



Mälardalen University
School of Innovation Design and Engineering
Västerås, Sweden

Thesis for the Degree of Bachelor of Science in Engineering -
Computer Network Engineering
15.0 credits

CAPACITY PLANNING AND SERVER DESIGN FOR A WEB SERVICE

Felipe Retamales
fra13001@student.mdh.se

Examiner: Mats Björkman
Mälardalen University, Västerås, Sweden

Supervisor: Wasif Afzal
Mälardalen University, Västerås, Sweden

Company supervisor: Jan Lundqvist,
Devo IT, Uppsala

May 25, 2016

Abstract

Devo IT and its subsidiary SoftRobot AB are planning to offer a new service to its clients to further enhance growth of the company. This new service is a website that allows clients to upload documents that are converted into machine-editable text. The website and its underlying database are developed by SoftRobot's developer, but they lack the hardware. Using Cisco's PPDIIO network life cycle model, the three first stages of the project are identified. The "prepare" stage is already done by Devo IT where the project's justifications are discussed. In the subsequent "planning" stage, the needs of the project, are identified and a gap analysis is made of what is needed. The "design" phase elaborates which specific hardware and software is needed for the project.

Three servers are needed, one for the main web server, a second for the database, and a third as a cache server for relieving the load on the database. These are planned as virtual machines, so that they can be located on the same physical machine and easily moved if necessary. The disk space required for the database is calculated with test documents since the average document size, the number of clients and how much they upload is known. Disks with adequate size can therefore be chosen. Different ways of improving performance and lowering failure rate of the disks are discussed with the means of RAID levels. These levels improve both disks reliability and performance in different ways. RAID 10 is designated for the database and RAID 1 for the web server and the caching server since those levels are the most suited for these applications. CPU and memory requirements are chosen based on availability and cost. Network bandwidth is analyzed and confirmed as sufficient with the help of the test database and since the bottleneck would be the CPU converting the uploaded documents. Software for backup and administration of the virtual machines are chosen comparing their functionality and the requirements for the project and their cost. After the hardware needs are identified, an cost analysis is made between hosting in Devo IT's server room or outsourcing it to an external company. It was revealed, as Devo IT suspected, that outsourcing costs more.

The results of this thesis enables Devo IT to create a service with good quality, which will meet the clients expectations and also make Devo IT grow as a company with new clients and increased revenue.

Contents

List of Figures	I
List of Tables	II
Glossary	III
Acronyms	IV
Acknowledgement	V
1 Introduction and background	1
2 Problem formulation	4
3 Related work	5
4 Method	7
4.1 Cisco's PPDIO model	7
4.2 Needs assessment and gap analysis	9
4.2.1 Pre-conditions	9
4.2.2 Desired outcome	10
4.2.3 Gap analysis and method	10
5 Design	12
5.1 Power	12
5.2 CPU	12
5.3 Climate control in Devo IT's server room	13
5.4 Disks	13
5.4.1 RAID	14
5.4.2 Throughput calculation	15
5.4.3 Space	18
5.5 Network	19
5.5.1 Bandwidth	19
5.6 Hypervisor	19
5.6.1 Administration	21
5.7 SQL Server	22
5.8 Web server (software)	22
5.8.1 Reverse caching proxy	23
5.9 Security	23
5.9.1 Firewall	23

5.9.2	SSL certificate	24
5.10	Backup	24
5.11	Surveillance	26
5.12	Inhouse vs outsourced hosting	27
6	Discussion & Future work	28
7	Conclusion	30
	References	31
	URL References	33

List of Figures

1	Depiction of gap analysis	9
2	RAID layout for VMs	17
3	Predicted size of the SQL-database	18
4	Layout of servers in a physical machine	21
5	SSL encryption	24

List of Tables

1	RAID level comparison	17
2	Price of baremetal hypervisors (2016)	20
3	Price of backup software (2016)	26
4	Cost calculation for 1 server and 1 Windows 2012 R2 license .	27

Glossary

Application Programming Interface

A set of libraries, protocols or tools that helps developers write code that interfaces with other software

Failover

When a service uses two or more servers simultaneously and still functions if one fails, the servers have failover. Alternatively, if server A is in standby and server B is handling the service and server A fails and server B seamlessly takes control of the service, it also has failover.

Man-in-the-middle Attack

Often abbreviated MITM or MIM, is when a hacker sits between two parties and impersonates both, in an effort to gain access to data.

Network Attached Storage

A small computer that connects to the network and provides centralized storage for users. It can be customized with hard drives of different sizes and models.

Optical Character Recognition

A program that scans documents and converts the handwritten or printed text into characters that can be used in programs such as Microsoft Word

Queue Depth

Queue depth is how many operations a hard drive has in its queue. For example, if there is 1 read and 3 write operations in the queue, the queue depth is 4.

Service Level Agreement

A contract between a service provider and a client defining for example service quality and expectations.

Virtual Machine

A Virtual Machine is a physical computer that is installed on a hypervisor that emulates its hardware. The main advantage is that many virtual machines can be installed and run simultaneously on the same physical computer.

Acronyms

API Application Programming Interface

CPU Central Processing Unit

EOL End Of Life

HDD Hard Disk Drive

IOPS Input/Output Per Second

IPMI Intelligent Platform Management Interface

KVM Kernel-based Virtual Machine

NAS Network Attached Storage

OCR Optical Character Recognition

OS Operating System

QD Queue Depth

RAID Redundant Array of Independent Disks

RAM Random Access Memory

SQL Structured Query Language

SSD Solid-State Drive

SSL Secure Sockets Layer

TLS Transport Layer Security

UPS Uninterruptible Power Supply

VM Virtual Machine

Acknowledgement

First and foremost, I would like to express my gratitude to my supervisor, Dr. Wasif Afzal, for supporting and helping me these past 10 weeks. His advice was essential to this thesis and has taught me many lessons and insights on the workings of academic research.

I also like to thank Mr. Jan Lundqvist at Devo IT for giving me the opportunity and providing me with the equipment to make this thesis a reality.

I would also like to thank my parents, Denise and Jorge and my sister Barbara and her family for always pushing me in my studies and making me learn the values of science and knowledge.

I would especially like to thank my girlfriend Cecilia. Without her unconditional support and encouragement I would surely never have completed this thesis and my degree.

Last but not least, my two cats, Findus and Frasse, for always waking me up early in the morning reminding me to go and study! (or maybe they just wanted food).

1 Introduction and background

Businesses handle a lot of documents. Whether the company is an small one-man business or a giant corporation, documents are part of everyday work. Documents are often physically filed in an archive that keeps growing and growing. Trying to search for a document that was filed many years ago, by an employee that no longer works there, can be a time-consuming labor. A worse scenario is that the papers are destroyed in a flood or a fire, and that can be catastrophic for a business.

Therefore, it is a good idea to digitize and create a more efficient workflow and to make offices a step closer to the “paperless office”.

The space that the documents are stored in can therefore, after the documents are digitized, be used in a more efficient way or if they are leased, the storehouse contract can be canceled and the money spent elsewhere.

Digitizing documents take a lot of time and effort. Often, the documents have to be processed by an Optical Character Recognition (OCR) software. This requires some skill and can be prone to errors due to the software not being able to correctly scan the documents. Thus, it is a good choice to outsource this operation.

Instead of letting the clients mail in the physical papers to the company that digitize their documents which can take days, a better option is to let the clients scan the documents and upload them to a website to be processed automatically. This way the clients can get their documents and data almost immediately and it also creates an efficient workflow for both the clients who upload the documents and the company that handle the digitized documents. Through a website, the clients can get a nice overview and also access all their documents from one single place.

Any company website that handles its clients should be fast, reliable and always available. Consequently, this creates some requirements on both the website’s underlying software and the hardware that is used. They should be reliable, secure, capable of handling all clients without choking, and deliver this around the clock for every day of the year. In the event of an extreme hardware failure, the website should also still function with the means of redundant hardware and transparent failover. Carefully chosen hardware, software and configuration is therefore important to fulfill these requirements.

1 INTRODUCTION AND BACKGROUND

Devo IT is a company located in Uppsala, Sweden. Their main business is to digitize documents so that they can be easily accessed from a computer. Documents that are digitized include invoices, medical records, plans, and diapositives among others. SoftRobot AB is a subsidiary to Devo IT and their focus is to create software that automate various financial processes. In the second quarter of 2016, Devo IT plans to launch a new service/website. One functionality of this site is to allow users to upload documents to SoftRobot AB's servers.

This website is crucial for the company because it helps them retain their current customers and present a progressive and positive outlook of the company, and also attract new clients and enable the company to grow.

The motivation for this project is to contribute in the successful implementation of this web-based service and to ensure that the operational web service meet user expectations. From the user perspective, this means that the service should always be available and also be responsive enough to not degrade the user's experience. For the company the implementation should not cost too much, be capable of scaling to some degree, i.e., be capable of high loads without slowing down the user's experience too much and be easily maintained and supervised.

The website needs certain hardware to be able to handle the load of initially 200 clients. This means that the hardware must be planned in a fashion so that it is adequate but also does not have extreme overcapacity. The website is also connected to a backend database that handles among other things the documents and client information. Hence, it is also important to properly dimension the hardware needs for the database.

Devo IT's clients are predicted to grow significantly over the coming years, so it is also important that the hardware can scale to some degree. For instance, the hard drives that the database are on should not only be fast enough to deliver data to the web server, they should also be prepared for the significant growth of the company, meaning additional future clients and growth of the database. It is also of essence to choose a good enough Central Processing Unit (CPU) for the web server so that it can handle the load from the clients and have capacity to grow. Bear in mind that choosing a CPU with extreme performance is unnecessary at the beginning, since it probably will not be needed until later in the potential scale up of the website capacity. It is also a waste of money to have a CPU that barely has anything to do. All of this, of course, need to be decided by keeping in view the company's budget.

1 INTRODUCTION AND BACKGROUND

The company has its own datacenter but is open to outsource the hardware (e.g. servers, switches and routers) to an external party if that is more economically viable. For this reason, the hardware needs must first be calculated, and then be compared to what hosting companies can offer. It is also important to keep in mind that even if it is cheaper to rent dedicated servers in a hosting company, in the long run it can pay off to purchase the needed hardware, even if the initial cost is higher. Therefore, it is important to have a somewhat reliable projected growth estimate to better be able to determine the hardware needs.

Choosing the correct software is crucial as well. The software for both the web server and database has to have the features needed for the website to function properly. Additionally there has to be software for surveillance as well as backup.

Surveillance of the physical hardware is critical so that the server can be gracefully shutdown in case of an impending failure and the bad component can be swapped. This is also necessary for hardware upgrade planning and additional purchases. Likewise, it is vital to be able to monitor the software to see that it is functioning properly.

A good working backup system is crucial for a busy company with many clients. In case of unexpected failure in hardware or software, it is preferable that backups are planned in such way that they can easily be used and restored. An outline of what and how often backups will be made is imperative. Often backups are deemed unnecessary and costly, but when problems arise it is good to know that they are available.

Along with this comes the issue of licenses. They have to be applicable to the given scenario and not break anything in the license agreement. The cost of the licenses is a great factor since they play a big part in the budget. Careful planning both now and for the future is very important in keeping the license cost down.

Finally, there is the capability of the network. It has to be able to handle the data traffic without being congested. This means that the network equipment has to be analyzed along with the Internet connection(s). Furthermore, the clients usage of the bandwidth has to be computed.

In summary this thesis will investigate the following subjects:

2 PROBLEM FORMULATION

- Investigate the necessary hardware for the server:
 - Choose a suitable CPU
 - Calculate the required disk space for the database
 - Determine the size of Random Access Memory (RAM) for the server
- Investigate what software is needed for:
 - Underlying Operating System (OS)
 - Web server and SQL server
 - Surveillance, backup and administration
- Investigate the network bandwidth available and compare to what is needed
- Compare outsourcing vs inhouse costs
- Secure the necessary network parts
- Plan for future improvements

2 Problem formulation

In discussions with SoftRobot AB, two main focuses arose. These focuses are further divided into sub-components below:

- How do you choose hardware for a specific website to function properly?
 - What are the CPU, hard-drive, memory requirements and are there other component required for the server(s)?
 - What are the necessary bandwidth requirements?
 - Is it viable for SoftRobot AB to host the site themselves or is it a better option to outsource to an external datacenter?
- How can the necessary security, hardware surveillance and backup be achieved?
 - What software is needed for the basic operation of the website including security, surveillance and backup?
 - How can such software be installed in SoftRobot AB's network?

3 Related work

The performance of web servers is typically reliable and they do not cause any problems that often. Sometimes, though, they are overloaded with requests, for example when a new software version is released and a large amount of people are trying to download it, or when tickets for a big event get released and the servers can not handle all the requests at the same time. The most simple solution is to upgrade the server(s) or to even buy more. This is of course not feasible for all companies since it can cost a large amount of money. A better solution would be to improve the servers software to handle the requests in a better way.

In an article by Schroeder and Harchol-Balter [1], they discuss how scheduling affects the performance of an overloaded server. Their research is questioning if fair queuing is the best schedule algorithm or if shortest-remaining-processing-time (SRPT) queueing is better. The main difference between those two algorithms is that fair queuing does not differentiate between requests that take a large amount of time and those that can be executed quickly. SRPT queueing, on the other hand, values small requests more and processes those before larger ones. Their result is that SPRT is better, specially if the server is overloaded. Since it processes more requests, the queue will not fill up as fast as with fair queuing. Consequently, less packets are being dropped due to TCP timeout. The long requests are not penalized as one might think, they take about the same time to be processed in both fair and SRPT queueing. In fair queuing, they get negligible amount of processing time compared to their size, since they have to share the resources with a lot of other requests and in SRPT long requests are put aside. When the overload has decreased to manageable amounts, fair queuing still has a lot of both small and large requests, as opposed to SRPT that only has the large ones to handle. This leads to SRPT as a better way to handle requests in an overloaded system.

Maintaining information security is also very important since the information that flow on the Internet can easily be captured. Since a lot of the information that the clients upload to SoftRobot's servers contain sensitive information, it is extremely important that no data or information gets into the wrong hands. Even if the data is not sensitive, a lot of harm can be done through man-in-the-middle attacks. It is also recommended by Google to encrypt all information that flows through the Internet [2]. These requirements can be met if encryption is implemented between the server and the clients, and can be acquired with Transport Layer Security (TLS). TLS puts a lot of

3 RELATED WORK

stress on the CPU since it requires heavy mathematical computations. To work around this, there exists special add-on cards that alleviates the CPU from such computations and also accelerates it. According to Coarfa et al. [3], these accelerator add-on cards can help in certain cases, but in others, it is more preferable to swap the CPU with a faster one or even add an additional CPU. Because the CPU performance increases with new models, the percentage that the CPU uses for TLS becomes lower and lower, leading to more power being available to other processes. This leads to, according to the authors, that if many TLS connections are made, instead of trying to lower CPU cost of TLS encryption, it is more efficient to try to improve the website's performance.

An interesting article was written by Lee and Kim [4] about the effects of a sudden increase in requests through a forward cache proxy. A forward cache proxy relays the request from a client to a server, and then delivers the answer back to the client and also caches the data. If subsequent requests are made for the same data, the proxy server delivers directly without requesting from the original server. They first propose implementing a Redundant Array of Independent Disks (RAID) on the server but later reject it on grounds that it is too expensive to maintain. Later, they propose to set a delay cache, i.e., delay the requests to the web server and do not cache the data. At first, it does not seem intuitive that a delayed request will decrease the response time but the numbers say that it does. The explanation to this is that the server delays the requests until the load has decreased to a manageable level. Only then it starts forwarding the requests. This creates a slight delay for some of the users but the overall response time does in fact improve. Although a forward cache server is not applicable here (as a reverse cache as Varnish [19] or Nginx [20] is needed) it is interesting to see how web data response time can be improved for clients. This would greatly increase the bandwidth speed and network performance for a lot of companies.

In a study by Prakash et al. [5], the authors make a comparison between two web servers, Apache and Nginx that are widely used [21]. The main difference between these two are that Nginx has an event-driven design and Apache is process-driven. The latter spawns a "worker" for each request in contrast to the former which only has one for all requests. According to their results Apache needs more memory and is slower than Nginx, but is somewhat more robust in extreme loads, i.e., has less TCP timeouts. Nginx has a great advantage in that it scales more efficiently under heavy load though.

Clark et al. [6] have an interesting article on the subject of live migration of a Virtual Machine (VM). The ability to migrate VMs that are operating between different physical machines is very important since this can help even out the load between servers. It is also crucial to be able to do this with minimal downtime in order that the clients accessing the server do not experience unnecessary delays in their work. To do this, the OS has to be able to see what parts of the memory are not being used and migrate those first. The parts that are constantly rewritten are marked as “dirty” and are only transferred in the final moments. This unfortunately leads to the VM being suspended, transferred and resumed at the other server. The tests involved migration of VMs running web servers and game servers and the results show that even if both servers were heavily used, the downtime was at maximum a few hundred milliseconds. In an extreme case, where large portions of the memory were rewritten, the downtime was 3.5 seconds but these cases were very rare in real-life situations. This delay information provides an insight into the down times in case of an unexpected event that forces migration of live VMs.

There are a lot of other studies available that handle the topics discussed in this thesis. Especially in the area of improving and optimizing websites depending on clients behavior on the websites. These papers are discussed in the respective sections, for example relevant study in the optimization phase is mentioned in Section 4.1.

4 Method

4.1 Cisco’s PPDIO model

Cisco has a good model of how the life-cycle of a network implementation should be, called PPDIOO, which stands for Preparation, Plan, Design, Implementation, Operation and Optimization [7, p. 37]. These phases all have different goals and are very much dependent on each other. Even though the model is mainly geared towards network development, it is also applicable to other scenarios in an IT environment and even outside IT.

- **Preparation**

The first stage is when the company establishes the requirements. Justifications of the necessities of the new services or upgrades are discussed, often with top management. The main goal here is a higher picture of

the needs of the company, i.e., why do we need this, is it financially viable etc. and not technical solutions.

- **Plan**

In the plan phase, more details of the new service are discussed, i.e., what do we need to accomplish this new service, what are the requirements, can we do it ourselves, budget limits etc.

- **Design**

When the prepare and plan phases are finished, the engineers take all that information and start designing the new service. A list of all the equipment that is needed throughout the project is created, a time plan is estimated, security concerns are taken into account etc. Everything that is needed later on in the project is listed here.

- **Implementation**

In the implementation stage, the equipment acquired is installed and configured according to the plan that was proposed in earlier steps. Here it is a good idea to test the design that was researched in the previous steps to see if there are major faults or flaws. It is better to change the design here than to realize later on that a major part of the design is not functioning and has to be redone. This could potentially save a lot of time.

- **Operation**

In the operation phase, the network and its services are fully functioning and are performing adequate. Most major faults are addressed and fixed. The network is monitored and data regarding, for example, its performance is saved for future reference. The data collected here gives a good foundation of what can be improved in the optimization phase.

- **Optimization**

In the last phase, the services are evaluated and minor faults and errors are corrected. The administrators should always find ways to continuously improve the performance with evaluation of the data collected in the operation phase. The original goals and requirements should be rechecked to see if they are met or if there is a need for a change in the implementation.

At the start of this thesis, Devo IT already completed the preparation phase. They have a need for a website and have ensured that it is a viable step and will create a great opportunity for the company to grow, as explained in Section 1. The next step, the planning phase, was conducted using needs assessment and gap analysis. This is explained in Section 4.2. The result of

those analyses is used for the design phase. These two phases are the major themes in this thesis. Initially the implementation phase was also being addressed, but time constraints made it a low priority, and this is further elaborated in the design part (Section 5) .

4.2 Needs assessment and gap analysis

According to Watkins et al. [8], a need is the difference between the current conditions and the desired outcome. In Figure 1, it can clearly be seen that a gap exists between those two. It is important to identify those two points before starting to develop a process that goes from point A to point B. The next sections identify what the current conditions are, what the desired outcomes are, and most important what process is needed for the desired outcome to be achieved.

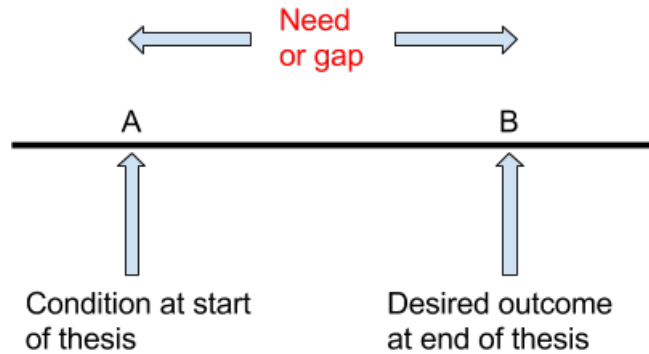


Figure 1: Depiction of gap analysis

4.2.1 Pre-conditions

At the start of the project, there is already an existing company network used for the existing services. The network consists of scanners and printers, and direct connections to external clients . Their network is outsourced to an external company that have developed and maintained it. There is an existing firewall configured in the network, blocking unwanted access from the Internet. The website and database to be used for the newly developed service of Devo IT were being developed by SoftRobot AB's inhouse developers before and during the work with the new network assessments. These were developed in a Microsoft environment. Regarding software for the database,

there was an existing license for the database being used by the developers. This license is a Microsoft SQL 2008 R2 Standard license, which is a production license. No other software or hardware was available for this project in house.

4.2.2 Desired outcome

The main goal of the project is that a fully functional website is up and running and that the network can handle the initial load of 200 clients, accessing the website and uploading and accessing their documents to and from the database. At the end of this thesis, the desired outcome is that the plan and design phases are completed and the implementation stage is starting with the main goal in focus. All the hardware needed for the project are evaluated and purchased. The network design should have been evaluated as well as its capacity. The necessary software for the servers has been purchased after carefully considerations of their respective functionality, license costs and easiness of both maintenance and administration. Both the database and web server have the necessary licenses as well as the underlying system to be able to function properly. A detailed backup plan is established and implemented.

4.2.3 Gap analysis and method

To make sure that the desired outcome is reached, first and foremost a needs assessment has to be made to be able to analyze what exactly is needed for the website to function properly. The hardware for the server has to be chosen with considerations taken in to account with respect to the different components; the CPU, disks and RAM are important since they play the biggest part in the server's performance. On an existing server an analysis can easily be made of the current conditions, and on that basis identification of bottlenecks, their correction and upgrade of the other components can be done. Since the website is not up and running on any server, and no load tests can be made, choosing some of the components can be challenging. There is an existing database with test documents that can be analyzed. These documents can give a good indication of how large the database will be with 200 clients, which is the target clientele for this project, and in the near future when more clients are connected and using the new service on the server. Since those calculations gives a hint of how much data will flow

through the network, they can also help to determine if the network is going to be a bottleneck.

The CPU is difficult to choose since there is no functioning server with the website to analyze. Choosing it is a matter of price and availability on the current market, this is also the matter of choosing the configuration giving the lowest total license cost for the server. Almost all servers today are equipped with dual sockets. If the CPU acquired is deemed to not be able to sufficiently handle all the clients, an additional CPU can easily be purchased and installed. Instead of investing a lot of time in evaluating the exact CPU needed, this is a good viable option since the cost would not be high.

The RAM needed is also not easy to estimate without the possibility to test on an existing hardware prior to purchasing. Fortunately, the RAM prices are quite low in comparison with the other hardware and software licenses, making it favorable to invest in a RAM with higher capacity than might be needed. If the need should arise in the future, additional RAM sticks can be purchased and installed in the existing server.

Virtualization of servers has to be taken into consideration. Questions like “is it a good option in this scenario or is it not necessary?” need to be answered. Both cost and ease of administration has to be compatible with the project. Backup and security has also an important part in this decision.

Since the website is written in ASP.NET, it is appropriate that the OS for the website should be Windows-based since ASP.NET is tightly bound to the Windows universe. The database is developed for Microsoft SQL, so the OS should also be Windows-based.

Software for the backup jobs should have the necessary functions for delivering efficient and error-free backups. These should also be compatible with the underlying virtualization server if such is implemented, or the database and website servers themselves. All software must have the appropriate license for this project. Important factors are the End Of Life (EOL) of all the programs, if there is an upgrade path and of course the cost.

If the virtualization path is taken, there must be a plan for how these VMs are administered. Different hypervisors have different ways of administering the VMs and sometimes extra functions that are needed can cost extra.

When all the software and hardware needs are established, a comparison should be made between hosting the website inhouse or outsourcing it. It should be planned not only for the present but also if the chosen option is able to handle the future load.

5 Design

5.1 Power

Having numerous devices and equipment requires a lot of electrical power. This means that clean and uninterrupted power is vital for keeping the equipment running. It is therefore a good idea to have some sort of protection against power outages and unclean power. If a computer has a sudden loss of power, the file-systems can be corrupted and the data lost. This is unacceptable in an environment where the loss of data can cost a lot of money. An Uninterruptible Power Supply (UPS) can keep the system running until the power comes back on and even shut down all servers if the outage is very long.

There are 3 main kinds of UPSs [22]:

- **Online** UPSs converts the power from AC to DC and then back from DC to AC. This creates total isolation from external power.
- **Line-interactive** has basic power protection. The internal circuit regulates the voltage up and down as well as cleaning the current from electrical noise, brownouts etc.
- **Offline or standby** UPSs feeds the equipment directly using utility power and only provides power when the external power is lost . Often is also protects from power spikes.

There was an UPS available in Devo IT's serverroom that could be utilized for this project. It is an line-interactive APC 900 PRO with 540 watts of power. Unfortunately this UPS is connected to another network so it cannot be used to shut down the servers in case of a long power outage. It will however keep the machines online and also clean the power from bad electrical components such as the aforementioned spikes and noise.

5.2 CPU

Choosing the right CPU is crucial because the OCR-reading depends heavily on it. The more powerful the CPU, the shorter time it takes for the OCR to complete and the user can continue with their work. Industry standard today is Intel Xeon processors which are very powerful and are used in a great percentage of servers all over the world [23] [24].

In a developer's computer, OCR scan of a document took in general between 10–15 seconds. This specific computer has a CPU from AMD model FX-8320. An Xeon CPU which fits into the budget and is available at the time is an E5-2620 version 3. The AMD CPU has an rating of 8016 in Passmarks benchmark site [25], compared to the Intel which has 10 004. This should give a slight performance increase, improving the OCR process. According to the developers the OCR-software can scale to multiple CPU-cores but there is no way to estimate how well it does that. It should also benefit more with more cores with lower clock rate rather than fewer cores with higher clock rate. Since most server have two sockets for dual CPUs, it should be clear after an evaluation period if the server would benefit from an additional CPU.

5.3 Climate control in Devo IT's server room

The server room that is available for this project is spacious and have at the moment multiple servers used for other projects. There is air conditioning equipment that controls the air temperature, filters and if necessary also dehumidifies the air. It is set to maintain the temperature at 20°C.

5.4 Disks

The Structured Query Language (SQL) database mainly uses the hard drives and does not require much CPU, so hard drive performance is most important. The documents are stored inside the database so it has to handle those in addition to all transaction data. This means that the database is not a pure database but also a file storage. This has to be taken in consideration when deciding hard drives.

Choosing Solid-State Drive (SSD) can be a good option if the database is relatively small since SSD are generally very fast. At higher capacity those are too expensive but if speed is really important and the budget is adequate SSD is recommended. Unfortunately Softrobot's database is predicted to grow very large (as seen in figure 3 in Section 5.4.3) and it would cost as lot of money to place the database on SSDs. Therefore, standard Hard Disk Drive (HDD) are chosen as main storage disks for the database. The hypervisor should also have HDDs since it does not benefit from fast disks once it has booted the server.

Since a reverse cache proxy (see section 5.8.1) is implemented, the cached files benefit greatly from being accessed from a fast drive. Since the caching does not have to be extremely large, the caching server is placed on fast SSDs.

5.4.1 RAID

Redundant Array of Independent Disks (RAID) is a term invented by Patterson, Gibson and Katz at the University of California in 1987. In an article [9] they presented a way to improve the reliability and performance of disks since they were not improving as rapidly as CPU and RAM. This caused problems since the CPU had to wait for the disks to deliver the information, consequently wasting a lot of performance capacity that could be used in a more efficient way. The paper explained some ways of improvement that later evolved into a number of ways to configure disks. These have their own strengths and weaknesses in performance, capacity and reliability:

- RAID 0 splits the data in blocks that are distributed to 2 or more disks. The advantage is that the data is read and written at a very fast rate. One big drawback is that if one disk fails, the array fails completely i.e. the data is lost. RAID 0 should therefore only be used in environments where the performance is extremely important and data loss is acceptable, like for example temporary files in video editing [9].
- RAID 1 mirrors the data to two or more disks. The main advantage here is that if one disk fails the other(s) can still function properly. This is however the most expensive RAID because most of the capacity is used for fault tolerance [9].
- RAID 2 & 3 involve striping the data to multiple disks in different ways. These methods are rarely used and are therefore not discussed any further [10, p. 18].
- RAID 4 stripes the data on to multiple disks but also writes parity data on a dedicated disk. This means that for reading, RAID 4 is very fast but writing is slow because the parity has to be calculated for the data that is written. The array can also tolerate one drive failure because the lost data can be calculated with the parity data. This “degraded array” of course makes the throughput slower but still functional. [9]

Adding a new drive to degraded array puts a lot of stress on the other drives because all the data has to be read for the parity data to be recalculated. This can be fatal because if another drive fails, all the data is lost, and most drives purchased at the same time will statistically also fail at about the same time. This also affects RAID 5 & 6. [11, p. 169]

- RAID 5 & 6 also stripes the data to disks but the parity is not located on a dedicated disk but instead spread out over all the drives. The difference between RAID 5 and 6 is that for 5, the parity only has 1 copy but for 6, the parity has two copies located on separate disks. The result is that RAID 5 can have one failed drive and RAID 6 can loose two and still function. This gives a greater margin in case of failure [9].
- RAID 10 is a combination of RAID 1 and 0. Two sets of mirrors are used and then the data is striped between the mirrors. This gives very good performance and also good fault protection.

5.4.2 Throughput calculation

Drive performance can be measured in different ways. Input/Output Per Second (IOPS) is often advertised by hard drive manufacturers since it usually has large numbers and customers like that. Unfortunately those numbers are difficult to compare since you do not know what they are based on. Are all read operations? All write? Mixed read/write with 4 kbyte files? 16 Kbyte files? All manufacturers use different tests to gain the highest value possible to be able to compete with other companies. The only way to get accurate results is to test drives yourself or go to renowned technical sites that often do thorough tests.

Tests are often done in read, write or mixed ways with different types of files, typically 2^n kbyte where n is a natural number. Another factor is if the tests are made sequentially or random. Sequentially means that the data is read/written continuously over the disk platters as opposed to random in which the data is scattered in random blocks all over the drive. A drive with good sequential performance is more aimed at archives, backups and video editing, i.e. where there are large files involved. Drive with good random performance on the other hand work best with small files like database operations and mail servers. Hence the focus here will be at random performance. Since MS SQL work with pages that are 8 kbyte big, it is only logical to look at the values that are for that size [12, p. 1101].

The different RAID levels in Section 5.4.1 all have their respective pros and cons, especially in throughput speed. To calculate how the levels compare to each other, a Seagate 1.8TB Enterprise Performance 10K v8 drive was chosen for comparison. This drive is aimed at enterprise level mission critical servers for high performance computing, storage arrays, and databases. According to Tom's IT PRO's tests, the random write operations at 8 kbyte are quite stable at ≈ 400 IOPS for the Seagate drive, regardless of the Queue Depth (QD) [26].

The read performance is a bit different where IOPS varies depending on the QD. The explanation is that the hard drive can optimize the requests when there are more than one in the queue, resulting in higher IOPS. Assuming the database's load is heavy, the most appropriate QD would be on the upper half. The maximum IOPS according to Tom's IT test is with QD 128 and 256 where the IOPS is ≈ 450 with random read operations at 8 kbyte [26].

- RAID 0 has all the data striped. This gives that both read and write operations delay are cut in half, i.e. IOPS are doubled [13].
- RAID 1 has similar write speed as a single disk. Data has to be written to both disks, but that can be done in parallel. Read is more complicated. In Linux and Windows 2000 Server and later, the raid driver tries to distribute multiple read operation to multiple disks, thus greatly increasing the performance [27] [28]. If read distribution is not implemented, the performance is of a single disk [13].
- RAID 4 is fast at reading since the data is striped. The throughput is $N - 1 \cdot IOPS$, where N is the total number of discs. Writing is a bad case. The parity causes extra reading and writing, essentially halving the IOPS. [13]
- RAID 5 slightly improves the read since all the disks can be used. For writing it is not as bad as RAID 4, here it is $\frac{N}{4} \cdot IOPS$ [13].
- RAID 6 adds more redundancy but loses disk space. The formula is the same as for RAID 5.
- Since RAID 10 is a combination of RAID 1 and 0, read and write speed is doubled from RAID 1.

Table 1 show a comparison between raid levels. The numbers are produced with the help of formulas from *Operating Systems: Three Easy Pieces* [13]. The calculations in Table 1 are in extreme cases where there are only read or only write operations. This is of course not real-world translatable but it gives a good overview of how the different RAID-levels perform compared to

	Read (IOPS)	Write (IOPS)	Space available
Single disk	450	400	1GB
RAID 0	1800	1600	4GB
RAID 1	450–1800	400	1GB
RAID 4	1350	200	3GB
RAID 5	1800	400	3GB
RAID 6	1800	400	2GB
RAID 10	900–1800	800	2GB

Table 1: RAID level comparison (for IOPS less is better)

each other. Note that all arrays are computed with 4 disks since that is the minimum for RAID 10. This makes it easier to compare the RAID-level’s performance.

Table 1 shows that RAID 0 gives the absolute best performance and also best space that is available for usage. Unfortunately it has, as mentioned before, no protection against a disk failure. The more disks are in the array, the more likely is that one will fail and the whole array is lost. RAID 1 overcomes the fault protection but the writing does not improve. RAID 4 has extremely poor performance when writing and should therefore not be used in a database scenario. RAID 5 and 6 improves the writing but it is still not good. The best solution is therefore to use RAID 10 for the database since it gives the best performance and fault protection, and RAID 1 for the hypervisor and cache server since they do not require extreme performance. The cache server is still very fast because it is placed on SSD disks. Figure 2 shows the selected raid levels for the different VMs.

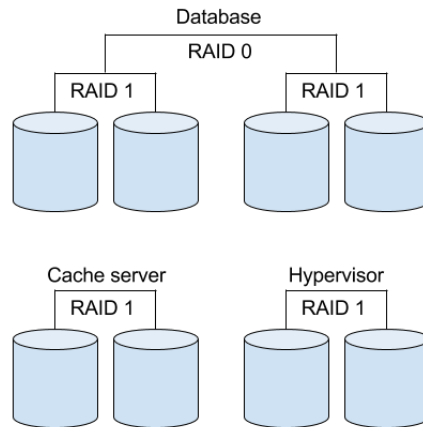


Figure 2: RAID layout for VMs

5.4.3 Space

Disk space has to be carefully considered for the SQL server since all the data is stored there. The SQL receives the document from the web server and stores it along with a duplicate in another format depending on the clients' needs. If the original document has several pages, the pages are saved as a separate file. Estimates from a test database gives that the duplicates are the same size as the original. The database will therefore need twice the amount of space as all the original files combined.

The average document in the test database is approximately 175 kbyte in size and roughly 175 000 documents are uploaded per month with 200 clients. Given those numbers the database would grow $175 \times 175\,000 \times 2 = 61\,250\,000$ kbyte every month, which is ≈ 58.4 GB. In 6 months, the database should have grown to $6 \times 58.4 \approx 350$ Gb. This gives a linear growth that is depicted in blue in Figure 3. That calculation is for a fixed number of clients (200). Devo IT's prediction is that more clients are added every month, and in 6 months the clientele should be about 500 clients. This would give an slightly exponential growth which is depicted in Figure 3 with the red line. The database would grow to ≈ 600 GB and that should be more accurate to this scenario. The growth will of course not increase forever, and the number of clients should reach a steady state at some point in the future.

The documents are mainly used in a short time-span after uploading. After that they are stored and very rarely accessed. To keep the database size smaller, the duplicates should be deleted after some time, keeping only the original files.

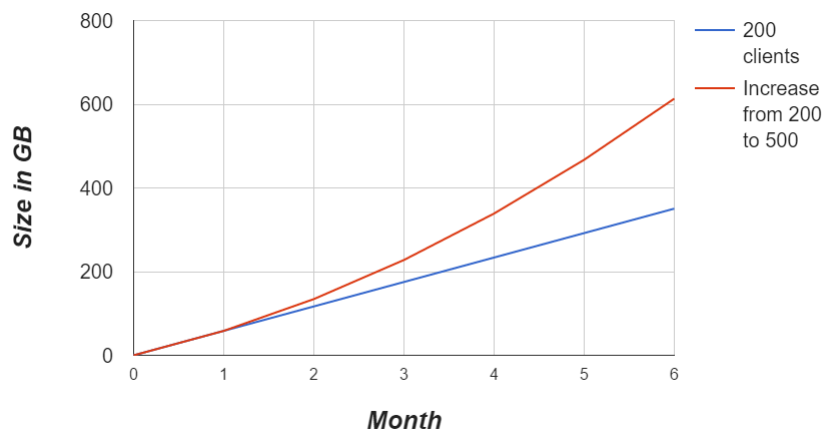


Figure 3: Predicted size of the SQL-database

5.5 Network

As previously mentioned, the network is maintained by Devo IT's supplier. There are numerous switches and servers used for their internal network. They have dual Internet connection installed in their data room with automatic failover. There is also a firewall that protects the internal network from external Internet. A 1000/1000 Gbit/s from Bahnhof is used as the main connection and a 100/100 Mbit/s connection from Bredbandsbolaget that is in standby and only used as a backup. The connection from Bahnhof is split 50/50 between SoftRobot's web server and Devo IT's internal network. This leaves 500 Mbit/s for the web server to handle the clients.

5.5.1 Bandwidth

The documents that are uploaded to the server are, as mentioned in Section 5.4.3, on average 175 kbyte and there are about 175 000 uploaded every month for 200 clients. That means 30 Gbyte uploaded every month. Assuming the clients only upload on work days (20 every month), this means $30/20 = 1.5$ Gbyte (12 Gbit) per day.

Suppose a worst-case scenario where all clients upload a day worth of documents simultaneously. In theory, it would take $12 \text{ Gbit} / 0.5 \text{ Gbit} \cdot s^{-1} = 24$ seconds to upload all documents, excluding the overhead from the TCP/IP protocol, the pages and images from the website and packet loss. Since the network speed is so fast, the delay will not be in the bandwidth but on the CPU. The processor will not be able to handle that load since every document has to be OCR processed and the webpages has to be rendered by the API which takes time for the CPU to compute. The bandwidth from the Internet is therefore established as more than sufficient.

5.6 Hypervisor

Virtualization is a great way to improve server efficiency. An OS that is installed on a hypervisor can be moved to another server, the hardware assigned to the VM can easily be adjusted depending on needs, and backups can be made of the entire OS on-the-fly.

There are a two main ways to virtualize an OS [14]:

Type 1, or native or bare-metal, is directly tied to the physical hardware and the guest OS lies on top of the hypervisor. This gives very good performance and is usually used in servers. Examples of bare-metal are VMware ESXI [29], Microsoft Hyper-V [30], Citrix XenServer [31] and Kernel-based Virtual Machine (KVM) [32].

Type 2 hypervisors are installed on a consumer operating system like Windows or OS X. The OS sits on top of the hardware and there lies the hypervisor. Consequently there is an extra layer between the guest OS and the physical hardware. Performance can be good but almost never as fast as bare-metal hypervisor. Examples include VMware Workstation [33], Parallels Desktop [34] and Oracle VirtualBox [35].

Type 1 is being used here since the project is server-oriented. Choosing the exact hypervisor can be tricky. Often tests have to be made to see which one gives the best performance in regards to for instance CPU, disks and network speed. This project has limited time and to thoroughly test all hypervisors would be wasting a lot of time. Price is also a factor since the budget is limited. Table 2 shows the most popular hypervisors and their respective license and cost [15]. VMware ESXI is only sold in the package VMware Vsphere which is very expensive and thus not a viable option here. The company uses mostly Microsoft products and does not have much experience with Linux operating systems. KVM and Xen are therefore not considered, leaving Hyper-V as the remaining candidate.

	License	Price
VMware Vsphere Std	Proprietary	€894.50/CPU
Microsoft Hyper-V	Proprietary	Free
KVM	Open-source	Free
Xen	Open-source	Free

Table 2: Price of baremetal hypervisors (2016)

The latest version (2012 R2 as of feb. 2016) of Hyper-V has many features that are great for this project [36]:

- Checkpoints and snapshots
- Dynamic Memory
- Live migration
- Failover Clustering

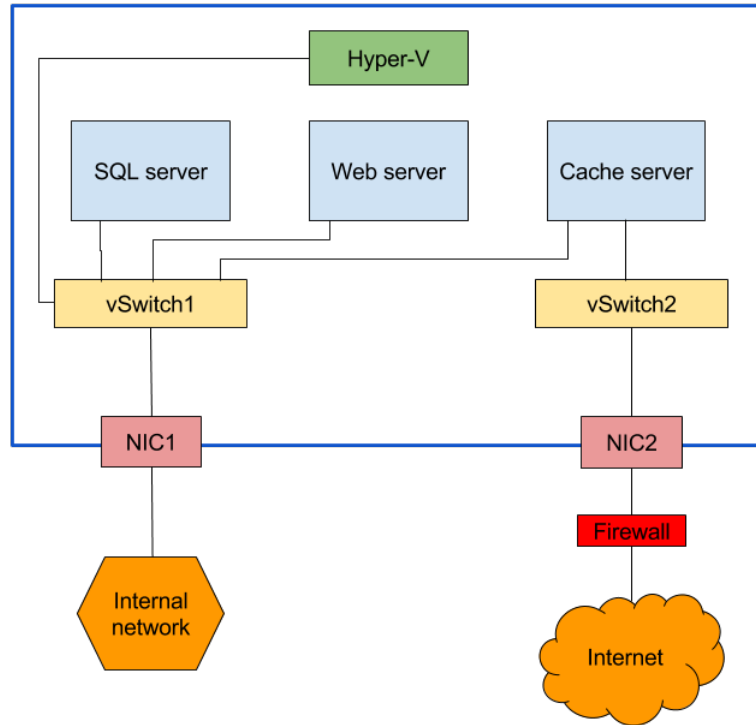


Figure 4: Layout of servers in a physical machine

Figure 4 show the layout of the VMs in the hypervisor. The only VM that is facing out towards the Internet is the cache server. Both the web server and the SQL-database are only accessible from inside the local network. This layout creates a very secure environment for the VMs.

5.6.1 Administration

There has to be a easy and logical way to administer and monitor the VMs for Devo IT. Hyper-V greatly benefits of being in an domain to be able to use Microsofts hyper-V manager. This however is waste of resources and makes things unnecessary complicated since all computers which administers Hyper-V has to be part of the domain. Without a domain it is very tricky to connect to the hypervisor [37]. A viable option is to use an external program.

Two programs for Hyper-V administration are Microsoft's System Center [38] and 5Nine Manager [39]. System Center costs \$1 323 for a 2-CPU license [38] and 5Nine Manager \$349 also for a 2-CPU license. Since they both have the needed features for this project, 5Nine Manager should be chosen.

5.7 SQL Server

The database was created in Microsoft SQL 2008 R2. Since the licenses for MS SQL are costly it would be a good idea to migrate the databases to an open-source model e.g. MySQL or MariaDB. This would unfortunately take a lot of time and effort for the database developers and is thus not a viable option at this moment. The only option is therefore to continue using Microsoft SQL. The migration to an open-source alternative should be discussed in the future though.

As mentioned before the developers are using a server with Microsoft SQL 2008 R2 Standard. This license is for a full production server. Buying a license for that same version would not be a good idea since its EOL is in July 2019. No more security updates are made after that. A better option is to purchase a latest version which is (as of May 2016) SQL 2014. This version has security updates until July 2024. [40]

5.8 Web server (software)

The web server files were written in C# in ASP.NET in Visual Studio. This Application Programming Interface (API) is although available on open-source software as Linux, very much geared to Windows servers. A thorough testing of the website on Linux would take lots of time and effort. The OS for the web server should therefore be Windows Server. Another factor is that the OCR-software (Finereader 12 Corporate) is a Windows-only program. Putting the web server and the OCR-program on different VMs would be unnecessarily complicated and not add anything in the form of performance or lower cost.

The latest version of Windows server today is Windows Server 2012 R2. It comes in four different editions [16, p. 2]: Foundation, Essentials, Standard and Datacenter. These have different features depending on the needs of the company. Foundation can only be installed in a server with one CPU so that edition is not adequate here [41]. The Essentials edition can be virtualized 1 time per license, meaning that in this scenario two licenses would have to be purchased. Additionally it has a RAM limit set to 64GB which can be a factor in the future growth. The two remaining contenders, Standard and Datacenter, have the exact same feature set except from the virtualization rights. With a Standard license, two Windows Server VMs can be installed but a Datacenter license gives the right to install unlimited VMs. In an environment where Windows Server would be virtualized in many VMs, it

would be a good option to buy the Datacenter edition. Here though it will only be virtualized two times per server, one for the web server and another for the SQL database. A Standard license is therefore sufficient in this case.

5.8.1 Reverse caching proxy

A lot of the data that is transmitted over Devo's network is static data. A high percentage are the pdf, tiff and png files that clients upload or request from the servers. That data can be cached so that it relieves both the web server and the SQL server. This can be created with a reverse caching proxy. The proxy is placed before the web server so that when it receives requests, it forwards them to the web server. When the web server sends its response, the proxy sends them to the user and at the same time caches certain files. What files to cache is of course configurable depending on the data that is transferred. Subsequent requests for that file are then sent directly from the caching server instead of being processed by the web server. This alleviates both the SQL server and web server, especially if there are a lot of requests for the same data.

The proxy can also be configured to load-balance between multiple web servers, creating a very good environment with seamless failover. This is a very good feature because the clients would not notice if a server fails.

To keep costs down, an open source alternative is chosen. The biggest competitors are Apache and Nginx. Apache has a problem handling a large load of simultaneous requests [17, p. 76], therefore Nginx was chosen as the caching server.

5.9 Security

5.9.1 Firewall

All the servers and equipment are behind a firewall that is installed and maintained by Devo IT's network maintainer. This does not mean though that the servers should not be secured. All services that are not needed on the servers should be turned off and its firewalls should block all incoming and outgoing data that is not relevant for the operation of the web server.

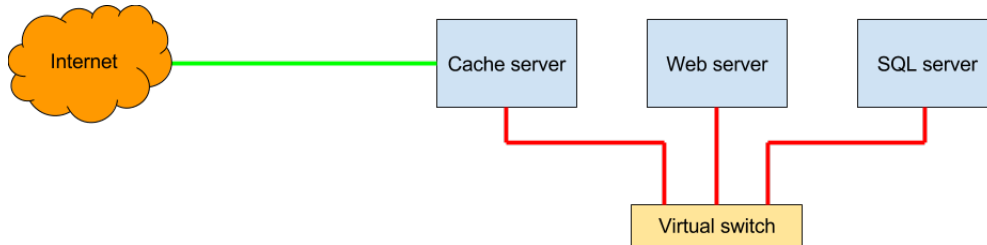


Figure 5: SSL encryption

5.9.2 SSL certificate

Since the clients' documents contain sensitive information such as medical journals and invoices, the traffic must be encrypted. This can be accomplished with Secure Sockets Layer (SSL). Encryption and consequently SSL provides three important things [18, p. 18]:

- Encryption provides *authentication*, meaning that the other person/company really are who they claim to be.
- It also provides *confidentiality*, that secures the data so nobody else can read it.
- The last piece is *integrity*, which assures that the data is not tampered with.

With these three items, the data communication between a client and a server becomes very secure.

A drawback of this is that since the data is encrypted, the caching proxy can not identify what is in the data packets, and therefore can not cache any information. This leads to that the caching server must terminate the SSL connection to be able to cache the data. The connection from the caching server to the web server and SQL is not encrypted, but this is not a major security issue since those servers are not accessible from the outside.

Figure 5 shows where the data is encrypted (green) and where it is not (red).

5.10 Backup

A great advantage with virtual servers is that the backup and restoration are greatly simplified. Before virtual servers it was a difficult task to create backup of the whole server because the OS sometimes locks certain files and

those cannot be read. The only solution was to shutdown the entire server and boot a backup OS to read the hard drive. Restoration had to be done in a similar fashion.

With virtual servers, the hypervisor does not have to shutdown since the virtual server is just a file in the OS. The backups can easily be scheduled and start even if the virtual server is online! Restoration is also greatly enhanced and a crashed server can be restored in a matter of minutes.

Backups should be made with the 3-2-1 rule in mind [42][43]:

3. The first number is how many copies of the file that should exist. This greatly decreases the possibility that all the backups should fail.
2. The data should also be stored on two different medias. For example in addition to the data in an internal drive, a copy should exist on an internal drive on another computer or an external drive such as an external USB drive, DVDs and tape drives.
1. One copy should also exist in an offsite location. The purpose of this is in case of a location incident such as robbery or a fire.

There are a few backup programs for Hyper-V. It would take too long to evaluate all of them, so three main contenders are compared here: Veeam Backup Essentials, Unitrends Enterprise Backup and Altaro VM Backup. All three companies offers several different version of their respective software. The programs should at least have the following features:

- **Backup to network drive** With many servers it is a good idea to have one place with all backups. A Network Attached Storage (NAS) is an easy way to store the backups, is cheaper than a server and hard drives can be chosen depending on the size of the backups.
- **Backup live VM** Powering off an VM is not a good option because it can cause financial losses since clients are not provided the service they want. The backup can also take a lot of time and cause errors if other services are depending on it. Live VM backup is a critical functionality in the backup software.
- **Scheduled backup** Backing up one server may not be so hard but when there are 50 servers it can be a daunting task. Although not critical, it is good to be able to schedule backups to the middle of the night when there is low access to the servers. This means less load on the servers and no missed backups because the backup engineer was not available or absentminded.

- **Incremental backup** The SQL database is very large and a lot of its data are static files such as png and pdf. Backing up those files every day is a waste of network capacity, disk performance and disk space. A better way is to only back up the files that have changed. This makes the daily backups a lot smaller. A full backup can be scheduled on weekends when there is less activity and small incremental backups on weekdays when the load is higher. A drawback with this method is that if a restoration is made, the backup program has to add all incremental backups to catch up to the latest, and you often want the latest backup. This can be remedied with reverse incremental backup, in which the incremental backup is added to the full backup and the difference is saved as an earlier backup. This gives the benefit that the latest backup is always a full backup and earlier backups are differential files.

	Version	2 year, 1 host (8-core CPU)	2 year, 2 hosts (8-core CPU x4)
Veeam B&R	Standard	6240 SEK	24960 SEK
Unitrends	Essentials	\$299	\$1196
Altaro	Unlimited	€495	€990

Table 3: Price of backup software (2016)

Given the above requirements, Table 3 shows the backup programs with the matching functionality and what they would cost in the future. Veeam is clearly much more expensive than both Unitrends and Altaro that are about the same price. Choosing between Altaro and Unitrends is a matter of taste for the administrators since both programs have very similar features.

5.11 Surveillance

To monitor the hardware status is vital to ensure that all components have good health. This also creates a good chance to detect a failing component and replace it before it brings down the server in a spectacular crash. A good way to do this is with the help of the Intelligent Platform Management Interface (IPMI). This is a small, often Linux-based, computer that comes included in many servers, but is totally independent of the main server. It can therefore monitor the server's components without affecting the server's performance. A good way is using a Linux-based OS like Ubuntu [44] and installing a monitoring software like Icinga [45] or Zabbix [46]. Configuring this

and connecting it to the IPMI-enabled network interface on the servers gives a very good overview of the different servers, their respective components and their performance.

5.12 Inhouse vs outsourced hosting

There are many hosting companies available today that offer a services of good quality. Since the data is of the sensitive kind, the servers must be located in Sweden. Two contenders that offer the required service are FSdata and iPeer. They both have the necessary hardware options and offer good service level agreements. As seen in Table 4, the cost of renting 1 server costs between 73 000 and 85 000 for a two year contract. After that time period, another contract has to be signed for probably the same amount of money. Compared to for example a server from Hewlett-Packard, the cost is much lower and it is only one payment.

	FSdata	iPeer	Hewlett-Packard (1-time cost)
Cost (24 months)	85 000	73 000	55 000

Table 4: Cost calculation for 1 server and 1 Windows 2012 R2 license

If the servers are only needed for a smaller time period e.g. less than a year without much special hardware, for example extra disks and software, it is probably a good idea to rent them at a hosting company. But if the servers are needed for a longer period, purchasing the hardware pays of in the long run.

In addition to the hardware cost there are other aspect that have to be taken into consideration. The following list shows the most important pros and cons on inhouse vs outsourced hosting:

- Pros of inhouse hosting
 - Good control over the hardware
 - Data is on site
- Cons of inhouse hosting
 - Electrical cost
 - Maintenance cost (personnel)
 - Up-scaling has to be considered

- Pros of outsourced hosting
 - No hardware worries, the hosting company handles it
 - Low initial cost
 - Can be up and running in hours
 - Often good 24/7 support
- Cons of outsourced hosting
 - Limited control of hardware and data
 - Can be costly in the long run

6 Discussion & Future work

With the design phase finished, all software and hardware requirements should be clear. Reviewing the problem formulation in Section 2, it is clear that all of the questions have been answered. The server hardware including disks, CPU and RAM are defined and ready to be purchased. The CPU requirements were the most difficult since there was no possible way to decide what exact model of CPU should be purchased. Consequentially, both CPU and RAM are chosen entirely on availability. The disk space needed for the SQL-server was calculated and with that information the bandwidth could also be analyzed and its requirements established. The software needed was investigated and their features and costs where compared to each other and also to the requirements for the project.

Concerning network security, not much is needed since Devo IT's own network administrators have already established good security against outer threats. The virtual servers' security should be tightened though, even if they are placed behind a firewall. There is almost never a good motivation to skip security enhancements. A greater concern though would be the website and its code that could have some vulnerabilities, but that is out of the scope for this thesis. The administration and surveillance programs for both the hardware and the VMs are chosen, keeping in view the costs. For backing up, there are two programs: Unitrends and Altaro, that cost about the same and have similar features. In the implementation phase, both programs can further be tested and a choice between those two be made.

A crucial decision was if Devo IT should buy all the hardware and maintain the servers themselves, or outsource it to an external company. As expected, the cost is higher to outsource the hardware. The longer a server is expected to run, the more it pays off to buy the hardware. For smaller temporary projects, outsourced servers are cheap and flexible.

Initially the time plan included the ‘implementing’ phase in Cisco’s PPDIOO model. This unfortunately had to be moved outside of this thesis since several factors made it impossible. There was a tight schedule for the developers to finish both the website and the database. This made the script for testing the load on a server impossible since it would have taken too much time from the developers. Another factor was that that website and database were developed tightly together and could not be easily copied to a new server. To be able to test the website and database would have given important information about how a server handles the client. That information could have been used to more carefully choose server hardware as CPU and RAM. These were instead chosen on market availability and could, in the end, prove not to be efficient enough. Adding or exchanging components is not severely expensive but this will be an unnecessary cost that could have been avoided.

The next step in the project is the ‘implementation’ phase. It should not cause major problems since there is a well laid out plan for the project. During the operation and optimization phases, most major and minor issues should be addressed and resolved. It should also give a great deal of information regarding the server’s performance. The CPU should be the most interesting to see how well it performs and if an extra CPU is needed or if it should be exchanged for with a more powerful one.

A big single point of failure is if there is only one server processing the clients documents. For testing and evaluation of the servers performance, this can be overlooked but in full operation with many clients that rely on this service, there should be at least two servers. If server A crashes then server B can take over the operation until the administrators can get server A back online. Server B may be severely impacted with the heavy load but the service will at least be functioning, albeit not at its best performance. This is the most important factor in providing a good quality of service for the clients and should be addressed during or after the ‘optimization’ phase.

Since the servers are not properly monitored by the UPS, there is a probability that the batteries inside the UPS will not last through a long power outage. Without the UPS being able to shut down the servers cleanly, there is a great risk that the file systems on board become corrupted and data is lost. There should be an UPS specifically for this project.

7 Conclusion

The thesis concludes the ‘preparation’, ‘plan’ and ‘design’ phases of Devo IT’s project. The need analysis is finalized and there is a quite clear list of both hardware and software that is needed in the remaining stages of the project. Thorough research and analysis results in efficient selection of servers that not only perform great, but is also cost efficient in the form of avoiding unnecessary purchases and time delays. Less hardware also means less electrical power being used and also less waste, which creates a healthier and greener environment.

It may be interpreted that Devo IT’s project from here on is a smooth downhill ride. On the contrary, it will take a lot of time and effort to advance the project to a stage where it is acceptable as a 24/7 service for clients. The road towards a defined goal can be clear at the beginning but may and will almost certainly change, often in places that are not thought of. A flexible well laid out plan with the help of plenty of available resources can help in the battle against unforeseen consequences. Nevertheless, this thesis will provide a good base for Devo IT to create a service that will greatly enhance their growth and attract a lot of new clients.

References

- [1] B. Schroeder and M. Harchol-Balter. “Web servers under overload”. In: *ACM Trans. Inter. Tech.* 6.1 (Feb. 2006), pp. 20–52. DOI: 10.1145/1125274.1125276.
- [2] P. Far and I. Grigorik. *HTTPS Everywhere*. Invited talk at Google I/O. June 2014.
- [3] C. Coarfa, P. Druschel, and D. S. Wallach. “Performance analysis of TLS Web servers”. In: *ACM Trans. Comput. Syst.* 24.1 (Feb. 2006), pp. 39–69. DOI: 10.1145/1124153.1124155.
- [4] D. Lee and K. J. Kim. “A Study on Improving Web Cache Server Performance Using Delayed Caching”. In: *2010 International Conference on Information Science and Applications*. Institute of Electrical & Electronics Engineers (IEEE), Apr. 2010. DOI: 10.1109/icisa.2010.5480543.
- [5] P. P, B. R, and M. Kamath. “Performance analysis of process driven and event driven web servers”. In: *2015 IEEE 9th International Conference on Intelligent Systems and Control (ISCO)*. Institute of Electrical & Electronics Engineers (IEEE), Jan. 2015. DOI: 10.1109/isco.2015.7282230.
- [6] C. Clark et al. “Live Migration of Virtual Machines”. In: *In Proceedings of the 2nd ACM/USENIX Symposium on Networked Systems Design and Implementation (NSDI)*. 2005, pp. 273–286.
- [7] R. Froom, B. Sivasubramanian, and E. Frahim. *Implementing Cisco IP Switched Networks (SWITCH) Foundation Learning Guide: Foundation learning for SWITCH 642-813 (Foundation Learning Guides)*. Cisco Press, 2010. ISBN: 1587058847.
- [8] R. Watkins, M. W. Meiers, and Y. Visser. *A Guide to Assessing Needs: Essential Tools for Collecting Information, Making Decisions, and Achieving Development Results (World Bank Training Series)*. World Bank Publications, 2012. ISBN: 0821388681.
- [9] D. A. Patterson, G. A. Gibson, and R. H. Katz. *A Case for Redundant Arrays of Inexpensive Disks (RAID)*. Tech. rep. UCB/CSD-87-391. EECS Department, University of California, Berkeley, Dec. 1987.
- [10] D. Vadala. *Managing RAID on Linux*. O’Reilly Media, 2002. ISBN: 1565927303.

-
- [11] E. Marcus and H. Stern. *Blueprints for High Availability*. Wiley, 2003. ISBN: 0471430269.
 - [12] R. Rankins et al. *Microsoft SQL Server 2008 R2 Unleashed*. Sams Publishing, 2010. ISBN: 0672330563.
 - [13] R. H. Arpaci-Dusseau and A. C. Arpaci-Dusseau. *Operating Systems: Three Easy Pieces*. 0.91. Arpaci-Dusseau Books, May 2015.
 - [14] J. Sahoo, S. Mohapatra, and R. Lath. “Virtualization: A Survey on Concepts, Taxonomy and Associated Security Issues”. In: *2010 Second International Conference on Computer and Network Technology*. Institute of Electrical & Electronics Engineers (IEEE), 2010. DOI: 10.1109/iccnt.2010.49.
 - [15] D. Rountree and I. Castrillo. *The Basics of Cloud Computing: Understanding the Fundamentals of Cloud Computing in Theory and Practice*. Syngress, 2013. ISBN: 0124059325.
 - [16] M. Minasi et al. *Mastering Windows Server 2012 R2*. Sybex, 2013. ISBN: 1118289420.
 - [17] D. Aivaliotis. *Mastering Nginx*. Packt Publishing, 2013. ISBN: 1849517-444.
 - [18] S. A. Thomas. *SSL & TLS Essentials: Securing the Web*. Wiley, 2000. ISBN: 0471383546.

URL References

- [19] P.-H. Kamp. *Varnish HTTP Cache*. URL: <https://www.varnish-cache.org/> (visited on 05/14/2016).
- [20] Nginx Inc. *Nginx*. URL: <https://www.nginx.com/> (visited on 05/14/2016).
- [21] Netcraft. *January 2016 Web Server Survey*. URL: <http://news.netcraft.com/archives/2016/01/26/january-2016-web-server-survey.html> (visited on 05/14/2016).
- [22] Eaton. *UPS Basics*. URL: http://www.eaton.com/ecm/groups/public/@pub/@eaton/@corp/documents/content/pct_1548681.pdf (visited on 05/25/2016).
- [23] M. J. Miller. *Can AMD, ARM, or IBM Dent Intel's Server Dominance?* URL: <http://forwardthinking.pcmag.com/none/323603-can-amd-arm-or-ibm-dent-intel-s-server-dominance> (visited on 05/14/2016).
- [24] Trefis Team. *Two Factors That Can Alter Intels Valuation By More Than 10*. URL: <http://www.forbes.com/sites/greatspeculations/2015/05/21/two-factors-that-can-alter-intels-valuation-by-more-than-10> (visited on 05/14/2016).
- [25] PassMark Software. *CPU Benchmarks*. URL: <https://www.cpubenchmark.net/> (visited on 05/14/2016).
- [26] P. Alcorn. *Seagate Enterprise Performance 10K v8 And TurboBoost HDD Review*. URL: <http://www.tomsitpro.com/articles/seagate-enterprise-performance-10k-v8-turboboost-hdd-review,2-937.html> (visited on 05/14/2016).
- [27] *Multiple Device Driver md(4) - Linux man page*. URL: <http://man7.org/linux/man-pages/man4/md.4.html> (visited on 05/14/2016).
- [28] Microsoft. *Mirrored Volumes*. URL: <https://technet.microsoft.com/en-us/library/cc938487.aspx> (visited on 05/22/2016).
- [29] VMware Inc. *VMware ESXi*. URL: <http://www.vmware.com/products/esxi-and-esx/> (visited on 05/22/2016).
- [30] Microsoft. *Hyper-V*. URL: [https://technet.microsoft.com/en-us/library/mt169373\(v=ws.11\).aspx](https://technet.microsoft.com/en-us/library/mt169373(v=ws.11).aspx) (visited on 05/22/2016).

- [31] Citrix Systems. *XenServer, Open Source Virtualization*. URL: <http://xenserver.org/> (visited on 05/22/2016).
- [32] *Kernel Virtual Machine*. URL: <http://www.linux-kvm.org> (visited on 05/22/2016).
- [33] VMware Inc. *VMware Workstation Pro*. URL: <http://www.vmware.com/products/workstation> (visited on 05/22/2016).
- [34] Parallels IP Holdings GmbH. *Parallels Desktop for Mac*. URL: <http://www.parallels.com/eu/products/desktop> (visited on 05/22/2016).
- [35] Oracle. *VirtualBox*. URL: <https://www.virtualbox.org> (visited on 05/22/2016).
- [36] Microsoft. *Hyper-V*. URL: [https://technet.microsoft.com/en-us/library/mt169373\(v=ws.11\).aspx](https://technet.microsoft.com/en-us/library/mt169373(v=ws.11).aspx) (visited on 05/15/2016).
- [37] J. Howard. *Guide to remote manage Hyper-V servers and VM's in workgroups or standalone*. URL: <https://social.technet.microsoft.com/Forums/windowsserver/en-US/fe80f0aa-0697-4657-a1da-19d36b036698/guide-to-remote-manage-hyperv-servers-and-vms-in-workgroups-or-standalone> (visited on 05/22/2016).
- [38] Microsoft. *System Center*. URL: <http://www.microsoft.com/en-us/server-cloud/products/system-center-2012-r2/> (visited on 05/22/2016).
- [39] 5nine Software. *5nine Software*. URL: <http://www.5nine.com/> (visited on 05/22/2016).
- [40] Microsoft. *Microsoft Support Lifecycle*. URL: <https://support.microsoft.com/en-us/lifecycle?C2=1044> (visited on 05/25/2016).
- [41] Microsoft. *Windows Server 2012 R2 Licensing Datasheet*. URL: http://download.microsoft.com/download/F/3/9/F39124F7-0177-463C-8A08-582463F96C9D/Windows_Server_2012_R2_Licensing_Datasheet.pdf (visited on 05/15/2016).
- [42] M. Levkina. *How to follow the 3-2-1 backup rule with Veeam Backup & Replication*. URL: <https://www.veeam.com/blog/how-to-follow-the-3-2-1-backup-rule-with-veeam-backup-replication.html> (visited on 05/14/2016).
- [43] Y. Pusin. *The 3-2-1 Backup Strategy*. URL: <https://www.backblaze.com/blog/the-3-2-1-backup-strategy/> (visited on 05/14/2016).

URL REFERENCES

URL REFERENCES

- [44] Canonical. *Ubuntu*. URL: <http://www.ubuntu.com> (visited on 05/22/2016).
- [45] The Icinga Project. *Icinga*. URL: <https://www.icinga.org/> (visited on 05/22/2016).
- [46] Zabbix LLC. *Zabbix, The Enterprise-class Monitoring Solution for Everyone*. URL: <http://www.zabbix.com> (visited on 05/22/2016).