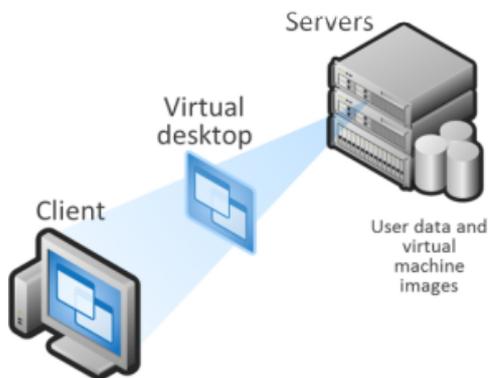


Bachelor Thesis 2016

Virtual Desktop Sizing



Source: <http://bit.ly/29WkyW>

Student : Aleksandr Mikheev

Professor : David Wannier

Department : Business Information Technology

Submitted on : 22th July 2016

ABSTRACT

This paper is intended to describe a process of choosing and building a demo of virtualization solution for a group of people in campus of EPFL located in a city of Sion, Valais Wallis. The aim of the paper is to compare different desktop virtualization solutions and choose the best one within the given customer's requirements and infrastructure. The second goal of the paper is to provide a demo implementation of the selected virtualization solution with guidelines describing how it was created and notes on specific customizations required within the given internal computer structure of the campus.

The document presents some theory behind desktop virtualization specifically and the problem customer has with the current solution (chapter 1), comparison of different systems and description of the selected system (chapter 2 and 3) and how the demo solution was implemented within given requirements (chapters 4-6).

Most of the work is based on official specifications and documentation for desktop virtualization system. Since this thesis is a project made for a customer, the main deliverables of it are demo implementation and instructions on how it was created (Appendix I).

Keywords: desktop virtualization, VMWare Horizon, virtualization platform.

FOREWORD AND ACKNOWLEDGEMENTS

Virtualization in computer science is a large topic. There are multiple types of virtualization and they perform differently, sometimes completely, if compared on how they work. As an author of the paper, my main intention in selecting this topic was that desktop virtualization nowadays includes most of those types of virtualization combined together within a single platform. Usually the structure of desktop virtualization starts from a generic virtualization platform, which creates an abstraction for running different types of virtual machines, and goes to the actual virtualization of a full user's computer, which, if examined, hardly differs from a computer running on a real hardware. Since desktop virtualization evolved, it started to gain more features. For example, now it also gives abilities to virtualize not only a complete computer but even a single application, which gives broad possibilities for computer systems of the future.

Another reason in choosing the topic was that I see changes in how computer systems work over time and it seems to me that desktop virtualization is the next trend there. For the last few years all intensive computations were moved to a "client-server" model, where end-user's computer doesn't actually perform tasks but only provides an interface for displaying and interacting with data while doing computations on a separate server. Desktop virtualization brings this idea even further: instead of user interacting with a client user interface inside an operating system, which runs on a real hardware, user starts his work by running a client user interface for the operating system itself. As an author of the project, I found it very useful to gain experience in implementing such a solution because it seems logical to me that this idea will eventually replace usual work stations.

This topic was done as a working project accompanied with the thesis paper. During project development the most problematic were technical difficulties because such systems are still new and for some topics there is not enough information. These problems were eliminated by testing different possible implementations and finding those that work the way it was defined by requirements.

I would like to thank Voiselle Joël Gilbert André Robert Emmanuel for providing me with the opportunity to do this research inside EPFL's campus in Sion and helping in all matters, which required customer's involvement. Also, I would like to thank my thesis coordinator, David Wannier, who helped me with choosing the right thesis topic and provided help on the first premises.

TABLE OF CONTENTS

ABSTRACT	I
FOREWORD AND ACKNOWLEDGEMENTS	II
TABLE OF CONTENTS	III
LIST OF TABLES	V
1. INTRODUCTION	1
1.1. VIRTUALIZATION AND DESKTOP VIRTUALIZATION	1
1.2. CUSTOMER AND EXISTING CUSTOMER’S SOLUTION	4
1.3. PROBLEM TO BE ADDRESSED DURING THE PROJECT	6
1.4. PROJECT DEVELOPMENT.....	6
2. CHOICE OF A SYSTEM	8
2.1. GROUPS OF VIRTUALIZATION SYSTEMS	8
2.2. COMPARISON CRITERIA OF PRODUCTS EXPLAINED.....	9
2.3. SYSTEMS COMPARISON AND COMPARISON CONCLUSIONS	11
3. SYSTEM ARCHITECTURE.....	13
3.1. SYSTEM COMPONENTS.....	14
3.1.1. <i>Active Directory</i>	15
3.1.2. <i>ESX Host</i>	15
3.1.3. <i>vCenter Server</i>	15
3.1.4. <i>View Connection Server</i>	16
3.1.5. <i>Client workstation</i>	17
3.2. DIFFERENT APPROACHES TO IMPLEMENTING SYSTEM ARCHITECTURE	17
3.2.1. <i>First architecture</i>	18
3.2.2. <i>Second architecture</i>	20
4. SYSTEM IMPLEMENTATION	22
4.1.1. <i>Preparing the hardware</i>	22
4.1.2. <i>Setting up ESXi</i>	22
4.2. SETTING UP ESXI USING VSPHERE CLIENT	24
4.3. SETTING UP VCENTER SERVER	24
4.4. SETTING UP HORIZON VIEW CONNECTION SERVER	25
4.5. TESTING THE IMPLEMENTED SYSTEM.....	26

5. PROJECT MANAGEMENT	28
5.1. PROJECT ORGANIZATION (METHODOLOGY)	28
5.2. COMMUNICATION WITH THE CUSTOMER	28
5.3. MAINTAINING TASKS	28
5.4. WORK ORGANIZATION	29
5.5. DISCUSSION ON THE RESULTS BETWEEN ITERATIONS OF TASKS	29
5.6. RELEASE ROADMAP	30
6. PROJECT REALIZATION	31
6.1.1. <i>First month</i>	31
6.1.2. <i>Second month</i>	32
6.1.3. <i>Third month</i>	34
7. CONCLUSIONS	35
7.1. WORK CONCLUSION	35
7.2. POSSIBLE IMPROVEMENTS	35
7.3. PERSONAL CONCLUSION	36
REFERENCES	37
APPENDIX I: SETTING UP VIRTUAL DESKTOP INFRASTRUCTURE WITH VMWARE HORIZON	39
APPENDIX II: HORIZON DESKTOP VIRTUAL MACHINES TESTING INSTRUCTIONS	62
AUTHOR'S DECLARATION	69

LIST OF TABLES

Table 1. Groups of desktop virtualization providers 8
Table 2. Desktop virtualization platforms comparison..... 11

LIST OF FIGURES

Figure 1. Virtualization defined visually 1

Figure 2. Desktop virtualization 2

Figure 3. Some of the characteristics of desktop virtualization solutions 3

Figure 4. Components within Horizon 7 infrastructure 14

Figure 5. Selected "recommended topology" from VMWare for the vCenter Server 16

Figure 6. First system architecture design 18

Figure 7. vSphere Web Client - Remote Console feature 19

Figure 8. First system architecture design 20

Figure 9. ESXi vSphere Host Client administrative panel 23

Figure 10. vSphere Client main window 24

Figure 11. vCenter Server with ESXi host added 25

Figure 12. Horizon View Connection Server with desktop pool installed..... 26

Figure 13. Burndown chart 30

Figure 14. Horizon 7 Connection Server installation wizard 32

LIST OF ABBREVIATIONS

Cloud platform (cloud provider) - company that offers remote servers hosted within the company's infrastructure.

DHCP server - a server that automatically provides client computers with IP addresses.

Emulation - process of enabling one computer to look and behave like it is another computer.

Fully qualified domain name (FQDN) - complete domain name for a specific computer within some network.

HyperThreading - technology by Intel, which makes all processor cores to be doubled, with some restrictions on how this “doubled” cores work.

Internal network (intranet) - network belonging to an organization that is accessible only by organization's members.

IP address - a semi-unique numerical identification label of a computer.

SSL - security technology for encrypted communication between receiver and sender.

Virtual desktop - a client computer that is emulated with a virtualization solution.

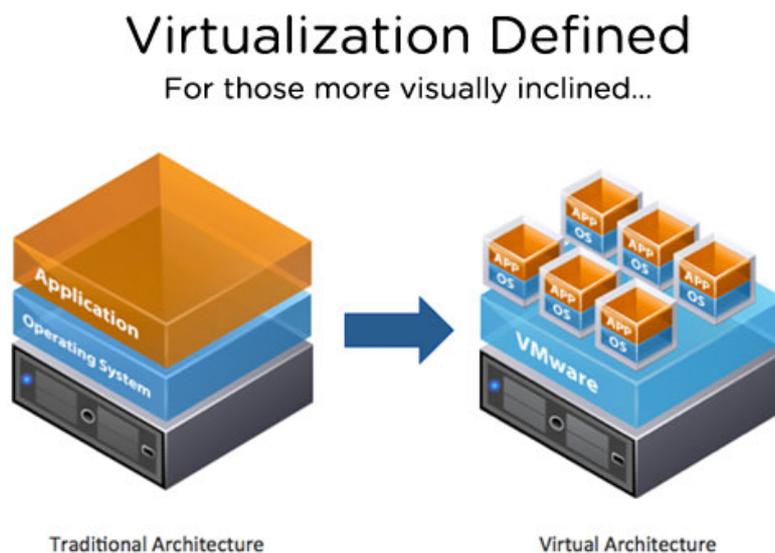
1. INTRODUCTION

Although terms like “virtualization” and “virtual desktop” are commonly used throughout all Computer Science and Information Technologies, finding complete, yet understandable, definitions for them can be hard to achieve. The main reason for that is based on complexity of the term “virtualization” because it is not a new term and it grew throughout the time, initially appearing in 1960s. Nowadays the term includes many areas and, depending on the subject area and implementations, it can have very broad meanings. According to Graziano (2011, p. 3-4), the most drastically changes in virtualization occurred throughout 2000’s, “[when] x86 based systems became more powerful”. For this paper, different definitions of virtualization and virtual desktops were reviewed, selected and their key characteristics were combined together to form an overall picture specific for the subject area and required project’s goals.

1.1. Virtualization and desktop virtualization

Amit Singh (2004) provides, as he says, a “loose definition” for virtualization as a term. This is one of the most understandable and complete definition that can be found at the time of writing: “Virtualization is a framework or methodology of dividing the resources of a computer into multiple execution environments, by applying one or more concepts or technologies such as hardware and software partitioning, time-sharing, partial or complete machine simulation, emulation, quality of service, and many others”.

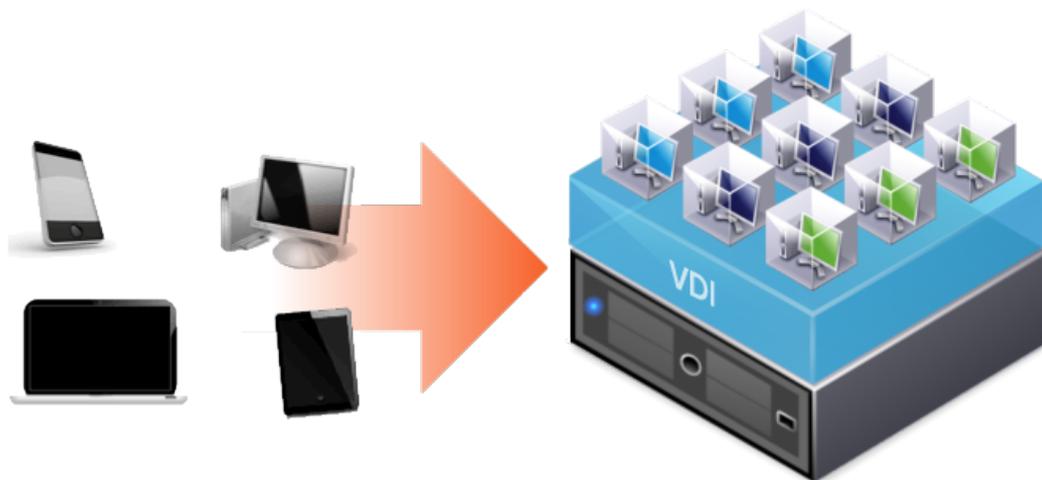
Figure 1. Virtualization defined visually



Source: (Al-Aqqad, 2014)

Although Singh’s definition contains technical wording, if re-phrased and re-compiled, the term becomes clear. Basically, what is meant under “virtualization” is division of computer resources to provide a simulation of multiple computers running simultaneously while having only a single real computer. Virtual computers (which are actually represented by virtual machines) act like any real computer: they have an operating system and software. Software runs within virtual machine’s operating system and it is isolated from the hardware that actually executes it. For the software to run correctly, it still usually needs some hardware, therefore virtualization platform provides some simulated hardware. There are also many other types and characteristics of virtualization, and how it can be applied, but, for the purpose of the paper, we will use this definition. As a simple explanation of virtualization, figure 1 can be used for reference: there is a simple example of how virtualization looks like compared to traditional computer architecture from a high-level perspective.

Figure 2. Desktop virtualization

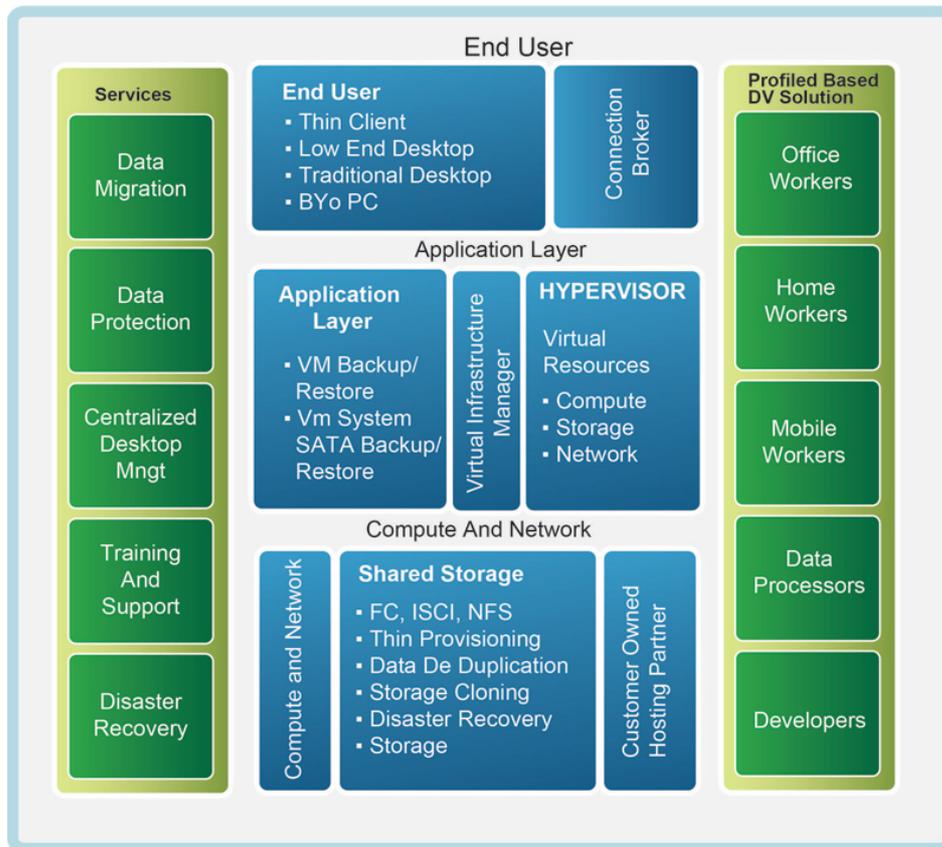


Source: (Pure Storage, 2016)

Desktop virtualization is the concept of isolating a logical operating system (OS) instance from the client that is used to access it. (TechTarget, 2011) The idea behind desktop virtualization is that user uses some simple interface for accessing and interacting with a computer, which exists only virtually, running on a server connected to the interface through a network or other kind of connection. On contrary to “client-server” communication, in desktop virtualization the user doesn’t change his workflow to the new environment because all interaction looks and performs like user interacts with a personal computer, not a centralized server. As can be seen on figure 2, the overall picture resembles the same way as with more generic virtualization shown on figure 1 but, instead of

virtualizing some parts of a normal computer, the virtualization provider creates a full emulation of this computer.

Figure 3. Some of the characteristics of desktop virtualization solutions



Source: (Locuz, 2012)

As shown on the figure 3, there are many characteristics of what desktop virtualization provides, and, therefore why it can be beneficial for the customer. Main characteristics useful for this project and explanation why they are useful are as follows:

- independency from hardware, which includes possibility to change components of user's virtual computer fast and a tolerance to the client's local hardware failure (all data persists on the server)
- centralized control over client computers: all components of company's infrastructure can be managed through a single point of access easier

- control over hardware resources, e.g. they can be used for other tasks during hours when there is small demand from users. For example, resources can be used for other computation tasks like scientific computation
- since all resources are at a single point, there is no need in over-estimating them because all users don't have the highest resource demands at the same time, therefore resources can be brought more efficiently with idea that even for "peek moments" server doesn't need the same resources as if all client computers were standalone systems, which require to have "peek" resources to be available at a client machine all the time

1.2. Customer and existing customer's solution

The customer for the project is a group in the Valais Wallis campus of École polytechnique fédérale de Lausanne (EPFL), which is dedicated to scientific research in energetic optimization in urban systems and industrial processes. Most of the work happening in the group is research produced over long periods of time: some people work there for up to 4 years but some can work for only a few months. Normally there are up to 40 people using their desktop computers on a daily basis. Because of different timeframes when people come and go, it becomes hard to maintain all computers in a decent state and re-install environment for new users who might come for only 3 months.

Currently people working at the site use desktop computers for connecting to EPFL network and authenticating against Active Directory. Most of the users use the same applications to work on their projects and rarely install other software. Because sometimes users work on their projects for years, it is important to save user-generated information, which can be anything from a few small files to gigabytes of data requiring special software.

All computers in the laboratory have pretty much the same hardware and run within the same network, which is provided through a direct connection to the network of EPFL in Lausanne. The network speed is fast enough for accessing shared resources on internal network but it can be a problem if the whole desktop operating system should be transferred from the Internet to the user. Also, regarding network, all computers have a dynamic external IP address, which changes every 1.5 hour.

For the reason of implementing a virtualization environment a server, Dell PowerEdge R730, was purchased. It has 8 hard drives, each 1,2 terabytes in the size, 2 Intel Xeon 14-core processors with HyperThreading technology, 256 gigabytes of memory and a networking capabilities of 4 connections by 10 gigabits per seconds plus 2 network adapters capable of a speed 1 gigabit per second. Initially, that server would be used for scientific projects requiring specific virtual machines. Later on, when

Aleksandr Mikheev

the project completes, if the capabilities of it will be found satisfying, it will be also used as a main server for the desktop virtualization.

Main contact with the campus was Voiselle Joël Gilbert André Robert Emmanuel, who is in charge of IT in this research group. His tasks contain developing, maintaining and extending the optimization platform along with maintaining virtual machines and general IT help.

1.3. Problem to be addressed during the project

Because there are many new users coming and leaving the research group every few months, it is not feasible to reinstall manually operating system and all software on the desktop computer in order to have a new, clean installation on the machine for the new users. Right now all computer maintenance is done manually and it is a big problem for the IT support. Using desktop virtualization, such tasks could be automated and new users would be able to start working a few hours, or even minutes, after they request a new machine.

Also, some computations require a lot of computer power. Executing such computations on a client computer is nearly impossible because of the time required for them to complete on a regular computer. If a virtualization solution is used, these computations could be executed on a server, for example, during the night, when desktop virtualization is not used and there are free resources on the virtualization server.

The client defined requirements for the virtualized desktop computers as follows: a hard drive 100 gigabytes in size, 2-core processor, 8 gigabytes of memory.

Therefore, the problems to be addressed in the paper are as follow: create a demo of desktop virtualization platform, provide reasoning for whether this kind of solution would comply with the customer's requirements if executed on the purchased server and define, whether it would be possible to use such a solution as a place for computational tasks of users' projects during the night periods of time.

1.4. Project development

Although some parts overlapped with other, the project can be divided in 3 main parts, which occurred sequentially:

1. Finding, analyzing and comparing desktop virtualization solutions, which are popular nowadays. As a result of this part, one solution is selected for demo implementation.
2. Analyzing current customer's infrastructure and how a new system would fit it. This part is iterative meaning that it isn't possible to create, even a high-level view on how new system should be implemented, without trying to do it in different ways. All virtualization solutions today are complex systems, which interact with different parts of existing infrastructure and have different possible implementations. Since not all of the possible implementations can be achieved or be really useful for the customer, the main goal is to find a solution which fits customer's needs within the given environment.

3. Process of implementing a demo of virtualization solution for testing.

Although analysis and comparison of different solutions is useful for the customer, the main deliverable of the project for company is step-by-step instructions to achieve what was done in the demo implementation. These instructions should contain enough explanation of what have been done and why so that it is understandable how to use them in case of environment different from what has been given for testing purposes for the project. The complete result of this part is in Appendix I.

2. CHOICE OF A SYSTEM

There are many desktop virtualization solutions on the market right now. First of all, different solutions were gathered and grouped by where they are installed: on a hardware within an existing infrastructure (standard providers) or if they are provided by a cloud platform as a service.

2.1. Groups of virtualization systems

During the process of collection of data on cloud providers it was decided to only include well-known providers with a good 24/7 support because customer asked for maintainable solution since desktop machines are critical for the company and even a few hours of outage should not occur under normal circumstances. Also, since this infrastructure would contain all information created by users, it is key that provider can be trusted on saving and maintaining the data.

Table 1. Groups of desktop virtualization providers

Cloud providers, which have desktop virtualization for Windows included	Solutions, which should be located on a physical server within company's network
Microsoft Enterprise Mobility Suite	VMWare Horizon
Amazon Workspaces	Citrix XenDesktop
	Parallels Remote Application Server

Source: data collected by author

Looking at table 1, first thing to note is that VMWare Horizon and Citrix XenDesktop are long-known competitors inside this market. Both of products also provide cloud solutions but they are mainly famous for their solutions for virtual desktops served from the network within the company. Other companies (Parallels, Microsoft and Amazon) are new to the market: all of them just started the business of desktop virtualization from 1 to 3 years ago but the names of the companies are still valuable within general virtualization market.

After creating the table, by combining providers by this criterion, 2 main differences between this groups become clear: how price of the solution is calculated and how the solution integrates within existing infrastructure.

For cloud providers the price is a monthly fee and for standard providers the price is the price of hardware plus the price of owning the software. If considered as a setup that is profitable to evaluate for short period of time (up to 2-3 years), cloud providers are ideal because they don't require big

investments and can be cancelled any month. In a long term, it is clear that cloud providers would cost more than their competitors because their prices contain a price of renting the hardware required for virtual desktops, which is greater than buying a standalone server. As James (2015) writes: “...cloud is great for small setups but with the added complexity, you won’t get the performance/cost ratio of a dedicated server”. Also, since cloud providers are usually hard to integrate within existing infrastructure and they require some networking changes, the overall cost of ownership with them includes price of changing infrastructure and dealing with problems after those changes are made. For example, it will be hard to do user authentication users against private Active Directory with cloud providers. As for this project, implementing a demo of such a solution that would be showing how to do those infrastructural changes might not be even possible to do within a given timeframe.

For standard providers the main disadvantage is that the complete price of the hardware and software they require should be paid at the moment before implementing such a solution. Also, since standard providers integrate within existing infrastructure, they require more planning to be done and a better understanding of how they work inside. For cloud providers all internal operations are hidden: administrator doesn’t have an access to the actual hardware running virtualization platform and, therefore, this part of the job is fully outsourced to the support team.

At the moment of completing gathering information about solution providers the company already acquired a new physical server, which main goal was to enable desktop virtualization within the research group, therefore the main advantage of renting in cloud solutions was gone.

2.2. Comparison criteria of products explained

After discussion with the customer, standard providers were chosen for a more detailed comparison. Features important for the customer were selected and discussed in a matter of their importance. The importance is either “important” (red in the table) or “recommended” (green in the table). By the agreement with customer, the final solution should have all “important” features to be implemented and “recommended” features are those that can influence on the choice between systems that have all “important” features.

Linux support - along with supporting Windows as an operating system for client computers, virtualization platform should be able to run virtual machines with Linux. Mainly this is meant for running an internal web server; important feature.

Automatic environment configuration - operations like installing and configuring software on the virtual desktop should be possible to automate; important feature.

Stateful virtualization support - it essential that the data is being saved as long as the user needs it. For that reason, stateful virtualization is important because virtual desktop is saved between user logins; important feature.

Alerts about resource consumption overload - there should be a notification system that informs administrator about overload in server's resources consumption; important feature.

Full-stack solution - it is best that virtualization platform and components within it come from a single provider. That way it is possible to receive support in any questions regarding virtualization from this provider and there are no cross-interaction problems within providers; important feature.

User environment management - ideally, all data generated by user (settings, files) should be possible to retrieve and re-use between virtual machines. This way it would be possible for the customer to manage virtual machines in a "stateless" mode but still provide all advantages of a "stateful" virtual machine like; recommended feature.

Connect with any operating system - client computer connecting to the virtual desktop should be able to do it from Windows, Linux and macOS; recommended feature.

GPU computation virtualization - some projects occurring within a research group may require computations running on a graphical card. Because of that, it should be possible to have an integration with graphical card systems like NVidia GRID in order to do such computations; recommended feature.

Mobile devices support - in the future it might be good for users to be able to access their virtual desktops from mobile/tablet devices; recommended feature.

2.3. Systems comparison and comparison conclusions

Table 2. Desktop virtualization platforms comparison

Criterion	Citrix XenDesktop	VMWare Horizon	Parallels Remote Application Server
Linux support	Yes	Yes	No
Automatic environment configuration	Yes	Yes	Yes
Stateful virtualization support	Yes	Yes	Yes
Alerts about resource consumption overload	Yes	Yes	Yes
Full-stack solution	Yes	Yes	No
User environment management	Partial ^a	Yes	Yes
Connect with any operating system	Yes	Yes	No
GPU computation virtualization	Yes	Yes	No
Mobile devices support	Yes	Yes	Yes

Source: data collected by author

- a. “Partial” used in the “User environment management” for Citrix XenDesktop refers to the fact that this platform has a limited support of this feature: it is possible to save and restore user’s environment but this operation is used mainly as a backup - it does not integrate with “stateless” virtualization and, therefore, can’t be used in a scenario provided by the customer.

Data collected for the comparison in table 2 came from companies’ official websites and marketing materials. Because companies usually don’t like to disclose their missing features, some information was implied from the systems’ documentation.

Although Parallels Remote Application Server is an interesting solution because it is a new player on the market and it has many things done differently, it lacks two important features, which makes it not viable for this project. First of all, it doesn’t have support virtualizing of Linux operating system, which makes it unusable for customer’s requirements. Secondly, it is not a “full-stack” solution

because all desktop virtualization within this system happens on other provider's platforms. Basically Parallels Remote Application Server provides only the interface and monitoring for other desktop virtualization platforms like those provided by Citrix and VMWare.

When it comes to comparison between Citrix XenDesktop and VMWare Horizon, there is no clear leader: only 1 "recommended" parameter of comparison is significantly different between these products but otherwise products provide the same features for all project's requirements. This is mostly due to the fact that both companies have been competing on the market for years and they integrated features of each other during this competition.

It is worth noting that, although Citrix XenDesktop is considered a "full-stack" virtualization provider, many XenDesktop installations involve VMWare's ESXi server. According to Hidalgo (2014), this is due to "simplicity of the deployment and implementation, which translates into less hidden costs".

Although there is no clear leader among Citrix XenDesktop and VMWare Horizon, the latter one complies to requirements more precisely than the former in the part of user environment management. Also, as it turned out after the comparison, customer already used solutions from VMWare's virtualization platform, including VMWare ESXi and VMWare vCenter Server. VMWare Horizon is built upon those two products therefore it would be easier for the customer to implement and work with this solution, which can be a big benefit. Because of this, VMWare Horizon was chosen as a main system for implementation and Citrix XenDesktop was chosen as a backup variant, in case it won't be possible to create a demo of Horizon-powered virtualization system within the company's infrastructure.

3. SYSTEM ARCHITECTURE

Since VMWare Horizon 7 consists of many virtualization solutions working together, there are many decisions to be made during setting up the system. Probably the most important decisions are architectural because system architecture is very hard to alter after it was already implemented. This kind of decisions include networking, choosing of systems and design of how those systems interact. Of course, since this paper talks about a real project, all of those decisions should be made according to the customer's environment, without suggesting something that is not possible or hard to implement within existing computer system.

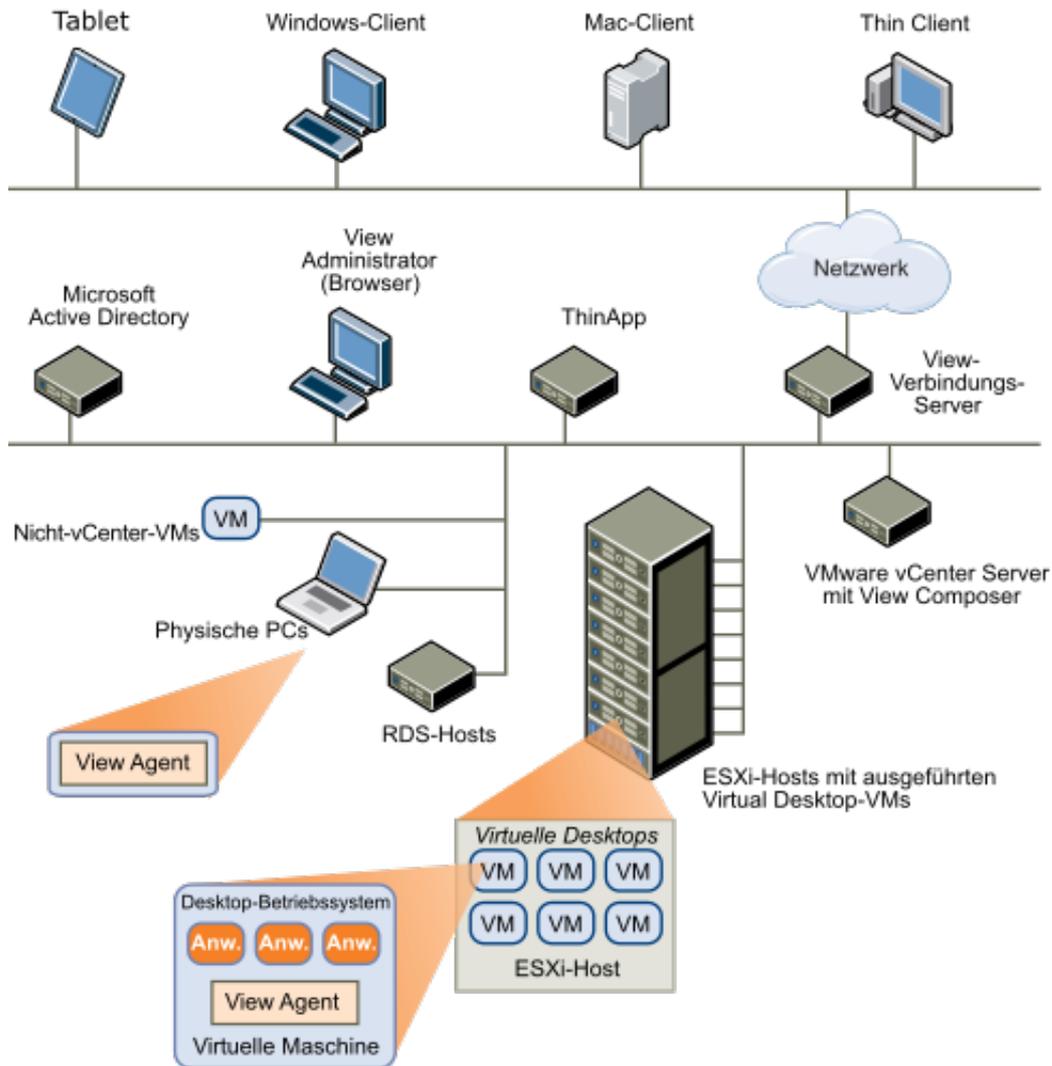
Since not all VMWare Horizon's components limitations and features are put straightforward in the documentation, a big amount of time on the project was spent for determining what is and what isn't possible to do with this system.

Among most notable limitations of selected system that isn't properly described is networking. Most of VMWare's components require a unique static IP address, which should be set up for the component's machine before the installation. Also, as it turned out during the project, most of the components do not work well with multiple networks combined together. Since EPFL's network is built on an idea that all computers have an external IP address, it would be a good solution to create a small separate "management" network for all servers and/or components of those servers. This kind of network would provide an additional level of security by protecting all highly valuable services which do not require interaction with normal users to be accessible only to a system administrator, who has an access to this kind of network. The first complete working system architecture planned was based on this idea. Later on, it turned out that, although it is possible to implement, VMWare doesn't recommend to use multiple networks within its' virtualization components, therefore other system architecture was built up, according to VMWare's best practices and how EPFL's network works.

3.1. System components

There are many components interacting within Horizon.

Figure 4. Components within Horizon 7 infrastructure



Source: (VMWare, 2016)

The picture in figure 4 contains a high-level representation of all main components in VMWare Horizon 7 infrastructure: Client devices represented by tablets, thin clients and other kinds of user computers, View Connection Server, vCenter Server, ESXi Host and Active Directory. Also it contains additional components like ThinApp and View Composer, which are not essential and can be added later.

All of the main components, on the other hand, are important to be analyzed because their correct implementation is the goal of the demo for the project.

3.1.1. Active Directory

Active Directory is used throughout solution's virtualization platform for system configuration and user authentication. First of all, Active Directory can be used as a DNS server so that some of the components within the company's intranet are at least accessible by their unique domain name and not just their IP address. Secondly, Active Directory can be used in vCenter Server as a main authentication method because this way there is no requirement on creating separate user profiles inside vCenter Server for different user roles - user accounts will be centralized and managed in the Active Directory.

Third and main role of Active Directory within VMWare Horizon is View Connection Server. This component uses Active Directory both for user authentication and computer configuration. Since all virtual desktop computers should comply to policies of the internal organization's network, View Connection Server uses existing Active Directory infrastructure for that goal. To access their virtual desktops, all users authenticate using their Active Directory credentials. Based on those credentials, View Connection Server can determine whether user is allowed to use the virtual machine and what are limitations and features given to him based on Active Directory's Group Policy Objects and View Policies defined inside View Connection Server. VMWare also provides Group Policy Objects rules templates so that features specific to View Connection Server-provided machines are also possible to be configured.

Active Directory is a part of the system, which should be hosted outside of the main server. Since Active Directory already exists within company's infrastructure, the existing system was used for the purposes of the project.

3.1.2. ESX Host

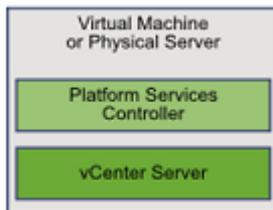
ESXi Host is a main server inside the virtualization platform. Its' purpose is to run virtual machines prepared and executed by other servers or clients. The main networking requirement for it is that it should have a dedicated static external IP address to be available to other components.

3.1.3. vCenter Server

vCenter Server - main virtual machine execution server. It is used to administer virtual machines on an ESXi Host and provides services for other more high-level execution servers and components to run and manage their own virtual machines. vCenter Server is central to VMWare's desktop virtualization because it is a big part of an overall architecture. Also, vCenter Server contains

different components which can be installed together or independently. For example, one of such a component is “Single-Sign On” - unified way of user authentication within the VMWare virtualization solution. vCenter Server should have a dedicated static external IP address to be available to other components and system administrator. It can be run inside the ESXi host it manages, therefore making it easy to

Figure 5. Selected "recommended topology" from VMWare for the vCenter Server



Source: (VMware, Inc., 2016)

As for the vCenter Server installation, because this component itself contains multiple parts described above, VMWare has different topologies recommended for installation. (VMware, Inc., 2016) Since the goal of this paper is to provide simple yet working solution, the first recommended topology was selected: one Single Sign-On domain with one Single Sign-On site and one vCenter Server with embedded Platform Services Controller within a single virtual machine. This topology supports all of the features of vCenter Server, including possibility to backup and restore configuration, and doesn't require many steps to be followed.

Depending on the company's needs, installation should be adjusted accordingly: vCenter Server's database, instead of using built-in vPostgres, can be moved to an external Oracle database. According to VMWare, this is not the case for the customer right now, because limitations of vPostgres are relatively big:

vPostgres on windows is limited to 20 hosts and 200 virtual machines. Upgrades where SQL express was installed will be converted to vPostgres. The vCenter Server Appliance supports embedded vPostgres at full scale, 1000 host and 10,000 virtual machines and is the recommended database for the vCenter Server appliance. (VMWare, Inc., 2015)

3.1.4. View Connection Server

View Connection Server - desktop virtualization server, which is in charge of all operations tied with desktop virtualization inside VMWare's platform. It provides administrating interface for all desktop and application virtualization services, maintains virtual desktops and acts as an information

broker between virtual desktops and client workstations. It also should have a dedicated static external IP address to be available to other components and users.

3.1.5. Client workstation

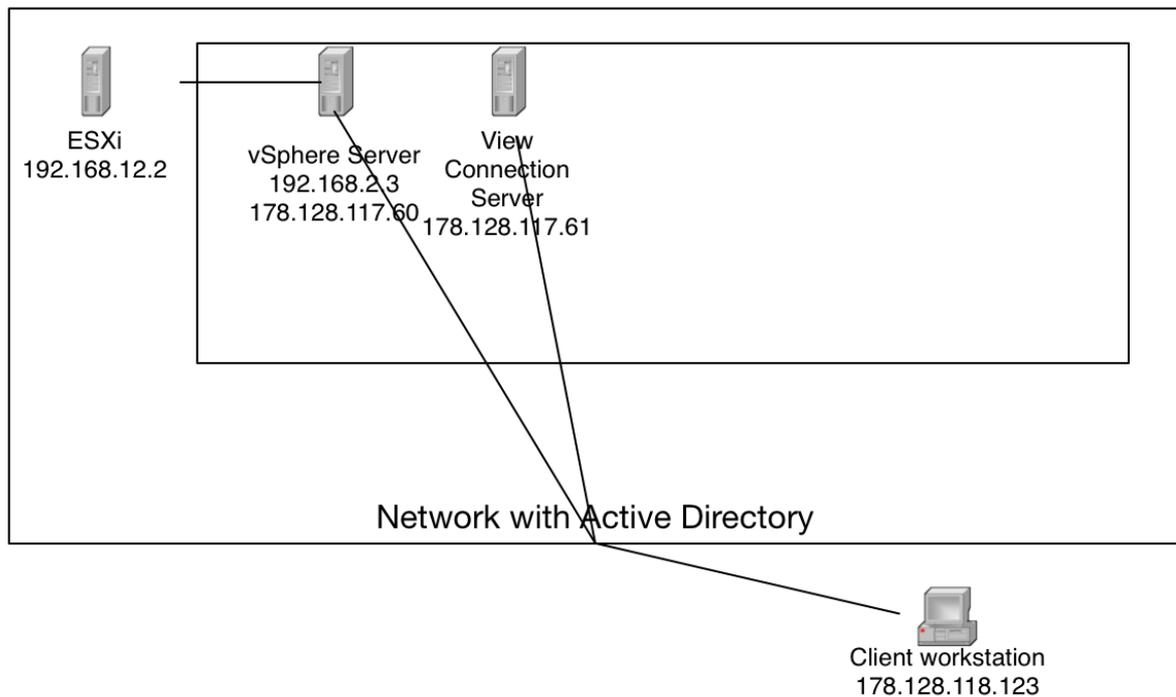
Client workstation is an interface for displaying and providing interaction with a desktop virtual machine. It can include anything from a smartphone to a Mac computer, including “thin client” computers. The workstation should be on the same network as View Connection Server but otherwise it doesn’t have any networking requirements.

3.2. Different approaches to implementing system architecture

Desktop virtualization systems, due to their complexity, can be created completely different and their architecture mostly depends on how the network in the company works and which hardware company owns. Probably because of that, VMWare doesn’t have any “reference” implementation of the overall system and the only complete picture of Horizon 7 architecture is shown in figure 5. Therefore, all components first should be understood one-by-one, how they work and what do they require. Only then they can be connected together by a system administrator. In the case of this project, the author of the paper was in charge of this work. After understanding all components and their requirements, and after applying customer’s computer environment on those components, 2 different system architectures were proposed as a result.

3.2.1. First architecture

Figure 6. First system architecture design



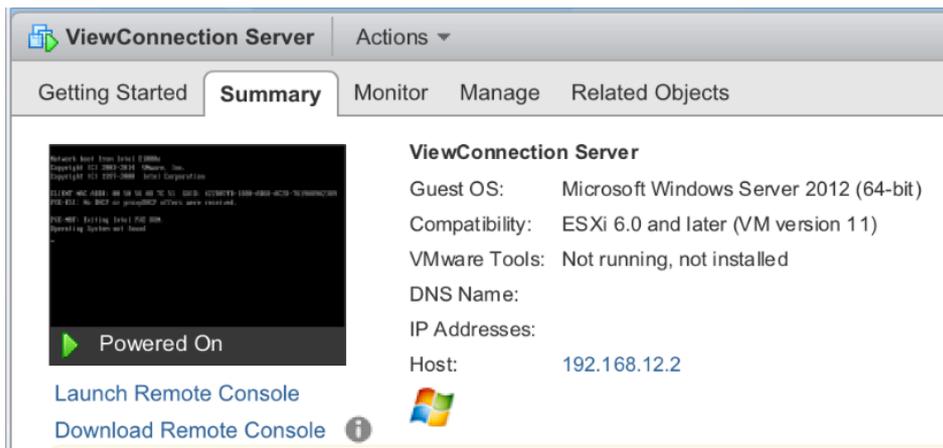
Source: image produced by author

On the figure 6 first system architecture plan is shown: network connections are represented with lines, nesting of systems is represented with boxes near the “parent” systems and examples of IP addresses are given under the components.

Initial architecture design was done according to VMWare’s “ESXi Networking Security Recommendations” in the “vSphere Installation and Setup” document. As this document states: “A management network isolates client traffic, command-line interface (CLI) or API traffic, and third-party software traffic from normal traffic. This network should be accessible only by system, network, and security administrators.”. (VMware, Inc., 2015, p. 168) The term “management network” here is used within VMWare’s ESXi server terminology and refers to the main network where ESXi is accessible. Therefore, the main idea behind this recommendation is that ESXi should be placed on a separate network, which is accessible only by server components which require access to it. In the case of this project, the only component requiring direct access to the ESXi was vCenter Server. All other server components, including vCenter Server and View Connection Server, should be placed on a static internal network.

ESXi is the only main component where authentication is only possible using a standard password. Other components provide ways to ensure security by extending it with, for example, Active Directory authentication or other secure authentication mechanisms. Because of that, it made a perfect sense to implement such an architecture.

Figure 7. vSphere Web Client - Remote Console feature



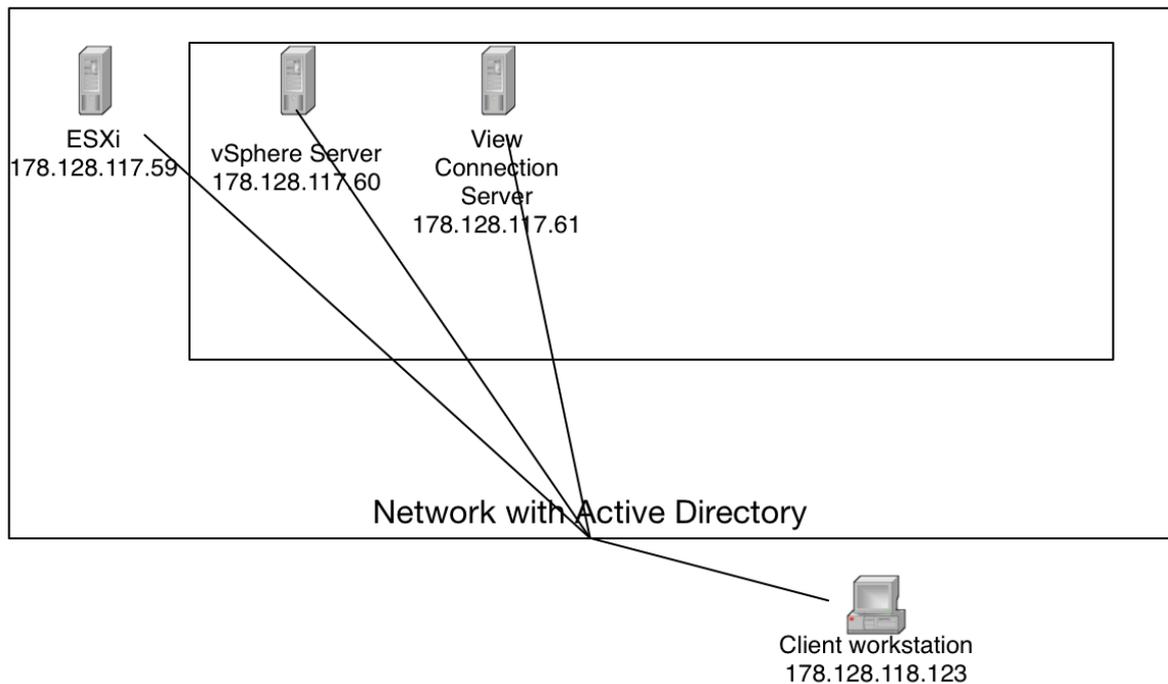
Source: author's screenshot

Although initially this design seemed to be possible to achieve, it turned out that this is not an easy task to be done. After implementing such an architecture, multiple problems occurred. First of all, some parts of vCenter Server assume that ESXi is managed through the same network as the network through which user accesses vSphere Web Client. As a result, it is not possible to use features like "Launch Remote Console" inside vSphere Web Client, shown on the figure 7, because the application launches using that button tries to reach ESXi by an address in the private network. Also, there are some communicational problems occurring between vCenter Server and ESXi when using this architecture: connection may drop any time and restoration may or may not be possible.

As it turned out later, the last problem was due to how network routing is built within vCenter Server: it considers all networks connected to as they are the only main network and routing is configured through the first network, which vCenter Server finds. I wasn't the only one who stumbled at that problem. According to Lam (2016), VMWare doesn't actually support this kind of scenario for vCenter Server: "Although it is possible to add additional vNICs to the VCSA, as far as I have been told, this is currently not officially supported by VMware.". In the article mentioned there are ways describing how to overcome this limitation but they require a good knowledge of Linux operating system and how networking there works. Also, since it's not a standard scenario, VMWare probably won't support such a configuration. Because of that, this system architecture was dropped in favor of a second one.

3.2.2. Second architecture

Figure 8. First system architecture design



Source: image produced by author

As seen on the figure 8, the second architecture looks not very different from the first one. From a high-level perspective, which is shown there, only one network connection changed: ESXi now connects straight to the network and not to the vCenter Server.

This change on the high-level picture of infrastructure actually changes a lot. First of all, ESXi gains networking communication and, therefore, receives an ability to synchronize time and data with an external network. Secondly, with this change, computers other than vCenter Server are able to communicate with the ESXi. This is convenient because, for example, if vCenter Server fails to start for some reason, there is an easy way to evaluate what have gone wrong without needing a direct access to the private network between ESXi and vCenter Server. Thirdly and mainly, this change fixes the problem mentioned in the first system architecture design: vCenter Server now has only one network.

Of course, this change comes with constrains. Mainly it makes ESXi the most vulnerable component in the system because authentication constrains in ESXi are not very configurable (only a single password exists) and also because ESXi becomes a single point of failure: if, for example, a DDoS-

attack targets it, there is no possible way to access any of the virtualization resources through the network.

To implement this architecture, a new installation of the system was carried out because there is no way to reconfigure the networking in such a way without breaking a connection through which the reconfiguration occurs. During a few weeks this installation was working without any troubles concerning system design, which was not the case with the first system architecture.

4. SYSTEM IMPLEMENTATION

After finding and testing the optimal system architecture, which complied to the customer's requirements and performed well over time, the system implementation phase of the project started. During this phase all information gathered about VMWare's virtualization platform and specific configuration required was applied to provide the testing environment. Also, while implementing the demo system, installation instructions were created, which included step-by-step instructions on how customer can implement the same system. These instructions are meant to provide sufficient information to not only replicate the demo environment but also to explain critical parts of the installation and reasoning of why certain configuration was chosen for each of the components. They are included in Appendix I. This chapter is only the summary of the final system implementation process.

For using the platform, client software is required to connect to the virtual desktops. A tutorial on using different approaches for that was created so that users can test the demo implementation. It is included in Appendix II. The tutorial was built the way that, when the system is implemented for real, there are only a few changes that are required for this tutorial to be re-used with the production system.

4.1.1. Preparing the hardware

First of all, BIOS settings on the hardware should be configured. The main options are boot settings, which depend on the hardware, and time settings - the time on the server should be set correctly and in the UTC time zone. This is important because otherwise conditions may occur, when new operations happening in ESXi will have time set earlier than operations that already occurred before. This kind of situation can corrupt the whole system.

Secondly, network interfaces should be configured properly. The server needs at least two network cards: first card should be connected to a static network and the second should be connected to a DHCP server-enabled network.

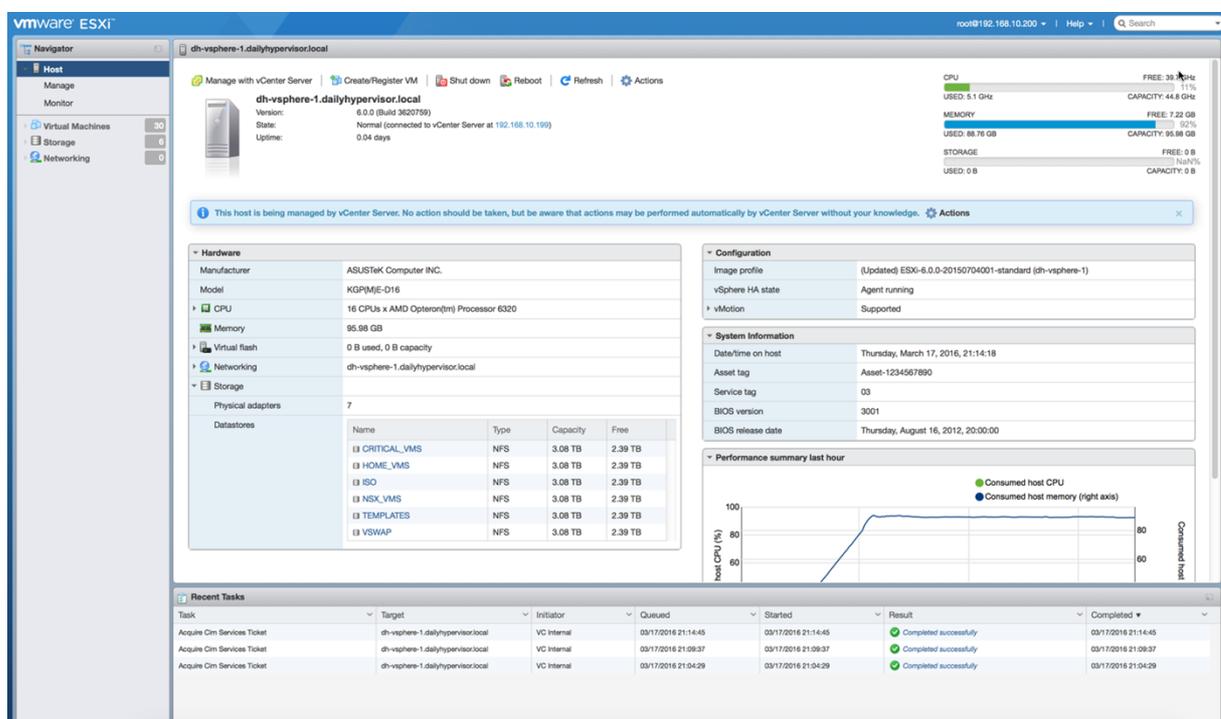
4.1.2. Setting up ESXi.

After writing ESXi installation image on a CD or USB device, the process of ESXi installation is straightforward: most of the options when installing on a fresh hardware have only one value and the most important setting is selecting a correct "Management Network" for ESXi. Since "Management Network" is a network through which clients connect to ESXi for configuration and accessing virtualization level, it is best to set a static IP address to it. Therefore, the correct setting for it is

the first network card. The second option, which should be set up properly is the “root” password for the system. It is best to set strong password of at least 12 characters long.

After installation completes, the management network should be properly configured. Since we are only using a static IPv4 address and already selected network card(s) for the management network, we need to supply information about ESXi’s IP address information. Another option to be taken in account is “DNS Configuration” - here we have to supply information about computer’s hostname and DNS servers. For easier ESXi’s integration within the network, it is recommended to set up DNS servers of the network’s internal Active Directory.

Figure 9. ESXi vSphere Host Client administrative panel

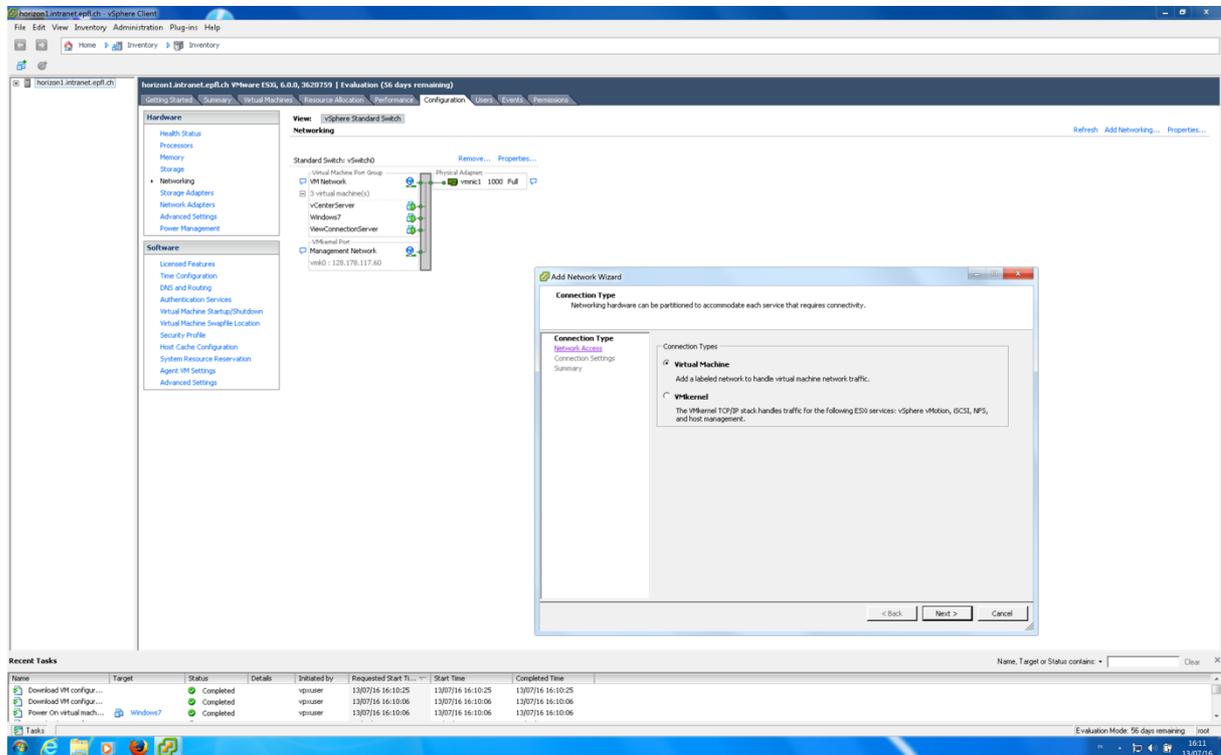


Source: (Smith, 2016)

When configuration is finished, to ensure that everything works fine, we can go to the server’s IP address in a web browser and see the ESXi’s vSphere Host Client shown in the figure 9.

4.2. Setting up ESXi using vSphere Client

Figure 10. vSphere Client main window



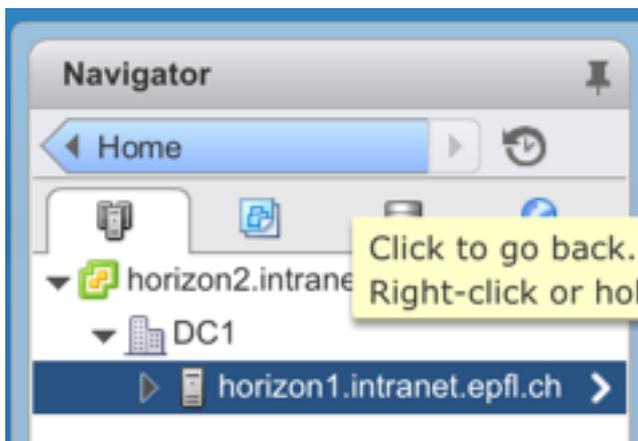
Source: author's screenshot

To properly configure the second (DHCP server-enabled) network and time synchronization, a connection from vSphere Client application (shown on figure 10) is required. After installing the software on the administrator's computer a connection to the ESXi should be done. Since by default all server components use self-generated SSL certificates, a warning about an untrusted certificate should have been suppressed. After that, a second network was created with the previously unused network interface assigned to it. Also, NTP servers were configured in order for ESXi to synchronize time from the network.

4.3. Setting up vCenter Server

vCenter Server comes in 2 editions: a standalone appliance and a server application installed in Windows Server. During the system architecture phase of the project, the appliance edition was selected. The appliance was installed on an ESXi host. After installation, the ESXi host was added to the hosts that are managed by the vCenter Server.

Figure 11. vCenter Server with ESXi host added



Source: author's screenshot

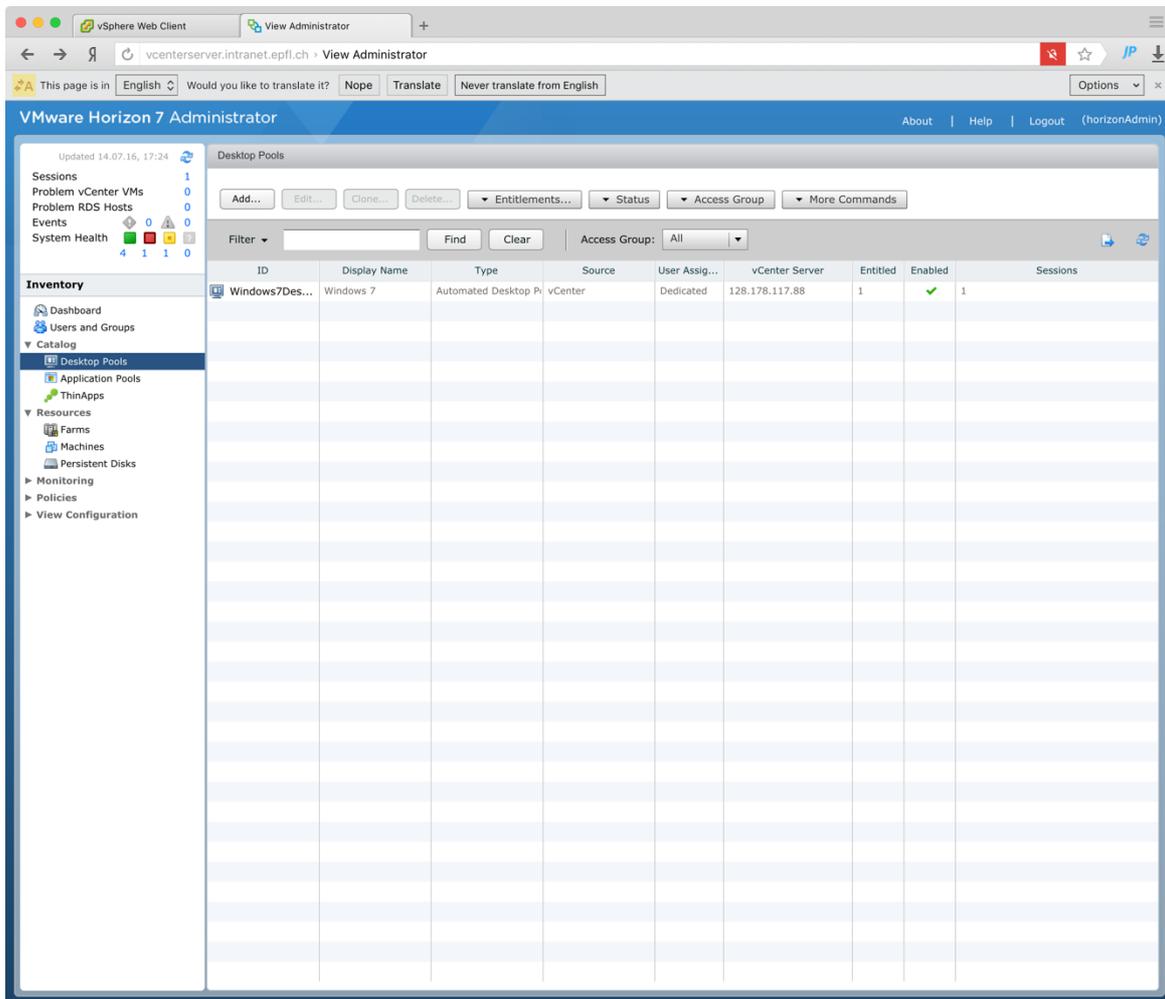
When everything is done and ESXi host was added, the “Navigator” in vCenter should look like it is shown in the figure 11.

4.4. Setting up Horizon View Connection Server

First of all, to implement a View Connection Server, a virtual machine with Windows Server was set up created inside vCenter Server. The View Connection Server has been installed on this virtual machine and set up for connection with Active Directory.

After that vCenter Server was added to the View Connection Server. Within vCenter Server a “master” virtual machine was created for the virtualized desktops. This machine was configured so that it is connected to the internal Active Directory, updated and optimized for using as a virtual desktop. With that done, Horizon Agent software was installed for the View Connection Server to be able to communicate with the “master” virtual machine.

Figure 12. Horizon View Connection Server with desktop pool installed



Source: author’s screenshot

The last step in configuring Horizon View Connection Server was the process of converting “master” virtual machine to a template and creating a “Desktop Pool” backed by this template in Horizon View Connection Server. After setting up users, who are able to access virtual machines, the implementation was complete and Horizon View Connection Server should look like on figure 12.

4.5. Testing the implemented system

The computer, on which testing environment was installed, didn’t have a big amount of computational power and, after installing all server components, had about 8.5 gigabytes of free memory, 4 free CPU cores and 90 gigabytes of free disk space. Because of that, to measure virtualization overhead on resources, two different tests were performed.

First test was focusing workload and how much additional resources virtualization takes up from the platform for each virtual machine. The test included 1 virtual machine, which have assigned with 4 CPU cores and 8 gigabytes of memory. When this virtual machine was working under heavy load, up to 95% of all resources were used on the server and notifications about low resources occurred in administrative panel of vServer Center. The virtual desktop still performed normally and there was no lag or other kind of problem. Therefore, the impact of virtualization is very small on the additional resources usage in terms of a virtual machine.

During the second test, since the first one performed good, concurrency was the main concern. For that, 2 virtual desktops were created instead, each of them having 2 CPU cores and 4 gigabytes of memory. For simulating 2 client users, connections through a browser and Horizon Client to the virtual desktop were performed. During the test, once again, no problems were encountered.

Since the client of a project also wanted to test the environment, virtual machines made for the second test were once again re-created and this environment, along with instruction for testing users, were given back to the customer.

5. PROJECT MANAGEMENT

Project management is an important part of the project because organization and planning steps that had to be made in order to achieve success. With help of planning project ahead it was possible to make right decisions regarding goals at each stage of the project and manage technical problems occurring during the project.

5.1. Project organization (methodology)

The process of project development was iterative: every few days, depending on the complexity of current iteration, completed tasks were reviewed and, if some of those were not possible to achieve within the estimated deadline, a re-evaluation of them was done in order to take actions and fix the overall project flow. As a result, during the project, something similar to Agile methodology was used although there was not a lot of documentation because there was no requirement on that in the project's goals.

5.2. Communication with the customer

All meetings with the customer were organized in person by a special appointment. The main goal of meetings was to gather customer's feedback on the current progress of the work done and to get requirements regarding next stages. These meetings were only organized during early stages of the project, when there only theoretical parts were done. All important information from meetings was noted.

Along with meetings, questions about requirements and company's infrastructure, which didn't require explanation, were asked by e-mail. All important information from e-mail conversations was also noted so that nothing is missed and the result of the project would meet customer's expectations.

5.3. Maintaining tasks

Most of the tasks on the project were sequential, therefore all tasks were placed in an ordered list where each preceding task was required to be completed before the task coming after it. This way it was easy to not only understand current state of the project but also, when changes were required, to re-organize the list and see new priorities.

The tasks in the project could be divided on "milestones" and normal tasks. "Milestones" were the big tasks, without which completing the project was not possible. Normal tasks are the tasks happening inside the "milestones", which by itself are not so important and most of them can be substituted if they were not possible to achieve. Normally all tasks should have been done but some

tasks, due to changes in planning and problems occurring during the demo implementation, were partly dismissed

5.4. Work organization

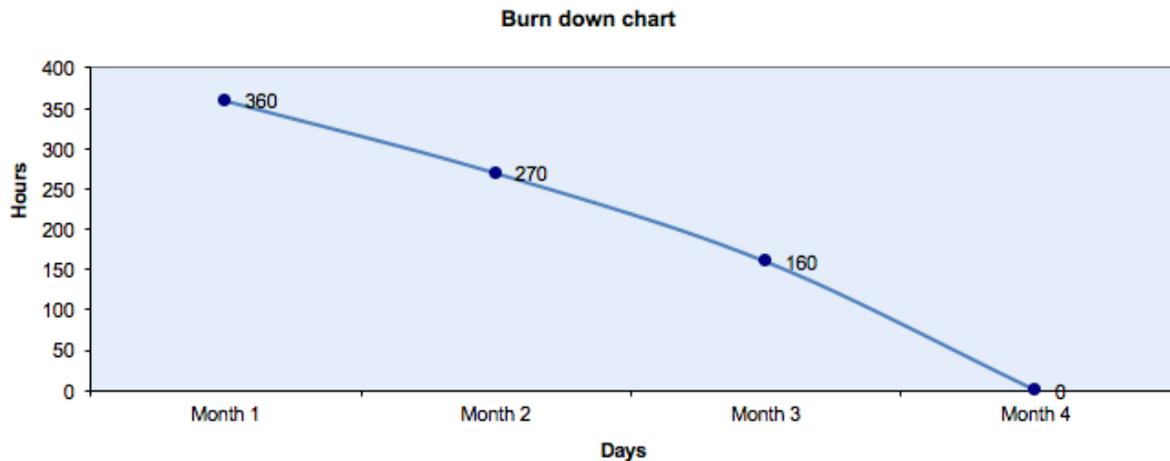
Most of the work was done alone because this project is not a group work. Normally I started with gathering theoretical information on a task by reading system's documentation and other information sources. After understanding main requirements and steps to achieve the task, I started implementing it and documenting implementation's steps that were successful. Sometimes, when there were technical difficulties, which I couldn't resolve myself due to technical limitations, which I couldn't overcome because I didn't have access rights or required materials, I asked the customer for help. This way it worked very well because I tried to do it as a last resort, and, as I think, I didn't create any excessive anxiety from my presence.

5.5. Discussion on the results between iterations of tasks

After implementation part of the project started, on one hand, there was no need in special meetings with the customer because most of the questions were answered before. On the hand, many small questions on how to implement the demo of the system started to come up. Especially this questions occurred at times when one task was already finished and the other one was on early stages of development. If it was possible, to provide customer with a context of the question, I demonstrated or described what was achieved until that moment and why I am asking this question. This way I could get both answer to my question and some feedback on what I have done to the moment. Since usually we were in the same office, there was no need to organize any special meetings for that and it occurred on a daily basis.

5.6. Release roadmap

Figure 13. Burndown chart



Source: author's image

The burndown chart (shown in figure 13) represents how much hour were spent during each month on the tasks related to the project. Of course, this is only a rough estimation but it gives a hint on how the work was distributed and whether all tasks were completed.

At the end of the project, all main tasks were done and the solution (demo implementation) was presented to the customer. Since the solution was accepted and it contained all tasks from the release roadmap, the burndown chart in the figure 13 ends at a point 0.

Because first two months' time devoted to the project work was shorter than in the third month, almost half of working hours (160 from 360) spent on the project belong to the last month of work. Partly this is also because during first two months of work there were technical difficulties not all of which were expected and, in the result, I did less tasks than I planned. Due to this, during first two months of work, I have done about a 25 hours of work less than I planned.

6. PROJECT REALIZATION

The whole project took almost 3 months to complete. Each month there was a meeting with coordinator to inform him about current status and what have been done within the month.

6.1.1. First month

A meeting with a customer was conducted long before the project started, initial goals were set in advance. Because of that it was possible to start the project and have communication with the customer via e-mail.

For the first two weeks theoretical data on virtualization providers was gathered and compared. After that, a first meeting with the customer was conducted. During that meeting we discussed information I found on virtualization providers and agreed on the comparison criteria.

The problem at this stage was that during initial requirements gathering there was no hardware purchased and, therefore, it was agreed that all virtualization providers should be equally compared. During the meeting it turned out that the hardware would arrive soon and there is already no reason to use this kind of providers within comparison. The solution for the problem was to put aside cloud providers and, since requirements changed, find more information on standard providers because part of the criteria were based on the type of the provider.

For the next two weeks the main goal was to finish comparison and start analysis of customer's infrastructure projected on the chosen virtualization solution. At this stage communication with the customer was made via e-mail. One of the risks were deadlines: because of the changed requirements and the fact that customer needed time to provide hardware for testing and demo implementation. This risk was undertaken without big problems because customer provided hardware just on time after the comparison was finished.

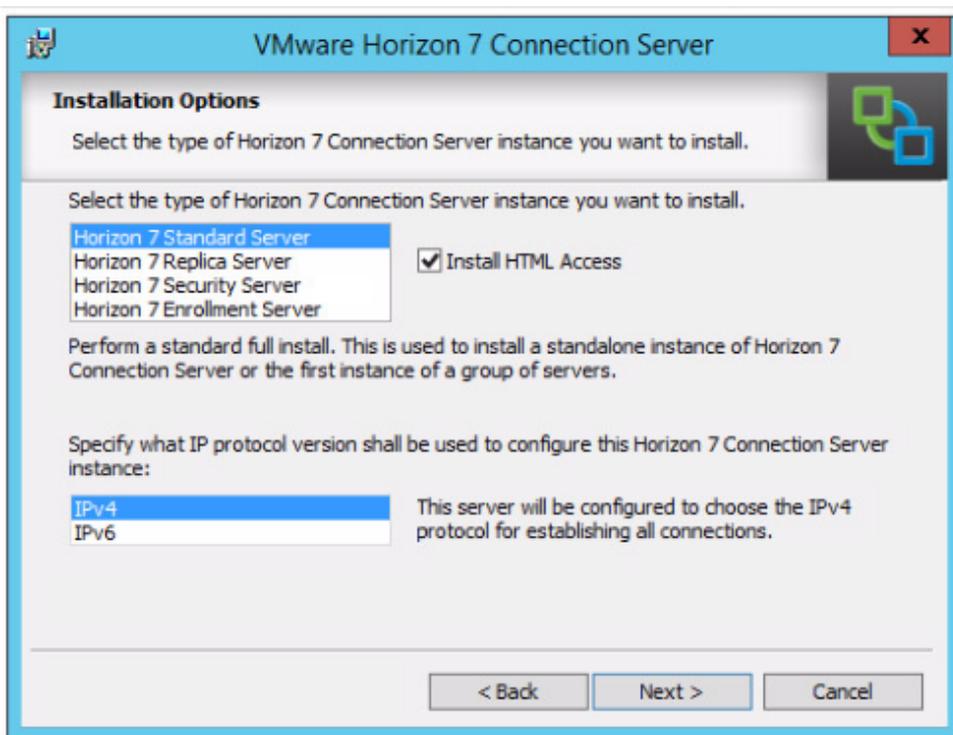
When it came to analyzing customer's infrastructure, initial time estimations were wrong. Mostly it was due to the fact that I didn't know much about how EPFL's network and authentication works and my assumptions, mostly, appeared to be wrong. Of course, this kind of problems are common within and I expected that. Because the amount of work required to implement a demo was not clear at the start of the project, and during first month I understood that it would be harder than I thought, a demo of a second system, which was planned as an optional task, was dropped from the list of tasks. Also, there were hardware limitations: from what I understood, Citrix XenDesktop required at least 2 hardware servers and I had only 1 computer which I could use for it.

6.1.2. Second month

At the start of the second month I had to finish analysis of company’s existing infrastructure and start to gather information on what had to be done to implement a demo within it.

Main parts of the infrastructure became understandable to me after a first week of analysis conducted in the previous month, therefore this month I first started with gathering information about components of VMWare virtualization platform.

Figure 14. Horizon 7 Connection Server installation wizard



Source: author’s screenshot

During analysis of server components, I first read documentation about them and then tried to implement them according to documentation. Since Horizon (View Connection Server component) is the actual system that is in charge of serving virtual desktops, I decided to go with it first. As it turned out, there are many dependencies for View Connection Server to be properly installed (installer is shown on a figure 14): it requires a separate installation of Windows server, joining computer in the Active Directory and creating accounts for the desktop virtualization-related administrators, users and computers inside the Active Directory. Because I didn’t have administrative rights inside EPFL’s Active Directory, I first tried to simulate their environment by creating a local Active Directory installation on the same computer. It wasn’t clear from the documentation but, when I tried to do

this, I met one of the first restrictions: it is not possible to have the same computer being the Active Directory server and a View Connection Server.

Because I was limited on hardware, there was a problem of running Active Directory on a separate machine like it was required from the VMWare's documentation. I've thought about my alternatives and there were only two available at the time. First alternative was to create the whole system inside a virtualized environment. The second alternative was to gather all information, what was needed to from Active Directory by the virtualization platform, create a plan, how these requirements could fit within EPFL's policies (it is obvious that I couldn't have, for example administrative rights on real computers and users inside it) and ask customer whether my plan is possible to implement inside their Active Directory. When I weighted my alternatives, I decided that, although emulating EPFL's network is not a very hard thing to do, since I don't know all of the caveats, I can miss with my assumptions and instructions for system implementation will become useless. Therefore, I first went with the second option and asked customer for enough rights so that I could test the system inside of the Active Directory in the EPFL network. As usual, Joël was fast to process my request and I got everything I needed on the next day.

Another big problem I had with View Connection Server installation was also misrepresented in the documentation. Normally, for all components, VMWare allows to use dynamic IP addresses assigned by a DHCP server combined with or without DNS names for the computers. Although it is not very recommended for a production system, for the testing it seemed to be enough because at the time I had no access to a static IP address inside the EPFL's intranet. When problem with Active Directory requirements from the View Connection Server was resolved, I received another error from the installation wizard (figure 14). As it turned out, View Connection Server was different from other server components in a way that it always enforced a requirement on a static IP address configured for the machine it runs on. When I tried to fix the issue with assigning a DHCP server-provided IP address as a static address for the machine, EPFL network stopped responding on any network requests. When I contacted the customer on the behalf of acquiring static IP address for the server, I understood what was the problem: the networking in the internal network was working different from what I expected and static IP addresses are only possible to be acquired on another, dedicated network.

When I fixed all problems mentioned, I was able to understand View Connection Server and other components of the virtualization platform became more clear to me. Therefore, I started to work on the first system architecture and tried to implement it.

6.1.3. Third month

During the third month I found that my first system architecture didn't work as I expected - there were constant problems somehow related to networking. After digging on the subject in search engines, I found an article describing my problem and solutions for it. After that I was able to create a second system architecture, which was working fine and proved to be stable.

Next, after contacting customer, I gained additional information on virtual desktops configuration and started the last part of implementation. The only problem, which occurred during this process was the required operating system on the virtual desktop. Once again, it is not mentioned in the documentation but not all editions of Windows operating systems are supported by VMWare Horizon. Since I had no access to "Enterprise" editions of those operating systems, I asked the customer for a help and he provided me with the required file.

When implementation was done, I first tested it myself, using two techniques that I could imagine and then I presented it to the customer. Since everything was working as expected, we agreed on that I would re-create the testing environment so that real users would be able to test it with their accounts.

When all of this was done, I devoted the time left for creating documentation.

7. CONCLUSIONS

7.1. Work conclusion

As a result of the study different desktop virtualization platform were reviewed and compared. The final implementation of a system requested by the customer was done and tested. During the workload test it was proven that the solution's virtualization platform impacts very small on how much additional resources it takes up for each virtual machine to run. The virtualization platform itself (vCenter Server and View Connection Server combined), according to system requirements, should not take more than 18 gigabytes of memory and up to 8 computer cores. Therefore, it was proven that the server purchased by the company for the virtualization platform should be enough to provide at least 40 client virtual desktops with only a small resource overlap between the machines. As for possibility of executing tasks not related to desktop virtualization, for example, computations, this is also possible because the whole solution runs on a generic virtualization platform (ESXi), which can also be used for such tasks.

It is important to note that the solution was accepted by the client because it conformed to all of his requirements and did exactly what was asked.

7.2. Possible improvements

Systems like VMWare Horizon are very complex and always have things to be improved. The work done during the project was limited in time and available resources. Because of that, in the resulting solution and project work itself there are already several things that could be improved if the project would be continued.

First of all, a look at system redundancy should be made because right now this system by design is a single point of failure. Of course, platforms like Horizon are well-tested and have millions of users world-wide but there are still some concerns on a possibility of a system failure, which would result in all desktop computers not accessible to the users. The redundancy itself is a large topic and it requires additional resources, like another server of the same configuration and a load-balancing server between this two servers, to be acquired.

Secondly, in a production system, SSL certificates should be set up for the servers to be accessed without SSL warnings.

Thirdly, VMWare's features like Instant Clone combined with User environment management should be reviewed more closely because they can provide even more resources savings while giving abilities to modify users' virtual desktops on-the-fly, without requiring users to manually backup and restore information they created.

7.3. Personal conclusion

As for me, this was a great experience of working inside a modern company that has resources and a willingness to change and evolve. This project gave me not only knowledge in virtualization platforms and other related technologies, it also provided me with an interesting situation where I had to analyze and understand a complex (for me) infrastructure, which serves for thousands of users every day. I had nice people helping and access inside a research group that I would never be able to visit in other cases. I would be glad if something like this would happen to me again.

REFERENCES

- Al-Aqqad, E. (2014, September 4). *Learn Virtualization & VMware vSphere from Scratch*. Retrieved from Virtualization Team: <http://www.virtualizationteam.com/certifications/learn-virtualization-vmware-vsphere-from-scratch.html>
- Conference Board of Canada. (2011, April). *U.S. and Overseas Travel to Canada, Short Term Competitive Outlook, Third quarter of 2011*. Retrieved from <http://en-corporate.canada.travel/research/trends-outlook>
- Hidalgo, M. (2014, December 17). *It is observed that VMware ESXi server is a preferred choice for implementing Citrix XenApp & XenDesktop. Why is it so?* Retrieved from Quora: <https://www.quora.com/It-is-observed-that-VMware-ESXi-server-is-a-preferred-choice-for-implementing-Citrix-XenApp-XenDesktop-Why-is-it-so>
- Hume-Pratuch, J. (2012, 05 31). *A Prescription for Success: How to Cite Product Information in APA Style*. Retrieved from <http://blog.apastyle.org/apastyle/2012/05/how-to-cite-product-inserts-in-apa-style.html>
- James. (2015, August 7). *Servers: buying hardware vs renting hardware vs cloud*. Retrieved from LoadingDeck: <https://www.loadingdeck.com/blog/servers-buying-hardware-vs-renting-hardware-vs-cloud>
- Lam, W. (2016, January 2). *Caveats when multi-homing the vCenter Server Appliance 6.x w/multiple vNICs*. Retrieved from virtuallyGhetto: <http://www.virtuallyghetto.com/2016/02/caveats-when-multi-homing-the-vcenter-server-appliance-6-x-wmultiple-vnics.html>
- Locuz. (2012). *VDI & BYOD*. Retrieved from Locuz: <http://www.locuz.com/vdi-byod.html>
- Madden, B. (2015, August 11). *Prediction: By end of 2016, VMware Horizon will be neck-and-neck with Citrix XenDesktop/XenApp*. Retrieved from brianmadden: <http://www.brianmadden.com/opinion/Prediction-By-end-of-2016-VMware-Horizon-will-be-neck-and-neck-with-Citrix-XenDesktop-XenApp>
- Pure Storage. (2016, July 3). *What is VDI?* Retrieved from Pure Storage: <http://50.56.80.148/applications/vdi/what-is-vdi/>
- Singh, A. (2004, January). *An Introduction to Virtualization*. Retrieved from kernelthread: <http://www.kernelthread.com/publications/virtualization/>

TechTarget. (2011, November). *What is desktop virtualization?* Retrieved from TechTarget:
<http://searchvirtualdesktop.techtarget.com/definition/desktop-virtualization>

VMWare. (2016). *How the Components Fit Together*. Retrieved from Documentation Center for
VMWare Horizon 7 version 7.0: <http://pubs.vmware.com/horizon-7-view/index.jsp#com.vmware.horizon-view.planning.doc/GUID-C6AD7E43-FC69-4D8C-AB97-423FC0DCDB3D.html>

VMWare, Inc. (2015, February 9). *vSphere 6 - Clarifying the misinformation*. Retrieved from VMware
vSphere Blog: <https://blogs.vmware.com/vsphere/2015/02/vsphere-6-clarifying-misinformation.html>

VMware, Inc. (2015). *vSphere Installation and Setup*. Palo Alto: VMWare, Inc.

VMware, Inc. (2016, February 13). *List of recommended topologies for VMware vSphere 6.0.x (2108548)*. Retrieved from VMWare KB:
https://kb.vmware.com/selfservice/microsites/search.do?language=en_US&cmd=displayKC&externalId=2108548

Appendix I: Setting up virtual desktop infrastructure with VMWare Horizon

Setting up virtual desktop infrastructure with VMWare Horizon

July 2016

1. INTRODUCTION

This document provides an overview of how a demo implementation of VMWare Horizon 7 was installed and configured. Also it contains information on additional topics, which, for different, mainly infrastructural, reasons, were not possible to implement fully but still were reviewed and best possible solutions were then derived by the author.

All options requiring user input, which are not shown on the screenshots and can be misunderstood, have example values provided in parenthesis. Those values are coming from the demo implementation and can be used to trace back to the implemented system so it is understandable in what they will result.

2. PREPARATION BEFORE INSTALLATION

VMWare Horizon requires 3 main server components: vSphere Hypervisor (ESXi), vCenter Server and View Connection Server. All of those components are built in a “pluggable” approach, therefore all of them can be installed in any order although, for configuration purposes, each next component requires a previous one to be already installed and properly configured.

As proved in the demo implementation, all required components can be installed within the ESXi installation, therefore only 1 “bare-metal” server is required. Along with that, there are additional components available both from VMWare and other providers: most of them add features to the required ones, including, for example, additional monitoring capabilities (vRealize Operations Manager) and possibilities for better operations automation (vRealize Automation). Among those components one can be actually useful for the desktop virtualization features expansion. View Composer is a server that can be used for creating “stateless” virtual desktops, where user’s operating system is a “link” to a “base” operating system and it is assumed that this kind of system doesn’t care about user’s personal files and can be re-created from scratch within seconds.

Before installing the solution, it is best to review and fulfill some of the prerequisites. In the next section they are described by what they are doing and whether they are important or not.

2.1. Hardware requirements

For the installation to be successful there are some minimal hardware requirements for the server:

- At least 2 network cards for DHCP and static networks access. This is due to network design inside the place, where DHCP server is only accessible on a special network where static IP addresses are not possible to setup. In case there are multiple network cards, still only 2 networks should be created by consolidating those network cards. The majority of network cards should be provided to the static network, where all server components reside, because it is the main gateway for all communication of users with their virtual machines and also a gateway for components’ management.
- There is a list of supported hardware for ESXi installation provided by VMWare: <http://www.vmware.com/resources/compatibility/search.php> It is worth noting that VMWare doesn't support other hardware outside of the given list and they should be contacted in case the server is not within the list. Otherwise, in case of troubles, even with paid support VMWare won't help you if the hardware isn't officially supported.
- Other hardware requirements include CPU, RAM and storage. Their usage is dependent on the amount of load but, as discussed, already existing server should be sufficient for the specified workload. It is also worth noting that one of the differences between stateful and stateless virtualization is that the latter requires more resources including: additional resources for running

“master” virtual machines, more I/O operations due to the “copy-on-write” behavior. Also, stateless virtualization uses less storage space. First of all, it happens because files from “master” virtual machines are not copied but linked inside user’s virtual machine. Secondly, it is due to the idea behind stateless virtual machines: they are thought as not long-running computers and, therefore, they get re-created more often than their stateful counterpart.

2.2. DNS

Although it is possible to install all of VMWare's components available only by their IP-address, according to VMWare's documentation, it is best to use DNS names for each kind of "server" component: ESXi, vCenter Server and View Connection Server (Horizon). First two components do not integrate within Active Directory, therefore they require DNS entries to be created manually within the DNS server. The last component - VMWare View Connection Server should be placed on a Windows Server and acts as a part of existing Active Directory. Therefore, a dynamic DNS entry is automatically configured within EPFL's internal network (intranet.epfl.ch). This kind of entry will work without any problem, but it is still best to use a static DNS entry configured specifically for this server too.

When creating DNS records, be sure to add entries in both Forward (A Record) and Reverse (PTR Record) Zones. Otherwise, in case only Forward Zone is set up some components will fail during installation or configuration, also, upgrade of components won't possible. For the demo, only Active Directory-created records were used for all servers (horizon1.intranet.epfl.ch, horizon2.intranet.epfl.ch, vcenterserver.intranet.epfl.ch), after testing that scenario, since this kind of DNS records are not persistent, ESXi and vCenter Server were changed to use only IP addresses as identification.

2.3. Active Directory

Minimal recommended Active Directory entries for a system should look as follows (in parenthesis there are names of those units inside the testing environment):

- A group holding all vCenter Server administrator accounts (vCenterAdministrators).
- A Group holding all ViewConnection administrator accounts (ViewAdministrators).
- Administrator accounts, which should be added to the according groups above, depending on their privileges (horizonAdmin).
- Accounts for virtual machine templates 1 per template used for setting up group policies and authenticating computer within Active Directory (horizonMaster).
- A Computer account for View Connection Server (vCenterServer).
- A Group holding all users of the desktop virtualization (ViewUsers).
- User accounts, which are allowed to use remote desktop computers placed in the group holding all users of the desktop virtualization own group (horizonUser, horizonUser2).

For the demo environment, additional computer accounts (horizon1, horizon2) were used to test DNS-based setup but they are not needed in a real setup.

 Horizon Admin	User	
 Horizon User	User	
 HORIZON1	Computer	horizon1.intranet.epfl.ch
 HORIZON2	Computer	horizon2.intranet.epfl.ch
 horizonMaster	Computer	
 User 2 Horizon	User	
 vCenterAdministrators	Security Group...	
 vCenterServer	Computer	
 ViewAdministrators	Security Group...	
 ViewUsers	Security Group...	

Figure 15. Active directory testing OU (HorizonTesting) structure.

Regarding Active Directory, there are also Group Policy Administrative Template Files for View Connection Server, which can be used for optimizing and securing remote desktops. Those templates can be downloaded from VMWare Horizon 7 download page and installed in the Active Directory used for desktop virtual machines.

2.4. General requirements

During all installation and using infrastructure you should be connected to the internal company's network because all actions happen within the internal network and cannot be performed, for example, from EPFL's public Wi-Fi.

Also, the computer, from which installation of the system and administrative actions are performed, should have a modern, Windows operating system and web browser with Adobe Flash installed.

3. VMWARE ESXI

3.1. Before we begin

To start installation, a copy of “ESXi ISO image” is required. It is very recommended to download it from VMWare because many bugs were fixed and some features, like Instant-Clones, appeared only in the recent ESXi version.

Also, computer, on which ESXi should be installed, should have a connection to EPFL’s static network and a storage device connected (either a hard drive or a network drive), on which ESXi should be installed.

3.2. Preparing installation media

Since ESXi is a component installed on a bare hardware, some physical installation source is required for the setup. The source can be a CD, USB device or a network-boot device (PXE). Because of easiness in preparation and speed of installation, I would recommend to use a USB device. To write an image on it, software like “Rufus” can be used: <https://rufus.akeo.ie>

3.3. BIOS settings

Be sure that the computer used for installation is either in UEFI or BIOS mode and it doesn’t change later on. It is recommended to do new installation in the UEFI mode.

Time in BIOS should be set up according to UTC time zone. Although ESXi can be set up to use external time servers, for system stability (to avoid race conditions when previously set time is after current real time), it is advised to setup time and time zone before starting the installation.

3.4. Installing ESXi

1. After inserting boot media in the server and booting it, setup will start to load files required for installation. This process takes from 30 seconds to 10 minutes depending on the type of boot media.
2. To install ESXi, follow the on-screen instructions: start installation, accept the license and answer the questions mentioned below.
3. The installer will scan for local and network drives and will propose you with device where to install ESXi. It is usually recommended to install ESXi on a fast drive, e.g. SSD, SSD NAS or USB drive. After selecting the drive, you need to confirm that you are aware that all existing information on the drive will be gone. When the setup comes to the step where you are asked for the “Storage Device”, be sure to select the correct device for the ESXi installation.

4. After selecting the storage device, you will be prompted for selecting a “Keyboard Layout”. Be sure to select the one that is correct to the keyboard used with the server.
5. On the step, where you are asked for a “root” password, be sure to setup a strong password because it is the main connection security policy of ESXi. Usually it is better to have a password of at least 12 characters and best practice for now is to have it about 28 characters long, if only alphanumeric symbols are used.
6. After setting up the “root” password, you will be asked whether you would like to finish the installation - confirm it and wait for the completion.
7. When installation completes, you will be asked if you want the server to be rebooted. Be sure to remove installation media before confirming this step.

3.5. Configuring ESXi

Before beginning configuration there are a few things to note about ESXi configuration. In ESXi there is a “Management Network”, which is a network through which a remote or local client can connect to ESXi for configuration. Because of that, this network is better to be configured with a static IP address. Another kind of a network required for this project is a DHCP network, which will be configured after this chapter. There can be multiple NICs connected to the same Management network and this practice should be applied when there is not enough bandwidth on the current network.

ESXi provides 2 main ways of configuration: direct access using console access to the machine and configuration through a network using vSphere Client, ESXi’s vSphere Host Client or vCenter Web Client. Along with it there are also an SSH access and ESXi Shell, which can be useful when something in configuration is broken on a level that network configuration through the clients mentioned before is not possible.

Direct access is mainly used for configuring a Management network and troubleshooting options. Since right now EPFL uses IPv4 as a main networking protocol, IPv6 won’t be configured nor covered here or anywhere later in this project:

1. When ESXi boots, if you are connecting to ESXi directly, you will see a main console screen.
2. From here you need to go to “Customize System/View Logs” menu item by pressing F2 and select “Configure Management Network”.
3. First of all, select the Network Adapter(s) used for Management network in the respective menu item. Depending on whether there is a device on the other end of Ethernet cable, ESXi will mark NIC with Status “Connected” or “Disconnected”, therefore this can be used for reference to check which network adapter is representing an adaptor in the Network Adapters list.

4. Go to "IPv4 Configuration" to start setting up the Management Network settings. Here you basically have to select the network setting to be statically addressed and the network information about ESXi's IP address.
5. Go to "DNS Configuration" and configure a hostname (if any) of the computer along with DNS servers list. Since we are using an internal network with Active Directory, it would be useful to set up addresses of DNS servers in Active Directory. This way ESXi will be able to setup time through AD's NTP servers and communicate with other computers on the intranet using their internal hostname.

After setting up basic networking, all other configuration can and should be done through remote connection to ESXi because direct access is designed only for initial setup and troubleshooting problems, which can't be solved remotely.

4. VSPHERE CLIENT

Creating a new network not belonging to already setup network devices inside ESXi using vCenter Server is not possible to do. Because we need to setup a DHCP network for client machines, using a vSphere Client is required for this. Also, NTP servers for ESXi should be configured the same way.

4.1. Installation and first login

1. Installation of a vSphere Client is done relatively easy: after downloading the application (.exe file) you need to install it by double-clicking and proceeding through a usual setup wizard (shown in figure 2).

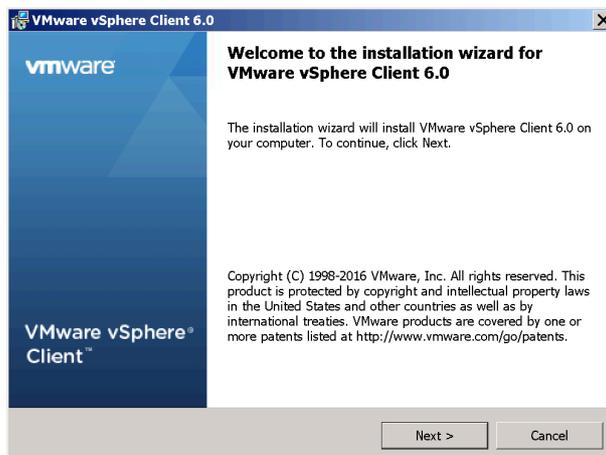


Figure 16. vSphere Client setup wizard

2. After installing vSphere Client, open the application and provide it with data from your ESXi server: IP address (128.178.117.60), User name (root) and password (Admin#).
3. When you get a “Security Warning”, be sure to check the fingerprint of the certificate used on the ESXi. If the fingerprints match, check the box for installing certificate and click “Ignore” in the window shown in figure 3.



Figure 17. vSphere Client - Security Warning

4.2. ESXi configuration using vSphere Client

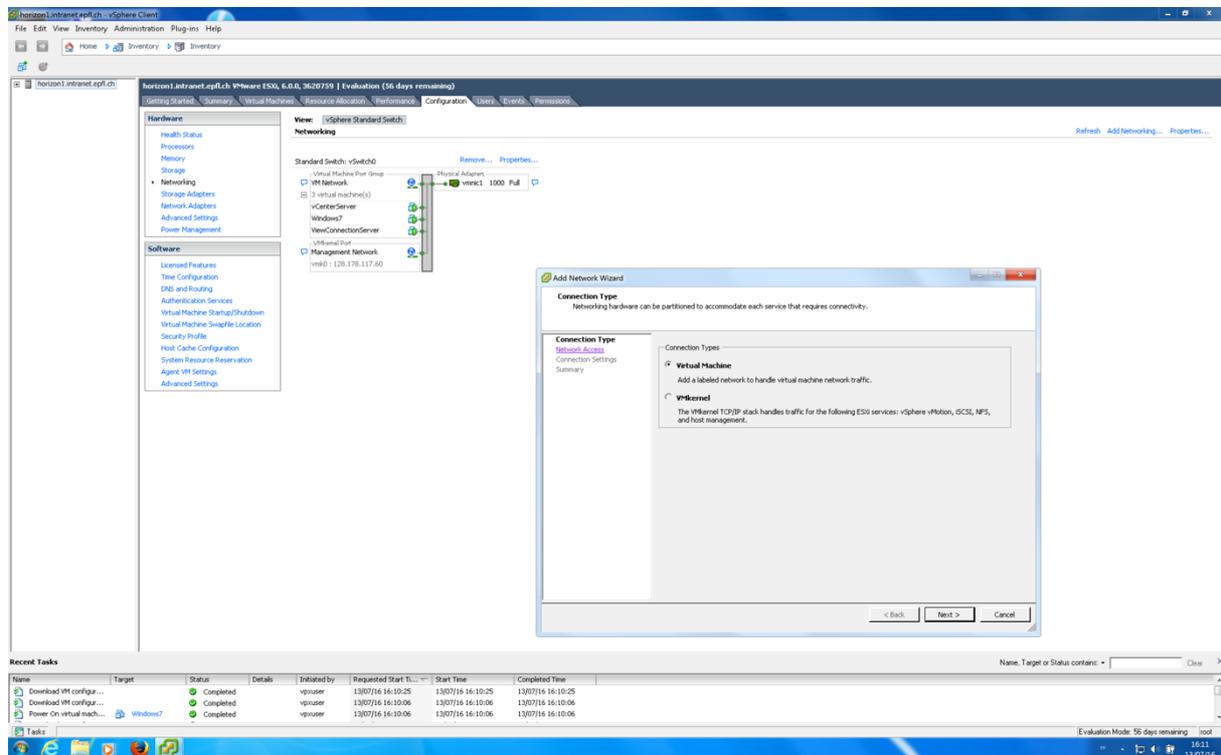


Figure 18. vSphere Client main window

1. When you see the main window of vSphere Client, select the server in the left pane and go to the "Configuration" tab in the main window.
2. In the first appeared menu "Hardware" select "Networking" and click on "Add Networking" in the right corner of the window.
3. In the new window for the "Connection Types" option select "Virtual Machine" and click Next. In the next dialog select network interfaces which are connected to the DHCP network of EPFL and click "Next" again. All other steps do not require any changes and you can just click "Next" for them. The new "VM Network" should appear in the main window.
4. To configure NTP servers, which is highly recommended for the production setup. In the "Software" menu select "Time configuration". Here you need to provide the NTP servers and select that the NTP client should be started automatically. Since ESXi resides in the Active Directory-enabled network, it is recommended to also use Active Directory's NTP servers.

5. VCENTER SERVER

There are 2 main types of vCenter installation: on a server running Microsoft Windows Server 2008 SP2 or later, or it can be deployed as a complete solution - vCenter Server Appliance. Since the second option provides the same capabilities but it is relatively easier to install, it was selected for the implementation.

5.1. Installation

1. Mount downloaded .iso file with vCenter Server Appliance.
2. Go into “vcsa” folder, run the .exe file with Client Integration Plugin and follow the setup wizard to finish installation.
3. Go into the root folder of a mounted disk and open “index.html” file.
4. When the web browser launches, allow the execution of Client Integration Plugin.

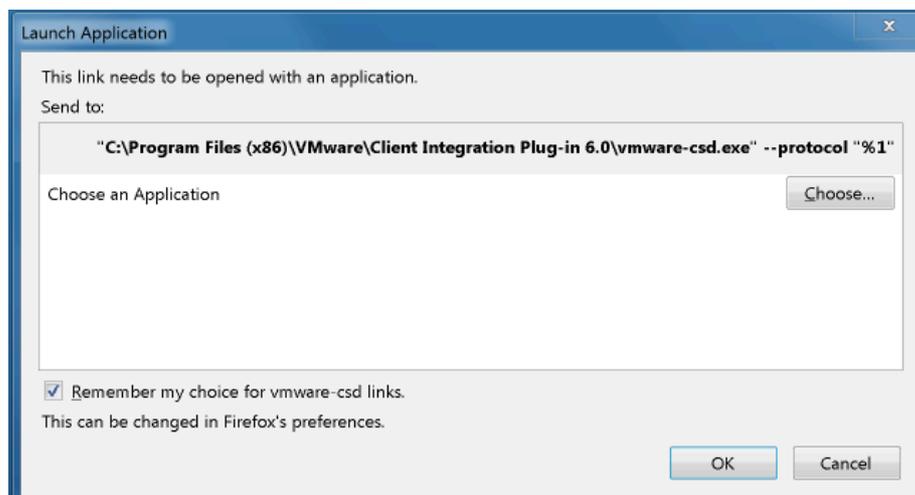


Figure 19. Client Integration Plugin execution

5. On the opened page select the “Install” option because we are going to install a new vCenter Server.
6. After accepting the license, in the “Connect to target server” window provide information about ESXi server and its login credentials. When you receive a certificate warning, be sure to check certificate’s fingerprint and accept it.



Figure 20. Certificate Warning window

7. In the “Select deployment type” select the option “Install vCenter Server with an embedded Platform Services Controller”.
8. In the “Set up Single Sign-On” select “Create a new SSO domain” and provide a password for the domain’s administrator account. The most important option here is “SSO Domain name” - it should be a unique name within your network. A good “SSO Domain name” is, for example, computer’s DNS name. In the “SSO Site name” give the name of SSO domain - usually it would be just a short name of the vCenter Server on the network.

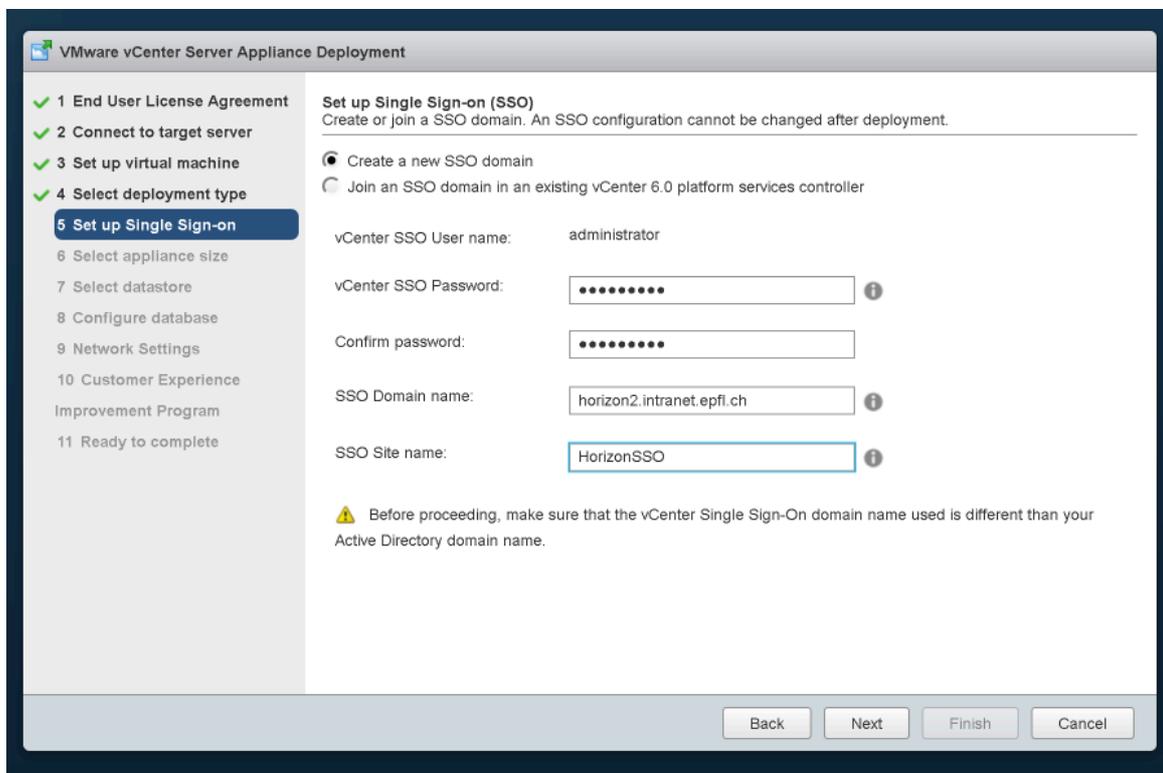


Figure 21. vCenter Server - Set up Single Sign-on

9. In the “Select appliance size” tab you need to select the size of a vCenter Server. Both “tiny” and “small” configurations should be enough for the production setup. The difference in the resource load here is really not so, it is better to choose “small”, which takes 4 CPUs and 16 GB of RAM instead of 2 CPUs and 8 GB of RAM and gives a lot of space for future system growth.
10. In the “Select datastore” usually there is only one datastore available, so choosing it is enough.
11. In the “Configure database” select “Use an embedded database (PostgreSQL)” because Oracle database is usually used only in very big environments, where vCenter Server maintains more than 10000 virtual machines.
12. In the “Network Settings” choose the first network (the management network) and IPv4 as the “IP address family”. For the “Network type” select “static” and provide network configuration for the vCenter Server. In the “Configure time sync” be sure to select the “Synchronize appliance time with ESXi host”. Also, it is possible to enable SSH access to the vCenter Server at the end of this pane. It can be useful in situations where large files need to be uploaded to the vCenter - in this case you would be able to use “scp” utility for that.

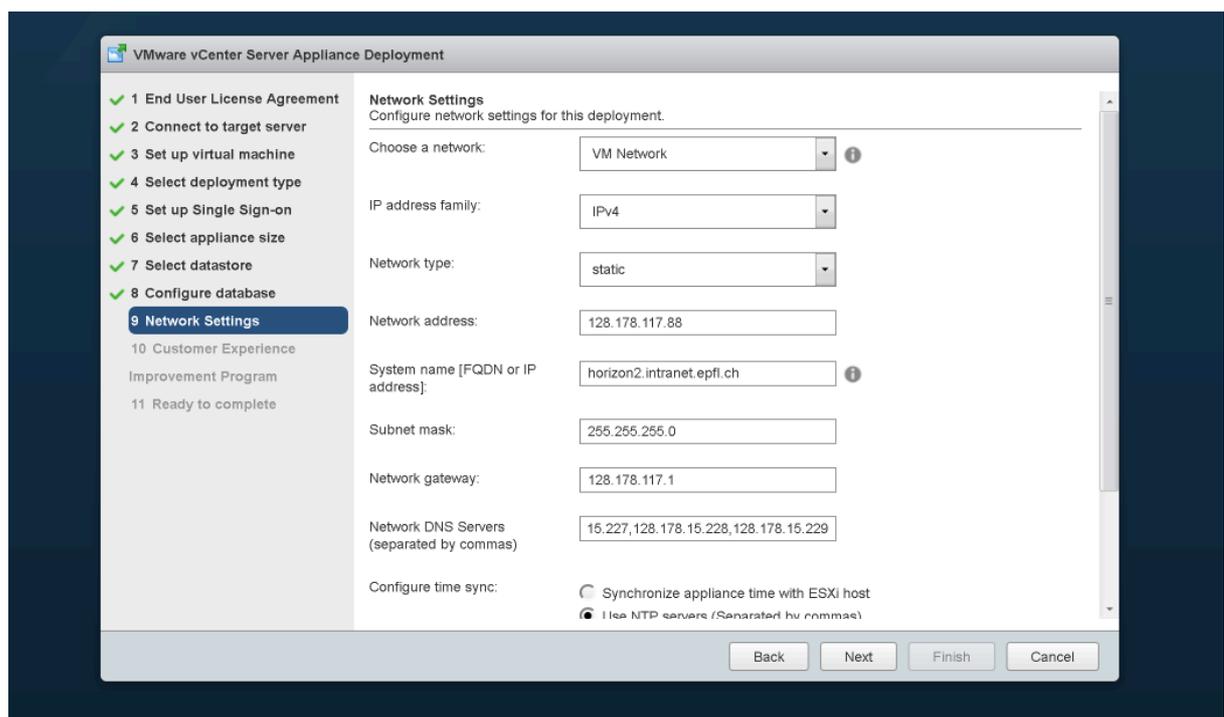


Figure 22. vCenter Server - Network Settings

13. When you get to “Ready to complete” step, be sure to review the resulting installation configuration that it complies to what you provided before.
14. After clicking the “Finish” button, the installation will start. Be sure not to interrupt it with other network actions because it can result in a broken installation of vCenter Server.

5.2. Adding ESXi host to the vCenter Server

The next step is to add the first ESXi host inside the vCenter Server. For this, go to the vCenter Server's main page, which is located at the IP address or FQDN which was set up in the previous step's "Network Settings".

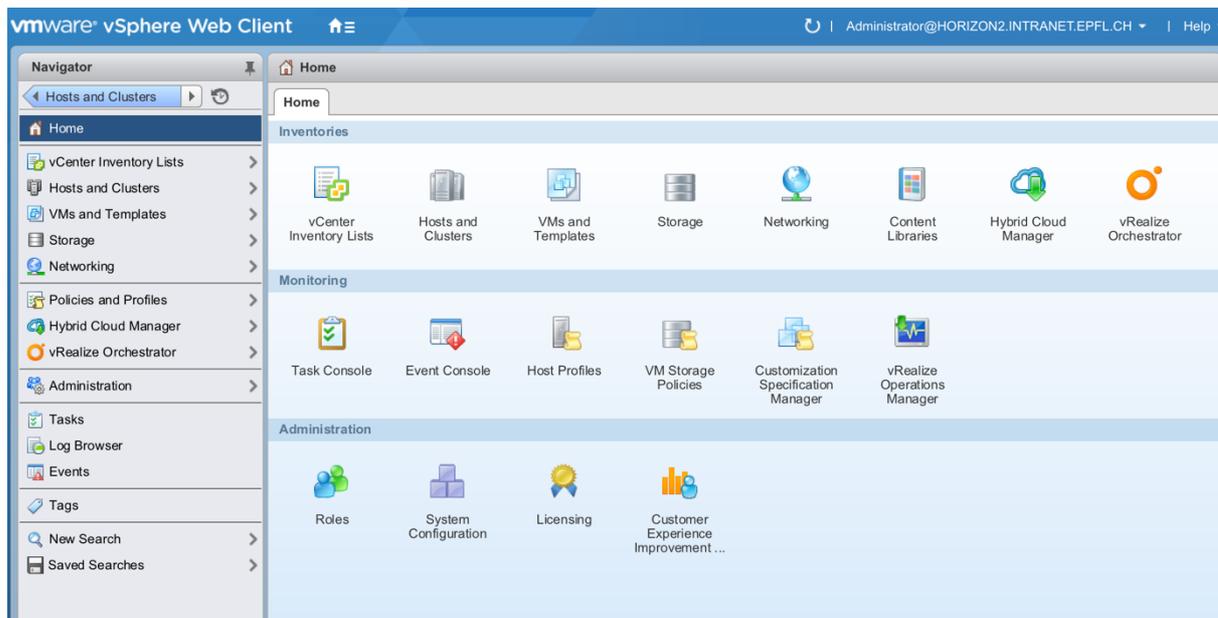


Figure 23. vCenter Server's main page

1. Within the main page of vCenter Server select "Hosts and Clusters". In the "Getting Started" dialog click on "Create Datacenter" and provide a name for the new datacenter.
2. When the datacenter is created, click on "Add Host" button and a dialog will appear. In the dialog, provide the FQDN or IP address of the ESXi host, click "Next", provide "root" user's authentication data for ESXi and click "Next" again. When you receive a certificate warning, be sure to check certificate's fingerprint and accept it.
3. Click on "Next" in the "Host summary" and in "Assign license" be sure to add a new valid license for the ESXi and select it.
4. On the "Lockdown mode" be sure to use an appropriate value: usually, it is better to select "Normal" so that only vCenter Server can access ESXi on the network and there is still a possibility to troubleshoot problems by direct connection to the server.
5. In the VM location select the datacenter created in step 1.
6. In the "Ready to complete" section click on "Finish".

6. HORIZON VIEW CONNECTION SERVER

6.1. Creating a virtual machine for View Connection Server

1. Create a new virtual machine in the vCenter Server by right-clicking on the vCenter's datacenter and selecting "New Virtual Machine->New Virtual Machine".

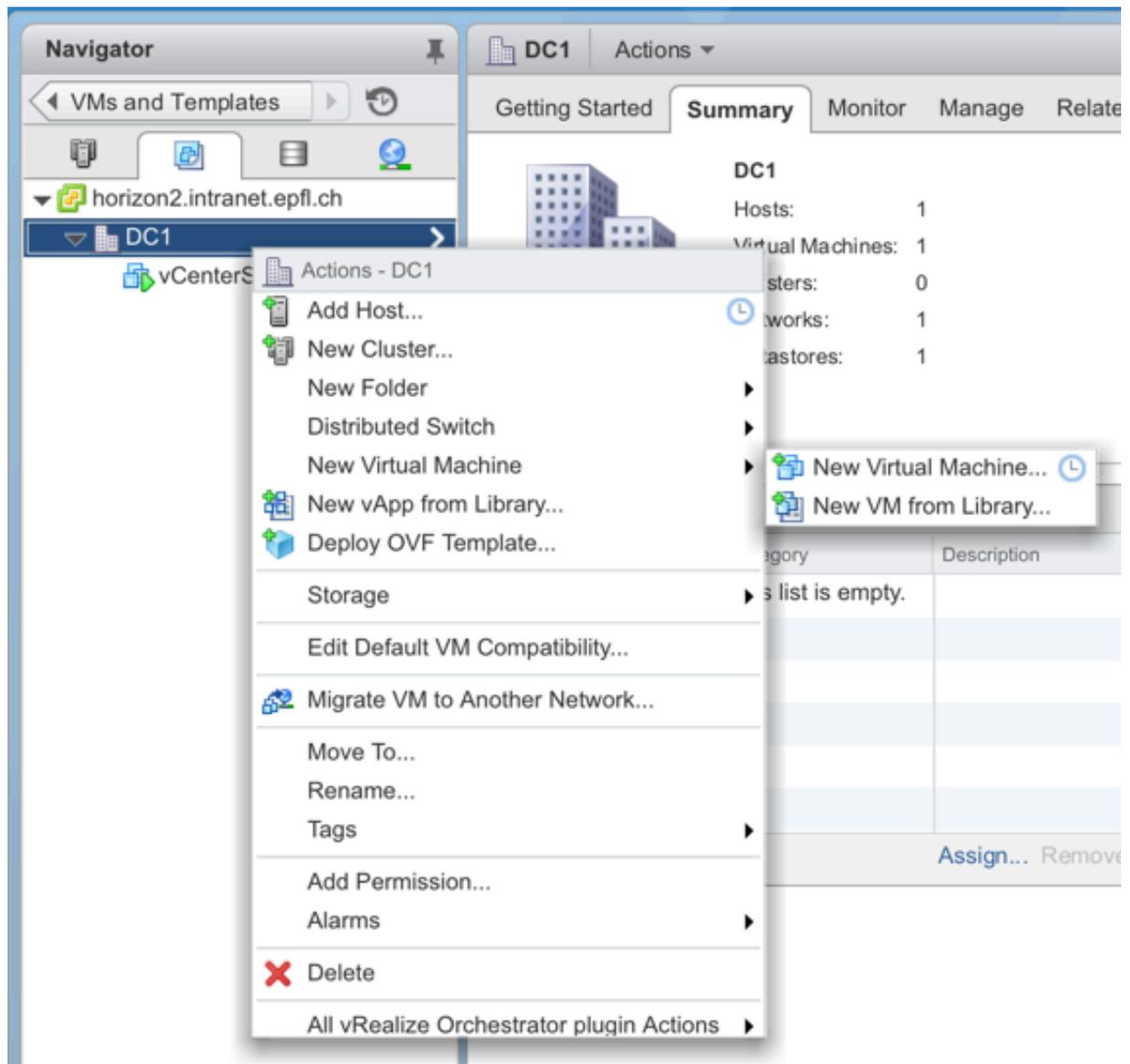
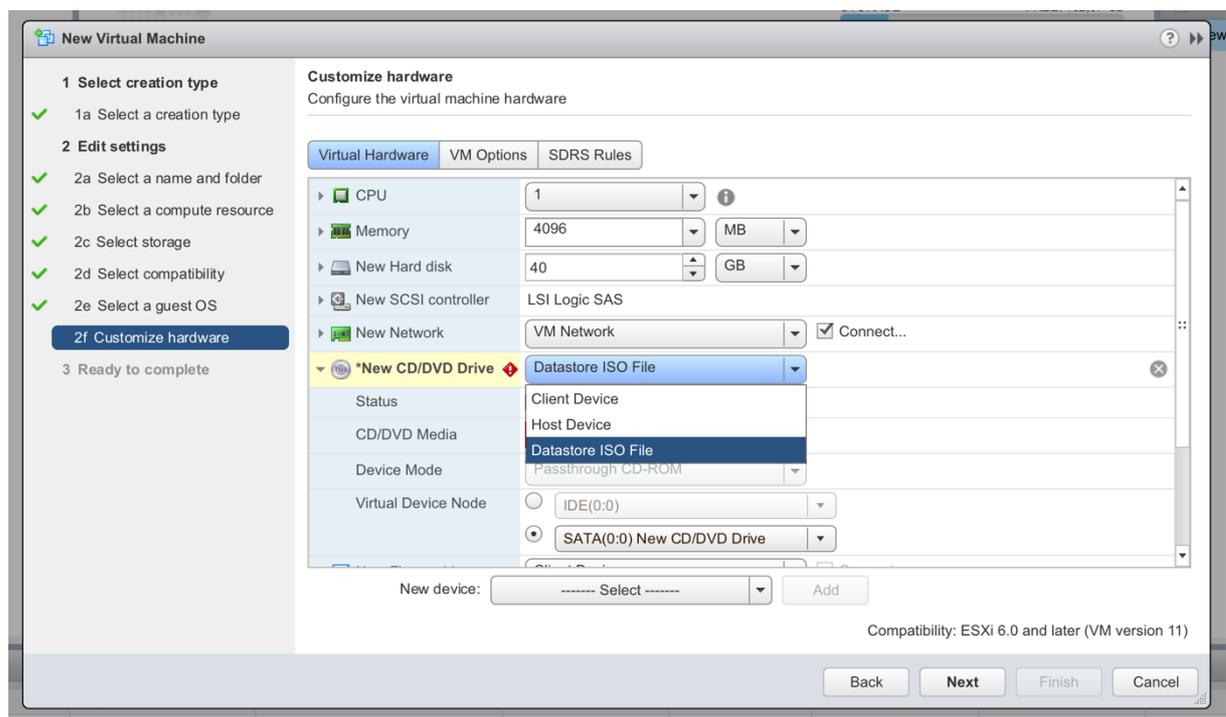


Figure 24. vCenter - New virtual machine creation

2. In the new window input a name, select the datacenter and datastore for the Horizon's virtual machine.
3. On the "Select compatibility" tab be sure that "ESXi 6.0 and later".

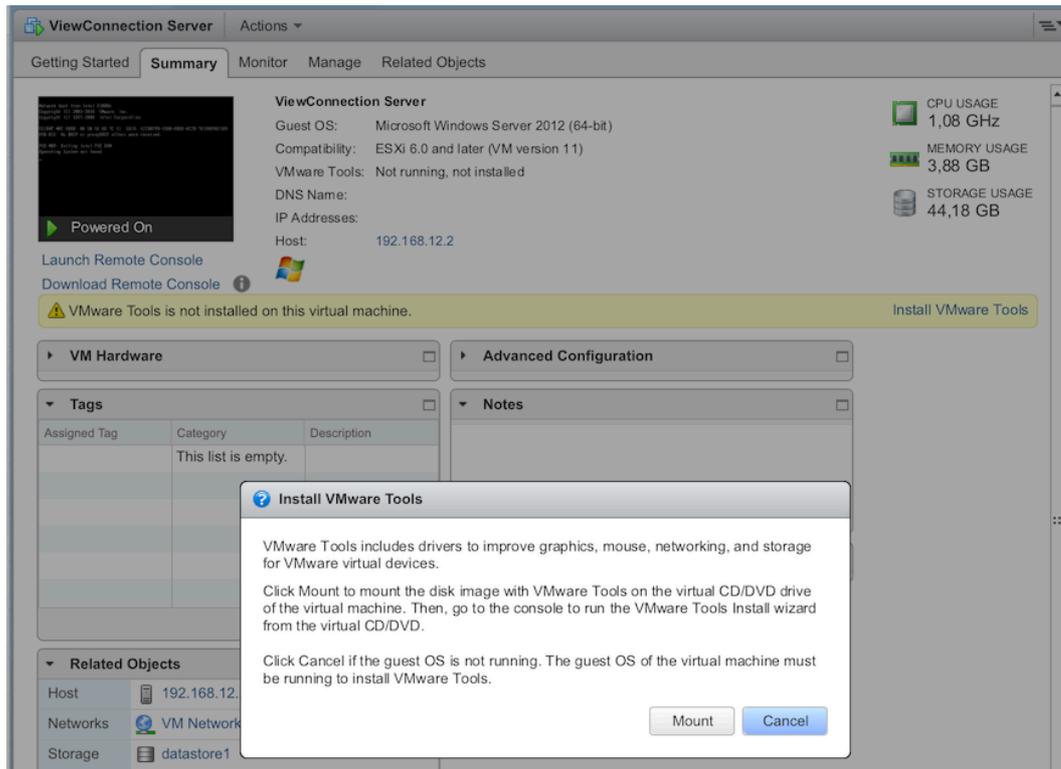
4. For the “Select a guest OS” tab select the “Guest OS Family” as “Windows” and “Guest OS Version” to the version of Windows Server you are installing.
5. On the “Customize hardware” connect an ISO file for the installation and be sure to provide enough memory for the server because how View Connection Server is set up depends on this. According to VMWare, for deployments of “50 or more” virtual desktops, it is required to set at least 10 gigabytes of memory for the View Connection Server. There is no defined demand on CPU power, so, this can be changed later. Also, VMWare highly recommends to change the type of network adaptor for all Windows virtual machines, starting from Windows 7, to be “VMXNET 3”.



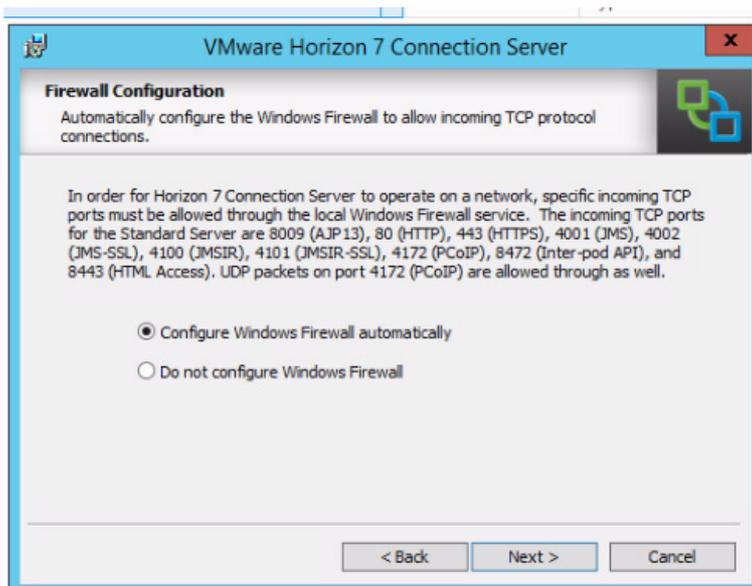
6. After clicking “Next” and “Finish” the virtual machine will be created. Start it by selecting it from the list of virtual machines and clicking “Power-Power On”.
7. Connect to the virtual machine using “Launch Remote Console” feature on the “Summary” tab of the virtual machine’s page.

6.2. Installing View Connection Server.

1. When the virtual machine boots, install the Windows Server Standart or Datacenter edition with GUI and setup administrator’s password.
2. In vCenter Server, on the Summary tab, click on “Install VMWare Tools”.



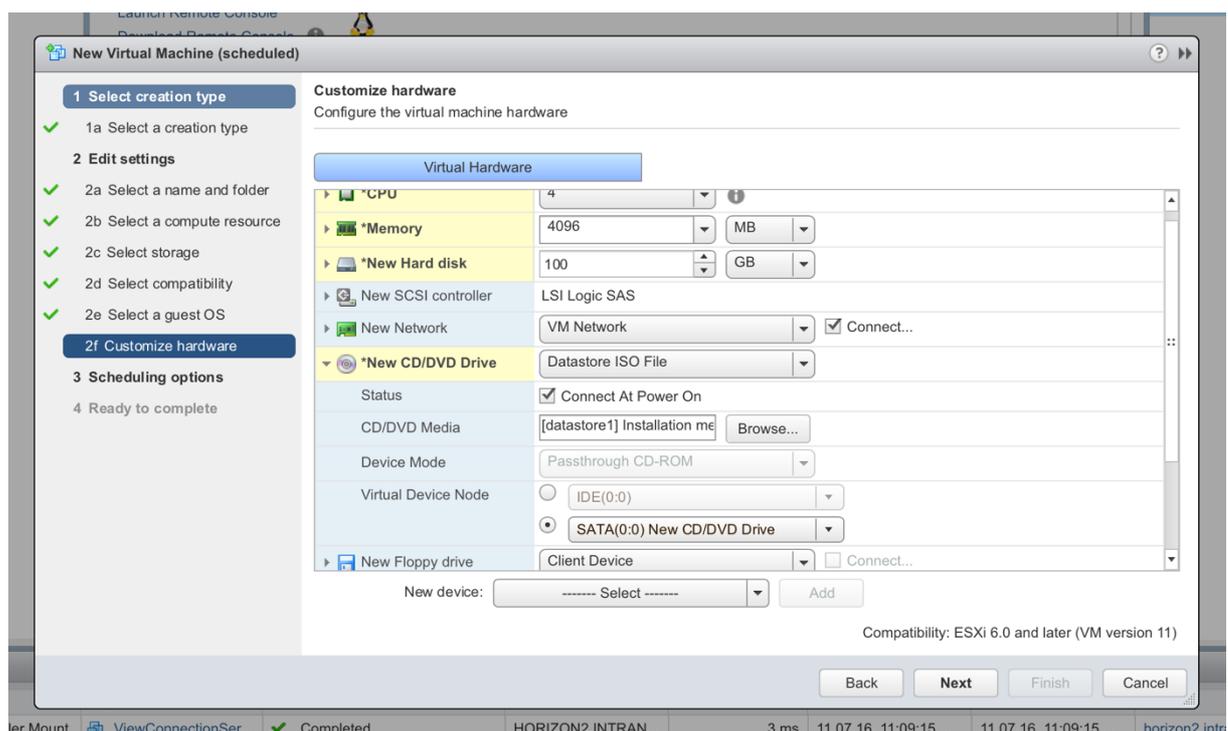
3. In the virtual machine install the loaded VMWare Tools and restart the machine.
4. Configure networking on the virtual machine so that it has a static IP address.
5. Connect virtual machine to the Active Directory.
6. Transfer View Connection Server installation files to the virtual machine and start the setup wizard.
7. In the setup wizard select “Horizon 7 Standard Server”, “IPv4”, “HTML Access” and “Configure Windows Firewall”.



8. In the “Initial Horizon 7 Administrators” setup wizard be sure to select “Aurhorize a specific domain user or domain group” option and provide its name using a format shown in the example.
9. When prompted for “Domain Login” provide Active Directory’s administrator user credentials.
10. After installation completes the window with a View Connection Server virtual machine can be closed.

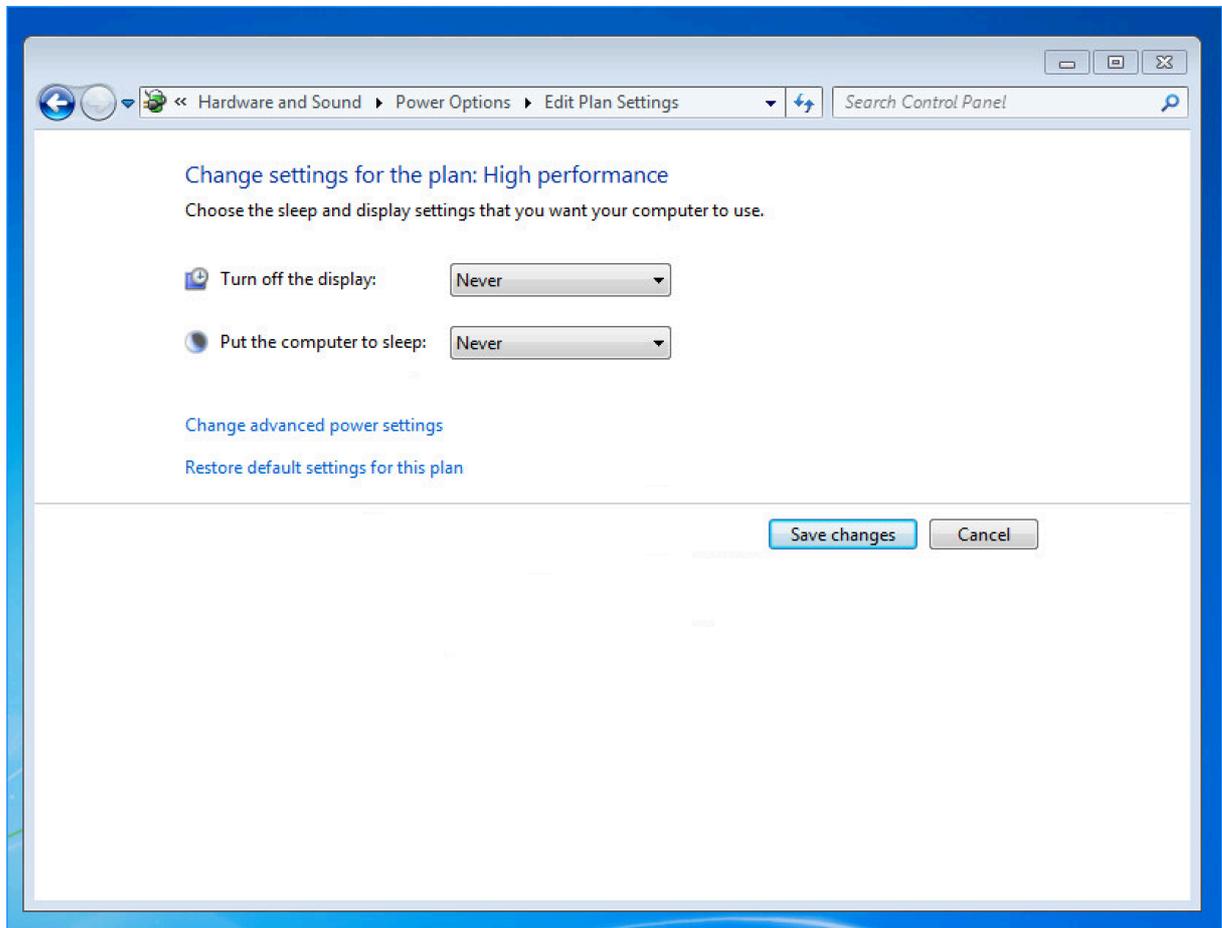
6.3. Master virtual machine creation

1. Login to vCenter Server and create a new virtual machine similar to the virtual machine with Windows Server. The only difference in configuration on the “Customize hardware step”: all settings from this page will be duplicated in all virtual desktops, therefore settings should be carefully adjusted. Also, another network should be selected: instead of the default (management) network, select the one that has a DHCP server running inside it. All virtual desktops require their own IP addresses and it is not possible to have this machine on a static network.



2. After installing Windows, be sure to install VMWare Tools.
3. Join computer to the Active Directory, if not done now, it is not possible to install updates on the machine.
4. Login and logoff in the machine with Active Directory user account so that all policies are applied.
5. Install all available updates from the Windows Update.
6. In Control Panel change go to “Power Options”, click on “High performance” plan and click on “Change plan settings”.

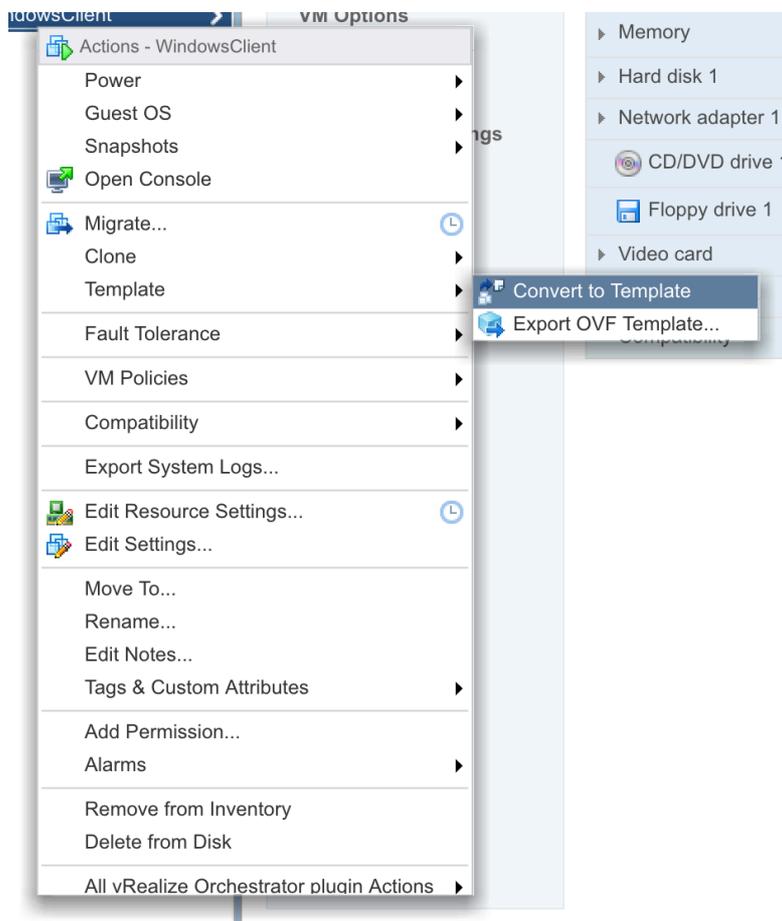
7. Be sure that this power plan has both display and computer sleep set to “Never”. If this setting is missing, after user disconnects from virtual desktop without shutting it down, he won’t be able to use the machine again.



8. If the users should be allowed to install new software on the machine, in Control Panel go to “Computer Management”. In the left menu select “System Tools->Local Users and Groups->Groups”. Go to “Administrators” group properties and add a new member group from the Active Directory.
9. There are different optimizations that could be done to the “master” virtual machine in order for it to work faster. The best solution found is VMWare’ tool “VMWare OS optimization tool” (<https://labs.vmware.com/flings/vmware-os-optimization-tool>), which can analyze the virtual machine and apply different performance tweaks depending on the operating system and environment installed on it.
10. Transfer Horizon Agent installation files to the virtual machine and start the setup wizard.
11. Select “Custom Setup” option and features required for the virtual machine. Settings that need consideration are USB Redirection and VMWare Horizon Instant Clone. By default, VMWare Horizon View Composer Agent is installed, which can’t be combined with VMWare Horizon Instant Clone.

Since we are not setting up a View Composer server, Instant Clone can be a useful feature in the “master” image.

12. When done with features, click on “Next” and select the option “Enable the Remote Desktop capability”. This way users will be able to connect to their virtual machines without View Connection Server by using only Remote Desktop protocol.
13. When installation finishes, shutdown the virtual machine and go to the vCenter Server.
14. Be sure to disconnect installation image from the virtual machine’s settings.
15. In the list of virtual machines click on the “master” virtual machine and select “Template->Convert to Template”.



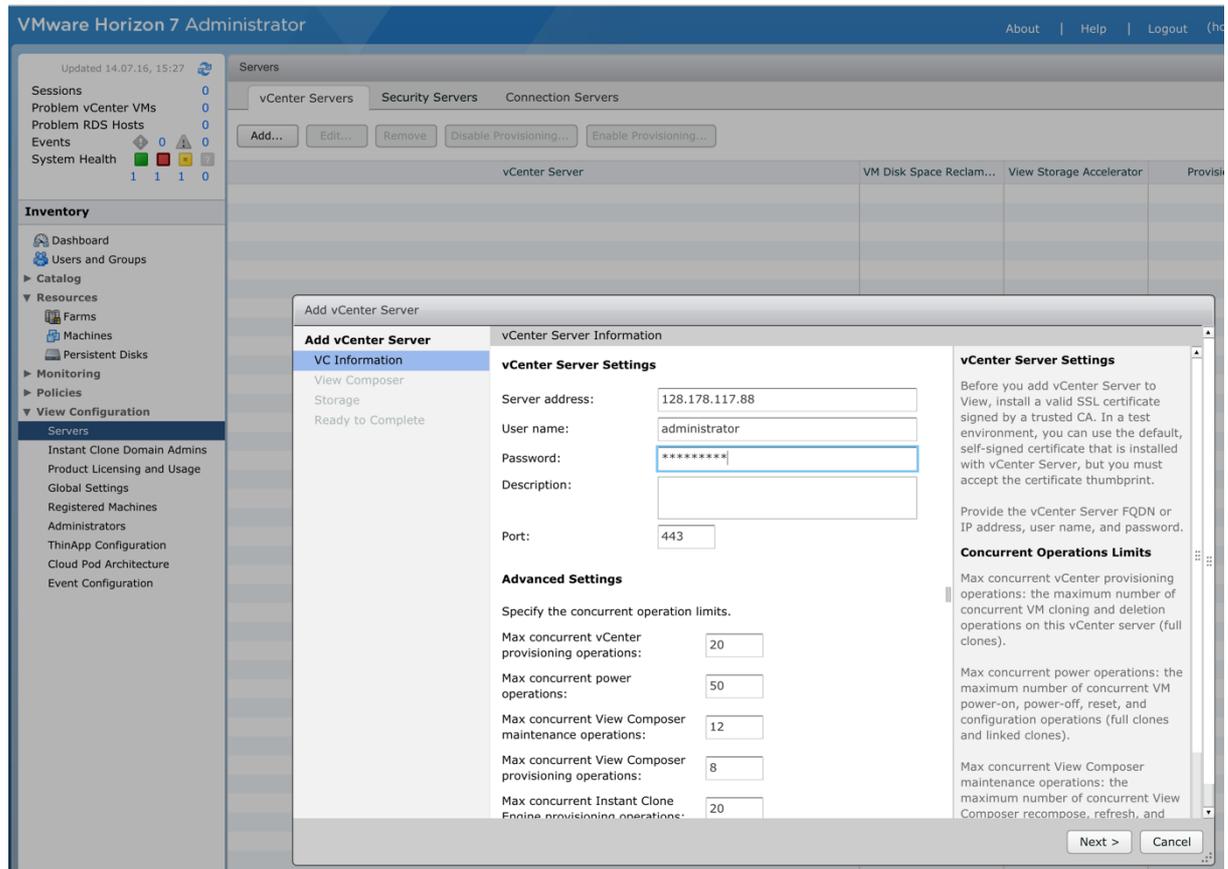
6.4. Configuring vCenter Server inside View Connection Server

1. Login to the View Connection Server with Active Directory by using its IP address or FQDN with added “/admin” at the end.

<https://vcenterserver.intranet.epfl.ch/admin/#>

2. When you see the “No valid license present” warning, click on “Edit license” and provide Horizon 7 license acquired from VMWare.

3. In the left menu click on “View Configuration” and select “Servers” option.
4. On the “vCenter Servers” tab click on “Add” and provide credentials for the vCenter Server.



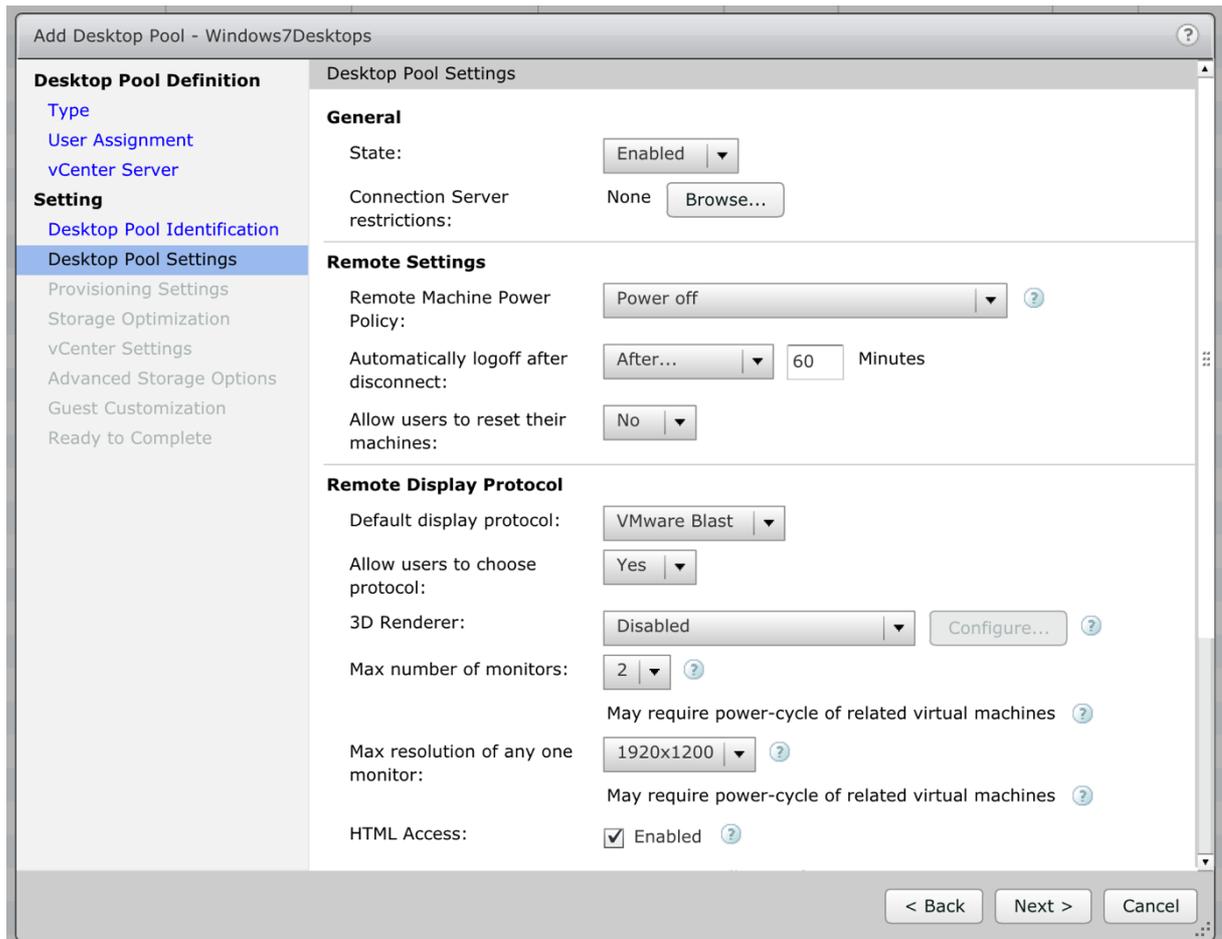
5. On the “View Composer” step select “Do not use View Composer” and click “Next”
6. On the “Storage” step check the “Reclaim VM disk space” and “Enable View Storage Accelerator”. For the “Default host cache size” provide the maximum value of 2048 MB. View Storage Accelerator is useful because it saves on amount of requests to the ESXi for blocks of data so that the most common blocks are saved within the View Connection Server itself.
7. When you click “Finish” on the “Ready to Complete” step, the vCenter Server should appear in the list shown like it is shown below.



6.5. Creating a Desktop pool

1. In the left menu of View Connection Server select “Catalog->Desktop Pools”
2. At the top of the window click on “Add” and select an “Automated Desktop Pool” type.

3. In the “User assignment” be sure to select “Dedicated” and check the option “Enable automatic assignment”. This way all virtual desktops will be “wired” to the users that first logged in on them.
4. In the “Desktop Pool Settings” be sure to select “VMWare Blast” as “Default display protocol” because it is much faster than other options. Also, be sure to configure “Remote settings” settings group as it is not recommended to have unused virtual machines always powered on.

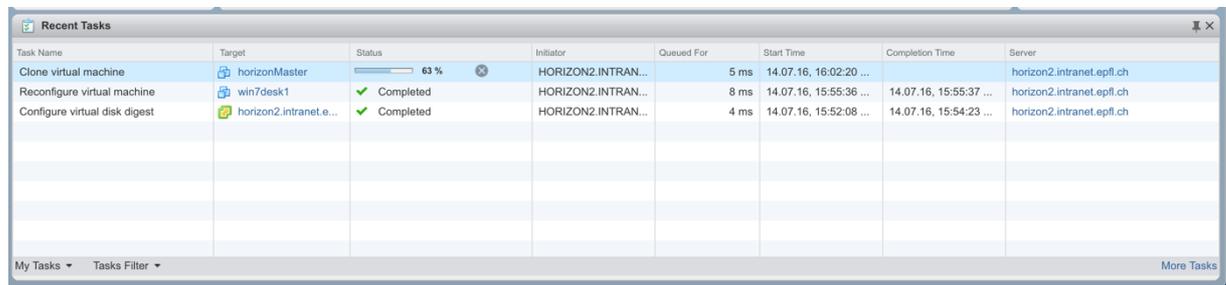


5. In the “Provisioning settings” important values are “Max number of machines” and “Number of spare machines”. First setting defines how many virtual desktops will exist in the pool and the second one defines how many virtual machines will be created in advance and powered on before new user makes a first login to them. In order to have higher performance of a system, this setting should be something small so that additional resources are not spent on virtual machines that do completely nothing.
6. On the “vCenter Settings” step be sure to select the right template. If the template is not on the list, it is probably due to the fact that the virtual machines’ operating system is not supported by VMWare Horizon.

7. Other options in the wizard do not require any changes. At the end of pool creation be sure to select “Entitle users after this wizard finishes”.
8. In the “Entitlements” window click on “Add” and provide search conditions for Users or Groups within Active Directory that are supposed to use the virtual desktops.

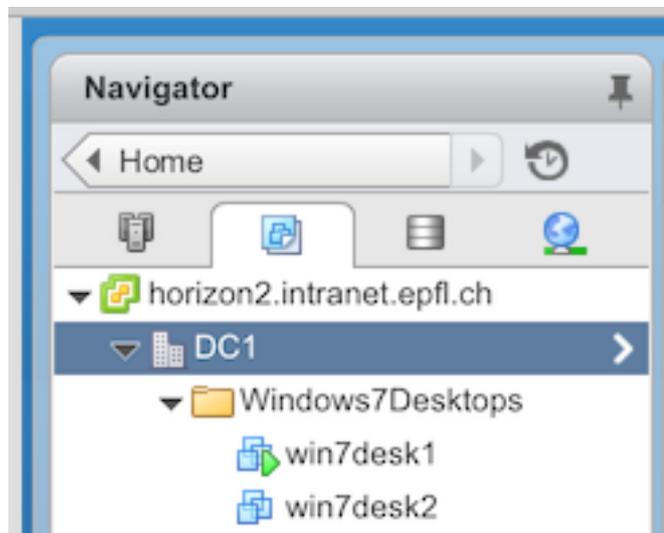
7. MONITORING VIEW CONNECTION SERVER OPERATIONS

1. When new user logs in to a virtual desktop, a “Clone” operation inside vCenter occurs for provisioning a new virtual desktop in advance for the next user:



Task Name	Target	Status	Initiator	Queued For	Start Time	Completion Time	Server
Clone virtual machine	horizonMaster	63 %	HORIZON2.INTRAN...	5 ms	14.07.16, 16:02:20 ...		horizon2.intranet.epfl.ch
Reconfigure virtual machine	win7desk1	✓ Completed	HORIZON2.INTRAN...	8 ms	14.07.16, 15:55:36 ...	14.07.16, 15:55:37 ...	horizon2.intranet.epfl.ch
Configure virtual disk digest	horizon2.intranet.e...	✓ Completed	HORIZON2.INTRAN...	4 ms	14.07.16, 15:52:08 ...	14.07.16, 15:54:23 ...	horizon2.intranet.epfl.ch

2. All virtual desktops are put in a folder created by the “Pool ID”, which is configured during “Desktop Pool Identification” step in the “Add Desktop Pool” wizard.



Appendix II: Horizon desktop virtual machines testing instructions

Horizon desktop virtual machines testing instructions

July 2016

1. Client software installation

There are two options for using virtual desktop machines from a personal computer.

The first option is to install VMWare Horizon Client application. This way almost all features of a real computer can be emulated: removable devices connected to computer can be accessed from virtual machine, files between virtual machine and your personal computer can be synchronized and there is almost none performance lag.

Another option is Horizon HTML Access. This way there is no need on installing a separate application because the only requirement is a modern web browser (using Internet Explorer prior version 10 is discouraged). But, since HTML Access is using a browser, there are some limitations: you can rarely but still see performance lags, some keyboard modifiers (like Ctrl+T) do not work correctly and using removable devices isn't possible here. Otherwise this solution is easier to setup and configure because it doesn't require any additional software installation and can be used by almost any computer inside internal network.

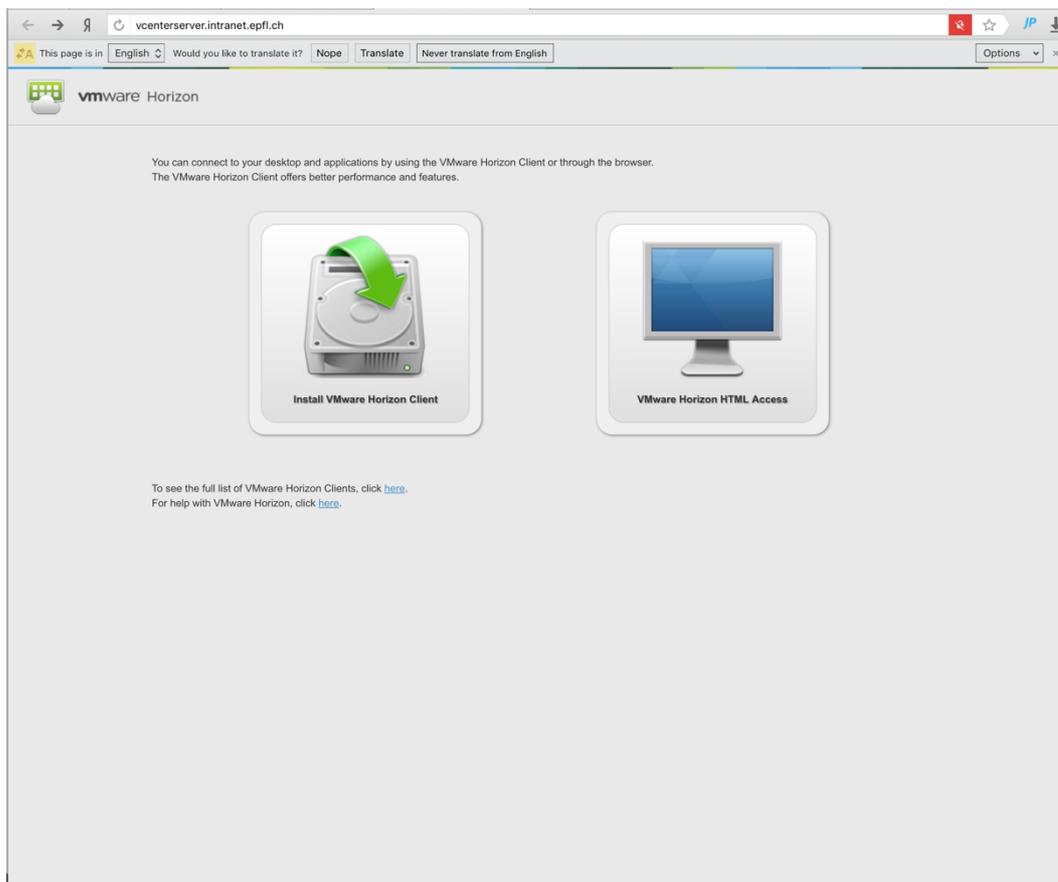


Figure 25. Horizon server client main page.

1.1. Horizon Client installation and configuration

- 1) Navigate to: <https://vcenterserver.intranet.epfl.ch/vsphere-client/> (or, if it doesn't work, to <https://128.178.117.88/vsphere-client/>) and click on "Install VMWare Horizon Client".
- 2) According to on-screen instructions, install the application.
- 3) Launch "VMWare Horizon Client" and in the given form input "vcenterserver.intranet.epfl.ch" and leave the option "Always connect at launch" checked.

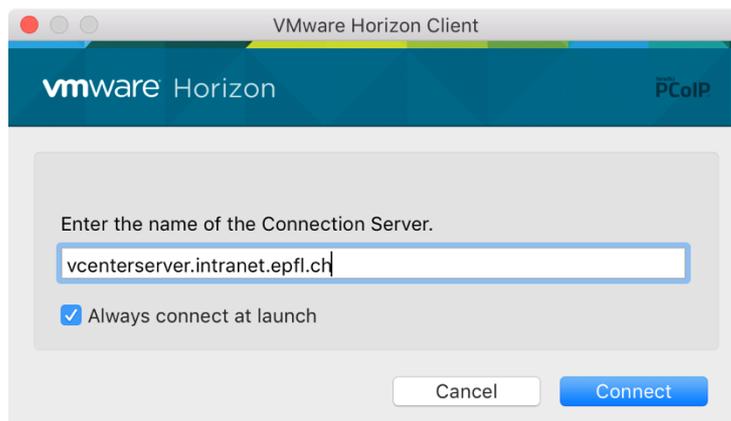


Figure 26. Horizon Client - New Server window

- 4) When you get a "Untrusted server connection" warning, click "Continue":

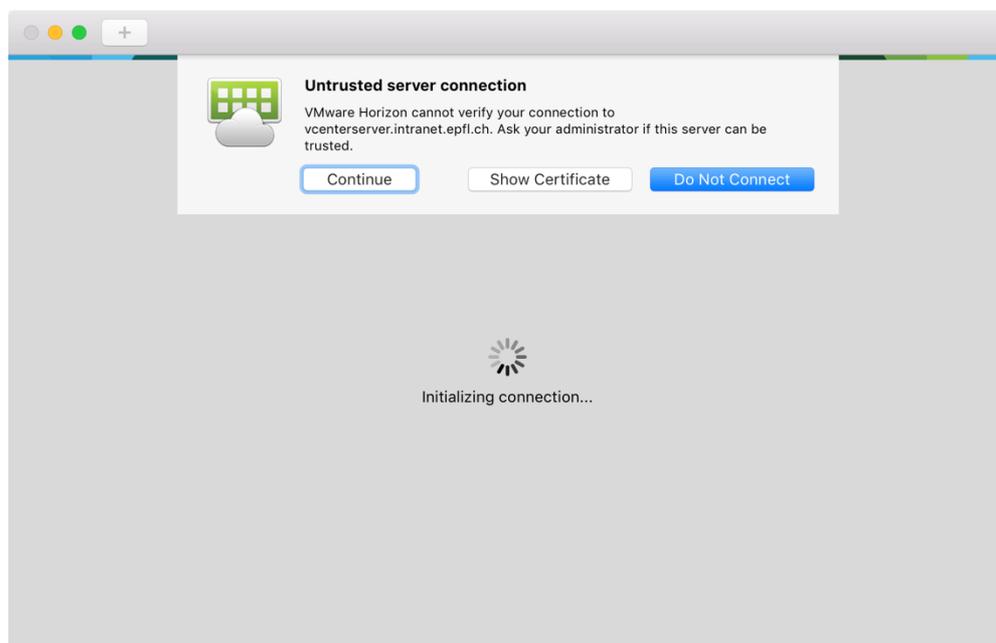


Figure 27. Horizon Client - Untrusted server connection

- 5) In the “Login” window provide credentials, which you were given for testing purposes and leave the default Domain selected.
- 6) After clicking “Login” you will be given a window of virtual machines that you can start. Select the one you are given and machine will automatically log you into the testing account:

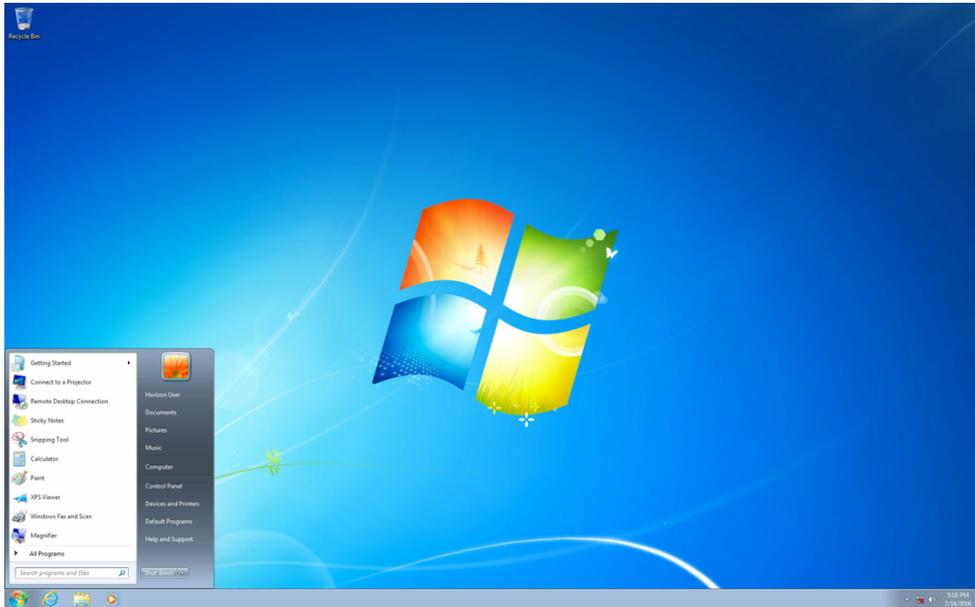


Figure 28. Horizon Client - Desktop logged in

- 7) For higher performance it is recommended to use VMWare Blast connection protocol. To enable it, click on the Settings in the Window from step 5 and in “Connect Via” select VMWare Blast.

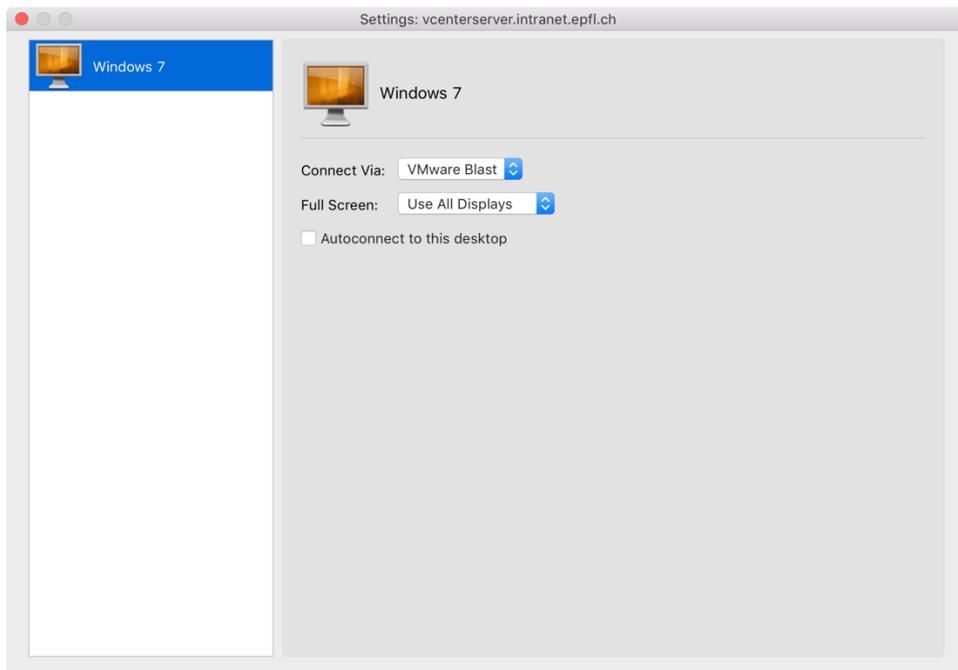


Figure 29. Horizon Client - VMWare Blast configuration

- 8) To share removable devices or folders from your local computer to the remote desktop, go to Horizon Client's "Preferences" and on the "Sharing" tab select required options.

1.2. Using Horizon HTML Access

- 1) Navigate to: <https://vcenterserver.intranet.epfl.ch/vsphere-client/> (or, if it doesn't work, to <https://128.178.117.88/vsphere-client/>) and click on "VMWare Horizon HTML Access".
- 2) In the login window provide credentials, which you were given for testing purposes, leave the default domain selected and click "Login":

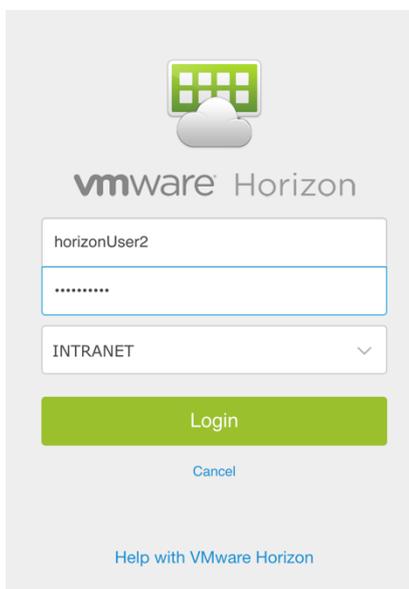


Figure 30. Horizon HTML Access - Login window

- 3) After clicking on virtual desktop it will start loading:

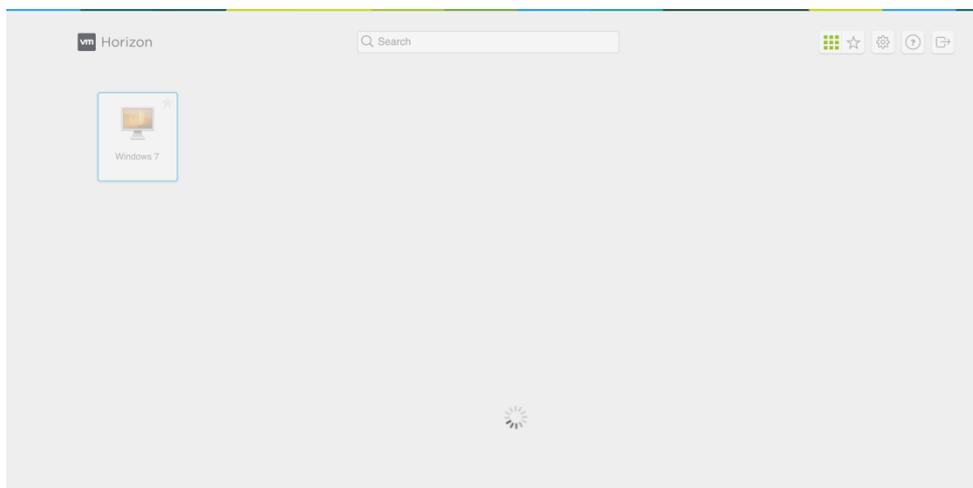


Figure 31. Horizon HTML Access - Desktop loading

- 4) There are not many settings available for Horizon HTML Access and usually their default values are fine. You can find all of them can be found at the right corner when after you finish step 2 and see a list of virtual desktops available.

2. Notes on how virtual desktops work

- Right now shutting down is configured the same as "Logout" action: the machine is turned off immediately. If there is a need for "placing machine on hold", it can be done by closing the Client application window or browser tab.
- 60 minutes after any kind of Horizon Client disconnects, the desktop is automatically shut down. It is recommended to shut down machine using Shutdown because otherwise problems in saving data can occur.
- Since it is a testing environment, some things can take time. For example, after logout it can take about a minute before the user can successfully login back. This is due to low testing server resources.

Aleksandr Mikheev

AUTHOR'S DECLARATION

I hereby declare that I have carried out this final research project on my own, without any help, other than the references listed in the list of references and that I have only used the sources mentioned. I will not provide a copy of this paper to a third party without the permission of the department head and of my advisor, including the partner company with which I collaborated on this project.

Aleksandr Mikheev

Date
