ΑΛΕΞΑΝΔΡΕΙΟ Τ.Ε.Ι. ΘΕΣΣΑΛΟΝΙΚΗΣ

ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

# ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ
# VIDEO STORAGE AND PROCESSING ON A PRIVATE CLOUD

| | |
|---|---|
| **Του φοιτητή** | **Επιβλέποντες καθηγητές** |
| **Δανιηλίδη Ορέστη** | **Dr. Martin Fleury (Univ. of Essex)** |
| **Αρ. Μητρώου: 06/3072** | **Δρ. Περικλής Χατζημίσιος (ΑΤΕΙΘ)** |

**Θεσσαλονίκη 2013**

Alexander Technological Educational Institute Of Thessaloniki

# B.Sc. Thesis

# VIDEO STORAGE AND PROCESSING ON A PRIVATE CLOUD

**Author/Student**

**Daniilidis Orestis**

**06/3072**

**Supervisors**

**Dr. Martin Fleury (Univ. of Essex)**

**Dr. Periklis Chatzimisios (ATEITHE)**

**Department of Information Technology - Thessaloniki 2013**

## ABSTRACT

Cloud computing provides an increasingly popular way to use computing and storage technology and has a major impact on the way businesses deliver services. A lot of IT companies are offering computation, storage and application hosting services in a widely accessible way. All these computational resources are being delivered to the users-costumers following a pay-per-use pricing model, where the user is charged according to a licensing agreement. The services, as described later in this thesis, can be offered in three ways, IaaS (Infrastructure as a Service), PaaS (Platform as a service) and SaaS (Software as a Service).

The purpose of this thesis is the creation of a representation of a private cloud on a single server and the creation of a video transcoding application working on it. For the implementation of the cloud part we employ System Center Virtual Machine Manager 2012 SP1, where we create three Hyper-V virtual machines specifying each of their tasks and assigning the recourses needed for them. The application, is created in java and we use a library provided by FFmpeg, a cross-platform solution to record, convert and stream audio and video. The application gives the user the option to store a video in his/her own directory, convert it to various types, as well as change its properties and download it.

This thesis has gone through numerous stages in order to get its final form. At the beginning the main task was to work on an existing cloud platform to host the application, but after a detailed discussion with the supervisors, we decided to make an implementation from bare-metal and use our own resources.

## KEYWORDS

Cloud computing, private cloud, SaaS, virtualization, virtual machine, Hyper –V, System Center Virtual Machine Manager (SCVMM), FFmpeg, transcoding, H.264.

## ΠΕΡΙΛΗΨΗ

Το cloud computing αποτελεί μια ολοένα και πιο δημοφιλή τεχνολογία επεξεργασίας και αποθήκευσης και έχει μεγάλο αντίκτυπο στον τρόπο που οι εταιρίες προσφέρουν υπηρεσίες. Πολλές IT εταιρίες προσφέρουν σαν υπηρεσία επεξεργασία και αποθήκευση δεδομένων όπως και αποθήκευση εφαρμογών με έναν πλήρως προσβάσιμο τρόπο. Τα παραπάνω παρέχονται στους χρήστες-πελάτες ακολουθώντας ένα pay-per-use μοντέλο κοστολόγησης, βάση του οποίου ο χρήστης χρεώνεται σύμφωνα με κάποιους προσυμφωνημένους όρους χρήσης. Οι υπηρεσίες, όπως περιγράφονται στη συνέχεια, μπορούν να έχουν τρεις μορφές, IaaS (Infrastructure as a Service), PaaS (Platform as a service) και SaaS (Software as a Service).

Σκοπός της παρούσας εργασίας είναι η δημιουργία μιας απεικόνισης ενός private cloud σε ένα μοναδικό server και η δημιουργία μιας video transcoding εφαρμογής αποθηκευμένης σε αυτόν. Για την υλοποίηση του cloud χρησιμοποιήθηκε το System Center Virtual Machine Manager 2012 SP1, με το οποίο δημιούργησα τρία Hyper-V virtual machines αναλύοντας τα καθήκοντα του καθενός και κατανέμοντας τους διαθέσιμους πόρους σε αυτά. Όσον αφορά στην εφαρμογή, που υλοποιήθηκε σε java, χρησιμοποιήσαμε το FFmpeg, μια βιβλιοθήκη για την εγγραφή, μετατροπή και μετάδοση ήχου και εικόνας. Η εφαρμογή δίνει στο χρήστη τη δυνατότητα να αποθηκεύσει ένα video σε ένα δικό του φάκελο, να αλλάξει τον τύπο του αρχείου, όπως και να αλλάξει τις ιδιότητές του και να το κατεβάσει.

Η παρούσα πτυχιακή έχει περάσει από πολλά στάδια μέχρι να φτάσει στην τελική της μορφή. Αρχικά η βασική ιδέα ήταν να δουλέψουμε σε ένα υπάρχον cloud για να αποθηκευτεί η εφαρμογή, αλλά μετά από αρκετές συζητήσεις με τούς καθηγητές μου, αποφασίσαμε να κάνουμε την υλοποίηση εξ’ ολοκλήρου χρησιμοποιώντας δικούς μας πόρους.

## ACKNOWLEDGEMENTS

With the completion of this thesis I would like to thank my supervisors Dr. Periklis Chatzimisios and Dr. Martin Fleury from the ATEI of Thessaloniki and the University of Essex respectively, as well as Dr. Al-Majeed Salah for their guidance and supportiveness. Also, many thanks to the PhD student Ibrahim K Musa for his help and advices on implementing the cloud part of the thesis and my colleagues Konstantina Christoforidou and Stelios Adamantidis for their contribution. Finally, all this would not be possible without the support and patience of my parents.

## CONTENTS

## FIGURES

## TABLES

## DETAILED STRUCTURE OF THE THESIS

This thesis is about the creation of a private cloud consisted of three virtual machines and the creation of an application that transcodes videos for registered users.

The first chapter covers the theoretical framework of this thesis. It is divided in two major parts. The first one is about cloud computing and the second one about video encoders-decoders. Both parts start with a historical overview, continue with a description of the terms associated with the topic and present all their features.

In the second chapter the cloud is being created. The chapter starts with a demonstration of the technical features of the server, as well as everything that needed to be installed in order for our cloud to take place. After that, there is a detailed step by step guide of the installation of System Center Virtual Machine Manager SP1, along with all its prerequisites. Also, the fabric and all the properties for the implementation are set up. In conclusion, we go through the creation of the virtual machines that will be used and all their properties.

The third and final chapter of the thesis covers all the aspects of our application. At the beginning there is a description of how the application works, with a presentation of the tasks of each virtual machine separately. Furthermore, you can find everything about the transcoding part of the application like the library used, the codecs supported by it, as well as all the properties the user has the option to set on his videos. The last part of the chapter consists of all the key parts of our code, written in java, along with a description of each method and the way the connection is being established.

# 1. THEORITICAL BACKGROUND

## INTRODUCTION

In this chapter you can find all the theoretical baselines you need to understand this thesis in addition to some interesting information. In the beginning there is an introduction to the cloud computing and its aspects, followed by its types and some examples. After that you can find information regarding videos, such as a historical overview and their most important characteristics and finally, a brief introduction to the encoders and decoders and their usage.

## 1.1 CLOUD COMPUTING

Cloud computing is a relatively new and promising technology. It is the result of evolution and adoption of existing technologies and its goal is to allow users to take advantage of many technologies, without the need to expertise with each one of them. The cloud aims to cut costs, and help the users focus on their core business instead of being impeded by IT obstacles.

The main idea is to share the resources in the most effective way in order to achieve coherence and economy. Usually the resources are shared by multiple users and are also dynamically re-allocated per demand. A common example used is a cloud computing that serves Europeans during their business hour for an application like an email server and the same resources are reallocated to serve Americans during their business hours for another application such as a web server.

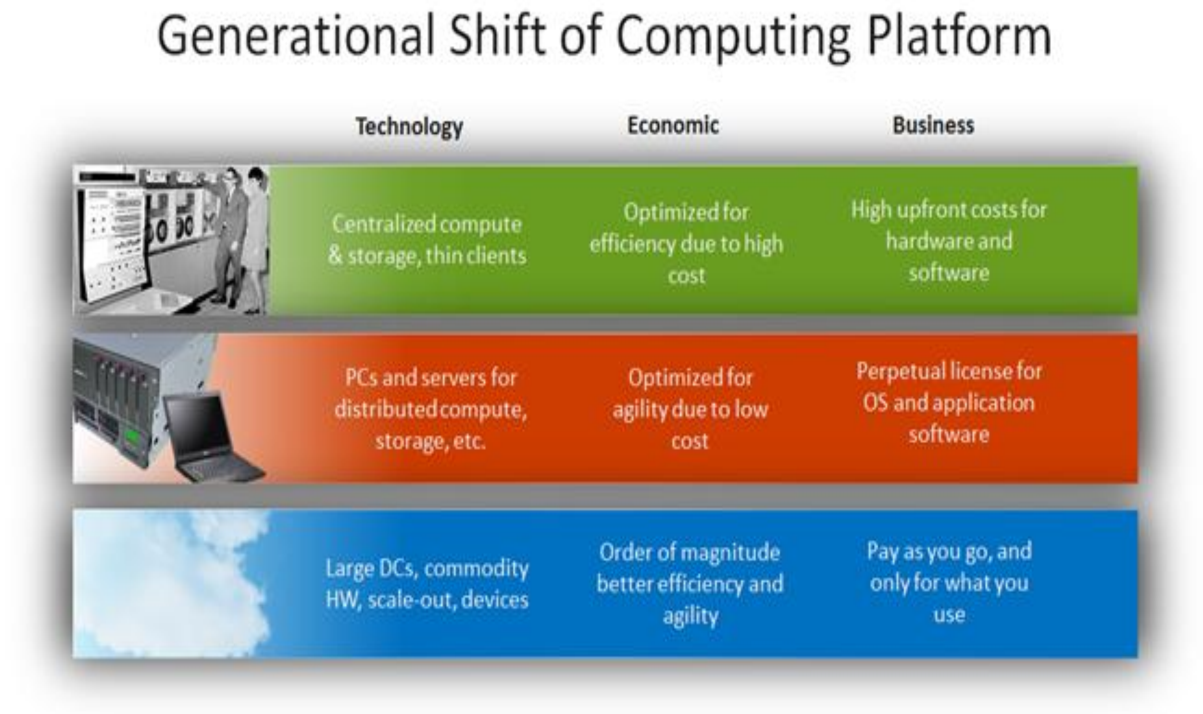
There is no commonly accepted definition worldwide. In science, it is described as a software system, with its components located on networked computers that communicate and coordinate their actions by passing messages. These components interact with each other in order to achieve a common goal. More commonly, it refers to network-based services that were seemingly provided by real server hardware but in fact was virtual hardware simulated by software running on one or more real machines. Such virtual servers do not physically exist and can therefore be moved.

### 1.1.1 HISTORY

Although cloud computing is a rather new term, it is built upon many years of distributed technologies. As a matter of fact, surprisingly, the whole idea of cloud computing exists for more than 50 years now. Back in 1961, John McCarthy stated that "computation may someday be organized as a public utility" and he also tried to describe how this could happen.

The underlying concept dates back to 1950s when large scale mainframe computers became available for first time. They were very big cabinets with a central processing unit and a main memory and no explicitly interactive interface, accessible via computers with fewer features. In the beginning they were used by corporate, academic and governmental organizations and later widely, but their cost was so high that people were thinking of how to eliminate the inactivity times and use it more efficiently. The solution was found some years later. Multiple users shared access from different terminals and used the CPU at the same time, which is known as time-sharing. Many visionaries described the main idea and usage of the existing nowadays cloud. Over the next few years and as computers were becoming more prevalent, new ways were created to make large-scale computing power available to more users through time sharing, experimenting with algorithms to provide the optimal use of the infrastructure, platform and applications with prioritized access to the CPU and efficiency for the end users.

Only a few years ago the term cloud computing didn't exist. There were other names to describe it such as virtualization, managed hosting, or even The Internet. Many researchers claim that the term cloud came from the cloud symbol that is frequently used to represent the Internet. On the other hand many others agree that the term cloud was first appeared at the telecommunication branch. At 1990s some companies started using virtual private network (VPN) services, instead of the dedicated point-to-point data circuits they used to offer. The quality was comparable and the cost was way lower. By switching traffic to balance the load, soon they were able to use their overall network bandwidth more effectively. The cloud symbol was used to separate the infrastructure that was provider's responsibility from where the user too over. Cloud computing is the same idea extended to cover servers and network infrastructures. The term cloud computing was first mentioned during a lecture by Ramnath Chellappa in 1997.



Generational Shift of Computing Platform

| | Technology | Economic | Business |
|---|---|---|---|
| | Centralized compute & storage, thin clients | Optimized for efficiency due to high cost | High upfront costs for hardware and software |
| | PCs and servers for distributed compute, storage, etc. | Optimized for agility due to low cost | Perpetual license for OS and application software |
| | Large DCs, commodity HW, scale-out, devices | Order of magnitude better efficiency and agility | Pay as you go, and only for what you use |

Figure 1 "Generational shift of cloud computing [1]"

Around 2000, after the Internet bubble, also known as dot-com bubble, Amazon played a key role in the development of cloud computing by updating their data centers. The new cloud architecture that was used led to the rapid improvement of the operation and development since new features could be added quicker and easier by small groups of people. Knowing that, they decided to provide cloud computing services to external customers so they launched the Amazon Web Services (AWS) in 2006. This provides online services for other web sites or client-side applications with a large computing capacity that is much faster and cheaper than building a physical server.

Eucalyptus, in 2008, was the first API-compatible open source platform for private cloud creation. The same year, OpenNebula, was the first open source software for creating private and hybrid clouds. In the meantime, IRMOS project focused their efforts on providing cloud solutions that guarantee the quality of services and allow interactive real time and multimedia applications enabling all their attributes, and this resulted to a real-time cloud environment.

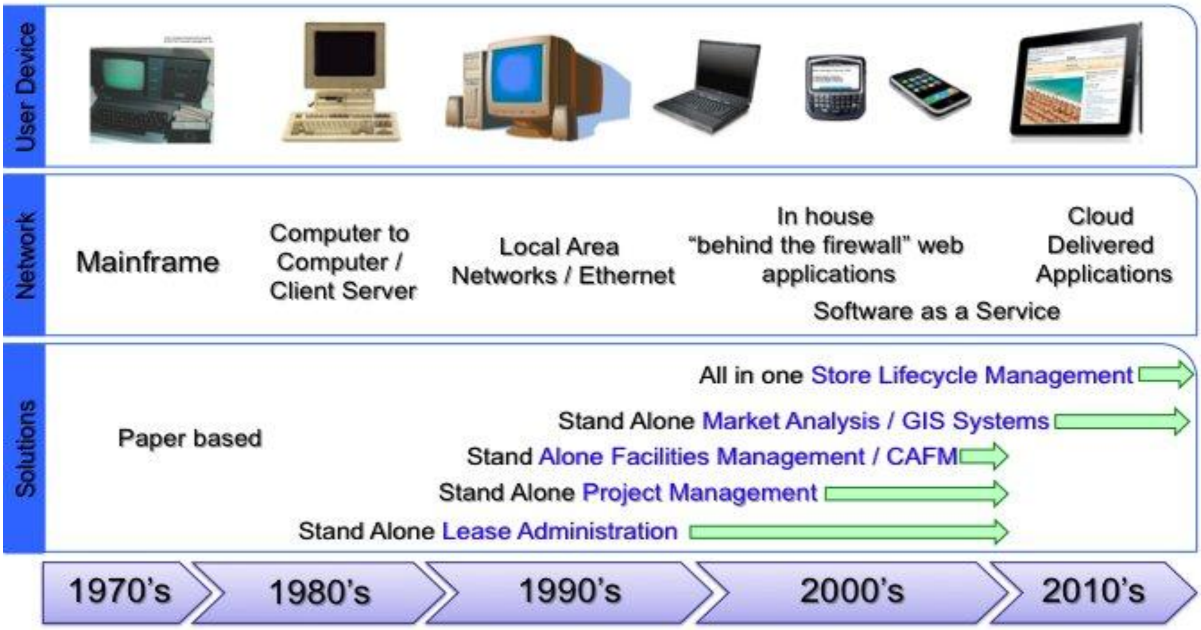

User Device
Network
Mainframe
Computer to Computer / Client Server
Local Area Networks / Ethernet
In house "behind the firewall" web applications
Cloud Delivered Applications
Software as a Service
Solutions
Paper based
All in one Store Lifecycle Management
Stand Alone Market Analysis / GIS Systems
Stand Alone Facilities Management / CAFM
Stand Alone Project Management
Stand Alone Lease Administration
1970's
1980's
1990's
2000's
2010's

Figure 2 "Cloud computing [2]"

Cloud computing is a very important part of Smarter Computing foundation. In 2011, IBM released the IBM SmartCloud to support Smarter Planet. The last years many

companies dealt with security, compliance, privacy, sustainability, abuse and evolution of cloud computing and it is still developing.

Nowadays there are many available clouds such as Amazon, Google, Microsoft EMC/VMware's Pivotal Initiative, Dot Cloud, Red hat, IBM, Linode, Engine Yard, Joyent, Bluelock, Salesforce.com, SoftLayer Technologies Inc., Verizon, CloudBees, ProfitBricks, Rackspace, Heroku, Savvis, CloudSigma, NaviSite, Citrix and Modern Infrastructure E-Zine.

### 1.1.2 FEATURES

Many companies instead of buying the dedicated hardware and depreciate it when needed, prefer to use a shared cloud infrastructure and pay for the usage. In general clouds are owned by large companies and are used by small or medium companies and researchers.

Cloud computing can be described as a group of integrated hardware and software that is accessible via the internet. The main technology that characterizes the cloud computing is virtualization. Virtualization abstracts the physical infrastructure and as a result, all the details and complexity of the underlying infrastructure are hidden from the end user, who can easily access, use and manage it through a simple user interface. By doing so, virtualization provides the agility to the users to focus on their projects, which means increased infrastructure usage and reduced cost. In addition, when the user wants to change the resources that are used, automated processes do all the work, reducing the possibility of human error and speeding up the process. In other words, virtualization means to separate the computer resources and place them in a lower layer, in order for applications and tasks to operate without a concern of those changes made in this lower layer. At this point, we need to mention that no virtualization means no cloud but cloud is not just virtualization. It goes far beyond that. We can have virtualization without a self-serving component, but in cloud computing self-service is an essential component. Self-service is the ability that the user has to add and remove cloud applications and change the infrastructure and configuration without requiring the assistance of a professional.

Finally, it is good to keep in mind that all cloud services have three characteristics that differentiate them from the traditional hosting:

- They are commercial (you can use them and pay for the usage on demand)
- They are flexible (you can use only what you need)
- They are fully managed by the cloud provider (you don't need complicated infrastructure, you just need a device with internet access)

### 1.1.3 DEFINITION

The definition of cloud computing is a little vague just like real clouds. The term "cloud computing" is a recent concept and therefore it has no single definition that has been accepted by all users. However, there are a few key features that are generally accepted as the core of cloud computing and the differences between most definitions are usually insignificant.

Since cloud is still developing there are many definitions by different researchers. Here are some of them:

Armbrust et al. 2009 states the following:

“Together with virtualization, clouds can be defined as computers that are networked anywhere in the world with the availability of paying the used clouds in a pay-per-use way, meaning that just the resources that are being used will be paid.”

In the same year, UC Berkeley RAD defines cloud as follows:

“Cloud Computing refers to both the applications delivered as services over the Internet and the hardware and systems software in the data centers that provide those services. The services themselves have long been referred to as Software as a Service (SaaS), so we use that term. The data center hardware and software is what we will call a Cloud. When a Cloud is made available in a pay-as-you-go manner to the public, we call it a Public Cloud; the service being sold is Utility Computing. We use the term Private Cloud to refer to internal data centers of a business or other organization that are not made available to the public. Thus, Cloud Computing is the sum of SaaS and Utility Computing, but does not normally include Private Clouds.”

National Institute of Standards and Technology (NIST) developed the following definition that is also known as the 5-3-4 definition of cloud computing:

“Cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g. networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. This cloud model promotes availability and is composed of five essential characteristics, three service models, and four deployment models.”

#### 1.1.3.1 ESSENTIAL CHARACTERISTICS

The five essential characteristics as described from NIST are:

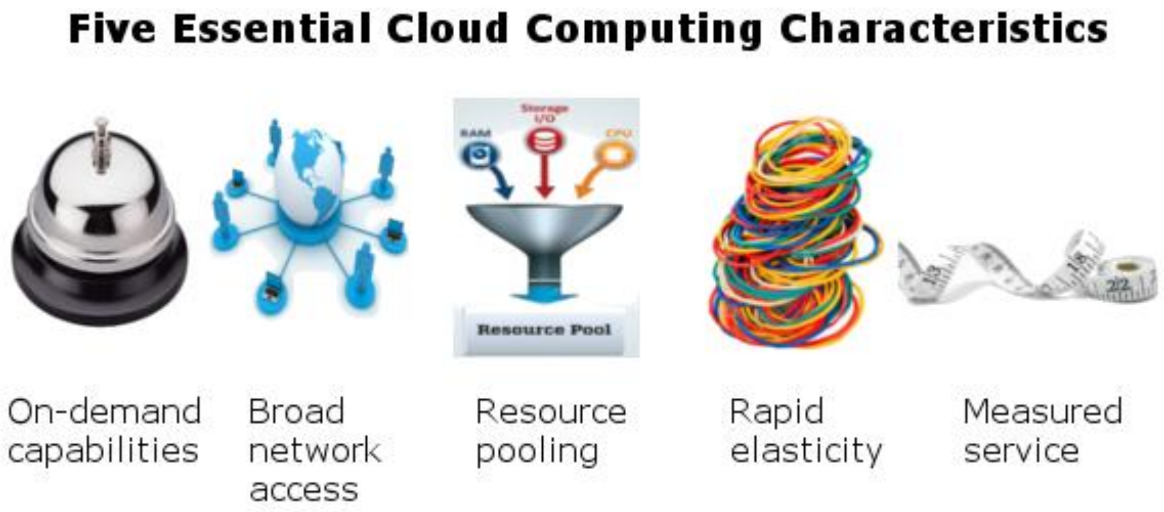


Figure 3 "The 5 essential characteristics of cloud computing"

- On-demand capabilities

  The cloud users can provision computing power on-demand, just like electricity. That means that you can adjust the resources that you use any time from any place. You can add or delete users and change storage, networks and software without requiring any human interaction. Usually, you are billed for only what you use.

- Broad network access

  You can access your data on cloud, using heterogeneous thin or thick client platforms such as computers, laptops, tablets or even mobile phones, wherever you are, as far as you have internet access. This mobility is particularly attractive for businesses so that during business hours or on off-times, employees can stay on top of projects, contracts, and customers whether they are on the road or in the office.

- Resource pooling

  Resource pooling can be described as an abstraction that helps users managing, categorizing, presenting and consuming the available resources. Generally, physical and virtual resources are pooled to serve multiple requests and many users can use the same set of resources at the same time. All the resources (storage, processing, memory, network bandwidth, and virtual machines) are dynamically assigned and reassigned on demand. The mechanism of processing power distribution, or the amount of memory, operates in such way that the system dynamically allocates these parameters according to customer requirements. Moreover, users have no control over the physical parameters. For example, they do not know where the resources are geographically located.

- Rapid elasticity

  The cloud is flexible and can be scaled up or down quickly, in order to meet the user's needs each time. You can easily add or remove users, software features, and other resources and all these seem to be unlimited. The scalability is a major advantage of cloud. For instance, let's pretend that a user has a website on a cloud and generally the traffic is almost the same every day, but today they had a unique offer or something else similar and the traffic was increased by 200%. If they had their own infrastructure, there is a strong possibility for this site to stop working because of the hardware and software limitations. On the other hand, cloud dynamically allocates all the necessary resources in order to make sure that site will be operational during the high load, and when traffic is back to normal, all the additional resources are removed automatically.

- Measured service

  This is a very important characteristic since cost is usually a problem when a company needs to update the existing infrastructure to include new technologies. Cloud is an affordable solution, as you pay only for what you use. It automatically controls and measures all the necessary resources depending on the needs of each user, and provides usage metrics transparently to both the provider and the user.

#### 1.1.3.2 SERVICE MODELS

The services are divided into three categories: Software-as-a-Service (SaaS), Platform-as-a-Service (PaaS) and Infrastructure-as-a-Service (IaaS).

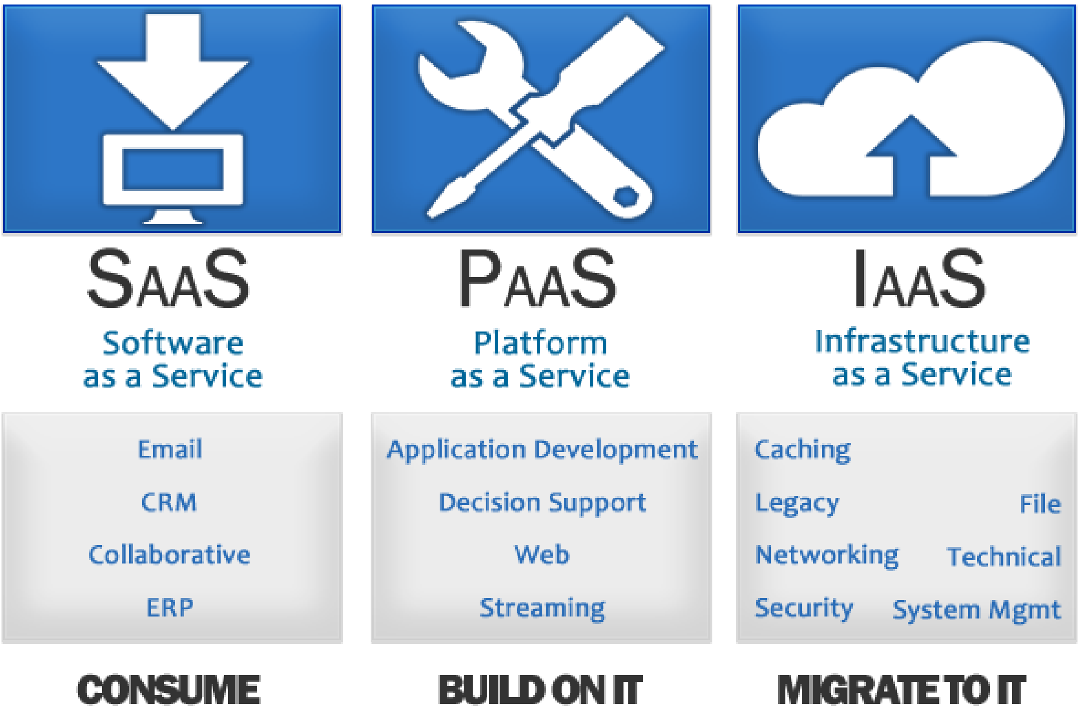


Figure 4 "Service Models [3]"

- Software-as-a-Service

  SaaS means on demand software. The main idea is that the user can use an application that is running on another system. Cloud providers install the application software on cloud and users just use it. This simplifies the procedure since the end user does not have to worry about software maintenance and upgrades and at the same time it eliminates the needs for hardware upgrades and compatibility errors. The cost is usually a standard monthly fee per user.

- Platform-as-a-Service

  PaaS means on demand platform. It provides the user with a run-time environment on demand. Usually it includes the operating system, the programming language execution environment, the database and the web server. Developers can create and test their programs without the cost of purchasing and maintain the hardware and software.

- Infrastructure-as-a-Service

IaaS means on demand infrastructure. User can get physical or virtual machines and other resources such as storage, and install operating system images and their application. In this model, users need to maintain their software but they do not need to purchase or maintain the hardware. Usually they are billed for what they used.

### 1.1.3.3 DEPLOYMENT MODELS

There are four deployment models.

- Private cloud

The private cloud infrastructure is used by a single business or organization, whether managed internally or by a third-party, hosted internally or externally and can have multiple users. Usually, it is on a private network and this restricts unwanted, public access. Obviously, it has higher security than other models, but the business has to purchase and maintain their own infrastructure. Commonly, we have private clouds when a business with existing hardware uses it for cloud usage. The virtualization of a business environment is not an easy procedure, but when completed, it highly improves

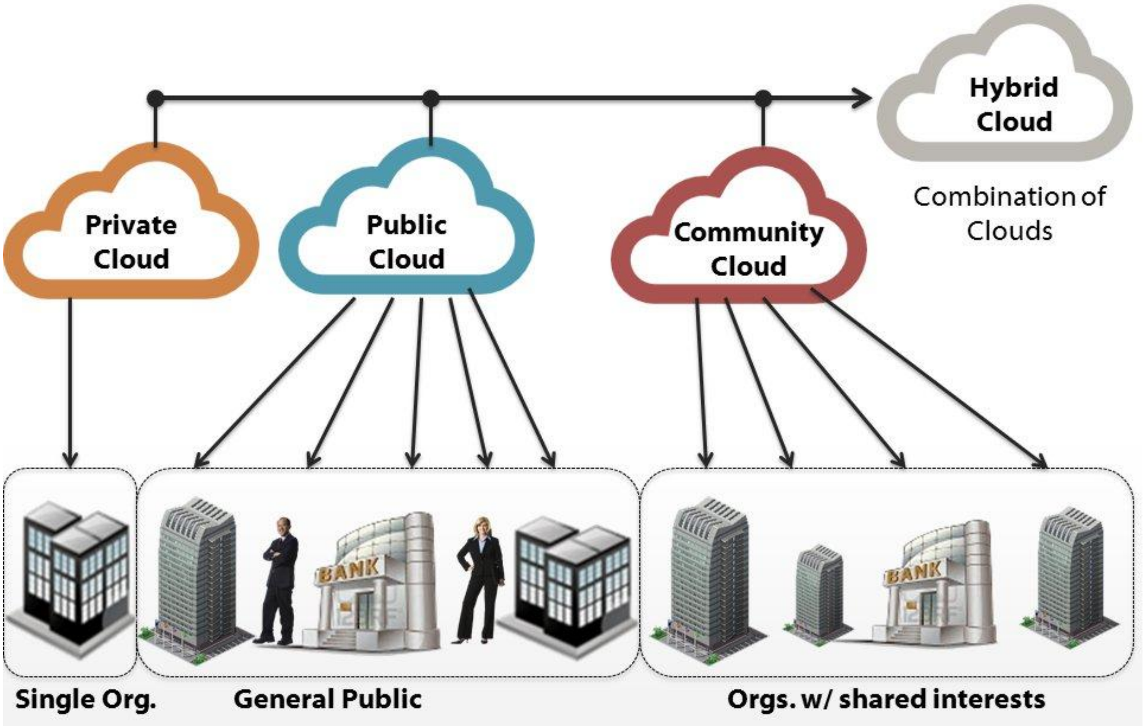


Figure 5 “Deployment models [4]”

both efficiency and productiveness. An example of a private cloud is the cloud at University of Iceland. The university owns a cluster that was set up as a grid, but this was limiting the software that could be installed without the need of an administrator. Cloud instances can be handled easier by the end users, who can also have their own software, so the university decided to set up Eucalyptus 16 IaaS software on the existing grid and now each student or faculty can get their computing instances where they can set up the software they need, any time, without the need of IT experts. That was a success scenario for the university, but many people state that the greatest advantage of cloud, the financial benefit, is lost, since you have to own or purchase the infrastructure, maintain and manage it.

- Community cloud

At community cloud we have cloud infrastructures that are shared between several businesses or organizations and their users have common concerns. As the private cloud, it can be managed internally or by a third-party, hosted internally or externally. An example here is the North European cloud computing project, or as it is commonly known, NEON 17. Many north European universities and research centers shared their computing facilities on a very large grid. The idea of this project is to create a large community cloud where universities and researches can share their work, ideas and knowledge. In order to make this happen, each university or center that takes part to this project, needs to change their infrastructure to a private cloud and then share all these private clouds, on one common community cloud.

- Public cloud

The public cloud infrastructure is available to general public, meaning that anyone on the internet can purchase their service. Actually, this is the most traditional type of cloud. All the users can use the computing resources from everywhere in the world and pay only for what they use. An example of public cloud is Amazon Elastic Compute Cloud (Amazon EC2) that provides resizable compute capacity with an easy interface. The cost is lower than in

Table 1 "Public and private cloud main characteristics"

| Characteristics | Public Cloud | Private Cloud |
|---|---|---|
| Initial cost | Almost zero | High |
| Customization | No | Yes |
| Privacy | No | Yes |
| Scalability | Yes | Yes, High Cost |

private or community cloud since the user does not have to purchase and/or maintain the infrastructure but he just uses the resources and pays for the usage. There are a lot of questions and doubts about security and privacy of the data but many steps have been made to overcome these issues.

- Hybrid cloud

  As its name reveals, the infrastructure of a hybrid cloud is a combination of two or more types of cloud. Each type of cloud remains a unique entity but the new hybrid cloud has the advantages of all combined types. For example, if a business has a private cloud with limited amount of resources, when the need of resources is more than the available ones, it can scale up temporarily to a public cloud to serve all the needs and then scale down back to private. The greater advantage here is financial because the business instead of purchasing extra resources for peak usage only can use the resources of the public cloud and pay for extra compute resources when they are needed and the data are hosted on the private cloud so the security level is higher than at a public cloud.

### 1.1.4 ADVANTAGES

There are many advantages that make cloud computing so popular and so widely used among businesses today. In small and medium sized businesses, there is often lack of time and financial resources and that hardens the purchase, deployment and maintenance of the infrastructure needed, such as software, server and storage. In cloud computing, businesses can access these resources using an Internet connection. They can expand or shrink the services and pay only for what they do use.

Here are some of the most important advantages of cloud:

- Less expensive

  Cloud computing helps companies avoid the upfront hardware cost that can be very high, depending on their needs. Since they don't have to build their own infrastructure, they have more time to focus on their projects so they can get their applications up and running faster.

  In addition, there is no need to have high-performance computers since the applications run in the cloud, not your computer. In fact you can use a computer with a small hard disk and memory that doesn't even have a CD or DVD drive, as no software programs will be loaded and no document files will be saved.

  Finally, usually the usage is metered with transparency and you pay only for what you use.

- Easily accessible

  All your data are always on the cloud. There is no need to transfer them from a place to another. You can access them from anywhere in the world. You just need to have a computer with internet access.

- Highly automated

  A common and hard decision for most companies is to choose between working with outdated software and high upgrade cost. With cloud there is no need to purchase the updated versions of the software you use. All the updates are automatically done for you.

- A limitless storage

  Even if your company has a huge volume of data, there is no need to worry, because cloud offers virtually unlimited storage. There are hundreds of Pbytes available there to be used.

- Flexible

The cloud is elastic, meaning that resource allocation can get bigger or smaller depending on demand. Elasticity enables scalability, which means that the cloud can scale up for peak demand and scale down for lighter demand. Scalability also means that an application can scale when adding users and when application requirements change.

It allows users to explore different options in order to find the best solution for their companies and they pay only for what they use. In addition, they have better manageability, less maintenance and can adjust their resources to meet their demands, even if this is something that they couldn’t predict from the beginning, without a lot of effort and in less time.

- Reliable

  Even if your personal computer crashes, all the data on the cloud will still be available, safe and accessible from another computer.

- Environmentally friendly

  In comparison with every business with its own hardware, cloud computing needs less energy since the same physical resources are used from many different companies.

- Easily maintained

  Application maintenance is easier because there is no need to install the application on each user’s computer. Each user is using a cloud instance and has access to the applications that are installed on the cloud.

### 1.1.5 SECURITY

Probably the greatest challenge a clouds provider has to deal with is security. The security and privacy of the clients own data and information depends on third-party. Moreover it is under discussion where all the data will go in case the cloud host disappears for any reason. The most common security issues have to do with the data access, recovery, accountability and malicious insiders. There are already multiple solutions but since cloud computing is still a developing and new technology, more effective ways of increasing security are being developed.

## 1.2 VIDEO

People always wanted to keep their memories alive and video is a very effective way to do this. Video is an illusion of still images that are displayed the one after the other creating a moving image and it also includes sound.

### 1.2.1 HISTORY

The first video tape recorder captured live images from television cameras in 1951. It was converting the camera's electrical impulses and saving the information on magnetic video tapes. The Ampex Quadruplex, the first successful video tape format was released in 1956 and it was when video recordings became portable. Of course the tapes were really wide and the docking bays huge. Over the next 30 years, several analog video formats were developed, including Betamax, JVC's, the HDTV cassettes that didn't last long and W-VHS. During that time digital storage was created in the form of digital cassettes, laserdiscs and DVDs, which all set the storage for modern digital formats.

Video compression first took place in 1984. The first digital video standard (H.120) was created by the International Telecommunication Union. Its bit rate was 2 Mbit/s and resolution 176x144. That was when researchers learned that in order to stream video at a usable bit rate, groups of pixels would have to be coded together. This led to future standards, which finally compressed video practically. In 1988 the International Telecommunication Union developed H.261, with a bit rate of maximum 2 Mbit/s and a resolution of 352x288. This was the first major digital video compresion standard, on which most video compression standards and codecs were originally based on. In 1991 we have the appearance of MPEG with a bit rate of 1.5 Mbit/s and a resolution of 352x288. A year later Microsoft releases AVI and assigns it to be its video container for Windows, but due to many limitations it was largely replaced by WMV. In 1994, MPEG-2 (H.262) is developed jointly with ITU. It offered much better resolution and higher bit rates and soon becomes the standard video codec used by DVDs and digital TV. Some years later, MPEG-4 and Hi-Def standards were developed. This meant better support for features like 3D rendering, digital rights management and high resolution. During this period other formats like DivX emerge based on MPEG-4(Part 2) standard. Also MPEG-4(part 10), H.264 and Microsoft's VC-1 emerge as standards that become part of Blue ray and HDTV.

### 1.2.2 NEWER VIDEO CONTAINERS

Some more recent video containers-formats are described below

- VOB (Video Object)

  It is the container format for DVDs. Based on MPEG it can contain video, audio, subtitles and menus.

- ASF (Advanced Systems Formats)

  ASF is Microsoft's competitor container format to QuickTime(MOV) and OGG. Specifically designed for streaming media, it is the container for WMV, which has become a standard for High-Definition and Blue-Ray videos.

- MP4 (MPEG)

  MP4 is a standard container for streaming MPEG-4(Part 14) videos. It was based directly on QuickTimes format, but provides support for additional MPEG features.

- FLV (Flash Video)

  FLV was first released in 2003 as an outgrowth of the video features of Flash SWF files. It is used to deliver MPEG video through Flash Player, and has been accepted as the default video format for many websites. Apple IOS devices don't support Flash or FLV.

- MOV (QuickTime)

  It is a format created by Apple as a modern answer to AVI, mostly for desktop videos. It can separate data from track edit lists and media offsets, which makes it capable of importing and editing in place.

- 3GP (Third Generation Partnership)

  It is used on mobile devices, such as 3G phones. 3GP was developed by the 3rd Generation Partnership Project as part of an effort to set specifications and standards for mobile telecommunication.

### 1.2.3 DEFINITION OF TERMS

First of all, it is important to state the difference between video formats, containers and codecs. A video format consists of both a video's container and its codec. The container describes the structure of the file and consists of information like where the various pieces are stored, how they are interleaved and which codecs are used by which video and its components. The container is usually identified by the file's extension. The codec is a way of encoding the video into a stream. It is the method used to encode the video and it usually has a major impact on the video's quality. Below there is a description of the most important characteristics of a video.

- Aspect Ratio

  It describes the dimensions of video screens and video picture elements. Since all popular video formats are rectilinear, they can be described by a ratio between width and height. Common aspect ratios are 5:4 (common with 1280x1024 displays), 4:3 (Old computer monitor standard), 16:10 (common with 1280x800 and 1680x1050 displays) and 16:9 (HD video standard.), which became the most common aspect ratio for computer displays in 2012.

- Frames per second (FPS)

  It is the frequency at which the consecutive pictures of a video are being produced. The standard value for FPS is 29.97. The higher this value is the smoother will be the image.

- Bit rate

  Bit rate is a measurement of the number of bits that are transmitted over a set length of time. It is quantified using the bits per second (bps) unit and higher bit rate means higher video quality.

### 1.2.4 TRANSCODING

Video encoding is the process of converting a video file from a non lossy format to a lossy format. The term transcoding is used when we convert a video file from a lossy format to a lossier one. A digital media file generally consists of a container with metadata information like the dimension and duration of the file, along with any number of tracks, usually video and audio. Each of these tracks has been encoded into a format that tries to maximize quality and minimize size. A transcoder will take these tracks from the container, decode them, filter them and multiplex the new ones in another container. Transcoding is usually used to convert high quality original videos to widely used formats that are easily sent over the internet, like H.264/AAC in an MP4 container. Also, by transcoding a video file to a specific container we can make it playable by a greater number of platforms. What also needs to be stated here is that since transcoding is a lossy process, it always lowers quality.

# 2. MY PRIVATE CLOUD

## INTRODUCTION

This chapter is about all the resources used (software and hardware) for this project, as well as the steps it went through to get from bare metal to the final stage of our cloud. That includes the technical features of the server, detailed instructions including all the prerequisites and finally a presentation of all the VMs used in addition to the resources distributed to them.

## 2.1 TECHNICAL FEATURES

### 2.1.1 HARDWARE

The hardware specifications of our system are the following:

- Processor: Intel(R) Core(TM)2 Quad CPU Q6600 @ 2.40GHz
- Installed memory (RAM): 4,00 GB
- System type: 64-bit Operating System, x64-based processor
- Hard drive: WDC WD5003AZEX-00K1GA0 ATA Device 500GB
- Network adapter (NIC): Atheros L1 Gigabit Ethernet 10/100/1000Base-T Controller

### 2.1.2 SOFTWARE

The software installed on the server for the whole implementation is the following:

- Windows  Server 2012 Standard 64-bit

  It's the operating system installed on the server.

- Ubuntu 12.04 LTS 32-bit

  It's the operating system installed in the three VMs.

- SQL Server 2012 standard SP1 64-bit

  It's installed in the Server as a prerequisite for SCVMM

- Hyper-V

  Hyper-V is a server hypervisor that enables consolidation of a single physical server into many virtual machines, all sharing the hardware resources of the host and are powered by Hyper-V.

  It isolates an assigned part of the physical server into child partitions and allocates them to different guest operating systems running in Hyper-V virtual machines. The operating system of the server acts as the primary host. Since the created virtual machines don't have direct access to the hardware resources of the host, Hyper-V can distribute the hardware and software resources needed to them. Only the installable role for Windows Server 2012 is needed for our implementation.

- SCVMM 2012 SP1 64-bit

  System Center Virtual Machine Manager (SCVMM) forms part of Microsoft's System Center suite. It enables the creation and management of private clouds made up from Hyper-V, VMware and XenServer virtualization hosts. It provides increased physical server utilization by making possible simple and fast consolidation on virtual infrastructure. SCVMM enables rapid provisioning of new virtual machines by the administrator and end users using a self-service provisioning tool. Finally, it provides the central management console to manage all the building blocks of a virtualized data center.

## 2.2 SETTING UP THE CLOUD

In this chapter you will see how a private cloud consisted of three hyper-v virtual machines can be created using System Center Virtual Machine Manager 2012 SP1. The operating system used is Windows Server 2012. First of all, we will present all the prerequisites that are needed for this attempt. After that we will describe in detail the steps of installing SCVMM 2012 SP1, setting up the fabric (storage and network) and creating the virtual machines. Finally, we will go through the properties of the VMs.

### 2.2.1 PREREQUISITES

This chapter is about all the requirements, hardware and software, needed for the whole implementation of this project. SCVMM consists of a number of separate components working together, each of whose system requirements will be described separately.

#### 2.2.1.1 SYSTEM REQUIREMENTS

Virtual Machine Manager (VMM) Server.

VMM can be scaled to manage huge clouds with more than 150 hosts. The system requirements in this case are higher. The processor should be at least 3.6GHz and minimum RAM is 4GB and recommended would be 8GB. On top of that, the SQL Server and the Library share should be placed on another server/cluster.

- VMM Console

  Logging directly to the VMM Management Server is not the best way to manage VMM. It is highly advisable to use the VMM Console, which should be installed on the administrators' machines. The system requirements for the console are:

Table 2 "System requirements for a VMM server"

| Hardware component | Minimum | Recommended |
|---|---|---|
| **Processor** | Pentium 4, 2 GHz (x64) | Dual-Processor, Dual-Core, 2.8 GHz (x64) or greater |
| **RAM** | 2 GB | 4 GB |
| **Hard disk space - without a local VMM database** | 2 GB | 40 GB |
| **Hard disk space - with a local, full version of Microsoft SQL Server** | 80 GB | 150 GB |

Table 3 "System requirements for a single VMM console"

| Hardware component | Minimum |
|---|---|
| **Processor** | Pentium 4, 1 GHz or greater |
| **RAM** | 2 GB |
| **Hard disk space** | 2 GB |

#### 2.2.1.2 SOFTWARE REQUIREMENTS

System Center Virtual Machine Manager (VMM) 2012 SP1 supports a wide variety of configurations. Therefore, the full list of supported software is quite long. For that reason, only the software requirements for our implementation are going to be described in this chapter.

- VMM Server and console
- Windows Server 2012 Standard or Datacenter
- WinRM enabled and running (in Windows Server 2012 it is by default)
- .NET Framework 4 or 4.5 (4.5 is installed in our server)
- Windows Assessment and Deployment Kid (ADK)
- SQL Server 2008 R2 64-bit SP1 or SP2 Standard, Enterprise or Datacenter or SQL Server 2012 SP1 64-bit Standard or Enterprise (the second one is installed in our server)
- Hyper-V Hosts

There should be a baseboard management controller (BMC) with a supported out-of-band management protocol. VMM supports the following out-of-band management protocols:

- Intelligent Platform Management Interface (IPMI) versions 1.5 or 2.0

- Data Center Management Interface (DCMI) version 1.0
- System Management Architecture for Server Hardware (SMASH) version 1.0 over WS-Management (WS-Man)
- Custom protocols such as Integrated Lights-Out (iLO).

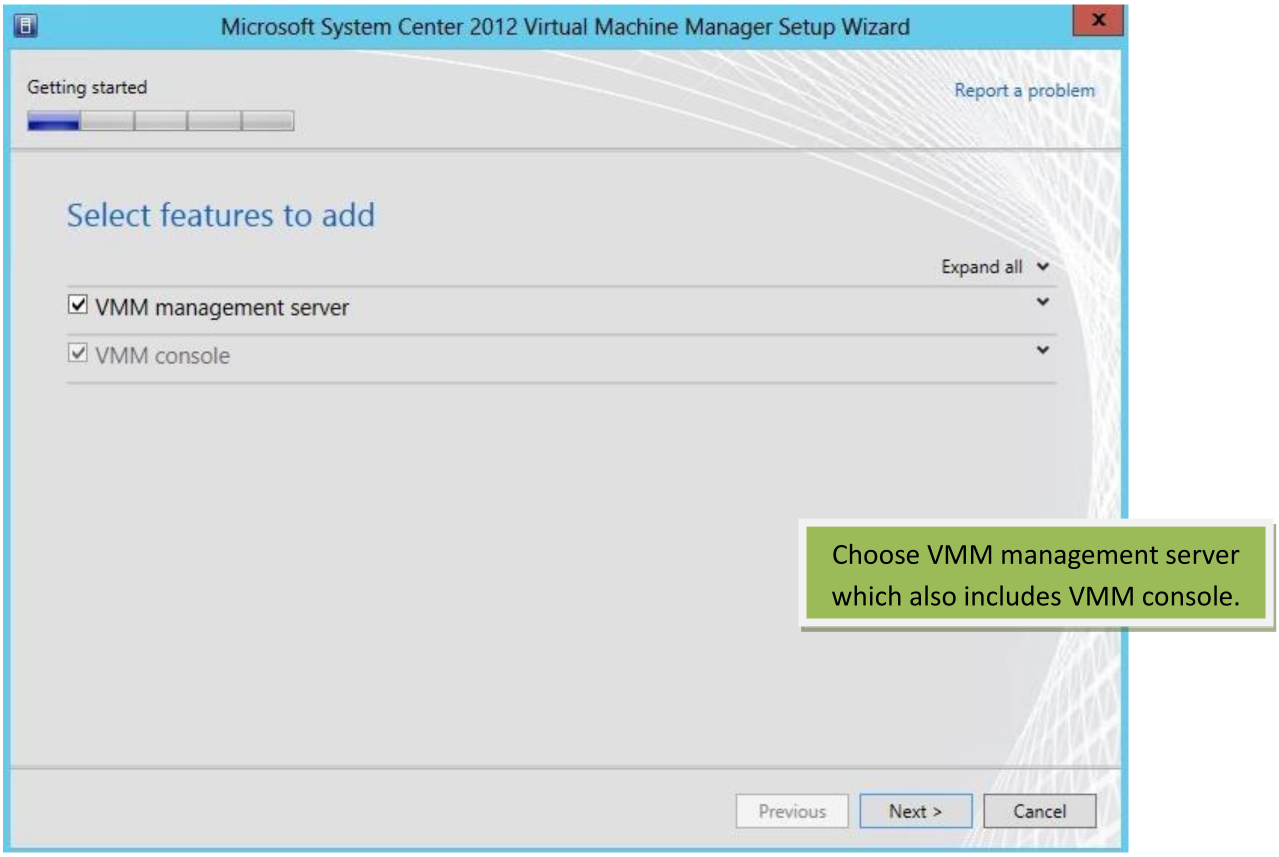
Networking and configuring (static IP address and administrator credentials) the BMCs in your Hyper-V hosts (current and future) is strongly recommended. VMM can make use of the BMC to deploy Hyper-V to bare metal hosts and to perform a power saving feature called Power Optimization (shutdown and power up idle hosts coordinated with VM Live Migration).

In addition a number of VMware vCenter and ESXi, as well as, Citrix XenServer 6.0 are supported but these are not going to be presented here since we make no use of them.

### 2.2.2 SCVMM 2012

Below are the steps followed for the installation of SCVMM 2012.

Note: Although System Center is not freeware, there is a trial version available at: http://www.microsoft.com/en-us/server-cloud/system-center/default.aspx



Microsoft System Center 2012 Virtual Machine Manager Setup Wizard

Getting started

Report a problem

Select features to add

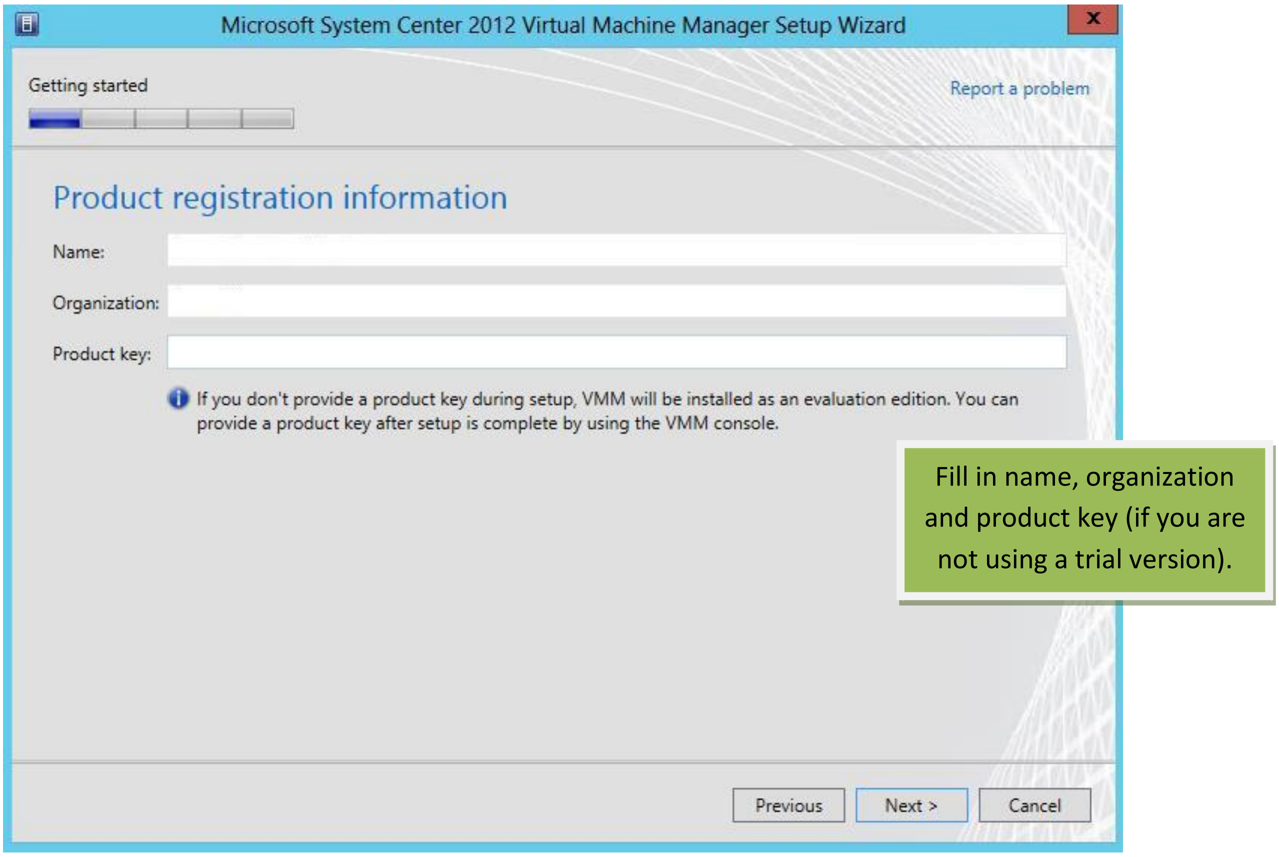
Expand all

☑ VMM management server

☑ VMM console

Choose VMM management server which also includes VMM console.

Figure 6 "Step 1: SCVMM 2012 installation"



Microsoft System Center 2012 Virtual Machine Manager Setup Wizard

Getting started

Report a problem

Product registration information

Name:

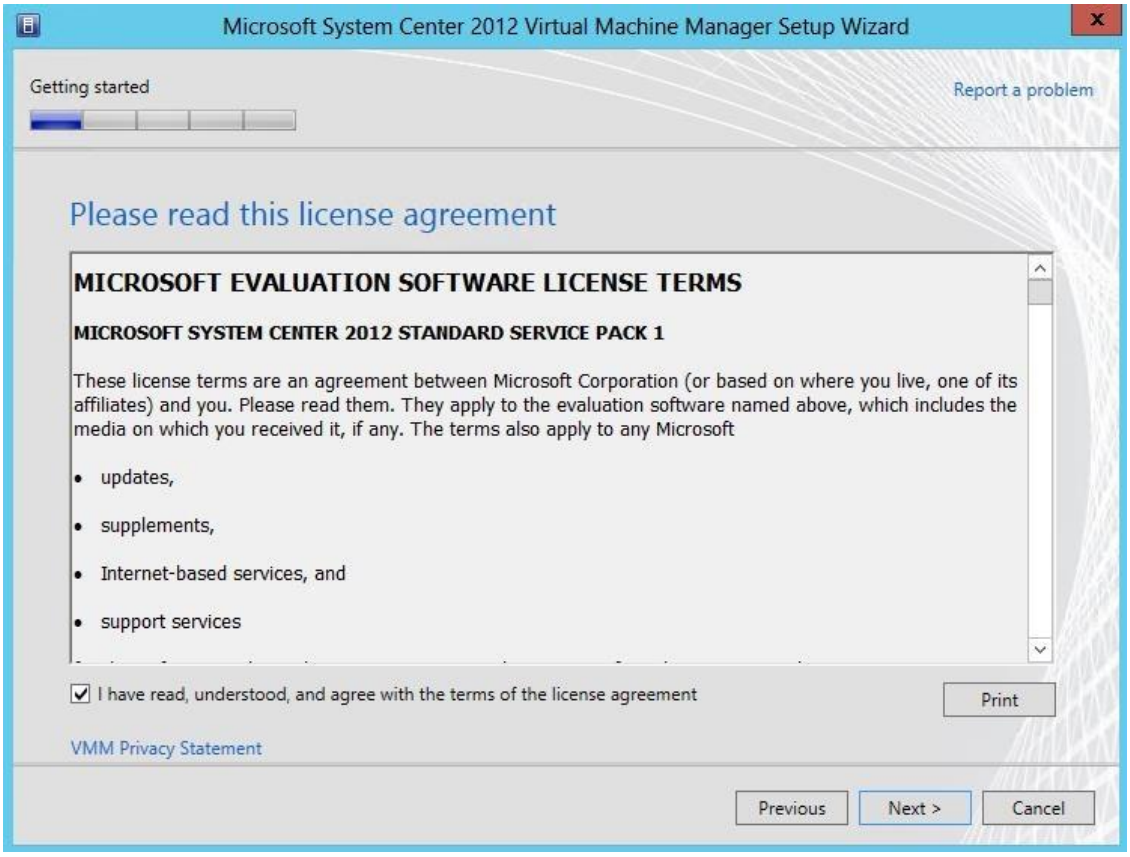
Organization:

Product key:

If you don't provide a product key during setup, VMM will be installed as an evaluation edition. You can provide a product key after setup is complete by using the VMM console.

Fill in name, organization and product key (if you are not using a trial version).

Figure 7"Step 2: SCVMM 2012 installation"


Microsoft System Center 2012 Virtual Machine Manager Setup Wizard
Getting started
Report a problem

Please read this license agreement

**MICROSOFT EVALUATION SOFTWARE LICENSE TERMS**

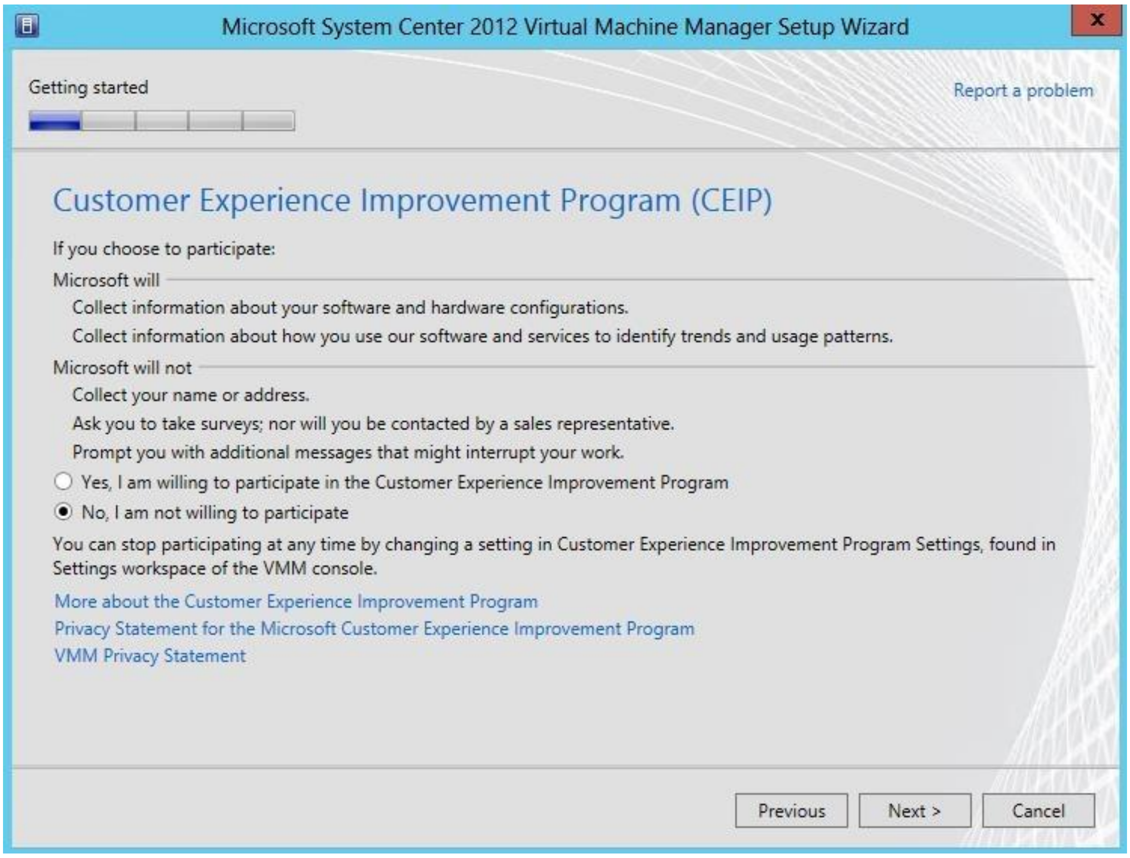
**MICROSOFT SYSTEM CENTER 2012 STANDARD SERVICE PACK 1**

These license terms are an agreement between Microsoft Corporation (or based on where you live, one of its affiliates) and you. Please read them. They apply to the evaluation software named above, which includes the media on which you received it, if any. The terms also apply to any Microsoft

- updates,
- supplements,
- Internet-based services, and
- support services

☑ I have read, understood, and agree with the terms of the license agreement

Figure 8 “Step 3: SCVMM 2012 installation”


Microsoft System Center 2012 Virtual Machine Manager Setup Wizard
Getting started
Report a problem

Customer Experience Improvement Program (CEIP)

If you choose to participate:

Microsoft will

Collect information about your software and hardware configurations.

Collect information about how you use our software and services to identify trends and usage patterns.

Microsoft will not

Collect your name or address.

Ask you to take surveys; nor will you be contacted by a sales representative.

Prompt you with additional messages that might interrupt your work.

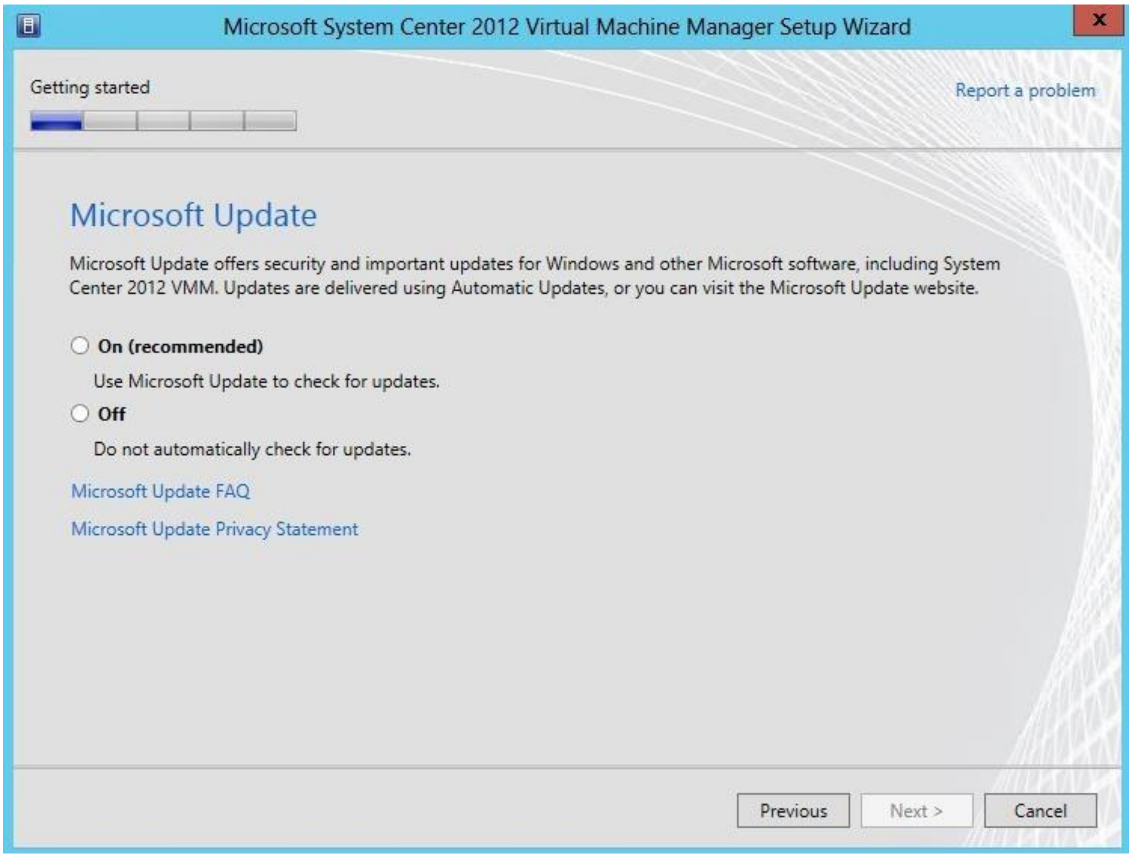
○ Yes, I am willing to participate in the Customer Experience Improvement Program

● No, I am not willing to participate

You can stop participating at any time by changing a setting in Customer Experience Improvement Program Settings, found in Settings workspace of the VMM console.

More about the Customer Experience Improvement Program
Privacy Statement for the Microsoft Customer Experience Improvement Program
VMM Privacy Statement
Previous
Next >
Cancel

Figure 9 “Step 4: SCVMM 2012 installation”



Microsoft System Center 2012 Virtual Machine Manager Setup Wizard
Getting started
Report a problem

## Microsoft Update

Microsoft Update offers security and important updates for Windows and other Microsoft software, including System Center 2012 VMM. Updates are delivered using Automatic Updates, or you can visit the Microsoft Update website.

○ **On (recommended)**

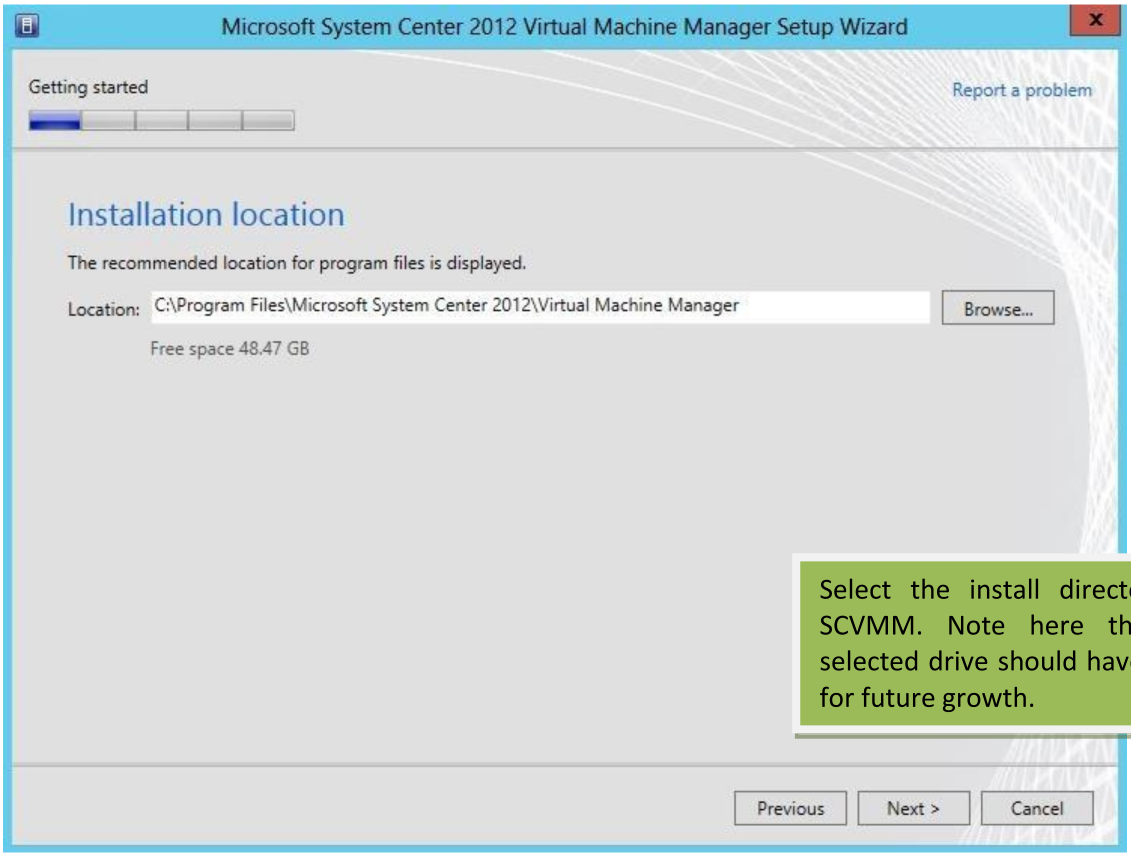
Use Microsoft Update to check for updates.

○ **Off**

Do not automatically check for updates.

Microsoft Update FAQ
Microsoft Update Privacy Statement
Previous
Next >
Cancel

Figure 10 “Step 5: SCVMM 2012 installation”



Microsoft System Center 2012 Virtual Machine Manager Setup Wizard
Getting started
Report a problem

## Installation location

The recommended location for program files is displayed.

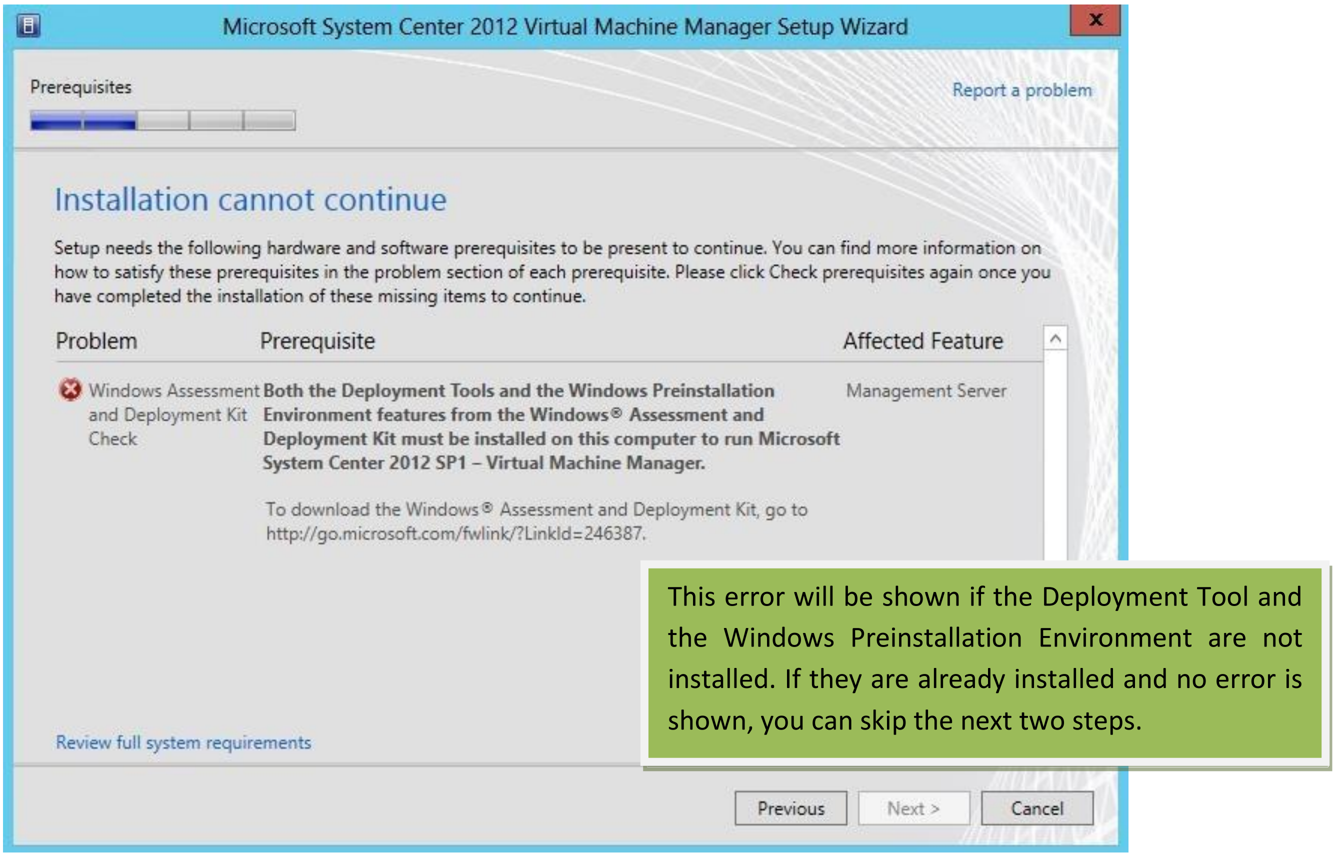
Location: C:\Program Files\Microsoft System Center 2012\Virtual Machine Manager

Free space 48.47 GB

Browse...
Previous
Next >
Cancel

Select the install directory for SCVMM. Note here that the selected drive should have space for future growth.

Figure 11 “Step 6: SCVMM 2012 installation”



Microsoft System Center 2012 Virtual Machine Manager Setup Wizard

Prerequisites

Report a problem

Installation cannot continue

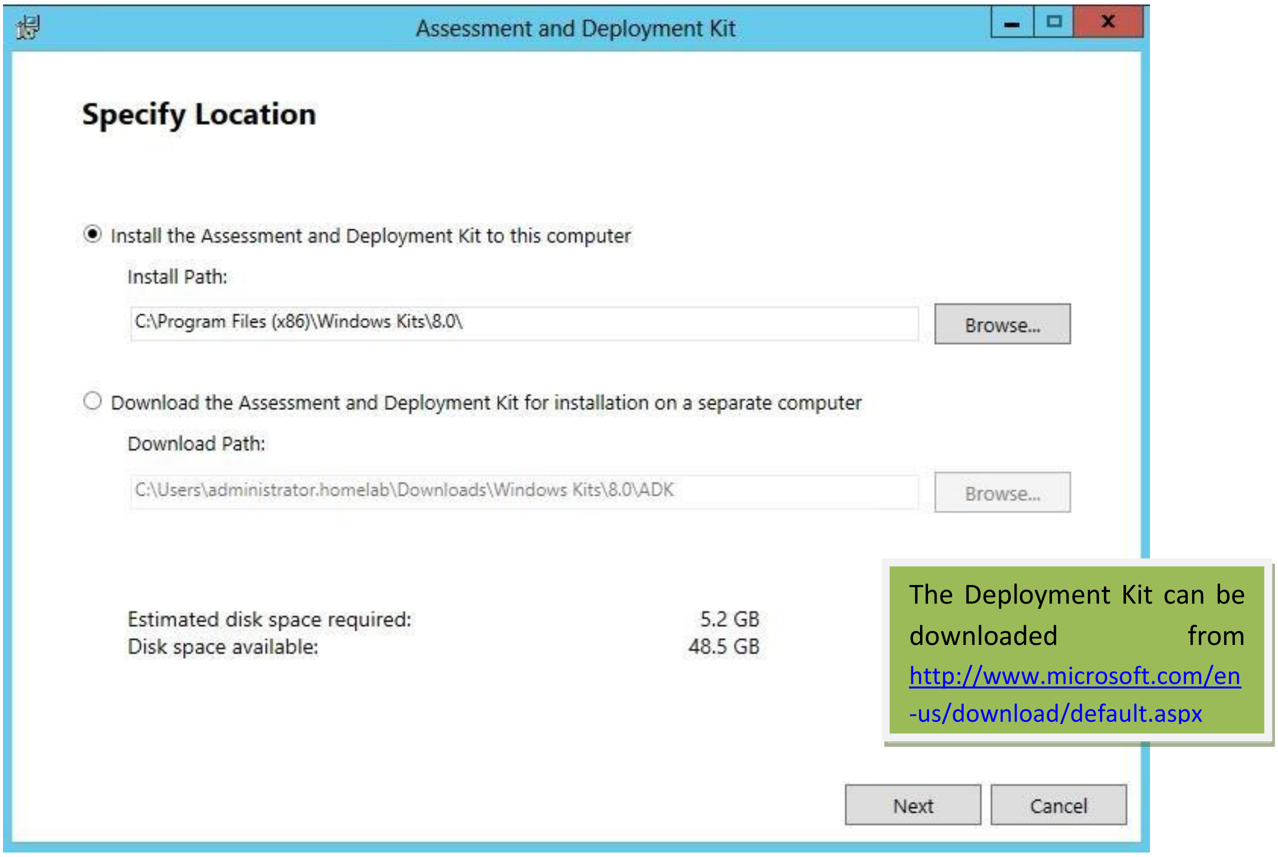
Setup needs the following hardware and software prerequisites to be present to continue. You can find more information on how to satisfy these prerequisites in the problem section of each prerequisite. Please click Check prerequisites again once you have completed the installation of these missing items to continue.

| Problem | Prerequisite | Affected Feature |
|---|---|---|
| ⊗ Windows Assessment and Deployment Kit Check | **Both the Deployment Tools and the Windows Preinstallation Environment features from the Windows® Assessment and Deployment Kit must be installed on this computer to run Microsoft System Center 2012 SP1 – Virtual Machine Manager.**<br>To download the Windows® Assessment and Deployment Kit, go to http://go.microsoft.com/fwlink/?LinkId=246387. | Management Server |

Review full system requirements

This error will be shown if the Deployment Tool and the Windows Preinstallation Environment are not installed. If they are already installed and no error is shown, you can skip the next two steps.

Figure 12 “Step 7: SCVMM 2012 installation”



Assessment and Deployment Kit

Specify Location

● Install the Assessment and Deployment Kit to this computer

Install Path: C:\Program Files (x86)\Windows Kits\8.0\

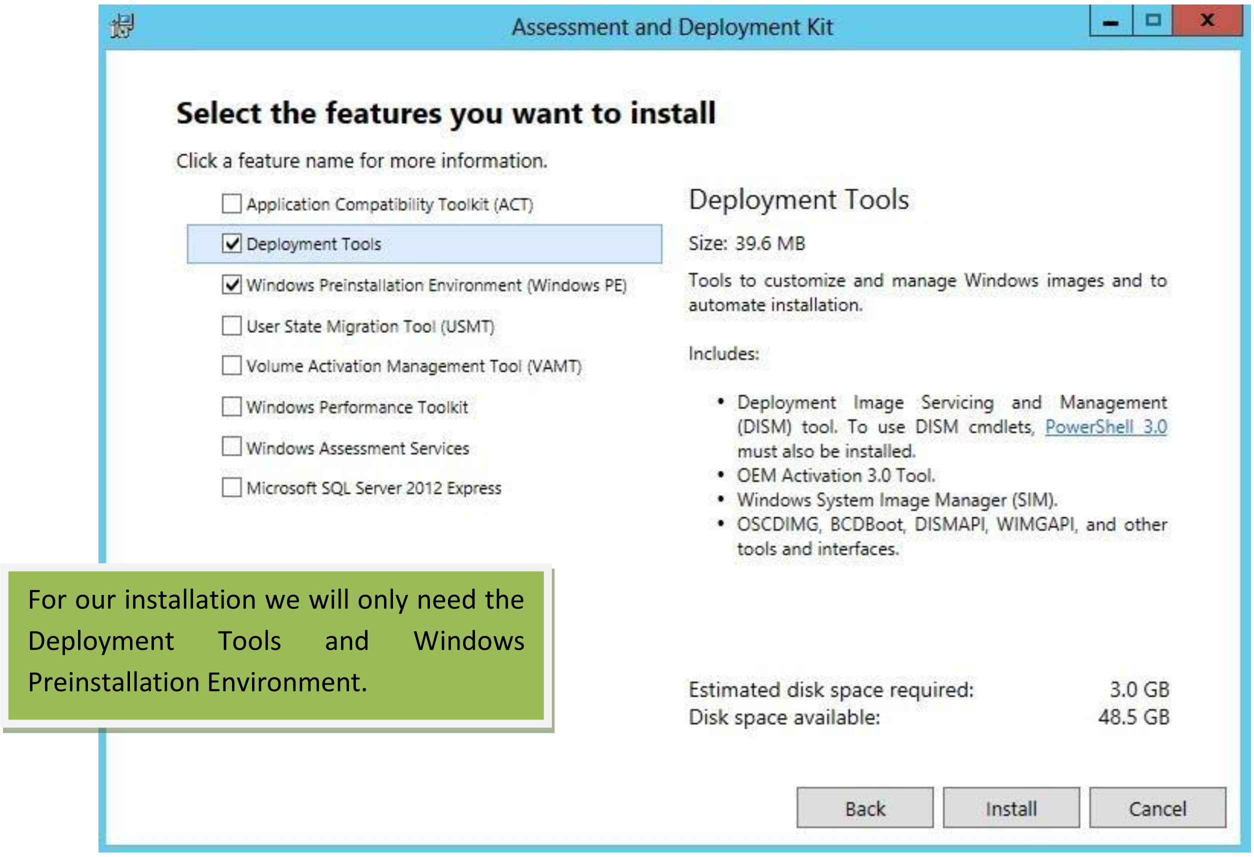
○ Download the Assessment and Deployment Kit for installation on a separate computer

Download Path: C:\Users\administrator.homelab\Downloads\Windows Kits\8.0\ADK

| | |
|---|---|
| Estimated disk space required: | 5.2 GB |
| Disk space available: | 48.5 GB |

The Deployment Kit can be downloaded from http://www.microsoft.com/en-us/download/default.aspx

Figure 13 “Step 8: SCVMM 2012 installation”



**Select the features you want to install**

Click a feature name for more information.

- ☐ Application Compatibility Toolkit (ACT)
- ☑ Deployment Tools
- ☑ Windows Preinstallation Environment (Windows PE)
- ☐ User State Migration Tool (USMT)
- ☐ Volume Activation Management Tool (VAMT)
- ☐ Windows Performance Toolkit
- ☐ Windows Assessment Services
- ☐ Microsoft SQL Server 2012 Express

Deployment Tools

Size: 39.6 MB

Tools to customize and manage Windows images and to automate installation.

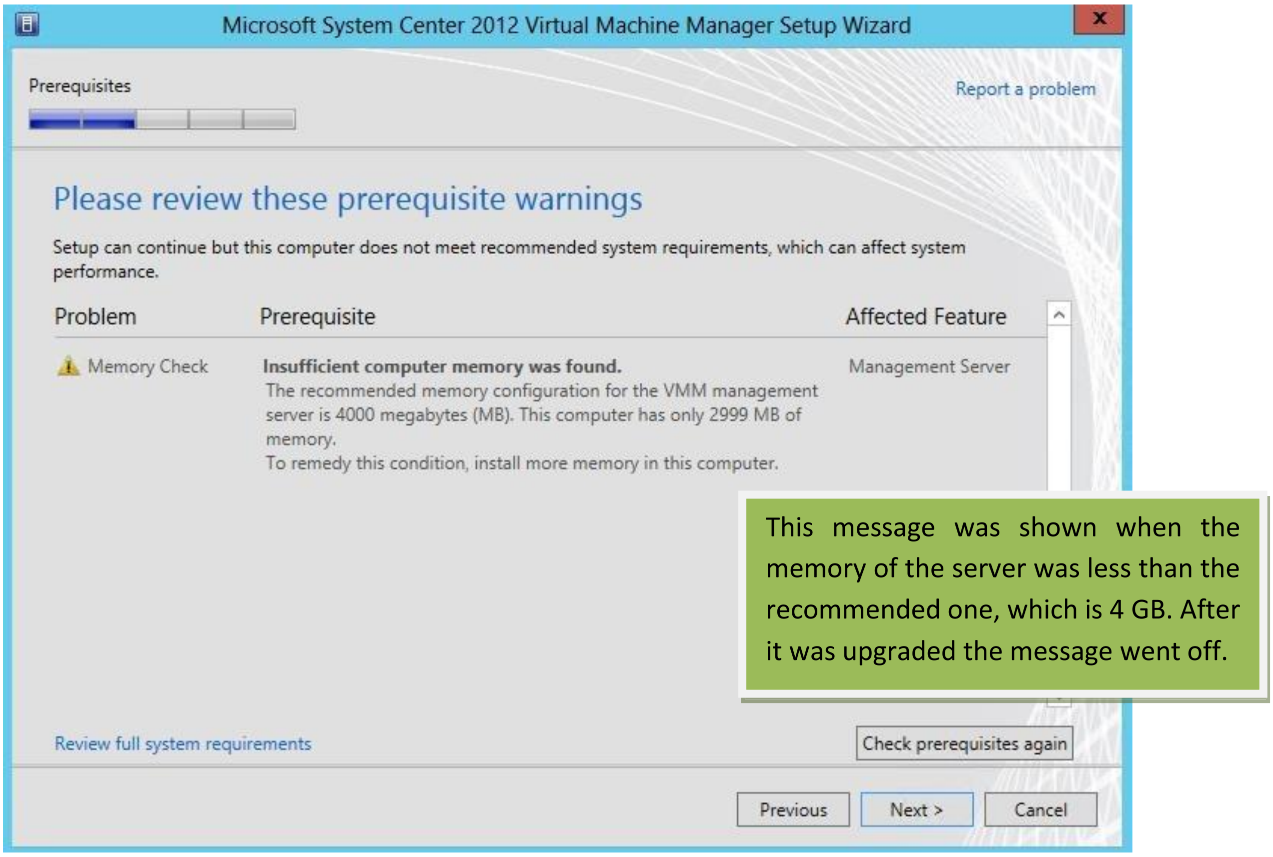
Includes:

- Deployment Image Servicing and Management (DISM) tool. To use DISM cmdlets, PowerShell 3.0 must also be installed.
- OEM Activation 3.0 Tool.
- Windows System Image Manager (SIM).
- OSCDIMG, BCDBoot, DISMAPI, WIMGAPI, and other tools and interfaces.

| | |
|---|---|
| Estimated disk space required: | 3.0 GB |
| Disk space available: | 48.5 GB |

For our installation we will only need the Deployment Tools and Windows Preinstallation Environment.

Figure 14 “Step 9: SCVMM 2012 installation”


Microsoft System Center 2012 Virtual Machine Manager Setup Wizard
Prerequisites
Report a problem

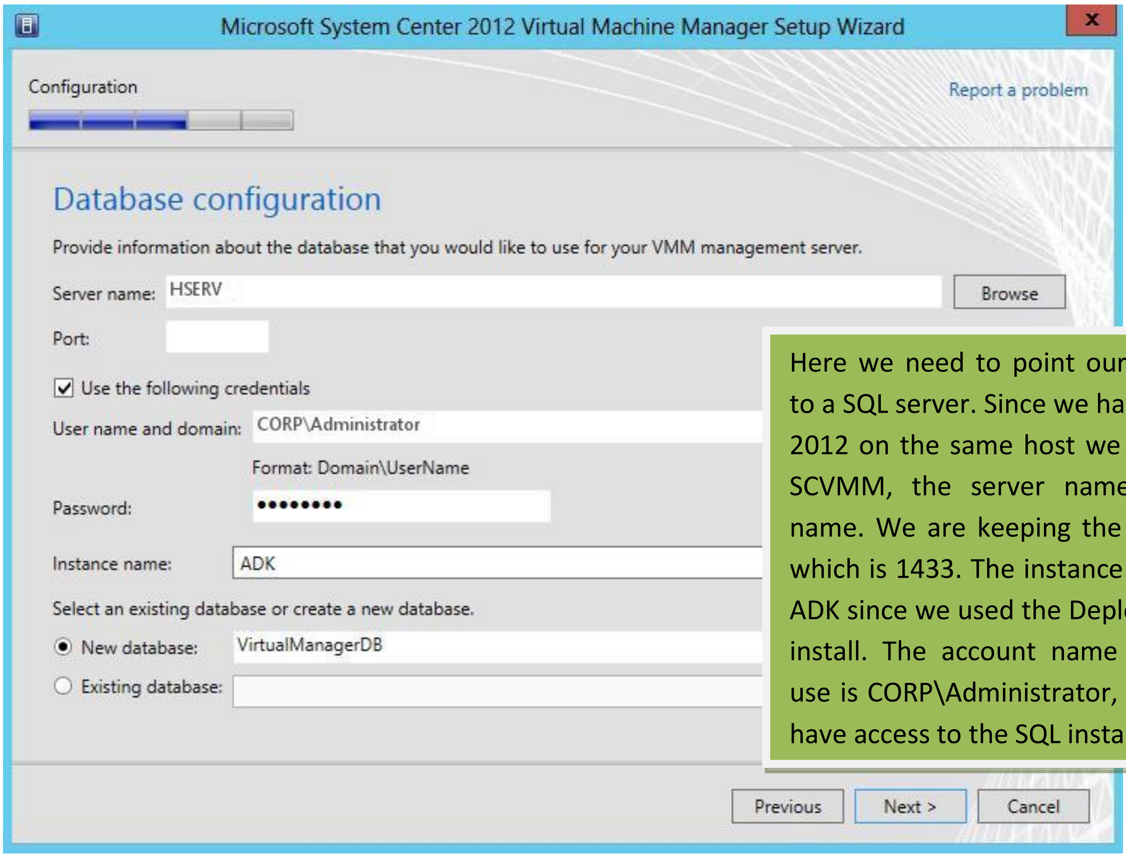
Please review these prerequisite warnings

Setup can continue but this computer does not meet recommended system requirements, which can affect system performance.

| Problem | Prerequisite | Affected Feature |
|---|---|---|
| Memory Check | **Insufficient computer memory was found.**<br>The recommended memory configuration for the VMM management server is 4000 megabytes (MB). This computer has only 2999 MB of memory.<br>To remedy this condition, install more memory in this computer. | Management Server |

This message was shown when the memory of the server was less than the recommended one, which is 4 GB. After it was upgraded the message went off.

Figure 15 “Step 10: SCVMM 2012 installation”



Microsoft System Center 2012 Virtual Machine Manager Setup Wizard

Configuration

Report a problem

Database configuration

Provide information about the database that you would like to use for your VMM management server.

Server name: HSERV [Browse]

Port:

☑ Use the following credentials

User name and domain: CORP\Administrator

Format: Domain\UserName

Password: ••••••••

Instance name: ADK

Select an existing database or create a new database.

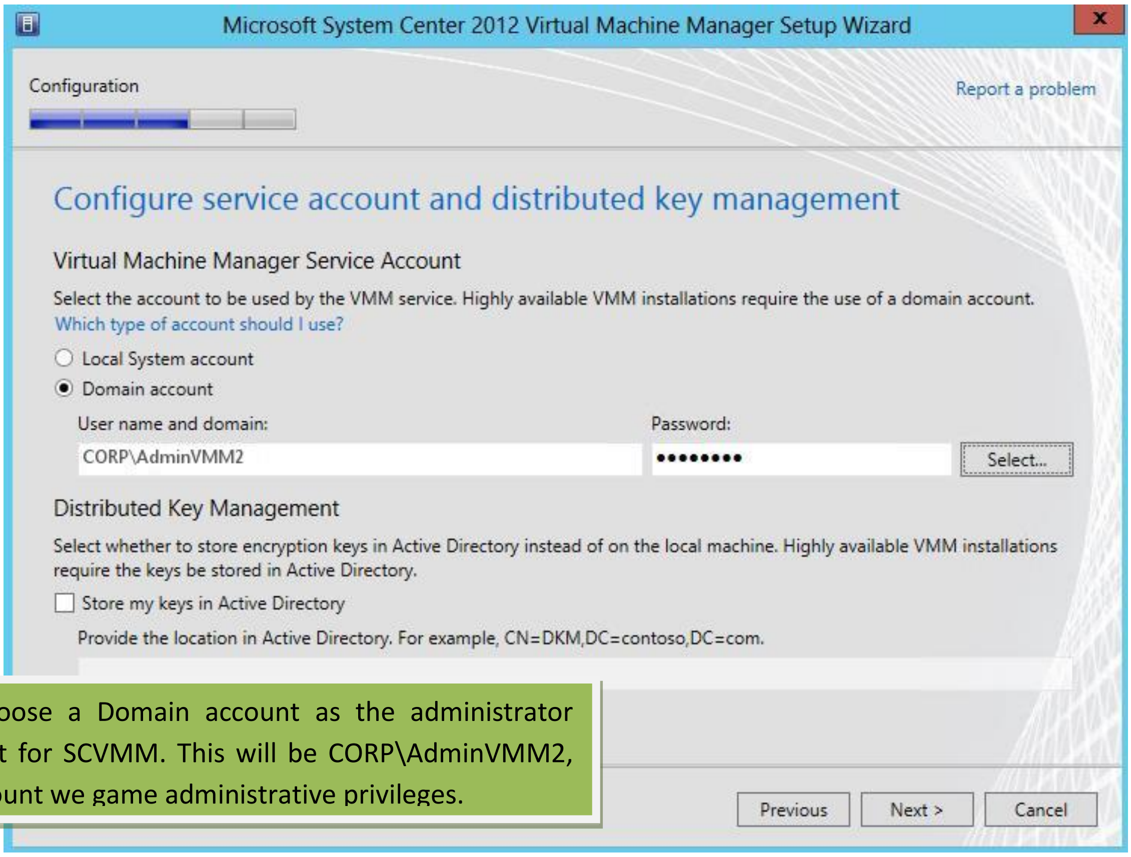
● New database: VirtualManagerDB

○ Existing database:

[Previous] [Next >] [Cancel]

Here we need to point our VMM install to a SQL server. Since we have SQL server 2012 on the same host we are installing SCVMM, the server name will be its name. We are keeping the default port, which is 1433. The instance name will be ADK since we used the Deployment Kit to install. The account name that we will use is CORP\Administrator, which should have access to the SQL instance.

Figure 16 “Step 11: SCVMM 2012 installation”



Microsoft System Center 2012 Virtual Machine Manager Setup Wizard

Configuration

Report a problem

Configure service account and distributed key management

Virtual Machine Manager Service Account

Select the account to be used by the VMM service. Highly available VMM installations require the use of a domain account.
Which type of account should I use?

○ Local System account

● Domain account

User name and domain: CORP\AdminVMM2

Password: •••••••• [Select...]

Distributed Key Management

Select whether to store encryption keys in Active Directory instead of on the local machine. Highly available VMM installations require the keys be stored in Active Directory.

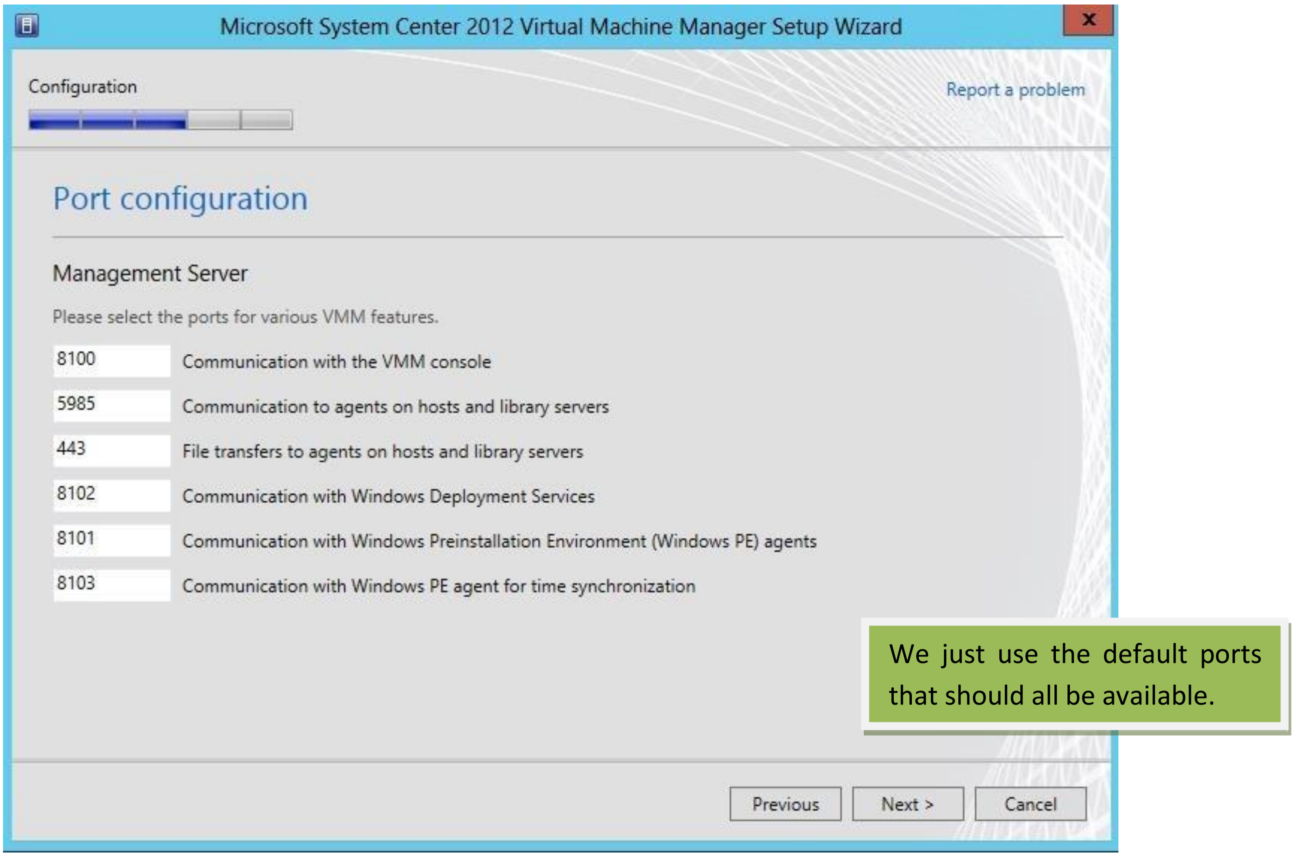
□ Store my keys in Active Directory

Provide the location in Active Directory. For example, CN=DKM,DC=contoso,DC=com.

[Previous] [Next >] [Cancel]

We choose a Domain account as the administrator account for SCVMM. This will be CORP\AdminVMM2, an account we game administrative privileges.

Figure 17 “Step 12: SCVMM 2012 installation”



Microsoft System Center 2012 Virtual Machine Manager Setup Wizard
Configuration
Report a problem

## Port configuration

Management Server

Please select the ports for various VMM features.

8100 Communication with the VMM console

5985 Communication to agents on hosts and library servers

443 File transfers to agents on hosts and library servers

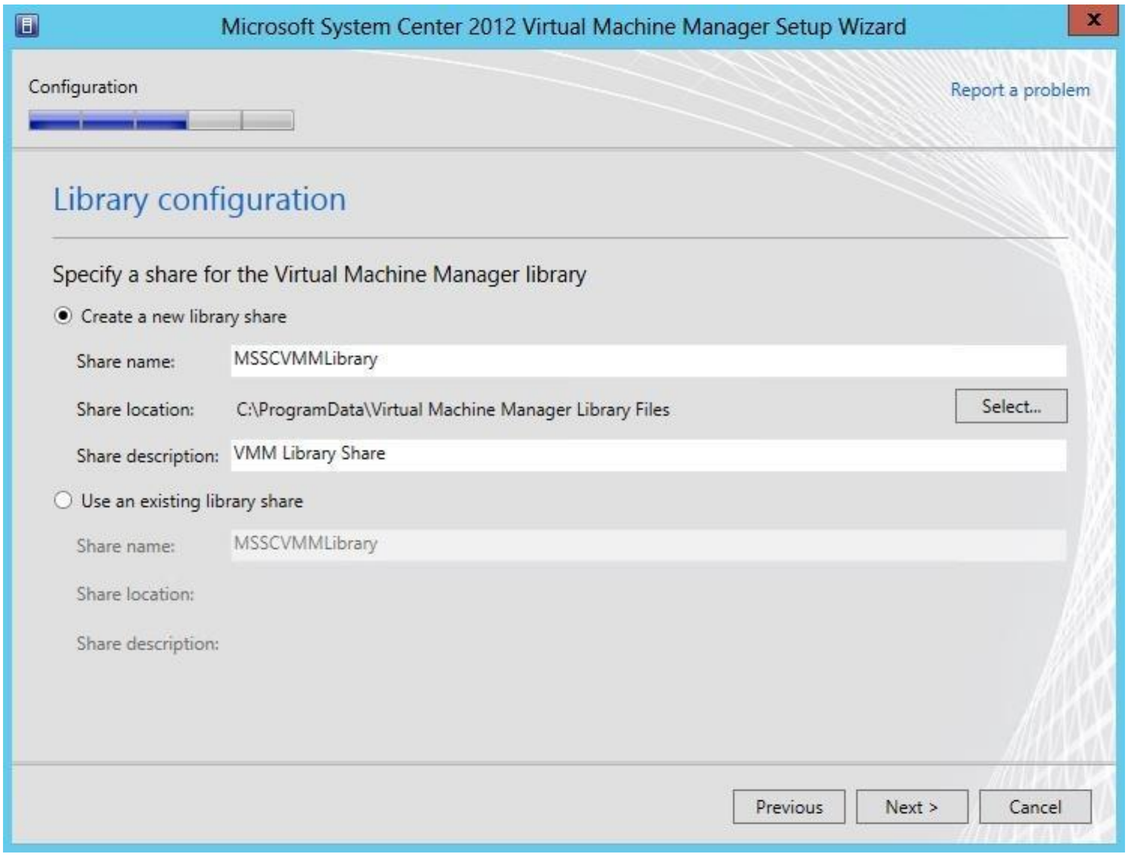
8102 Communication with Windows Deployment Services

8101 Communication with Windows Preinstallation Environment (Windows PE) agents

8103 Communication with Windows PE agent for time synchronization

We just use the default ports that should all be available.

Figure 18 “Step 13: SCVMM 2012 installation”



Microsoft System Center 2012 Virtual Machine Manager Setup Wizard
Configuration
Report a problem

## Library configuration

Specify a share for the Virtual Machine Manager library

● Create a new library share

Share name: MSSCVMMLibrary

Share location: C:\ProgramData\Virtual Machine Manager Library Files

Share description: VMM Library Share

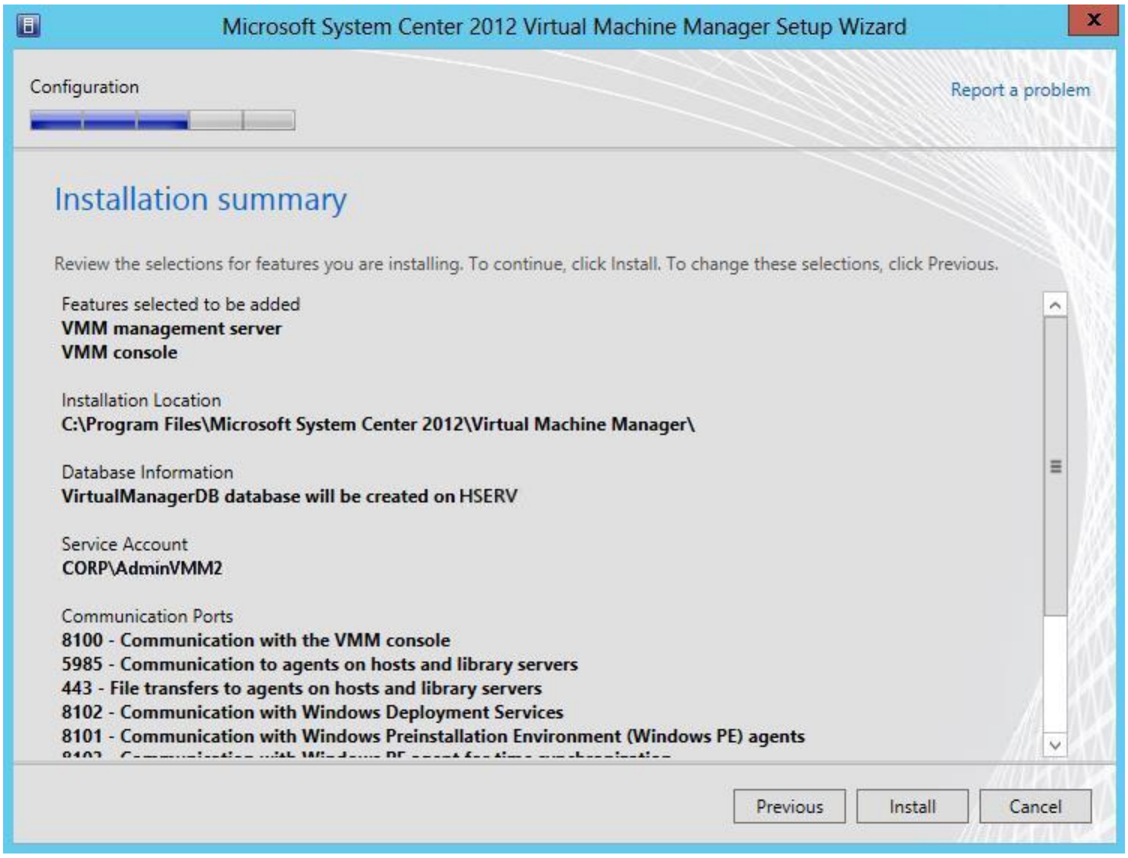
○ Use an existing library share

Share name: MSSCVMMLibrary

Share location:

Share description:

Figure 19 “Step 14: SCVMM 2012 installation”



Microsoft System Center 2012 Virtual Machine Manager Setup Wizard

Configuration

Report a problem

Installation summary

Review the selections for features you are installing. To continue, click Install. To change these selections, click Previous.

Features selected to be added
**VMM management server**
**VMM console**

Installation Location
**C:\Program Files\Microsoft System Center 2012\Virtual Machine Manager**

Database Information
**VirtualManagerDB database will be created on HSERV**

Service Account
**CORP\AdminVMM2**

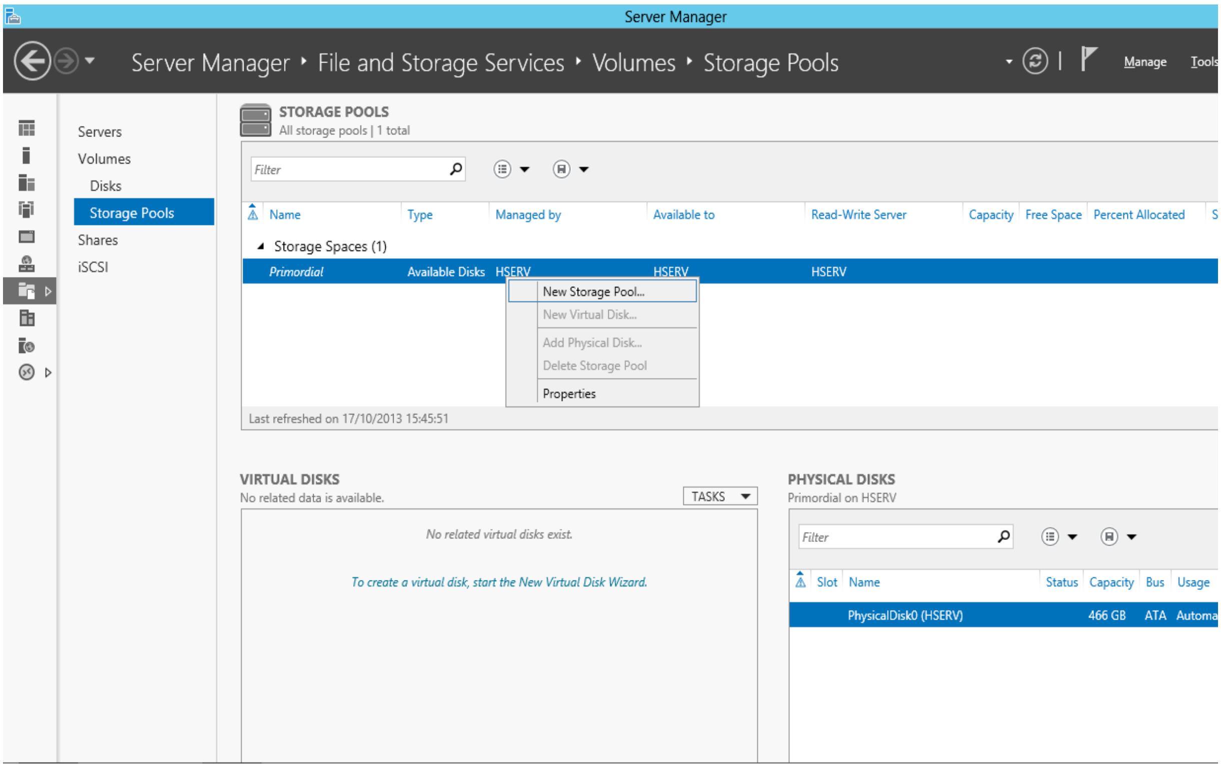
Communication Ports
**8100 - Communication with the VMM console**
**5985 - Communication to agents on hosts and library servers**
**443 - File transfers to agents on hosts and library servers**
**8102 - Communication with Windows Deployment Services**
**8101 - Communication with Windows Preinstallation Environment (Windows PE) agents**

Figure 20 "Step 15: SCVMM 2012 installation"

### 2.2.3 FABRIC

This chapter will demonstrate all the steps that were needed in order to set up the fabric part for this cloud. What needs to be mentioned here is that only one physical server is being used, so that is the only host added in CSVMM, named HSERV.corp.webclub.gr. For this reason the roles of this host would be Library, Host and VMM Server. Since this is done only because of lack of resources, it is highly recommended and according to the cloud computing features that there is a physical server for each one of these roles.

The first step is to create a virtual hard drive which will be assigned to the three VMs that are going to be used. Everything used by/for the application will be stored there. For the implementation of the above, there should be a storage pool, which will contain the virtual disk, which will contain a volume. Below there are snapshots of each step of the creation and assignment of the disk.



STORAGE POOLS
All storage pools | 1 total

| Name | Type | Managed by | Available to | Read-Write Server | Capacity | Free Space | Percent Allocated | S |
|---|---|---|---|---|---|---|---|---|
| Storage Spaces (1) | | | | | | | | |
| *Primordial* | Available Disks | HSERV | HSERV | HSERV | | | | |

Last refreshed on 17/10/2013 15:45:51

VIRTUAL DISKS
No related data is available.

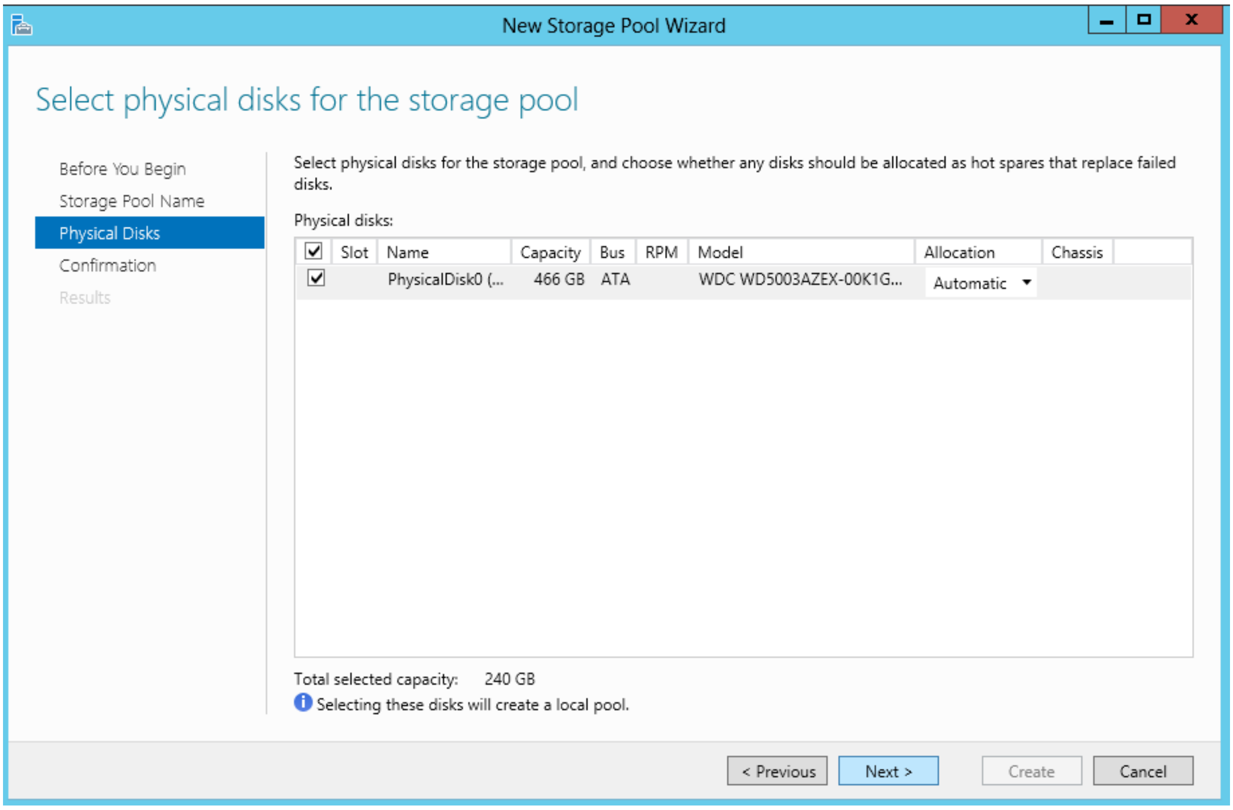
*No related virtual disks exist.*

*To create a virtual disk, start the New Virtual Disk Wizard.*

PHYSICAL DISKS
Primordial on HSERV

| Slot | Name | Status | Capacity | Bus | Usage |
|---|---|---|---|---|---|
| | PhysicalDisk0 (HSERV) | | 466 GB | ATA | Automa |

**Figure 21 “Step 1: Create and assign disk”**



Select physical disks for the storage pool

Select physical disks for the storage pool, and choose whether any disks should be allocated as hot spares that replace failed disks.

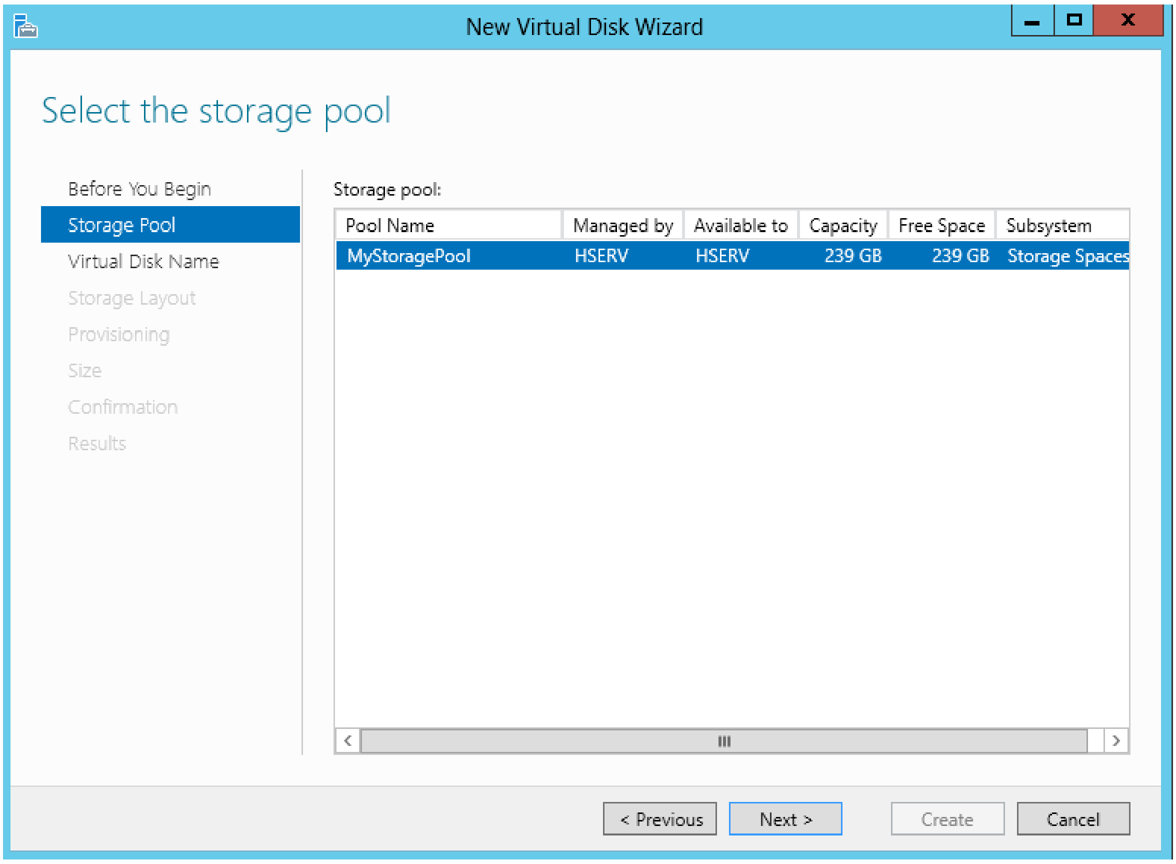
Physical disks:

| ☑ | Slot | Name | Capacity | Bus | RPM | Model | Allocation | Chassis |
|---|---|---|---|---|---|---|---|---|
| ☑ | | PhysicalDisk0 (... | 466 GB | ATA | | WDC WD5003AZEX-00K1G... | Automatic | |

Total selected capacity: 240 GB

Selecting these disks will create a local pool.

**Figure 22 “Step 2: Create and assign disk”**

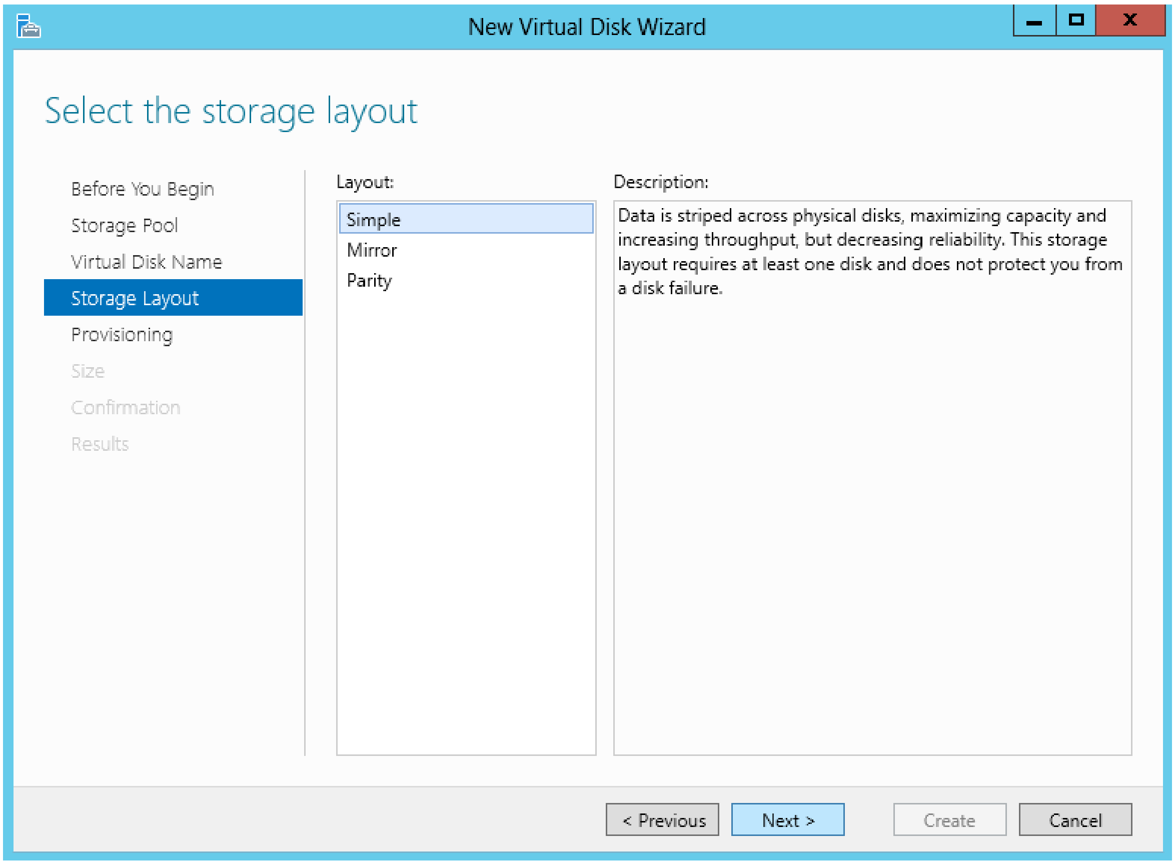


Select the storage pool

Storage pool:

| Pool Name | Managed by | Available to | Capacity | Free Space | Subsystem |
|---|---|---|---|---|---|
| MyStoragePool | HSERV | HSERV | 239 GB | 239 GB | Storage Spaces |

Figure 23 "Step 3: Create and assign disk"



Select the storage layout

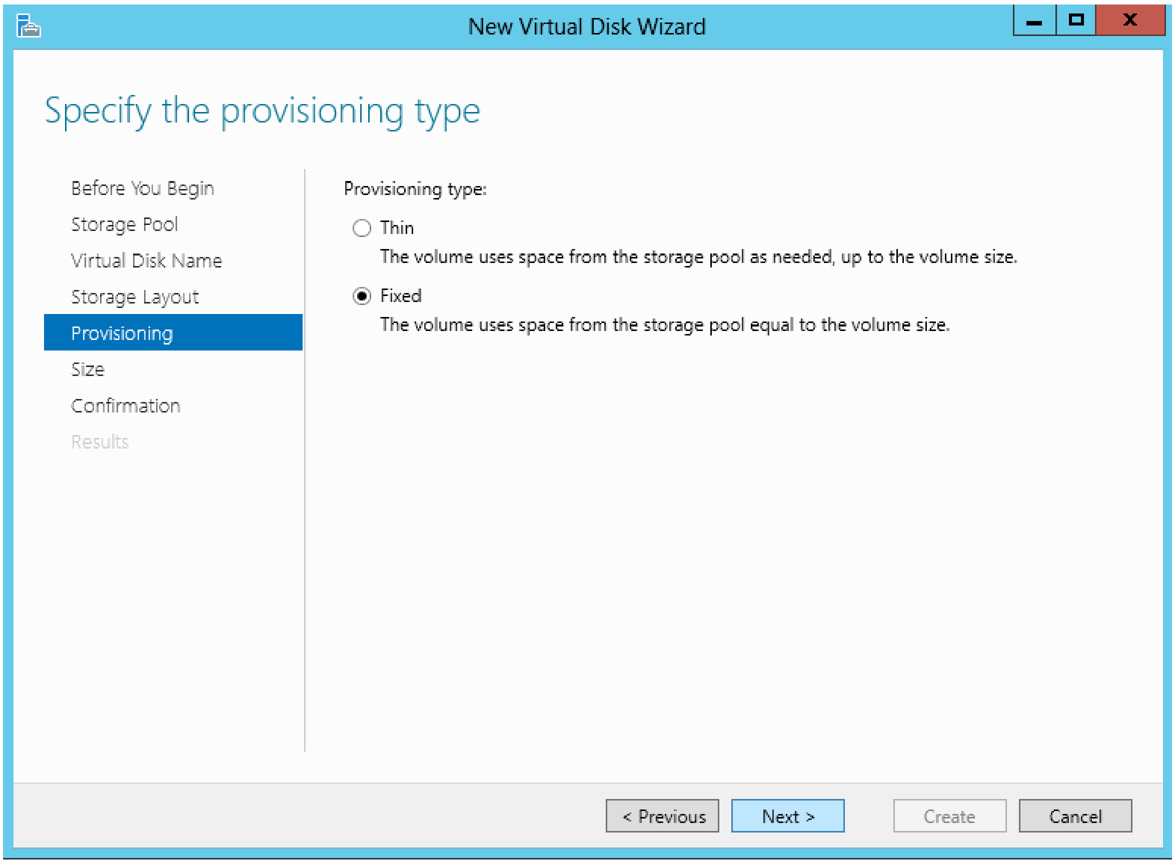
Layout:

- Simple
- Mirror
- Parity

Description:

Data is striped across physical disks, maximizing capacity and increasing throughput, but decreasing reliability. This storage layout requires at least one disk and does not protect you from a disk failure.

Figure 24 "Step 4: Create and assign disk"



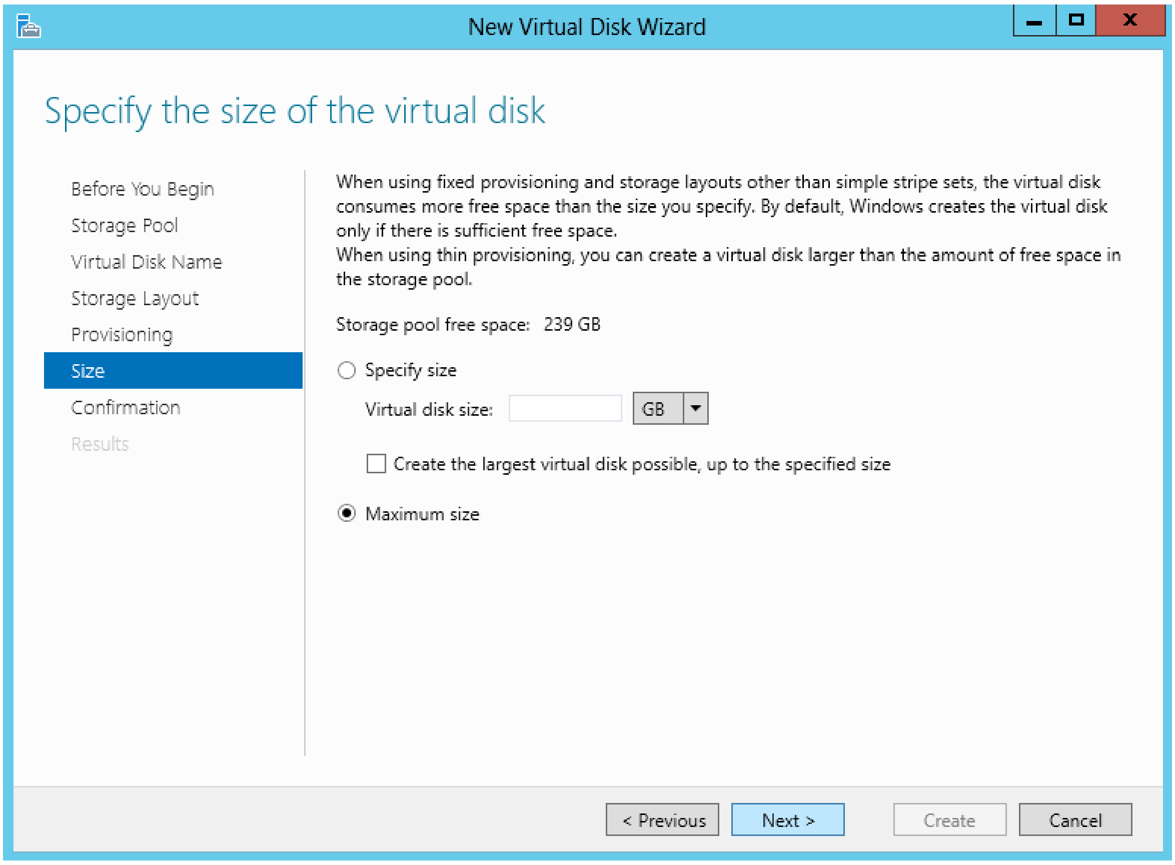
## Specify the provisioning type

Provisioning type:

○ Thin
The volume uses space from the storage pool as needed, up to the volume size.

● Fixed
The volume uses space from the storage pool equal to the volume size.

Figure 25 “Step 5: Create and assign disk”



## Specify the size of the virtual disk

When using fixed provisioning and storage layouts other than simple stripe sets, the virtual disk consumes more free space than the size you specify. By default, Windows creates the virtual disk only if there is sufficient free space.
When using thin provisioning, you can create a virtual disk larger than the amount of free space in the storage pool.

Storage pool free space: 239 GB

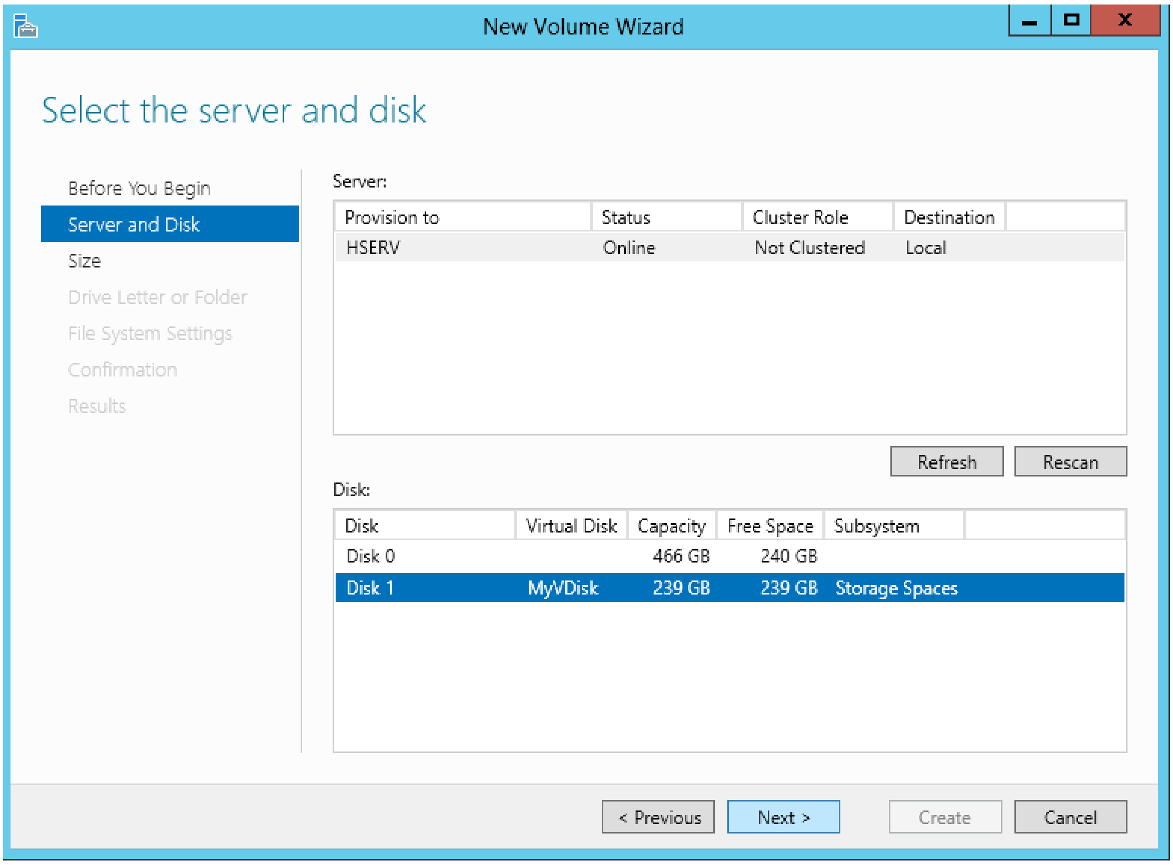
○ Specify size

Virtual disk size: GB

□ Create the largest virtual disk possible, up to the specified size

● Maximum size

Figure 26 “Step 6: Create and assign disk”



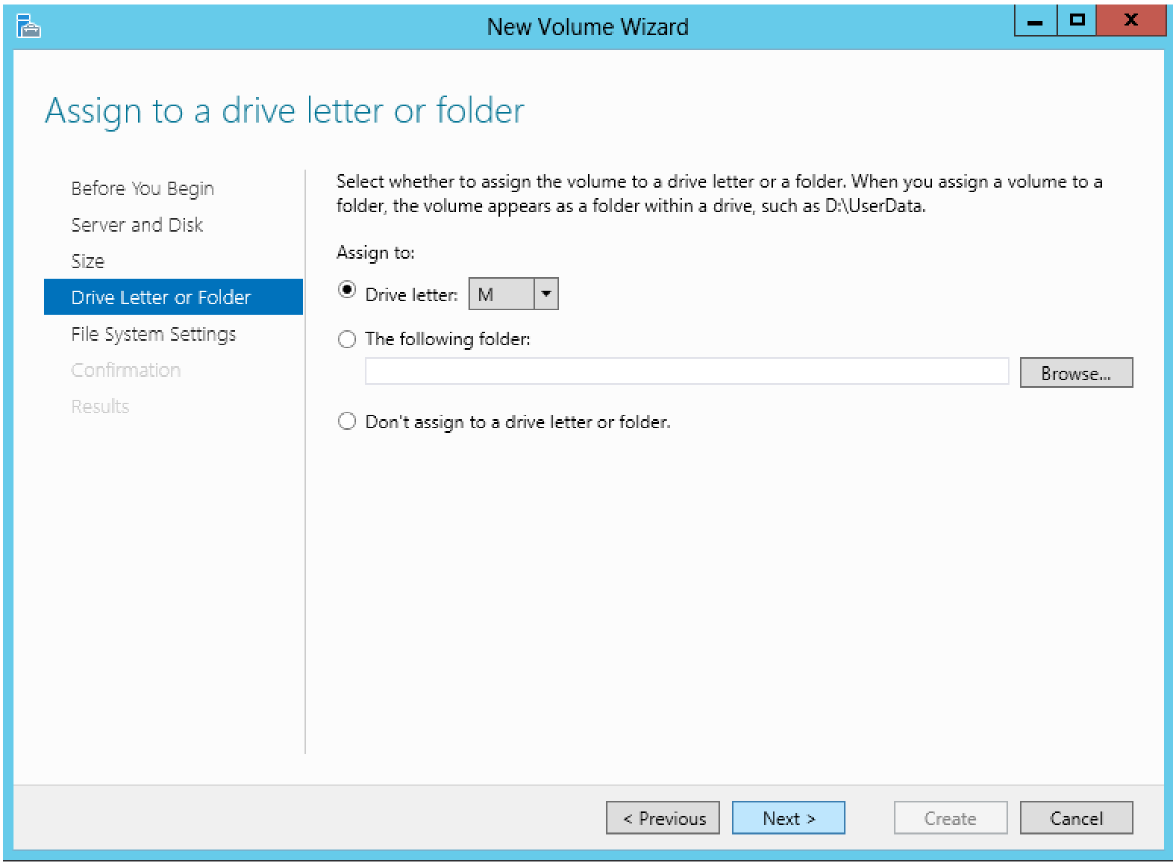
Server:

| Provision to | Status | Cluster Role | Destination |
|---|---|---|---|
| HSERV | Online | Not Clustered | Local |

Disk:

| Disk | Virtual Disk | Capacity | Free Space | Subsystem |
|---|---|---|---|---|
| Disk 0 | | 466 GB | 240 GB | |
| Disk 1 | MyVDisk | 239 GB | 239 GB | Storage Spaces |

Figure 27 “Step 7: Create and assign disk”



Select whether to assign the volume to a drive letter or a folder. When you assign a volume to a folder, the volume appears as a folder within a drive, such as D:\UserData.

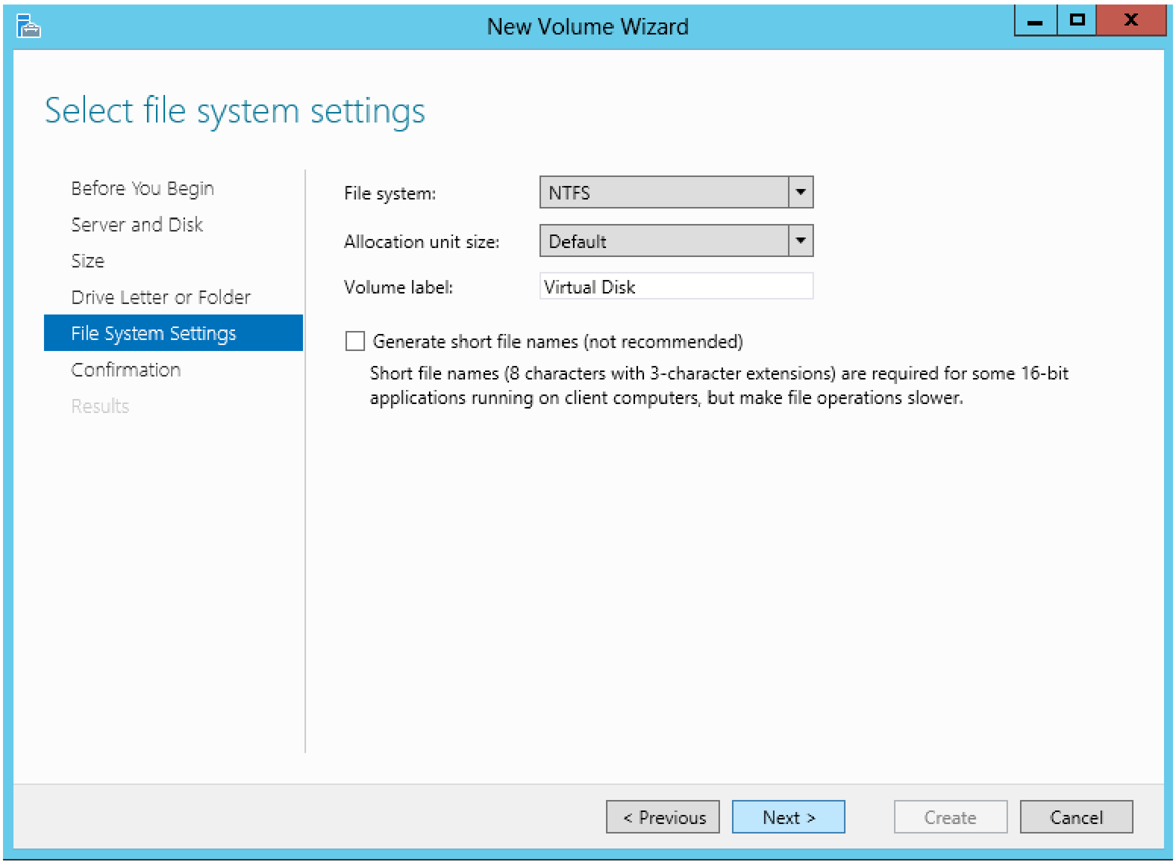
Assign to:

● Drive letter: M

○ The following folder:

○ Don't assign to a drive letter or folder.

Figure 28 “Step 8: Create and assign disk”



New Volume Wizard

Select file system settings

File system: NTFS

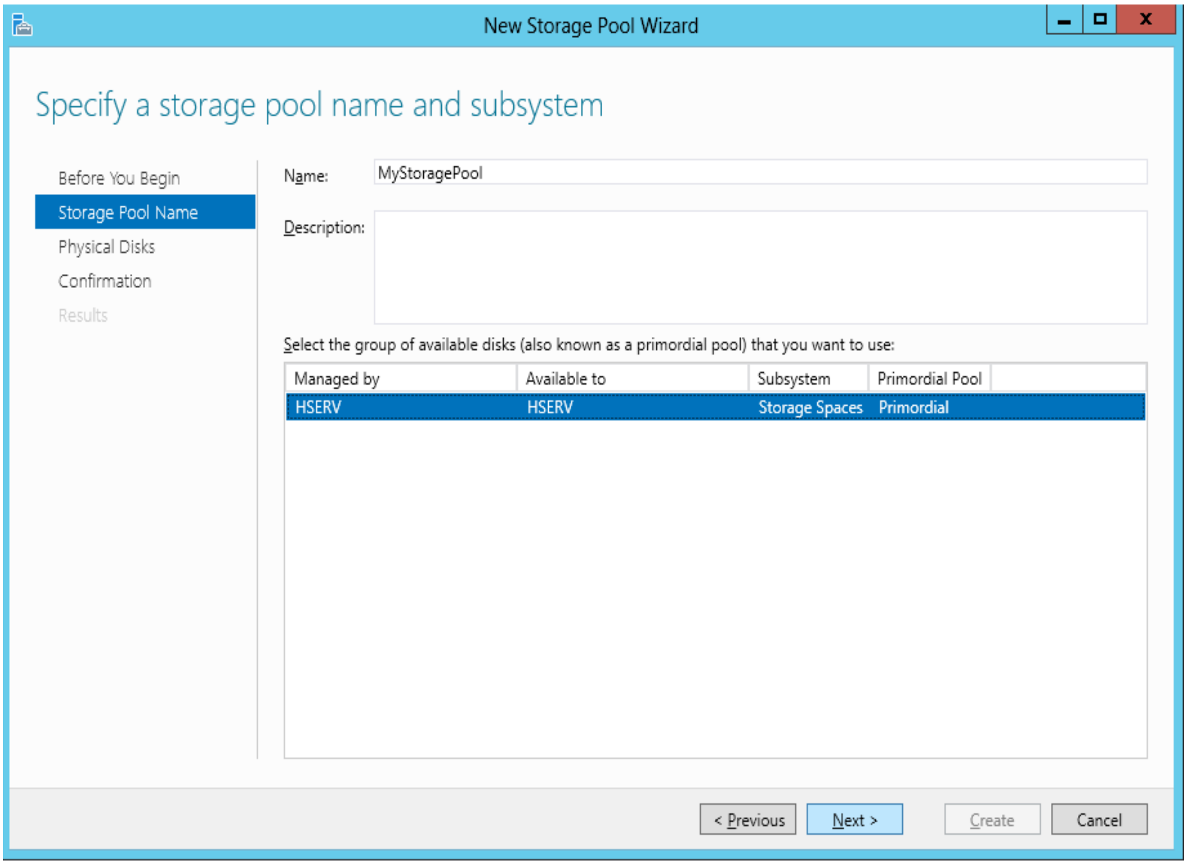
Allocation unit size: Default

Volume label: Virtual Disk

☐ Generate short file names (not recommended)

Short file names (8 characters with 3-character extensions) are required for some 16-bit applications running on client computers, but make file operations slower.

Figure 29 “Step 9: Create and assign disk”



New Storage Pool Wizard

Specify a storage pool name and subsystem

Name: MyStoragePool

Description:

Select the group of available disks (also known as a primordial pool) that you want to use:

| Managed by | Available to | Subsystem | Primordial Pool |
|---|---|---|---|
| HSERV | HSERV | Storage Spaces | Primordial |

Figure 30 “Step 10: Create and assign disk”

For the networking part we are using the logical network our server is connected to. In order for the VMs to integrate with is we have create a virtual switch (Atheros L1 Gigabit Ethernet 10/100/1000Base-T Controller - Virtual Switch). We also assigned a static IP to each of the VMs. Finally the load balancer used is Microsoft Network Load Balancing (NLB) added by default by SCVMM.

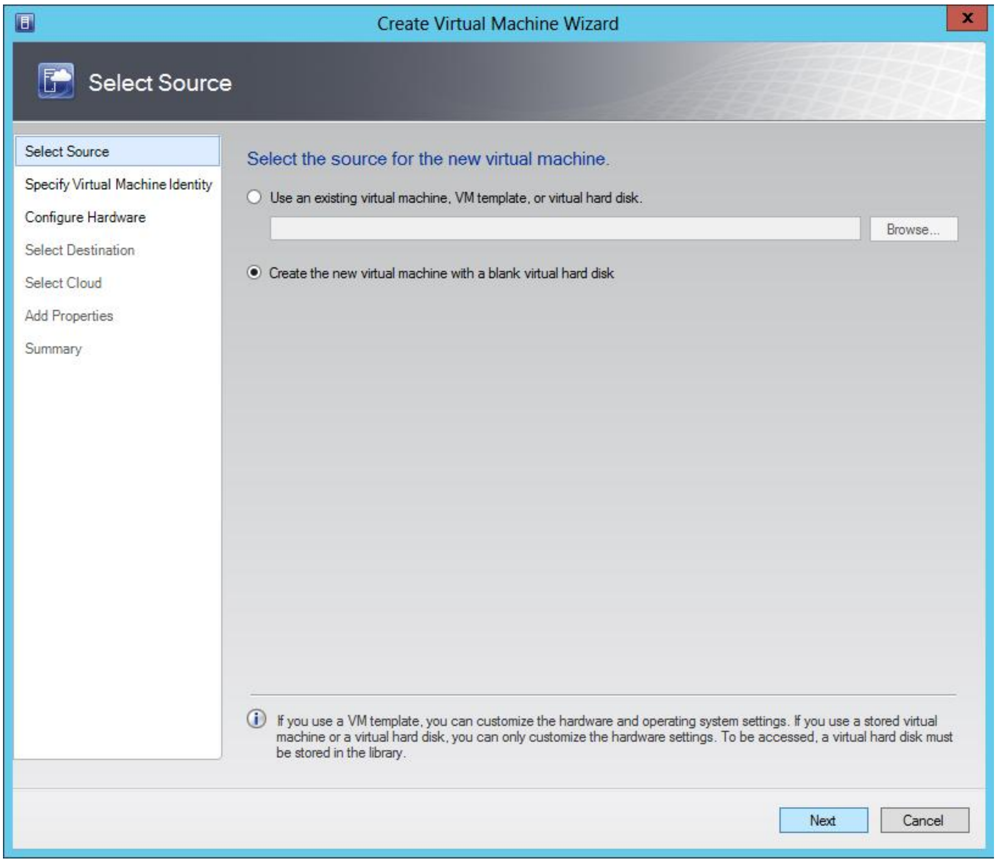
### 2.2.4 VIRTUAL MACHINES

For the implementation of the project, three VMs will be used which are named ApplicationVM, DatabaseVM and StorageVM. Below there is a step by step presentation of the creation of the first one.

The only specifications needed to be set are:

- Cloud Capability Profiles: Hyper-V
- Memory: 1024 MB
- Network Adapter 1: Connect to a VM network: Atheros L1 Gigabit Ethernet 10/100/1000Base-T Controller - Virtual Switch

The rest can remain with the default options.



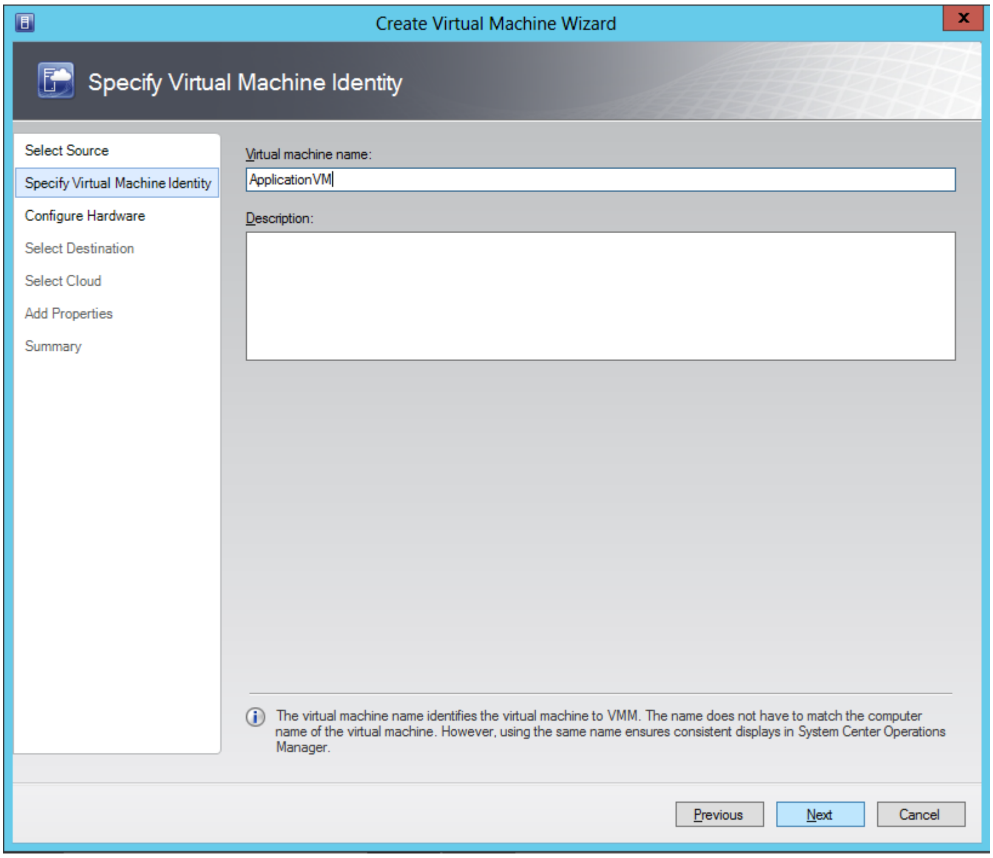
Select the source for the new virtual machine.

○ Use an existing virtual machine, VM template, or virtual hard disk.

● Create the new virtual machine with a blank virtual hard disk

If you use a VM template, you can customize the hardware and operating system settings. If you use a stored virtual machine or a virtual hard disk, you can only customize the hardware settings. To be accessed, a virtual hard disk must be stored in the library.

Figure 31 "Step 1: Create VM"

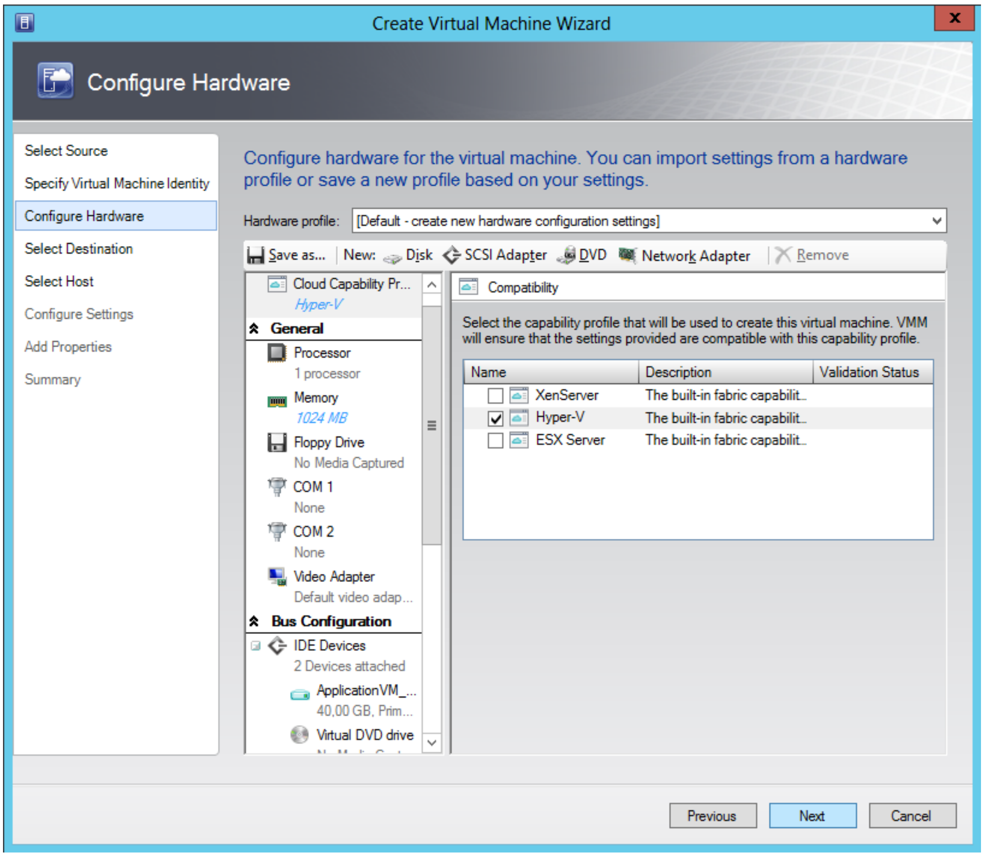


Virtual machine name: ApplicationVM

Description:

The virtual machine name identifies the virtual machine to VMM. The name does not have to match the computer name of the virtual machine. However, using the same name ensures consistent displays in System Center Operations Manager.

Figure 32 "Step 2: Create VM"



Configure hardware for the virtual machine. You can import settings from a hardware profile or save a new profile based on your settings.

Hardware profile: [Default - create new hardware configuration settings]

Cloud Capability Pr...
Hyper-V

**General**

Processor
1 processor

Memory
1024 MB

Floppy Drive
No Media Captured

COM 1
None

COM 2
None

Video Adapter
Default video adap...

**Bus Configuration**

IDE Devices
2 Devices attached

ApplicationVM_...
40,00 GB, Prim...

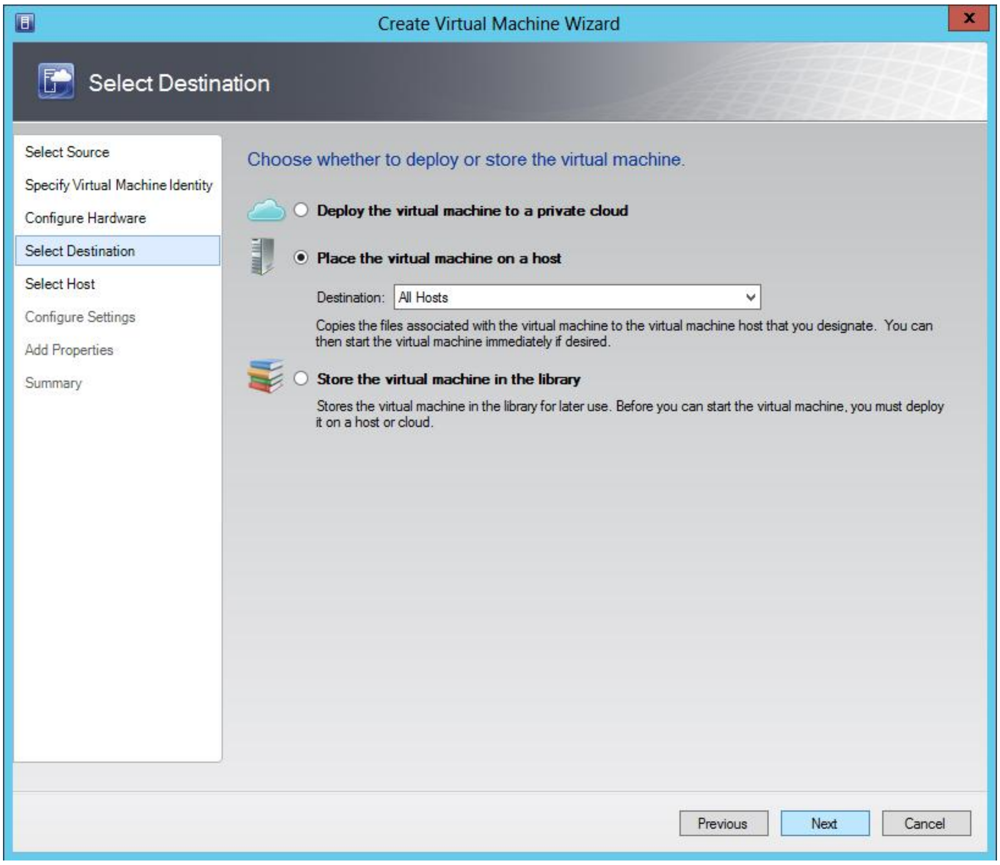
Virtual DVD drive

Compatibility

Select the capability profile that will be used to create this virtual machine. VMM will ensure that the settings provided are compatible with this capability profile.

| Name | Description | Validation Status |
|---|---|---|
| ☐ XenServer | The built-in fabric capabilit... | |
| ☑ Hyper-V | The built-in fabric capabilit... | |
| ☐ ESX Server | The built-in fabric capabilit... | |

Figure 33 “Step 3: Create VM”



Choose whether to deploy or store the virtual machine.

○ **Deploy the virtual machine to a private cloud**

● **Place the virtual machine on a host**

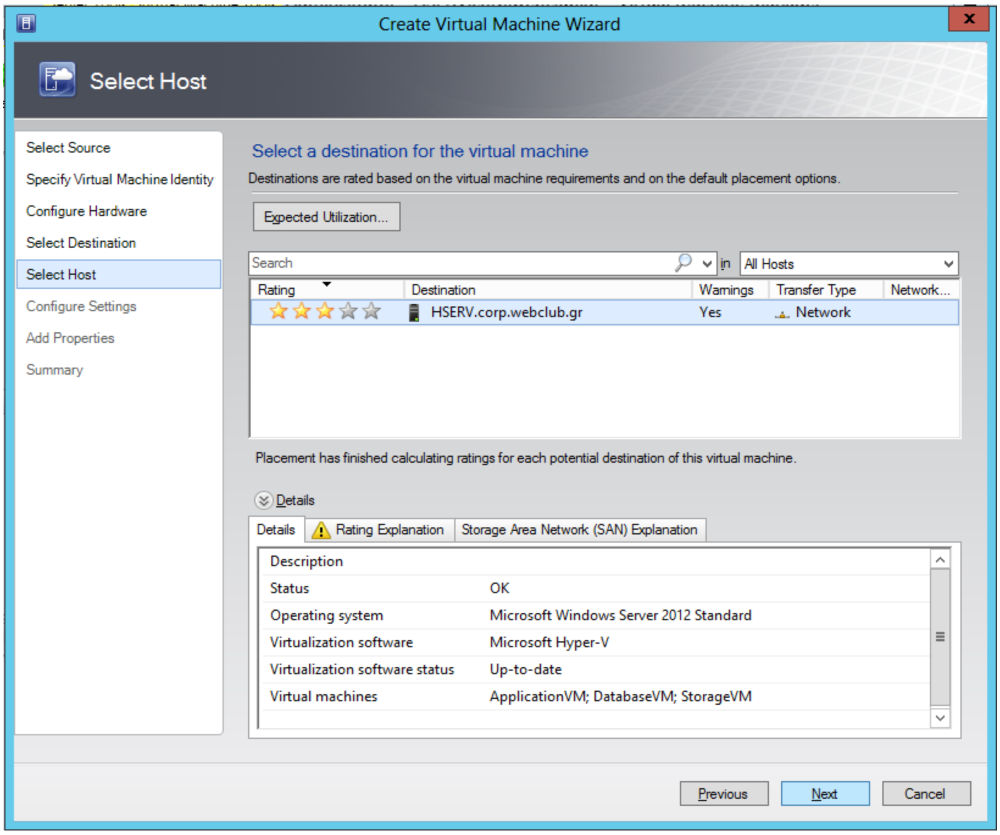
Destination: All Hosts

Copies the files associated with the virtual machine to the virtual machine host that you designate. You can then start the virtual machine immediately if desired.

○ **Store the virtual machine in the library**

Stores the virtual machine in the library for later use. Before you can start the virtual machine, you must deploy it on a host or cloud.

Figure 34 “Step 4: Create VM”



Select a destination for the virtual machine

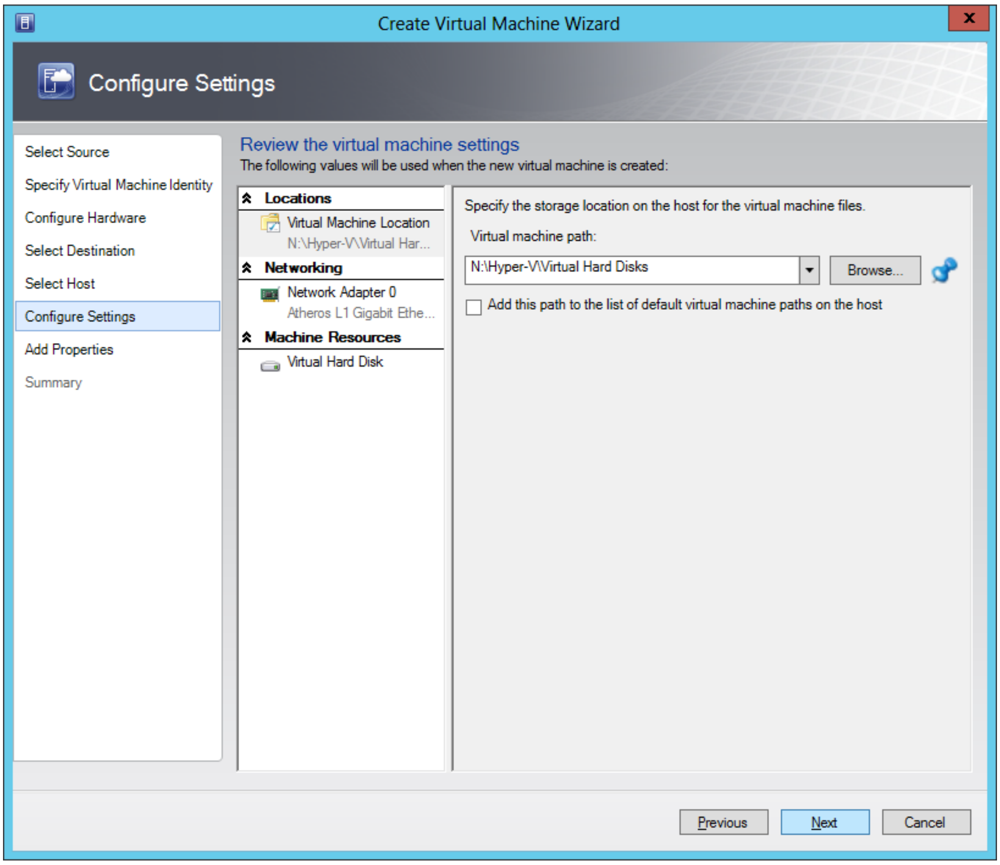
Destinations are rated based on the virtual machine requirements and on the default placement options.

| Rating | Destination | Warnings | Transfer Type | Network... |
|---|---|---|---|---|
| ★★★☆☆ | HSERV.corp.webclub.gr | Yes | Network | |

Placement has finished calculating ratings for each potential destination of this virtual machine.

| Description | |
|---|---|
| Status | OK |
| Operating system | Microsoft Windows Server 2012 Standard |
| Virtualization software | Microsoft Hyper-V |
| Virtualization software status | Up-to-date |
| Virtual machines | ApplicationVM; DatabaseVM; StorageVM |

Figure 35 “Step 5: Create VM”



Review the virtual machine settings

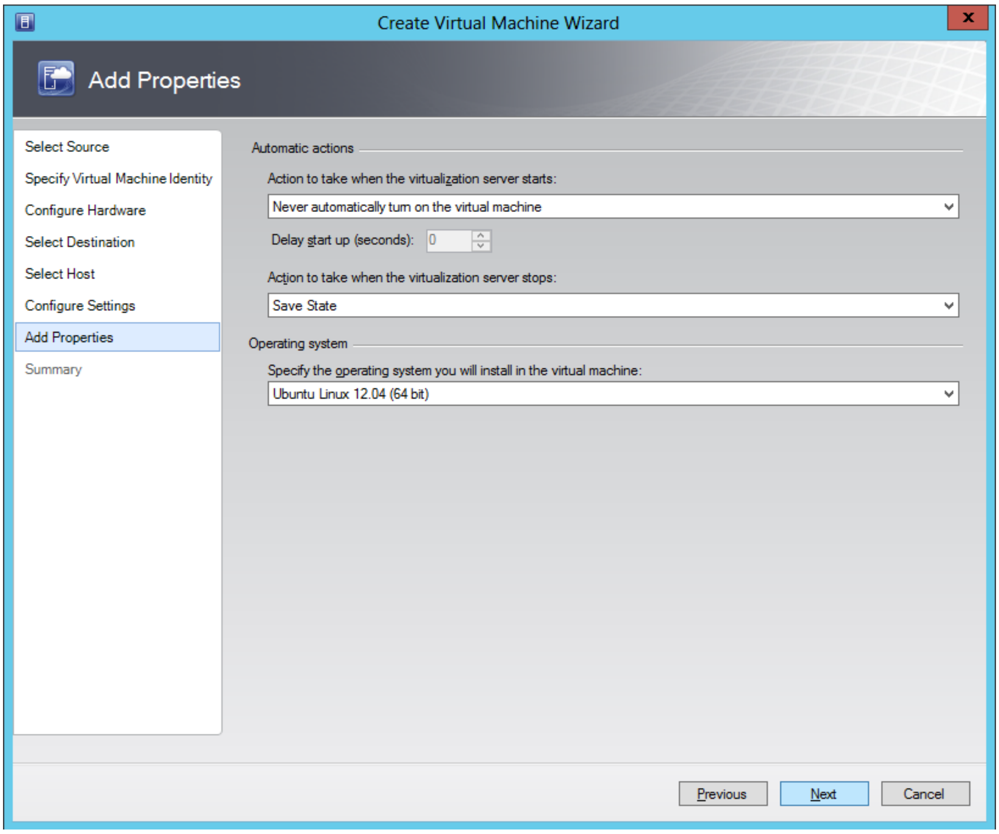
The following values will be used when the new virtual machine is created:

Specify the storage location on the host for the virtual machine files.

Virtual machine path: N:\Hyper-V\Virtual Hard Disks

□ Add this path to the list of default virtual machine paths on the host

Figure 36 “Step 6: Create VM”



Automatic actions

Action to take when the virtualization server starts:

Never automatically turn on the virtual machine

Delay start up (seconds): 0

Action to take when the virtualization server stops:

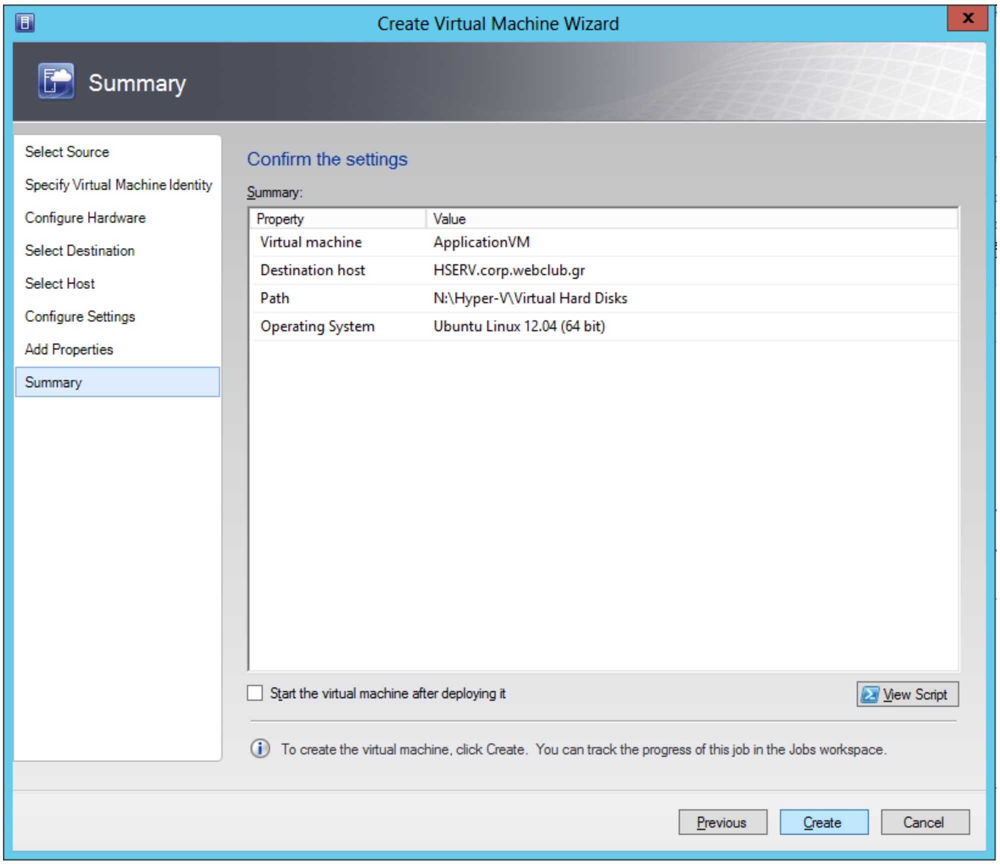
Save State

Operating system

Specify the operating system you will install in the virtual machine:

Ubuntu Linux 12.04 (64 bit)

Figure 37 "Step 7: Create VM"



Confirm the settings

Summary:

| Property | Value |
|---|---|
| Virtual machine | ApplicationVM |
| Destination host | HSERV.corp.webclub.gr |
| Path | N:\Hyper-V\Virtual Hard Disks |
| Operating System | Ubuntu Linux 12.04 (64 bit) |

□ Start the virtual machine after deploying it

To create the virtual machine, click Create. You can track the progress of this job in the Jobs workspace.

Figure 38 "Step 8: Create VM"

The properties assigned on each VM are:

ApplicationVM

- Number of processors: 1
- Memory: 1024 MB
- Virtual Hard Disk: fixed with 40 GB
- Cloud Compatibility Profile: Hyper-V
- MAC Adress: 00:15:5D:01:12:07
- IP Adress: 192.168.1.21

DatabaseVM

- Number of processors: 1
- Memory: 512 MB
- Virtual Hard Disk: fixed with 100 GB
- Cloud Compatibility Profile: Hyper-V
- MAC Adress: 00:15:5D:01:12:08
- IP Adress: 192.168.1.22

StorageVM

- Number of processors: 1
- Memory: 512 MB startup, 1024 MB max
- Virtual Hard Disk: fixed with 60 GB
- Cloud Compatibility Profile: Hyper-V
- MAC Adress: 00:15:5D:01:12:09
- IP Adress: 192.168.1.23

# 3. THE APPLICATION

## INTRODUCTION

Anything that has to do with the application, the software part of the thesis, is described in this chapter. First of all, you can find a description of the program, with the tasks assigned to each VM used. After that, there is a brief presentation of the library used for the encoding-transcoding, as well as all the formats that are supported and the properties that the user will be available to set. The chapter closes up with a description of all the key parts of code written in java.

## 3.1 DESCRIPTION

As mentioned in chapter 2.2.4 VIRTUAL MACHINES (page 46), our cloud consists of 3 Vms (ApplicationVM, DatabaseVM and StorageVM). What needs to be specified before you proceed is that in order to fully understand the following the application will be described as a client-server side app, where the client sends requests to the server and the server replies back. Below there is a description of each one of the VM's tasks.

- ApplicationVM

  This is where the service is running. All the proceeding and coding on behalf of the server is being executed in this VM.

  First of all, a service always running on this VM is waiting for requests on its IP. The connection is established after the client-user has successfully logged in. Then, he chooses whether he wants to download a video or upload one on the server. In order to be able to download files, he has to have them uploaded first, since he only has access to a specific folder created for him that contains all his videos. If he decides to upload, he just selects the file he wants to upload and it's being sent to the server. When he chooses to download one, he chooses the file he wants among his files and then chooses the properties (type, bitrate, resolution, etc.) he wants for it. After that the video is being transcoded according to the properties specified and sent to him.

- DatabaseVM

  This is where our database is being stored. It consists of one table with a row being created for each user that registers and 3 columns (id, username, password).

  When the user runs the application, he is prompt to log in or register by filling his username and password. After the credentials are authenticated by the system, he gains access to his folder and can start using the application.

- StorageVM

  This VM's task is to store all the users' folders. It's the one with the greatest capacity. Every time a user registers to the system, a folder is created for him named by his id in the database.

## 3.2 TRANSCODING

This chapter describes the tools that were used for transcoding the video files, according to the properties specified.

For this task, we are using FFmpeg. From the libraries provided by it we are using libavcodec, an audio/video codec library and the command-line tools used are ffmpeg for converting the video files and ffplay for playing them. The properties that have to be specified are the video file’s name, format-container, video codec, frame rate and aspect. The codec we are working with is H.264 and the available containers are AVI (Audio Video Interleave), MPEG-4, FLV and MOV.

## 3.3 CODE

In this section we are going to describe the java code of our application. As mentioned in chapter 3.1 Description (page 52), all the procedures will be taking place in the ApplicationVM where there is a jar running waiting for requests. The client will be connecting to this host and the host will be responsible for the connection with the other two VMs.

It is important to note here that we create a socket for the communication of the client with the server and we are using the protocol used is TCP. Also in order to have access and read/write capabilities from the ApplicationVM to the folder where the users' videos are stored, which is in the StorageVM, we are mounting the folder using samba. The application consists of three java projects named VideoServer, VideoCommonLib and VideoClient.

### 3.3.1 VideoCommonLib

This folder contains everything that should be directly accessible both from the server and the client. That includes all the messages that will be sent from the client to the server and vice versa that form the communication protocol we implemented, as well as the video properties, in case of a download request from the client. The classes of the project are ClientMessage, ServerMessage and VideoProperties.

#### 3.3.1.1 ClientMessage

This file contains an enum with all the messages that can be sent from the client to the server.

```
public enum ClientMessage {
	DIRECTORY_LISTING,
	REGISTER,
	AUTHENTICATE,
	DOWNLOAD,
	UPLOAD,
	CANCEL
}
```

- Directory_Listing asks the server to return a list with the video filenames inside the user's directory when the user chooses the download option.

- Register asks the server to register the user to the database.

- Authenticate asks the server to authenticate the user. We check if the user exists in the database.

- Download asks the server to send a video.

- Upload asks the server to be ready to receive a video.

- Cancel is not needed but can be used if for some reason we will want to ask the server to stop the session.

#### 3.3.1.2 ServerMessage

This file contains an enum with all the messages that can be sent from the server to the client. As you can understand from the messages, in our protocol the server is the one who is always replying to the client's requests.

```
public enum ServerMessage {
	AUTHENTICATION_FAILED,
	AUTHENTICATION_SUCCEEDED,
	USER_REGISTER_FAILED,
	USER_REGISTER_SUCCEDED
}
```

- Authentication_Failed informs the client that he couldn't authenticate the user by his credentials. The user is unable to login

- Authentication_Succeded informs the client that the server authenticated the user and he exists in the database. The user loged in successfully.

- User_Register_Failed informs the client that the user couldn't be added to the database. The user is unable to register

- User_Register_Succeded informs the client that the user was added to the database. The user was registered successfully.

#### 3.3.1.3 VideoProperties

This class is used to set the appropriate values that the client asked for the video he wants to download.

```
public VideoProperties(String filename, String container, String codec, String
frame, String aspect) {
		this.filename = filename;
		this.container = container;
		this.codec = codec;
		this.frame = frame;
		this.aspect = aspect;
}
```

### 3.3.2 VideoServer

This project contains all the coding that is being executed on the server. A jar file containing the VideoServer classes as well as the VideoCommonLib ones, will be always running on the ApplicationVM. The externam libraries added to this project are mysql-connector-java-5.1.18-bin.jar and jcifs-1.3.17.jar. The classes of the project are AuthenticatedUsers, DbConnections, MultiServer, MultiServerThread and VideoConvert.

#### 3.3.2.1 AuthenticatedUsers

This class is responsible for letting us know which users are currently logged in to our server. The usernames of the users are being stored in an ArrayList

```
private static ArrayList<String> LoggedInUsers = new ArrayList<String>(50);

	public static void loginUser(String username) {
		if (!LoggedInUsers.contains(username)) {
			LoggedInUsers.add(username);
		}
	}

	public static void logoutUser(String username) {
		LoggedInUsers.remove(username);
	}

	public static boolean isUserLoggedin(String username) {
		return LoggedInUsers.contains(username);
	 }
```

- loginUser is adding a user to the list.
- logoutUser is removing a user from the list.
- isUserLoggedin is returning a boolean value stating wether the user is logged in or not.

#### 3.3.2.2 DbConnections

This class is establishing the connection between the ApplicationVm and the mySql database in the DatabaseVM.

```
public static void createMySqlConnection() {
```

```
             try {
                    // Load the JDBC driver
                    DriverManager.registerDriver(new com.mysql.jdbc.Driver());

                    String serverName  = "192.168.1.22:3306";
                    // Create a connection to the database
                    String mydatabase = "test";
                    String url = "jdbc:mysql://" + serverName + "/" + mydatabase;
// a JDBC url
                    String username = "*****";
                    String password = "*****";
                    mySqlConnection = DriverManager.getConnection(url,
                                 username, password);

                    System.out.println("Connected to MySQL\n");
             } catch (SQLException e) {
                    //return "Error while connecting to database: " +
e.getMessage();
                    System.out.println("Error while connecting to database: "
                    + e.getMessage());
             }
```

### 3.3.2.3 MultiServer

This class contains the main method of the project on the server. It is always running, waiting for requests on a specific port.For every request it creates a new thread, which is an instance of MultiServerThread, to handle it.

```
public static void dbg(String s){System.out.println(s);}

    public static void main(String[] args) throws IOException {
        ServerSocket serverSocket = null;
        boolean listening = true;

             DbConnections.createMySqlConnection();

        try {
            serverSocket = new ServerSocket(8089);
                    System.err.println("Server waiting for conections on port
8089");
        } catch (IOException e) {
            System.err.println("Could not listen on port: 8089. Reason: " +
e.getMessage());
            System.exit(-1);
        }

        while (listening){
             MultiServerThread multiServerThread = new
MultiServerThread(serverSocket.accept());
             dbg("accepted");
             multiServerThread.start();
        }
        serverSocket.close();
}
```

### 3.3.2.4 MultiServerThread

This is the thread created for every request the server gets. It handles all the communication with the client any task that has to be done in the other two VMs, DatabaseVm and StorageVM. Because of the length of this class the whole code will not be shown here. We will only describe all the methods and their tasks.

```
public void run() {

        try {
            clientOutgoing = new
ObjectOutputStream(socket.getOutputStream());
            clientIncoming = new
ObjectInputStream(socket.getInputStream());
            Object msg = clientIncoming.readObject();

            if (!(msg instanceof ClientMessage)) {
                System.err.println("msg not of ClientMessage type:" +
msg.getClass().getName());
                return;
            }

            ClientMessage message = (ClientMessage) msg;

            switch (message) {
                case REGISTER:
                    register();
                    break;
                case AUTHENTICATE:
                    System.out.println("Authenticating...");
                    authenticate();
                    break;
                case DOWNLOAD:
                    System.out.println("DO DOWNLOAD...");
                    sending();
                    break;
                case UPLOAD:
                    System.out.println("DO UPLOAD...");
                    receiving();
                    break;
                case DIRECTORY_LISTING:
                    System.out.println("Sending directory contents");
                    sendDirectoryContents();
                    break;
                default:
                    System.err.println("Not supporting function " +
message);
            }
        } catch (IOException e) {
            System.err.println("Could not read data from client: " +
e.getMessage());
        } catch (ClassNotFoundException e) {
            System.err.println("Could not deserialize class: " +
e.getMessage());
        }
}
```

- run() is called every time a new thread is created. In the switch statement we are checking the message we got from the client and call the appropriate method to handle the request.

- register() reads the credentials (username and password) that were sent from the client and is calling a method named queryDbForRegister(String, String) that is adding the user to the database if he doesn't exist.

- authenticate() reads the credentials that that were sent from the client and is calling queryDbForLogin(String, String) that is adding a user the AuthenticatedUsers, if the user exists in the database and is not already logged in.

- sending() runs along with the method downloading() from the VideoClient and is sending a video file to the client. Before it sends the file, it calls VideoConvert.transcode(VideoProperties), which is converting the video file according to the properties that was sent from the client.

- receiving() runs along with the method uploading() from the VideoClient and is receiving a video file sent from the client.

- sendDirectoryContents() is called when the client asks to download a file. It returns a String table with all the files located in the user's folder so that he can choose which one to download.

- createDir(String) is called from queryDbForRegister. When a new user registers it gets his username and creates a folder on the mounted disk of the StorageVM named with it. For the specification of the path we are using samba and jicfs.

#### 3.3.2.5 VideoConvert

This class does the transcoding of the video. This is being done by executing a terminal command. The only method that needs to be explained is transcode(VideoProperties) which converts the video and stores it in a folder named «/transcode» which is located the user's folder and will contain all his videos that have been transcoded.

```
Process proc = rt.exec("ffmpeg" + "-i" + properties.getFilename()
                                 + "-f" + properties.getContainer()
                                 + "-c:v" + properties.getCodec()
                                 + "-aspect" + properties.getAspect()
                                 + "-r" + properties.getFrame()
                                 + "/transcoded/"
                                 + properties.getFilename());
```

### 3.3.3 VideoClient

This project contains all the coding that is being executed on the client, so it also has all the graphical part of the application. The classes of the project are LoginFrame, AppFrame, RegistrationFrame, VideoPropFrame and VideoClient.

#### 3.3.3.1 LoginFrame

This class has the main class of the VideoClient project and it is the first frame the user sees. If it is a new user he presses «register» and sees the RegistrationFrame window. If the user presses «Log in», VideoClient.authenticating(String, String) is being called given the username and password the use entered.

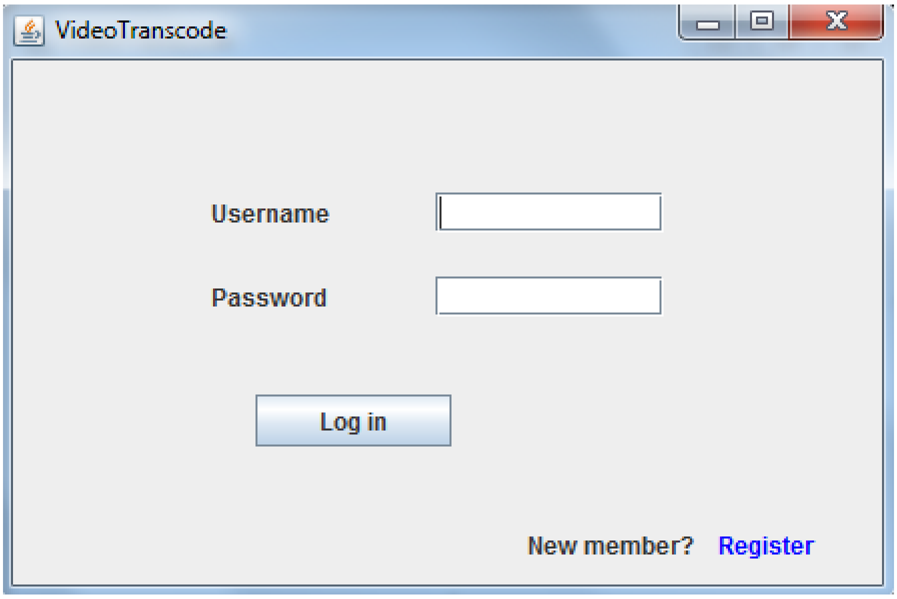


Figure 39 "Login form"

#### 3.3.3.2 AppFrame

This frame opens after the user is successfully logged in. When the button «Upload» is pressed the method VideoClient.sendMessage(ClientMessage.UPLOAD) is called and then the VideoClient class informs the server that the user wants to upload a video. When the button «Download» is pressed the user is being redirected to the VideoPropFrame window.

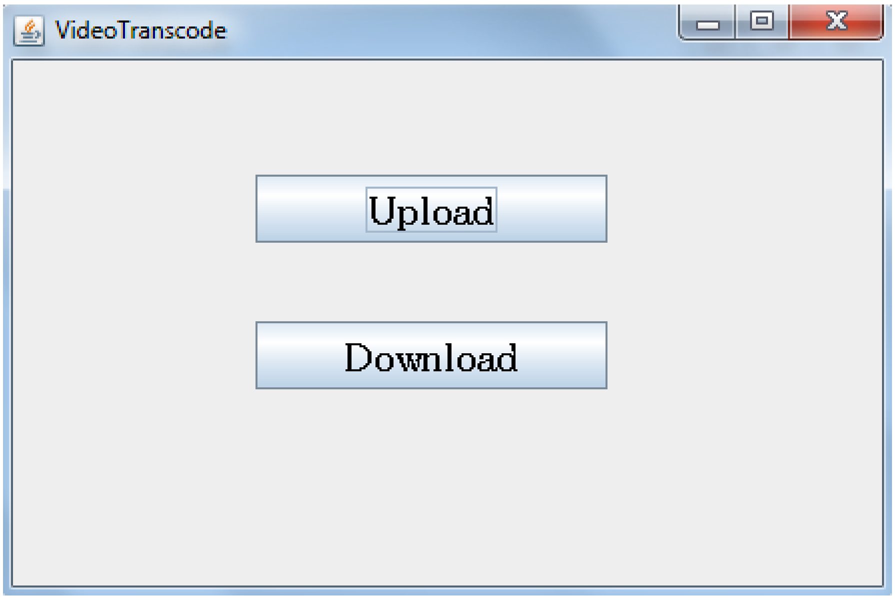


Figure 40 "Video app frame"

### 3.3.3.3 RegistrationFrame

The user gets this window when he presses «register» from the LoginFrame. Here he fills a desired username and password and presses «OK». When the «OK» button is pressed, the method VideoClient.registering(String, String) is called getting the username and the password the user filled earlier.

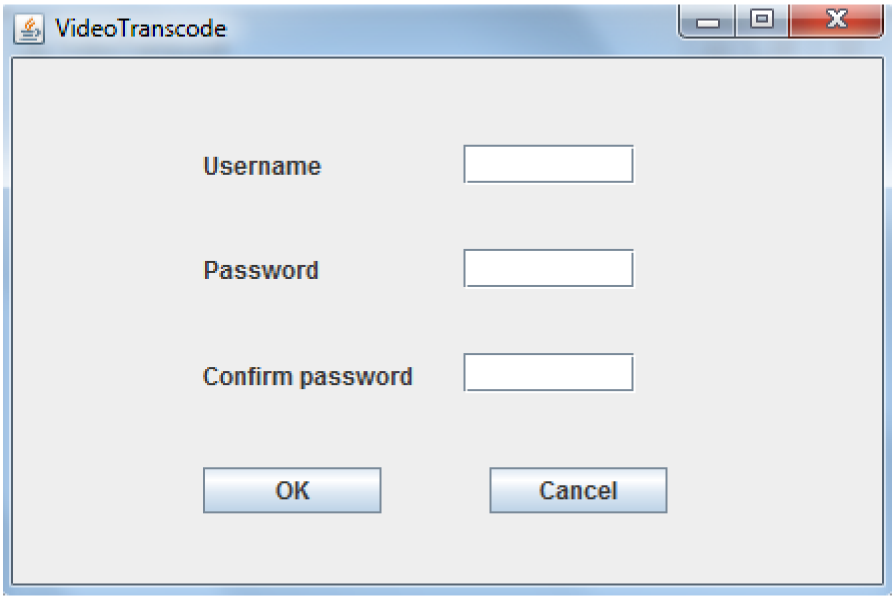
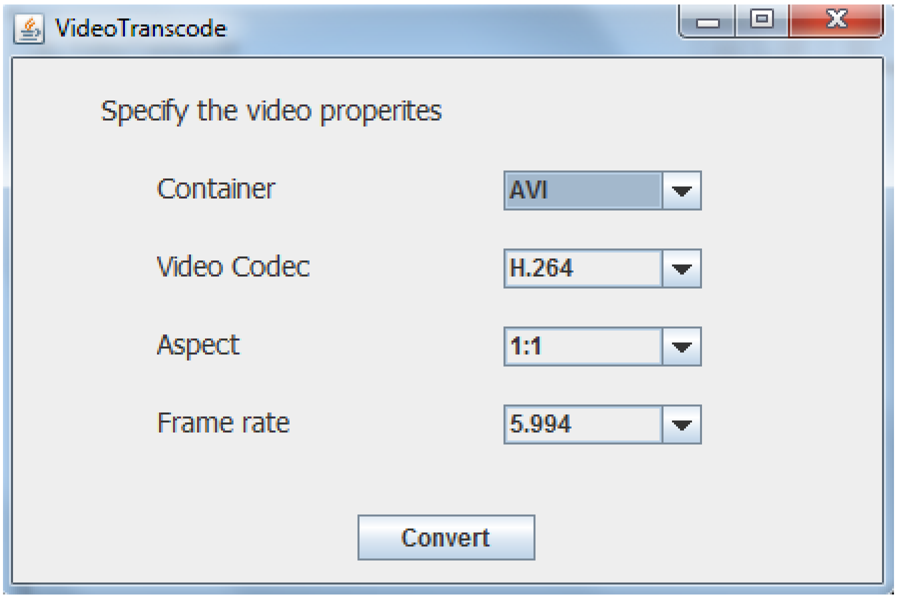


Figure 41 "Registration form"

### 3.3.3.4 VideoPropFrame



Specify the video properites

Container: AVI

Video Codec: H.264

Aspect: 1:1

Frame rate: 5.994

Figure 42 "Video properties frame"

The user sees this window after he chose to download a video from the server. Here he specifies the properties he want for the video. He can choose the container, aspect and frame rate he wants for it. The video codec we are working with is H.264. When the «Convert» button is pressed the user gets a list of all the videos that are stored on his folder on the server and he chooses the one he wants to convert and download. Then the method VideoClient.setProperties(VideoProperties) is called given the properties selected and right after that we call the method VideoClient.setMessage(ClientMessage.DOWNLOAD) that infors the server that user wants to download a video.

### 3.3.3.5 VideoClient

This class includes all the major methods from the part of the client. It creates the socket for the connection with the server and is responsible for the implementation of the whole protocol. Because of the length of this class the whole code will not be shown here. We will only describe all the methods and their tasks.

```
    static private String hostname = "dillo.dyndns.org";
    static private int port = 8089;
    static private Socket client = null;
     static private InputStream inputSt = null;
```

Information for the connection

The methods that are called first are registering(String, String) and authenticating(String, String).

- registering(String, String) is called from RegistrationFrame. Firstly, it calls the method connect(). After the connection is established successfully, it is sending the user's username and password to the server and it gets a reply stating if the user was registered successfully or not. If the user is registered, it shows the LoginFrame.

- authenticating(String, String) is called from LoginFrame. Firstly, it calls the method connect(). After the connection is established successfully, it is sending the user's username and password to the server and it gets a reply stating if the user was authenticated successfully or not. If the user is authenticated, it shows the AppFrame.

```
        public static void sendMessage(ClientMessage msg) {
                switch (msg) {
                        case DOWNLOAD:
                                sendClientMessage(msg);
                                sendUsername();
                                downloading();
                                break;
                        case UPLOAD:
                                sendClientMessage(msg);
                                sendUsername();
                                uploading();
                        case DIRECTORY_LISTING:
                                getFileList();
                                break;
                        default:
                                throw new RuntimeException("invalid parameter " + msg);
                }
        }
```

- sendClientMessage(ClientMessage) sends a message of type ClientMessage to the server.

- sendUsername() sends the username to the server. Since the application is multithreaded we have to be sending the username of the user that made the request each time.

- downloading() runs along with the method sending() from VideoServer. At the begining it shows a file dialog to the user and prompts him to select where to save the video in his computer. After that it sends the properties specified from VideoPropFrame to the server and after the video is downloaded completely it shows the time that was needed.

- uploading() runs along with the method receiving() from VideoServer. At first it shows a file dialog to the user and promts him to select a video from his computer. Then it sends the video file to the server.

- getFileList() calls sendClientMessage(ClientMessage.DIRECTORY_LISTING) as well as sendUsername() and asks the server for a list of the video files stored in the folder of the user.

## 4. CONCLUSIONS AND FUTURE EXTENSIONS

Although cloud computing is not really a new concept, its popularity continues to grow rapidly. The number of IT businesses that are moving their data and applications “on the cloud” is and will keep increasing. That can also be shown by the fact that even new forms of cloud are still being created or combined, like the community and hybrid cloud respectively. The most popular form is the private cloud, according to the new 2013 Cloud Computing Survey conducted by Palmer Research and QuinStreet Enterprise [5]. Also, software as a service (SaaS) is presented to be more popular than both infrastructure as a service (IaaS) and platform as a service (PaaS).

This thesis covered the creation and presentation of a private cloud that could be used by a company to offer SaaS. The tool chosen for the creation of the cloud and the VMs is SCVMM where we created three Hyper-V virtual machines. It is worth mentioning here that there are several other tools and types of virtualization that could offer similar capabilities. The video library used (FFmpeg) is the one considered the most popular by many IT professionals and the video codec that the application works with (H.264) can offer a great range of properties to be set. A major aspect this thesis did not go into in detail is security, which is something a company willing to convert its product to a cloud service should consider.

The major obstacle this project had to overcome was the limited number of available resources. The most noticeable was the use of only one physical server that had to get all the roles in SCVMM. Another drawback was the limited pool of the available RAM. There were only 4GB that had to be distributed to the three VMS plus the central node that is running all the tools used for the cloud implementation. Although the above have a major impact on the speed and performance of the application and the whole system, there is scalability, a feature necessary for a cloud. That means that with the import of the resources needed and their assignment to the VMS we will get a great boost without having to make any further changes. Even the assignment of the appropriate roles in case of the use of a second server can be easily done with the tools used.

The service described in the thesis consists of an application that offers video storage and transcoding. At this point, the application offers limited possibilities to the end-user, which means that there is a big number of aspects that could be added. First of all, the support of more codecs and containers. Some really useful ones would be WMV and DivX, because of their frequent use from the public and a number of codecs for mobile devises and consoles like MP4 for Android or Apple devices, as well as Microsoft's Xbox or Sony's PS3 or PSP. What can also be added is the possibility for the user to add some more properties to the final video. Some possible enhancements would be the import of the video's scale size as well as its bitrate. Of course this would add some noticeable complexity to the application's source code, since the possible values for any of these fields-properties would affect all the others. That's not a big issue since FFmpeg won't try to transcode a video using properties that can not be combined, something that would result in a not viable video file. Finally, with the enhancement of the protocol used for the communication of the server with the client, the server could retrieve information like the client's operating system, which would inform him about his device. That could provide the user with the option to just ask for a video to be suitable for his device and specifications, without having to change any properties manually.

A really useful feature this application is not offering at this point is the ability for the user to interact with other users of the service. In the implementation described before in this thesis, the only one who has access to a folder is the user that created it with his registration. What could be added here is the possibility for him to give at least read privileges to others, so that he could also share his videos. That could also lead to the implementation of live streaming between users, although it would require another protocol to be used.

## REFERENCES

1. (n.d.) Retrieved on September 2013, from theworldstechnology.com: http://www.theworldstechnology.com/company.html

2. (n.d.) Retrieved on August 2013, from lucernex.com: http://www.lucernex.com/files/index.php/blog/next-generation-technology

3. (n.d.) Retrieved on August 2013, from computingcloudstorage.com: http://www.computingcloudstorage.com/cloud-computing-customer-services/

4. (n.d.) Retrieved on July 2013, from elementsolutions.com: http://www.elementsolutions.com/2013/08/08/part-2-cloud-computing-demystified-3s-4d-5e-is-all-you-need-to-know/

5. (n.d.) Retrieved on October 2013, from prweb.com: http://www.prweb.com/releases/2013/6/prweb10833469.htm

## FURTHER READING

[1] Di Giacomo, D., Brunzel, T. (2010), Cloud Computing Evaluation – How it Differs to Traditional IT Outsoursing, Master Thesis (Jönköping University), Jönköping

[2] Saleem, R. (2011), Cloud Computing's Effect on Enterprises...in terms of Costs and Security, Master Thesis (Lund University), Lund

[3] Skúlason, R. (2011), Architectural Operations in Cloud Computing, Master Thesis (University of Iceland), Reykjavik

[4] Bakker, J. (2011), The benefits of cloud computing in IT intensive organizations, Master Thesis (Erasmus School of Economics), Rotterdam

[5] Calvert, K., Donahoo, M. (2008), TCP/IP Sockets in Java, Elsevier Inc., Morgan Kaufmann Publishers, London

[6] Retrieved June 2013, from cisco.com:

http://www.cisco.com/web/about/ac123/ac147/archived_issues/ipj_12-3/123_cloud1.html

[7] Retrieved September 2013, from technet.com:

http://blogs.technet.com/b/yungchou/archive/2011/03/03/chou-s-theories-of-cloud-computing-the-5-3-2-principle.aspx

[8] Retrieved September 2013, from technet.com:

http://blogs.technet.com/b/kevinremde/archive/2013/05/28/apr2013.aspx

[9] Retrieved May 2013, from microsoft.com:

http://www.microsoft.com/en-us/server-cloud/hyper-v-server/default.aspx

[10] Retrieved May 2013, from openstack.org:

http://www.openstack.org/

[11] Retrieved December 2012, from eyeos.com:

http://www.eyeos.com/

[12] Retrieved November 2013, from webopedia.com:

http://www.webopedia.com/quick_ref/cloud_computing_terms.asp

[13] Retrieved November 2013, from webopedia.com:

http://www.webopedia.com/quick_ref/cloud_computing.asp

[14] Retrieved November 2013, from nist.gov:

http://csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf

[15] Retrieved July 2013, from virtuallycloud9.com:

http://virtuallycloud9.com/index.php/2013/04/build-your-private-cloud-installing-virtual-machine-manager-step-by-step/

[16] Retrieved October 2013, from wikipedia.org:

http://en.wikipedia.org/wiki/Transcoding

[17] Retrieved October 2013, from wikipedia.org:

http://en.wikipedia.org/wiki/Lossy_compression

[18] Retrieved October 2013, from ffmpeg.org:

http://www.ffmpeg.org/documentation.html

[16] Retrieved December 2012, from baylor.edu:

http://cs.baylor.edu/~donahoo/practical/JavaSockets/textcode.html