FACULTY OF SCIENCE AND TECHNOLOGY

MASTER'S THESIS

Study program/ Specialization: Computer Science	Spring semester, 2014 Open access
Writer: Anders Hole	…… (Writer's signature)
Faculty supervisor: Reggie Davidrajuh External supervisor(s): Hans Olav Aske	
Thesis title: Data Center High Availability for Integrated Machine Control Platforms	
Credits (ECTS): 30	
Key words: High Availability, Fault tolerance, Virtualization, Remote experience latency	Pages: 65 + enclosure: 1 Stavanger, 12.06.2014

Data Center High Availability for Integrated Machine Control Platforms

Anders Hole

2014

Department of Electrical Engineering and Computer Science
University of Stavanger

ABSTRACT

National Oilwell Varco is a multinational company providing solutions to the oil industry. The main focus in Stavanger office is the Cyberbase chair and control system. This chair is a operators interface with machinery and software that controls the drilling process. The high operation costs of an oil exploration rig, downtime is expensive and should be avoided. The current computer system used in machine control is setup uses multiple paths, which are dependent on all components within a path to function. This thesis looks at the current setup and tries to eliminate downtime, by building multiple redundant paths. By using virtualization and high availability functionality, less downtime and higher reliability for the data center is achieved. Remote desktop protocols for connecting client machines to a virtual machine are tested against each other, and an industry latency requirement. Tests reveal a large difference between protocols, and find one more suited.

PREFACE

I would like to thank my supervisors Reggie Davidrajuh and Hans Olav Aske for their feedback and guidance. My family and my dear friend Alfred have provided me with invaluable support during my time as a student. I would also like to thank the people of Drilling Data Center and Instrumentation & Monitoring departments at National Oilwell Varco Forus, for help and support throughout the thesis.

Table of Contents

1 MOTIVATION 1

2 INTRODUCTION 3

2.1 STRUCTURE OF THESIS 4

3 BACKGROUND AND OVERVEIW 5

3.1 ABBREVIATIONS 5
3.2 OFTEN USED TERMS 6
3.3 NATIONAL OILWELL VARCO INC. 6
3.4 DEPENDABILITY 7
3.4.1 Reliability 9
3.4.2 Availability 9
3.4.3 Hardware setup 10
3.5 MONITORING AND ALERTING 14
3.6 SECURITY 15

4 DESIRED SPECIFICATION 16

5 CHALLENGES IN SYSTEM DEVELOPMENT 17

5.1 SELF-DEVELOPED VS TURNKEY SOLUTION 17
5.2 STORAGE 17
5.2.1 Networked data stores 18
5.2.2 Virtual Disks 18
5.2.3 Deduplication 19
5.3 COMPUTER HARDWARE 20
5.3.1 Server 20
5.3.2 Client 20
5.4 MONITORING 20
5.5 NETWORK 21
5.5.1 Jumbo frames 21
5.5.2 Spanning Tree Protocol 22
5.6 VIRTUALIZATION TECHNOLOGY AND PRODUCTS 22
5.7 SYSTEM SOLUTION EVALUATION 28
5.7.1 Storage 28
5.7.2 Virtualization 30
5.7.3 Remote connection client 30
5.7.4 Security 31
5.7.5 Comparison of available products 32

6 SYSTEM TEST RESULTS AND ANALYSIS 33

6.1 TEST SETUP 33
6.2 HYPERVISOR-HARDWARE LAYER 34
6.2.1 Hardware failures 36
6.2.2 Failure summary 44
6.3 VIRTUAL MACHINE 45

6.3.1 Prime95 45
6.3.2 Futuremark PCMark 7 47
6.3.3 Anvil Storage Utility 48
6.4 OPERATORS VIEW 51
6.4.1 HMI Application 51
6.4.2 CCTV system and latency 55
6.4.3 WPF Benchmark 56
6.4.4 Camera video visual loop test 56
6.4.5 Broadcast test 58
7 CONCLUSION 63
7.1 FURTHER WORK 63
8 APPENDIX 66
8.1 VERSIONS 66

1 MOTIVATION

Downtime due to hardware related failures is costly in server environments. Predictive measures can protect against failure of components, but often depend on a single point of failure. This does not ensure the stability of the system as a whole. The goal of this thesis is to look into and explain methods, technology and solutions that can provide better protection against system downtime by using virtualization.

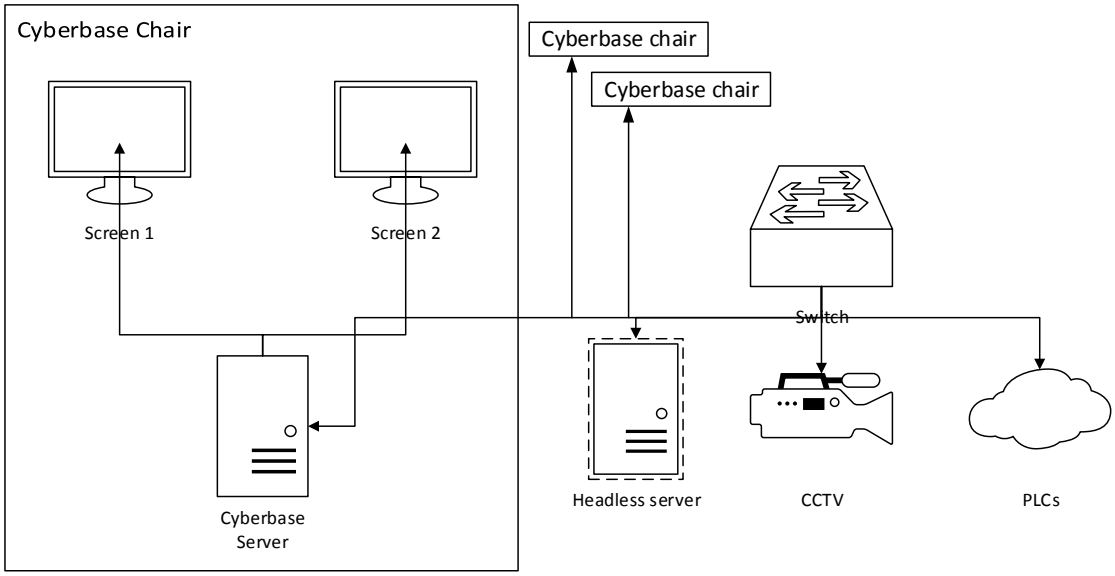


Figure 1-1 Configuration of the Cyberbase chair system currently in use

Figure 1-1 shows a system that consists of a dedicated server running control software in a Microsoft Windows environment. If one of the servers encounter a problem, the graphical user interface presented to an operator may become unusable until the problem is fixed. This setup makes the system vulnerable to errors. The objective is to find a more robust solution that decreases the potential for errors and provides high availability through making the components less tightly coupled. This could involve virtualizing servers and running them in a cluster, looking into sharing of CPU, RAM, storage and network to come up with a solution that can provide redundancy and bumpless failover functionality. It needs to be a fault tolerant solution to interface with control software, while maintaining the same performance required for operating the system. Investigation of possible improvements or alternatives to surrounding systems such as network might also be taken into consideration as a prerequisite for the solution.

2 INTRODUCTION

Downtime is costly in server operation. If a critical component failure renders a system inoperable, the monetary implications could be within thousands of dollars very quickly. Financial loss is not the only or the most important factor if a system experience downtime. Irreplaceable data can be lost and lead to issues between the parties managing and operating the equipment, and the consumer that uses it. When providing server solutions to a costumer this is a very critical aspect. If the costumer experiences severe problems, they will most likely use other providers in the future. The software of a solution can be exceptionally good without having an impact on the costumer's total impression if hardware problems arise and cause instability.

National Oilwell Varco's server and network solutions are not the conventional single centralized datacenter providing services to thousands of users. Usually only a handful of people use the system on a regular basis, but many more rely on the services the system provides. The equipment, which these servers control, is the key function of every drilling rig. Downtime, although affecting few users, will rapidly lead to very large costs. The going day rate of an offshore drilling ship or semi-submersible run up to $600,000 US dollars each day [1]. Operating companies demand safety and efficiency and rely on historical data to ensure this. In the event that such data is lost the analytical capabilities disappear. Connection speed from offshore rigs may vary from a few kbps to multiple Mbps, this makes backing up data to an external location difficult in some cases. Communication from non-fixed rigs are satellite based and may be unstable. On-rig data storage must thus always exist. Generally, duplicating data over multiple disks gives the required storage redundancy. Replacing a disk will fix a disk failure, but what if another component such as the drive controller fails. It is important to secure the data against any failure by having redundancy in every system element. Repairing failed equipment during operation offshore takes considerable time during preparation, shipping and installation, and may hinder ongoing operations. To avoid downtime, equipment should be made redundant and fault tolerant where possible. Personnel involved in daily rig operations often have little knowledge about the system composition, and cannot undertake complicated tasks. They are not IT professionals and cannot be expected to reconfigure a new server in the event of a failure. These personnel should not do management of a system, since configuration errors could lead to severe problems. Reconfiguring the system should be

possible during operation, if not it should have enough tolerance to wait until scheduled downtime periods.

2.1 Structure of thesis

Chapter 3 contains background information and explain concepts used later in the thesis. In chapter 4 the desired operation is described without consideration to the available products or solutions. Chapter 5 looks into approaches of fulfilling the desired specification, by using and self-developed or a turnkey solution. The features from different virtualization software providers are evaluated up against the desired specification. In chapter 6 the solution from the previous chapter is setup and tested. Both general and NOV specific tests are performed and the test results are discussed and compared to the desired specification. Chapter 7 concludes the thesis.

3 BACKGROUND AND OVERVEIW

3.1 Abbreviations

Abbreviation	*Description*
CCTV	*Closed Circuit Television*
CPU	*Central Processing Unit*
DCN	*Drilling Control Network*
DMZ	*Demilitarized zone (perimeter network)*
DPI	*Deep Packet Inspection*
FPS	*Frames per Second*
FT	*Fault Tolerance*
GFX	*Graphics Card/Adapter*
HA	*High Availability*
HCL	*Hardware Compatibility List*
IOPS	*in Input/Output Operations per Second*
JF	*Jumbo Frame*
KVM	*Keyboard, Video and Mouse*
LAN	*Local Area Network*
MTTF	*Mean Time To Failure*
MTTR	*Mean Time To Repair*
NAS	*Network Attached Storage*
NIC	*Network Interface Card*
OS	*Operating System*
PSU	*Power Supply Unit*
PXE	*Preboot Execution Environment*
RAID	*Redundant Array of Inexpensive Disks*
RAIN	*Redundant Array of Inexpensive Nodes*
RAM	*Random Access Memory*
RD	*Remote Desktop*
RDP	*Remote Desktop Protocol (Microsoft product)*
SAN	*Storage Area Network*
SNMP	*Simple Network Management Protocol*
SSD	*Solid State Drive*
VD	*Virtual Disk*
VDI	*Virtual Destop Infrastructure*
VM	*Virtual Machine*
VPN	*Virtual Private Network*
WAN	*Wide Area Network*

3.2 Often used terms

HA – High Availability

A system is High Available if the uptime is significantly longer than the downtime, even if stochastic components fail. Reducing single points of failure to a minimum will help to achieve HA.

FT – Fault Tolerance

Fault tolerance ensures that an OS or service keeps running when a hardware failure occurs. FT differs from HA because it does not allow any failure related downtime.

NIC teaming

NIC teaming combines multiple NICs together to provide network HA, load balancing or both.

Remote desktop

Remote desktop is used to describe the concept where the desktop of a remote machine is displayed locally. Both Microsoft's Remote Desktop Protocol and VMWare View fall under this category.

Server naming

NOV servers are named as Server <letter>, often shortened to Serv<letter>, the same naming will be used in this thesis. Server <number> is used when referring to a hypervisor Host.

3.3 National Oilwell Varco Inc.

NOV is a global company that provides drilling and production equipment for the oil industry. This includes draw-works, mud system, blow out preventer and other equipment that are essential in the drilling process. The main-focus at NOV Forus is the Cyberbase drillers chair, used to integrate the control of equipment on and surrounding the drill floor. Cyberbase chairs are the drillers' interface with the software that controls the drilling process and drilling equipment. The software also provides a connection to third party products through the Cyberbase System. Each offshore drilling rig is equipped with two to six chairs. This is a

crucial component to maintaining control, monitor and supervise the equipment used in the drilling process.

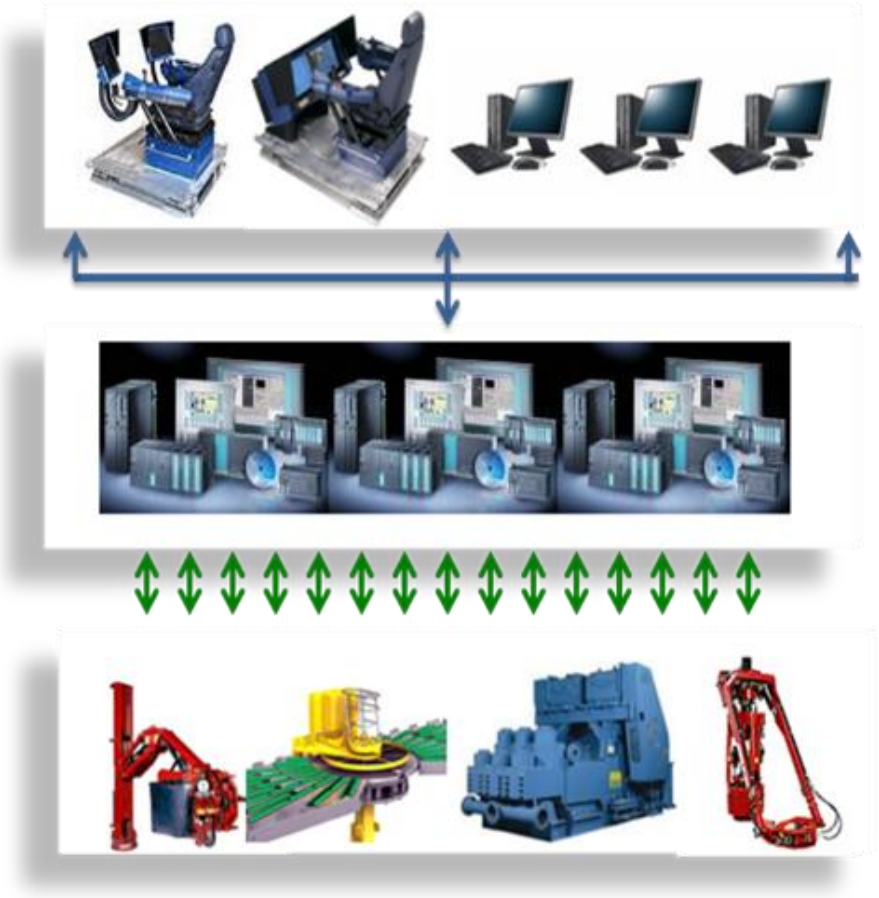


Figure 3-1 from top: Cyberbase chair and terminals, control logic, field machines.

3.4 Dependability

All electrical components will degrade and over time fail to operate in the intended way. The failure may be a consequence of one critical internal component failing before other components, or many components failing concurrently due to external factors. The *Capacitor Plague* [2] in 2002 made capacitors fail long before their expected lifetime due to problems with the electrolyte. External factors, especially temperature, have a great impact on a components lifetime. The failure of a fan can cause temperatures to rise beyond the component's limits, and greatly shorten life expectancy.

Predicting the reliability is done by testing a population of components and see how many fail. These tests can often produce a misleading answer.

In 2013 there were a population of 66302 25-year olds in Norway. During the period of one year the operational time of the population is 66302 years. During 2013 there were 48 deaths in the population, giving a failure/death rate of $\frac{48}{66302} = 0{,}0007240$. MTBF can be found by inverting the failure rate $\frac{1}{0{,}0007240} \approx 1381\ years$. A single person cannot be expected to live 1381 years. In reality, the expected MTBF is 81 years based on field data. This example applies a constant failure rate to the whole lifetime of a person, while in reality the failure is more bathtub shaped. When a component ages, more failures occur due to wear out. Assuming a constant failure rate throughout a components intended lifetime have in field gathered data been proven wrong. A study on HDD life expectancy [3] stated

"…Failure rate is not constant with age, and that, rather than a significant infant mortality effect, we see a significant early onset of wear-out degradation. That is, replacement rates in our data grew constantly with age, an effect often assumed not to set in until after a nominal lifetime of 5 years."

MTBF can be used to give an indication of the probability for failures in the components intended lifespan, not the length of the lifespan. Failure rates advertised by vendors are very seldom backed up by underlying data sets.

Comparisons between failure rates have to be on similar terms. Environmental factors such as temperature, humidity and vibration have to be equal in all tests, or normalized against each other. Comparing the expected failure rate of a component that have been tested in a harsh conditions, against one that have been tested in perfect conditions will not give a meaningful answer. The parameters and assumptions of the stress test are very important to be able to compare different components on similar terms. Server composition may have components that effect each other. A low quality PSU may give a component a lower life expectancy because the component was tested with a better PSU. System components can effect each other in multiple ways that are hard to test for separately. Calculating a total life expectancy for a system of components therefore only gives an approximation. If an actual value is required, the whole system must be tested and field data must be gathered. Due to the rapid development of computers, this process may take a longer time than is feasible.

3.4.1 Reliability

Reliability measures the probability that a component is working over a time period T. If the component works in time t what is the probability that it still works in $t + 1$. Downtime, planned or unplanned, does not affect the measure.

The reliability of hardware components often follow a bathtub curve. The curve has high failure rates in the beginning and at the end of a components lifecycle. Failures during shipment, installing and configuring are gathered under infant mortality. During the operational period failures occur at random and are evenly distributed over the time period. Wear out failure occurs when components approach their lifecycle time.

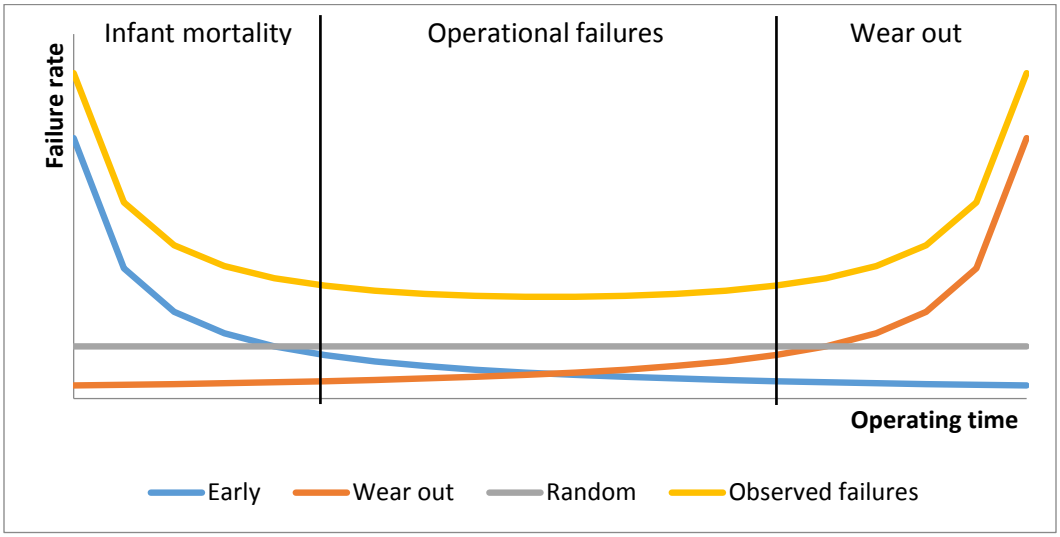


Figure 3-2 Reliability Bathtub Curve

In an ideal scenario, the infant mortality period should be finished before taking the system into operational use. This would allow the manufacturer to handle all infant problems. Performing a burn-in test before deeming a server ready for use, will move some of the infant mortality from operation to commissioning. Tests for CPU, RAM, GFX and disk should run for multiple hours at high intensity for a valid result. Accurately determining when the infant mortality step is over can be difficult and must come from experience with a specific setup over a long time period.

3.4.2 Availability

Availability is a measure between the time a component is delivering services as intended and other states that are unintended. Availability excludes scheduled downtime. The availability scope defines which components are included in an availability measure. Scopes with the same failed component will have different availability. A disk mirroring configuration with one

disk failed will from the disk controller's setup be a functional state, but not a desired operation mode. The overlying OS will not see the failure and will mark the disk as functional.

Availability can be expressed by Mean Time between Failure (MTBF) and Mean Time to Repair (MTTR)

$$avaliabilty = \frac{MTBF}{MTBF + MTTR}$$

For many hardware components, MTBF is often a large and incomprehensible number, and can be a misleading figure. MTBF is a population statistic and cannot be used to determine the exact behavior of one individual component.

Annular failure rate (AFR) is used to give a number of how many units fail per year calculated as an exponential distribution. This assumes that the component is running a full year (8760 hours).

$$AFR_{year} = e^{\frac{-8760\ h * MTBF * years}{MTBF}}$$

3.4.3 Hardware setup

3.4.3.1 Hardware reliability and failures

Calculations in this chapter assume that the reliability of similar components are independent. This is a simplification, and cannot be assumed for real life applications. Faults in the production process can make a certain batch of components fail before MTTF. Multiple components from the same batch may fail within a short time period, degrading the usefulness of a failover setup. It is recommended to use components from different batches to minimize the probability for production flaws.

3.4.3.1.1 System

A computer have many interconnected parts that all must work in order to provide the desired functionality. The typical desktop construction features no fault tolerance from the hardware side. If a single component fails, the whole system fail and cannot continue operation before the failed component is replaced.

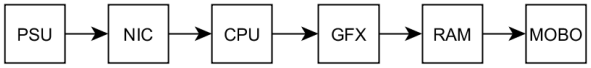


Figure 3-3 Typical server setup

$$R_{server} = R_{PSU} * R_{NIC} * R_{CPU} * R_{GFX} * R_{RAM} * R_{Mobo}$$

Redundant PSUs are used in multiple server configurations. They are simple to implement compared to other “smarter” components since no additional software/drivers are required. Since PSU is a self-contained component with fans (that have a low MTBF) they are often the weakest link in the series of components that make up a server. Making the PSU redundant therefore has a great impact on the reliability of the system.

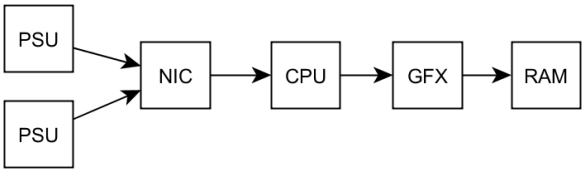


Figure 3-4 Server with redundant PSUs

$$R_{server} = (1 - (1 - R_{PSU})^2) * R_{NIC} * R_{CPU} * R_{GFX} * R_{RAM} * R_{Mobo}$$

	MTBF in hours	
Component	**Worst**	**Best**
Power Supply Unit	150 000	350 000
Network Interface Card	200 000	350 000
Central Processing Unit	500 000	4 000 000
Random Access Memory	8 300 000	50 000 000
Graphics Card	800 000	1 000 000
Motherboard	150 000	350 000

Table 3-1 MTBF for various hardware components

Typical MTBF values can be found in Table 3-1. PSU and motherboard have some of the lowest values. Some of these values are not confirmed by the manufacturer and should only be used as a general guide.

Some redundancy functionality is only available in server hardware. Sacrificing functionality that prevents less common failures are often acceptable in desktop environments. Particular RAM failures have a big impact and are hard to troubleshoot. Error Check and Correction

memory are widely used in server hardware, but is at a higher price point. To use ECC RAM both the CPU and motherboard must support ECC. This often means that a full server hardware setup must be used to enable ECC, yielding a higher system cost.

NIC link aggregation for Windows previously relied on implementation in the driver. This was only supported on some high cost server type NICs. Windows Server 2012 introduced native support for teaming [4]. Most hypervisors also support it natively.

3.4.3.1.2 Storage

Storage is one of the most important features of a computer system. In oil exploration millions of dollars are spent on gathering data to determine if an oil field should be developed or not. Traditional Hard Disk Drives can be perceived as less reliable due to mechanical moving parts, but manufacturers often claim a MTBF up to 1.5 million hours. Solid State Drives have the same MTBF but contain no moving parts, and are better suited for harsh environments with much vibration. Comparing the maximum shock values under operation for a SSD and HDD, the SSD is rated for 1500G [5] compared the HDDs 400G [6].

3.4.3.1.2.1 Unrecoverable Bit error rate

When reading and writing data every storage device have a possibility of data corruption. Write errors are referred to as “silent corruption” since the device controller thinks the data is intact, and does not detect errors before the data is read back. Unrecoverable Bit Error Rate (URE) gives the probability for reading corrupt data back. URE rates for modern storage devices range from 10^{-14} to 10^{-17}, or an error in 1 bit out of 12.5TB to 12.5PB. Worst case for a 2TB disk an error will occur for every third full data read. This rate is acceptable for single disks, but when combining multiple drives in RAID-5, it could become a problem. During normal RAID-5 operation, the storage controller (if aware of the error) could read the data from the other drives. When a disk failure occurs this is not possible since no replica exists. With six 2TB drives in RAID-5, the probability for a bit-error is higher because data is distributed over multiple disks, while the URE is the same for one and six disks. Multiple articles use this logic to claim RAID-5 dead. Assumptions made to arrive at this statement make the problem seemingly worse than it actually is. One read error during array rebuild causes the whole process to stop and eventually leads to data loss. In practice this is not very likely. RAID-6 is better than RAID-5 since the double duplication of data, can handle a disk failure and still have

two intact copies. But RAID-6 will eventually suffer from the same problems as RAID-5 if size increases and error rate decreases at the same rate as it has been. Using a RAID controller, or software RAID, with checksum support that periodically check data and correct errors will reduce the possibility for read error and silent data corruption.

Unrecoverable Bit Error Rate	Bytes	Data read to produce one error
10^{-14}	$1{,}25 * 10^{13}$	12,5 TB
10^{-15}	$1{,}25 * 10^{14}$	125 TB
10^{-16}	$1{,}25 * 10^{15}$	1250 TB
10^{-17}	$1{,}25 * 10^{16}$	12500 TB

Table 3-2 Unrecoverable bit rate

3.4.3.1.2.2 Drive setup

OSs have a highly randomized disk read and write operations, and therefore require storage with good performance and low seek-latency. RAID-5 and 6 provide lower performance than other levels since they have to calculate parity and write to multiple disks in an unsequential manner. Network latency is a deciding factor when connecting a data store through network.

Because of the potential rebuild errors of large RAID-5 and 6 setups, they are not recommended for storing mission critical VMs. RAID-10, combining RAID0 nested in RAID1, only reads data from the other pair when rebuilding, limiting the maximum data needed to be read to $2 * drive\ capacity$. For VM data stores RAID10 is recommended since it provides a low probability for data loss, combined with good performance.

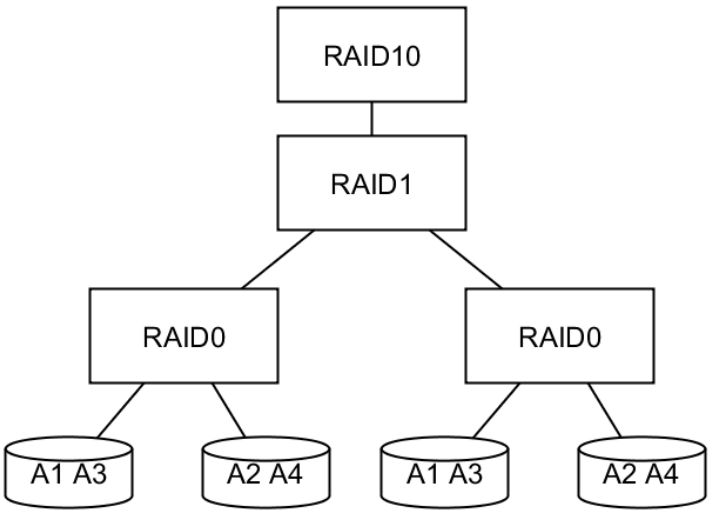


Figure 3-5 RAID 10

3.5 Monitoring and alerting

SNMP is a widely used protocol to poll and manage network equipment and appliances. Two services are used, an agent that runs on every monitored machine, and a management service that controls and gather information. SNMP enables two different ways to gather information, pull and push. The management server can run an active service that pull data from an address on monitored devices on given time intervals. The address is an Object Identifier (OID), an identifier that holds a defined value. If an agent detects a state change, it can generate and push a trap to the management server. The management picks up the trap, and conveys an alarm. SNMP is the preferred monitoring solution because it is well supported by most network devices and *NOV Cyberbase System Monitoring* software.

Most server grade hardware support a form of out-of-band management technology, which allows administrators KVM access to a machine over the network. The functionality is provided through hardware, and allows access even though no OS is installed. This enables administrators to connect and manage machines and allows troubleshooting even if the OS is in a non-functional state. Implementations of this technology include Intel vPro Technology [7] and Intelligent Platform Management Interface (IPMI) [8].

3.6 Security

Security is a very high priority in a control system. Viruses designed to infect control systems and PLCs (such as Stuxnet), have displayed vulnerability. Consequences of an infected control system could be disastrous, with high potential for human loss. DCN is a closed network without connection to the internet with the exclusion of a service terminal secured behind a firewall.

Service personnel use this terminal to perform tasks connected either physically or through a VPN. Hypervisors can include a firewall that sits between the network and the management of the hypervisor. However, this firewall does not normally filter any traffic between a virtual machine and the network. Hypervisors are vulnerable to infection since they have little protecting them. Detection of malware in virtual environments mainly consist of three different approaches, VM- or network-based and hybrid. Monitoring each VM with security software, similar to non-virtualized computers, provides detection of malicious software in that VM. This approach will consume resources since it has to run on every VM. VMs performing similar operations may run the same detection procedure once for every VM, and the anti-malware software running in the VM will most likely not detect malware on the hypervisor. Hypervisor-malware is referred to as a “Blue pill”, and “Red Pill” is anti-malware software that detects malicious software.

If one assumes that threats arrive through the network, a centralized software can monitor and filter each network stream and prevent it from arriving at the destination if it finds irregularities. DPI checks the content of a network packet for signs of malicious activity. Network delay will increase, but all traffic is monitored, providing hypervisor protection. During operation, control environments have little high-risk data traffic. The majority of network traffic is low risk delay-sensitive packets for machine control. A security measure scanning all packets and introducing delay might be too expensive compared to the benefits gained.

The third category is running anti-malware software in the hypervisor, protecting the host and VMs running on it. The software could be an own VM with special privileges, that allow it to monitor the host.

4 Desired specification

The system as a whole should allow failure of any single component in the system without having an impact on overall system stability and provide easy management and alerting, while ensuring strong security. The implementation of a high available system should be as transparent as possible for the end user, with no noticeable changes in application speed and latency compared to the current configuration. A centralized management interface will allow administrators control and monitoring of servers and hardware, to reduce downtime.

A VM should be as independent from the physical hardware as possible to allow for changing hardware without any configuration changes in the VM. This will allow easy change of servers and server hardware. Moving VMs to new hardware as technology evolves, increases computing power while reducing costs. New software will not be constrained to a certain type of hardware, since only the hypervisor will need new drivers when changing the physical hardware.

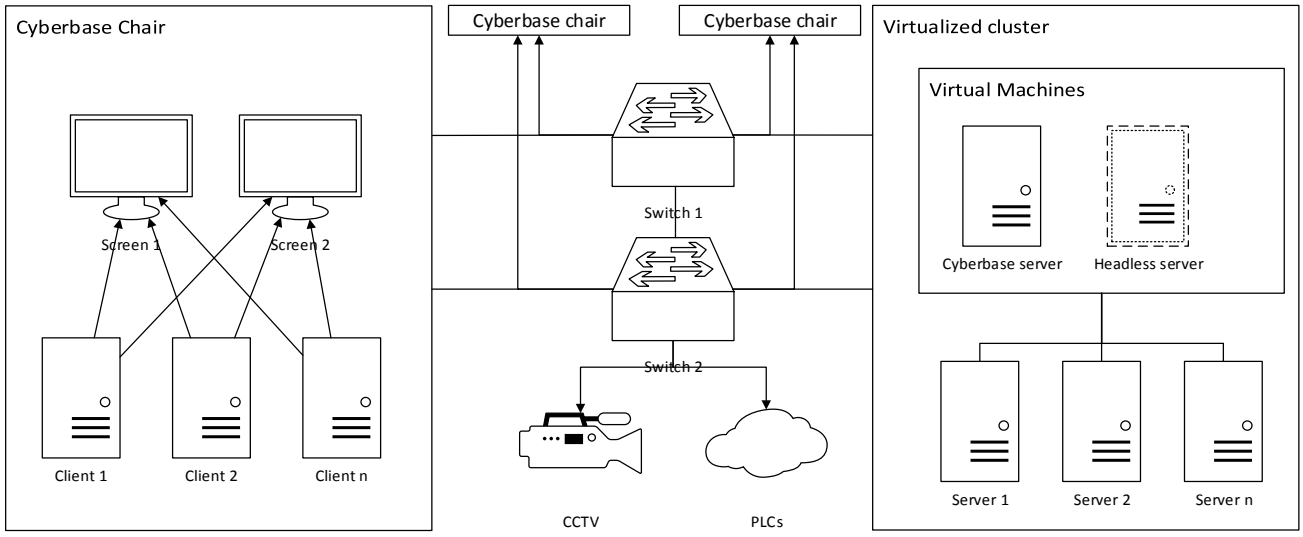


Figure 4-1 Possible configuration of fault tolerant system

Physical servers must be able to run in a setup that allows at least one server to fail without the system becoming unstable. It may take a long time for parts to arrive offshore and the system should be able to run in degraded mode while still protecting against failures. For this scenario, servers can be configured servers to run in a RAIN. Where server redundancy is similar to RAID levels.

5 CHALLENGES IN SYSTEM DEVELOPMENT

5.1 Self-developed vs turnkey solution

Developing and testing a distributed fault tolerant system is challenging due to its complexity and coupling of elements. Testing is very time-consuming and not always conclusive [9].

Developing a hypervisor and supporting functions that would provide the features desired would require large amounts of work. Testing needs to be extensive and time consuming since many products work together in different configurations. Faults can elude the testing phase and may cause serious problems when implemented for the end user. An in-house developed system would require extensive testing over long periods, without any guarantees that the results would work in a real world application. An alternative is to make software itself fault tolerant and independent of the OS through a project such as Apache Zookeeper [10]. This would allow programs to run the background processes distributed over multiple servers with a software displaying GUI to a user, while the server performs calculations. If a server should fail, the server side software would continue running on another server. Client side software handles the switch from the failed to the active server. This would be a reasonable solution if there were only one program or product that needed protection. Currently NOV have over 15 different software products that run in separate OSs, many with their own databases. Programs running within the same OS are a much higher number. Making all the programs distributed would require time and attention, and would hinder further development. Making new software distributed before it could be proven/sold will add extra time and costs to an eventual product.

This thesis will try to find a trailed and tested solution that fulfills the desired specification. The solution would preferably not require any changes in current software, and allow new programs to become fault tolerant without changes to the software.

5.2 Storage

Hypervisors use VDs to emulate physical hard drives. There are three general categories of VM storage. Local storage, each server has its VMs located on a local disk. The disk is then only accessible for a local VM on that machine. Shared storage, a SAN or NAS provides an area that multiple hosts can connect to and access VMs. Distributed sharing of local storage, where multiple nodes combine their local storage to form a self-contained SAN. A fault tolerant VM needs to have its VDs on a shared storage that is available to all the servers in a cluster.

5.2.1 Networked data stores

Network Attached Storage gives consumers file-level access to storage through Ethernet. Typical protocols include Server Message Block and Network File System. A NAS is typically one server with directly attached storage.

A Storage Area Network (SAN) facilitates block-level access to storage through the Ethernet, Fibre Channel, InfiniBand with others. iSCSI is the “de facto” protocol used for communication of these links.

5.2.2 Virtual Disks

Most hypervisors can thin provision disks. Thin provisioning grants disks a maximum size, but the disks only use the necessary amount. This allows over provisioning and better utilization of available storage. The storage capacity can grow dynamically when it is required, without the VM OS needing to have explicit support for thin provisioning. Thick provisioning have two methods for allocating storage, eager and lazy zeroing. The difference lies in how much of the disk will be setup during allocation. Eager zeroing creates a VD and zeroes every block in the partition during allocation. Lazy zeroing creates a VD with a given size, but does not allocate/zero the blocks before using them.

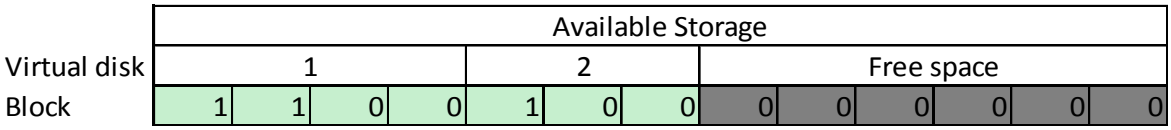


Figure 5-1 Eager Zeroed Thick Provisioning

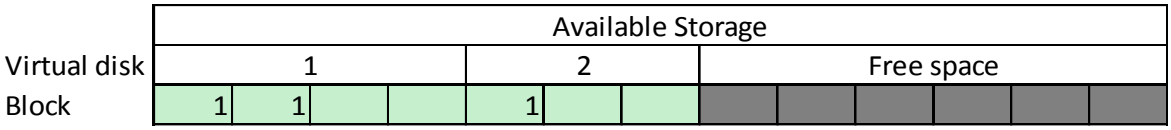


Figure 5-2 Thick Provisioning Lazy Zeroed

Thin provisioned disks does not allocate blocks during creation, instead they dynamically allocate free blocks when needed. When writing new files the thin provision storage provider needs to allocate free space on the data store. This requires resources and can slow down performance. It is hard to expect the exact future storage needs for machines. Over provisioning storage is the main advantage of thin provisioning, granting the machines more storage than is available. The machines that use all storage space granted, use the space that,

in a thick provisioning scenario, would be allocated, but may not be used by another machine. Thin provisioning can lead to over usage, where the VDs grow bigger than storage. Using all storage available will most likely cause virtual machines to crash.

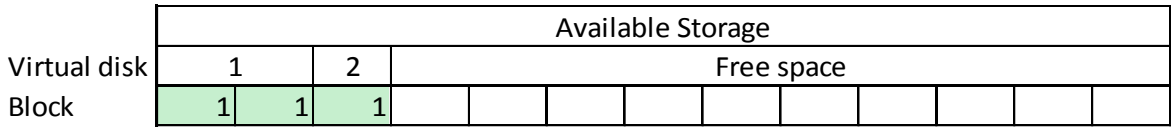


Figure 5-3 Thin Provisioning

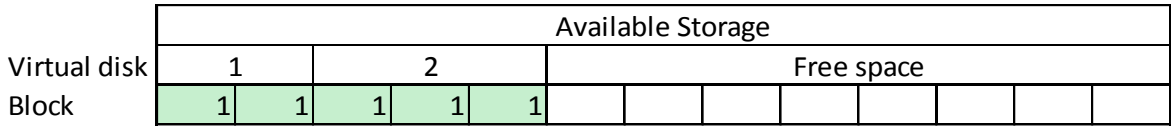


Figure 5-4 Thin Provisioning expanded virtual disk 2

Figure 5-3 and Figure 5-4 show the expansion of a thin provisioned disk, from the disk's perspective. The VD contains a limit on how large the disk can become.

5.2.3 Deduplication

Deduplication is a technology that allows for more efficient utilization of storage than conventional solutions. If much of the data is duplicates of each other, an algorithm tries to find either file-level or block-level duplicates, and combine the two duplicates to store one single copy. Replacing duplicates with pointers gives a reduction in storage size. This can give a reduction of stored data in the range from 0-99% depending on the type of data and deduplication method used. File-level deduplication checks entire files against each other and only labels the file as a duplicate if an exact match exists. Almost identical files will be stored as separate files, not giving any advantage. Block-level deduplication checks each block for similarity. This results in a higher deduplication ratio, but a larger metadata log. Adjusting the block size according to the data stored will allow for optimal performance. Low block size will result in a large metadata file, but high deduplication ratio. To perform comparisons, deduplication requires resources from CPU and RAM. Disk access can be high, depending on where the metadata log is stored. In a hybrid-storage solution, where SSDs are used for cache and HDDs for permanent storage, deduplication is useful because it can allow more files to remain in cache. When using deduplication as the first stop towards storing, the data takes up less space and there is room for more data, resulting in less delay when retrieving the data.

5.3 Computer hardware

5.3.1 Server

Hypervisor hardware provides the foundation for a high tolerance setup. Some components allow redundancy by default, others must be used together with software. Redundancy functionality in core components such as CPU, motherboard and power supplies is commonly available and does not require extra resources in use. Server grade hardware utilizes components with this functionality, for instance ECC memory and redundant power supplies.

Hardware support varies in different operating systems. Hypervisor manufacturers provide a HCL, a list containing compatible hardware. The HCL is generally restrictive and only list verified components. Using components not on the HCL should be avoided.

5.3.2 Client

A client machine that connects to a virtualized server can either be a fully featured computer, small zero client or a thin client. Each type of client runs a simple OS that makes a connection to a server possible. The thin client only provide KVM functionality, therefore a failure of a client will not affect a VM. Another client can resume the session from where the failed client stopped. Adding multiple of “standby” clients will provide fault tolerance on the client side.

A thin client can perform some features but still relies on a central server to function. The client can for instance do some hardware acceleration, such as decoding video. Performing the task on the thin client reduces network traffic and server load.

Zero clients are more dependent on a central server than a thin client. Zero clients are more secure because they have fewer points of attack and no local storage. A Zero client usually boots from PXE, but both types can use this functionality. Network booting a client OS makes administration easier, but lowers the system independence. Since all the configuration parameters are centrally located, applying changes to all clients is easy. If the service providing PXE features experiences failure, the client cannot boot. This could become a single point of failure.

5.4 Monitoring

Monitoring the status and health of system components are crucial to ensure the detection and handling of failures. An alarm system monitors hardware and will alert the operator if a

component is not performing normally. Components such as servers, switches, power supplies, UPS and more are monitored. Monitors retrieve information from a device by polling values. Active and performance monitors are used. An active monitor polls a device and checks the status. An alarm will trigger if a value is detected unhealthy i.e. a disk is unplugged. Performance monitors check the value against a limit, this allows early warnings for such as disk write error. Early detection of errors is always preferable. Many network monitoring softwares can trend values to predict failures and give alerts before a failure happens. To fully utilize the monitoring capabilities, all devices must support monitoring through a standard protocol such as SNMP.

5.5 Network

The network interconnects devices and enables communication and partitioning. High availability in servers will not work without a functional network. Redundant links should be used between hypervisors and switches. Combining multiple NICs will allow traffic to continue through another NIC if a failure occurs. Multiple connections between the server and network switches ensure connectivity in the event of a switch failure. Interconnecting switches provide more paths and are required to handle NIC failures on multiple devices.

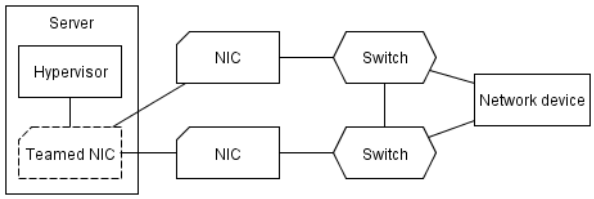


Figure 5-5 Redundant network

5.5.1 Jumbo frames

Standard Ethernet packets consist of 1500 bytes that are encapsulated in the media layers of the OSI stack. Encapsulation requires resources from the computer, especially CPU. To lighten the system load from network communications Jumbo Frames were developed. JF have a high packet size, making it possible to gather multiple Standard Ethernet packets into one JF. The packet size is not set by a standard, making multiple lengths available. 9000 byte size is commonly used, and make room for six Standard Ethernet frames in one JF.

5.5.2 Spanning Tree Protocol

STP (IEEE 802.1D) is used to prevent loops when connecting multiple layer 2 network switches in a mesh topology, and allows for redundant links that fail over. A root device in the network is either elected or configured to calculate the minimum spanning tree. The election process gathers the MAC-addresses for all the switches in the network, and selects the switch with the lowest and therefore oldest MAC as root. This can cause a suboptimal setup in networks where both old and new network equipment is present. An old switch will be chosen to perform operations that newer switches could do faster. Manually configuring a root switch is therefore a better option. The root switch determines the shortest path to each device, based on a least-cost tree. Ports not within the least-cost tree are set as blocked. If a link fails, the root device calculates a new least-cost tree and implements it throughout the network.

5.6 Virtualization technology and products

Virtualization can be used to enable seamless uptime during hardware failures, it is a technology that abstracts software from the physical hardware. This makes an operating system independent of the underlying hardware configurations. Virtualization hypervisors divides into two types. Native or bare metal that runs as its own operating system, or hosted running on top of an operating system. Bare metal hypervisors provide the best configurability, security as well as performance and is most common in server environments. A VM is a computer that runs on a hypervisor.

Configuration of the physical machine does not differ from a regular machine. A VM can use virtual devices that may or may not be a representation of the physical hardware. This allows multiple VMs to run on one hypervisor. Implementing fault tolerance on the server side can be accomplished by using multiple hypervisor setups in a cluster. A central application handles the failure of nodes. Servers are setup with a central storage that handles all the data stores. The servers themselves only store the hypervisor OS locally.

Migration is what happens when a VM moves from one server to another. The move generally classifies into two categories, online and offline. In online the VM is running through the process and is seamlessly available to users during the move. Offline mode shuts down the VM and disconnects the users, before restarting the VM at the other host.

Pass-through allows hardware to be directly attached to the VM. A VM can have expansion cards and mainboard features, such as disk and network controllers, directly mapped to it. This

allows applications that rely on special hardware to work as usual, even though they are running in a virtual environment. However this makes the virtual machines dependent on specific hardware and disables HA functionality in most hypervisors. Avoiding the use of pass-through is preferable.

Monitoring of the server that a hypervisor run on, can be separated into two general categories, hypervisor internal and external. Internal monitoring run a service in the hypervisor and checks hardware status. This can be accomplished by using a driver module that lets the hypervisor contact hardware directly or letting the hypervisor gather out-of-band management information. External monitoring does not involve the hypervisor OS, but gets values directly through out-of-band management. External monitoring is OS independent can run with together with all hypervisors, if the hardware configuration supports it.

It is important to separate server and desktop virtualization. The former takes a server, a machine that runs headless, and converts it to run on a hypervisor. Desktop virtualization also run on a hypervisor, but focuses on the client/operator and the interface he uses to perform tasks. If many clients use the same setup, only separated by small settings such as machine name, IP, user settings etc. a desktop virtualization with a “golden” base image is can be made. This image have the general setup, and only changes done by users are stored in an own file, reducing storage needs. Many desktop virtualization providers have their own client program that connects to the virtual server(s). This program provides an optimized user experience by reducing bandwidth required. The program gives a user the appearance of working on their local machine, when they are working in a VM that handles the processing. This software can run on many different types of hardware and since the server performs the computing, only a small amount of processing power is required.

5.6.1.1 Remote Desktop Protocols

Users connecting to a virtual desktop use a remote desktop protocol to facilitate a normal user experience. The RD protocols divide into two areas of application, remote assistance and remote experience. Remote assistance is mainly used by support personnel to assist users with IT problems. Interacting with the same desktop multiple places at once is called shadowing. Remote framebuffer is often used in RD protocols for shadowing. This method is based around

"*put a rectangle of pixel data at a given x,y position*" [11], but still includes more advanced features such as compression of rectangles. Remote assistance have less focus on a desktop user experience and more on usability and seamlessness, since sessions are short and not used for everyday tasks.

Remote experience tries to give a user the appearance of being on a local desktop, with high performance and support for graphics and videos. Remote experience VMs are often hosted in data centres in a Virtual Desktop Infrastructure that allow users from different locations, both local and remote, to connect. Remote experience protocols often strive to use as little bandwidth as possible to reduce requirements on both server- and client-side. This can be achieved by using a lossy compression algorithm or optimizing the protocol, i.e. by sending graphics objects and making the client render them. Some protocols require an own machine that brokers client connections. This simplifies management but becomes a single point of failure. The remote desktop must allow a direct connection between server and client. Both types of remote desktop can use a combination of TCP and UDP. UDP allows for lower bandwidth consumption, due to its unreliable connectionless nature without acknowledgement of packets. TCP is connection-based and considered reliable compared to UDP. Remote desktop protocols often use UDP for streaming screen to the client and TCP for sending user inputs back to the server.

5.6.1.2 VMWare

VMWare, subsidiary of EMC, is one of the market leaders in virtualization. They provide multiple virtualization solutions, both native and nested. ESXi is their native hypervisor. A vSphere vCenter Server manage multiple ESXi hosts, and enables various high availability features.

ESXi has a built in SNMP service, and support physical hardware monitoring through third party modules. VMotion is VMware's protocol for VM migration [12].

5.6.1.2.1 Failure handling

VMWare provides several features that handle a failure in different levels, according to how critical the VM uptime is. High Availability ensures that a VM moves from a failed host and starts in a new one. HA requires shared storage and at least two independent network links. If a node fails an offline migration is performed. Mission critical VMs can use Fault Tolerance to

ensure higher reliability. FT runs a primary and secondary VM in parallel on two different nodes. If the primary node fails, the other seamlessly takes over. FT requires an extra network link in addition to the two used for HA. One major limitation with FT is that it (at the time being) does not support more than one CPU core and therefore the VM performance may suffer. Both HA and FT require a central vCenter Server to monitor and manage the nodes and VMs, but does not require a running server to perform the functionality.

5.6.1.2.2 Storage

VMWare vSAN enables distributed storage from ESXi hypervisor nodes. The local storage of the nodes can be combined into a pool with similar level of configuration as RAID 0, 1 or 5. Storage pools in vSAN, are a hybrid configuration consisting of both SSD and traditional HDD drives [13]. A SSD caches the data before it is written to HDD. The SSD capacity will not contribute to the total storage capacity of a node. Not being allowed to use a pure flash based for storage will negatively affect performance and reliability of vSAN as data store.

vSAN allows all nodes access to the storage pools, even though they have no local storage. This makes it possible to have a skewed relationship between storage and computational nodes, where storage nodes can perform both tasks. If a vSAN data store run in RAID-1 configuration, the cluster can be set to tolerate a number of failures up to $n - 1$, where n is the number of data storage nodes.

5.6.1.2.3 Security

VMWare vShield monitors the network traffic between the network and a host, and intercommunication between VMs. Applying security profiles on a VM basis, can help reduce resource usage.

5.6.1.2.4 Remote Desktop

VMWare View is VMWare's remote experience solution. View Agent service running in each VM enables connection, client device redirection and management. VMWare View is made for use with a central management server, however a direct connection add-on enables 1:1 connections without the management server. VMWare offers connection clients for a most popular OSs and thin clients.

5.6.1.3 Citrix

Citrix provides virtualization services through their Xen product range. The core product XenServer is a native open source hypervisor, licensed under a General Public License. Citrix offers a paid version which includes more features, 24/7 support, automated updates and access to technical articles. XenServer coes with a built in SNMP service for monitoring. A XenCenter manages a pool of XenServers [14] [15].

5.6.1.3.1 Failure handling

XenMotion allows for offline and online migration, if a physical host fails only offline is supported. Citrix does not have a functionality that allows failover with zero downtime, but third party extentions are available. These will not be covered in this thesis.

Citrix HA focuses on reliably detecting failures and shutting down hosts to avoid multiple machines from performing simultaneous operations. To achieve this XenServer monitor both the data store and hosts in a pool by heartbeat. Regular writes to the data store avoids one VM from running on multiple hosts in a split-brain scenario. If a pool is separated into multiple parts, the smaller group of hosts shut down hypervisor operations on a very low level, Citrix calls this functionality server fencing.

5.6.1.3.2 Remote Desktop

XenDesktop leverages Citrix's remote experience and can run in Windows Server or as a self-contained virtual machine/appliance. XenDesktop does not allow direct connections, and is therefore not applicable to the desired setup.

5.6.1.4 Microsoft

Hyper-V is a Microsoft hypervisor product. It can run as a pure native hypervisor or as a role from Windows Server. In both cases, it is a native hypervisor, but Hyper-V virtualizes the Windows Servers as a "root" VM that have a closer coupling with the hypervisor than other hypervisors. Hyper-V run a SNMP service, that can be used for monitoring purposes.

5.6.1.4.1 Failure handling

Microsoft offers Replica as one option to reduce downtime. This feature enables replication between sites over a LAN or WAN connection. Because of the high delay and reduced bandwidth, Replica does not use heartbeats to detect host failure. The head server writes a log with all changes to the slave nodes on an interval that adjusts according to the data rate between the servers. Replica periodically checks the log for updates, failures are detected

when one host has not written to the log within a given interval. By default log updates happen every 5 minutes, a timeout will not occur until 25 minutes, amounting to 30 minutes from host failure to detection. The Replica system's scope is too far from the scope of the desired specification, because of it's long downtime.

Microsoft Guest Clustering support high availability by restarting a VM on another node when a failure occurs, but a Windows Server OS is required to use this functionality and is therefore not applicable.

5.6.1.4.2 Remote desktop

Microsoft's Remote Desktop Protocol has been integrated in every Windows OS since Windows 2000. RDP version 8 adds support for DirectX11 and device redirection. RDP support delta-rendering and added graphics performance through RemoteFX. The shadowing functionally featured in version 7 was removed in version 8 due to security issues. RDP is a direct connection centered protocol, but also support a connection broker.

Company	High Avaliability	Fault tolerance	Self-contained HA Storage	Thin provisioning	Management
VMWare	Yes	Yes (1 core)	Yes	Yes	Web
Citrix	Yes	No	No	Yes	
Microsoft	Yes (Windows Server)	No	No	Yes	

Table 5-1 Functionality of Virtualization products

	Streaming technique	Rendering type	Shadow connections	Connection method	Protocol
VMware View	TCP/UDP	Delta	No	Broker, direct	RDP, PCoIP
MS RDP<6.0	TCP	Frame/dirty	No	Broker, direct	RDP
MS RDP 8.0	TCP/UDP	Delta	No	Broker, direct	RDP
Citrix XenApp	TCP/UDP	Delta	View mode only	Broker	RDP, ICA

Table 5-2 Remote desktop comparison

5.7 System Solution Evaluation

5.7.1 Storage

A single central storage device would be a single point of failure and is not acceptable. If all the VMs are stored on such a device, a failure would make a big impact on the system as a whole. This configuration would reduce the ruggedness compared to the current system. At least two specialized storage servers running in fault tolerant configuration would be required. If the number of storage servers is less than the number of servers, the failure tolerance of storage system lowers the overall tolerance. Two servers would need to have an exceptional good uptime. Storage systems with HA capability are very expensive, easily exceeding the cost of the current servers. Systems would require certifications to allow offshore use, adding further cost. Utilizing existing hardware to form a storage cluster facilitated by software would be a less expensive option, but might have performance issues. Mirroring data on two or more hosts will give a lot of overhead. This would amount to RAID-1 on top of RAID-1, 5 or 6, giving a usable space of $\frac{disks - redundancy\ disks}{servers\ in\ cluster} * disksize$ which is a poor utilization. Should one server fail, one whole copy of the data will become offline, and a minimum of three servers are required to maintain continued system protection. The active server has to synchronize with every passive server constantly yielding duplicate network traffic from one source to multiple sinks. Finding user friendly and reliable software to manage the storage could be a problem, since many HA solutions are complete systems consisting of both hardware and software.

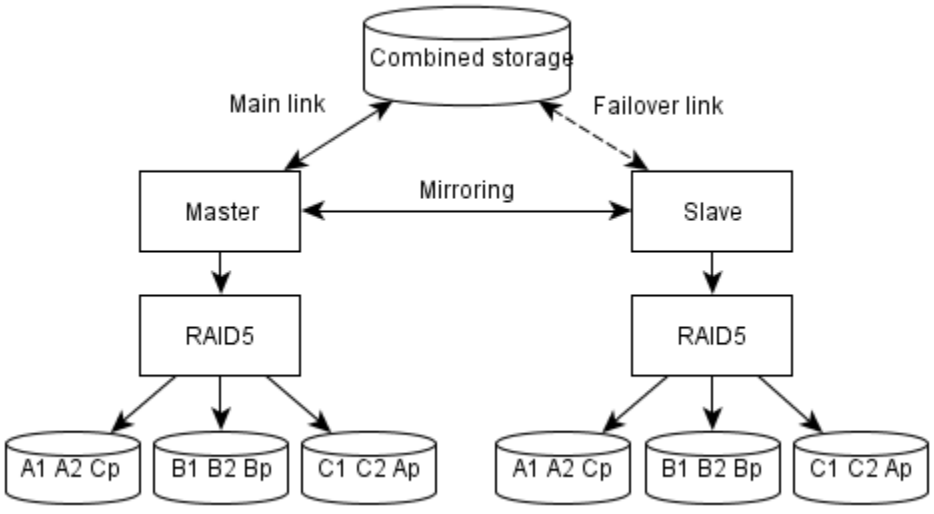


Figure 5-6 Mirrored storage

Building a RAIN and sharing the storage across servers running the VMs would give better utilization than an active/passive mirror configuration, but such a setup requires resources to facilitate storage. As the disk usage of a VM increases, the storage-facilitator will use more resources. This gives less resources available to VMs compared to storing data in an independent cluster. Total usable space is the same as for regular RAID arrays.

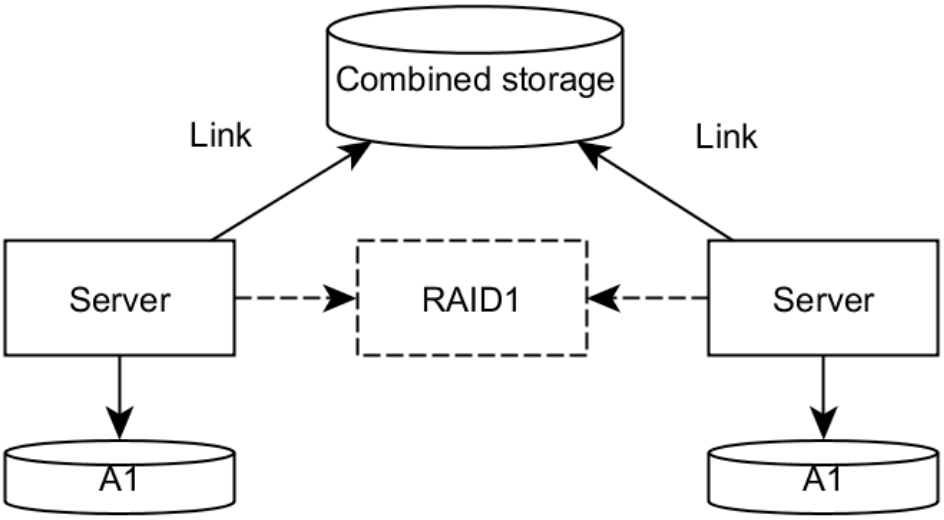


Figure 5-7 Distributed storage

Using deduplication in combination with a RAID-1, a system can increase the amount of storage available while still having a high redundancy. This will increase the storage utilization. If hypervisors don't have deduplication capabilities the storage facilitator can be run as a VM, but this have several drawbacks. VMs that provides storage to their parent hypervisor complicates the system a great deal. It also divides the storage in more independent layers than desired. A problem with the storage VM could render the whole system unusable, and hard to troubleshoot for service personnel.

VMWare vSAN the most fitting alternative, providing HA storage distributed between the nodes in the cluster. It allows multiple storage configurations to form a RAIN, where multiple hardware failures will not cause the system to halt. The drives are directly connected to the hypervisor, without the use of RAID controllers, and can be managed through a central management console. This does not increase the complexity or difficulty for system operators.

5.7.2 Virtualization

Virtualization is the technology to use when making multiple software/OS more resistant to hardware failures. Providing an HA platform for developers allow current and future software to automatically gain an independence from hardware related failures.

Comparing the solutions from the three virtualization providers mentioned, no one can provide all the features needed for a system with bumpless failover. VMWare, Citrix and Microsoft provide failover capabilities, but they all have downtime while the VM restarts on another host. VMWare FT runs a VM in parallel on two hosts similar to a VM RAID-1, but has severe limitations that make it inapplicable for the desired system. RAIN-1 will allow for bumpless failover between two hosts but does not protect if both hosts fail simultaneously, or allow for a hot spare. VMWare FT currently only supports one virtual core, restricting the resources available to the VM. The performance implications for a program designed to run on multiple cores would be high. If a VM does not have enough resources to operate satisfactory, the user experience will decrease.

Microsoft's solutions for high availability are directed towards their own services or third party programs running in Windows Server OS, which render it impossible to use in this application.

5.7.3 Remote connection client

Both thin and zero clients have their respective advantages and disadvantages. Thin clients are independent from a client management server, by booting from a locally stored OS. But they are harder to update and configure compared to zero clients that PXE boot over the network. The hybrid solution sketched in Figure 5-8, combines local storage with central management and PXE booting, and combine the advantages of both thin and zero clients.

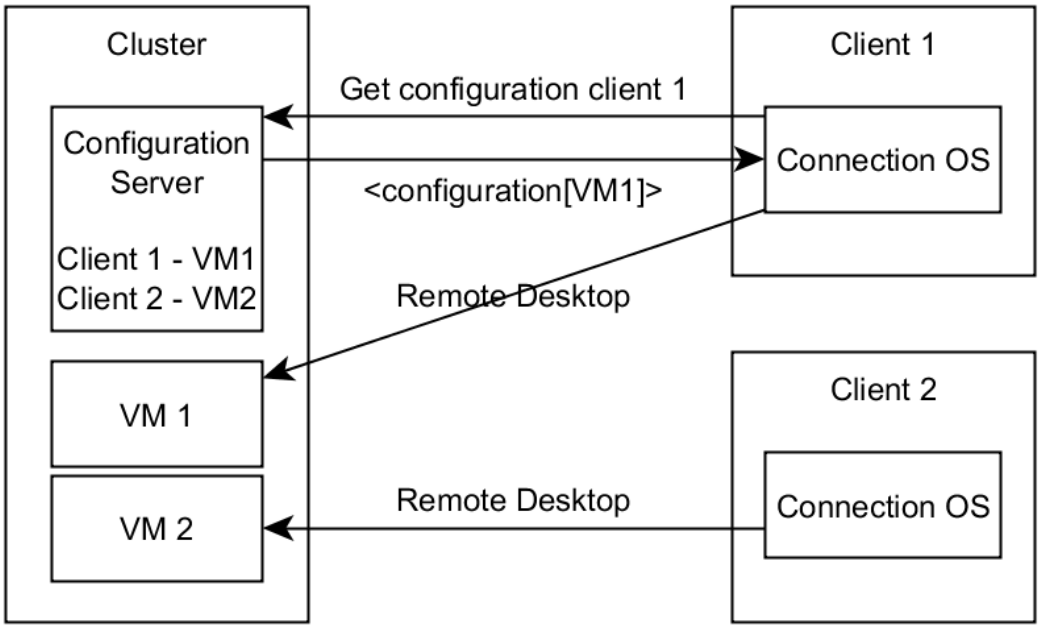


Figure 5-8 Connection client setup

A central configuration server (CS) runs in and benefits from the high availability features it the cluster offer. The CS contains a PXE server, which the clients boot from to update settings and OS. Each time the client boots, it performs a check to see if a new OS version is available from the CS. If the PXE server is faulty the clients boot from a connection or thin client OS stored locally. When the connection OS boot, it connects to the CS and gets information on which VM to it should initiate a RD session with. The last successful connection configuration is stored locally on the VM and is used if the CS is offline. This setup is relatively complex but offers central management and allows clients to function without a management server.

5.7.4 Security

A system lacking in security will be rejected no matter how many of the other specifications it satisfies. Anti-malware software must run in monitor mode only, detection of irregular activity can only notify users. Automatic deletion or quarantining of files can hinder the functionality of the system.

A strict firewall policy can hinder some attacks against the network. Most hypervisors support a firewall filtering before any traffic is directed to a VM. Applying firewalls to intercommunication between VMs can be considered if the VMs OS firewall does not provide sufficient security. During normal operation, networks involved in the system are closed from the outside, analyzing all traffic and VMs could introduce undesired delay in the network. CCTV used to

control machinery is dependent on low latency to function properly. Continuous monitoring of network and VMs may require too many resources, compared to the added security they provide. Monitoring for malware can be used in high-risk periods. Phases such as initial setup, commissioning, service and upgrades, when multiple service terminals are connected to the network, have a higher threat level than normal operation. Monitoring during these phases give protection, while not degrading performance during operation. Responsibility to enable scanning is put on the service personnel with potential hazardous consequences if service is performed without scanning activated. Some security must therefore exist that doesn't allow outside connections to the system before enabling security features.

5.7.5 Comparison of available products

More information about virtualization and developing fault tolerant systems are found in [16] [17].

Comparing the available virtualization and failover capabilities of products available, VMWare provides a solution closest to the features sought after in the desired specifications. While VMWare FT have the functionality needed, significant drawbacks exist. Using one CPU core will degrade performance. Because of these limitations using vSAN for storage and VMWare HA to reduce downtime provides the relatively best setup. Zero downtime is not achieved, but the reliability is increased compared to the current solution.

Equal firewall rules can be implemented for every hypervisor. Firewall features alone don't give an advantage to either product. Citrix encourages using an independent security specialized OS, running as an own VM. This method can be used no matter which hypervisor is used. Third party OS would require more configuration and complicate the internal networking of a host to allow traffic interception, but could give more advanced features. Microsoft does not provide a security product, especially for Hyper-V, but has anti malware software for their OSs. This software is not applicable since it does not run without user interaction. VMWare vShield integrates with ESXi to provide DPI both between hosts and VMs. Without including third party solutions, vShield is the best security application alternative.

6 SYSTEM TEST RESULTS AND ANALYSIS

This chapter contains the tests performed on the system, dividing tests into layers, where each layer is dependent of the previous/underlying layer to function properly.

Testing hardware components that provide self-contained high availability capabilities will be assumed working and not receive testing as other components and software. These components include redundant PSU and watchdog timers that operate on a hardware level, and do not require any interaction from drivers or software.

6.1 Test setup

As discussed in 5.7.5, VMWare provides the best total solution for increasing reliability, although it does not provide zero downtime. Only VMware's vSAN will be tested in the section due to that no other products can compare with the features it offers.

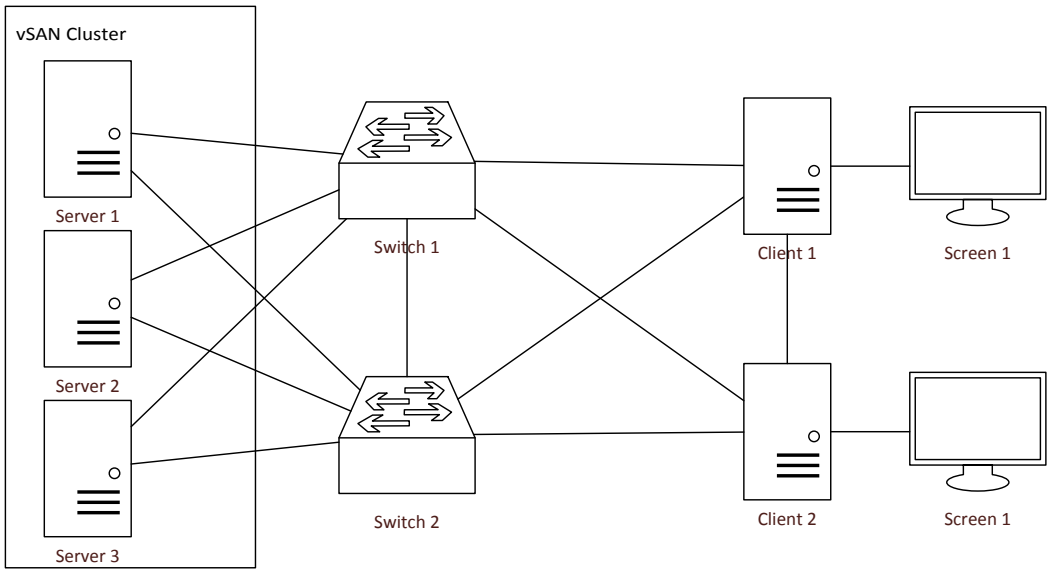


Figure 6-1 Test setup

Two different computer node types, server and client are used for testing. Each node has one LSI 9260 RAID controller, with single drive RAID0/Pass-through mode for testing. Client machines are fully-fledged computers, capable of running multiple OS without any performance issues. This avoids that the computational performance of the clients become a limiting factor. The servers are connected to two HP A5120-24G switches by two cables from each server to a switch. The switches are connected to each other by redundant cabling, and

use STP to avoid network loops. The thin clients are connected to the switches by two teamed NICs.

	Server	Thin client
CPU	Intel i7-i3770	i5-4570S
Chipset	Q77	Q87
RAM	32 GB DDR3	16 GB DDR3
GPU	Intel HD4000	Intel HD4600
NIC	4 x Intel 82574L	2 x Intel i211
SSD	512 GB	128 GB
HDD	1 TB	-
Other	LSI 9260 RAID	-

Table 6-1 Server and thin client hardware setup

The virtual machines and thin clients used for testing have been automatically installed and configured using a Windows deployment system. This ensures that settings and updates are equal for all of the machines.

The computer hardware used in this test is mainly high-end desktop class, but some components are server class. The Q77 chipset does not support ECC memory, an important feature for ensuring system stability and avoiding memory fault. The four-core i7 processor is aimed at performance desktops, but virtualization support makes it capable of running hypervisors with good performance. The Server's four network cards are server grade with CPU offloading. Out-of-band management technology is not supported by the NIC, complicating remote management. All the network equipment run at gigabit, VMWare recommend 10 gigabit networking for vSAN [18]. SSD drives used in servers and thin clients are Transcend enterprise grade drives connected by SATAIII. The drives support Trim and NCQ. The server hardware is compliant with VMWare's HCL.

6.2 Hypervisor-Hardware Layer

The first layer is hypervisor and hardware interconnection. Ideally all HA functionality is placed in this layer, making the overlaying layers work without any changes. Testing of OS and software in the above layers is done to verify that the HA is working correctly, and does not introduce any faults. File systems are especially vulnerable to abrupt changes when machines crash, since software must close files after writing so they do not become corrupt. Different

methods can be applied, either closing the file after each write, or closing the file when the program closes. The last alternative is often a bad solution when a machine abruptly shuts down. Each application has its own way of handling such failures, and will have to be tested independently to ensure that the specific application pass a hardware failure test. During testing, it is assumed file corruption has not occurred if Windows start without errors. The BIOS of the physical servers is setup to “resume last power state” after power is lost. This causes the server to automatically power on and resumes operation after a power loss. When power is disconnected on a running Windows 7 machine, a troubleshooting dialog appears during next boot process. This dialog display several choices to enable easy recovery of the machine, the dialog waits 30 seconds before a normal boot is automatically continued. This dialog should in a production scenario be disabled to allow VM a lower reboot time. In testing, this dialog is enabled and 30 seconds is deducted from the reboot time for a power failed VM.

Hypervisor testing use TCPing [19] to determine if a host and its services are running. This program opens a TCP connection to a host with a specified port. Using ICMP ping will only give the status of a network card/service. The difference between a machine replying to ICMP ping and the services running can differ. The hypervisor hosts used in testing replied to ICMP 36 seconds before replying to a TCP connection. Every target is pinged to check availability and response times of services. The timeout delay is set to 500 milliseconds. Services not responding within the time will be marked as unavailable. Most important is the port(s) used for remote desktop, as they show how long the downtime is before the system can resume operation. The hypervisors are monitored on port 443, used by vCenter and client for host management. The startup time of GUI applications are not tested. The latency for hosts are monitored in each test, but components insignificant for a specific test are removed from test results. Online or offline bars display the test results to increase readability. A system bar shows the status for the system as a whole.

6.2.1 Hardware failures

The system should be seamlessly operational during a hardware failure, to satisfy the requirements in this layer.

Step	Time (s)
BIOS POST	8
LSI	36
PXE init	8
ESXi	194
(vSAN)	*(96)*
total	246

Table 6-2 Server boot time

Table 6-2 shows the different steps involved in the boot process of a server, and the time consumed by each. This is results from a server cold booting while connected to the vSAN cluster. Booting ESXi is the most time consuming operation, using a total of 194 seconds of which 96 is used for loading vSAN. The LSI RAID card also requires a considerable amount of time considering that it only does pass-through of the connected drives. Replacing the LSI card with the onboard Intel SATA chipset would reduce boot time, unfortunately the vSAN version tested did not support the SATA chipset used in the test servers. In comparison a server in the current system, use 21 seconds to boot into Windows without the LSI card.

6.2.1.1 Hypervisor failure

Unexpected reboots can happen to servers in any environment. In the event of such a scenario where a host shuts down and powers up again, the system must be consistently stable to allow continued operation and failure notification. The test checks how long it takes to power cycle a hypervisor and the effects it might cause to the rest of the system. Servers 1, 2 and 3 are running, when server 2 has an abrupt power failure before the power is reconnected.

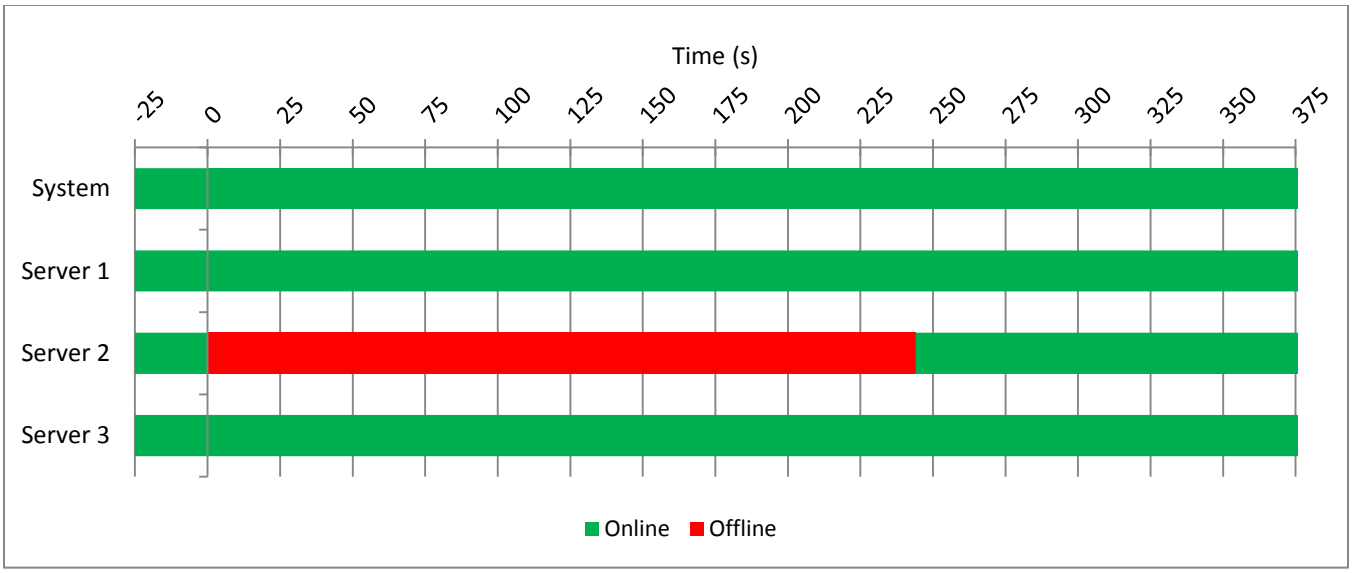


Figure 6-2 Host failure

After power failure at time 0, the system is still operational and no irregularities are observed in the latency time of each host. The host is offline for 244 seconds before coming back online and the cluster continues to operate throughout the test. A single host failure does not have an impact on the system.

6.2.1.2 Hypervisor and VM failure

This test checks the time and influence a power failure of a physical server running a VM have on the system. A VM is running on server 2 when the power is abruptly cut. After the server has powered down, power is reconnected.

Figure 6-3 Host and VM power failure

After the power failure the ping latency for server 3, the one running vCenter, shortly spiked indicating that the management is aware and handling the failure. The VM is offline for 114 seconds before restarting on server 3. Server 3 has some small latency spikes during test, probably since the move and reboots operations for the VM are resource intensive. Server 2 resumes operation after 248 seconds, making the boot time 2 seconds longer than the boot time found in Table 6-2 and 4 seconds longer than 6.2.1.1. Running a VM in the cluster does not seem to have an impact on the time it takes a host to reboot. 114 seconds unavailability for the VM is considerably better than the current solution, but is not close to the zero downtime sought after in the desired specification.

6.2.1.3 Host and VM failure without management server

In the event that the management server is offline, HA features still have to be attained for the cluster, or the management server becomes a single point of failure. In this test the management server is manually powered down, before a host and VM failure occurs. The VM should automatically be restarted on a different host the same way as in Hypervisor and VM failure test.

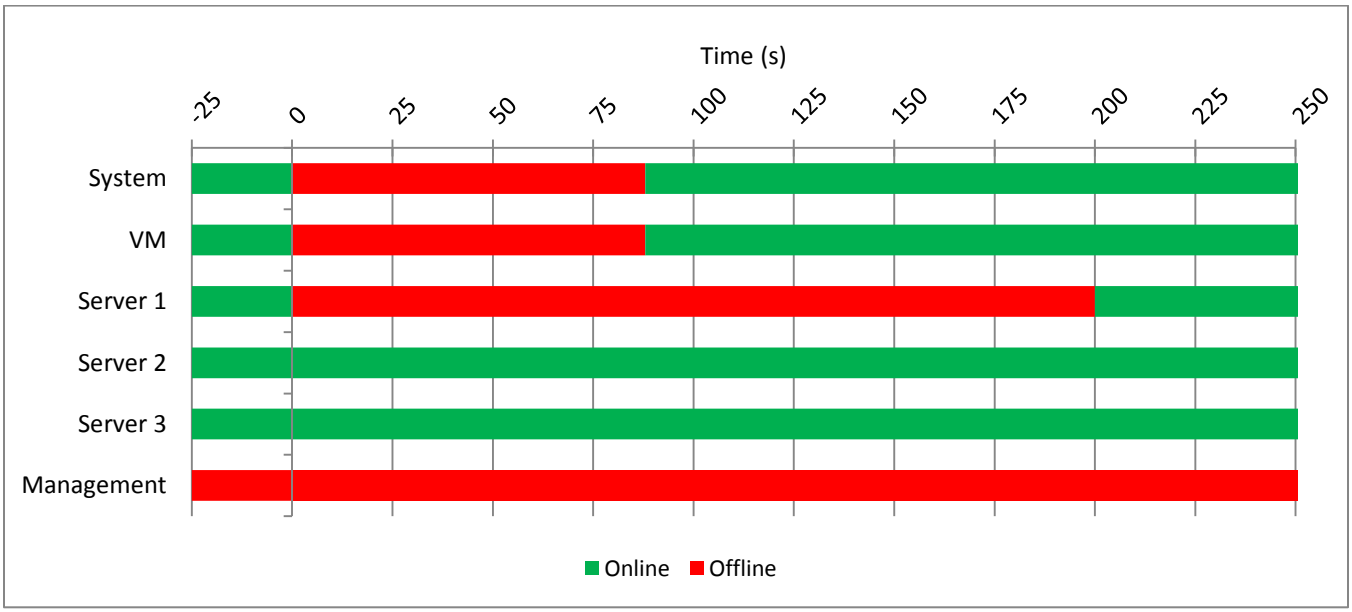


Figure 6-4 Host failure without management

The management server is offline for the duration of the test. At time zero server 1 which runs the VM fails, and the server and VM goes down. The VM and server 1 are available after 88 and 200 seconds respectively. The results from this test show that the management server does not have to be online for high availability functionality to work. In this test, the system regains full functionality faster than the test with an operational management server. This shows that the management server slows down operation while active. The 26-second difference is most likely caused by slowness somewhere in the management system. Using the management server under testing always seemed slow, but no tests results for this exist. The management virtual appliance, have been assigned resources in compliance or exceeding VMware's recommendations.

6.2.1.4 Drive failure

Drives are important both for storing data generated in operation and house the VM data stores. The failure of a redundancy drive should not have an impact on the system. This test disconnects drives from their hypervisor host, and monitors the system for changes. The cache SSD and storage HDD are disconnected simultaneously by unplugging power and data cables. The drives are disconnected from server 1 and 2, with a waiting period in between.

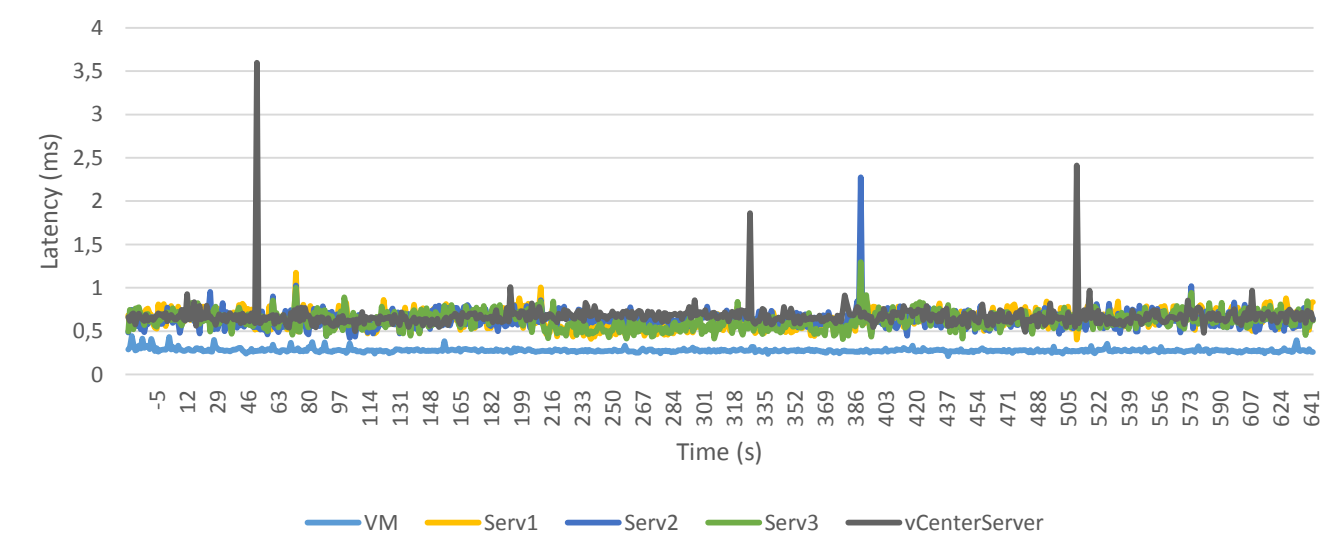


Figure 6-5 Drive failures latency

At time zero the drives are disconnected from server 2. The system does not show any signs of a disk failure. 120 seconds after the first failure, drives are removed from server 2 without any noticeable results. After letting the cluster run a while after the two failures, all drives are simultaneously reconnected. This causes a small latency spike on all hosts at 389 seconds. The VM run without any noticeable effects from the failures and latency varies between 0,2 and 0,4 milliseconds for the whole test. The vCenter management server occasionally has very small latency spikes. These are a very low increases and does not have any influence on the cluster.

6.2.1.5 Switch failure

Each server is connected to two network switches with redundant links. In the event of a switch failure the servers should automatically fail over and continue operation through the other switch. This test checks how the cluster handles a single switch failure. A VM is running on the host server 2 throughout the test.

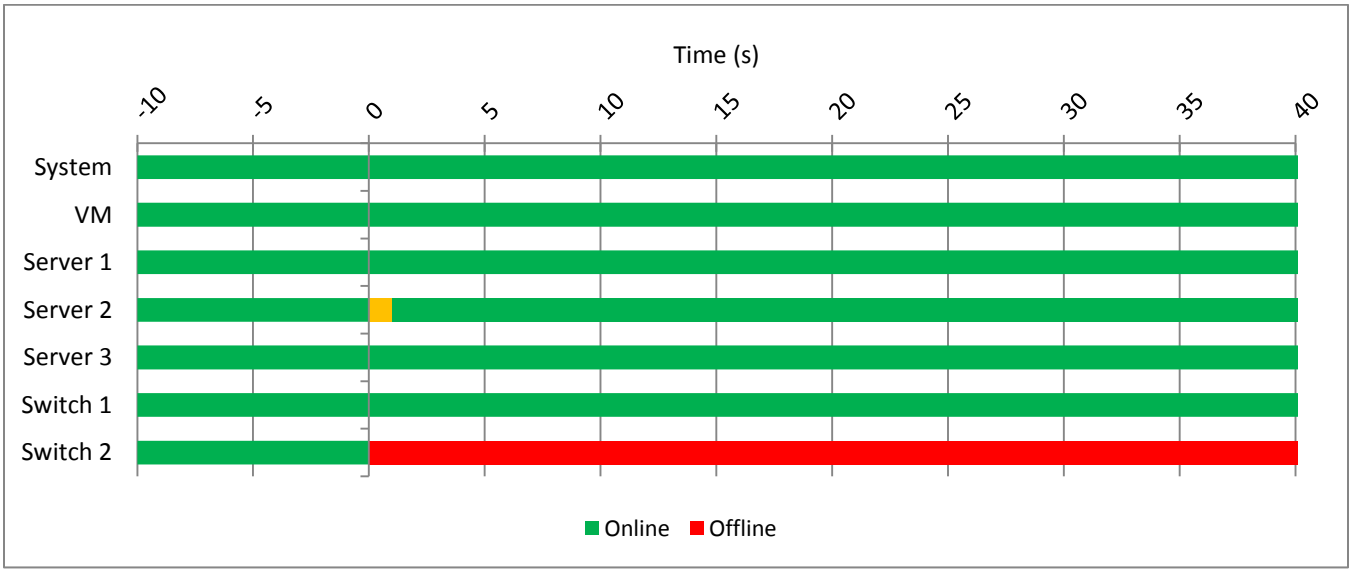


Figure 6-6 Switch failure device uptime

After 18 seconds the power is disconnected from switch 2. Server 2 drops one ping immediately after the switch fails. This is probably due to that the active network between the host and the server running ping were connected through switch 2. Connection to server 2 is regained on the next ping. Since the management of server 2 is not directly involved in operation, one dropped ping of the hypervisor is not a serious failure. The VM running on server 2 does not timeout throughout the test, and is continuously available. The hypervisor automatically switches the VM from a failed to an active path.

6.2.1.6 Switch reconnection

The network is setup with the STP protocol to avoid loops. When a new switch connects to the network a recalculation of the minimum paths will be initiated. This can happen in case a switch is offline for a short time period, such as a power cycle. In this test a switch is taken offline and restarted. Switch 1 is configured as the STP root device, and remains powered on for the test. Switch 2 is power cycled at the start of the test.

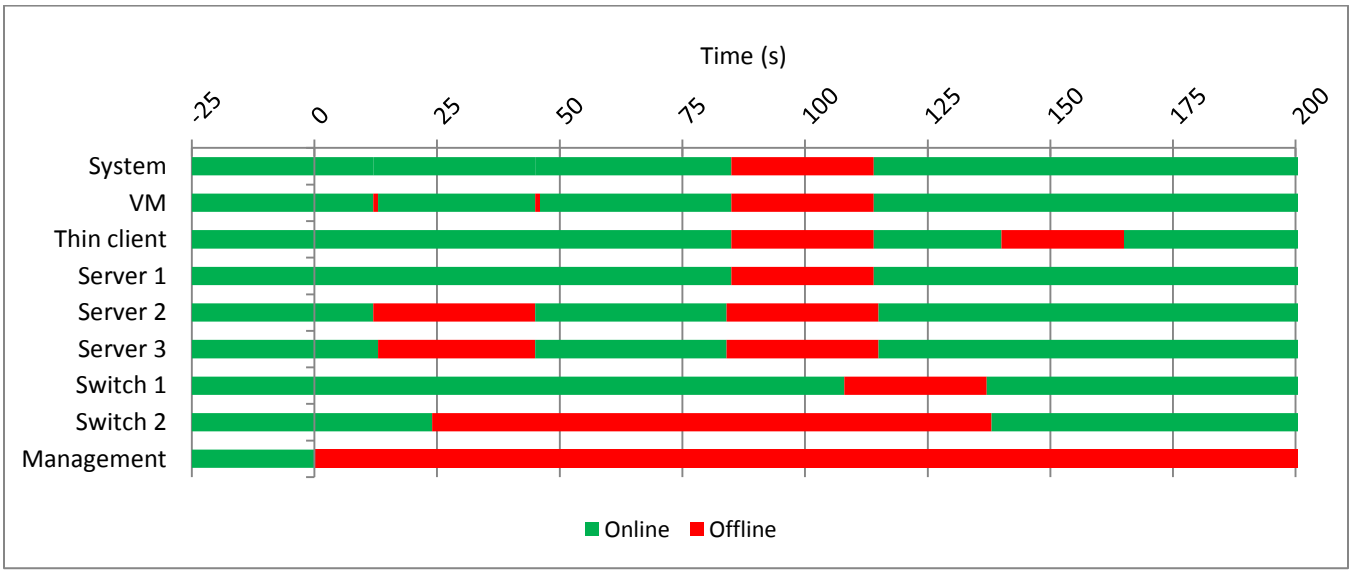


Figure 6-7 Switch reconnect

Switch 2 goes offline at time zero and remains unreachable for 114 seconds. After 12 seconds the VM drops one ping and server 2 and 3 is unreachable from 13 to 44 seconds, a total of 31 seconds. At 45 seconds when server 2 and 3 is reachable, the VM once again drops one ping. After 84 seconds the whole network goes offline for 30 seconds until 114, this is due to STP reconfiguration, which is setup to last 30 seconds by default. Reconfiguration time and other STP values can be changed to allow for a faster reconfiguration, but is kept at the default setting during testing not to cause a source of error. The results from this test show that network failures have a big impact on the system, mainly due to the use of STPs methods for reconfiguration. Instead of adjusting the values of STP, the replacement IEEE 802.1aq standard or equivalent should be adopted. The network equipment used in this test did not support this technology.

6.2.1.7 Blackout test

Exploration rigs often have an erratic power supply. Using a UPS will smooth out a short power problems, but since the equipment is located in an explosive atmosphere, abrupt power cuts are sometimes necessary. The cluster must be able to handle and restart a blackout without any interaction. The test will cut and afterwards reconnect power to all servers and switches simultaneously. Thin clients are not affected by the test, and remains powered on for the duration of the test.

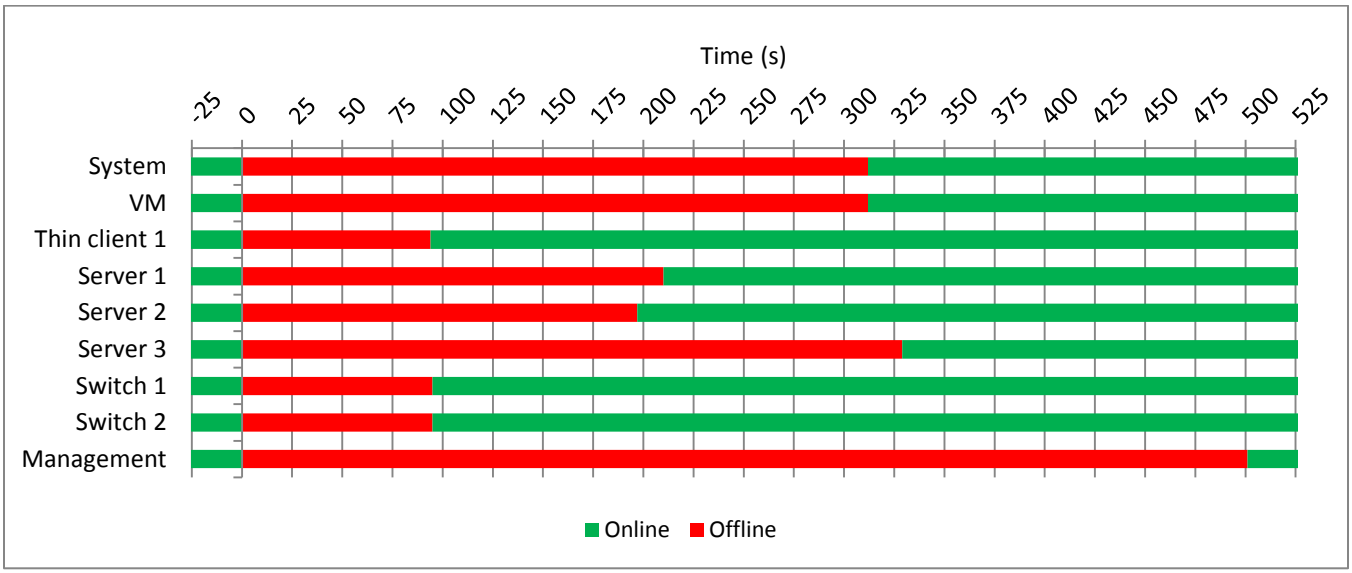


Figure 6-8 Blackout test

The equipment is running fully functionally before the power is removed for 5 seconds. The time of the blackout happens at 0. The two switches are first to regain functionality after 95 seconds, when the management interface comes back online and the thin client replies to ping. Server 1 and server 2 use 210 and 197 seconds, while server 3 use 329 seconds, a difference of 132s from the first to last due to vSAN configuration part in the boot process. After 312 seconds the virtual machine becomes available. The management server used 541 seconds. For an operator checking the system status after a power failure, 541 seconds is a very long wait.

6.2.2 Failure summary

The hardware tests show that vSAN handle failures without causing system stability problems. Host failures do not affect the either active VM or other hosts. When a host with an active VM fails, some downtime occurs, but the system quickly reboots the VM on another host without problems. This functionality still works even if the management server is offline. The cluster handles drive failures and resynchronization of data while active VMs run in the cluster. In the test setup with three nodes, the drives in two hosts can fail without causing downtime.

The system handles a single switch failure without problems, but due to the network equipment used in testing downtime occurs during abrupt power cycles. When a switch comes back online, STP initialization causes the network to go offline for 30 seconds. Without a functioning network, the system is also offline for this period. STP should be replaced with another protocol that does not have as much downtime.

The blackout test the show that the cluster restarts correctly without the need for user interaction. The use of vSAN increases the reliability of the system, but does not give a bumpless failover. Different hardware nodes can fail simultaneously while still maintaining operation. In the case of hardware failures the tested setup is better than the current solution because a VM is moved to another host when a node fails, this does however come with certain inconveniences. The time it takes from power on to a machine is ready for operation is considerably increased. A server running Windows natively use 21 seconds to cold boot into Windows, while a host and VM combined use 312 seconds. This is a significant difference, but the frequency of blackouts is too low to make the vSAN solution unusable.

6.3 Virtual machine

No clearly defined minimum requirements exist for the NOV software. The software ranges from single-threaded with low requirements, to multi-threaded applications requiring a powerful server. To ensure that the current performance is sufficient a wide range of tests are required.

Computational power through CPU has traditionally been the main-focus of servers. Since (virtualized) servers mainly are backend with no visible GUI for the end user, graphics performance has not been prioritized. Most modern GUI OS can use hardware acceleration to improve graphics. Since the introduction of AERO interface in Windows Vista, the GUI have been reliant on graphics to function satisfactory.

A native OS have the processor exclusively available and can decide which program/thread should get the resources. VMs have to share processors between each other. Assigning more virtual cores to a VM will increase performance, as long as physical cores are not over-provisioned. A VMWare virtual machine waits until the required amount of physical cores are ready, before assigning them to a VM. Only one VM in addition to the management server, is active during benchmarking to ensure that other VMs do not interfere with test results. The VM doing benchmarking, run on a different host than the management VM.

6.3.1 Prime95

Prime95 [20] is a tool for generating high processor loads. It runs multiple threads to stress all cores of the CPU. Prime95 operates in three different modes calculating small FFT, in-place FFT, and a mode called blend. Blend mode tests both CPU and RAM, while the others test CPU and CPU cache. The blend mode is a good burn in test, stressing both CPU and RAM. A benchmark mode that runs different FFT and records the time for each calculation is used to gather CPU performance.

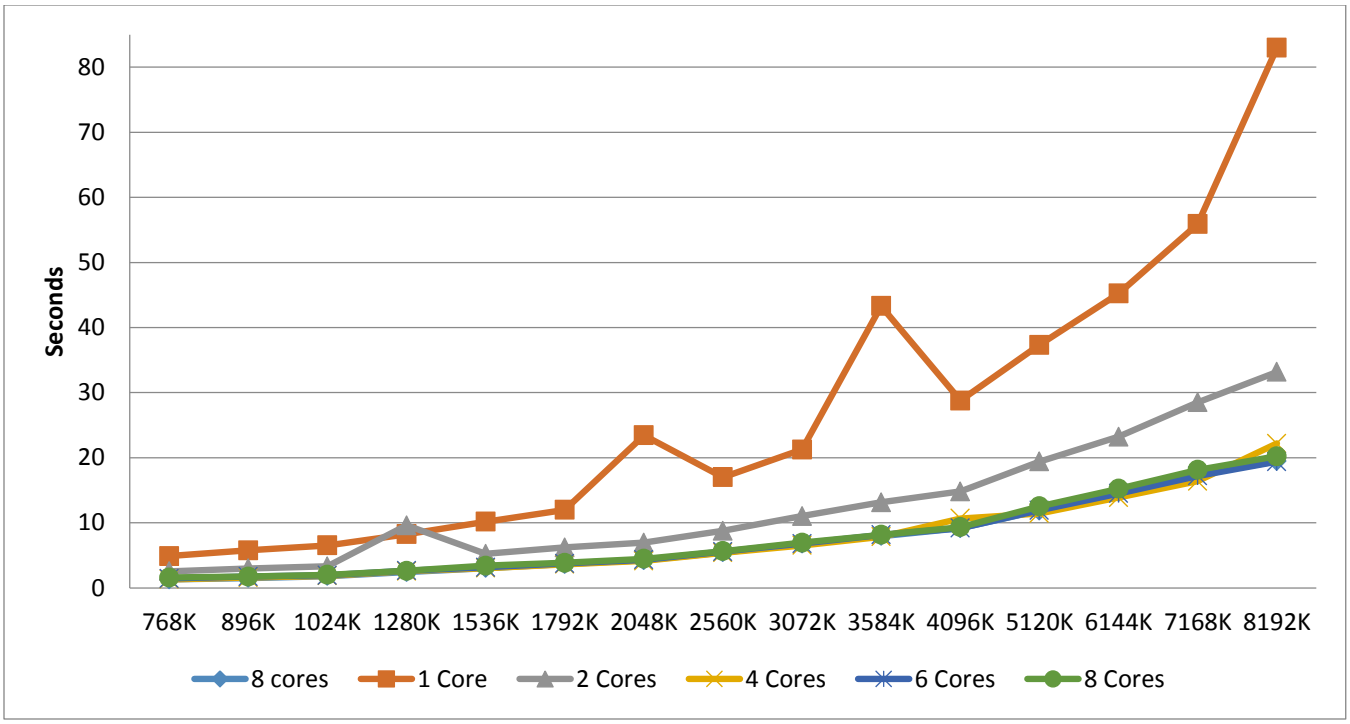


Figure 6-9 Prime95 results

The Intel i7 processor used in the servers, have four physical cores and employ hyper-threading to give the appearance of eight cores. Hyper-threading is a technology that speeds up the “feed rate” into a processor by having two inputs, but since there is only four actual cores the FFT processing won’t be any faster.

Test results show that one and two core configurations clearly separate from four and higher. The difference between 4, 6 and 8 cores virtual and 8 native cores are almost non-existing. For computational performance similar to FFT calculation, more physical cores will increase performance. The CPU used in test servers is aimed at a workstation/desktop use. A server grade CPU with more cores at the same frequency, will most likely increase performance. However server CPUs often have a lower clock frequency and a performance increase is not given. Further testing comparing different processors is required to decide on an optimal solution. The processor choice together with amount of RAM will be deciding factors in establishing how many VMs can run on each physical server.

Since the CPU benchmark results for 4 to 8 cores are so similar, 4 cores will be used in the further tests.

6.3.2 Futuremark PCMark 7

Futuremark offers multiple products to test and measure computer performance. PCMark [21] test a computer's hardware configuration and adds a score from each component into a subtotal score. Some of the tests use DirectX to test graphics performance, a display is required for the tests to succeed. Running the test with RDP and View will reveal any difference in graphics capabilities for the two protocols. A full description of the tests is found in [22].

	Native	RDP	View	RDP/View difference
Pcmark score	4825	2541	2564	-0,5 %
Lightweight score	5355	4005	4047	-0,5 %
Productivity score	5584	4076	4137	-0,7 %
Entertainment score	3528	1502	1496	0,2 %
Creativity score	7491	4063	4043	0,2 %
Computation score	9832	3779	3777	0,0 %
System storage score	5363	3987	4013	-0,3 %
Raw system storage score	5998	1334	1333	0,0 %

Table 6-3 PCMark 7 test score

The two protocols gain a very similar score, and with the 0,7% difference this tests does not give an advantage to either.

6.3.3 Anvil Storage Utility

Anvil is a storage benchmark utility originally developed for SSD drives. Storage performance is often measured in Input/Output Operations per Second (IOPS). IOPS and MB/s is can be connected by the formula

$$IOPS = \left(\frac{MBps\ Throughput}{KB\ per\ IO}\right) * 1024.$$

No standard test parameters exist for IOPS, results from vendors and testers might vary. The test use Anvils default SSD tests of different sizes. Between them they provide both random and sequential I/O operations to disk.

The tests are performed on 4096 byte (4k) sectors since Windows 7 uses this size throughout memory and drives. VMWare VDs use a 1MB to align VD sectors with physical sectors [23]. This avoids the IO penalty that occurs when sectors are not aligned. This penalty grows depending on the amount of IO operations, and is worst for databases. Sequential tests performance from start to finish like a stream, while the other is random IOs. Due to the high seek time in HDDs the random tests yield low scores. An 8 GB test file is tested on a different partition than the OS. Queue Depth (QD) refers to the number of outstanding operations at a certain time. Multiple outstanding operations can increase performance since Native Command Queuing optimizes the order in which the operations are performed [24].

SSD and HDD are benchmarked on a native machine. The vSAN is setup with the SSD as cache and HDD as storage is tested through in a VM. Because the test results for the two types of VD are so similar, the thin provisioning graph have a square bullet point in addition to graph line to increase readability.

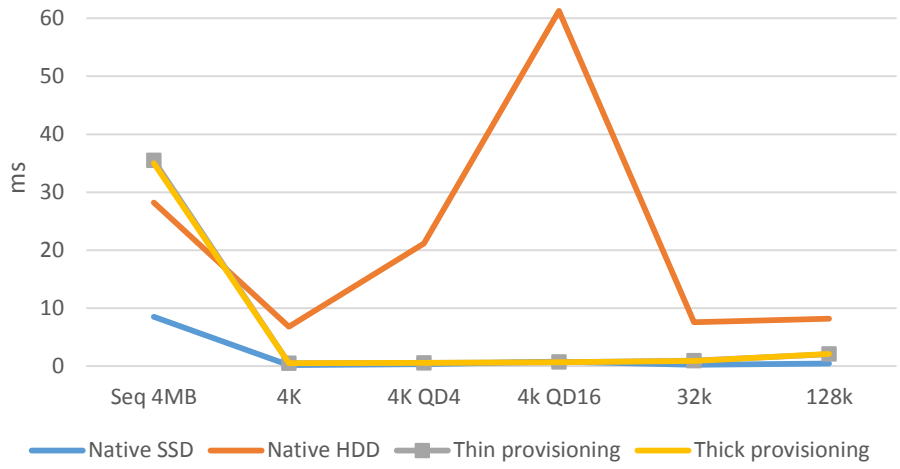


Figure 6-10 Drive read response time

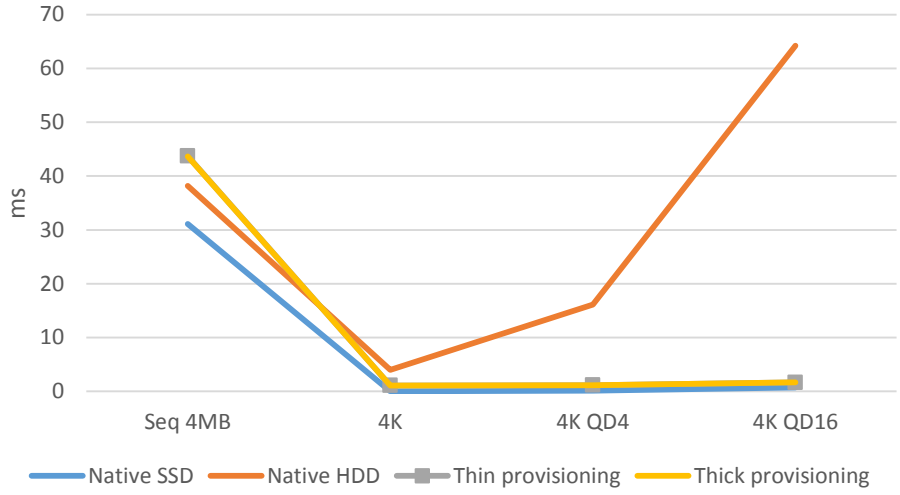


Figure 6-11 Drive write response time

The VDs have a higher sequential 4MB read/write response time than both native HDD and SSD, indicating worse sequential performance. For random 4k access the SSD caching improves the VDs response time compared to a native HDD.

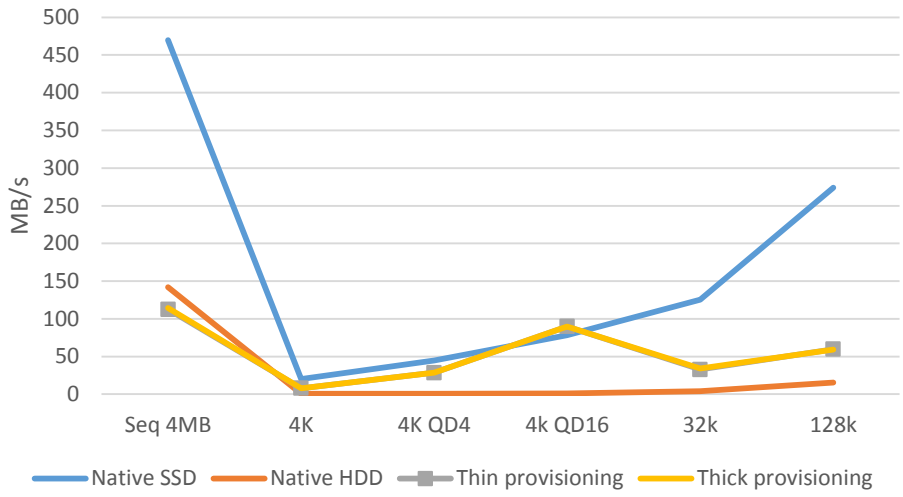


Figure 6-12 MB read per second

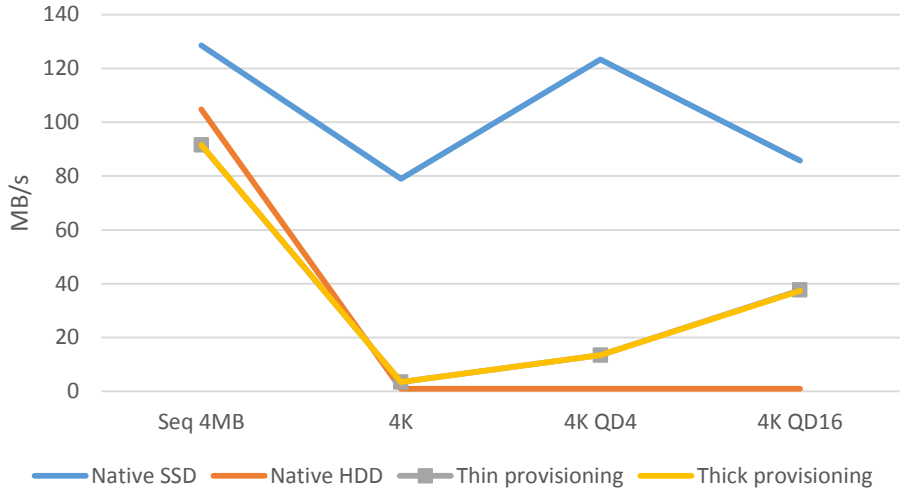


Figure 6-13 MB write per second

Throughput for read and write operations are worse than both the native drives in the sequential test. Sequential is an important measure for one process reading large files.

For storage of VMs random IOs are more valuable because multiple random files are trying to be accessed at once by many processes. The storage benchmark shows that vSAN hybrid storage performs better than a native magnetic HDD, but still is some distance behind a native SSD. Sequential operations have both a higher response time and lower read/write rate, but random access is well above HDD. Drive performance is most likely impeded by vSANs drive setups, where a magnetic drive has to be used for data storage. In deployments where the

The VSANs requirement of using HDDs for storage most likely impedes drive benchmark results. In addition to providing higher performance, SSD drives also tolerate more shock and vibration. In NOV data center deployments, where performance and ruggedness are more important than raw storage capacity, using only flash-based storage would be an advantage.

The largest difference between Thick- and thin-provisioned VDs are 2%, with an average of $0{,}05\%$. Considering these results there is no performance reason to pick thick- over thin-provisioning considering the features thin-provisioning offers.

6.4 Operators view

In a machine control environment, the delay from an event happens to the picture is displayed on screen must be kept at a minimum. The NORSOK standard [25] states that, *"CCTV monitor picture for drill floor equipment shall have time lag less than 250 milliseconds"*. The end-to-end delay of CCTV must be below this level.

Thin client connection time will add to the total delay of the system when a failure occurs. It is also a big part of the user experience. One VM is running in the cluster and each protocol connects to it multiple times, timed by a stopwatch. The results are averaged and displayed in Table 6-4.

Protocol	Connection time (s)
VMWare View	21
Microsoft RPD	16

Table 6-4 Connection time of remote desktop protocols

6.4.1 HMI Application

The Human Machine Interface application is the operator's view of machines and equipment, an application that runs over multiple screens in landscape mode. During operation, it is the only GUI interaction with all software components of the Cyberbase System. A responsive user interface is therefore very important. The application displays multiple different screens to the operator depending on the current ongoing. The user actively changes between the different screens.

The HMI application is developed in Windows Presentation Foundation framework. The graphics acceleration for WPF divides into three tiers [26]. Tier 0 is software acceleration only, for systems running a DirectX version lower than 9.0 or running without DirectX. Tier 1 and 2

runs on version 9.0 and higher. Tier 2 requires at least 120 MB of video ram and, while Tier 1 requires 60 MB. WPF uses dirty rectangle rendering technique, and only redraws changed pixels/rectangles since the previous update. A picture with mostly static elements will therefore require less redrawing than other rendering techniques.

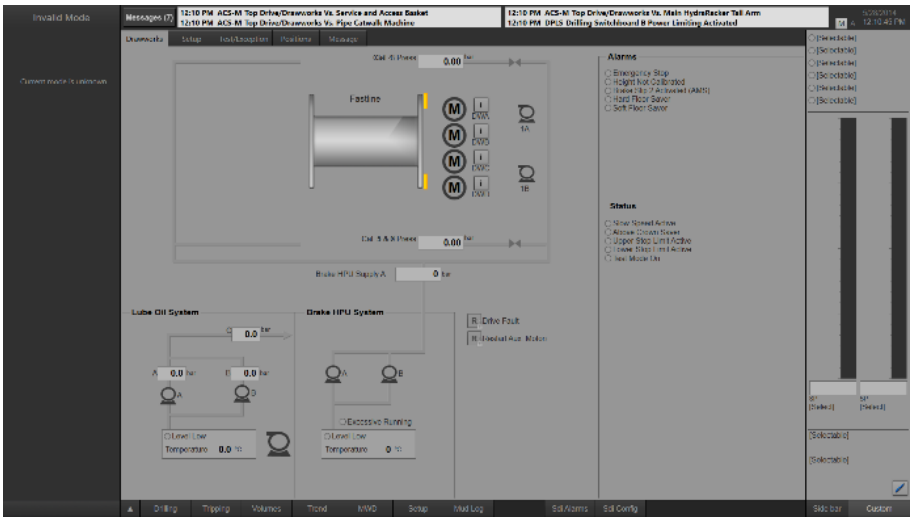


Figure 6-14 HMI application left half of screen

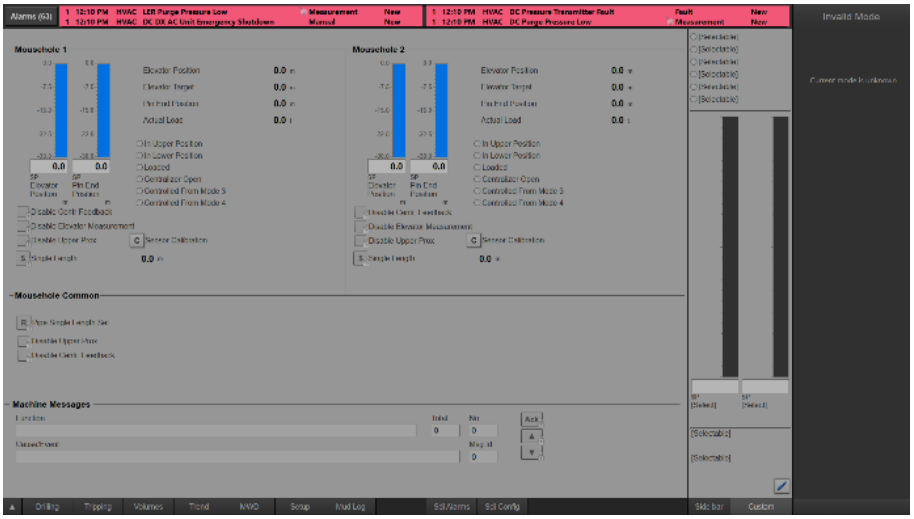


Figure 6-15 HMI application right half of screen

The HMI application displays different screens with a graphical representation of drill floor equipment, and their status. Due to the confidential nature of this application no detailed description of the specific screens will be given.

A built in benchmark measures two performance indicators from the application. The time it takes to initiate a screen, and the second the time it takes to render the contents to the display. The two tests are performed for 63 different screens. Performing the tests on a “cold” computer makes the application load all modules. A longer initialization time the first time displaying a screen compared to the second time is expected. The results reveal the worst-case load time.

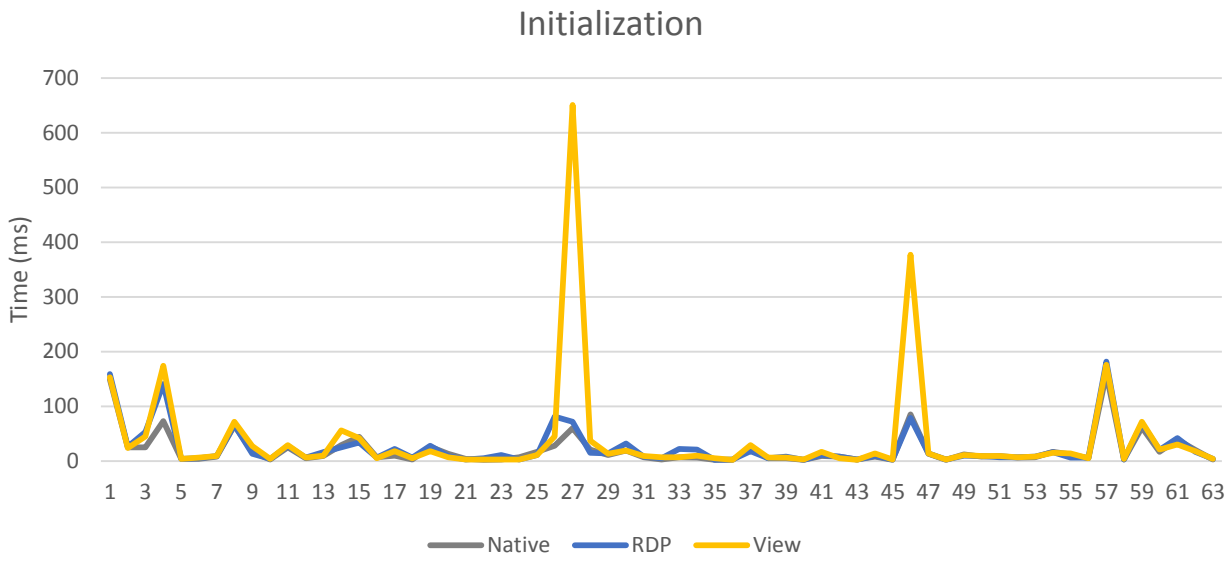


Figure 6-16 Class initialization test

Class initialization	Average time (ms)
Native	20,43
RDP	24,38
View	38,48

Table 6-5 Average initialization time

Figure 6-16 shows the average initialization results for the 63 screens. A native computer on average loads a screen in 20,42ms, while RDP uses 24,38ms s. This is an increase of 3,95ms or 19%. View has an average of 38,48ms, which is 18,08ms or 88% higher than native. In test 27 and 46 View has a significant spike compared to the other protocols, both these screens contain very heavy graphics. RDP clearly performs better than View in the initialization test.
The varied results and large spikes generated by View are not good for a control environment where software operations must be predictable.

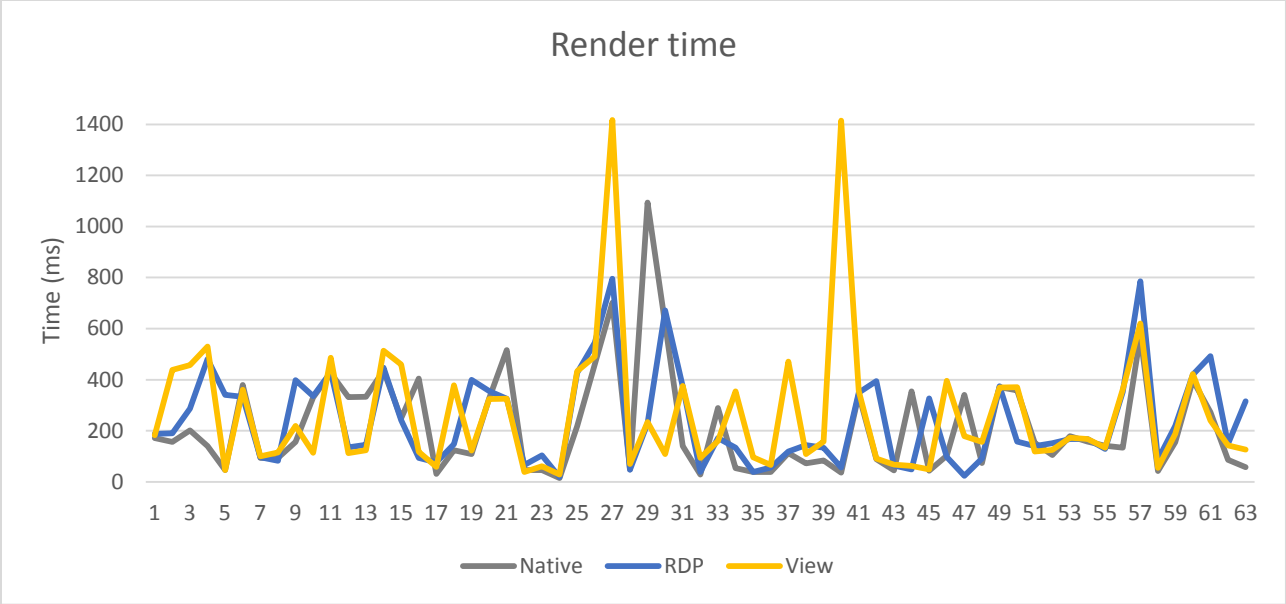


Figure 6-17 Render time

Render time	Average time (ms)
Native	218,46
RDP	241,11
View	265,56

Table 6-6 Average render time

The render time benchmark contains spikes in all of the three options, not giving a clearly visible advantage to anyone. In the total average time RDP and View is 22,65ms or 10%, and 47,10ms or 22% respectively. RDP performs better than View in both tests, and is more consistent without any spikes.

6.4.2 CCTV system and latency

Many latency-sensitive video streaming applications use a CCTV capture device to display analog video on a computer. One device is needed for every machine that displays video. To use such a device in a VM, it would have to be setup in a pass-through configuration. Pass-through makes a VM tied to a specific server, and does not allow migration.

Making the capture device and the physical machine less tightly coupled, allow both streaming and migration. The CCTV setup in this test uses a converter that allow for video streaming over Ethernet. Video from CCTV displayed in the HMI application have a typical frame rate of 10 FPS [27], but value is expected to increase and the protocol must handle minimum 20 FPS.

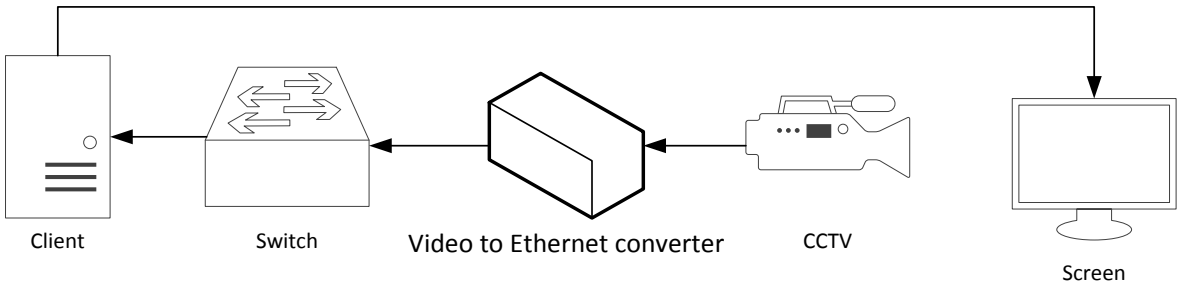


Figure 6-18 Camera latency determination setup

To establish the maximum latency a RD protocol can introduce, CCTV latency has to be determined. By streaming video of a video stream and a stopwatch, in a visual loop, the latency can be determined by comparing the time in each consecutive picture.

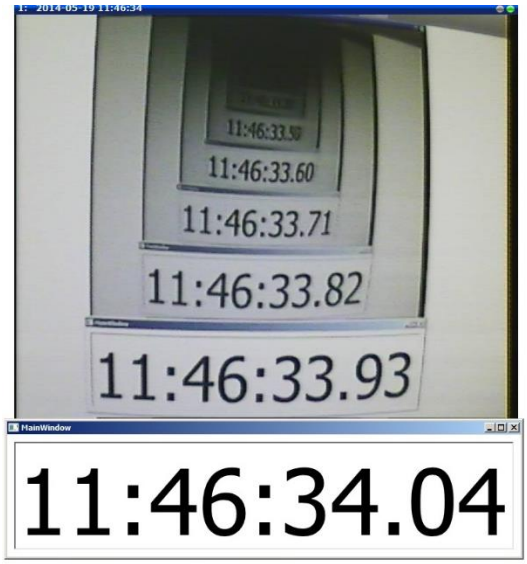


Figure 6-19 Camera latency testing

Connecting the camera to the computer through the Ethernet converter gave an average latency off $110ms$ as shown in Figure 6-19 Camera latency testing Figure 6-19. A RD protocol can have a maximum latency of 140ms, to comply with the maximum 250 milliseconds requirement.

6.4.3 WPF Benchmark

Since many remote desktop protocols only render elements that have changed since last update, latency will most likely change depending on activity. A scenario where a full screen update is necessary for every frame will put the heaviest load on the system. To simulate this worst-case scenario multiple randomly moving particles run when performing latency tests. The application run a user selectable amount of particles, increasing the amount of updates required for each step. In testing 0 to 10 000 particles with an interval of 1000 have been used. The tool Perforator, from WPF performance suite [28], is used to capture the frame rate of the client application. This tool is minimized in the lower screen corner to have as little impact on the moving particles as possible.

6.4.4 Camera video visual loop test

This test expands the latency test from 6.4.1 to stream video from multiple screens, running different RD protocols. A computer connects to a server through remote desktop and one have the OS running natively. All computers display the same video stream. It is important that no delay difference occur in the stream from the camera to the computers. Multicasting video to both computers should decrease the difference in transmission delay to a neglectable level. The camera streams video of the stopwatch to all computers simultaneously. Taking a one still picture of the stopwatch and all the computers will reveal the difference between real time, native and virtualized. The accuracy is dependent on how many frames per second the camera is capable of capturing. The camera used in testing is 25 FPS, giving a sampling interval of $\frac{1000ms}{25\ FPS} = 40\ ms$. This interval is possibly too large to determine the latency of each display protocol accurately.

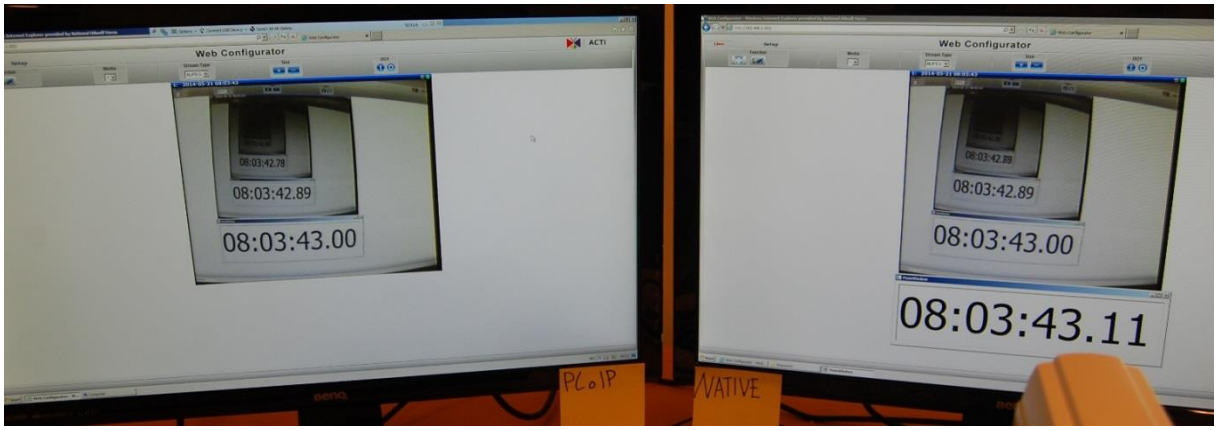


Figure 6-20 from left: PCoIP, Native

With small display changes, View and RDP can barely be seperated due to the accuracy of the camera. During a series of photographies, very few show a drifference.

No particles

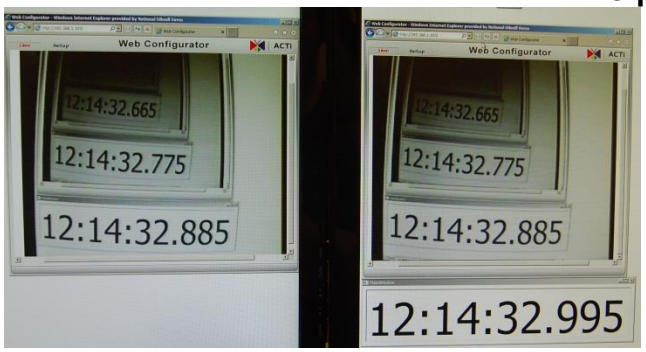


Figure 6-21 VMWare and native source

Figure 6-22 Microsoft RDP and native source

Figure 6-22 show a $110ms$ difference between protocol and native. Display of the native image contain small artifacts around the millisecond numbers, most likely caused by the camera. This indicates that the numbers have recently changed or that the camera capture process is too slow. When adding moving dots and forcing the protocols to rerender multiple parts of the screen the latency increases. 1000 moving dots make a few more still pictures show that the remote protocols are one frame behind the native, but the camera accuracy is still a problem and the framerate is too low to determine protocol latency.

1000 Moving dots

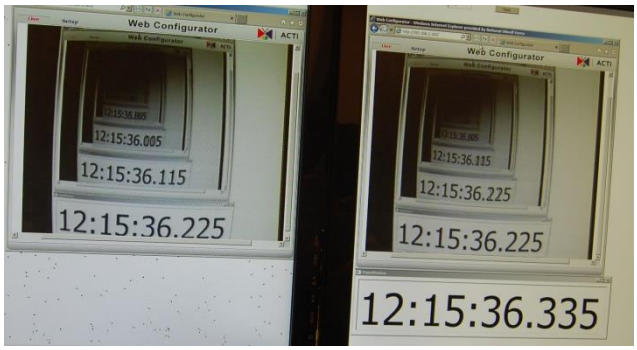


Figure 6-23 VMWare View and native source

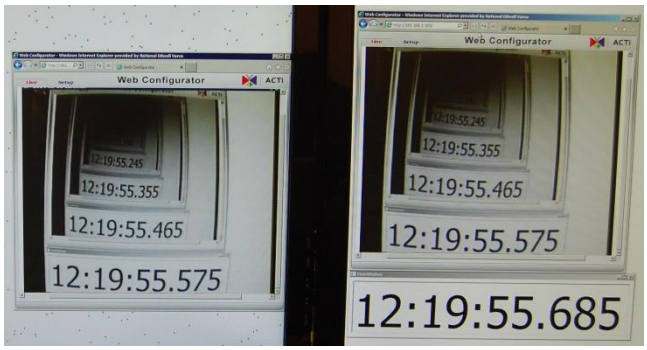


Figure 6-24 Microsoft RDP and native source

Increasing the number of particles to 10 000 makes a drastical impact on VMWare View. Latency is around 440 ms. RDP still have the same latency as previously, between 0 and 110 ms.

10 000 Moving dots

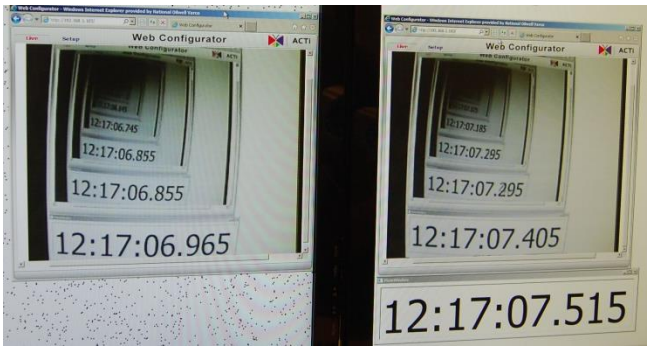


Figure 6-25 VMWare View and native source

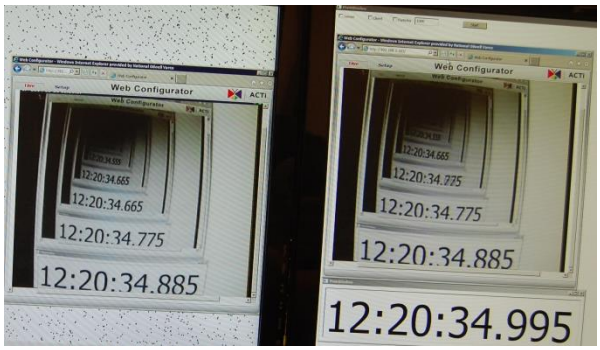


Figure 6-26 Microsoft RDP and native source

The video loop test proved not to be accurate enough to clearly determine the latency of the two protocols, caused by the camera setup used for testing. The test does show that VMWare View is slower than Microsoft RDP when there are many changes on the screen.

6.4.5 Broadcast test

The broadcast test determines remote desktop protocol latency by broadcasting UDP packets, with timestamps, to multiple clients running the different remote desktop protocols. The clients receive a packet and display the packet number and timestamp. Taking a still picture of the two screens gives the packet number and timestamp of a protocol compared to a native client.

Comparing timestamps for a protocol with the timestamp on the native client gives remote protocol latency. This test also use the WPF benchmark described in 6.4.3, to simulate a display changes and force a full screen redraw.

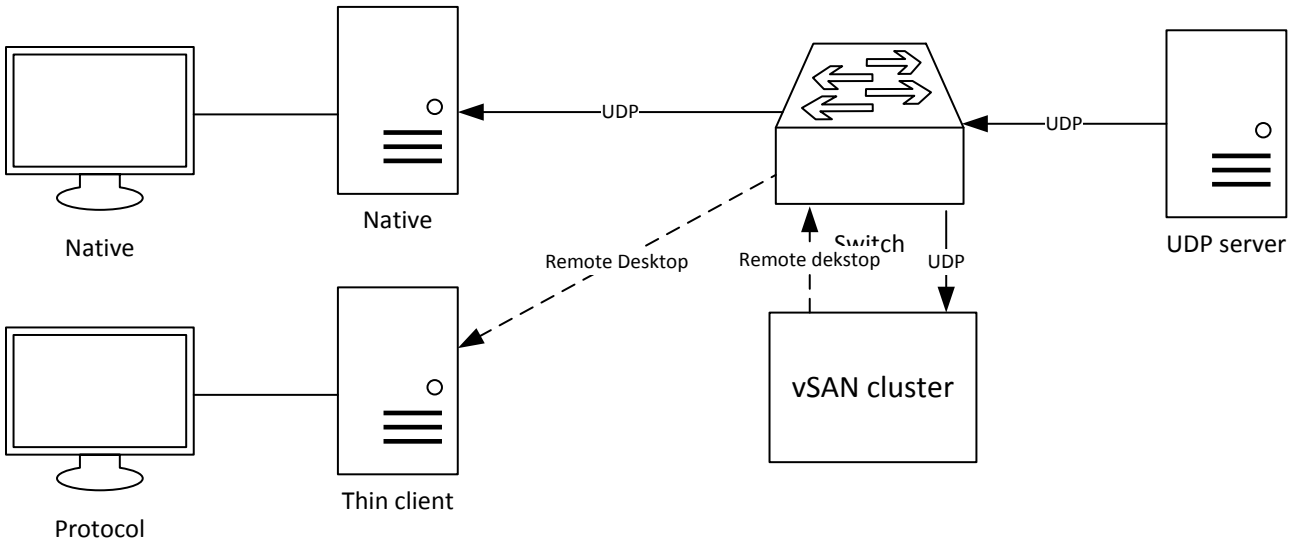


Figure 6-27 Broadcast test

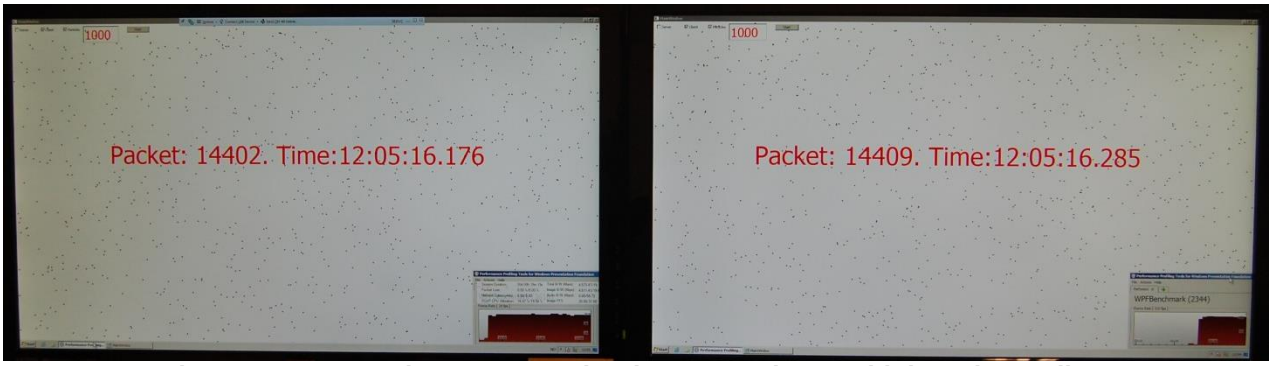
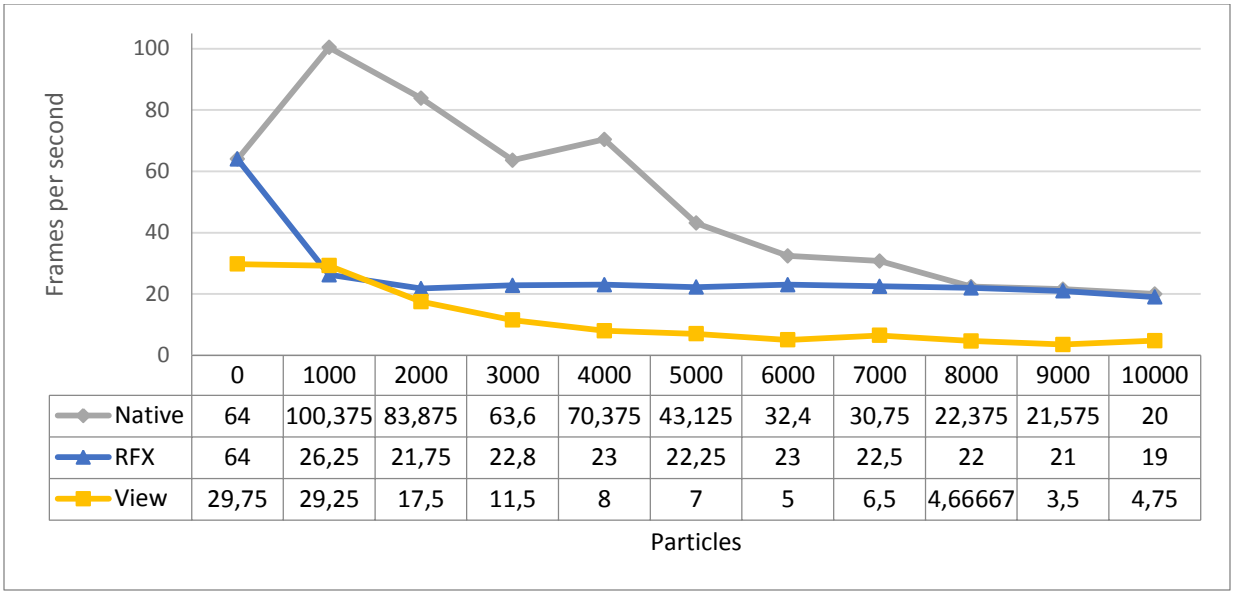


Figure 6-28 Measuring VMWare View latency and FPS with broadcast client

Before testing protocols against native, the two clients are checked against eachother to ensure that there is no difference between them when both run native mode.



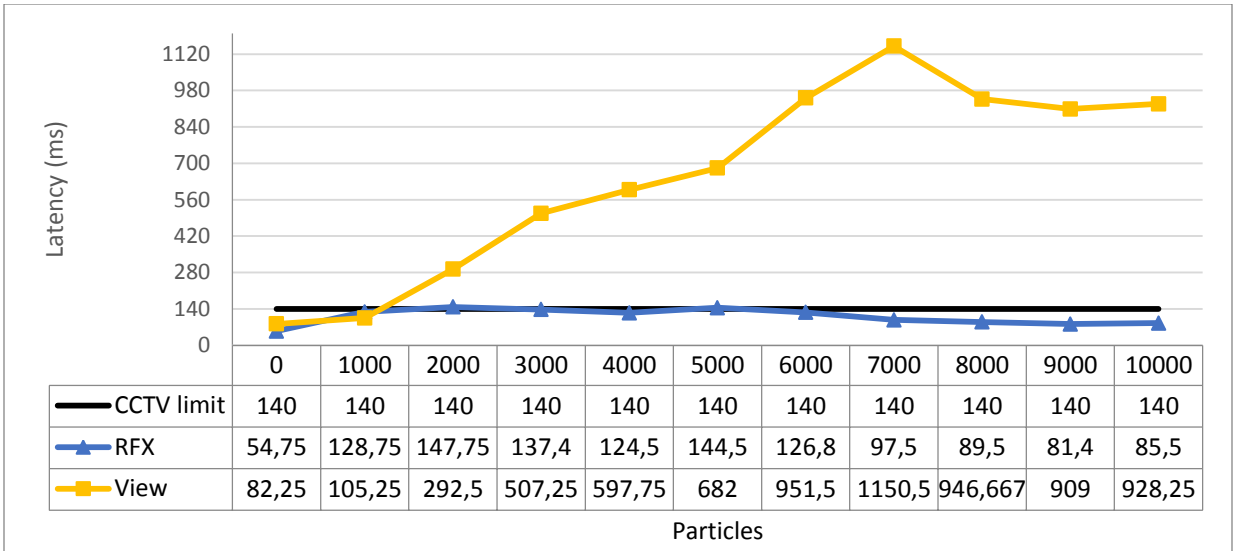
	0	1000	2000	3000	4000	5000	6000	7000	8000	9000	10000
Native	64	100,375	83,875	63,6	70,375	43,125	32,4	30,75	22,375	21,575	20
RFX	64	26,25	21,75	22,8	23	22,25	23	22,5	22	21	19
View	29,75	29,25	17,5	11,5	8	7	5	6,5	4,66667	3,5	4,75

Figure 6-29 Remote desktop frames per second comparison

When there is no movement in the picture, native has a frame rate of 64 before increasing to 100 FPS at 1000 particles. This is because WPF only redraw display changes, and when there are no particles in movement, 64 FPS is enough to update the display with the information from the incoming broadcast packets. When particles are moving, the native client draws updates at fast as possible resulting in a higher FPS until the hardware no longer can keep up the amount of particles.

RDP start at 64 FPS but drops to 26,25 at 1000 particles. From 2000 to 8000 particles, RDP stays relatively stable at around 23 FPS, before dropping to 21 at 9000 and 19 at 10 000 particles. Apart from a small jump at 4000 particles, the native client steadily decreases from 100,37 FPS at 1000 particles, to 22,37 at 8000. For the last three number of particles native and RDP are almost identical. RDP stays well above the CCTV requirement of 10 FPS, and can satisfy higher FPS from future versions.

VMWare View start with a frame rate of 26,75 and is stable from to 1000 particles, where it delivers 3 FPS higher than RDP. After from 2000 to 10 000 particles the FPS number drops from 17,5 to 3,5 at 9000 particles. The results from 4000 throughout 10 000 have a frame rate lower than the 10 FPS required by CCTV. This makes View unusable.



	0	1000	2000	3000	4000	5000	6000	7000	8000	9000	10000
CCTV limit	140	140	140	140	140	140	140	140	140	140	140
RFX	54,75	128,75	147,75	137,4	124,5	144,5	126,8	97,5	89,5	81,4	85,5
View	82,25	105,25	292,5	507,25	597,75	682	951,5	1150,5	946,667	909	928,25

Figure 6-30 Remote desktop latency comparison

Figure 6-30 show the measured latency for 0-10 000 particles. The maximum allowed latency of 140ms found in CCTV system and latency6.4.2 is plotted in the graph. RDP and View start at 54,75ms and 82,25ms respectively. At 1000 particles RDP increases to 128,76ms while View has a slower increase to 105,25ms. From 2000 particles and onward the two protocols separate. The measured latency for View increases significantly and reaches a top of 1150,50ms at 7000 particles. This is over eight times higher than the maximum required latency. RDPs highest value is 147,5ms at 2000 particles. RDP is stable around the CCTV limit from 2000 to 6000 before decreasing from 7000-10 000. Both protocols decrease latency when the three highest amount of particles are displayed. This is connected with the frame rate, which decreases in the same interval.

The results from FPS and latency test show that VMWare View is far from satisfying the CCTV performance requirements. Although the performance with a low number of particles is adequate, high latency and low FPS make the protocol unusable when the number of particles increase. RDP maintains a good FPS through the test. The latency is for the majority of the test below the 140ms limit, aside from the value at 2000 particles. Considering that this test were performed in an ideal environment without any other applications running or ongoing network activity, the RDP latency is too close to the maximum limit and will most likely exceed the limit when applied in a real system. If the CCTV system latency is lowered, more headroom

will be given to RD protocols and the total system latency with RDP can be lower than the 250ms requirement.

7 CONCLUSION

With the high costs of operating an exploration rig, single point of failures should be eliminated from operation critical components. By using virtualization, hardware failures can dealt with and give a longer window for repair while still having a fully functional system. Through the course of this thesis, we have studied methods for increasing the availability of the servers used in the NOV Cyberbase System.

Servers were converted from physical to virtual machines and run in a VMWare vSAN cluster. This cluster provides a fault tolerant data store, but requires a VM to restart in the event of a host failure. The goal of a bumpless failover is not achieved, but less downtime and higher dependability is. The cluster provides a dynamic expansion ability by adding more hosts for storage capacity, computational power or both. Testing showed that hardware failures are handled, and that single components can fail without affecting the stability of the system. Benchmarking revealed that VM performance was sufficient compared to a native host.

Two remote desktop protocols, Microsoft RDP and VMWare View, were tested against each other to determine performance and latency. NOV HMI application benchmark revealed that RDP had better and more stable results than View. The results from CCTV testing showed that RDP were on the limit of complying with the maximum latency, while maintaining a usable frame rate. View had latency over four times the CCTV requirement and a frame rate much lower than RDP. RDP had frame rate compliant with the requirements for current and future CCTV versions, but the latency is slightly too high for use.

7.1 Further work

Some more time should be spent on reducing downtime related to network failures. Especially a replacement for STP. Network virtualization is a topic that can be investigated.

A full test with all of NOVs software products have to be conducted. This test will reveal if software can handle failures. Software has to be benchmarked to find an optimal resource allocation for each VM. Developing and testing a redundant client setup with central management.

Bibliography

[1] Offshore Media Group, "Offshore.no," Offshore Media Group, 04 11 2013. [Online]. Available: http://www.offshore.no/Prosjekter/riggdata.aspx. [Accessed 04 11 2013].

[2] S. K. Moore and . Y.-T. Chiu, "IEEE Spectrum," IEEE, 1 February 2003. [Online]. Available: http://spectrum.ieee.org/computing/hardware/leaking-capacitors-muck-up-motherboards. [Accessed 13 February 2014].

[3] B. Schroeder and A. G. Gibson, "Disk failures in the real world: What does an MTTF of 1,000,000 hours mean to you?," USENIX, San Jose, CA, 2007.

[4] Microsoft Corporation, "Microsoft White paper," Microsoft, 2013. [Online]. Available: http://download.microsoft.com/download/0/2/1/021BE527-A882-41E6-A83B-8072BF58721E/Windows_Server_2012_R2_Overview_White_Paper.pdf. [Accessed 07 05 2014].

[5] Transcend, "TS512GSSD720 2.5” SATA Solid State Disk Data Sheet," 2013.

[6] HGST, "Travelstar® 7K10002.5-Inch Mobile 7200 RPM 9.5mm Hard Disk Drives," [Online]. Available: http://www.hgst.com/tech/techlib.nsf/techdocs/FF05B02FBBBF9E8288257AAF00686AD6/$file/TS7K1000_ds.pdf. [Accessed 03 2014].

[7] Intel Corporation, "4th Generation Intel Core vPro Processor Family Overview White Paper," [Online]. Available: http://www.intel.com/content/dam/www/public/us/en/documents/white-papers/high-performance-computing-4th-gen-core-vpro-overview-paper.pdf. [Accessed 2 2014].

[8] Intel, Hewlett Packard, NEC, Dell, "Intelligent Platform Management Interface Specification," 1 10 2013. [Online]. Available: http://www.intel.com/content/dam/www/public/us/en/documents/product-briefs/ipmi-second-gen-interface-spec-v2-rev1-1.pdf. [Accessed 02 2014].

[9] K. A. Somani and H. N. Vaidya, "Understanding Fault Tolerance and Reliability," IEEE, 1997, pp. 45-50.

[10] Apache Software Foundation, The, "Apache ZooKeeper," 04 11 2013. [Online]. Available: http://zookeeper.apache.org/. [Accessed 28 11 2013].

[11] T. Richardson, "RealVNC," 26 11 2010. [Online]. Available: http://www.realvnc.com/docs/rfbproto.pdf. [Accessed 05 2014].

[12] VMWare, "www.vmware.com," 2006. [Online]. Available: http://www.vmware.com/pdf/vi_architecture_wp.pdf. [Accessed 27 11 2013].

[13] VMware, Inc, "What’s New in VMware vSAN," 2013 10 03. [Online]. Available: http://www.vmware.com/files/pdf/products/vsan/VMware_Virtual_SAN_Whats_New.pdf. [Accessed 2013 11 04].

[14] Citrix, "www.citrix.com," 06 2013. [Online]. Available: https://www.citrix.com/content/dam/citrix/en_us/documents/products-solutions/citrix-xenserver-industry-leading-open-source-platform-for-cost-effective-cloud-server-and-desktop-virtualization.pdf?accessmode=direct. [Accessed 27 11 2013].

[15] Citrix Systems, "www.citrix.com," 2009. [Online]. Available: http://support.citrix.com/servlet/KbServlet/download/21018-102-

664364/High%20Availability%20for%20Citrix%20XenServer.pdf. [Accessed 06 05 2014].

[16] D. Kusnetzky, Virtualization: A Manager's Guide, Sebastopol: O'Reilly, 2011.

[17] K. Schmidt, High Availability and Disaster Recovery. Concepts, Design, Implementation, Berlin: Springer, 2006.

[18] VMWare inc., "VMware Virtual SAN Design and Sizing Guide," 03 2014. [Online]. Available: https://www.vmware.com/files/pdf/products/vsan/VSAN_Design_and_Sizing_Guide.pdf. [Accessed 04 2014].

[19] E. Fulkerson, "tcping.exe - ping over a tcp connection," [Online]. Available: http://www.elifulkerson.com/projects/tcping.php. [Accessed 25 04 2014].

[20] Mersenne Research, Inc., "http://www.mersenne.org/," [Online]. Available: http://www.mersenne.org/. [Accessed 29 04 2014].

[21] Futuremark, "Futuremark PCMark 7," [Online]. Available: http://www.futuremark.com/benchmarks/pcmark7. [Accessed 30 04 2014].

[22] Futuremark, "Futuremark PCMark 7," [Online]. Available: http://s3.amazonaws.com/download-aws.futuremark.com/PCMark_7_Whitepaper.pdf. [Accessed 06 2014].

[23] VMWare inc., "VMWare vSphere Performance Best Practises," [Online]. Available: http://www.vmware.com/pdf/Perf_Best_Practices_vSphere5.5.pdf. [Accessed 05 2014].

[24] Intel Corporation, Seagate Technology, "Serial ATA Native Command Queuing," 06 2003. [Online]. Available: http://www.seagate.com/docs/pdf/whitepaper/D2c_tech_paper_intc-stx_sata_ncq.pdf. [Accessed 05 2014].

[25] Norwegian Oil and Gas Association, *NORSOK D-001,* Standard Online AS, 2012.

[26] Microsoft Corporation, "Graphics Rendering Tiers," [Online]. Available: http://msdn.microsoft.com/en-us/ms742196.aspx. [Accessed 05 2014].

[27] S. Svendsen, Interviewee, *Product Owner CCTV, Instrumentation & Optimalization, National Oilwell Varco.* [Interview]. 06 05 2014.

[28] Microsoft inc., "WPF Performance Suite," [Online]. Available: http://msdn.microsoft.com/en-us/library/aa969767(v=vs.110).aspx. [Accessed 05 2014].

[29] K. M. Greenan, S. P. James and J. W. Jay, "Mean time to meaningless: MTTDL, Markov models, and storage system reliability," in *USENIX*, Boston, 2010.

8 APPENDIX

8.1 Software versions

Name	Version
Microsoft Windows 7	Ultimate
Microsoft Remote Desktop Protocol	8
VMWare ESXi	5.5.0
VMWare Tools	9.4.5
VMWare Horizon View Client	2.3.3
VMWare View Agent	5.3.1
VMWare View Agent Direct Connection	5.3.0