ImportToOM.ps1
```

6. After the script has completed running, type **exit**, and then press ENTER to close the **Administrator: Windows PowerShell** window.

Install the chargeback reports on the Service Manager Management server

1. Log on to the **Service Manager** Management server as **contoso\Administrator**.

2. If you have not already set execution policy to **remotesigned**, then start Windows Power shell as an administrator, type the following command, and then press ENTER:

```
Set-ExecutionPolicy –force RemoteSigned
```

```
Windows PowerShell
Copyright (C) 2012 Microsoft Corporation. All rights reserved.

PS C:\Users\administrator.CONTOSO> Set-ExecutionPolicy -force RemoteSigned
```

3. Type **exit** and then press ENTER to close the **Administrator: Windows PowerShell** window.

```
PS C:\Users\Administrator.CONTOSO> exit
```

4. Still from the **ServiceMGR** server**,** click **Start**, type in **Service Manager Shell** and then click on the **Service Manager Shell** icon to open the program.

5. Navigate to the **Chargeback** folder by typing in the command

```
cd chargeback
```

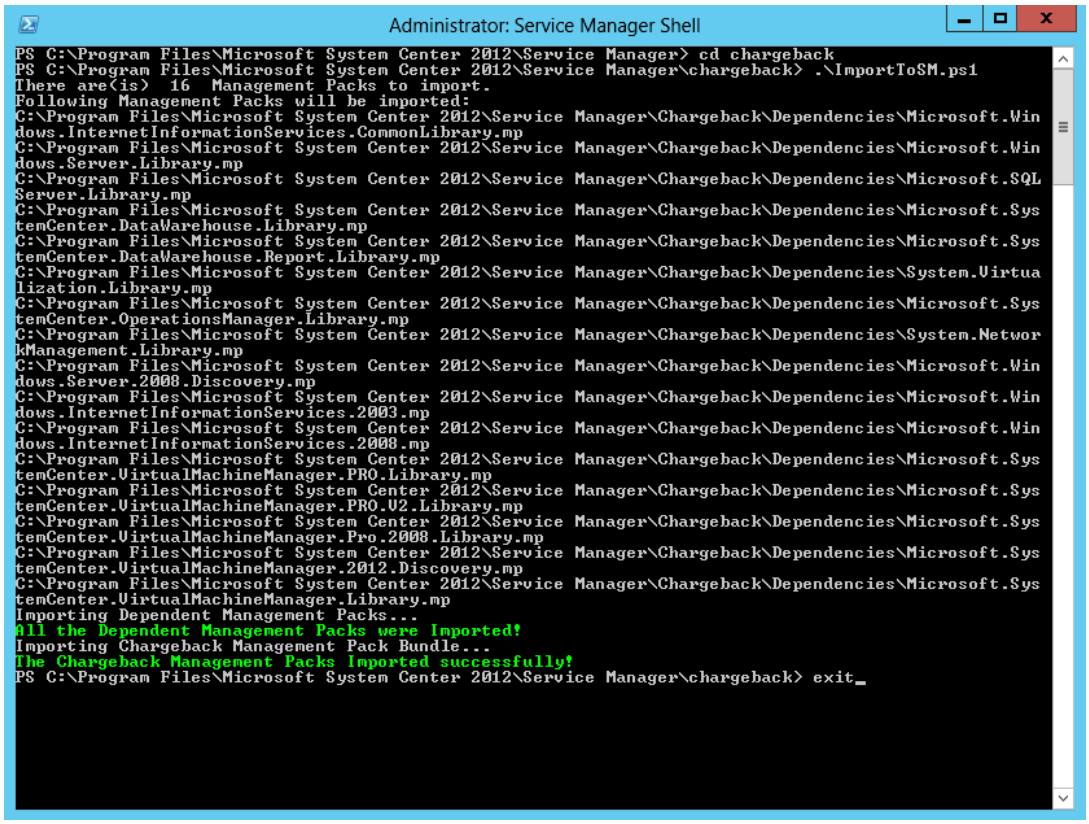
```
PS C:\Program Files\Microsoft System Center 2012\Service Manager> cd chargeback
```

6. Type the following command, and then press ENTER to run the PowerShell script that imports chargeback management packs and that add chargeback functionality to Service Manager:

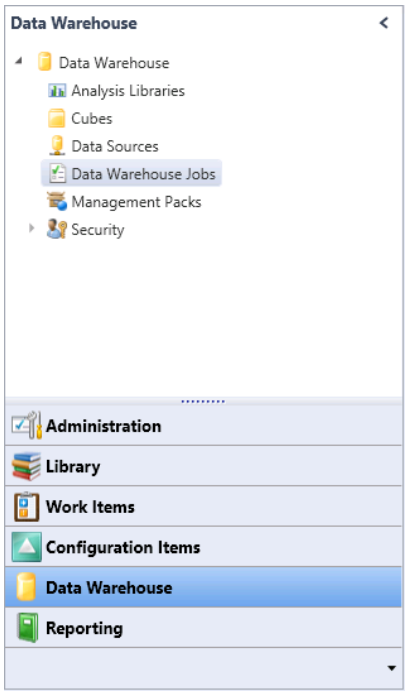
```
.\ImportToSM.ps1
```

```
PS C:\Program Files\Microsoft System Center 2012\Service Manager\chargeback> .\ImportToSM.ps1
```

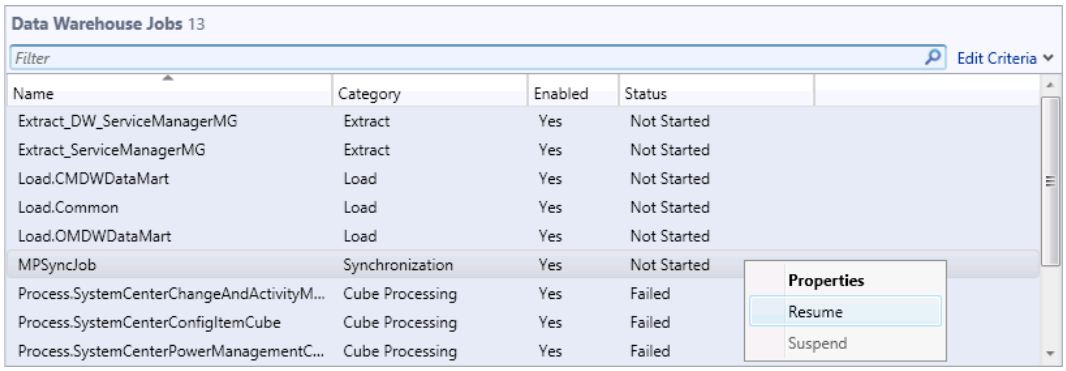
After the script has completed running, type **exit**, and then press ENTER to close the **Administrator: Windows PowerShell** window.



7. In the **Service Manager** console, click **Data Warehouse**, expand the **Data Warehouse** node, and then click on **Data Warehouse Jobs**.



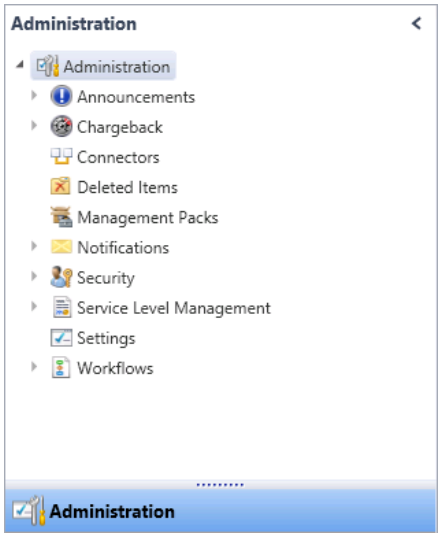
8. In the list of data warehouse jobs, select **MPSyncJob** and then from the Task list, click **Resume**.



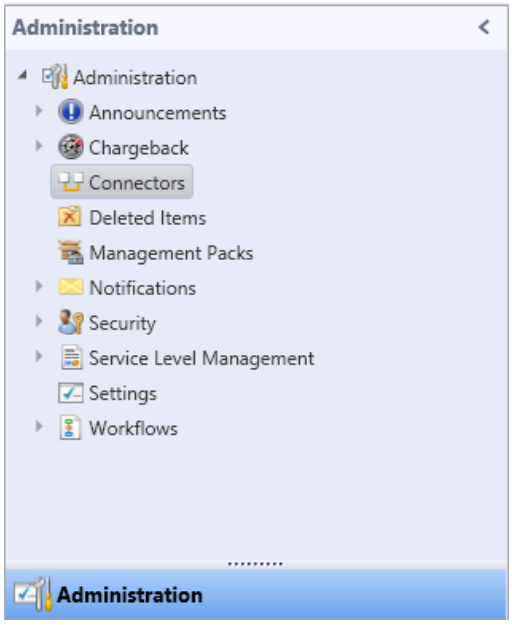
The next step is to configure the **Operations Manager CI connecter** and ensure that **Service Manager** has discovered the virtual machine information from **Operations Manager**

Create the Operations Manager CI connector

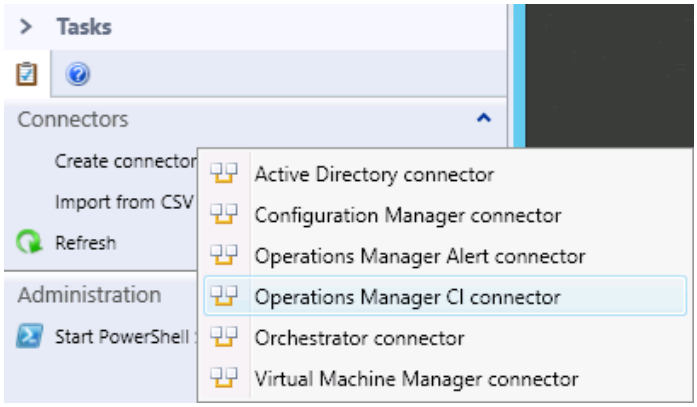
1. From the **ServiceMgr** server open the **Service Manager** console and click on the **Administration** workspace.



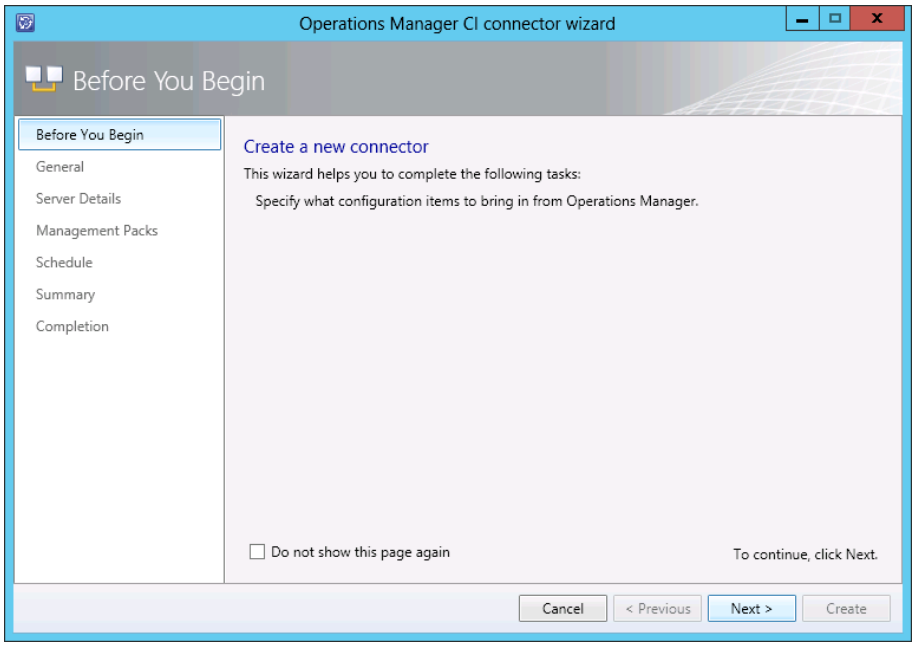
2. From the Administration workspace, expand **Administration**, and then click **Connectors**.



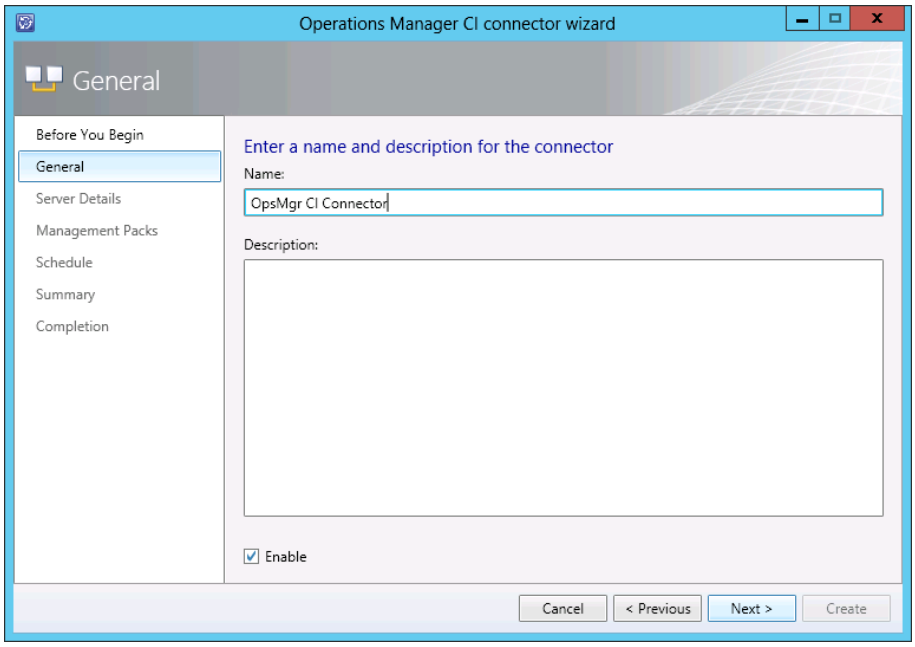
3. In the Tasks pane, under **Connectors**, click **Create Connector**, and then select **Operations Manager CI Connector**.



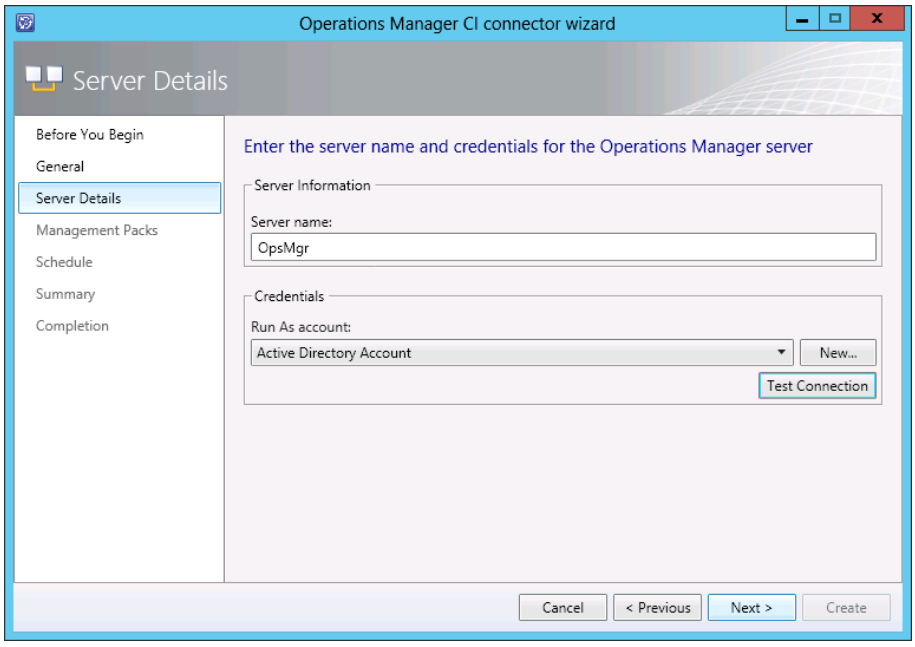
4. Click **Next** on the **Before you Begin** page.



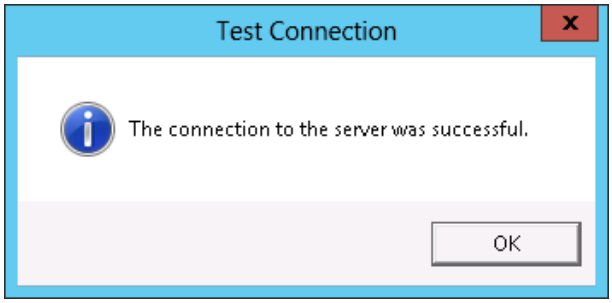
5. On the **General** page, in the **Name** box, type **OpsMgr CI Connector** for the new connector. Make sure that the **Enable** check box is selected, and then click **Next.**



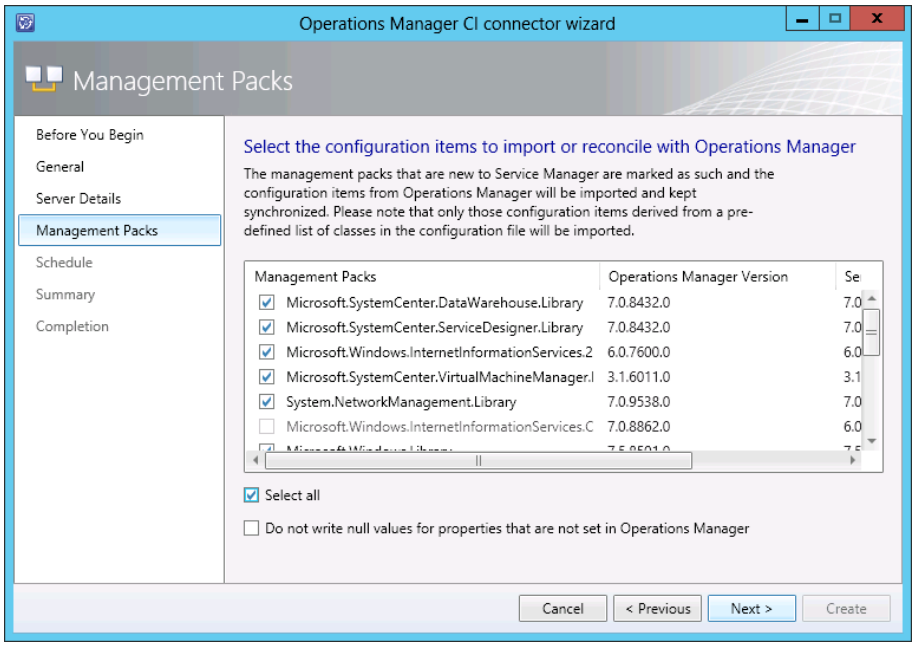
6. On the **Server Details** page, in the **Server name** box, type **OpsMgr** leave the **RunAs** account as **Active Directory Account**. Use the **contoso\administrator** account.



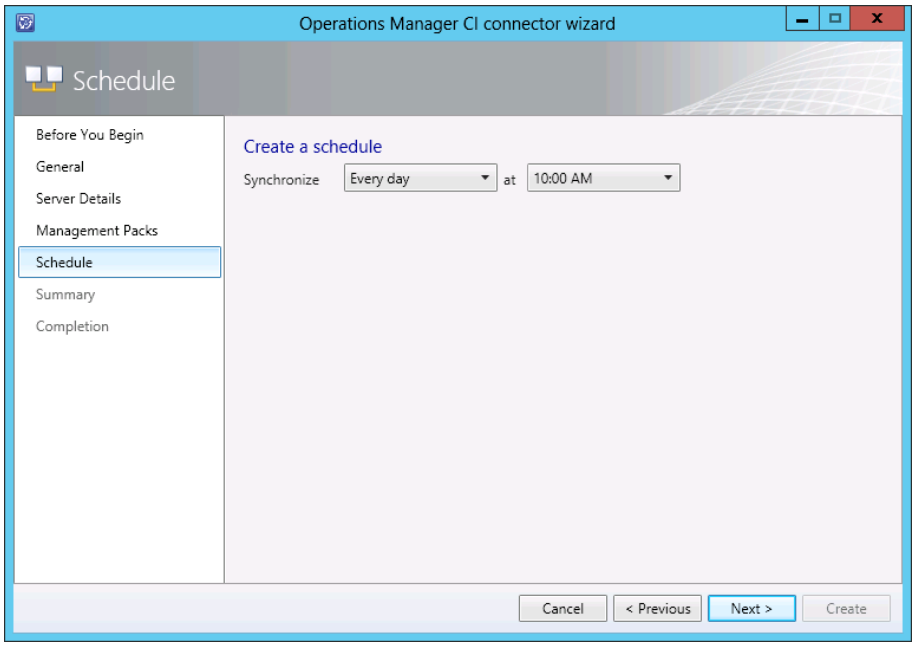
7. **Test Connection** and confirm that the connection confirms correctly, click on **OK** and then click on **Next**



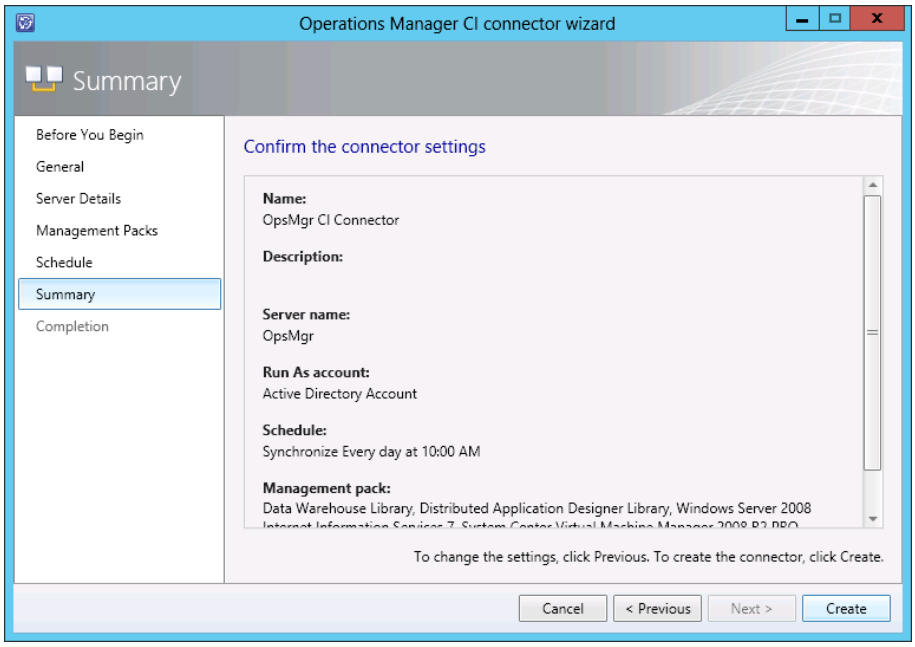
8. On the **MP Selection** page, click **Select all**, and then click **Next**



9. On the **Schedule** page, click **Next**, and then click **Next** again.



10. Verify your connector settings are as per the instructions and then click **Create**.



The import process can take up to an hour to complete; you can check the status of the import by opening the **Service Manager console** and checking the **status** of the **MPSync** job**.** Once the import has completed, close the

Service Manager console and then reopen it. Under the **Administration** tab, expand **Administration**, and confirm that **Chargeback** is now listed. Once it is there you can proceed to create your **price sheet**.

> **Important Note**
>
> *The Chargeback facility is reliant on the capturing of data over a period of time. Due to time constraints and the minimal pool of resources in this evaluation guide, Microsoft has provided a data generation tool that will pre-populate the Service Manager database with chargeback information. The data generation tool can be downloaded from here:* **http://go.microsoft.com/?linkid=9825976**

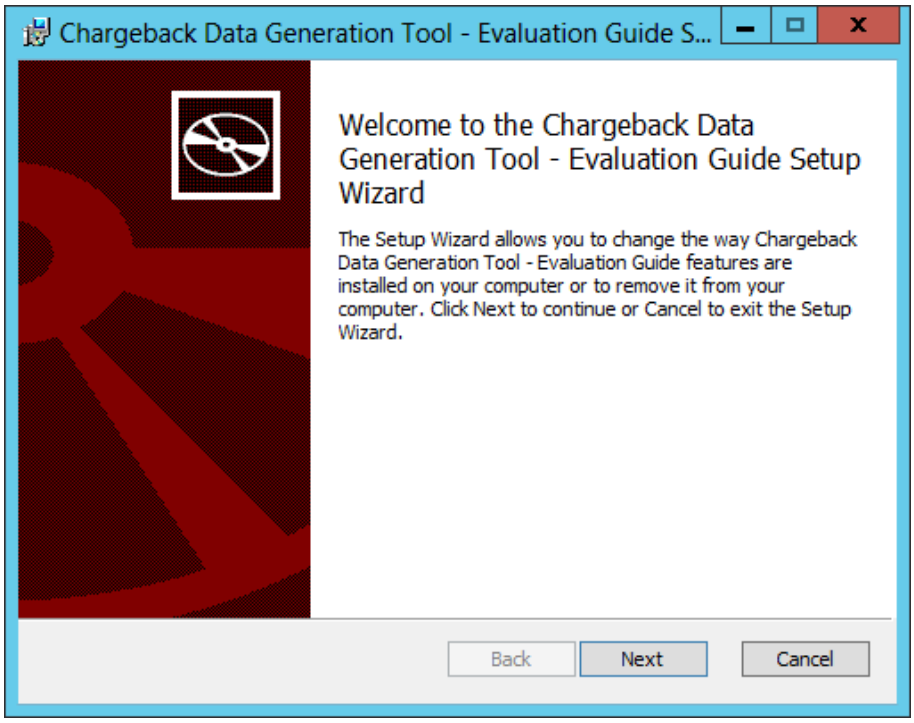
The installation of this tool is not required for the completion of this scenario, although it will help visualize what real-world results may look like.

To install the Chargeback Data Generation Tool follow these steps:

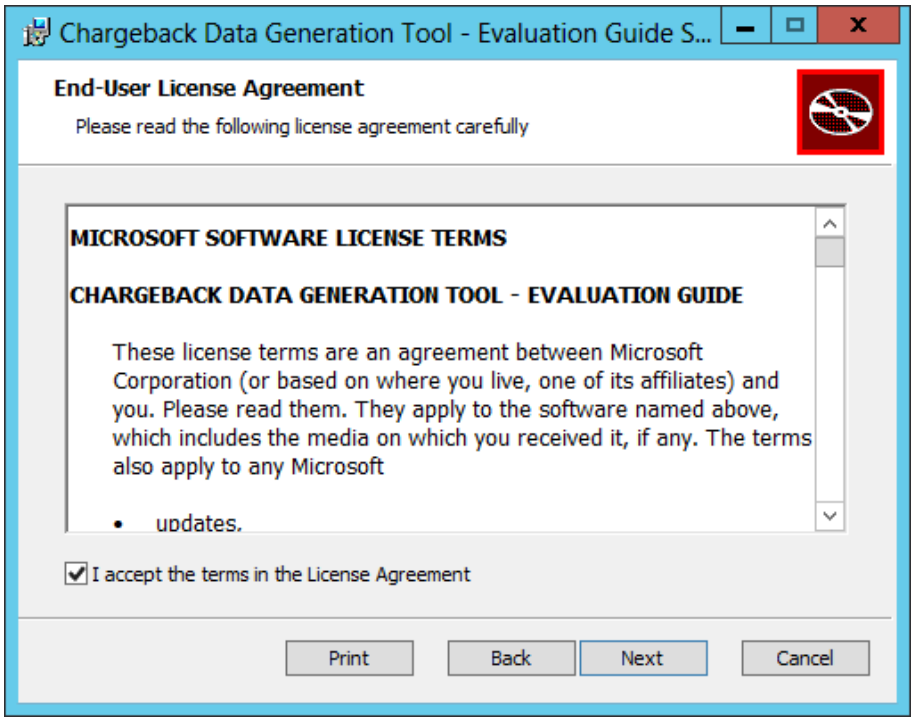
1. Download the data generation tool from **http://go.microsoft.com/?linkid=9825976** then copy the file to the server **ServiceMgr**

2. Extract the file and run the installer **Chargeback Data Generation Tool** on the server **ServiceMgr**

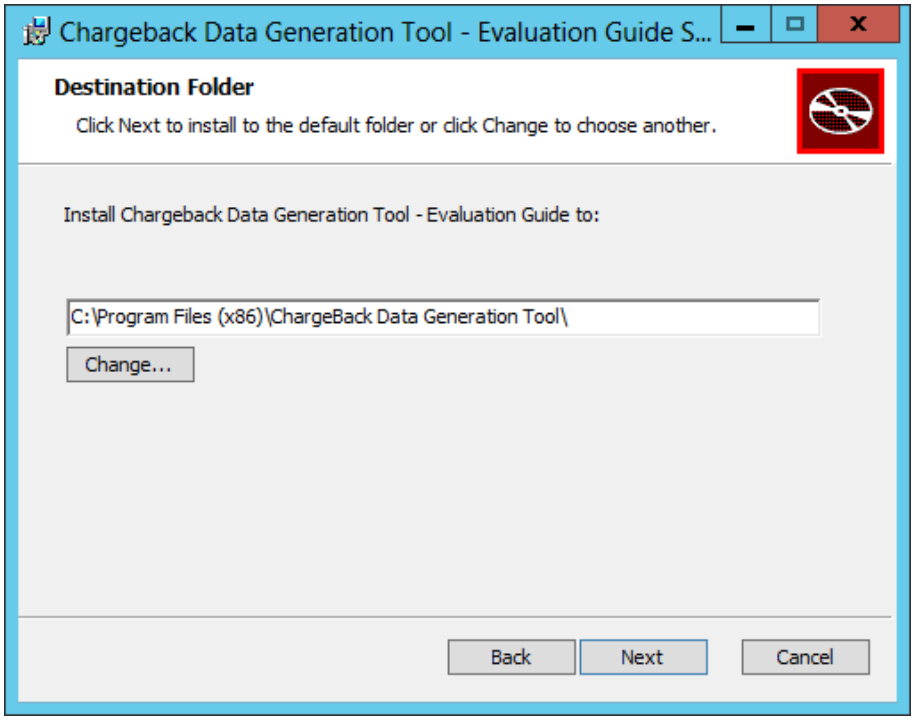
3. Click **Next**



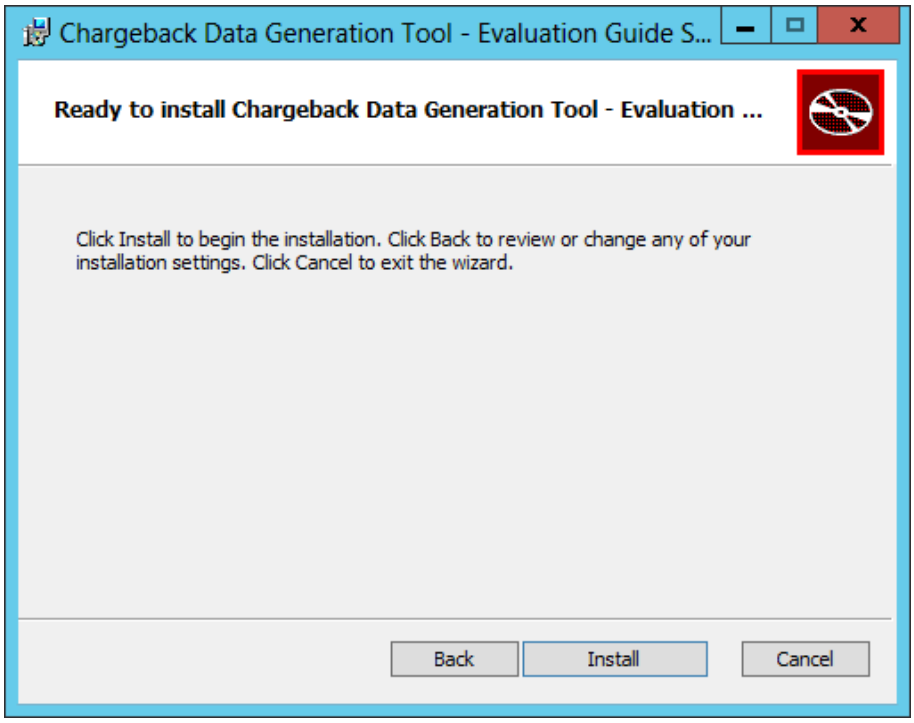
4. Check **I accept the terms in the License Agreement** then click **Next**



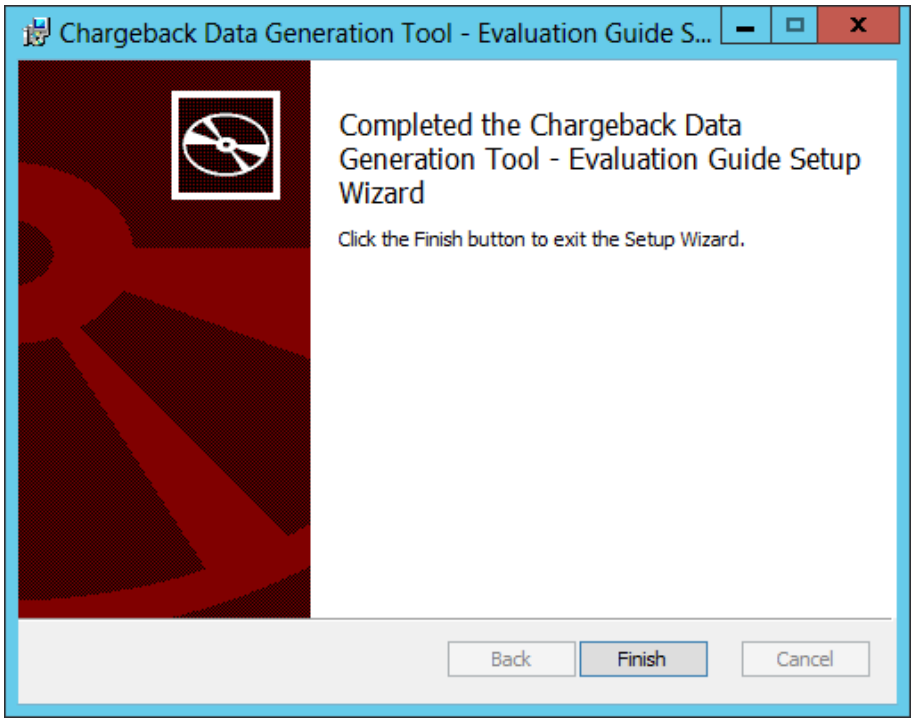
5. Confirm the install directory, then click **Next**



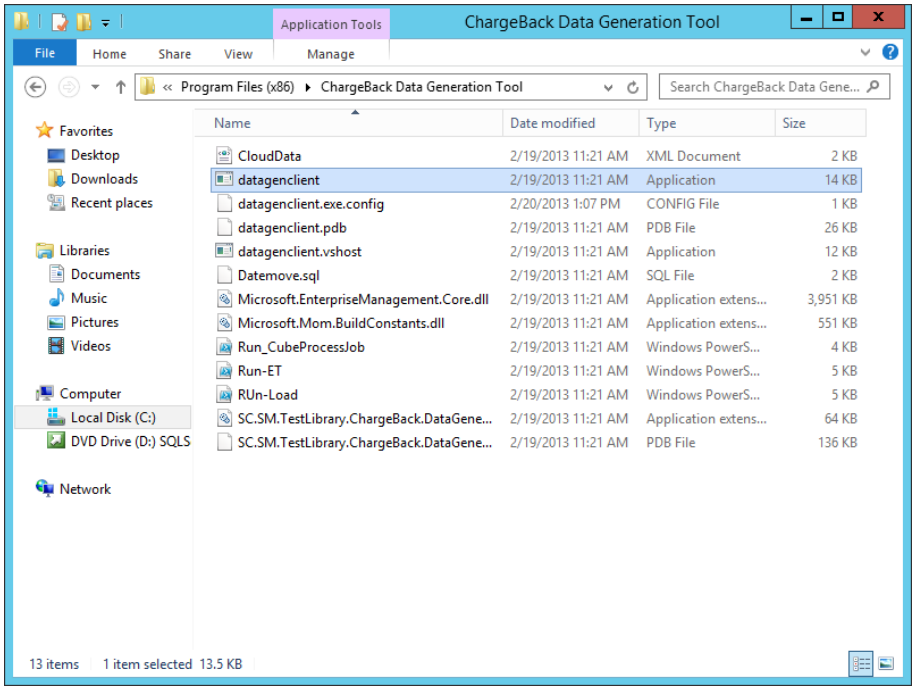
6. Click **Install**



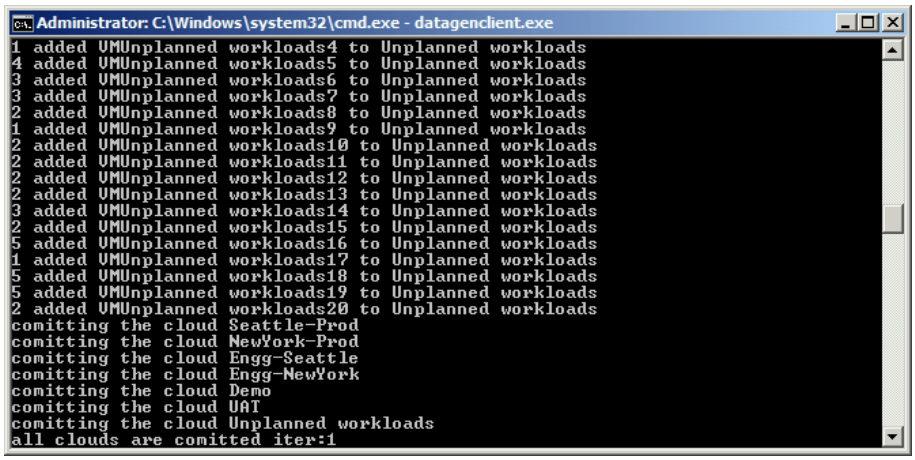
7. Click **Install**



8. Browse to the folder that you installed the tool to, in this example it is **C:\Program Files(x86)\ChargeBack Data Generation Tool** then run the file **datagenclient.exe**



9. Once the data generation has completed, close the CMD prompt.



This completes the installation of the data generation tool. When assigning Clouds to the price sheet created later in this scenario, ensure that you select the additional clouds created by this tool.

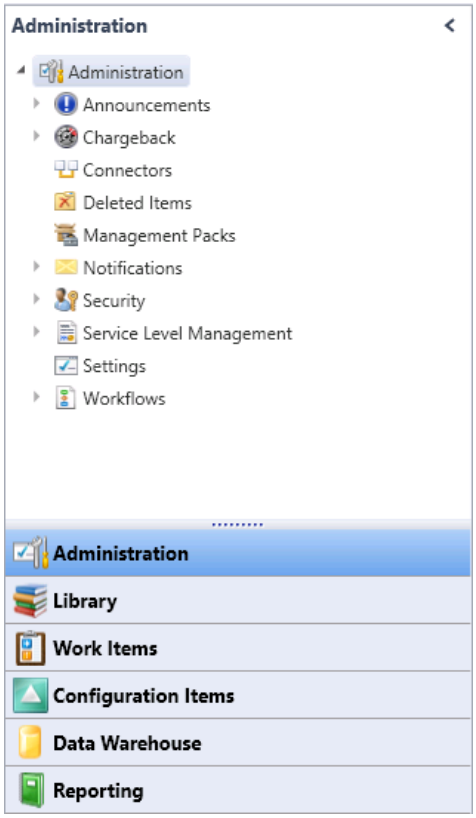
Create a Price Sheet

Price sheets in System Center 2012 – Service Manager allow you to define price policies on existing private cloud objects that are discovered from the Operations Manager CI connector. You can associate one or more private clouds to a price sheet. After association, price information contained in the price sheet is shown in OLAP cubes and in a sample Excel report. You can use the OLAP cube data to create your own customized reports using various reporting solutions such as SQL Server Reporting Services, SQL Server Analysis Services tools, and Microsoft Excel. Additionally, you can modify the sample Excel report for your organization's needs.

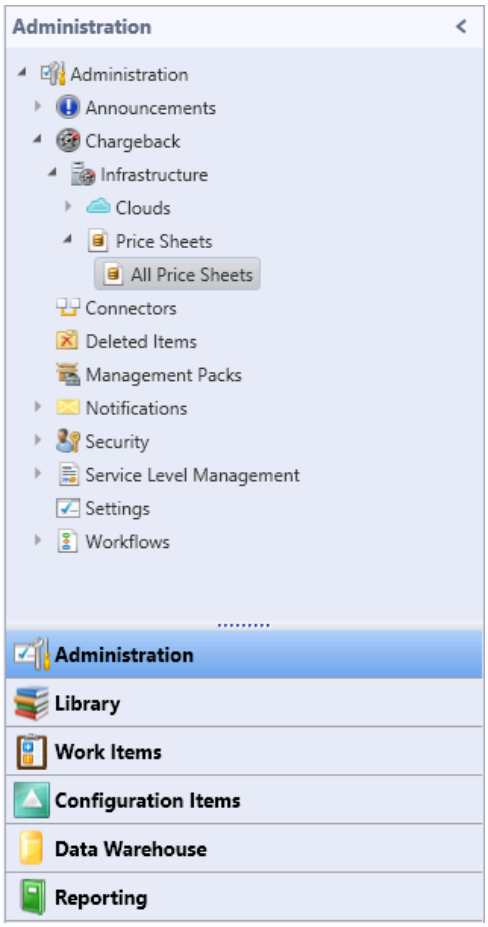
Although you can create multiple price sheets without publishing them, you must publish a price sheet before you can associate cloud objects to the price sheet.

To create a price sheet

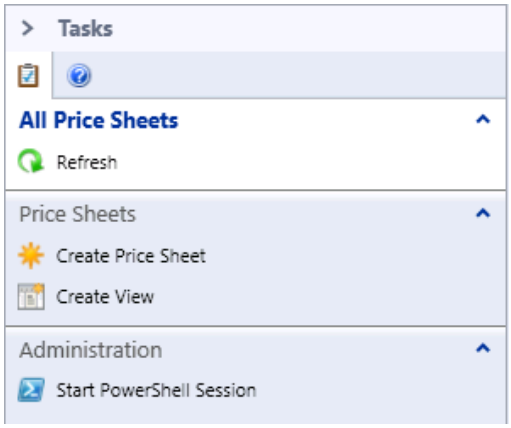
1. In the **Service Manager console**, select **Administration**



2. In the **Administration** pane, expand **Chargeback**, expand **Infrastructure**, expand **Price Sheets** and then select **All Price Sheets**

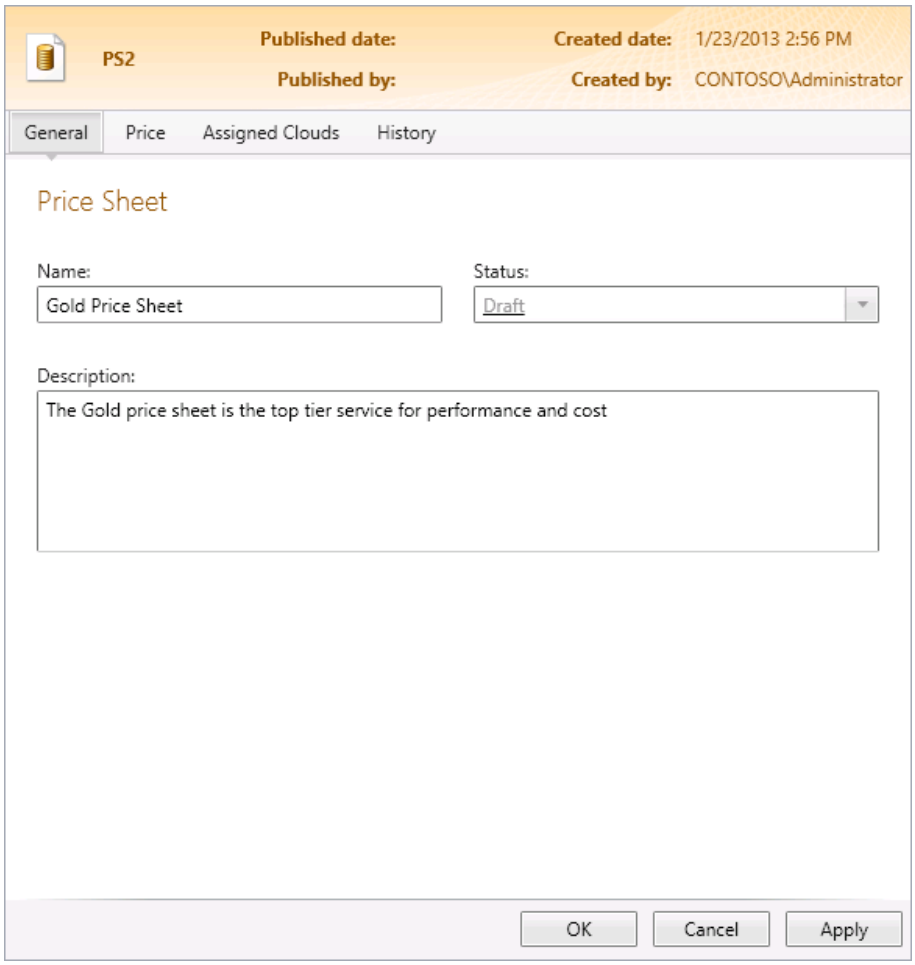


3. In the **Tasks** pane under **All Price Sheets**, click **Create Price Sheet**.



4. On the **General** tab, input the following values:

    - Name: **Gold Price Sheet.**
    - Description: **The Gold price sheet is the top tier service for performance and cost**

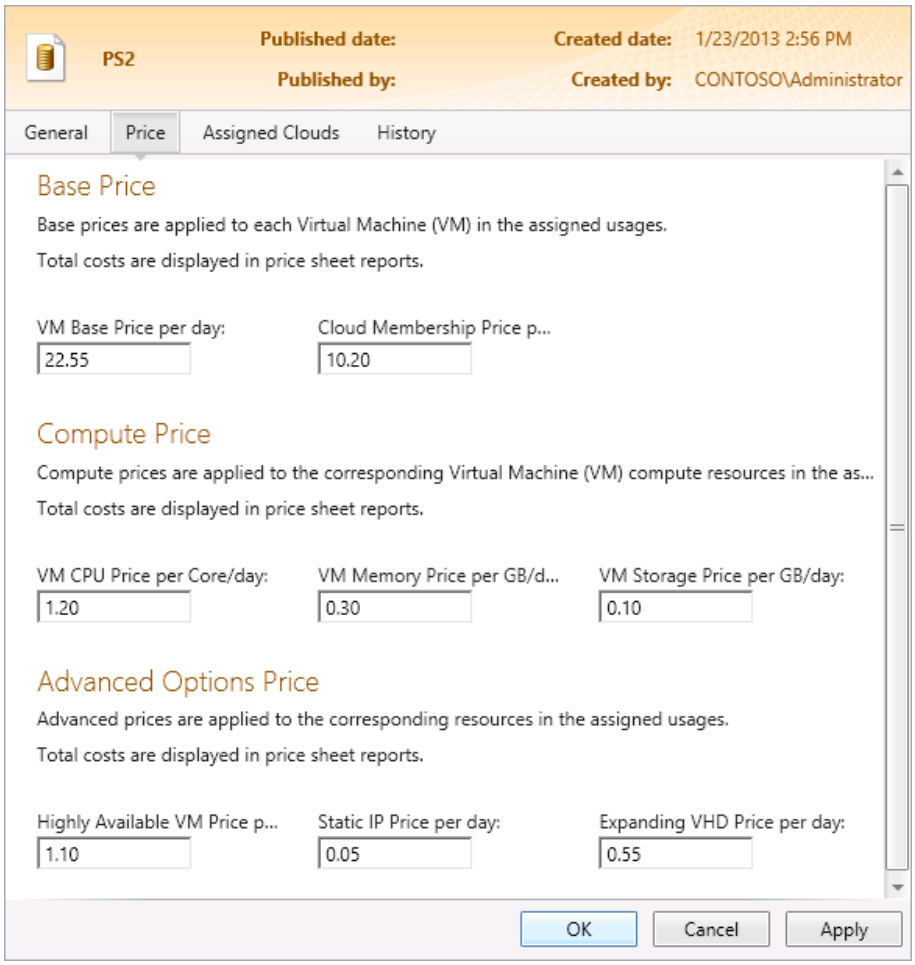


5. Click on the **Price** tab and input the following values:

    - VM Base Price per day: **22.55**
    - Cloud Membership Price per day: **10.20**
    - VM CPU Price per Core/day: **1.20**
    - VM Memory Price per GB/day: **.30**

- VM Storage Price per GB/day: **.10**
- Highly Available VM Price per day: **1.10**
- Static IP Price per day: **.05**
- Expanding VHD Price per day: **.55**

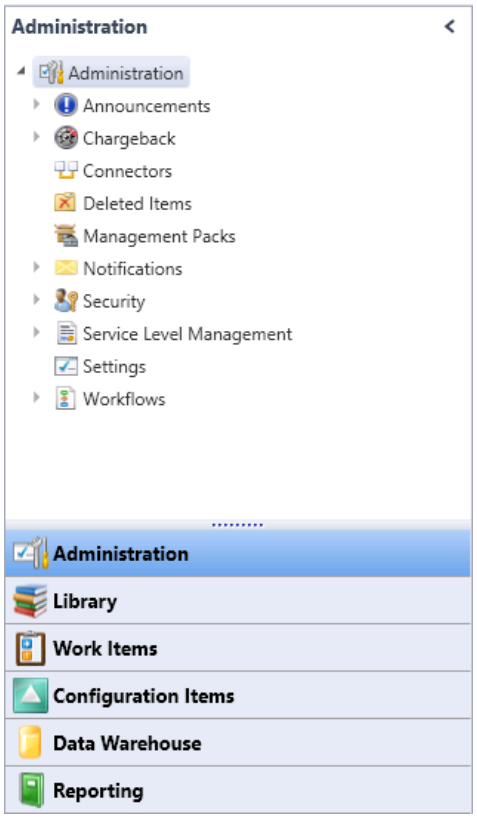
Click on **OK**



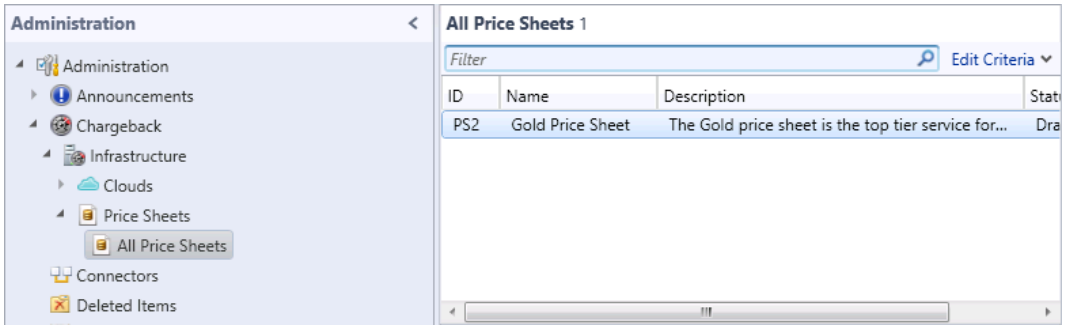
Publish the Gold Price Sheet

Jeff now needs to publish the price sheet he has created to make it available to assign to new or existing Clouds.

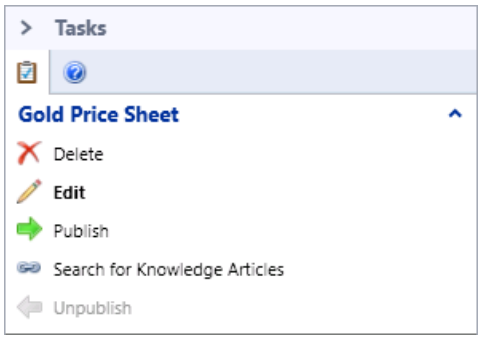
1. In the Service Manager console, select the **Administration** tab



2. In the **Administration** pane, expand **Price Sheets**, **All Price Sheets,** then select **Gold Price Sheet**

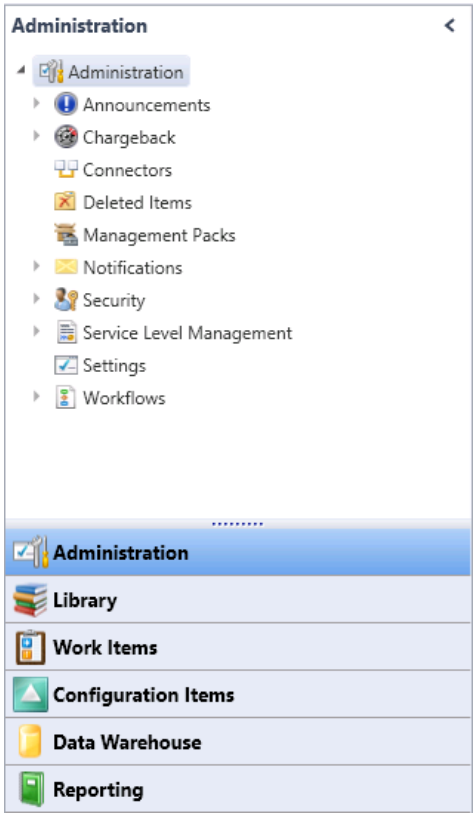


3. In the **Tasks** pane click **Publish**.

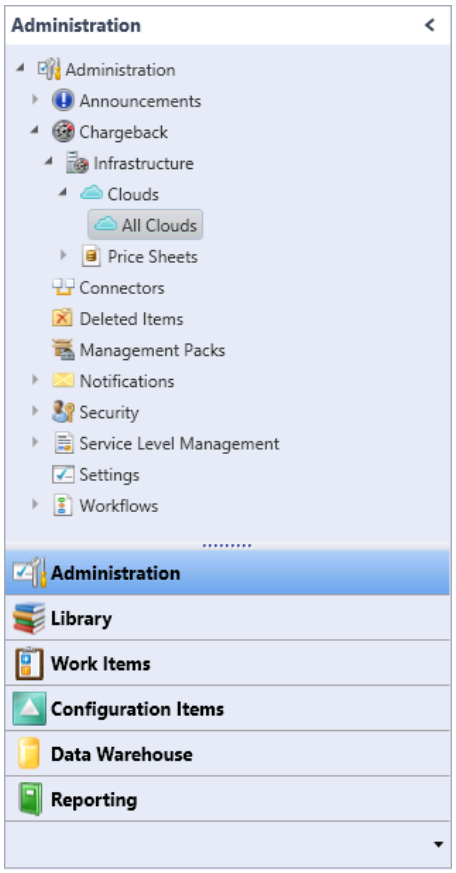


Assign the Gold Price Sheet to MyCloud

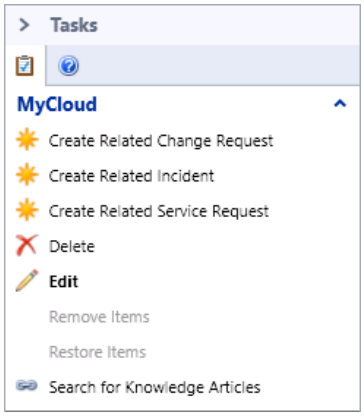
1. In the **Service Manager** console, select **Administration**



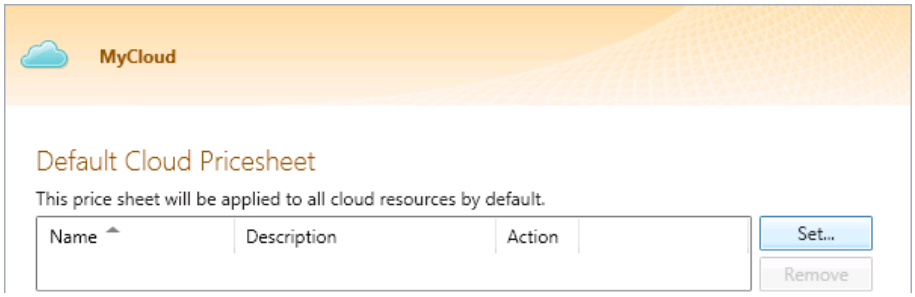
2. In the **Administration** pane, expand **Chargeback**, expand **Clouds**, and then select **All Clouds**



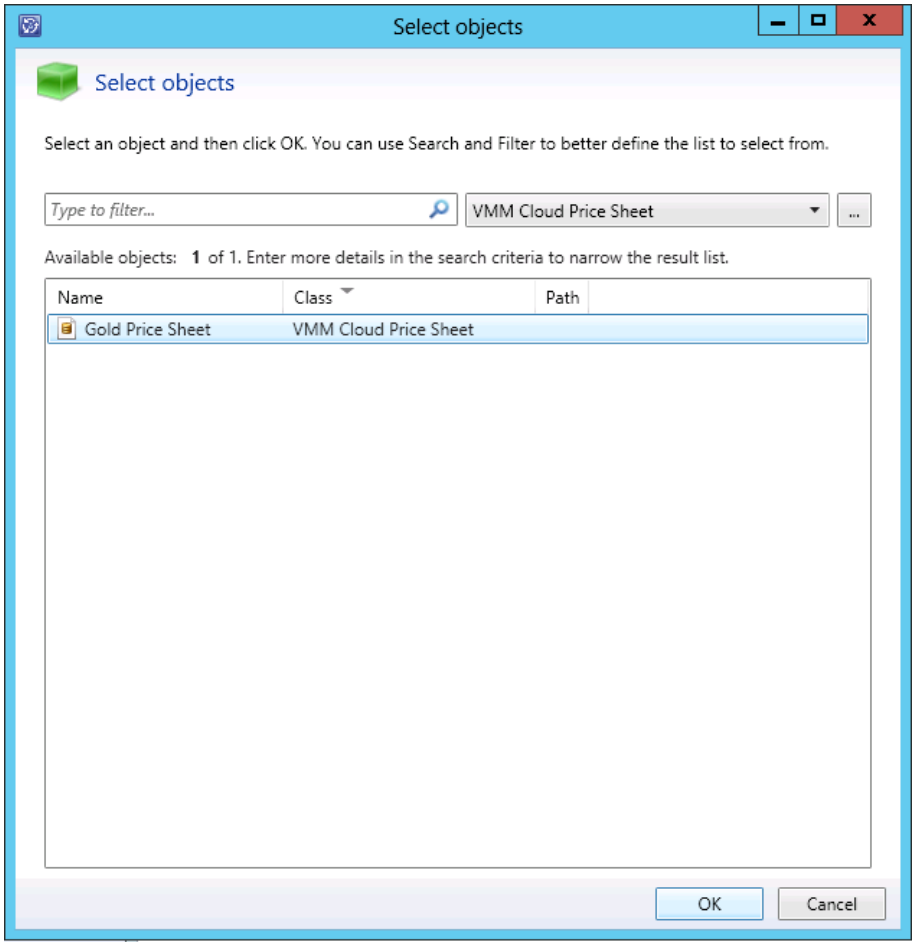
3. In the list of clouds, select "**MyCloud"** and then under Tasks, click **Edit**



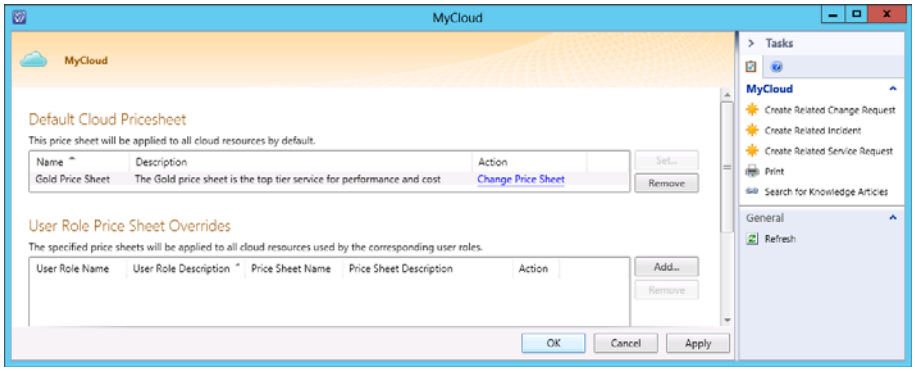
4. In the **MyCloud** form under **Default Cloud Pricesheet**, click **Set**



5. In the **Select objects** dialog box, select the **Gold Price Sheet** and then click **OK**.



6. Click **OK** to close the **MyCloud** form.



Important Note

If you have used the data generation tool to generate additional chargeback data, repeat steps **2-6** to associate the additional clouds to the **Gold Price Sheet**.

Chargeback reporting for visibility into usage

In System Center, chargeback is one of the tools that help you communicate with business units about how they consume capacity. This helps you by utilizing existing investments, proportionate to your customer's requests. System Center components help you manage the following processes:

- Quotas
- Leases
- Approvals
- Chargeback or Showback

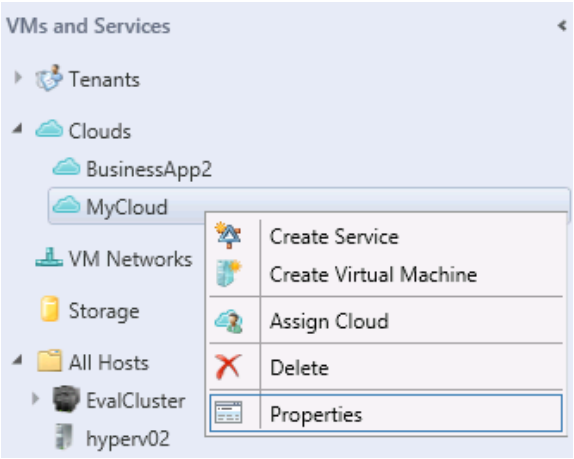
The theme of chargeback is cloud based pricing, where each cloud has its own price, based on SLA. Most often, you'll have many clouds with various SLAs for the clouds for different business units or organizations. Chargeback uses a price sheet, or rate card, for each cloud. This means you can have one price represented in a price sheet that contains various clouds addressing one SLA, and you can have another price sheet for a different SLA.

Prerequisites

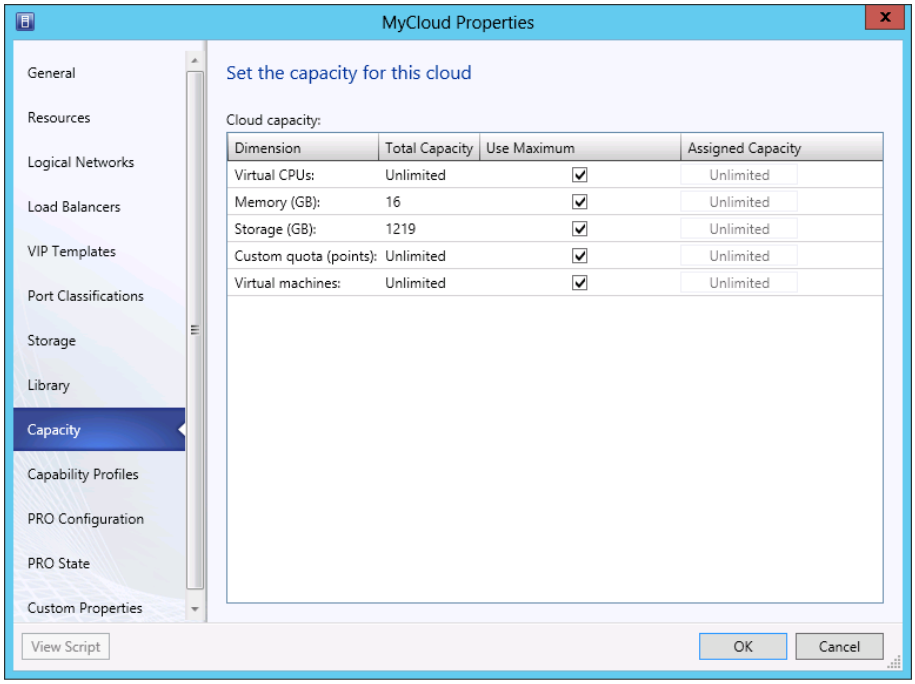
- Office Professional 2013 demo installed is installed on Guest01 as excel is required to view the Chargeback spread sheet report.
- Download the Microsoft Office Professional Plus 2013 demo from TechNet here http://technet.microsoft.com/en-us/evalcenter/jj192782.aspx

1. Firstly Jeff is required to change the properties of the **MyCloud** cloud settings. This is because the virtual machines we are adding to the cloud have a maximum dynamic memory allocation of just over a Terabyte of memory and will conflict with the clouds allowable RAM allocation.

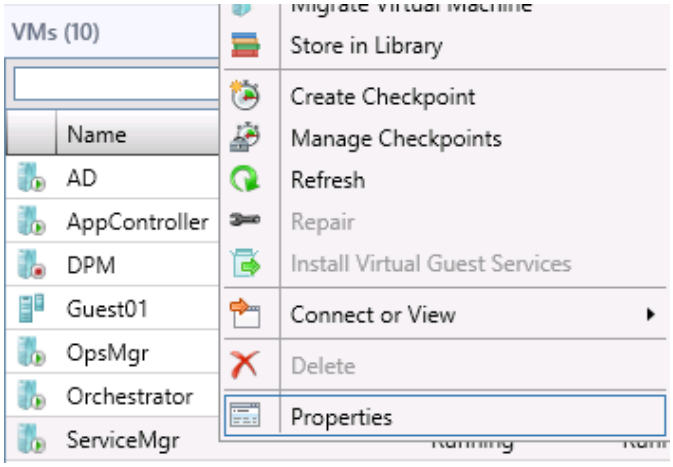
2. Jeff opens up the **Virtual Machine Manager console** on the VMM Server. He then clicks on **VMs and Services**, expands **Clouds**, right clicks on the **MyCloud** cloud and clicks **Properties**.



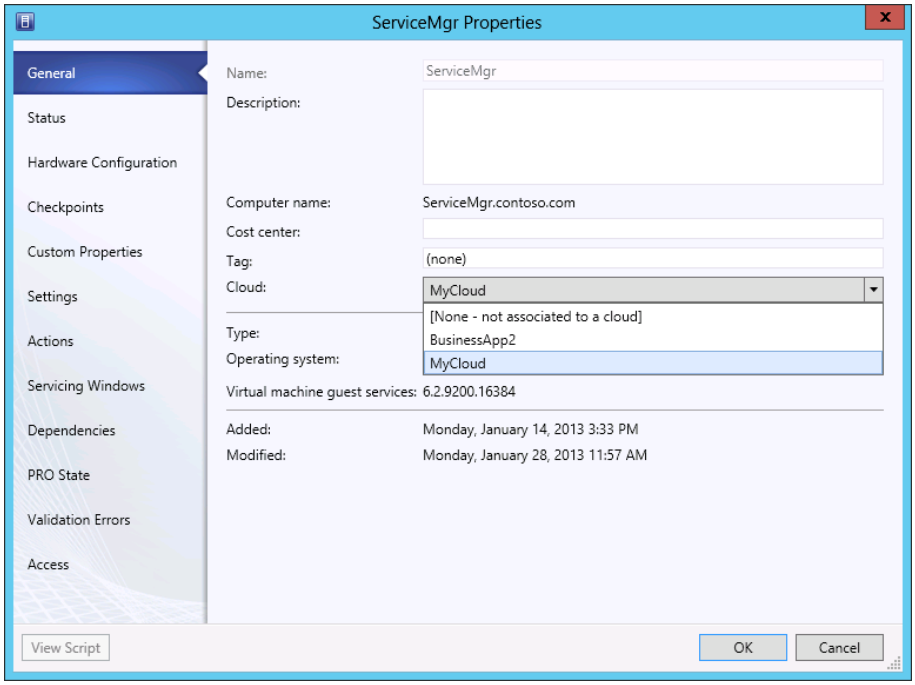
3. When the **MyCloud** properties open, Jeff clicks on the **Capacity** tab and ticks all the **Use Maximum** boxes to allow any Virtual Machine to be added to it and clicks **OK**



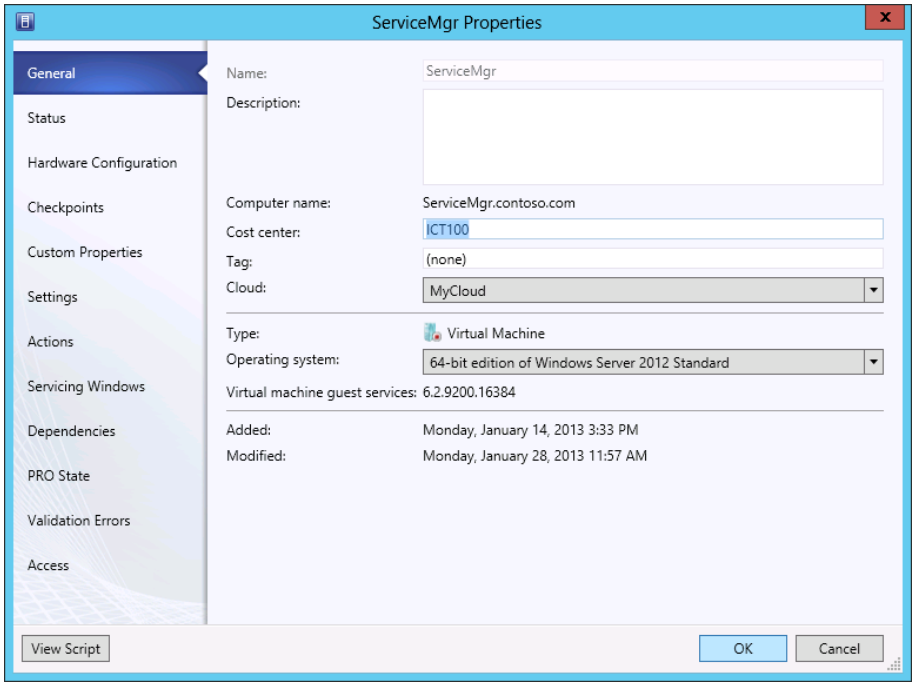
4. From within the **VMM** console, Jeff then navigates to the **VMs and Services** tab and from the available VM list clicks on the **ServiceMgr** server, he then right clicks on it and chooses **Properties**



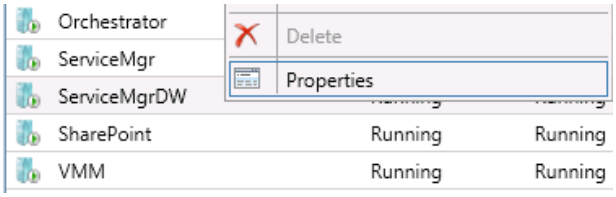
5. In the **General** section of the properties pane Jeff clicks on the drop down box next to **Cloud** and then adds this server to the **MyCloud** cloud.



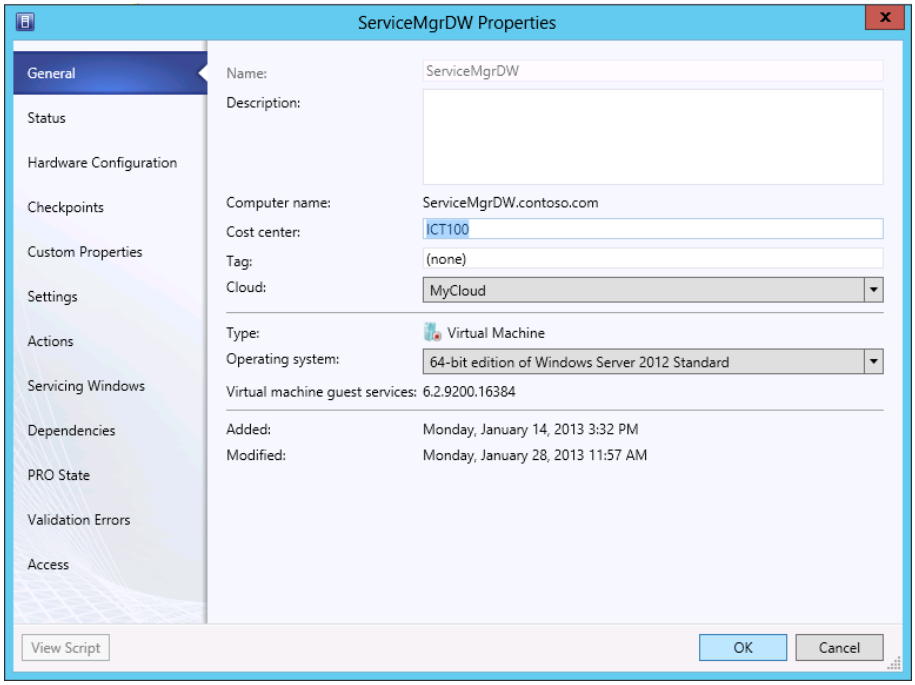
6. He then creates a new cost center for this service offering called **ICT100** and clicks **OK**



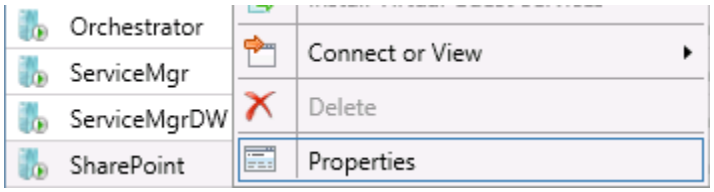
7. Jeff then chooses the **ServiceMgrDW** virtual machine, right clicks on this and then clicks on **Properties**



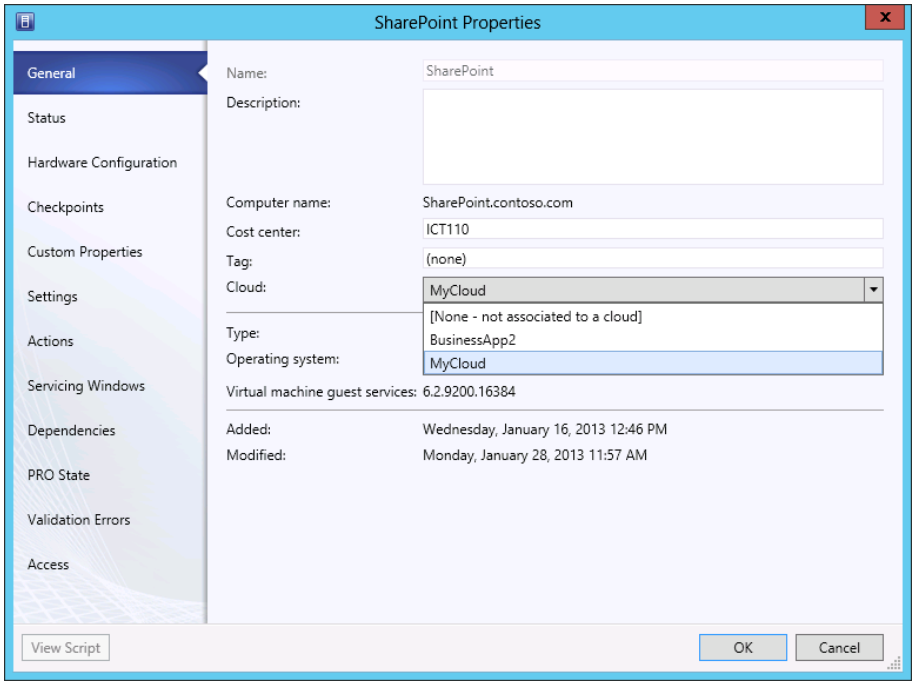
8. In the **General** section click on the drop down next to clouds and add this server to the **MyCloud** cloud. Jeff also adds the cost center we created earlier **ICT100** and clicks **OK**



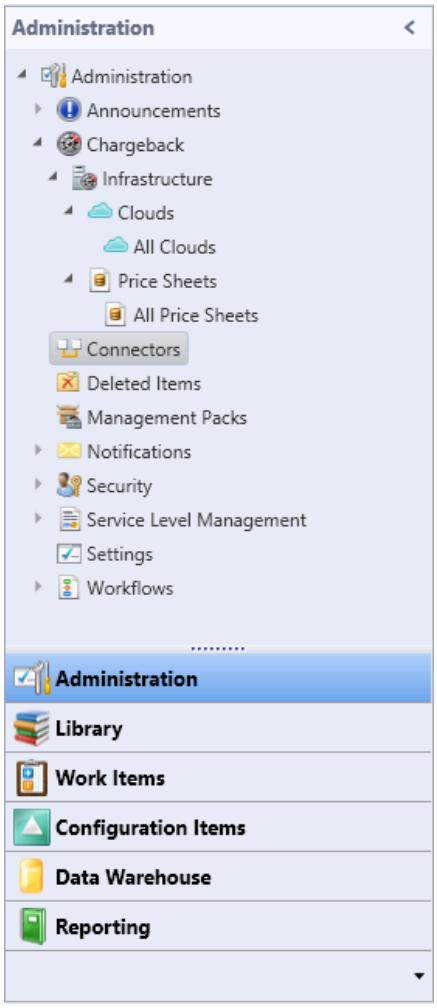
9. The last server he adds to this Cloud is the **SharePoint** server. Jeff right clicks on the this virtual machine and clicks **Properties**



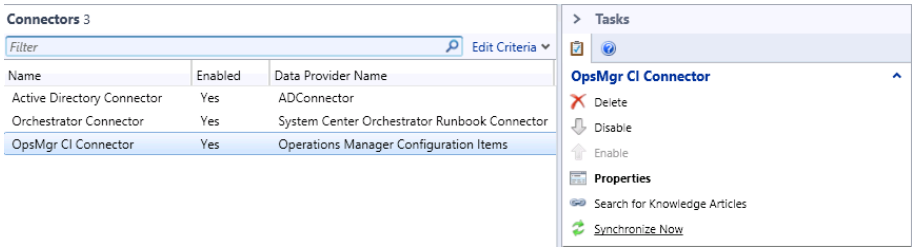
10. He then adds it to the **MyCloud** cloud but this time adds this machine to the alternative **ICT110** cost center and clicks **OK**



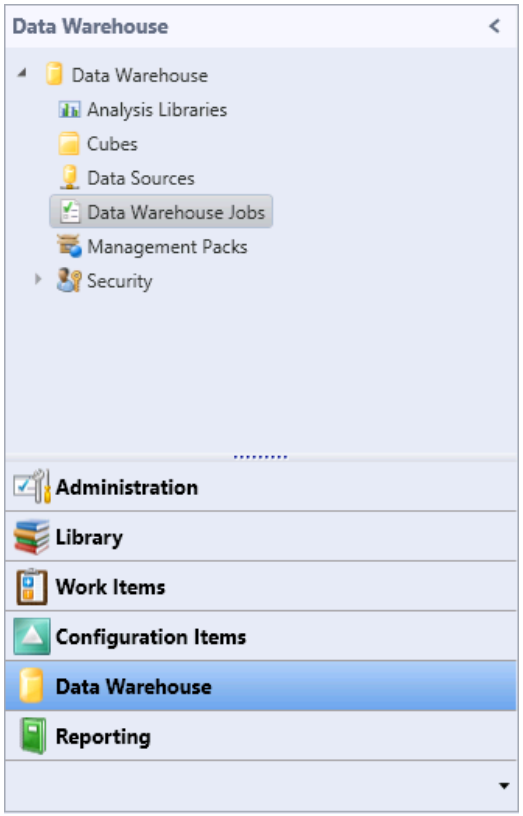
11. Jeff now jumps onto the **ServiceMgr** server and opens the **Service Manager Console**. From the **Administration** tab he now clicks on the **Connectors**



12. Jeff chooses the **OpsMgr CI Connector** we created earlier and from the available **Tasks** he forces a **Synchronize Now** from the available task list. Jeff has to wait for this synchronization to complete before carrying on; he hits F5 on the keyboard and watches the **Status** and **Percentage** columns to make sure the process finishes correctly.



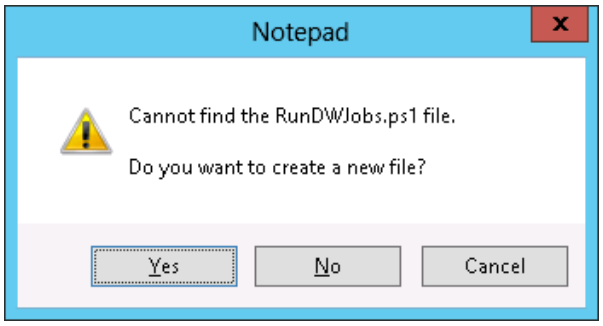
13. From the same **Service Manager console**, Jeff now clicks on **Data Warehouse** and clicks on **Data Warehouse Jobs**. He notes the jobs that are listed are part of the Service Manager Data Warehouse process, and are configured to run at either at specific schedule, i.e. once every 24 hours, where as some are running multiple times per hour. Jeff decides to write a script to have these jobs run now and in the correct sequence



14. Jeff then clicks on the start button at the bottom left corner of the desktop and opens up the **Service Manager Shell** program

15. He then types in "**notepad RunDWJobs.ps1**" and hits enter, this will now open up notepad and when prompted, he answers **Yes** to create the new file with that associated filename

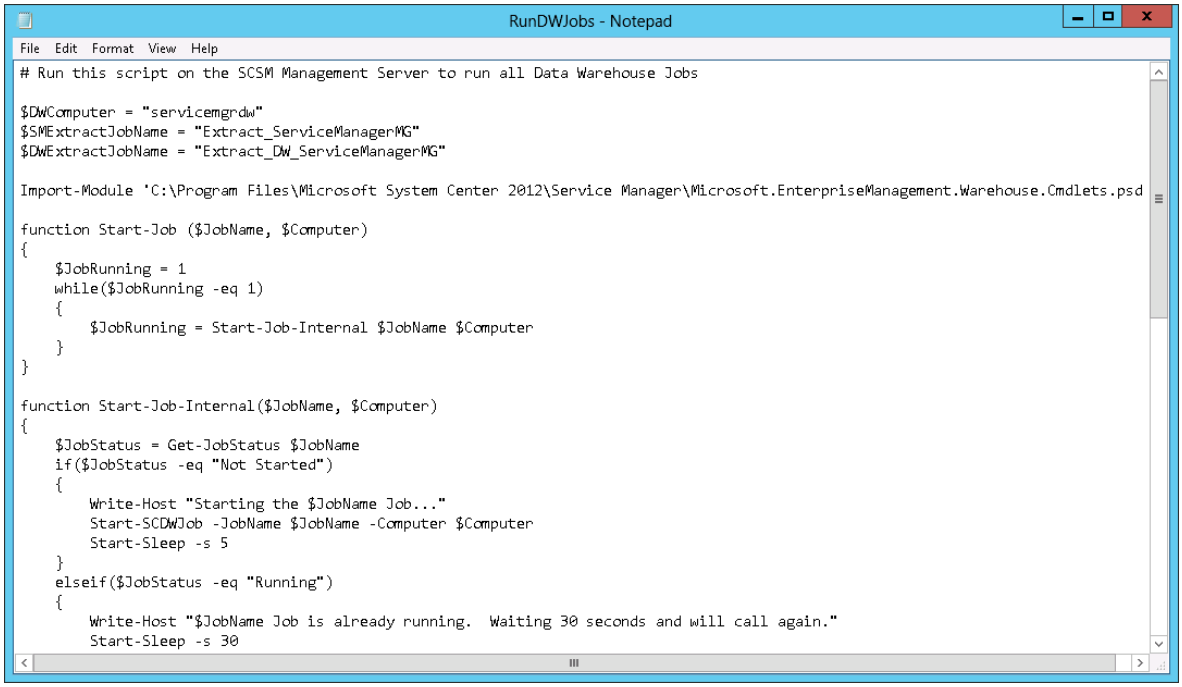
```
notepad RunDWJobs.ps1
```



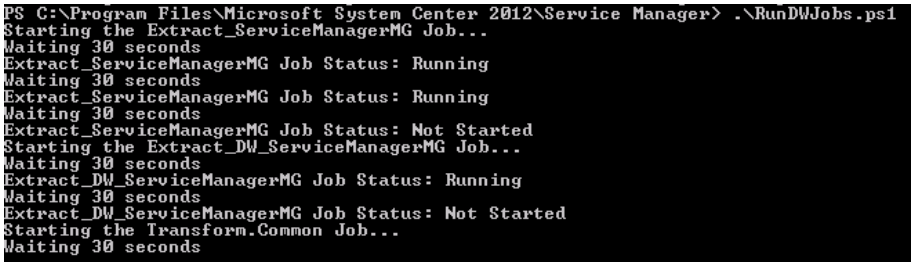
16. Jeff copies the script below into the notepad window, he then saves the file and exits notepad. This script is designed to run all the required data warehouse jobs in the correct order

```
# Run this script on the SCSM Management Server to run all Data Warehouse
Jobs

$DWComputer = "servicemgrdw"
$SMExtractJobName = "Extract_ServiceManagerMG"
$DWExtractJobName = "Extract_DW_ServiceManagerMG"

Import-Module 'C:\Program Files\Microsoft System Center 2012\Service
Manager\Microsoft.EnterpriseManagement.Warehouse.Cmdlets.psd1'

function Start-Job ($JobName, $Computer)
{
    $JobRunning = 1
    while($JobRunning -eq 1)
    {
        $JobRunning = Start-Job-Internal $JobName $Computer
    }
}

function Start-Job-Internal($JobName, $Computer)
{
    $JobStatus = Get-JobStatus $JobName
    if($JobStatus -eq "Not Started")
    {
        Write-Host "Starting the $JobName Job..."
        Start-SCDWJob -JobName $JobName -Computer $Computer
        Start-Sleep -s 5
    }
    elseif($JobStatus -eq "Running")
    {
        Write-Host "$JobName Job is already running.  Waiting 30 seconds and
will call again."
        Start-Sleep -s 30
```

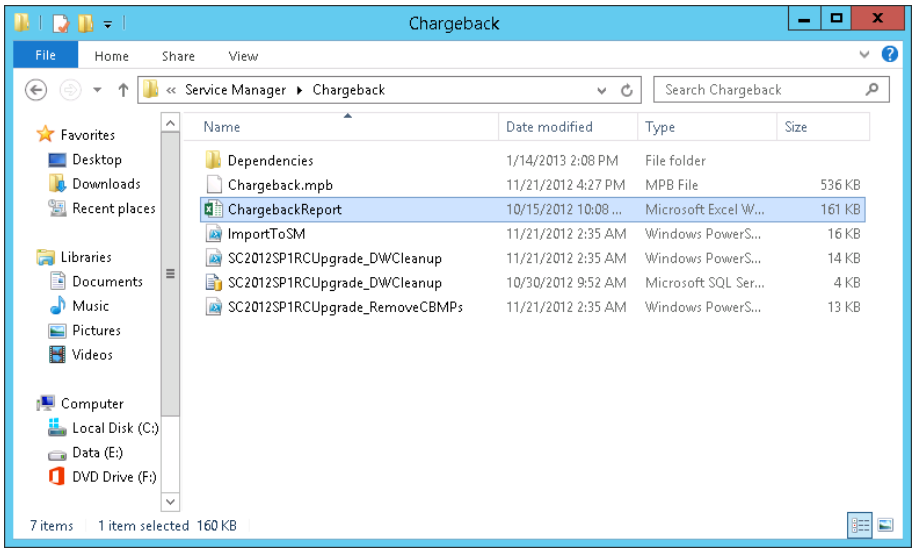
```
        return 1
    }
    else
    {
        Write-Host "Exiting since the job is in an unexpected status"
        exit
    }

    $JobStatus = "Running"
    while($JobStatus -eq "Running")
    {
        Write-Host "Waiting 30 seconds"
        Start-Sleep -s 30
        $JobStatus = Get-JobStatus $JobName
        Write-Host "$JobName Job Status: $JobStatus"
        if($JobStatus -ne "Running" -and $JobStatus -ne "Not Started")
        {
            Write-Host "Exiting since the job is in an unexpected status"
            exit
        }
    }
    return 0
}

function Get-JobStatus($JobName)
{
        $Job = Get-SCDWJob -JobName $JobName -Computer $Computer
        $JobStatus = $Job.Status
        return $JobStatus
}

#ETL
Start-Job $SMExtractJobName $DWComputer
Start-Job $DWExtractJobName $DWComputer
Start-Job "Transform.Common" $DWComputer
Start-Job "Load.Common" $DWComputer

#Cube processing
Start-Job "Process.SystemCenterServiceManagerChargebackCube" $DWComputer
```



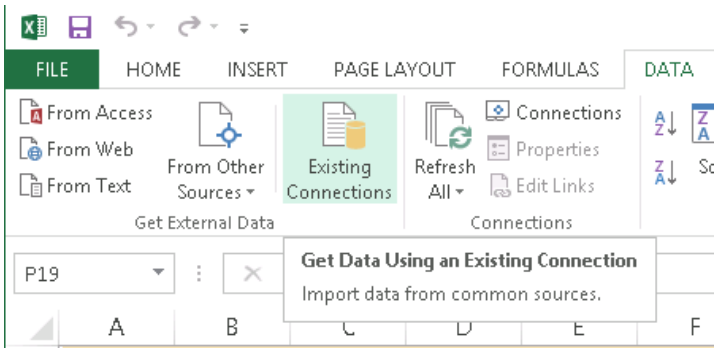
17. Back in the **Service Manager Shell** Jeff types in **.\RunDWJobs.ps1** and hits enter. This script will then start each of the required **Data Warehouse Jobs**. This process will take some time to run and by checking on the **Data Warehouse Jobs** in the Service Manager console you can check they have finished and have all run correctly.



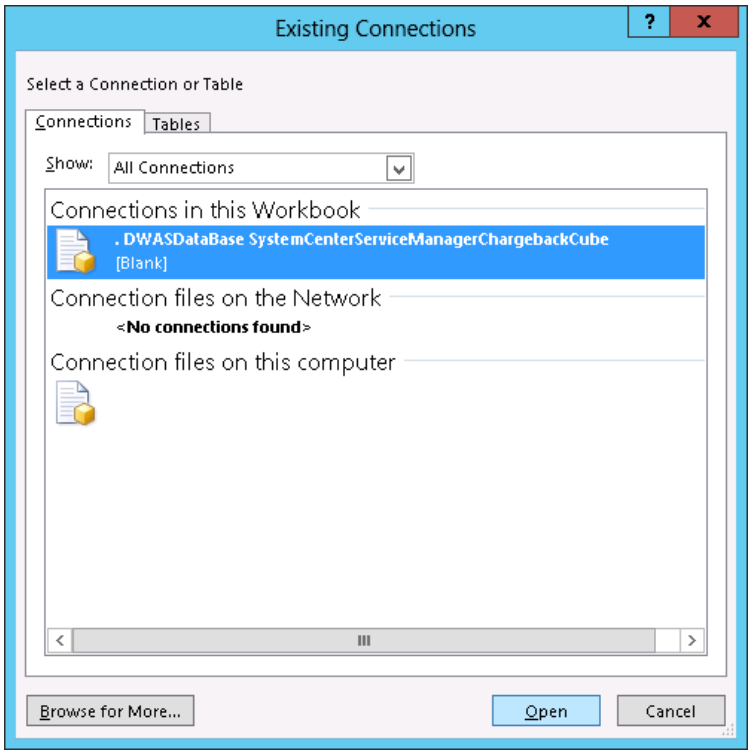
18. Jeff then connects to the **Guest01** server that we have installed the Microsoft Office 2013 Demo so we can view the chargeback reports. From within Explorer, browse to **\\servicemgr\c$\Program Files\Microsoft System Center 2012\Service Manager\Chargeback** and open the **ChargebackReport.xlsx** Excel spread sheet.



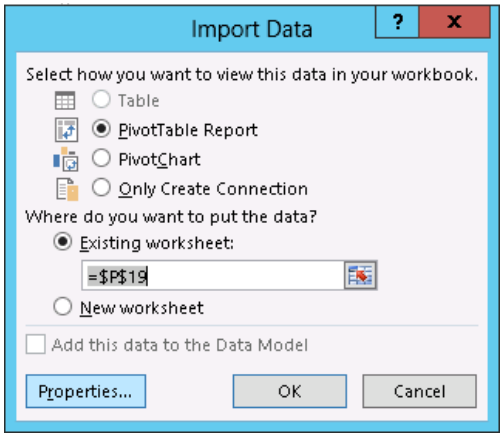
19. Jeff now has the chargeback report open but needs to configure the data source in the spread sheet to connect to the **Service Manager Data Warehouse**, as by default this report is configured to connect to **localhost**. From within **Excel** he clicks on the **Data** tab and then clicks on **Existing Connections**.



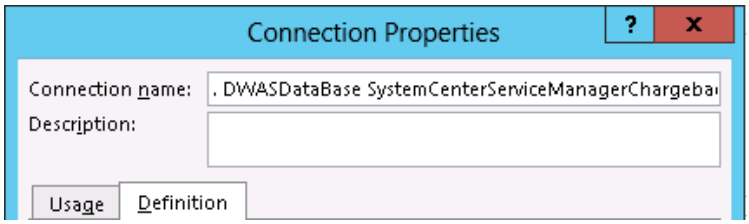
20. Jeff then clicks on the Connections in this workbook .DWASDatabase SystemCenterServiceManagerChargebackCube and clicks on **Open**.



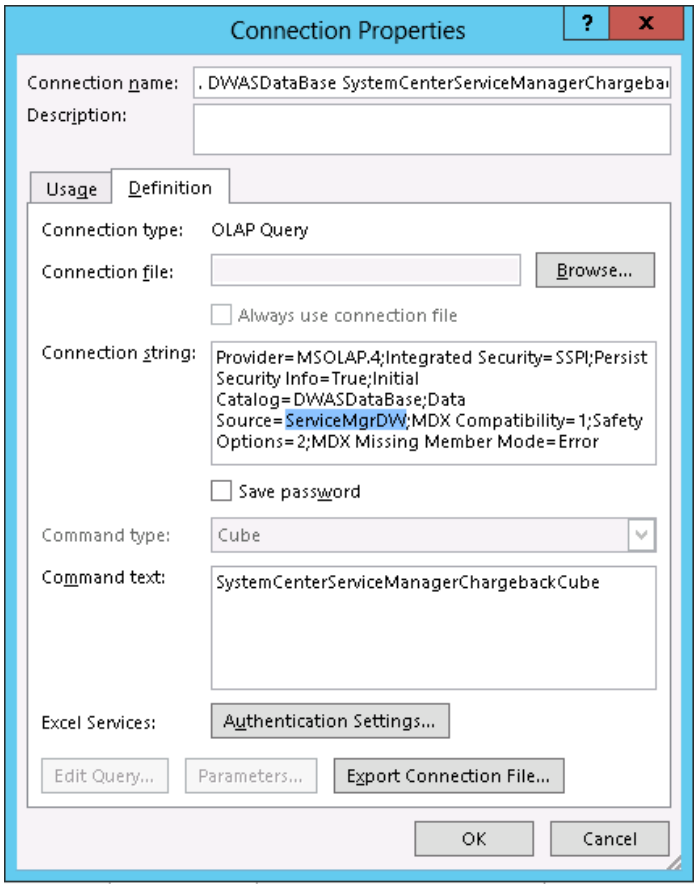
21. From the **Import Data** dialogue box click on **Properties**.



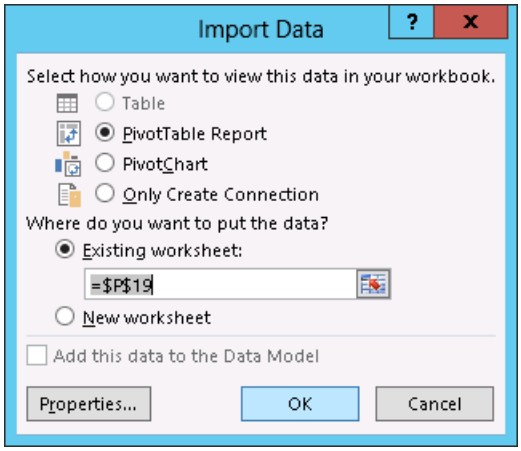
22. Then Jeff changes from the default view of the **Usage** tab at the top to the **Definition** tab



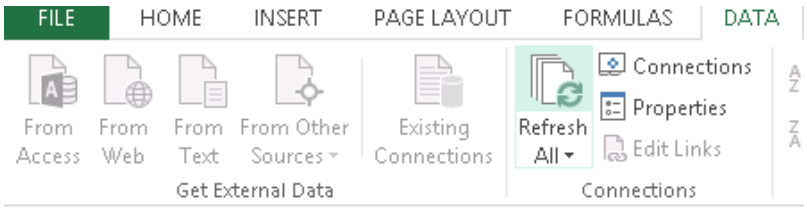
23. In the available connection string section Jeff changes the **localhost** entry in the **Source** argument to **ServiceMgrDW** and clicks **OK**



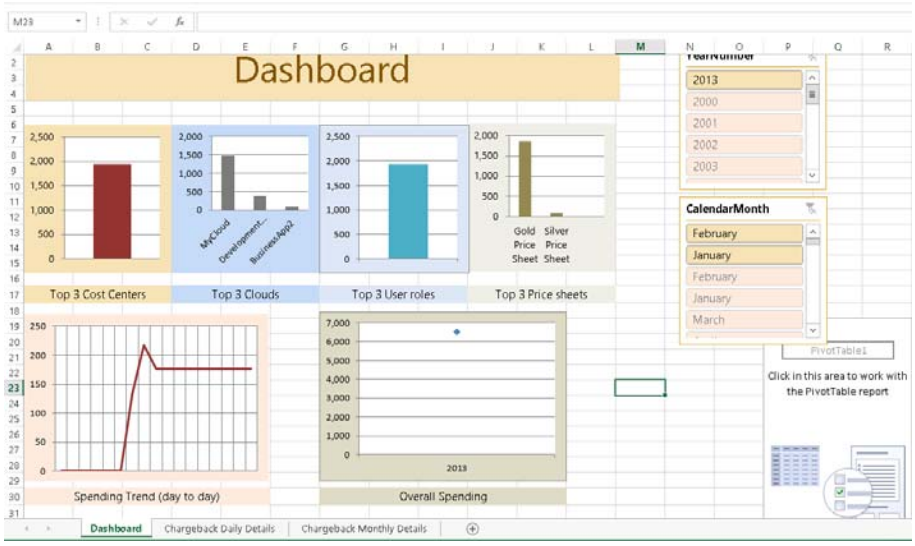
24. He then clicks **OK** when the **Import Data** window appears



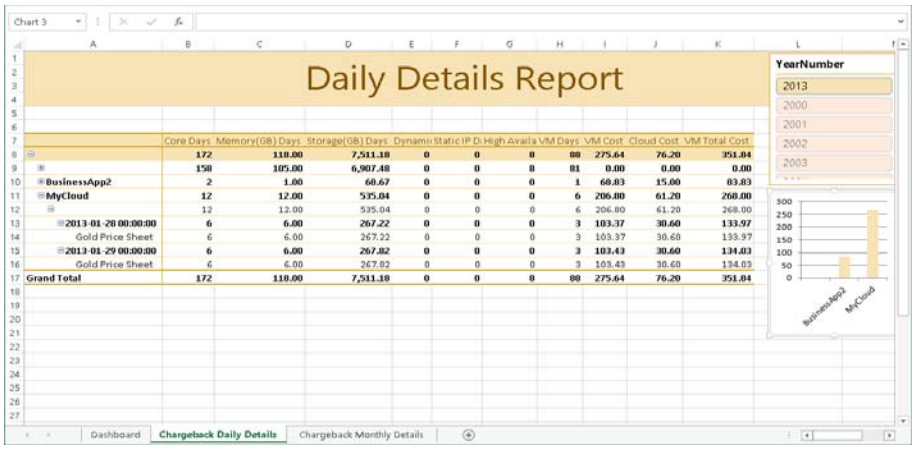
25. The chargeback report dashboard is now open but there is still no real world data available yet. To get an updated view of the information Jeff clicks on the **Data** tab within Excel and clicks on the **Refresh All** button. The Refresh All button will query Service Managers' Analysis Services Chargeback OLAP cube and imports the information required.

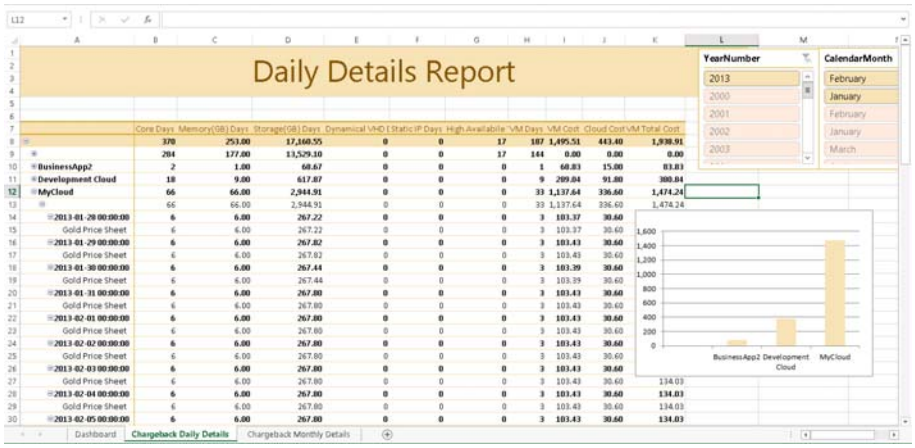


26. Jeff can now see the dashboard(s) in the spread sheet have been updated with all the information updated from Service Manager, Virtual Machine Manager and Operations Manager. The information displayed is derived from the association of the VMM Cloud(s) with the Chargeback Price Sheet assigned.

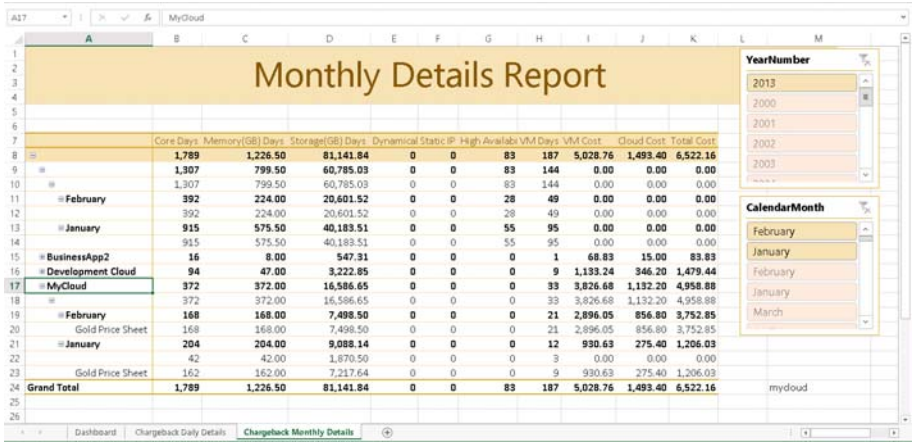


27. He then changes to the Chargeable Daily Details within the Spread sheet and notes the charges for each of the clouds deployed and their cost per day as defined by their associated price sheets.



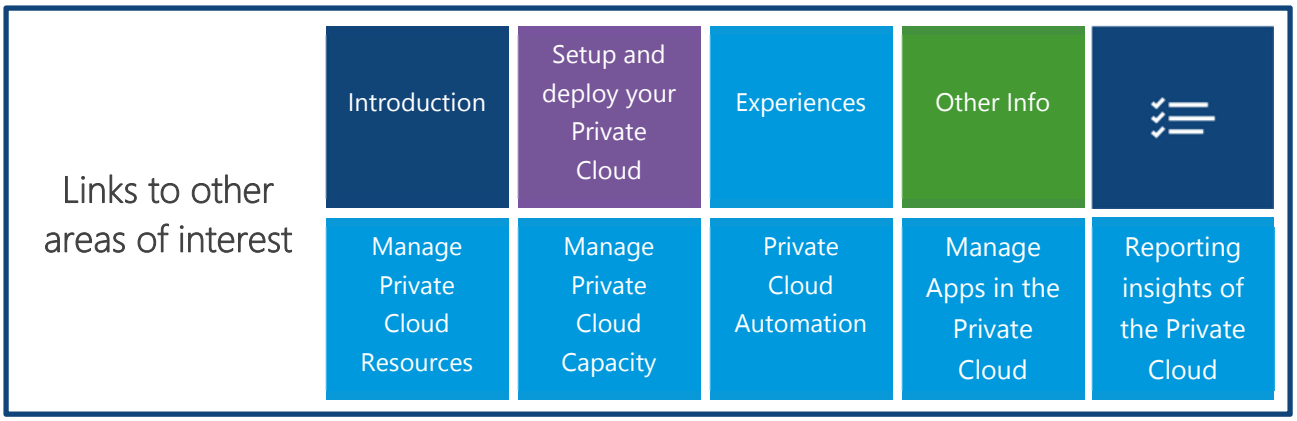


28. Finally Jeff checks out the Chargeable Monthly Details, he can drill further into the data and see exactly what cloud costs per month and how long each cloud has been deployed for.



Summary

Chargeback allows you to apply real world costs to both physical and virtual compute, storage and memory resources. Through these reports, Jeff is able to bill his external clients for the precise amount of resources they use, while keeping track of internal costs. Due to the granular nature of these reports, they can be applied to multiple levels of service, and across different grades of resources.

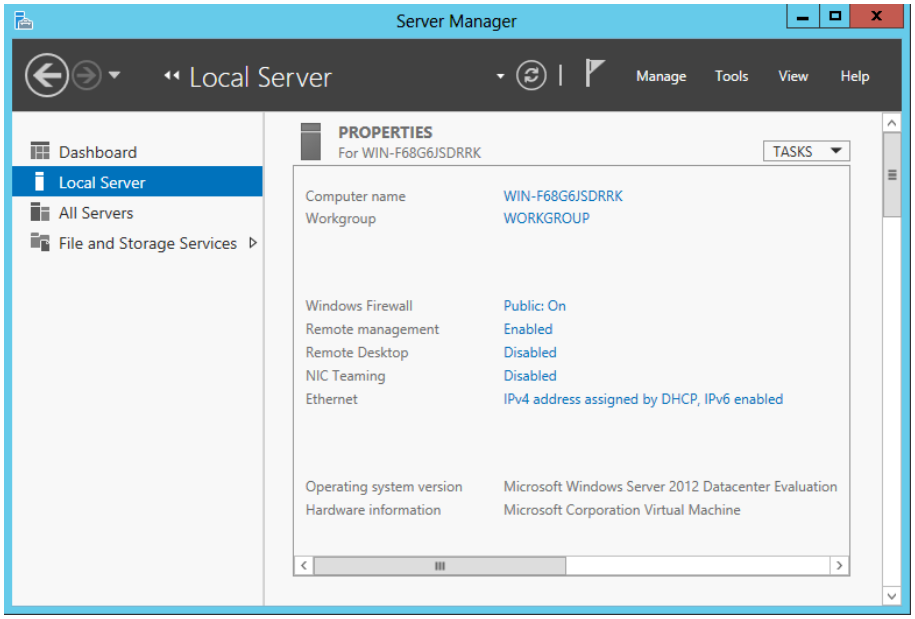

Links to other areas of interest
Introduction
Setup and deploy your Private Cloud
Experiences
Other Info
Manage Private Cloud Resources
Manage Private Cloud Capacity
Private Cloud Automation
Manage Apps in the Private Cloud
Reporting insights of the Private Cloud

Other Info

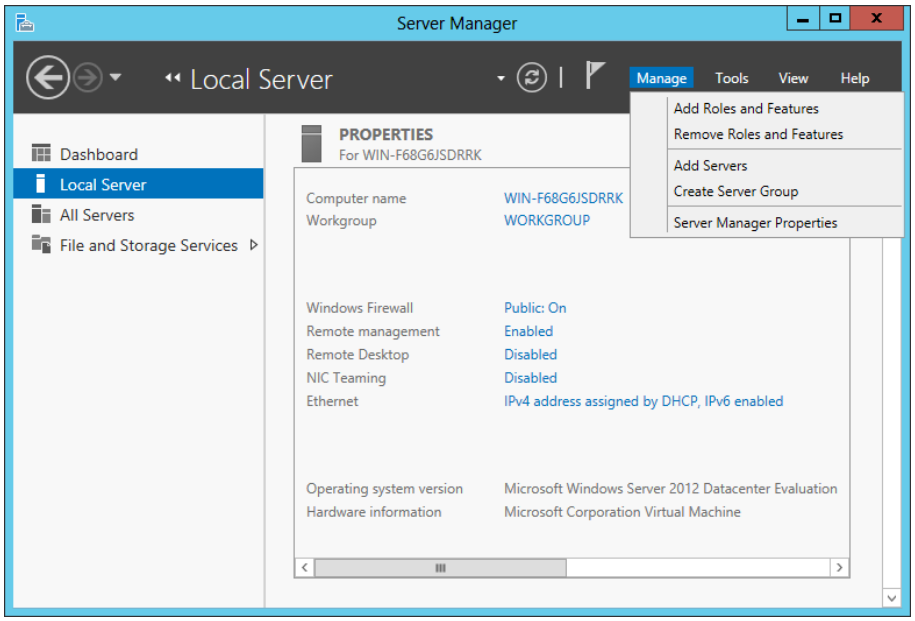
How to add a role or feature to Windows 2012

1. Open the **Server Manager** console

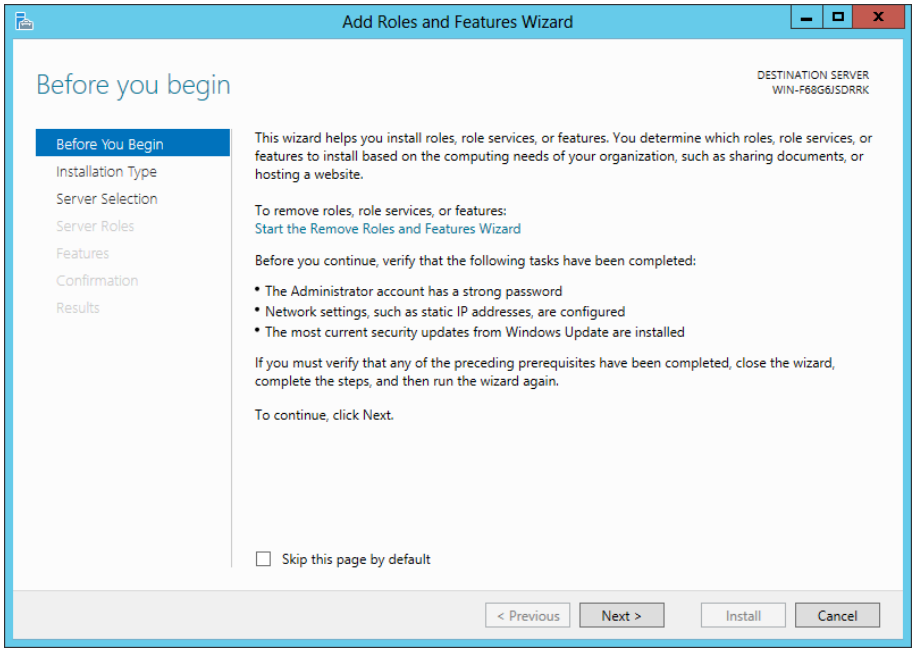
2. Click on Local Server



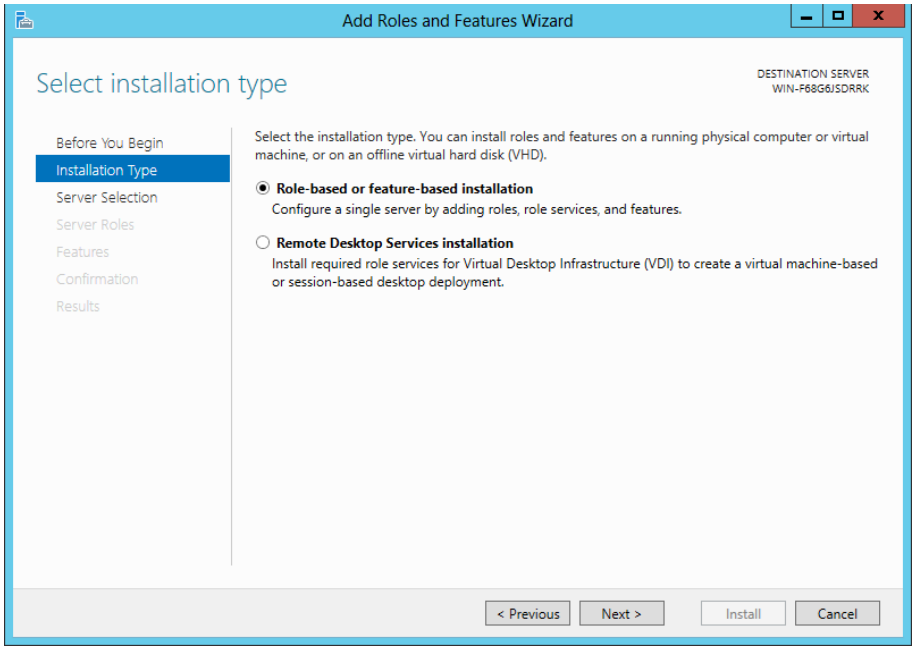
3. Click on Manage and select Add Roles and Features



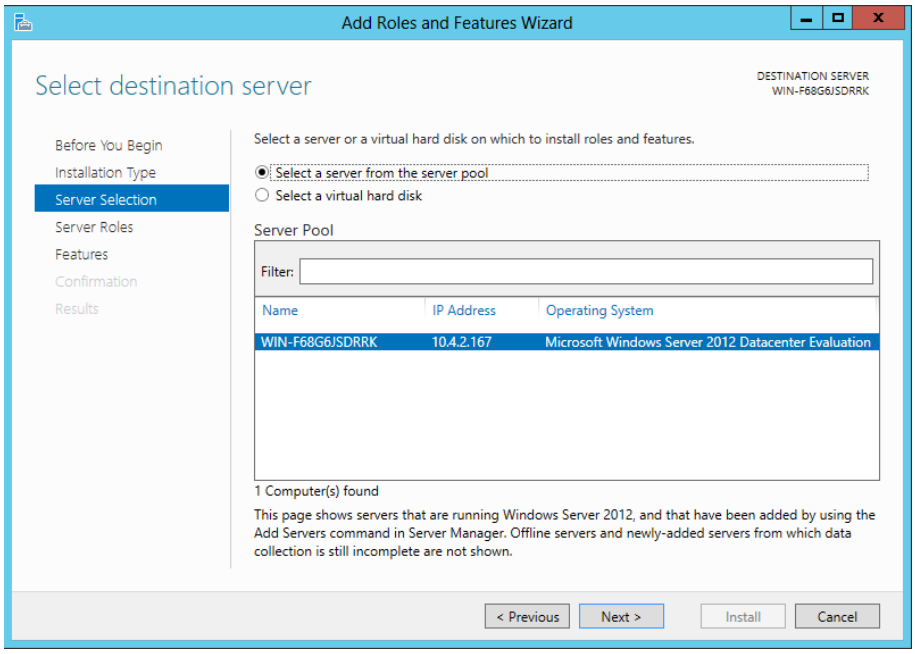
4. Click **Next**



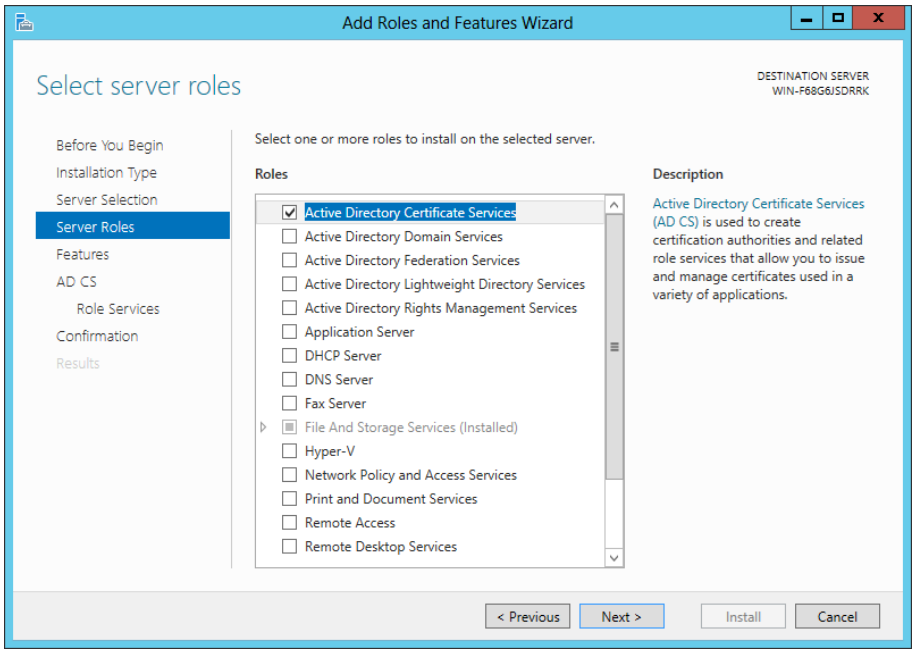
5. Select Role-based or feature based installation and click next



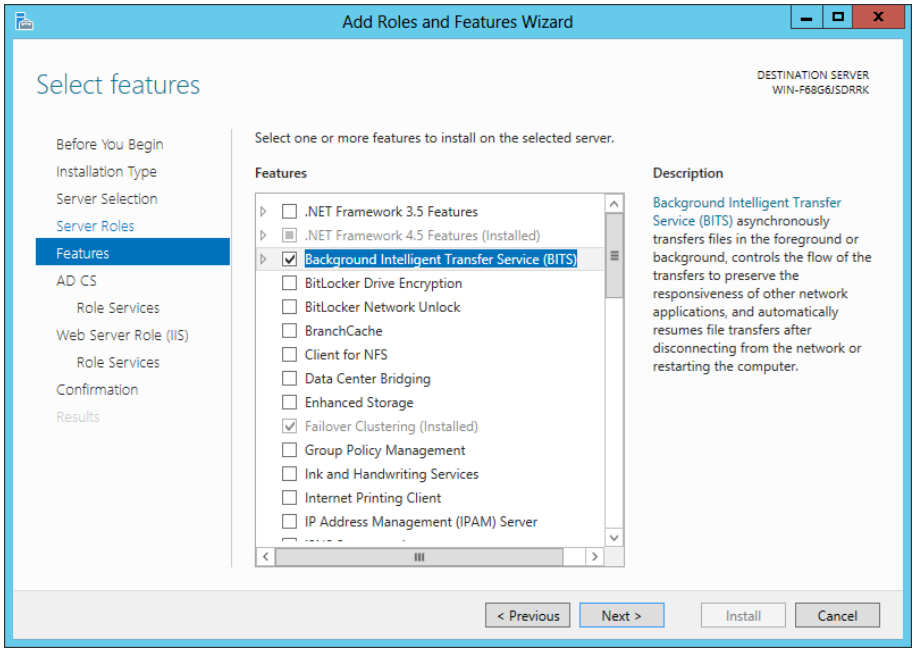
6. On the Server Selection step click **Next**



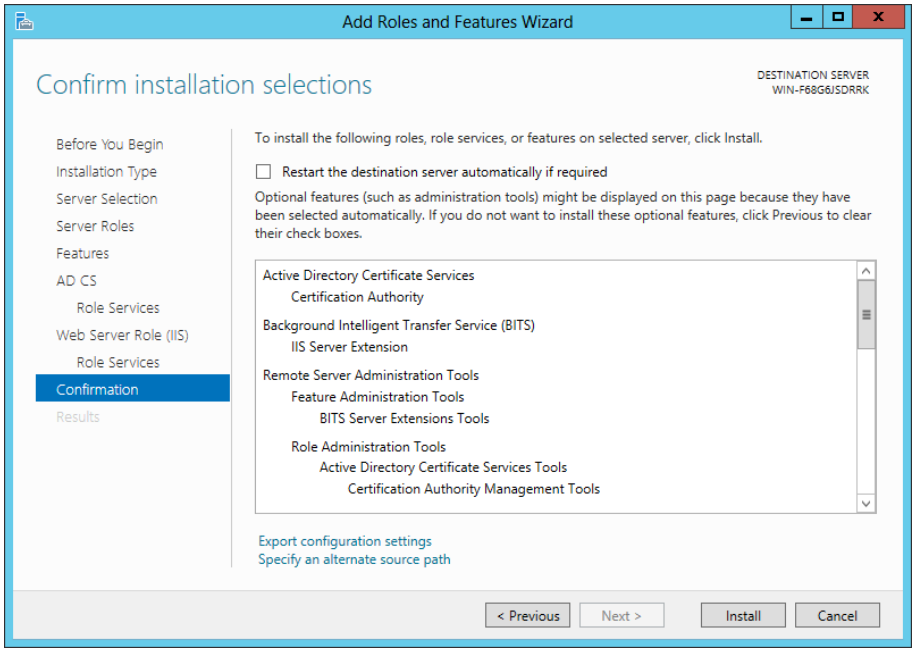
7. Select the **Role** you wish to install and click **Next**



8. Select any **Features** you wish to install and click **Next**

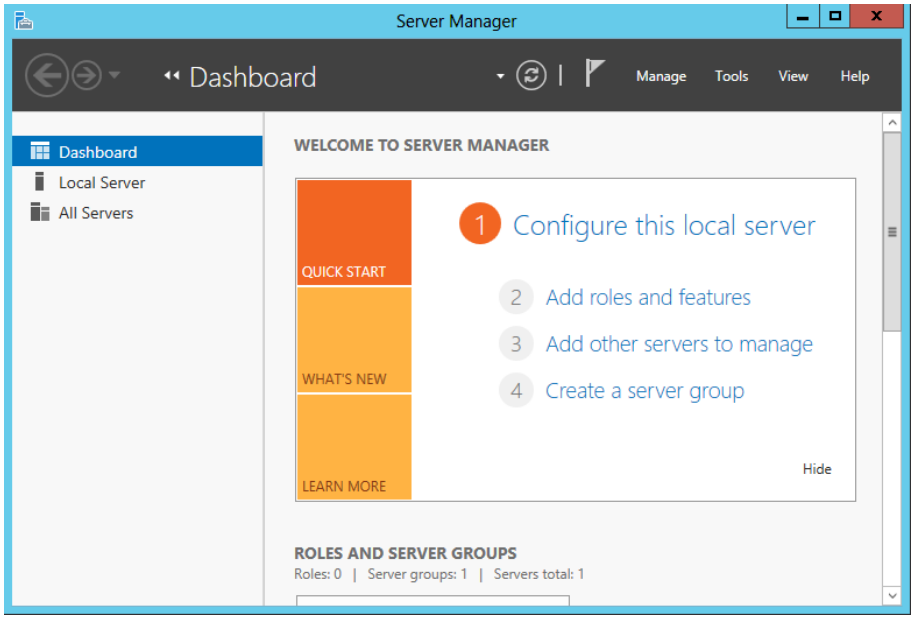


9. Some Roles may require extra configuration at this point. Follow the wizard through, answer all questions and click next until the **Confirm Installation** screen appears and click **Install**

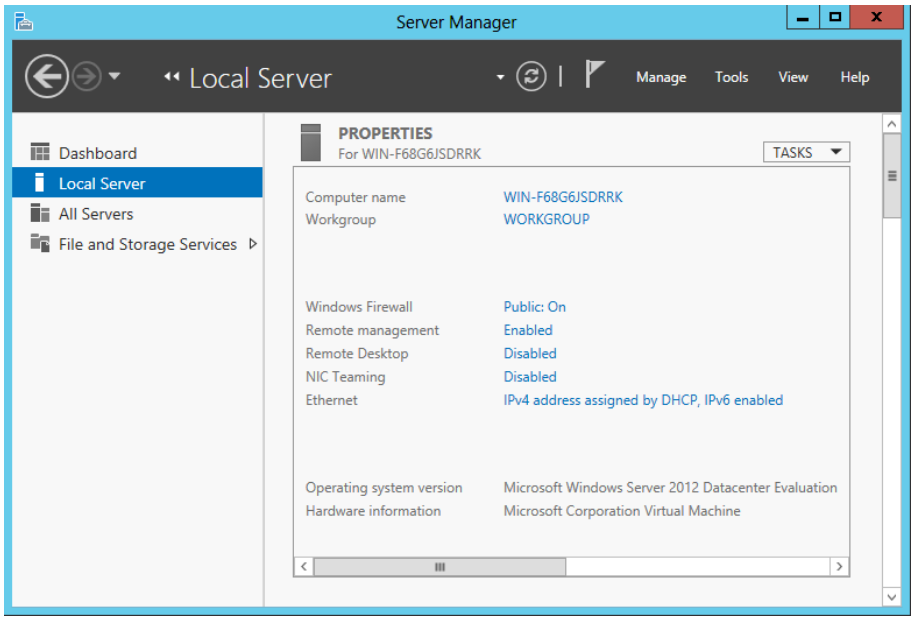


How to set an IPV4 address

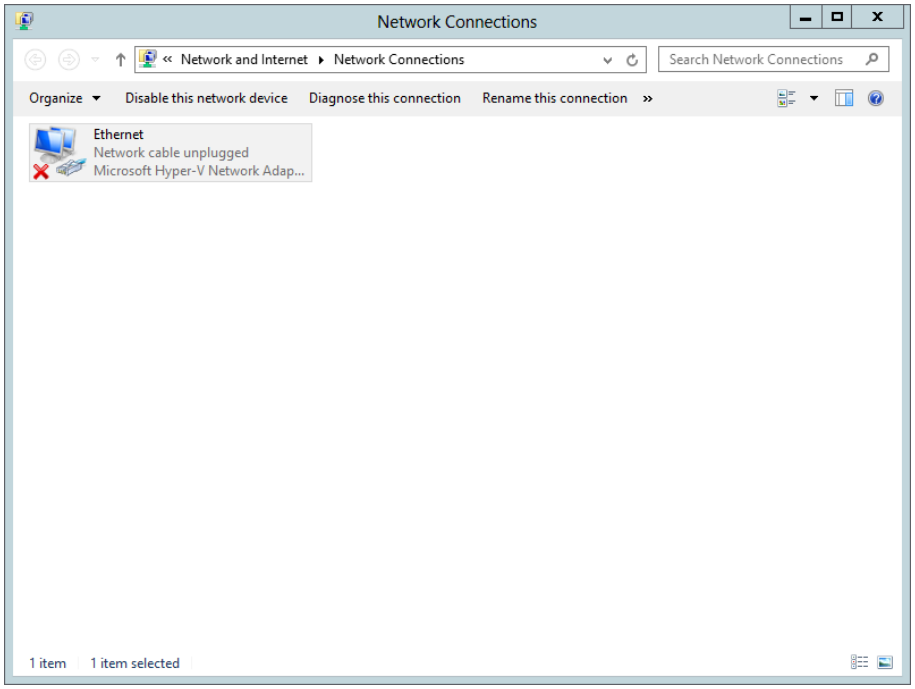
1. Open the **Server Manager** console



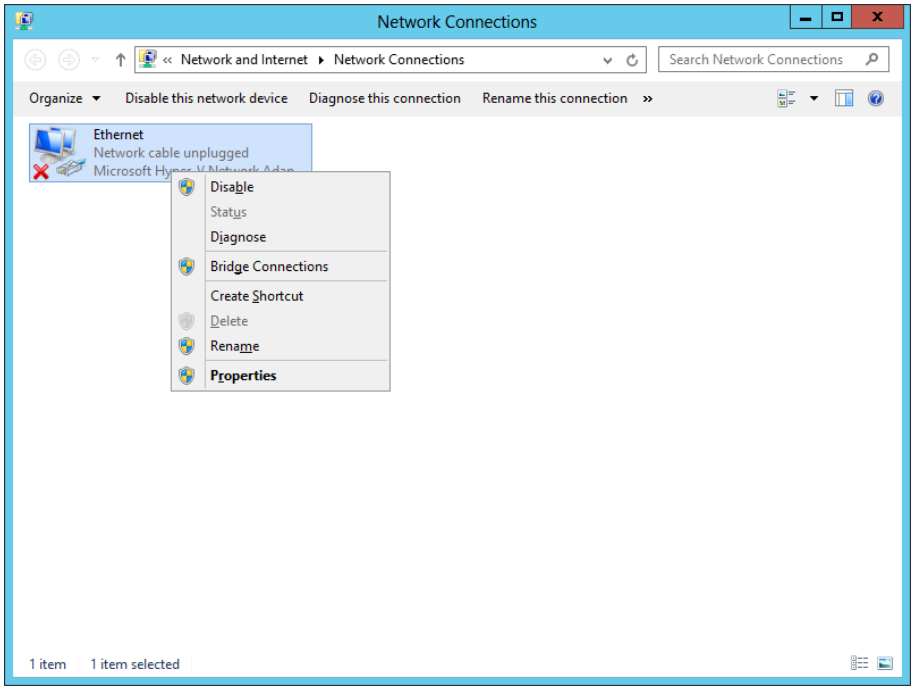
2. Click on Local Server



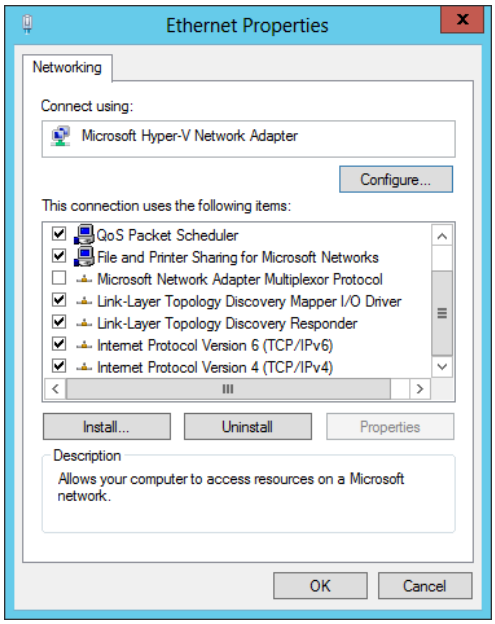
3. Click on the entry next to **Ethernet** (usually ***IPv4 Address assigned by DHCP*** if there is a DHCP server present). The **Network Connections** window will open



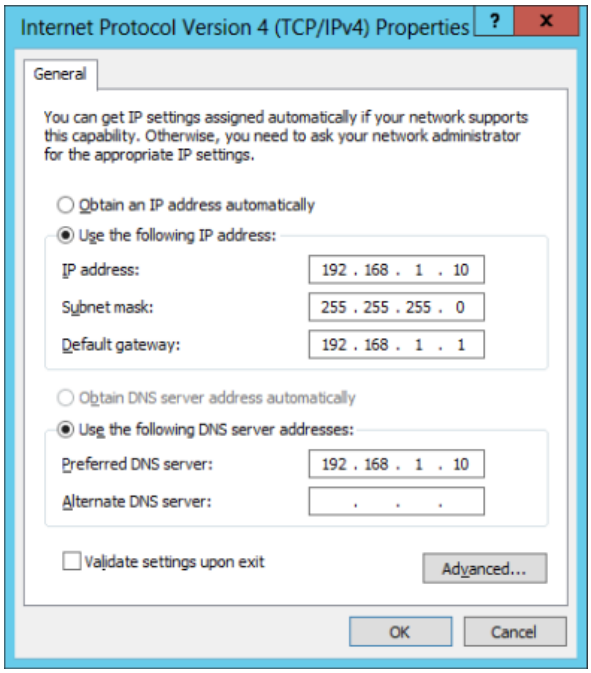
4. Right click the network icon labelled **Ethernet** and select **Properties**



5. Click on **Internet Protocol Version 4 (TCP/IPv4)** and click the **Properties** button



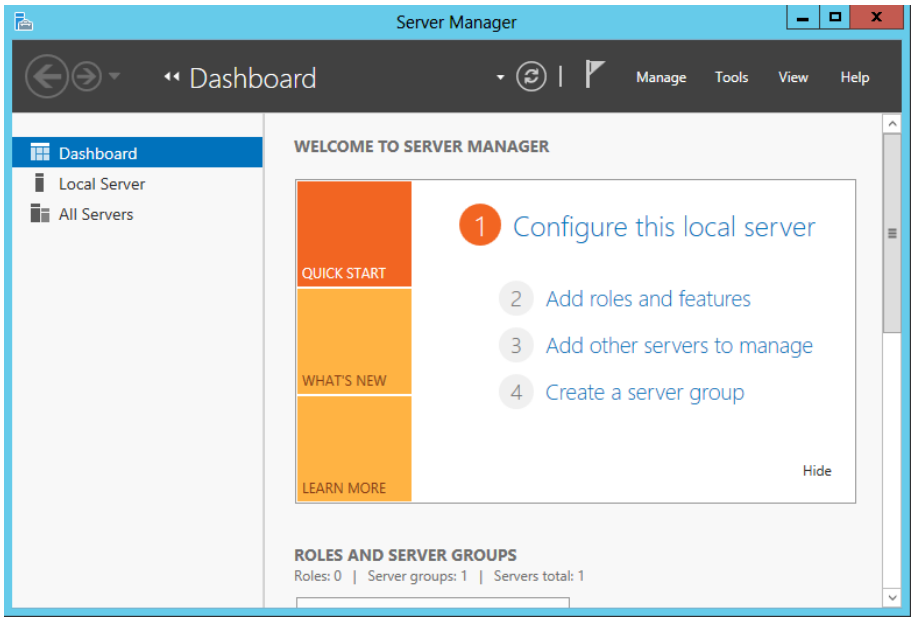
6. Enter the appropriate network information. Note that this information can be different in your environment. Click on **OK**



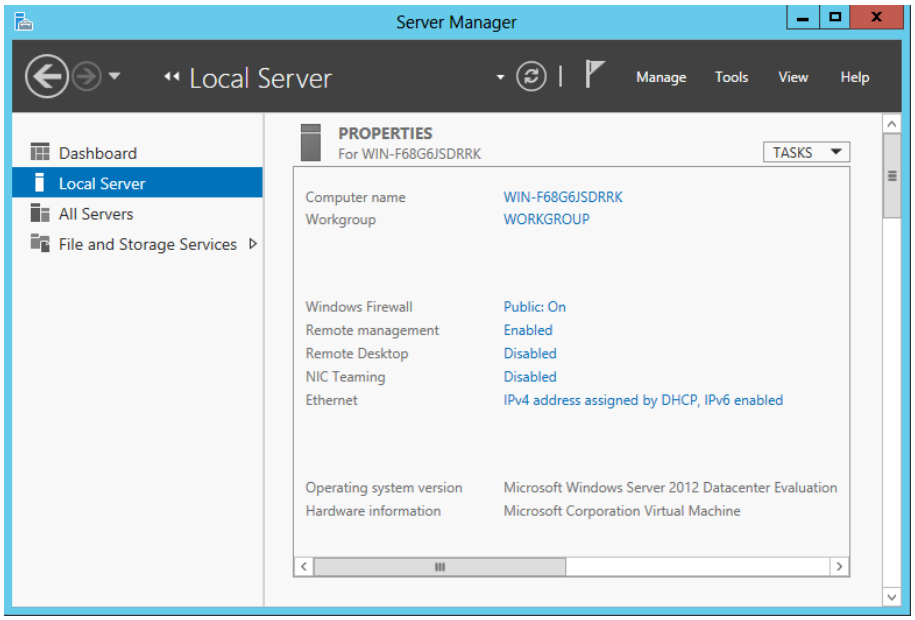
The static IP address has now been set.

How to join the domain

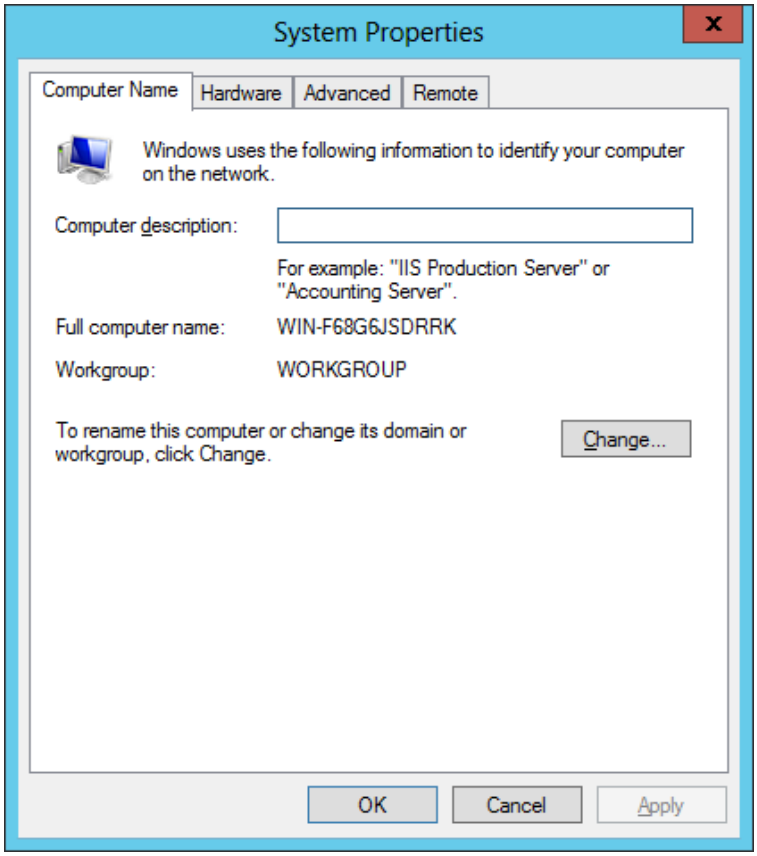
1. Open the **Server Manager** console



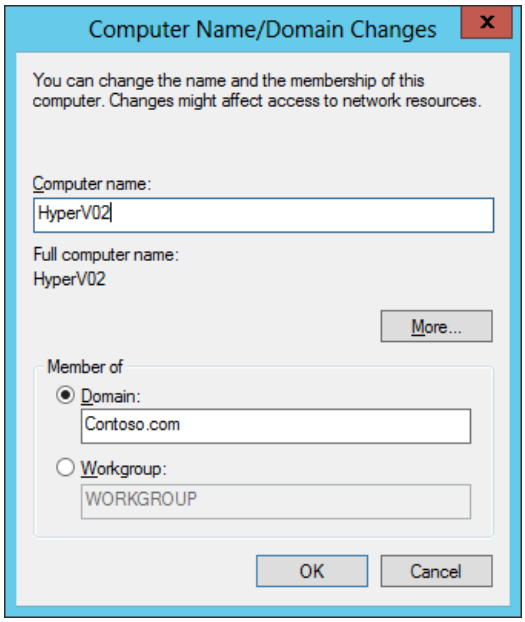
2. Click on **Local Server**, then click on **WORKGROUP**



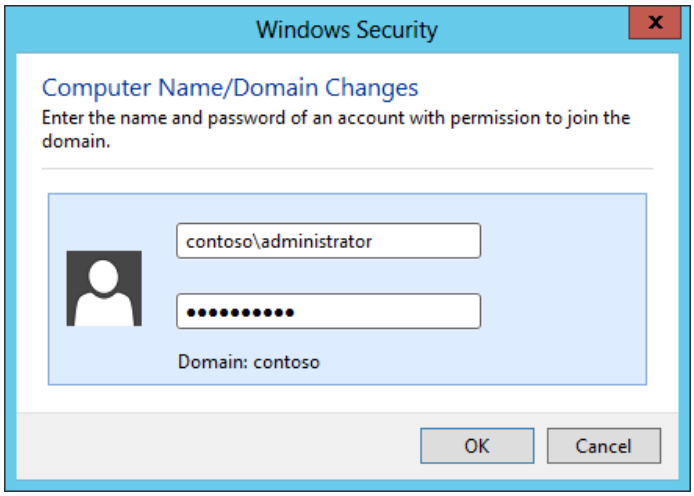
3. Click on **Change...**



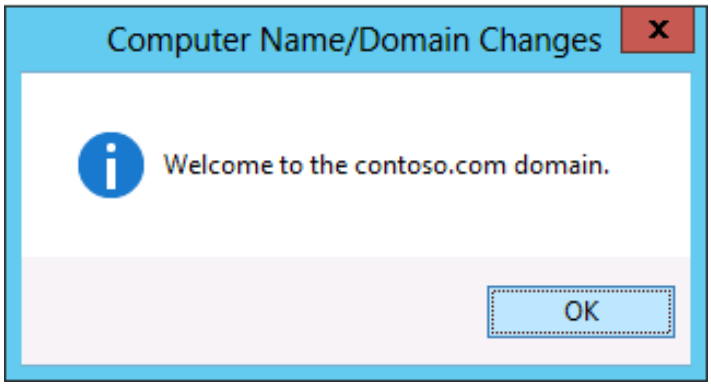
4. Input the name you would like to use, click the **Domain** radio button, then type the name of your domain e.g. **Contoso.com**. Click on **OK.**



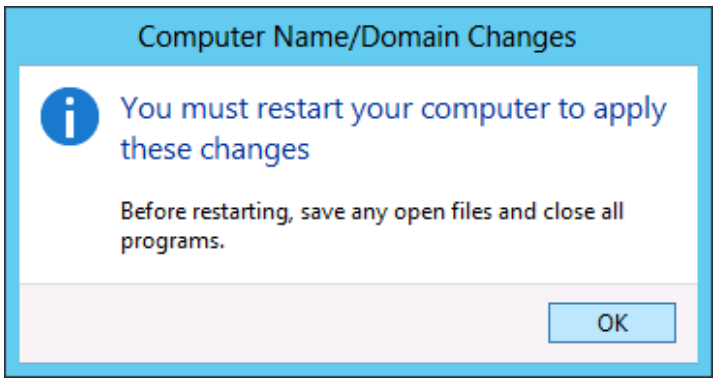
5. When prompted input the domain administrator credentials in the **Windows Security** dialog box. Click on **OK**



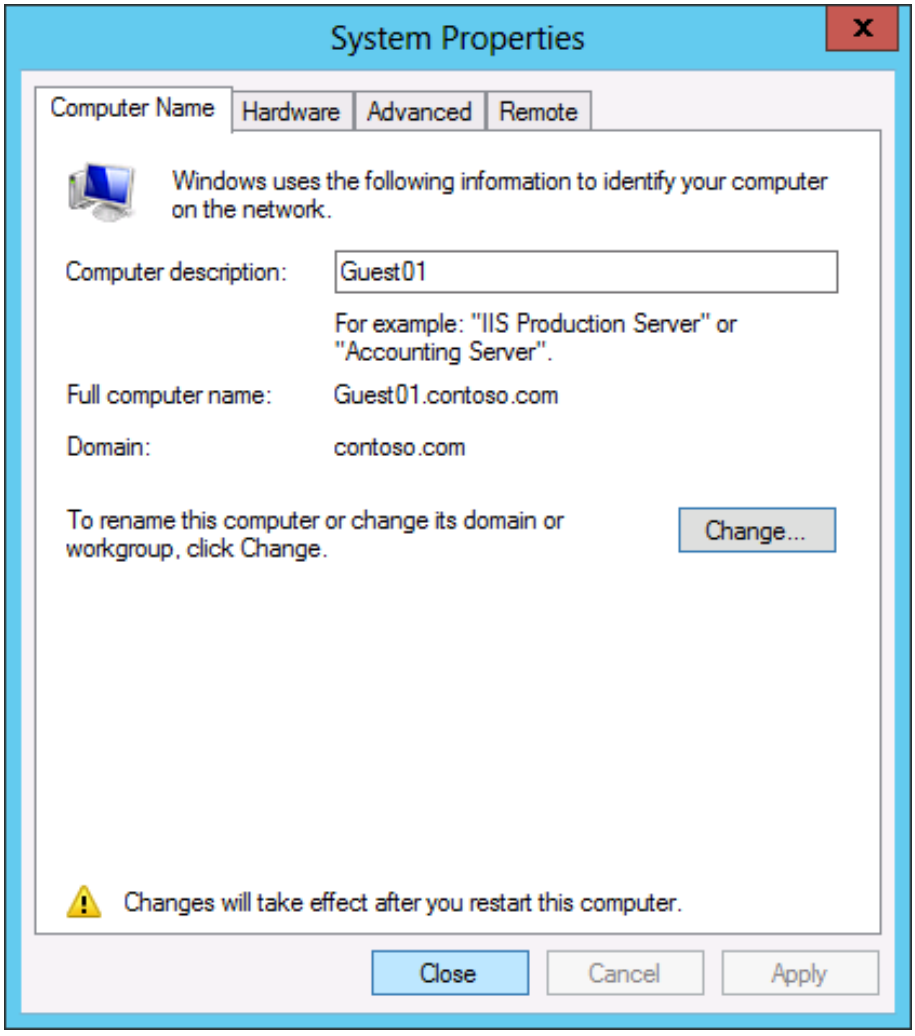
6. Click **OK**



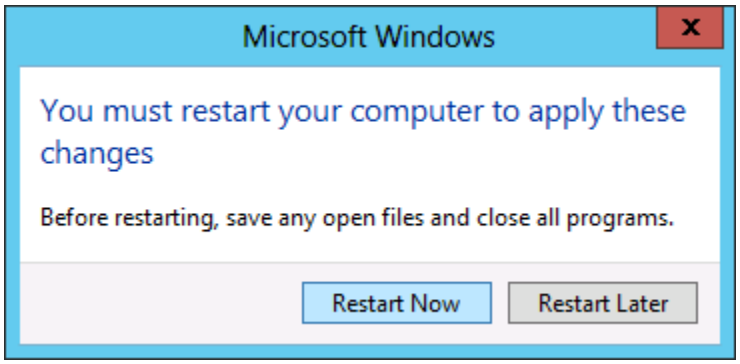
7. Click **OK**



8. Click **Close**



9. Click **Restart Now**



10. Now the server is a member of the domain and any group policies you have created will be applied.