Radboud University

Master's Thesis

Cloud Computing

Author:
Mark Spreeuwenberg
s0609846

Supervisors:
Marko van Eekelen
Ben Dankbaar
René Schreurs

Thesis number: 657
Faculty of Science - Computing Science
August 16, 2012

Abstract

Cloud computing is a hot topic today. Cloud computing is more than a hype; it is a change in how IT within an organization will be organized. Sometimes it is unclear what the definition of cloud computing is, but the common characteristics include access to IT resources over the Internet, these resources scale on-demand and they are being paid in a pay-as-you-go manner. Some well-known service models are Software as a Service (SaaS), Platform as a Service (PaaS) and Infrastructure as a Service (IaaS). These service models can be combined with a deployment model: public, private or hybrid. All these combinations have their advantages and disadvantages. For example SaaS will be easy to use for end users, but is less flexible while IaaS is meant for network architects and highly flexible. For a public cloud the advantage will be that it is often cheaper than a private cloud while there is less control over the software and hardware. Cloud computing is supposed to give quite some advantages to the service provider and customer. A service provider can become a specialized party in the offering of a service while a customer can work more efficiently and doesn't have to buy hardware for peak loads. Cloud computing has some disadvantages as well. One of the biggest risks is the unavailability of the service because of the dependence on the provider. Furthermore legislation can be a real obstruction for the adoption of the cloud. For some organizations can be prohibited to transfer personal data to a specific region. In particular circumstances it can be more advantageous to put products in the cloud than in other situations. For example when the application that runs in the cloud is stateless and when the application does not have much interaction with back-end systems. Furthermore the advantages for smaller organizations might be bigger than for large organizations because the small organizations might not be able to pay the start-up investment of an on-premise solution. A general remark is that there is not only one solution available and that the solution depends on different circumstances and business values. Important factors in defining the solution are the business values as well. Sometimes an organization has some values that do not correspond with the most efficient solution. Maybe the most important recommendation is that the movement to the cloud (once the software is standardized) should be done in phases. This means that is will be better to do this in small parts instead of moving to the cloud at once. This thesis contains a case study in which the circumstances within Aia Software are being researched and, based on this research, a cloud design for their product is developed. In the end the concept of the design is proven with a proof of concept. The proof of concept shows that it is possible to run several ITP installations on the same system without having the possibility of customers accessing the installation or data of other customers.

Preface

This master's thesis is the result of six months of research which I have done during an internship at Aia Software BV in Nijmegen. This research was performed as the final part of my study Computing Science, with Management and Technology as track, at the Radboud University in Nijmegen.
Aia Software is a global developer and supplier of the ITP Document Platform, a solution for the production and distribution of document output based on application data. The company was established in 1988 in the Netherlands and is serving the global market. Aia Software's ITP Document Platform is currently used by more than 1000 organizations in more than 30 countries worldwide. I would like to thank Aia Software for accepting me as an intern and providing me with the required resources.
I would also like to thank several people who made it possible for me to complete my study and thesis. First of all I would like to thank three people especially:

- *Prof. dr. Marko van Eekelen*, my supervisor at the Radboud University from the study computing science, for his guidance with the technical part and his feedback. His feedback really helped me with writing a scientific thesis that has enough content to graduate.

- *Prof. dr. Ben Dankbaar*, my supervisor at the Radboud University from the management and application track, for his ideas about the management aspects of this thesis. His feedback helped me to write a thesis that is understandable by non-technical people as well.

- *René Schreurs*, my supervisor at Aia Software, for providing me with all the required resources and knowledge about the company. His ideas and insights really helped me by creating the results.

Secondly I would like to thank my colleagues within Aia Software who shared their insights and ideas with me. They were always willing to answer my questions and participated in some discussions. Last but not least I would like to thank my family for supporting me during my entire study.

Contents

1 Introduction

For the study computing science this master's thesis was written as a result of an internship at Aia Software in Nijmegen. Research was carried out in this company about how they should implement a cloud based version of their ITP (Intelligent Text Processing) Document Platform.

1.1 Problem statement

Cloud computing is nowadays a trendy topic. In quite some papers there is a notion that organizations should switch now before it is too late. Cloud computing is expected to provide a lot of benefits for consumers as well as for software vendors. This is the reason that Aia Software wants to take a look at the possibilities of cloud computing for their product and whether this is possible at all. Since there is no cloud based solution yet, a lot of research has to be done on the impact of this for the design of the software. Aia is, for example, interested in questions concerning security. How can it be guaranteed that data of one client is not leaked to another client? Another interesting thing is to investigate the legislation about cloud computing. There are for example laws that state what the physical location of a server, on which the data of an organization is stored, should be. The solution should not take too long before a task is finished and it should be scalable in order to be able to extend it later on, when more capacity is needed.

1.2 Research Question

Based on the problem statement in the previous section I have defined a research question:

> Under what conditions is putting products in the cloud more advantageous for organizations than other solutions and how can this be done?

In order to get an answer to this research question I have defined a set of smaller sub questions:

1. What is cloud computing?
2. What possibilities are there for implementing cloud computing in general?
3. Which security aspects are important and what are the consequences for the design?
4. Which legislation is relevant and what are the consequences for the design?

5. What are the characteristics of Aia's current product with special attention for the architecture and security aspects?

6. What are the characteristics of the solution for Aia in the cloud?

7. What are the costs of the solution?

1.3 Approach

This section provides a description of the approach that was used to answer the research question and its sub questions. At the base of answering sub questions 1 to 4 lies a literature study. Since the legislation is very unclear when talking about cloud computing, there were interviews and conversations with people within the faculty of science that have knowledge about this topic.

Sub question 5 was answered by talking a lot to people within Aia Software since they are the people that have the experience with their own product. In particular there were a lot of informal talks with the Research and Development department because of their knowledge of the current possibilities and design of the application. The rest of the information was gathered by reading the company website and some other documents about the product. After reading those documents the information was verified by the people within the company.

In order to answer sub question 6, there were informal talks about what should be possible in the cloud. Different possibilities were presented to people within Aia Software in order to get feedback whether it could be a good solution. In order to answer this question, information about general cloud solutions was read first. After reading this information the relevant aspects were taken into account for Aia's solution. During the process of developing the solution it was also necessary to have some knowledge about the total costs and how to be profitable in the end, so information about the costs for the possible solutions is considered as well.

In order to show that the design of the solution really satisfied the needs, a small proof of concept was made. For this proof of concept a cloud platform had to be chosen on which the application should run. In order to choose a well suited platform, there were a lot of mail conversations and phone calls with providers. In the end it wasn't necessary to run the proof of concept at the side of a cloud service provider.

1.4 Thesis outline

This chapter has given some information about the relevance of the topic, research questions and the approach. The main topic will be explained in chapter 2. In this chapter the concept of cloud computing will be explained. In chapter 3 there will

Part	Chapter	Title	Sub questions	Description
	1	Introduction	-	Introducing the thesis
I	2	Cloud Computing	1	Literature study
	3	Security	3	Relevant security issues
	4	Legislation	4	Relevant legislation
	5	Platforms	2	Existing platforms
II	6	Cloud Solutions	1, 2	Possible solutions
	7	Business model	1, 2	Business models for cloud computing
III	8	Current Situation	5	Current product of Aia Software
	9	ITP in the Cloud	6, 7	Aia Software's product in the cloud
	10	Proof of Concept	6	Small proof of concept of the architecture
IV	11	Conclusion	All	Main conclusion
	12	Future Work	-	Future research
	13	Academic Reflection	-	Reflection on the product and process

Table 1: Thesis outline

be some further research about security aspects in combination with cloud computing. Questions like "how can it be kept secure?" and "what are new threats in cloud based environment?" are being answered in this chapter. The next chapter (chapter 4) provides more information about the relevant legislation for an organization that wants to start using (or offering) cloud computing. Chapter 5 gives an overview of the existing platforms that are available and their advantages and disadvantages. Chapter 6 continues with an overview of possible models and solutions for implementing cloud services. This chapter will describe the advantages and disadvantages of the different solutions. Business models for offering cloud services are provided in chapter 7. In chapter 8 the current situation within Aia Software will be described. After all the information in the previous chapters is presented, in chapter 9 this information will be used in a case study. This case study will be the situation within Aia Software. In this chapter an advice will be given about the best way for Aia Software to put their product in the cloud. A proof of concept for this solution is created and presented in 10. The conclusion and the future work are written in chapter 11 and 12. Finally, chapter 13 provides a reflection on the process and the product.

Part I

General Information

2 Cloud computing

2.1 Definition

The name cloud computing refers to the images of clouds that are representing networks and the Internet in most drawings. Basically, cloud computing makes data and applications available through the Internet. By doing this, data and applications can be accessed from everywhere. Cloud computing is not a new technology or a new device; it is a new way of using existing technology and devices. It is hard to find a clear definition of cloud computing but the following definition by Forrester Research is helpful because it contains all the elements that are commonly associated with cloud computing:

> Cloud Computing: *"A standardized IT capability (services, software or infrastructure) delivered via Internet technologies in a pay-per-use, self-service way"* [13].

With cloud computing it becomes easier to access data with several devices. Especially for mobile devices this can be really useful since the only thing that is needed, is an Internet connection. In figure 1 a diagram is shown of a cloud based solution.

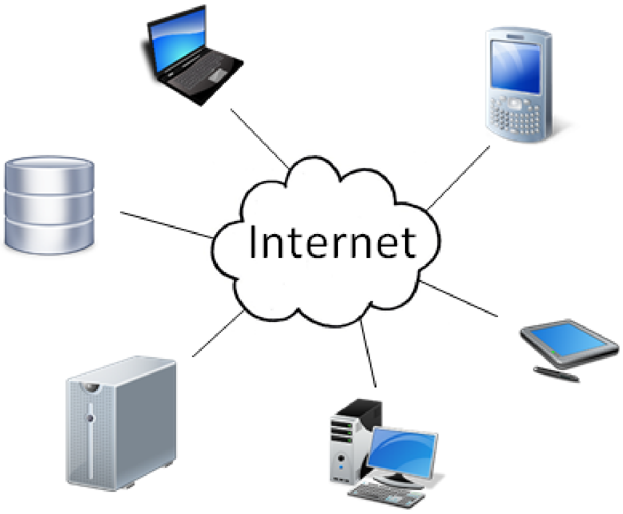

Figure 1: Cloud computing diagram

This figure shows different devices like notebooks, desktops, smartphones, tablets, servers and databases that are connected to the Internet. Storing data and running applications will be done in such a way that they can be used by devices that are connected to the Internet. An important aspect (for my research as well) that is not mentioned in this definition is that cloud computing is supposed to provide scalability and elasticity. Forrester Research mentions three major trends that gave birth to cloud computing:

1. Industrialized IT, through increased commoditization, standardization, consolidation and globalization. So, the same software is used by more and more users/organizations that are spread all over the world.

2. Tech populism, a popular culture where technology is consumed by and focused on end consumers, and is no longer the exclusive terrain of organizational buyers. This means that not only organizations are buying the technology, but nowadays a lot of people are buying/using software for personal use.

3. Technology that is embodied in the business or business technology instead of information technology. This means that business has more knowledge about the technology that is used inside their organization than that IT people can have. The business has knowledge about the processes within an organization and therefore they should be able to implement some things instead of the IT people who don't have this knowledge.

Cloud computing was pushed into the real world because of the pressure on IT to save money [26, 38, 50]. In the past, maintenance costs have kept on rising with the result that the budget for innovation decreased since the total budget didn't change.

Cloud computing will change the role of decision makers in a way that might not be expected. The cloud should be used to outperform the competition. The biggest impact for the provider of cloud computing is not to increase time to market and agility, which are important goals for an organization (for the consumer of cloud services as well), but the bigger impact for the provider is that it transforms the business models. So an organization that wants to deliver cloud services as well should consider another business model because it will not be selling products any more, but it will be selling a service, which requires another business model. Today's economy has changed from a pattern of predictable cycles to a situation where there are high and low peaks. Nicholas Carr states in his book The Big Switch that in the coming decade or so, corporate IT will be more or less switched off in favor of cloud resources. Brian Garvey even states that 2012 will be the year of the cloud [36].

As already mentioned, another effect of cloud computing is that products become services. This is something that has to be taken into account for the relationship between a vendor and a client. Since the product has become a service, it is easy for the client to stop using this service and switch to another vendor. Cloud computing is not a goal in itself; it helps a company to become part of the current ecosystem. An opinion that is shared by a lot of people is that when an organization ignores cloud computing, it will not get the business and when the organization is not maintaining good relationships with its clients, it will lose market share. So it does not make sense to resist against the shift to this new kind of ecosystem because it is inevitable. Cloud computing won't go away anymore; it is the future [26, 36].

The shift from products becoming services results in the distinction between ownership and usage of IT assets. When an organization delivers cloud solutions it has an advantage because they can realize economies of scale and standardize products. Nevertheless it also introduces some questions for the client [30, 26]:

- How to guarantee security and reliability?
- How to integrate the external and internal assets into one usable solution?
- How are back-ups being organized?
- How is the security organized?
- How is privacy organized?
- Is the data stored in a private or a public cloud?
- Where is the data stored?
- Who is the owner of the data once it is stored?
- Will there be access to the data when the provider goes bankrupt?
- What is the up-time and availability of the service?
- What are the costs of expanding capacity and functionality?
- What are the advantages of moving to a cloud solution?

Furthermore it is important to start with a process oriented vision in order to create the right processes for quick responses while preventing future spaghetti resulting from a series of quick solutions. This means that the business processes within an organization, instead of the technological possibilities, should be guiding the design of a cloud infrastructure. By doing this, the infrastructure will improve the business processes of which the entire organization can benefit. So it is important to have some clear reasons, other than because it's technically possible, why a cloud solution should be used. Furthermore it is important to create a governance and enterprise architecture that is created from the perspective of best possible

customer service instead of from the perspective of risk management. [26]
Another aspect to which an organization has to pay attention, is how it is going to earn money. It can be via a pay-per-use model or based on fixed price per time slot. For a software vendor the cloud model brings some advantages:

1. It creates a new way to sell its software
2. It creates a constant revenue stream
3. It offers new ways to lock users into a platform or data format

Some vendors are also scared that their profit margins will decrease and that they will provide an entry to the market for new competitors. It is important to look to the future because using the cloud model only for short term advantages will, eventually, cost an organization more than it will return. Since there is no such thing as one cloud it is difficult to implement the cloud model. When people are talking about “the cloud”, this most often refers to a solution in one particular situation. Since those separate solutions can have their own implementations, it can be hard to combine these implementations. It is not enough to deliver cloud services only. It is also important to rethink the business model when starting to deliver these services. When this is not done there will be the risk of losing money. So, an organization should really have a clear vision on how they are going to earn money in this situation. [26]

2.2 History

In the history of IT first there were mainframes (1960s). In the 1980s the client server model was introduced. Today cloud computing really seems to be breaking through.
When looking from the point where there was Internet, there first was access to the Internet only, which was made available via an ISP (Internet Service Provider). After some time there was access to servers via this Internet. After this racks for equipment that could be accessed via the Internet arrived. All the previous was made possible via an ISP. After this stage the ASPs (Application Service Providers) arrived. They provided the possibility to host applications on a server and access it via the Internet. Nowadays the cloud has arrived which offers an infrastructure for hosting applications and data. Although ASPs and CSPs (Cloud Service Providers) look very similar, there are some differences. For example ASPs were less focused on multi-tenancy, which means that multiple customers working with the same system or application was not supported. Another difference is that cloud applications are designed for web usage while ASP applications often had nothing more than a simple web interface attached to a large client-server application.

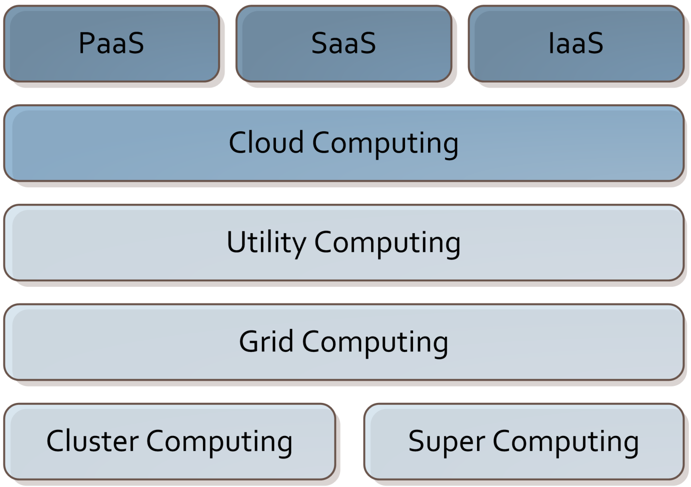

Figure 2: Cloud Computing Resources [25]

Cloud computing shares characteristics with some other forms of computing which is shown in figure 2. Jenkins [25] mentions that cloud computing is the commercial version of utility computing of which it adopts the concept [22]. Utility computing is the packaging of computing resources such as computation, storage and services as a metered service. This means that the equipment is basically rented. Utility computing has characteristics of grid computing. Grid computing refers to the technology in which computational power of different domains, that are loosely coupled, is combined to reach a common goal. Grid computing at its turn has characteristics of both cluster computing and supercomputing. Cluster computing refers to the technology in which computer clusters are created. The computers in these clusters are loosely coupled (but stronger than with grid computing) and work together as a single system to perform some computational operations. Supercomputing refers to usage of supercomputers. Supercomputers are computers that are at the front-line of current processing capacity. Those computers are used for highly calculation-intensive tasks.

2.3 Future

Since cloud computing is nowadays very trendy, it seems logical to invest in this technology. Nevertheless it is wise to take a look in the future and how it is expected to develop. It is somewhat hard to predict but the common opinion about cloud computing is that its usage will increase enormously in the coming years, especially when IT budgets have to be cut. For example Cisco predicts an increase of a factor 12 in the traffic for data centers between 2010 and 2015 [53]. Forrester expects an increase in the total cloud computing market of $40.7 billion in 2011 to over $241 billion in 2020. Gartner Research expects cloud computing to be a $150 billion business by 2014 and according to AMI partners, small and medium businesses are expected to spend over $100 billion on cloud computing by 2014 [17].

The next step will be that it doesn't matter on which hardware a specific cloud is running. Like the SETI (Search for Extraterrestrial Intelligence) project [57] the computational power of other clouds can be used when it isn't needed for that cloud. It is possible for people to subscribe to the SETI project. This subscription means that when the subscriber doesn't use his computer, the computational power can be used for this project. Since cloud computing is just getting interesting at this moment, it will take a lot of years before this new situation, in which overcapacity can be used by other CSPs, can be reached.

2.4 Strengths

As already said, cloud computing is supposed to give many advantages to software vendors as well as clients. For the clients a very obvious advantage is that they do not have to buy the hardware and software anymore since it is no longer a product but a service. Another advantage for a client is that it is in a lot of cases (but definitely not all), much easier to switch because he is just using a service, which means he did not buy any expensive software or hardware equipment that has to be recovered first. A software vendor can standardize a specific service and therefore realize an economy of scale. When this situation is reached, the vendor will be able to reduce its operating costs which can result in a lower service cost. This is something that cannot be reached when the internal IT department of the client has to do all this work. In the current situation about two-thirds of the corporate IT budget goes to routine support and maintenance activities [26]. By implementing a cloud strategy an organization will be able to work more efficiently as well. Employees will no longer have to wait for their colleagues to send information since the data is accessible by all the employees from any place as long as there is an Internet connection. Furthermore, employees don't have to wait for their internal IT department to provide them with extra resources.

Cloud computing furthermore results in the fact that the IT department does not have to provide the end-users with resources because the provision of those resources will be done (automatically) by the cloud service provider. It will be easier to change an organization's infrastructure because there is no privately owned hardware involved. A cloud solution is also device and location independent because everything is running online. Because of this it is possible to use those applications on any device that has Internet access. By using virtualization it is very easy to share servers and storage. By doing this the equipment will be used more efficiently. Nowadays the average capacity utilization of a server is about

10-30% [17]. Buyers of cloud based solutions don't have to consider their peak-load because there is more than enough capacity in the cloud. Once the peak is over, the resources can be scaled down again. When an organization buys its own hardware they will have to consider whether there should be capacity for the peak or that there will be some underprovisioning during the peaks (which results in users that are not being served) [11]. In the cloud, applications can run on multiple servers so this will increase the reliability of the service as well. Furthermore it is easier to maintain applications in the cloud since there is no need to install it on every single computer or server within a specific organization.

2.5 Weaknesses

Although cloud computing has many advantages, there are also some problems that might arise while implementing a cloud based solution. While designing an architecture for a product that has to live in the cloud, attention has to be paid to these problems and how they could be solved. Since the client has the possibility to switch to another vendor easily (only when no provider specific technology is used), this will result in a risk for the software vendor. The software vendor has to make sure that there is a good relationship with the client.

When data will be stored outside the company, for the customer there will also be a serious risk that the data gets into the hands of the wrong people, so this is something that should be avoided. This is especially important for cloud computing since the resources may be shared by several clients. A CSP (Cloud Service Provider) also has access to the data so for a CSP it is really important to gain some trust from its customers. Privacy becomes an even bigger risk when data will be stored on several systems.

Another barrier consists of legislation [27]. Especially the European regulations are very strict on privacy. Sometimes the regulations are even unclear, this happens in particular when country borders are crossed. There are several rules that state something about the physical location of the data that has to be stored or processed. Rules in different countries might be as diverse that it is needed to implement a, more costly, hybrid solution. For an organization that wants to buy a cloud solution in order to offer its own software (running on that solution) as a cloud service, legislation will be even more complex. Such an organization has to comply with the legislation that applies to itself, and the legislation that applies to its customers since creating a cloud service that cannot be used by its customers is useless.

As already said, looking with a long term perspective is really important as well. For example, what happens when a service provider goes bankrupt? What will happen with the data? Or what will happen if a provider merges with an organization's competitor? Since it is very hard to predict what will happen, this is a

serious risk for a cloud customer.
The main risk for the cloud service user, as described by Pearson [12], is that he is being forced to give personal information against his will. For the organization that is using the cloud service the main risk is the non-compliance to policies, legislation and loss of reputation. The main risk for the implementers (the people that develop applications for a platform) of cloud platforms is the exposure of sensitive information on the platform, loss of reputation and a lack of user trust. For providers of applications on top of cloud platforms the main risk is the legal non-compliance, loss of reputation and the usage of personal information in a way that was not intended originally. For the data subject (the subject of which the data contains information) the main risk is the exposure of personal information.
An important thing to have in mind is that the cloud will never be 100% reliable and secure but this is not different from the traditional IT products since software can be very complex and components of different producers have to work together. All the software that is used right now can crash or the support can be stopped. Some people say that this is not the biggest problem. The major consideration should be the contingency plan [4].
Most CSPs offer a SLA (Service Level Agreement) of 99.95% [51, 55]. This SLA will be enough for small and medium sized organizations, but will be insufficient for mission critical applications for large organizations. Although cloud computing is expected to reduce the costs for a company, this might be different when companies grow larger. McKinsey Consulting found that a typical data center of a large organization can run at lower costs than when it would run in the cloud [17]. The reason for this should be that cloud service providers overcharge large companies for their services.
Another uncertainty arises when more companies are using cloud computing. How will the service providers react to this? How are they going to compete with each other [1]? Are they prepared to take more risks in order to keep the prices as low as possible? Since business will rely on the cloud services this will be an important aspect, especially when there is a lock-in effect with a specific cloud service provider because of provider-specific technology. When common technology is used, and no special services are offered, this lock-in effect will not exist and the customer of a cloud service can switch very easily to another provider.

2.6 Success factors

According to the Business Software Alliance (BSA) there are some important factors that an organization needs to have in mind in order to let cloud computing be a success in general [2].

Ensuring privacy: In order to introduce a cloud successfully, an organization must be able to move data freely around the cloud while at the same time giving users the faith that their data will not be disclosed.

Promoting security: Since privacy is very important for users, an organization has to show that it understands and manages the risks that cloud computing brings along.

Battling cybercrime: There must be a clear set of laws concerning cloud computing. This is something that can be hardly influenced by an organization but since there are some rules, they have to be taken into account.

Protecting intellectual property: In addition to the previous point, there have to be some clear laws and enforcement against misappropriation and infringement of developments that underlie the cloud. In every country there are different laws about this topic [49].

Ensuring portability/harmonization of international rules: Since cloud offers data availability around the world, it would be really nice to have some kind of standard for the data exchange.

Promoting free trade: A cloud becomes really useful when data can flow freely around the world so there must be as few barriers for free trade as possible.

Establishing the necessary IT infrastructure: For the cloud to work, users need to have a broadband connection so there must be some promotion for getting the infrastructure that can offer this.

2.7 Service models

Service providers deliver their services according to three fundamental models: Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS) [25]. IaaS is supposed to be the most basic version and when moving to the other versions, an organization, which is buying the solution, has to be concerned about fewer details. So with an IaaS solution an organization has to consider operating systems and applications and in SaaS they can just consume the service of the software. Figure 3 shows how the different service models influence the efficiency of the resources and how it affects the control and costs for the cloud user. Since the efficiency increases when moving from standalone servers to a SaaS solution, the costs can decrease because less hardware is needed.

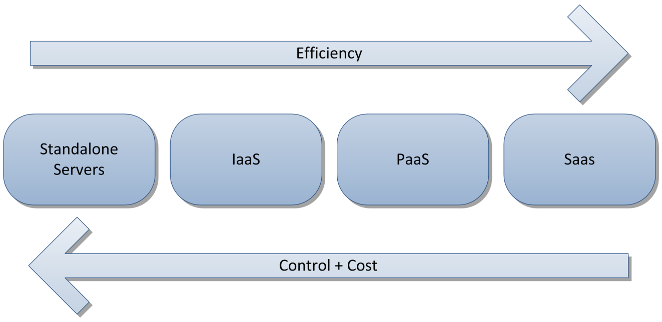

Figure 3: Efficiency vs. Control

2.7.1 Infrastructure as a Service

IaaS is the most basic service model since it offers a total infrastructure. This includes processing power and storage capacity. Sometimes firewalls and load balancers are offered but this functionality often belongs to the PaaS model. Providers with this service model have large amounts of storage and processing power which they supply on demand. The customers of these providers can install some operating system images as well as applications. By doing this, the customer has a lot of freedom to create his own environment. The downside of this is that basically only network architects can work with it since a complete network has to be set up. In this situation the CSP often owns all the equipment and the operating systems and applications will run all in the cloud. This has as an advantage that when the hardware at the client's side breaks or gets stolen, it is easy to replace it. The client will just have to buy some new hardware to access the cloud and there he can find all the data. Mark Russinovich (Microsoft) explains the IaaS solution as servers on demand. An IaaS solution is the only service model that delivers completely platform independent resources to its users [56].

2.7.2 Platform as a Service

In this model providers offer a computing platform and/or a solution stack. This includes operating systems, programming language execution environments, databases and web servers. In this situation the customer does not have to pay attention to load balancing, firewalls and so on. The customer still has to develop its own applications which than can be run on the platform. This means that this solution will be suitable for developers. Mark Russinovich explains PaaS as on demand application hosting [56]. This solution brings the risk of the lock-in effect. Since applications are being developed for a specific platform, it might be the case that those applications will run on this platform only. When this happens it will be hard to switch from provider because in that situation it will be necessary to develop existing applications for the new platform again.

2.7.3 Software as a Service

In the SaaS model CSPs install application software that can be accessed via cloud clients. The customer does not have to manage the infrastructure or platform. The difference between a cloud application and a "normal" application is the elasticity. Cloud applications are more scalable. Furthermore the CSP will do the updates, back-ups and maintenance so the client does not have to worry about this. For the provider of the application it is easier to maintain because it doesn't have to be done for every customer separately. The SaaS solution is explained by Mark Russinovich as applications on demand [56]. Since the SaaS solution offers applications, the end user will be able to work with this immediately.

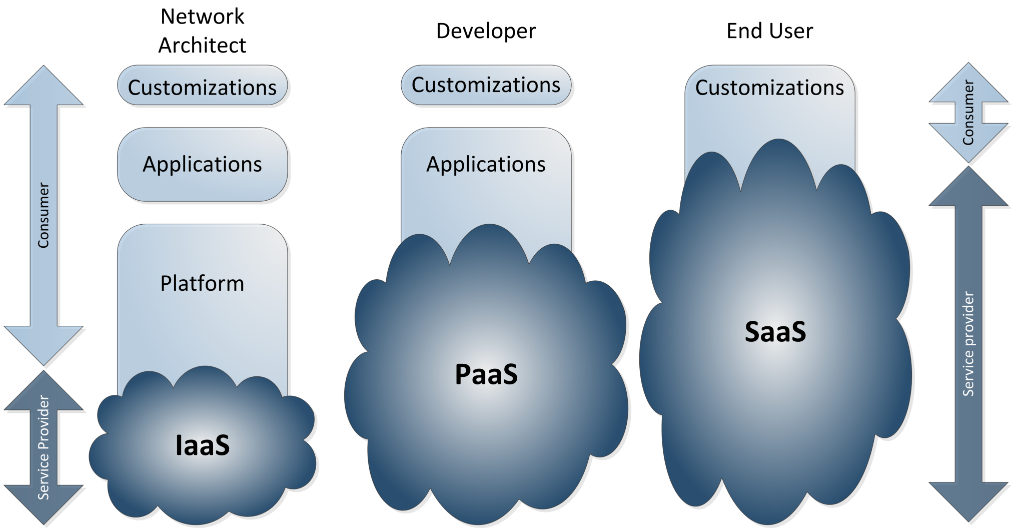

Figure 4: Overview of the service models

	IaaS	**PaaS**	**SaaS**
User	Network architect	Developer	End user
Advantage	All control	No concerns operating systems	Efficient hardware use
Down-side	Less efficiency	Risk of lock-in	Less control

Table 2: Comparison of service models

2.8 Deployment models

There are different deployment models for implementing a cloud based solution. For a customer of a CSP there are several reasons to choose for a particular type of cloud.

2.8.1 Public cloud

In a public cloud, applications and resources are made available to the public. Services in a public cloud will be accessed through the Internet only. 22% of the government agencies selects this option [39], now or in the near future. Contrary to what a lot of people think, it is possible to store personal data in the cloud. When this is wanted the client should talk to the cloud service provider in order to figure out which security measures have to be added [46]. This is called a gap analysis. During the Oracle Cloud Conference [28] it was told that an organization should use this deployment model unless it does not meet the requirements. The advantages of the public cloud are that the customer does not have to buy any equipment and that the resources can be shared among different customers. The result of this is that IT will become a bit greener and the operation cost can go down because the equipment is used more efficiently. The downside of this model is that an organization gets less control over its hardware.

2.8.2 Private cloud

A private cloud has an infrastructure that is used by only one organization. By implementing this model an organization can choose to buy or rent the equipment that is needed because this equipment cannot be used by other parties. 30% of the government agencies selects this model for implementing a cloud based solution [39] (now or in the near future). The hardware for the private cloud can be located on-premise or in a data center of the vendor. The advantage of this model is that an organization has total control over the hardware and that the organization has to worry less about the security since no other organizations are using the hardware. An organization can choose to buy the hardware itself or use a CSP (Cloud Service Provider) instead. The downside of not using a CSP is that the hardware still has to be bought which results in higher costs, but the CSP can still offer some scalability advantages. Furthermore the buying of the hardware results in the fact that there will still be a limit on computational power and storage. So this means that the organization still has to buy those resources based on the peak-load.

2.8.3 Hybrid cloud

A hybrid cloud consists of a combination of two or more other types of clouds. By implementing such a cloud, an organization can benefit from the advantages of both clouds. 25% of the government agencies currently (or in the near future) selects for the hybrid cloud solution [39]. The advantage of this is that an organization can benefit from the advantages of both public and private clouds. For example, an organization can keep its business critical information inside the organization and run other processes outside the organization. Another possibility is that an organization runs all the applications in its private cloud and only uses the public cloud when the private cloud lacks for example computational power. The downside of this model is that the efficiency will be somewhat lower than the efficiency of public cloud.

2.8.4 Community cloud

Community clouds share the infrastructure between organizations in the same community. The costs are spread over fewer parties than in a public cloud but over more parties than in a private cloud. So this is basically a private cloud for a community. This means that the advantages and disadvantages of the private cloud model are spread over a larger community.

Cloud	**Advantages**	**Disadvantages**
Public	- Efficient use of hardware - No need to buy hardware	- Data is stored off-premise
Private	- Control over hardware - Control over data - High costs	- Hardware for peak loads - Hardware has to be bought or leased (less efficiently)
Hybrid	- Business critical information can stay on premise	- Less efficiency than a public solution
Community	- Costs can be spread - More efficient use of hardware	- Hardware has to be bought or leased (less efficiently) - Less efficient than a public solution

Table 3: Comparison of deployment models

3 Security

Security is one of the terms that is very often mentioned when talking about cloud computing. A lot of people think of insecure use of software when they hear the term cloud computing [9]. This chapter provides an explanation about new threats and how these can be handled. Most people are scared to start using cloud services because they think it is not 100% secure. For these people it is important to realize that the current IT systems also have performance failures and data leaks. Since large IT systems are very complex, it is almost impossible to get a proof that a system is 100% secure. There are quite some news items nowadays, in which there is a story about data leaks or IT systems that could not be used for quite some time. So, it is important to research how much cloud computing differs from the existing systems at this point. In fact, cloud systems are often even more secure than on-premise systems since back-up and failover facilities are organized much better. Of course this has to be the case since the consequences of bad security are much worse when something goes wrong.
The most important thing with security is that an organization uses the "security by design" principle. This means that security is implemented in the design of an application. Once the design is made without having security in mind, it will be hard to create a solution that is secure.

3.1 Cloud computing threats and risks

3.1.1 Top threats by the CSA

The Cloud Security Alliance (CSA) has identified top threats that cloud computing brings along. They have created a list with the seven top threats. This section is based on research of the CSA [34].

Abuse and nefarious use of cloud computing Cloud service providers often offer their customers unlimited processing power, network capacity and storage capacity. Together with the fact that it is very easy to register for those services (often anybody with a valid credit card can register), it can end up in an unwanted situation because there is a high degree of anonymity. Because of this anonymity it is easy to abuse all the available computing power. One can use this computing power for example for password and key cracking, captcha solving, etc. The result of this abuse is that entire blocks of network addresses get blocked which will result in innocent addresses being blocked as well. This threat is not applicable for the SaaS model since an organization cannot develop new applications. It will be hard to determine when some kind of behavior has to be classified as abuse. This means that it will be difficult to prevent this.

Insecure interfaces and APIs Cloud computing providers often offer a set of interfaces or APIs to their customers for interaction with the cloud services. The security of the cloud services is dependent on the security of those APIs.

Malicious insiders This threat is not specific for cloud computing but cloud computing amplifies it because of the convergence of the IT services. Since data of different customers are combined on the same servers, the risk of a malicious insider increases. Because of the lack of transparency at the service provider (especially into the hiring of people) the customer has to trust the provider that he doesn't hire "bad" people. This could be prevented by monitoring but this is in conflict with the privacy of the users. Certificates of good conduct and certification of personnel could also solve a part of the problem but there will still be a small risk. This problem also exists in the on-premise situation, nevertheless in the on-premise situation there was slightly more information about the personnel.

Shared technology issues This threat only applies to IaaS providers. IaaS providers often share their resources for customers in order to let it be scalable. This might result in some problems since most components were not designed to offer strong isolation properties for a multi-tenant architecture. So when this is not taken into account, it might happen that some data is leaked to other customers or people. Existing virtualization software is intended to solve this problem so this risk will particularly exist when an organization develops its own cloud.

Data loss or leakage Since data is stored in the cloud the CSP can be responsible for back-ups of the running system in case its hardware fails. Furthermore the loss of an encoding key may result in data becoming unusable. This is not cloud specific but especially in the cloud, stored data should be encrypted. In order to prevent data leakage, there should be an authorization part in front of the application that checks permissions for the specific application.

Account or service hijacking This is not a new type of threat since attacking methods like phishing, fraud and bug exploitation exist already for quite some time. Nevertheless cloud solutions add some new threat: the attacker can for example eavesdrop someone's activities or use someone's account as base for other attacks.

Unknown risk profile Cloud computing has several advantages to the business. The business does not have to buy and maintain hardware so there are some financial and operational benefits. Nevertheless those benefits have to be weighed against the security concerns. Attention should be paid to the risks that the cloud solution brings along and the impact that it can have for the business. This means that if there is no clear overview of the possible risks, the lack of this overview will be the biggest vulnerability. Security through obscurity (using secrecy of design or implementation to provide security) might be the easiest way but this can result in some unknown exploitations. In fact security by obscurity is never a good idea.

3.1.2 Cloud computing risks by Gartner

The CSA is not the only group that has identified some security risks. Gartner has published seven specific security issues that customers should raise with vendors before selecting a cloud vendor [40].

Privileged user access Since sensitive data is stored outside the organization, there is much less control about who can access the data. For a customer it is important to gain as much information as possible about the people that have access to the data. It might be a good idea to ask information about the hiring process and the oversight of people that have access to the data. This risk also exists when talking about outsourcing. Nevertheless, the potential group of people it much larger when talking about cloud computing because the data can be accessed via the Internet.

Regulatory compliance Customers are responsible for the security of their own data, even when the data is stored at a cloud service provider. Those providers are subject to audits and certifications so when a provider refuses to undergo this process, according to Gartner, he might be signaling that he can be used for only the most trivial functions.

Data location Since the cloud is not located at a single location, it can be unclear where the data of a customer will be stored. In some cases it is even unclear in which country the data will be stored. For a customer it can be important to ask the provider whether he can store the data in a specific jurisdiction.

Data segregation Data in the cloud is typically located in a shared environment. This means that data will be stored next to data of other customers. This data should be separated and invisible for other customers. Encryption is only a part of the solution since it doesn't cure everything. For example, encryption does not prevent data deletion. Furthermore an encryption mistake can make data useless. Therefore it is important to ask a provider how the encryption process is organized and which other activities are undertaken to keep the data separated.

Recovery Although every provider will tell how great their uptime is, it is important to ask how data replication is organized when there are failures in the system. Once data is not replicated it might result in a total failure of the system. A customer should ask the provider whether it is possible to do a complete restore operation and how much time it will take to complete.

Investigative support According to Gartner, investigating inappropriate or illegal activity might be impossible in cloud computing. The reason for this is that logs and data of multiple customers can be co-located or can be spread over an ever changing set of hosts.

Long-term viability When choosing for a cloud service provider, this provider is not expected to go bankrupt or get acquired by a larger company. Nevertheless this might happen and therefore a customer should ask what happens with his data in this situation. He should ask whether the data remains available and which format will be used to export the data from the cloud. This format is important for the ability to import the data in a replacement application.

3.2 CIA Analysis

Important security aspects when talking about information security are confidentiality, integrity and availability [24]. Since users do no longer physically possess their data, the confidentiality, integrity and availability can be at risk. There are some well-known solutions for these problems, but in the cloud many of these solutions don't seem to be very practical. In the cloud it is even more important than in on-premise solutions that all the security measures will incur as less as possible overhead.

3.2.1 Confidentiality

Confidentiality is the term that refers to the prevention of data disclosure to unauthorized people. Confidentiality is necessary (but not sufficient) for the privacy of people that have data in the system. For a cloud service provider it is really important to guarantee confidentiality to their customers since a lot of customers might be running with the same resources. Attention has to be paid to (unexpected) data leakage since there also might be some temporary files that are stored in for example some cache or shared file system.

The simplest way for a user to protect its data privacy is by using encryption. When a modern encryption standard is used, nobody else will be able to decrypt the data. Nevertheless, because of performance reasons this doesn't seem to be a very practical solution. It is for example difficult to search for keywords in encrypted data. Downloading all the data and decrypting it locally will generate too much Internet traffic. Storing data in the cloud is almost useless if people cannot search and utilize that data [9].

3.2.2 Integrity

Integrity refers to the fact that data cannot be modified by unauthorized people. An organization doesn't want customers to see data of other customers but it is also not wanted that they can alter data from other customers. Data might be stored on the same storage devices so there has to be a solution for setting access rights in order to prevent people from altering data that does not belong to them. For integrity it is again important to notice that existing encryption techniques will be impractical when it comes to checking integrity. For these techniques it is often required to have a local copy of the data and this local copy isn't available anymore since all the data will be stored in the cloud.

3.2.3 Availability

Availability refers to the time that a system will be available for usage. Nowadays there are a lot of systems that have to be available 24/7. This requires a security model that guarantees this. Availability is very important in cloud based solutions since organizations want to use the software whenever they need it. This results in the fact that it is unacceptable to have some downtime of the system when updating to a new version of the software. Those updates should also be done in such a way that the client doesn't have to change anything in their own software because this probably will result in downtime of the system for that client. In order to guarantee availability, most cloud service providers will set up a Service Level Agreement (SLA). Nevertheless these SLAs are not always meaningful. For example a provider can guarantee an availability of 99.999% and give a discount of 10% when this is not reached. Since the infrastructure is not designed to reach this availability [17], the provider is actually offering a 10% discount in exchange for the benefit of claiming that SLA. So as a customer it is important to pay close attention to the details of the SLA. It is important to weigh the benefits against the impact on the business when the SLA is not met [6]. Another aspect that might cause some problems for availability is that a lot of CSPs only offer email or web-based support. For a cloud service provider it is really important to match the performance of the service to the level of performance and reliability to which the customer got used to when maintaining their own data center.

Another potential risk for cloud computing is a denial of service (DoS) attack. Although the cloud in theory has unlimited resources, it can still be exhausted [3].

Once the cloud gets exhausted the problems are much bigger compared to the on-premise situation because much more customers are using the same instance.

3.3 Privacy

Privacy is an important issue within the area of cloud computing [12]. Combining privacy and multi-tenancy is one of the most critical and difficult challenges for the public cloud. Two aspects of privacy are especially important to have in mind when designing a cloud system: legal compliance and user trust. As with security it is important to consider the privacy right from the start instead of trying to implement it in a later stage (security by design). Since privacy is a human right, there is some legislation about this topic about what is allowed and what is not. More information about the legislation can be found in chapter 4. The term privacy sensitive data might be used differently by different people. Pearson [12] describes this term as information that includes the following:

Personally identifiable information: information (e.g. name, address) that could be used to identify or locate a person or information (e.g. credit card number, IP address) that can be combined with other information to identify a person.

Sensitive information: information like someone's health situation, sexual background, financial information or job performance. This kind of information needs extra safeguards.

Usage data: data which is collected from computer devices, behavioral information about which contents are viewed and which products are used.

Unique device identities: other types of information that might be uniquely traceable to a user device.

It is not only the personal data that is worth protection. Behavioral information should also be protected. Since resources are shared among users, information about the usage of one user can be leaked to another user. With this information someone could reverse-engineer for example data about the customer-base or revenue size. This can be done by, for example, monitoring the CPU usage or the memory usage. When one user is using some computational power of the cloud, this computational power cannot be assigned to another user.

With shared resources there will also be the risk that someone who shares resources with a user who does malicious or unethical things, has the risk that he gets blamed for it. Even when the person or organization can prove that it was someone else who did the malicious things, the reputation is already damaged [23].

3.3.1 Privacy requirements

Pearson [12] mentions some key privacy principles that have to be met:

Notice, openness and transparency: during the collection of personal information it is important to tell users how their information is used. Once this way of using information is changed, the user should be notified about this. Privacy policies must be made available to the users.

Choice, consent and control: the user must have a choice about which information is collected. Data subjects must give their approval for collecting, using and disclosing their personal data.

Scope/minimization: only data that is required during the processes should be collected. The amount of data collected should be minimized.

Access and accuracy: users must have the ability to check which data is held and check its accuracy. All the personal information has to be accurate.

Security safeguards: safeguards must prevent unauthorized access, disclosure, copying, use and modification of personal data.

(Challenging) compliance: customers must be able to challenge the privacy process. All the operations that are performed on the data have to be compliant to privacy legislation.

Purpose: there must be a clear purpose for the collection of personal data. Data subjects should be told why their data is collected.

Limiting use - disclosure and retention: collected data should be used only for the purpose for which it was collected. Personal data should be stored as long as necessary.

Accountability: an organization must appoint someone who will be responsible for ensuring that privacy policies are followed.

3.4 Identity and access management

Identity and access management (IAM) seems to be a problem for many organizations [32]. When using cloud based software, it is important that the policy of the cloud service provider matches the policy of the organization. The policy for the cloud service should be of the same level or a higher level than the policy of the organization that wants to use the service. This means that the protocols of the customer and provider have to be adapted [8]. In the best possible situation the end-user can use the cloud service in a way that he was used to when he was using on-premise applications. This means that there will be a single log-in. Unfortunately this isn't always possible.

When using on-premise systems, authentication is often based on Windows Authentication. Once a user logs on to its computer, these credentials are used to get access to other applications and resources as well. Once the applications are running outside the organization, these credentials often cannot be used anymore. This means that in cloud applications, there has to be another mechanism for identifying and authorizing people. This might be even one of the most important issues when talking about the cloud. A customer wants to be completely sure that another organization cannot get access to its data. Therefore the authorization part has to be really important. Especially when separate installations of a software product are running on the same hardware, it is important to have a routing mechanism that routes customers to their own installation. This means that such a routing mechanism has to do the authorization part as well. Once someone wants to do something in the cloud, this person should be identified. Afterwards it should be checked whether or not this person is allowed to perform the action on the requested resources.

For a CSP, the most "easy" solution to handle IAM is by developing an authentication mechanism for itself. This often results in the fact that customers cannot use their existing user accounts. Because of this there will be no Single Sign-On (SSO) possibility. Since a cloud application is often used by a lot of users, it will be useful to create a role based access control mechanism. By creating such a mechanism it will be easy to change the permissions of a lot of users and set permissions of new users.

Especially in cloud computing it is important to use the principle of least privilege. This means that a user should not have more permissions than necessary. Since a cloud application is exposed over the Internet, it is important to keep the attack surface as small as possible. This means that only the necessary elements are exposed. It is also important to check permissions "at the gate" this means that the permissions of a user are checked immediately when the user arrives at the application. Besides this, every layer in an application should check permissions. Once functionality of a system isn't used, it should be disabled in order to prevent people from performing unexpected operations on the system. All these core security principles are not cloud specific, but since cloud applications are widely exposed, they become more and more important.

As already said, for a customer it is important to adapt the IAM practices in such a way that they can be used in the cloud as well. In fact this is something that should be done together with the CSP. The Cloud Security Alliance [33] mentions some important IAM aspects that need attention:

Identity provisioning: Organizations that use cloud services want to have a secure and timely management of the provisioning and de-provisioning of users in the cloud. Organizations that have invested in user management would like to extend this to cloud services.

Authentication: For an organization it is required to have a trustworthy manner of authenticating users. Organizations must address challenges such as credential management, strong authentication, delegated authentication and managing trust across all types of cloud services.

Federation: Federated Identity Management is concerned with having a common set of policies, practices and protocols to manage the identity and trust into users and devices across organizations. In a cloud environment this means that identity attributes have to be exchanged in a secure manner between the identity provider (chosen by the customer) and the service provider.

Authorization and user profile management: Once a user is authenticated, the access control has to be managed. Therefore a cloud environment has to establish trusted user profile and policy information. This information has to be used to control access within a cloud service. A customer must be able to audit this process.

Compliance: For customers who rely on cloud services, it is important to understand the manner in which identity management is performed. Once this is understood, an organization has the information whether it complies with the internal requirements.

For a cloud service it is not enough to have credentials for every user. A user should also have the right permissions to the applications. This information should be provided to the cloud service in order to get access. Not only user identities should be provided, but also applications should identify themselves (when an application is being called by another application). Using standards as the Service Provisioning Markup Language (SPML) can help to automate this process. Of course the information about identities and permissions has to be sent over a secure line. When an organization is using a combination of multiple cloud services that communicate with each other, it is preferred that it is not necessary to enter credentials for every system. A Single Sign-On (SSO) mechanism can help with this. This means that different services have to trust each other. An industry standard that can help with this is Security Assertion Markup Language (SAML). This is a message format that contains authentication data.

3.5 Other challenges

Besides the security challenges that are mentioned in the previous sections there are still some other challenges:

- Proof of ownership
- Assured data deletion
- Remote assessment of fault tolerance

Proof of ownership is concerned with who the owner of the data is. This is not just a legal issue. Cloud service providers often perform de-duplication. This means that the provider removes duplicates of the data, even if different users have added the data. Since in this situation there is only one copy left there might be some problem in telling who the owner will be. Assured data deletion is concerned about the "real" deletion of data. When a user thinks he has deleted some data, the data should be really deleted. For example the CSP should not keep and definitely not use a copy that is not visible for the user. The only situation in which this could occur is that the data still exists in a backup, nevertheless the user must be aware of the existence of this backup. This is especially important when it concerns personal or business critical data. An organization should and wants to know where that kind of data is stored. Remote assessment of fault tolerance is the remote detection of hard-drive failure vulnerabilities in the cloud. When there is some hardware failure in the cloud, this must be repaired as quickly as possible to guarantee the availability of the services.

Another aspect that changes for the customer is that their applications cannot trust the server anymore since the server is moved to another domain, unless certificates are used which is mostly done. In case of a private cloud that is hosted on-premise this doesn't hold but in any other case this will be the situation. Furthermore IaaS solutions often rely on virtualization of the software but the virtual operating systems also rely on the underlying system. This means that security issues in the underlying system will also have impact on the virtualized environment.

4 Legal Aspects

4.1 Legislation

Cloud computing is not that new as one might have expected, but it was only recently that it became popular. Because of this there is not that much legislation about this topic. Since cloud computing offers the possibility to access data and applications over the entire world, there are many laws of different countries involved. This is exactly the point where there might be a problem. Different countries have different laws that can be contradictory.

4.1.1 The Netherlands

In the Netherlands relevant privacy issues are regulated in the "Wet Bescherming Persoonsgegevens" (WBP). For this law there are two groups of people involved: the responsible people and the processors of the data. Based on this law it is likely that the customer is the responsible person and that the provider will be the processor. Another situation is that the provider is co-responsible for the data. When a cloud service provider uses the data for personal purposes, the provider must comply with the WBP. The WBP is applicable when personal data is being processed on behalf of activities of a Dutch establishment. When a provider is classified as a processor only, it is important for the customer to establish an agreement with the provider about how to protect personal data. When using a private cloud solution a customer will have a higher influence on establishing some agreement compared to the public solution, because in the public solution the provider wants to standardize as much as possible.

Sometimes it can be a bit unclear to which parties the data of a customer will be passed on. In principle data may be passed on to countries with a protection level that fits the requirements of the WBP. Organizations that follow the "Safe Harbor Principles" are considered to have a conforming protection level. More information about the Safe Harbor Principles is in section 4.1.2. This results in the fact that cloud service providers like Google, Microsoft and Amazon are accepted as provider, since they comply with the principles. When the protection level doesn't fit the requirements there is still a possibility. In this situation an organization should get a permit from the ministry of justice. When requesting such a permit, an organization should know exactly to which countries its data is being moved.

There is no regulation about what happens when a cloud service provider becomes insolvent. Nevertheless in 2009 a Dutch judge ruled that a cloud service provider had to deliver its service while being in suspension. There are not that many opportunities to guarantee the delivery of a cloud service, so it is wise to have

a solution that can run on different platforms and check the performance of the provider.
In The Netherlands there is a lot of (European) legislation applicable in which the government can, under special circumstances, get access to data that is stored within companies. Organizations also have to consider the USA PATRIOT Act which grants the US government access to the data. This act is applicable to companies that are located in the United States, have a parent company in the United States, have a subsidiary in the United States [27] or even do business in the United States on a regular basis [16]. The governments from different countries can be prevented from reading the data of a PaaS and IaaS platform without notification, by using state of the art encryption [15]. Although an organization should definitely pay attention to this, they should not overreact by definitely not using an American CSP. It is just a matter of weighing the consequences. When an organization wants to be completely sure that its data does not get into the hands of American authorities, the best way is to keep data in a private environment that is completely managed by themselves [5].

4.1.2 European Union

In the European Digital Agenda the European Union member states have agreed to offer financial support for the EU strategy for cloud computing. Since the current regulations lack some clear rules and contain some barriers, a draft version for a directive has been created. The current barriers are primary concerned with responsibility, location, transparency, control and privacy.

EU Data Protection Directive On 25 January 2012 the European Union has published a draft version of the data protection directive which contains some regulations about "personal data" in the cloud [20, 27]. In order to find out which data falls under the DPD regulations, there has to be a clear definition of what personal data is. The DPD defines 'personal data' as:

> "any information relating to an identified or identifiable natural person ('data subject'); an identifiable person is one who can be identified, directly or indirectly, in particular by reference to an identification number or to one or more factors specific to his physical, physiological, mental, economic, cultural or social identity."

EU data protection responsibilities are imposed primarily on the controller and processor of data [21]. Since cloud service providers are considered to be a processor or controller, the European regulations will apply to them. The Queen Mary research [21] suggests that there should change something at this point because especially IaaS and PaaS providers most often do not control or process data, they

only offer facilities and tools.
In order to apply the legislation it should be clear who the owner of the data is. Determining this is not that easy as it sounds [19]. Even with a private cloud this can be hard because a user is not necessarily the owner of the infrastructure that is used. The ownership of the data center can also be divided into different categories, e.g. the owner of the building, the owner of the hardware and the owner of the software. And again the people or organizations who own the components that are mentioned before, don't necessarily manage and operate the services.
Since it might happen that data has to be transferred outside the European Economic Area (EEA), a solution has to be found for transferring data outside the EEA [18]. The most important characteristics of the DPD are listed below [27]:

- It is applicable to all processors of data that have one or more locations within the European Union. It doesn't matter whether the data processing takes place outside the European Union.
- It is applicable to companies that try to reach citizens of the European Union directly.
- Improving the regulations that are concerned with the applicability of the rules and the transmission of data outside the EU.
- Clarification of the term "permission"
- Organizations are subject to only one privacy authority, i.e. the authority in the country of their headquarters.
- The obligation to notify involved people when data is leaked.
- The obligation to assign a Data Protection Officer in entities with more than 250 employees.
- Privacy by default. Consumers should have to make less effort to guarantee their privacy. In order to collect personal data, explicit permission is needed.
- Transparency and minimization. A customer must be able to ask which data is stored within a company. This data may not be stored longer than necessary.
- The right to be forgotten. A person must have the possibility to ask an organization to remove all his personal data.
- Control about personal data. A person must be able to access his personal data in order to check or change it.

In order to guide the data transmission between the European Union and the United States a Safe Harbor Agreement has been set up (section 4.1.2). Nevertheless, when a European organization is doing business with an American organization, it will be automatically subject to the Patriot Act. Since all member states of the European Union and the Euro-parliament have to agree on this directive it will still take 2 to 3 years before it can be implemented.

Safe Harbor Principles The United States do not offer sufficient privacy protection according to the European Directive. Therefore a Safe Harbor agreement was negotiated for enabling the data transfer between Europe and the United States. This agreement is only applicable to information transfer between those two parties. An organization that is located in the United States can adhere to those principles. When this is done the organization is expected to provide sufficient protection. This agreement is not applicable to the financial services sector which has its own forms of compliance. The Safe Harbor Principles include the following [14]:

- Individuals must be informed about the purpose for collection of personal data.

- An option to opt-out on the provision of personal data to external parties or use of personal data for purposes other than the original purpose.

- Transfers of personal data to third parties may occur only to organizations that have sufficient data protection principles.

- Reasonable security of the personal data.

- The data must be relevant for the purposes for which it is used.

- The data subject must have access to his information in order to correct and delete information.

- There must be effective means of enforcing these rules.

A common opinion is that these safe harbor principles are not sufficient as well. There is not enough control on the compliance to the principles. German organizations that want to transfer their data to the United States will have to check the compliance to the safe harbor principles actively [10] while other countries don't. The safe harbor principles will be replaced by the new version of the European Data Protection Directive.

4.1.3 Conflict European Union and United States

As already mentioned, organizations that are related to the United States are subject to the Patriot Act. This act gives the American government the possibility to recover that is being controlled by the organization. This is always the case when the organization has its headquarters in the United States, but also when an organization has an establishment in the US. According to the European legislation it is not allowed to store personal data in a cloud service that is provided by an American provider. This is because a European organization must always be able to guarantee that personal data is stored adequately protected against unauthorized access.

The result of the Patriot Act is that data which doesn't leave European grounds can still be subject to this act because it can be stored on the servers of an American CSP. The conflict in laws can result in some problems when the American government wants to have the data that is stored on the servers. The American government can sanction the provider because of not giving the data and the European Commission can give the provider a fine for handing over the data to the American government.

Another problem is that it is hard to find out when the American government wants to have information because the CSP has to keep this information confidential. This also results in the fact that the owner of the data most often doesn't even know when his data is handed over to the American government [47]. This is also the most important difference with European legislation on this point. In Europe there is also legislation that can force a company to hand over data but this cannot be done without intervention of a judge and the hearing of the suspect. So, when data is stored in the United States, a company is not allowed to notify the owner of the data. When the data is encrypted, it is very likely that the data will be useless unless the owner hands over the decryption key. When this is done the owner at least knows that his data is being requested by the US government. Since a CSP is not obligated to decrypt the data when the key is not in its possession, it can be only the owner that hands over the key [41, 37].

4.2 Liability

One of the biggest risks of cloud computing is that the service becomes (temporarily) unavailable. Once this happens, the customer can experience a lot of damage. Direct damage from the service being unavailable is often less bad than indirect damage. Indirect damage consists of things like loss of business or loss of reputation.

Cloud service providers often don't want to be liable for a lot of consequent damage. Most CSPs will set a limit on the amount of damage for which they will be liable. This amount is often not more than at most a few times the amount of the fee that has to be paid for the service. The best thing for a customer to do, is making use of facilities that are offered by a cloud service provider to help increase availability. Examples of those facilities are: resilience, failover and disaster recovery options. In addition to using these facilities, it is also important to make good arrangements with the cloud service provider. These arrangements should be written in some kind of contract in order to be legal.

5 Cloud platforms

Nowadays there are several cloud platforms available so it is not necessary for an organization to create its own cloud or buy hardware for it. In this chapter there will be a description of some existing platforms among which an organization can choose.

5.1 Windows Azure Platform

Windows Azure is a cloud platform from Microsoft. With this platform it is possible to build, host and scale web applications, therefore it is classified as "Platform as a Service". The Windows Azure Platform consists of three products:

- Windows Azure (an operating system providing scalable computing and storage facilities)
- SQL Azure (a cloud-based, scale-out version of SQL Server)
- Windows Azure AppFabric (a collection of services supporting applications both in the cloud and on premise)

There is a 3-month free trial available of this platform. [59]
Microsoft guarantees a monthly SLA of 99.95% for Windows Azure. This SLA is reached by using two instances of the application or components. Furthermore data that is stored on the Windows Azure system is replicated at least three times of which there are two replications in the same data center, so the system can handle multiple failures at the same time. Russinovich [56] mentioned in his presentation that someone doesn't get the idea of cloud computing when he has to get his service offline when doing an update. This is something that is not needed in Microsoft's solution. The multiple instances will be updated one at a time so there will always be some working instance.
A primary goal of the Windows Azure platform is to be a foundation on which ISVs can create SaaS applications. Nevertheless it is possible for on-premise applications to access Windows Azure storage and SQL Azure databases [31] as well. Microsoft offers a monthly up-time of 99.9% for SQL Azure.
One of the biggest limitations of the platform is that SQL Azure has a maximum database size of 10 Gigabytes. This problem will be solved by sharding. A database shard is a horizontal partition in a database [63]. By doing this, the different pieces can be placed on multiple servers which divides the load over multiple servers as well. SQL Azure databases can be managed using the existing tools with which a lot of people are already familiar. The compute-instances can be divided over several data centers all over the world. By choosing the data center, the area in

which the data is stored can be controlled as well. Each compute-instance is a virtual machine that separates it from other customers. These compute-instances are offered in three different flavors. The Web Role is meant for web based front-ends on IIS; The Worker Role is for background processes and the VM Role offers a container in which standard Windows applications can be executed. This is a virtual machine of Windows Server 2008 R2 in which almost everything is configurable. Nevertheless it is often not possible to copy a local application to the cloud without surprises. The application has to be able to handle the scaling mechanism of Windows Azure. Furthermore the payment is done based on the number of transactions, so it might be that the developers have some influence on the costs. It might for example be a good idea to move multiple transactions into one big transaction. The pricing of these different compute-instances is the same for all. The differences are purely technical. This means that the web role is based on IIS technology and therefore comparable with a web server, and the worker role does not have this IIS technology and is comparable to a regular windows service.

Most often the complete applications (including all roles) will be built in `C#` or VB.Net because applications are built for the Common Language Runtime from .Net. Nevertheless it is possible to use other languages for which a clr-compiler is available. Languages for which this requirement holds include `C++`, JavaScript, PHP, Python and Ruby.

In order to divide the load over different parts of the application, Microsoft offers a traffic manager together with Windows Azure. This traffic manager offers a choice for the way in which load balancing will be done. The methods among which can be chosen are: Performance, Failover and Round Robin. Performance can be selected when hosted services run in different geographical locations and when the closest service should be chosen. Failover can be chosen when there is a primary service that should be chosen for all traffic but when there should also be backup services. These services can be running in the same or in a different data center. Round Robin can be selected when the requests should be equally distributed over a set of hosted services. Again these services can be running in the same or in a different data center. More information about the load balancing in Windows Azure can be found in appendix A [54].

Windows Azure also provides auto scaling functionality. This functionality uses rules that are defined in a XML format. There are two possibilities to scale an application. An application can be scaled by setting specific times to scale, this solution is only applicable when the workload is predictable. The second possibility is to scale based on current resource usage. For the first solution Windows Azure offers constraint rules and for the second solution reactive rules are being offered. Another important aspect to note is that it is possible to scale-out and scale-up. Scaling-out means that there are additional instances created while

scaling-up means that the instances are being enlarged.
Windows Azure has in general a good performance [7]. Some experts on Tweakers.net indicate that Windows Azure offers the best total package[44]. When an organization has MSDN access they will also have access to all the tools that are needed or preferred for the development of applications. In general Microsoft offers only a public solution. Nevertheless for large customers there is the possibility of creating a private solution. These configurations will exist of at least 1000 systems.

5.2 Amazon EC2

Amazon Elastic Compute Cloud (EC2) is a central part of the cloud computing platform of Amazon. Among the three biggest providers (Microsoft, Google and Amazon), Amazon is the only provider that offers real platform independent resources. The reason for this is that EC2 is an IaaS solution where the other providers focus more on a PaaS solution. This means that a customer has to consider the configuration of the hardware and which operating system should run on top of that architecture. EC2 gives users the possibility to run their own applications on virtual computers [58]. The service level agreement commitment of Amazon EC2 is 99.95% on an annual basis. To reach this level running several instances of the applications is necessary. There is a free tier available for 12 months. Once the SLA (less than half a day per year of downtime) is not reached, the customer will get a 10 % discount.
In order to use this solution, the customer must select or create Amazon Machine Image (AMI). An AMI consists of a boot partition and data about the virtual machine. Since this technology is based on Xen-virtualization, these images are easy to create by the customer. In order to launch the virtual machines (instances) the customer has to select a region in which the instances must be hosted. In order to increase the availability of the instances, they can be divided over different availability zones. These availability zones are data centers that are completely independent from each other. Data traffic within an availability zone is free of charge. During the configuration the customer can select the number of instances that has to run.
When data has to be stored in the cloud there are several possibilities. The customer can choose for Elastic Block Storage (EBS) in order to save status information of the virtual machine when an instance is shut down. An alternative for EBS is using the S3 cloud service of Amazon.
Instead of On-Demand instances, a customer can also choose for “reserved instances” or “spot instances”. Reserved instances can be configured in the same way as the regular instances but they will not be started yet. These instances can be bought for one or three years against a reduction in price. The reason for Amazon to do this is that they can predict the hardware that is needed in the

future. Spot instances are instances on which everybody can bid. The bidder with the highest bid can use the instance as long as his bid doesn't get exceeded. This type of instances can be interesting for batch jobs [42], since there is often nobody waiting for these jobs to complete.
In order to spread the load and guarantee availability, Amazon offers some functionality. They offer a load balancing, cloud monitoring and auto scaling. The Elastic Load Balancer can be used to spread the load over different EC2 instances. CloudWatch can be used to monitor EC2 instances, EBS volumes, Elastic Load Balancers, and Amazon RDS DB instances in real time. CloudWatch provides metrics such as CPU utilization, latency, and request counts. The auto scaling functionality gives the user the possibility to automate the process of adding or removing instances. The addition and removal of those instances can be based on the metrics that are provided by CloudWatch.

5.3 Google AppEngine

Google AppEngine is a cloud computing platform (PaaS) for developing and hosting applications in data centers managed by Google. AppEngine automatically allocates more resources when the number of requests increases [61]. AppEngine only supports Java, Python and Go (a programming language developed by Google itself). Google plans to support more languages in the future and says that AppEngine is developed to be language independent. An application that runs on Google AppEngine has to produce its results within 30 seconds [43], otherwise it will be stopped.

5.4 Force.com

Force.com is classified as a “Platform as a Service” as well. It is a cloud computing system from Salesforce.com with which applications can be developed. Once they are developed, they will be hosted on their servers. There is quite some criticism on the IDE and developer friendliness. The platform seems to have potential but currently seems to be inappropriate for customers who want to use Force.com as a standalone platform [60]. Once this platform is chosen, it is hard to switch to another platform or to remove everything from the cloud and run in in a privately owned data center. This is difficult because all software is created for the Force.com platform specifically and it will not run on other platforms. This will create a lock-in effect that might have some negative impact on your performance. The Force.com platform uses its own programming language which is called Apex.

5.5 Oracle cloud

Oracle offers all kinds of services, they can deliver an infrastructure as a service, a platform as a service or software as a service. All this will be done in a private cloud. During the summer of 2012 Oracle will start delivering public cloud services but this will not include the IaaS model. Oracle uses the same techniques for the private and public solutions so it will be easy to switch from private to public and the other way around. In case of a private cloud, an organization still has to buy the equipment but then the organization can choose to manage the hardware itself or let oracle manage the hardware and put it in their data center. With the oracle solution it is possible to remove the software from the cloud and run it into a private data center. This is made possible because it can run on the existing oracle software as well.
Oracle has an entire range of cloud solutions. It is possible to buy small parts of this solution or to buy a total solution in the form of an engineered system. With this solution a customer can buy one rack that contains all the components that are needed.

5.6 IBM SmartCloud

IBM SmartCloud is a part of the cloud computing solutions of IBM. SmartCloud includes IaaS, PaaS and SaaS through public, private and hybrid cloud models [62]. This platform provides cross-platform support without vendor lock-in. IBM mentions that compared to Oracle, they are much more experienced in delivering services. Oracle's product would still be in a very early state on the maturity curve. IBM's IaaS solution is very similar to Amazon's solution. In IBM's solution there is the possibility to select an image of the necessary VM. This VM can be configured to an organization's specific needs.

5.7 VMWare vCloud

VMWare vCloud is classified as an IaaS solution. VMWare offers a private, public and a hybrid solution. The public part of vCloud will be hosted by partners, VMWare will only deliver the virtualization technology on which partners can build their solutions.

5.8 Sentia

Sentia is a small Dutch application hosting provider. Currently Aia Software is in contact with Sentia about hosted and cloud solutions. That is the reason why it is interesting to take a look at Sentia as well. The company can help organizations by implementing their software into the Sentia Cloud. Sentia will help with the development and configurations of a specific application. By doing this they guarantee that the application will meet the performance and uptime needs of the customer. They guarantee an uptime in their SLA of 99.9%. In comparison to other CSPs like Amazon, Sentia is quite different. Where Amazon offers its customers to manage their cloud on their own, Sentia does not support this. Sentia offers fully managed hosting which means that they will do all the activities. When additional capacity is needed, it cannot be added by the customer but it should be done by the provider. Sentia can work with their private cloud as well as with clouds of Amazon and Rackspace. In both cases the cloud will be managed by people of Sentia. Although scaling activities have to be done by someone of the company, those activities can often be done within a day. Like the clouds of Amazon and Microsoft, Sentia also offers a server limit of 8 vCPUs (virtual CPUs), in addition to this processing power there is a limit of 128 GB of memory. The VMs at Sentia run on a RAID-10 array which means that it is already quite redundant. Furthermore there will be an offsite backup once a day and an offsite snapshot once a quarter. This snapshot is made to guarantee continuity. In contrast to Amazon and Microsoft, Sentia offers persistent storage of data inside a VM. This information was gathered via email with Sentia (appendix B.6). Last but not least, the Sentia cloud implementation are ISO27001 certified.

Part II

Solutions

6 Cloud solutions

For a cloud based solution there are several possibilities. In this chapter the possibilities for putting a product in the cloud will be discussed. Of course attention has to be paid to the architecture, but as mentioned before the business model will change as well, so this is something that should be kept in mind while designing a cloud solution.

6.1 General requirements

Outsourcing computation will have some risks for the users, especially when personal data or business critical data is needed for the computation. It is difficult to have a solution that meets the security requirements since the solution has to meet several challenges. The first challenge is that the computational complexity of the system should not be too high, it should be practically feasible. If this complexity is too high the user's costs can become prohibitively high or the outsourced computations might not be completed within a reasonable amount of time. Second it must provide sound security guarantees without restricting assumptions about the system. When the system assumptions are too restrictive the performance of the system will go down again so there has to be a good balance between security guarantees and practical performance. As third the system must provide benefits (e.g. financial savings, less effort, less responsibility) for the user compared to the local computation, otherwise the users will have no reason to outsource their computation. [9].
It is also important for an organization to check whether the software they want to put in the cloud is suitable for being put in the cloud. Some basic situations in which applications could be put in the cloud are:

- Applications that have little or no interaction with back-end systems.
- Web servers.
- Applications that have large fluctuations in the work-load.
- Applications that are used for short term.
- Applications that have to be set up quickly.

An application should not have much interaction with back-end system since all the data has to be transferred over the network or Internet. This will result in a lot of overhead and latency, therefore applications with a lot of interaction between them, should be located close to each other. Web servers are very suitable to be put in the cloud since they are accessible over the Internet anyway. When there

are large fluctuations in the work-load, these fluctuations can be easily captured by some cloud resources. By doing this an organization will not have to buy the hardware to handle the peak load. When an application is used for a short term it might be wise to use it as a cloud service, this prevents an organization from buying hardware for only a short term.
Once it is determined whether an application is suitable to be put in the cloud there are some general requirements that are commonly expected for a complete cloud solution. The requirements that almost everybody wants to have in a cloud solution are listed below:

- The application must perform as well as on-premise applications
- The solution must be scalable
- The solution must be secure
- The solution must have a high availability
- The virtual machines and services must run stateless

A cloud application will not be accepted as a replacement for an on-premise application when its performance is much worse. In order to guarantee this performance when the application is used by more people, the application has to be scalable in order to respond to the changing demand. Furthermore the application should have a high availability to guarantee that the user can use the application whenever he wants. Since security is a great concern when talking about cloud computing, the application has to be secure. In fact being secure is not enough, the security measures should be explainable to the potential customers. Last but not least, a cloud application has to run stateless, otherwise there will be the risk that when an instance breaks, data will be lost. This means that data cannot be stored in a service or virtual machine. Data has to be stored via specific storage services. This has the benefit that data will be accessible for all instances instead of one particular instance. Another advantage of stateless applications is that it has the same execution paths, every time it will be run. So, stateless applications do not contain a state. This means that they don't contain data and specific settings for every person or run. Such application will start with the same state every time it is used.

6.2 Target group

When moving to the cloud it is also important to consider the people that have to be reached. An important thing that a cloud service provider should do is thinking from the customer's viewpoint. In order to do this it is necessary to define the possible customers. It might be the case that only a specific group of people is interested in the solution so why focus on the rest. Another possibility is that large organizations are only interested in private solutions and smaller organizations are more interested in a public solution. This last point is actually not a shared opinion [17]. This means that a cloud service provider, once the potential customers are known, has to pay attention to how to satisfy specific needs. As already mentioned, in order to do this the customer's viewpoint has to be the starting point: which problems should be solved?; why is this interesting? Once the expectations of the customer are known, it is important to manage these expectations otherwise the customers might get dissatisfied. So it is important to communicate with the customer about the possibilities of the cloud solution.

6.3 Architectures

For a cloud solution there are several architectures possible. In figure 5 an architecture is shown in which an instance of the product is running for every customer. By implementing this, the customer has its own private solution which might give more confidence in the security because the application of one customer is separated from the other customer. Furthermore, more customization is possible. For the provider this solution results in more work because all separated instances have to be maintained, of course this could be compensated with the price. In figure 6 there is an architecture in which there is an instance of the product running for every version of the product. In this solution there can be several versions of the software next to each other. For the provider this will result in more software that has to be maintained. For the customer this means that he doesn't have the newest software automatically. This can be useful when there are software updates that require some kind of action at the client side. In figure 7 there is a picture of a solution in which all customers are using the same instance of a product. This is the most general solution of the three. For the customer this means that he is always using the newest version but that he cannot personalize the software that much. For the provider this means that he has to maintain only one version and that he has to provide a really scalable solution. In this solution it isn't likely that only one instance of the product will run because of availability concerns. There can be running copies of the same instance in order to guarantee the availability and maintainability of the system. When updating the system in such a situation it is just a matter of replacing all the copies with copies of the new version. When this is done by one at a time there will even be no downtime, which is an important requirement for a cloud service.

The products that are shown in the figures can run in the public cloud as well as the private cloud. Of course there is also a combination possible in which some components of the product will run in the cloud and the rest is done outside the

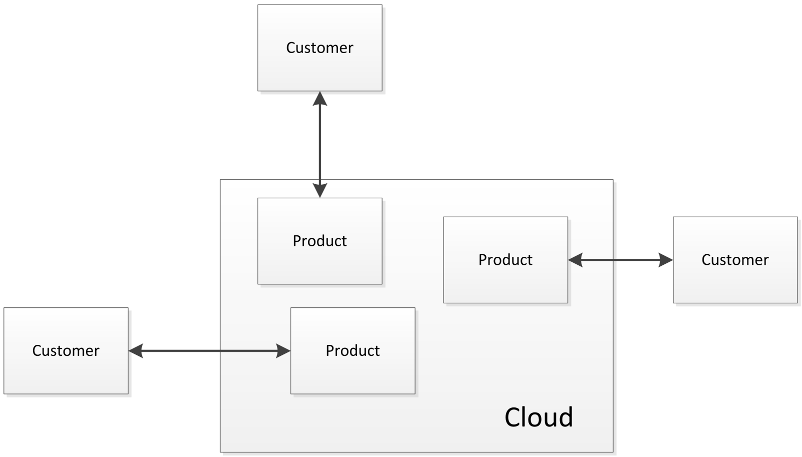

Figure 5: Cloud solution with an in instance for every customer

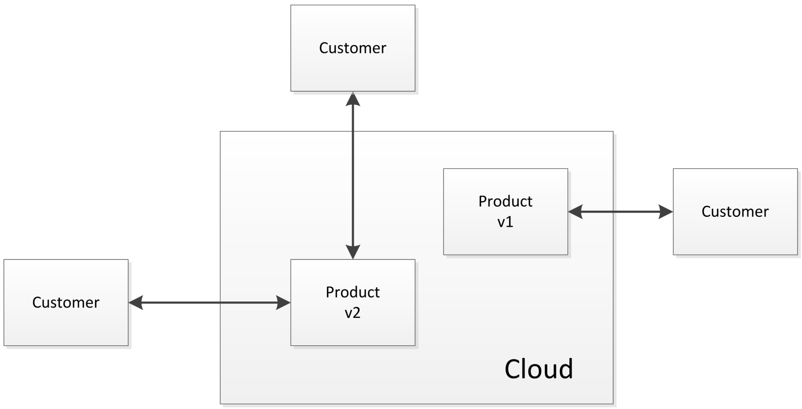

Figure 6: Cloud solution with an in instance for every version

cloud. Another possibility is that they run in both the public and private cloud. In this situation it is possible to give the customers a choice. It is possible to use the public cloud as spare resources in case that the private cloud gets overloaded. A second possibility is to offer the public cloud as a basic product and offer a private cloud is case that the public offering doesn't fit the customer's needs. Reasons for this might be found in legislation.

While on-premise applications often have access to many internal applications or storage resources, this is an undesirable situation when an application is moved to the public cloud. Since in a public cloud the same application is used by many different customers, it is not a good idea to give this application access to internal systems. In general it is a much better option to let the on-premise applications push the data that has to be processed to the cloud.

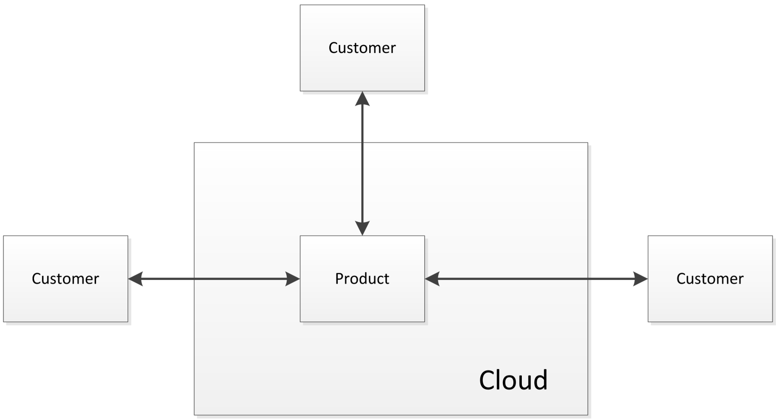

Figure 7: Cloud solution with one instance for all customers

In order to get the promised availability, it is often necessary to run multiple instances of the software in the cloud. By running multiple instances it is possible to increase performance. In order to increase this performance it is necessary to have some kind of load balancing mechanism that spreads the request over the multiple instances. When the load on an applications is very unpredictable, load balancing will not be enough. It is possible to add new instances by hand, but in a lot of situations it would me much more convenient to automate this (with a maximum number of instances set). In order to be able to automate the scaling process the platform on which an application is deployed, should offer an auto scaling mechanism.

6.4 Choosing a provider

Once an organization has decided that they want to use or offer a cloud service, they have to decide whether they are going to develop a cloud completely on their own or that they are going to use services from an existing provider. In chapter 5 there was a list of existing platforms but this is not the only thing about which an organization has to worry when choosing a provider. It is important to make arrangements about a lot of things. For example the SLA; it must be clear how this SLA is organized. A monthly SLA is better than an annual SLA since an annual SLA can have a larger period of time in which the service is not available. Furthermore it is important to make clear arrangements about the ownership of the data, back-up activities and data format in which data will be delivered when it is removed from the cloud. There is a general checklist which should be checked before signing a contract, this checklist can be found in the report of ENISA [35].

6.5 Implementation approach

Every company could start using the cloud but for existing companies it is important to do this in steps. Since existing software is not written for use in the cloud, there are some problems that could occur when this is not done. A company that wants to start using the cloud should start standardizing and consolidating. By doing this it will be guaranteed that the software is basically the same for everybody. If the software would be unique for a lot of persons it doesn't make sense to put it in the cloud because than there should be an instance for every customer and the maintenance benefits would be lost. With the consolidation an organization should try to find the components in the software that everybody uses and that could be combined. Furthermore the infrastructure should be virtualized since that creates the possibility to use the hardware more efficiently [28, 45]. Once all the steps are completed and the IAM properties are set, the software can be placed in the cloud. Of course it is also necessary to have a valid business model because otherwise it will be hard to make money with the solution.

7 Business model

The offering of cloud services results in a change in the way in which an organization can earn money. It is impossible for an organization to ask money for the software itself. Besides this, there are some other important factors that will change. It is much harder to create a lock-in effect based on costs because the entrance costs for the customer are relatively low, therefore it is easy for them to switch to another vendor. There are several possibilities to earn money as a cloud service provider for example by using one of the following pricing models:

Pay-per-use model: pay for every time when a request is done. This will result in costs that are less predictable because in some organizations it is very likely that there are peaks during the year in which the service will be used more. An exception to the unpredictable prices of this model occurs when an organization is capable of predicting the peak-loads.

Pay-per-seat model: pay for every user that is using the application. This results in predictable costs because the number of users to which access is given, is known. This predictability completely disappears when the software is used to process data from a website. In this situation it will be hard to tell what a user is. Once everybody that visits the website is a user, the costs will be highly unpredictable.

Subscription model: pay a fixed price every month. This will result in predictable costs since it will be the same every month regardless the number of users, the time of using, etc. In combination with this model it is possible to ask a fixed price every month for a specific amount of capacity. When more capacity is needed, the monthly subscription fee will increase.

The amount of resources used: pay for the amount of resources that is used (e.g. computation time, memory and storage). A customer cannot predict the costs in advance since he doesn't know how many resources are needed for a specific task.

It is also possible to have a mix of the pricing models. It is for example possible that a provider asks a small monthly fee and that the rest of the cost will be covered by a pay-per-use pricing model. In this situation the start-up costs can be covered and the variable costs can be covered with a variable price. All those possibilities have some advantages and disadvantages in a particular situation. The most important aspect might be that the customer understands what he is paying for. A lot of customers will not understand what they are paying for when they receive an invoice based on resource usage. They just don't know the relation between the tasks and the amount of resources they use. Because a lot of customers will not understand this relationship, it will be hard for them to check the final amount. So a cloud service provider should offer a metering service (used for billing) in which the customer will get confidence for the correctness of the bills. Furthermore it is important to give customers a choice since it might be that customers want different things. Some customer could be happy with the movement to the cloud but others might be not willing to put their data in the cloud. This last type of customer should not be forced to move to the cloud because than they will switch to another vendor. Instead of forcing them, there should be some kind of menu of which they can choose. It can be the case that those customers don't want to move everything to the cloud but that some parts still could be done in the cloud. Once the value of the customers that don't want to move to the cloud is less than the costs of offering the menu of possibilities, it will be a valid option to remove the existing solutions from the menu and offer a cloud solution only. Customers should also get the possibility to manage some things by themselves. This has some impact on the business model because it means less work for the provider.

As already mentioned before, for a CSP it is really important to gain the trust of the customer. When the customer doesn't trust the provider, he is certainly not going to pay for the services that the provider offers. The first step in gaining this trust is transparency. The provider should tell how different things are organized and why the amount specified at the bill is as high (or low) as it is.

As a start-up cloud service provider it might happen that the first year(s) there won't be a lot of (or even no) profit. This can happen because the start-up costs can be relatively high, as with all IT projects. Since revenues will be based on usage instead of licenses or hours, it will take some time before a stable revenue stream is generated. In the situation in which licenses were sold, a lot of money was earned at once while in the cloud situation these licenses aren't sold anymore. This can result in some lower income and higher personnel costs during the first years. In order to prevent this, there should be a plan on how to start the service. There should be an estimation of the resources that are needed, especially when the service will be provided from a data center that is owned by the cloud service provider. Nevertheless it is not always possible to prevent losses. For ISVs (Independent Software Vendors) it might take about 3 up to 5 years before moving to the cloud becomes profitable. So on the short-term moving to the cloud might cost an organization money, but for the long-term it can generate a more stable revenue stream. [29]

According to research of Heliview [29] an IT company that wants to offer cloud services, should do this with a "big bang". This means that the switch to the cloud should be done at once instead of in smaller parts. This means that it should be decided whether a cloud service or an on-premise solution will be offered. The reason for this is that it is hard to combine the different business models. The big bang approach has a high risk and that's why organizations are not using this approach. Because of this several companies have started a second organization to operate in the cloud business. For an average sized company it can take up to 11 months before their first cloud service could be launched. The bigger the company gets, the longer it will take before a cloud service can be launched. The big bang approach doesn't mean that the cloud service has to be released in one step, this can still (and should) be done in smaller steps.

Moving to the cloud has also some impact on the role of IT people within an IT company. Nowadays it is pretty common for an IT specialist to have the role of technology expert, when changing to cloud services, this role will be more like an adviser. The result of this is that there will be less or no work for some IT specialists. The IT adviser will need to have knowledge about the primary processes at the customer side, which he probably already had some.

Part III

Case Study: Aia Software

8 Current situation

8.1 The Company

Aia Software is the developer and supplier of the ITP Document Platform, a solution for the production and distribution of document output based on application data (more information in chapter 8.2). The company was established in 1988 in the Netherlands and is currently located in other countries as well. Aia Software BV is a subsidiary of Aia Holding BV. The complete organizational chart can be found in figure 8. The parts of Aia which are located in other countries are subsidiaries of Aia Holding BV as well, so the parts of Aia abroad are sister companies of Aia Software BV, which is located in Nijmegen. Aia Software BV, Aia Software UK Ltd, Aia Software Deutschland GmbH, and Aia Software North America Inc. are 100% owned by Aia Holding BV. It is Aia Holding BV that is the owner of the intellectual property of the organizations. Aia Software has customers in different countries and different sectors. A lot of customers are in the financial services sector, public sector, legal, IT and communication, manufacturing, retail, real estate, logistics and business services. Some of these sectors will put some additional requirements to the cloud solution. For example for some organization it is not allowed to bring their data outside Europe or even their own country.

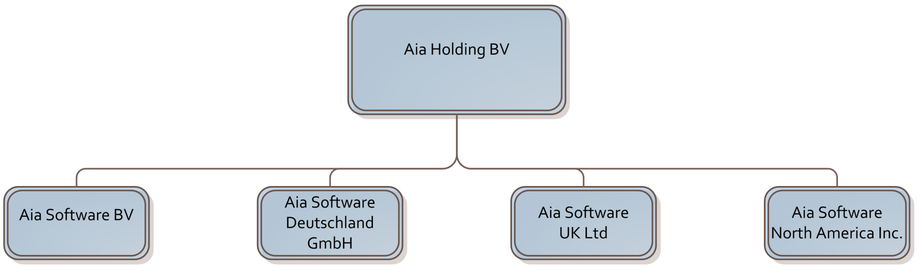
Aia Holding BV
Aia Software BV
Aia Software Deutschland GmbH
Aia Software UK Ltd
Aia Software North America Inc.

Figure 8: Organizational chart Aia Software

As shown in the figure, Aia Software BV has a sister company in the United States. This means that Aia Software North America Inc. is subject to the American law which includes the Patriot Act.

8.2 ITP Document Platform

Aia Software offers its customers the ITP Document Platform. With this platform a product is offered to let organizations manage their communication with their customers. Version 3.5 was replaced by version 4.2 in February 2012. The picture of the architecture (chapter 8.2.2) does not change for the new version. For the inner working (chapter 8.2.3) there were quite some changes that can make migration pretty hard. Upgrades of the software are included in the maintenance costs but are not mandatory. So, the customer has a choice to upgrade to a newer version or stay with the old version.

8.2.1 How does it work?

The ITP Information Architecture describes how someone should look at the product and its activities. The product consists of five different layers: integration, process, data, template and content. The integration layer is responsible for the integration with the business application at the client's side. The process layer is responsible for all the processing that has to be done on data that is retrieved via the integration layer such as conversion from doc to pdf. The data layer is responsible for retrieving data. The template layer contains all components and activities related to templates, i.e. document frameworks, corporate identity definitions, reusable document logic, etc. The content layer consists of components and activities that are responsible for generating the content of a document. Content generation is done via the use of 3 base elements:

- Text blocks
- QForms
- Content Wizards

A text block is a piece of text that can consist of pure text or text with data fields and they are stored in the repository. When a text block consists of pure text it is just a piece of text that can be re-used in several documents. When a text block contains a data field, this data field will operate like a variable. The value for a data field can be derived via the ITP data retrieval mechanism or via user interaction. QForms are used to determine the value of this data field via user interaction. These forms are used to ask the user the value of some relevant data fields. So QForms define the questions for the variable fields in a text block. Related data fields can be put in a field set.

Content wizards are used to enable users to define their own documents. By using a content wizard a user can define the content of a document by adding sections, subsections and text blocks. Furthermore the user can specify which of those elements are mandatory and which are optional. This is something that can be done by the user and actually this should be done by the user, since he is the person that has all this knowledge. The IT department should be only there for support and creating templates that the user can use. This is what is called the BOBMIC (Business Owned, Business Managed, IT Controlled) principle. [52]

8.2.2 Architecture

At this moment there are two versions: a full version and a free version. The free version of the Platform is still in development so there are some uncertainties about the architecture. Basically the free version is a limited version of the full version.

ITP Full version Aia's ITP is a product that consists of basically 3 parts. The first part is ITP/OnLine, the second part is ITP/Server and the third part is a range of document processors. ITP/OnLine is the part that is responsible for user interaction, the applications of the client can communicate with this part by opening a URL. When the creation of a document cannot be fully automated this part will ask the user for extra input. An additional part of this is the letter book in which the user can select which type of document he/she wants to create. ITP/OnLine sends this request to the ITP/Server, this part is basically a queuing mechanism that divides the requests over the document processors. ITP/Server can be called from outside directly as well, in particular this will happen when no user interaction is needed. After the division of the requests, the document processors take templates out of a database/repository and create a document (doc, docx, pdf, Email, SMS, Fax, Print stream, etc.).
During the installation of ITP/Server there are a few parts that are being installed on the system. First there will be the installation directory in which all files are present which are needed to run an ITP/Server instance. Secondly there is the ITP Work directory. This is a directory that exists for every ITP/Server. In this directory data about the host, port, license, etc. is available. This is also the directory in which session data and temporary data will be stored.
In principle ITP generates a doc or docx file without using a word processor. Once this document is produced a rendering engine has to be used to generate another file format out of it. For this step, ITP uses the Microsoft Office Word rendering engine. Because of this step there will be a Microsoft Word license needed when PDF files have to be generated. ITP/OnLine uses Microsoft IIS technology or the J2EE platform. Document processors have a shared file system to store temporary results. If a task has to wait for user input, a next task is started and the temporary result has to be stored on a shared file system because it might happen that another document processor will continue with this task. Data can be send to ITP in XML format, but ITP is also capable of gathering data out of a customer's databases and systems.
An additional component is the CCM (Customer Communication Management) part. With ITP/CCM it is possible to untap the value of an organization's CRM systems. For example an organization's customer can have a preference for a specific format of correspondence (email, mail, sms, etc.). This kind of information

is often stored in CRM systems. With ITP/CCM it is possible to use this information during the document generation process. A model of the ITP Document Platform can be found in figure 9.

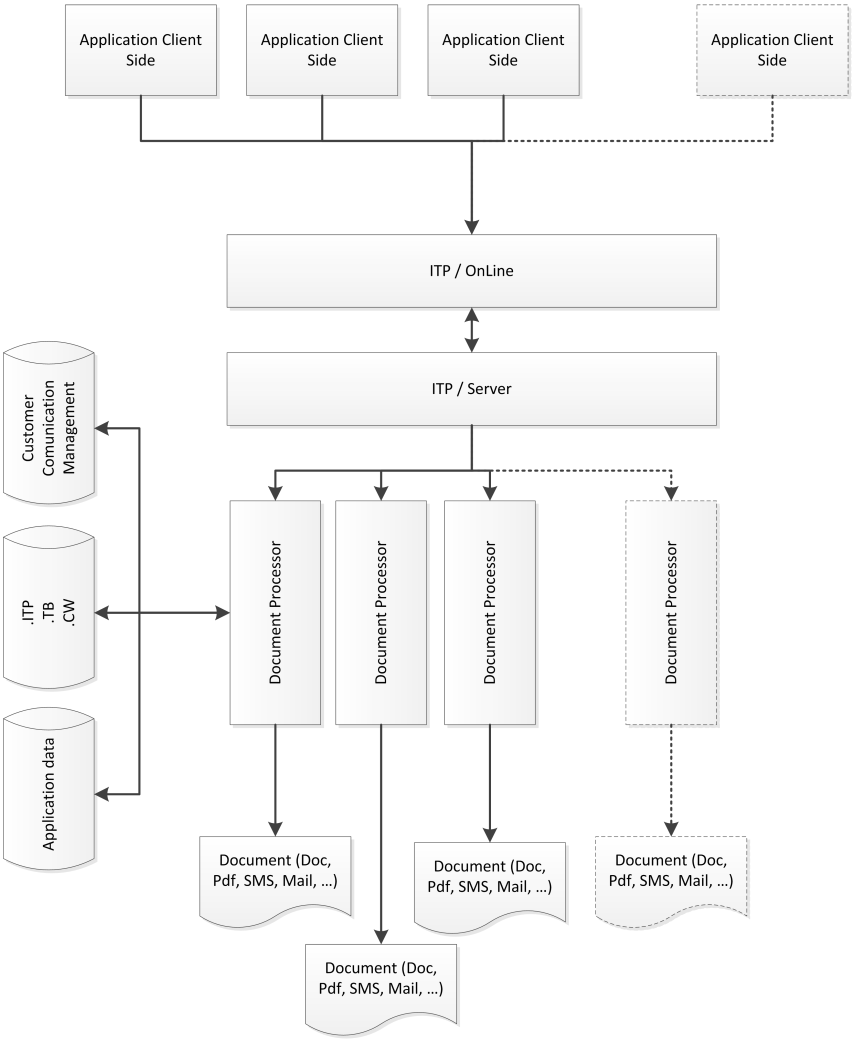

Figure 9: Model of the ITP Document Platform

ITP Express ITP Express is a free version of the ITP Document platform. It consists of exactly the same parts as the paid version. Nevertheless is has less functionality. The first important difference is that there are only doc files produced which results in the fact that there is no license of Word needed. Furthermore application data can be delivered in XML format only. Another difference is that there is no CCM added to the functionality. At this stage ITP Express will contain only one document processor. Another component that is not implemented in ITP Express is the batch part. The execution of batch jobs is not supported for ITP Express. A model of ITP Express can be found in figure 10. This free version is created to generate a lot of ITP users. Once those users are familiar with ITP, they might want to use more functionality for which has to be paid. By doing this, it should be possible to put less effort in getting small companies to buy the software which takes quite some work in the current situation.

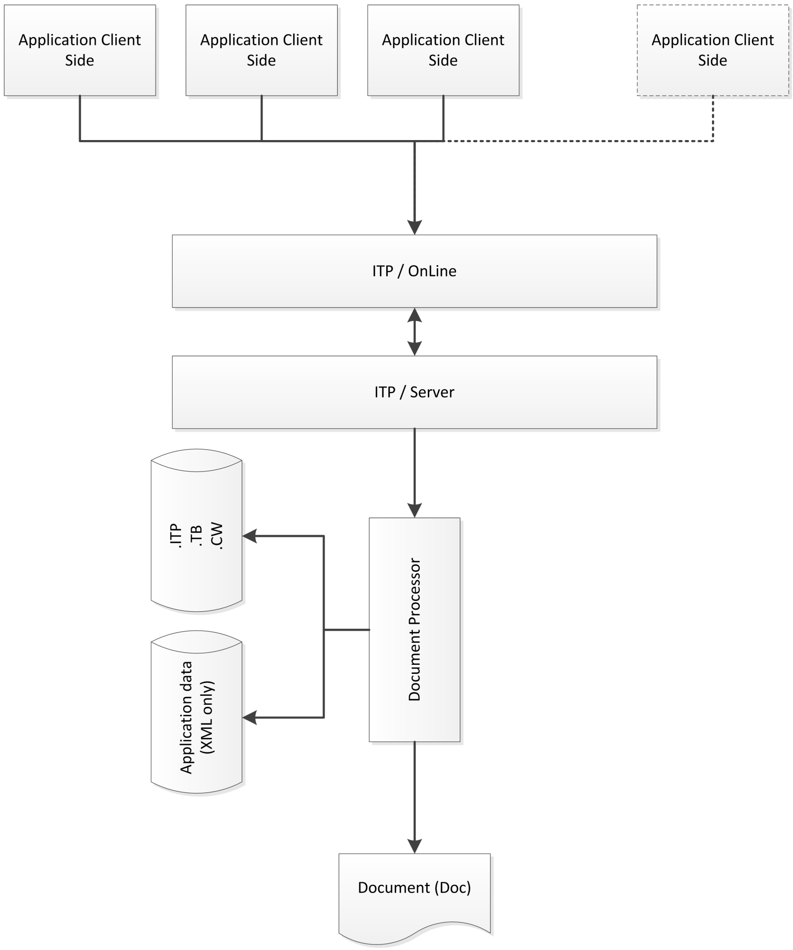

Figure 10: Model of ITP Express

8.2.3 Inner working

In this section a description is provided about how the different components of the architecture work together. In figure 11 there is shown a diagram about how different components work together.

1. (a) The user sends a request to ITP/OnLine. This is done by opening a URL to a specific application. In most cases the user will not type the URL itself but it will be opened by one of the customer's applications.

   (b) The user sends a request to ITP/Server. This can be done via different APIs (e.g. the COM interface, SOAP web services or MQSeries)

2. (a) When a request is send to ITP/OnLine, it will create a session on ITP/Server.

   (b) The session information is send back to ITP/OnLine.

3. Once ITP/Server has received the request, it will be placed in its queue and passed on to a document processor when there is one available.

4. The document processor uses the evaluator to perform all the necessary operations. The evaluator has access to a repository or content publication database from which the necessary data can be gathered. The evaluator has access to a CCM database as well for gathering information.

Once there is information from the user needed, it will be asked by an iteration of steps 5-10.

5. An XForm is send to the document processor.

6. The XForm is forwarded to ITP/Server. During the time that the document processor has to wait, the session will be suspended and another job can continue on the DP. Once the needed information is entered, the job can continue. The entire document will be built from scratch with the data that was already entered, being retrieved from cache. This rebuild is also done when the user clicks the back button in the browser. In fact after every step of data retrieval the entire document is built again. This rebuild is done because it might happen that some things (e.g. Text Blocks and questions to the user) are changed in the meantime. This can especially happen when a job is suspended for a longer time.

7. The XFrom is forwarded to ITP/OnLine.

8. Once the user has submitted the data, the response is send to ITP.

9. The response is forwarded to the document processor.

10. The response is forwarded to the evaluator.

11. After all data is retrieved from the repository, databases and user, the result will be send to the user. The result can be a doc, docx, pdf, etc. The user has to specify whether there are doc or docx files being produced. Other file formats are produced using Microsoft Word. So for the production of doc and docx MS Word is not needed.

12. The result document is send to the client application.

As shown in figure 11, it takes relatively many steps (step 5 to 10) in order to send an XFrom and its response. The reason for this is that ITP consists of several layers. The core functionality is basically the same for a lot of years. Afterwards functionality was added by creating an extra layer around the former product. So first there was the core functionality of producing a document, after some time a queuing mechanism (ITP/Server) was built around it. Again after some time interactive documents had to be produced and ITP/OnLine was added as an extra layer. At some point a web services interface was introduced as an extra layer in ITP/Server. This is how the ITP Document Platform is build and will be built for a long time. The result of this is that request and responses have to be passed on through several layers.

ITP/OnLine is an application that runs on an IIS web server or a J2EE web server. ITP/OnLine, ITP/Server and document processors can be installed on separate machines. In practice some of these ITP components are combined on machines. The sessions that are set up, are basically nothing else than Windows services. The data of those sessions and the data that is gathered during the session have to be stored in a persistent way because it has to be available when a session continues. Data is stored in sessions in order to create a state-full application.

ITP/OnLine is developed for usage within the intranet but there are some customers that access it over the Internet. This “change” is pretty easy to make since ITP/OnLine is a web application.

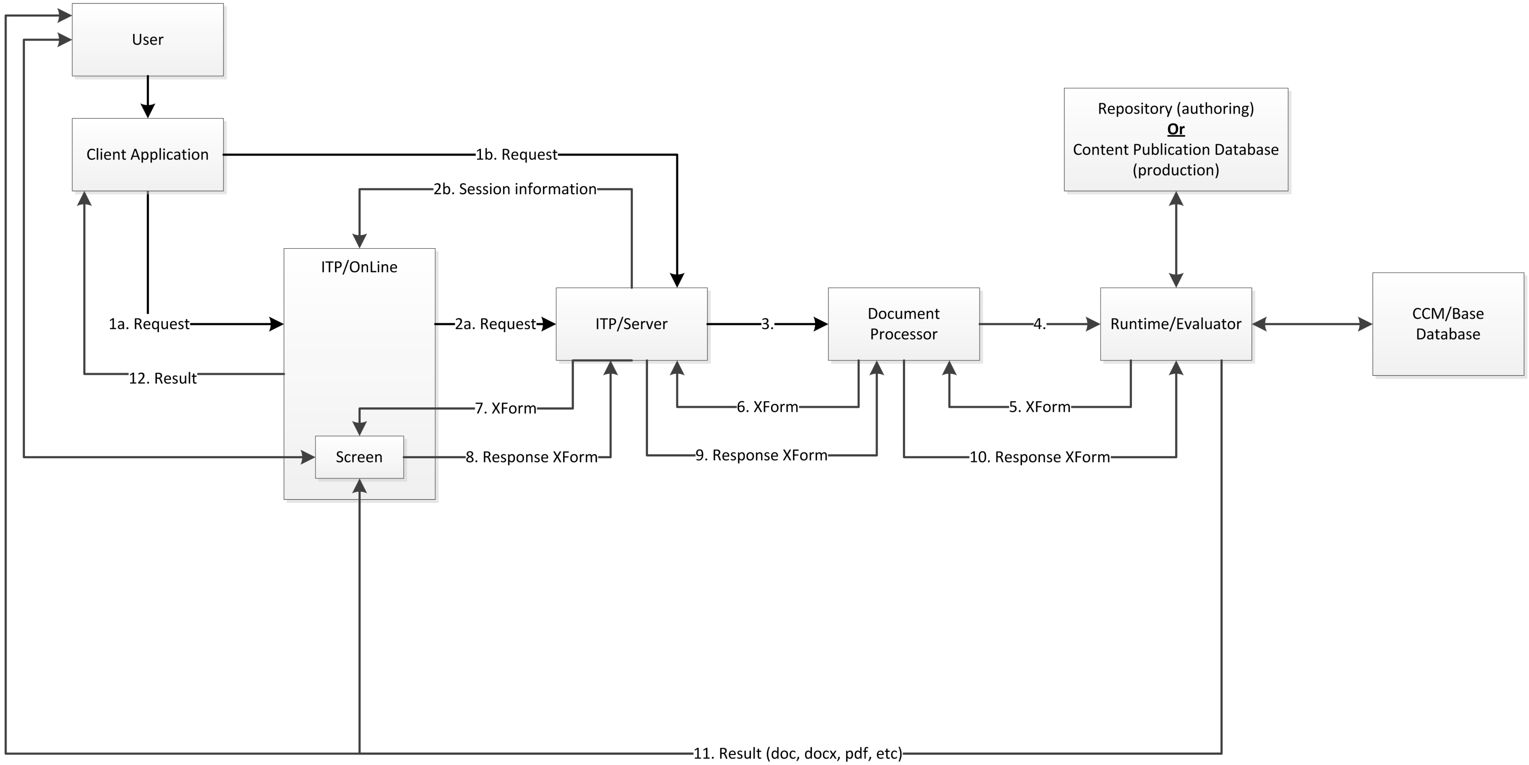

Figure 11: Diagram of communication channels in ITP

9 ITP in the Cloud

Aia Software wants to put the ITP Document Platform in the cloud and, preferably, wants a public solution in which several clients are using the same ITP installation or hardware. When customers don't want to share their ITP installation, there will be a private solution offered. In order to benefit from the advantages that cloud computing offers, there are some changes that have to be made. In this chapter there will be an advice about the best solution for Aia to implement a cloud based solution. This advice is created in consultation with Aia's employees. The first version of ITP that has to be put in the cloud is a version that contains the functionality of ITP Express plus PDF generation. This version is less complex compared to the entire platform. Nevertheless after putting this version in the cloud, it should be possible to put the entire ITP Document Platform in the cloud. This means that decisions that are made for this version should not exclude the addition of functionality of the entire platform.

9.1 Starting point

Aia Software has never run the ITP Document Platform as a real cloud solution. Nevertheless there is quite some experience with the virtualization of the product, in fact there are existing customers that run ITP from a hosting provider. So knowledge about virtualization is already present within the company. The hosted solution does not provide anything that is concerned with dynamic scalability or multi-tenancy and the hosting is performed by a third party that has ITP included in their own product.

9.2 Requirements

The ITP Document Platform is an application that is suited to be put in the cloud since it is an application without a lot of back-end systems. Furthermore the platform doesn't have a constant workload which is one of the characteristics that make an application suitable for the cloud [48]. Some basic requirements that are based on chapter 6 and set in consultation between Aia and me, are mentioned below.

- The solution must be easily scalable
- The response time must be at the same level as in the current situation
- The solution must be secure
    - No data leakage

  - No loss of data
  - No exceeding of the contract
- The software must be reachable at any time
- The solution must run on Windows
- The solution must be made in such a way that the entire platform can be put in the cloud afterwards
  - PDF generation
  - Sending letters/e-mail
  - Printing
- The solution must support programming languages that are currently used (`C++`, `C#`, `C`, Java)
- The solution must prevent vendor lock-in

The cloud platform that will be chosen has to support the generation of pdf as well, although this is not a functionality of ITP Express. The intention is to put the entire ITP Document Platform in the cloud after some time. This means that generation of other files than doc must still be possible. Since there might be a problem at this point, this should be considered while designing the architecture. Another very important requirement is that existing code has to be re-used as much as possible. By doing this the time to market of the cloud version will decrease a lot since there will be much less work. Furthermore this will reduce the risk of introducing errors.

9.2.1 Service model

Since Aia Software will not develop its own cloud, there has to be an idea on how to offer a cloud solution for the customer and which cloud solution should be purchased from existing cloud service providers like Microsoft, Amazon, IBM, etc. In chapter 2.7 there was a list of service models among which has to be chosen. For Aia this means that there are two service models that have to be chosen since they are going to purchase and sell a cloud solution. Aia should purchase an IaaS solution because a PaaS solution most often involves a high level of vendor lock-in. A PaaS solution typically provides some additional functionality like network load balancing, queues and so on. This kind of functionality is most often not present in an IaaS solution, but this is exactly the functionality which involves vendor lock-in.

For their customers, Aia should offer a SaaS solution. This is the best way to go because, for now, Aia only wants to offer the service of creating an output document. There is no need for the customer to install its own applications on the platform. The customers are only interested in getting their documents so ITP in the cloud will be a black box for them. They will just send a request to it and they will get a document back. Only a few customizations can be made via a special customer interface. For batch jobs there has to a mechanism that sends files to the customer in an asynchronous manner because batch jobs can take a lot of time to complete. This can be done by sending every single file to the customer, or wait for the job to complete and zip all files and then send it to the customer.

9.2.2 Deployment model

As described in chapter 2.8, there are several deployment models available which each have their own advantages and disadvantages. Since the public cloud is the most advantageous deployment model (most efficient) this will be the starting point for as well the solution that has to be purchased as the solution that will be offered. When this model meets all the requirements it is the best possible solution. Aia should purchase a public solution because this prevents Aia from buying hardware (which saves start-up costs). When hardware will be bought by Aia, it has to be maintained by Aia as well. This is something that requires some extra expertise which is not available within the company at the moment. Another reason is that when Aia has to buy the hardware for the cloud, the scalability will still be limited to the amount of resources that is purchased. It is possible for Aia to use a hybrid solution in order to remove the limit, but then again the problem of having less expertise about the maintenance occurs. Aia will primarily offer a public cloud service. For customers that don't want (or are not allowed) to use a public cloud, there should be a private cloud which runs almost the same software as the public cloud. This means that the basic solution will be public, but there will also be a private solution in case a customer has some really strong objections against the public solution. This might for example happen when a customer is not allowed to put its data in the cloud. Reasons for this are explained in chapter 4. The most important reason might be that the American government can get access to the data when the data is stored or processed in a data center from an American provider. Another possible reason is that personal data might not be stored outside a country or outside Europe.

9.2.3 Legislation

As with all cloud service providers, there is a lot of legislation applicable to Aia. Nevertheless, Aia will not put personal data in the cloud, it is Aia's customer that puts the data (temporarily) in the cloud. Therefore the customer will be responsible for what happens with the data. Of course the customer wants to have some guarantees about what happens with its data and some information about this should be put in a contract. Nevertheless, since Aia is processor of the data, the company will have responsibilities for what happens with the data as well. In general Aia can use all cloud service providers that are located in the EEA and providers that comply with the safe harbor principles.

Another important aspect is what happens when a cloud service provider (Aia or Aia's provider) fails to meet its SLA. A lot of this information has to be included in contracts between Aia and its customers, and in a contract between Aia and the cloud service provider from which Aia will be purchasing the infrastructure.

9.2.4 Platform

In chapter 5 there was a short list of existing cloud platforms among which an organization can choose. Since Aia needs to develop their software on an IaaS solution, the platform should be classified as an IaaS solution. From the viewpoint of complexity it might be useful to buy everything from one provider, but for from the viewpoint of availability this is the opposite. In this situation the provider can be held responsible for cloud failures. Since Microsoft, Google and Amazon are expected to have a sufficient protection level, they will comply with the Dutch law so those platforms could be chosen as well. The result of those providers being American, is that the USA PATRIOT Act will be applicable to them. The platform has to operate in such a way that the software which is running in the public cloud, can be easily transferred to the private cloud as well. Furthermore it is preferred to have as less lock-in effect as possible.

Some of the platforms that were mentioned in chapter 5 require a specific programming language. Aia is not planning to rewrite all the code of the ITP Document Platform so the chosen cloud platform should be able to handle (most of) the programming languages that are used in the current version of ITP. Because of this the solutions of Google and Salesforce.com cannot be used. Google AppEngine only supports Java, Python and Go which are not the required languages and Force.com only supports its own programming language Apex.

Since it was decided that a public solution is preferred, the platform should offer a public solution. Because of this reason it is not possible to use the solution from Oracle right now since they don't offer a public cloud solution. In a few months this could be different because Oracle will launch a public solution during the summer of 2012, but for now they don't. The same argument holds basically for vCloud from VMWare. In this platform there is also no direct public solution.

The offering of Sentia will be really interesting because it's a Dutch company. Furthermore the company is not that big which means that the response times in case of a problem are likely to be shorter. It will be likely that Aia will have more influence in this process as well. Since Sentia offers a fully managed hosting solution, a lot of work will be done by their specialists. Sentia offers the possibility to run on the Amazon cloud. Sentia also offers persistent storage within a VM. Because of all these possibilities, Sentia will be the best provider to start with.

The costs of this solution are highly dependent on the amount of resources that has to be purchased. The base amount, including operating system management, is set at 90 euros a month. The price per virtual CPU is set at 50 euros a month. 1 gigabyte of memory is offered at a price of 15 euros a month and 1 gigabyte of disk space will cost 50 cents per month. Every configuration that has to run 24/7 will cost an addition 150 euros a month for support. The rate for the network connection (1 Mb/s, including firewall and network service) is set at 125 euros a month. Additional costs to the cloud solution will be the licensing costs for the software that will be installed within the virtual machines. Some examples of software that has to be installed on a virtual machine are: Windows Server, SQL Server and Microsoft Word. Sentia can help Aia in getting these licenses. These licenses have to be paid in monthly payments as well.

9.2.5 Security

In chapter 3 there was an overview of different security aspects in relation to cloud computing. There are quite some top threats that are applicable to Aia Software but to which they don't have influence because these aspects are being controlled by the cloud service provider. For example people doing malicious things is something that cannot be prevented by Aia, the only thing they can do is to try to control how customers are using their software. Data leakage that occurs from shared technology problems is something that cannot be controlled by Aia either. The only thing that can be done is to make the solution a black box and keep data from customers separated. Of course there are also some aspects that have to be taken into account by Aia. Since the customer has to connect to the ITP system in the cloud, Aia must offer some APIs to make this possible. It is really important that those APIs are secure, otherwise there might be the risk that other people can get into the system or that data of a customer might leak to another customer. The login of a customer may never leak to a malicious person.

The privacy requirements mentioned in chapter 3.3.1 aren't all applicable as well. In the cloud solution there will be no data collection by ITP. The customer will have to send all the data to the ITP system. This is done because having a cloud solution that accesses databases from several customers is a situation that is undesirable. In the cloud data gathering has to be minimized, so the customer should send only the information that is needed for the documents. If there is more data send, there will be an unnecessary risk of privacy sensitive data leaking out

of the cloud. ITP in the cloud will only store privacy sensitive data temporarily (when the data is needed for a job that did not finish yet).
In order to keep customers away from each other's data, all access to files and other resources has to be set for a specific session (when a shared ITP/Server is being used). This means that a customer can only access his own required files within a session. Once the session is closed, there will be no access to the data until a new session is started. Furthermore the principle of least privilege should be used. By default a person should have no access to the system, once it is discovered that a person needs to have some rights, these rights have to be added to its account. This is called the principal of least privilege. Once several people need the same rights, role-based access can be implemented.
Identity and access management is of course highly relevant in order to keep data separated. Once there are separate systems for every customer, access rights can be set on folders and databases. As an extra service to the customer it would be nice when the authentication can be integrated in the enterprise solutions of the customer. For Aia Software it is important to have a guarantee that a customer cannot use more resources than the amount that is specified in its contract. For Aia's customers it is really important to have the knowledge that the confidentiality and the integrity of the data is guaranteed.

9.2.6 Standardization

As already mentioned in section 6.5, the first step when moving existing software to the cloud, is standardization. The current version of the ITP document platform isn't standardized since a lot of customer specific configuration has to be set within the software itself. In ITP/OnLine there will be the Online Apps that are customer specific. Once these applications are removed from ITP/OnLine and loaded just in time, the ITP/OnLine installation can be the same for every customer.
More configuration problems exist in ITP/Server. In the current version of ITP, the content publication database is installed next to the ITP/Server component on the same machine. Since this data is very customer specific, this should be removed from ITP/Server and put on a higher level. Another aspect that can be very different for every customer, is the scripting part. Every customer has his own scripts and it is really undesirable that a customer can use another customer's scripts, therefore the scripts should be removed from ITP/Server as well. All this configuration data should be loaded into ITP/Server, just before it is needed.

9.3 Challenges

9.3.1 PDF Generation

ITP Express can generate doc files only. Nevertheless, for the future it is the intention to put the entire ITP Document Platform in the cloud. This means that those doc files need some post processing. Since Microsoft Word is the best application to handle these files, this application should run in the cloud as well. There seemed to be a problem with the Word licenses when running the software in the cloud, but after some correspondence with Microsoft it was said that there is no licensing problem for our solution. One other option is to switch to OpenOffice. The downside of this solution is that doc files sometimes have a different layout when opening them in OpenOffice. Another option is to use Microsoft Sharepoint Server 2010. Sharepoint Server contains an extra component which is called Word Automation Services. These services can convert MS Word documents into other formats of MS Word documents plus PDF or XPS. Word Automation Services also provides the functionality to spool the output files to a printer. Furthermore the number of parallel processes can be specified in the Sharepoint GUI. This gives the possibility of calculating the maximum throughput of documents. The services also provide the functionality of monitoring jobs and restarting conversion when a failure occurred (with a maximum number of retries). When a document is converted, it is possible to automatically remove the source document. The downside of this solution is that a license for Sharepoint Server 2010 has to be bought since it isn't possible to use the services separately. Another downside of the Word Automation Services is that the frequency for checking jobs is 1 minute. This interval has to be set in the Sharepoint Portal. This means that it is possible that a user has to wait 1 minute before his conversion job is even started. This kind of behavior is unacceptable for interactive jobs, it is not done to let a user wait that long. A last more rigorous solution might be to stop using any kind of the existing conversion techniques and start creating, for example, tex files. This might have some other advantages like an already available web editor. This can result in some advantages when the models are being designed.

9.3.2 Multi-tenancy

The ITP Document Platform is currently designed to serve only one customer. When ITP will be put in the cloud there will be real benefits for Aia when a lot of customers can share the same system. When doing this it is important that the customer will not notice this in the performance. This means that all customers should be treated fairly according to the SLA. It is not acceptable when one customer has to wait for another customer to complete his jobs. An even more important aspect is that a customer cannot access another customer's data. So the multi-tenancy problem, basically exist of two smaller problems. One problem is how customer data can be kept separated and the other problem is how the performance for every customer can be guaranteed.

To separate customer data, there will be a separate content publication database (CPD) for every customer. This also solves the problem that data communication to the database cannot be load balanced. The databases can be accessed by the corresponding customer since this is necessary for the publication of the models. Furthermore these databases can contain a lot of additional configuration information in order to get ITP/Server stateless. Session data and temporary data will be stored in a database as well because it is undesirable that this data can be accessed other customers.

In order to guarantee direct response to a customer's request, there should always be some overcapacity (only when there is less capacity available than the capacity for which is paid by the customer). This overcapacity is necessary because it takes some time (this varies from 2 to 10 minutes) for a new instance to start. For a customer it is unacceptable to wait this amount of time. This problem could be partly solved by setting a trigger on the response time. Once the response time is increasing there can be a trigger to start a new instance before the moment that the response times in the SLA are reached. This will probably result in better performance, but when a lot of request will come at once, this option will not solve the problem. Another possibility to create a higher availability for multiple users, is by using two or more different providers. By doing this there will be no dependence on a single provider so when a failure at the provider's side occurs, it will be relatively easy to distribute to the other provider. It is possible to switch the load over different providers when the requests comes in, but this won't be the best solution since network traffic between different providers is relatively expensive compared to internal traffic. When two providers are being used, it is better to keep them separated and put one pool of customers on an ITP installation that runs with one provider and another pool of customers on installation at another provider. When a large failure occurs in this situation, it will still be relatively easy to switch customers to the other installation.

9.3.3 (Dynamic) scalability

ITP already has some kind of cloud implementation. This implementation is running on a VM that can be hosted by another party. This could be a seen as a first step of a cloud solution. Nevertheless the benefits of this solution for Aia are not present since the business model is not adapted to it. Furthermore this solution has to be implemented for every customer. When Aia is going to handle the hosting of the product, it would be nice if the product is scalable. It is even better when the solution is dynamically scalable. This means that the solution will automatically scale according to the demand. By doing this there is not much interaction needed. So the second step is that a “cloud operator” within Aia starts an implementation when there is a new customer and afterwards the application will automatically adjust to the demand. The last step that is left, is to create an application that supports multi-tenancy and that can scale automatically based on demand, without treating users unequally (according to their SLA). In order to get the ITP document platform highly scalable, it is necessary to make the installations as stateless as possible. In order to do this, the session information of ITP/Server has to be stored outside ITP/Server. By doing this, a job can run on every ITP/Server.

9.4 Business model

As already mentioned in the previous chapter it is important for an organization to pay attention to its business model when moving to the cloud. This is not different in the case of Aia since they do no longer sell the software itself. Billing should be done in such a way that the customer understands what he is paying for. It doesn’t make sense to send an invoice based on resources that are used. In order to prevent customers from being forced into the public cloud, there should be a private cloud solution as well which they can run in their own data center. Otherwise there will be the risk of losing those customers because they don’t want or are not allowed to put their data in the public cloud.

In the new situation Aia will not be selling a product anymore, they will be selling a service. This has a huge impact on the business model. In the new business model there are no such things as maintenance contracts. This means that maintenance costs and support costs must be covered by the pricing model that will be used. Since ITP will be a black box for the customer, and therefore also for the customer’s IT department, there must be a 24/7/365 support service (which does not have to be free of charge). Arrangements about this support can be made in the SLA which is set up with the customer. So, a part of these costs has to be added to the development costs, the costs of human resources, hardware, etc. Based on these accumulated costs a price per month could be derived. Besides this Aia will be billed, by the CSP, based on resource usage. These costs are highly variable so it could be a good idea to ask a monthly fee in combination with a usage fee. This usage fee could be based on the number of documents that is produced.

When there is a known maximum number of document processors that can be used by a customer, that customer can be billed based on this maximum number. In this situation there will be no loss of money when all customers are using their full usage.

In consultation with the management of Aia Software, it is decided that the customer must be able to predict his costs. This means that the customer will buy a maximum amount of ITP capacity for a fixed price per month. When this limit is reached the customer will have to wait till previous requests are processed.

9.5 Architecture

9.5.1 Potential Architectures

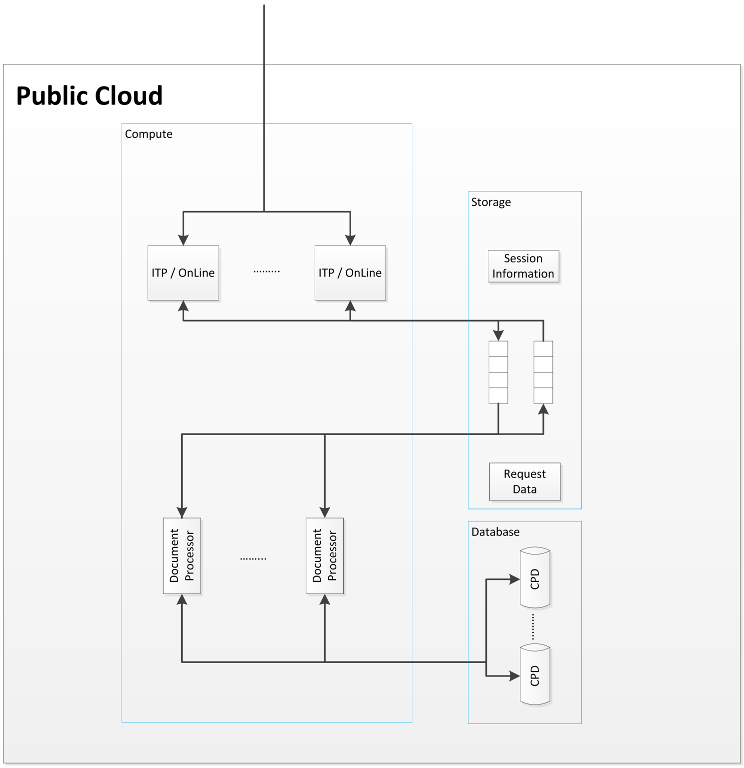

Figure 12: Using a queue

Figure 12 shows a picture in which a queue is introduced. A queue has some benefits since it provides load balancing, load leveling. This means that it should be able to replace the task of ITP/Server. In general, a queue can help to achieve the following business objectives:

Temporal decoupling: This allows message senders and receivers to work on independent schedules. This means that the sender and the receiver don't have to be online at the same time. This results in the fact that some parts can be taken offline without losing messages, they will just be continued when the service comes back online.

Load leveling: This allows work to be spread over time. This results in the fact that a message consumer doesn't have to be able to handle the peak load.

Load balancing: This allows multiple message consumers. Because of this it is possible to add multiple consumers during peak loads. This can help to keep a high throughput.

Loose coupling: This allows message producers and consumers to work completely independent from each other. This means that they can do their tasks without the need of interference of the other.

Large requests will be stored on a cloud storage service and a reference to it will be added to the queue. By doing this there are no large requests that have to be loaded into the memory. The downside of this design is that the queue is a service that has to be provided by Aia's CSP. This means that, unless Aia will develop its own queue, the number of CSPs will be very limited. Because of this it will be less easy to switch to another provider. Furthermore this solution will need relatively much work that has to be done to get it working.

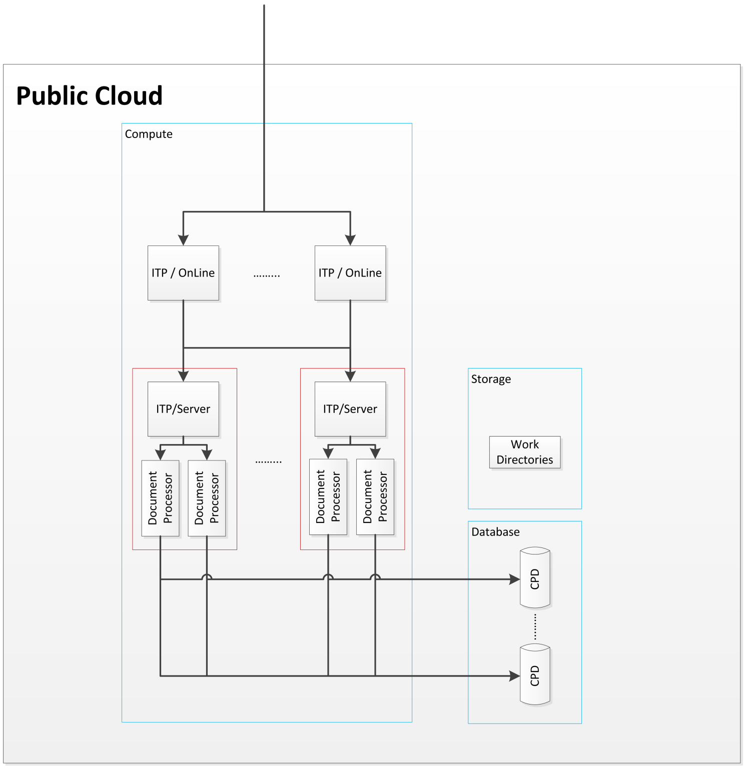

Figure 13: Combining ITP/Server and the Document Processors

Figure 13 shows a picture of an architecture in which ITP/OnLine can connect to several servers. The advantage of this solution is that the queue is replaced by ITP/Server which is proven technology. By having several instances of ITP/Server with a fixed number of DPs, there will always be spare capacity. The downside of this solution is that there has to be some kind of mechanism that guarantees that different tasks of the same session, return to the same ITP/Server. Since there is a maximum number of cores in most compute instances, the number of DPs belonging to an ITP/Server will also be limited. Nevertheless, in ITP 4.2 it is not possible to have a document processor installed on an external machine. This is something that can be solved by using this solution since the document processors will be installed on the same machine as the server or by making a small change in the code of the document processor. So, this won't be a deal breaker. Furthermore, because of the limited number of DPs per server, it is likely that there will be less ITP/OnLine instances needed than there are ITP/Servers needed. This will automatically result in the fact that ITP/OnLine has to communicate with several ITP/Server instances. This is currently not possible since this will result in some problems because session information has to be accessible by all ITP/Server instances. There will be a problem of race conditions since it is possible that a follow-up request will be processed while the previous request isn't finished. The other solution will be that it has to be guaranteed that follow-up requests have to be processed by the same instance of ITP/Server. Another problem might be that, when it is not possible to change the size of a compute instance, scaling has to be done with some pretty large instances which might result in quite some overcapacity.

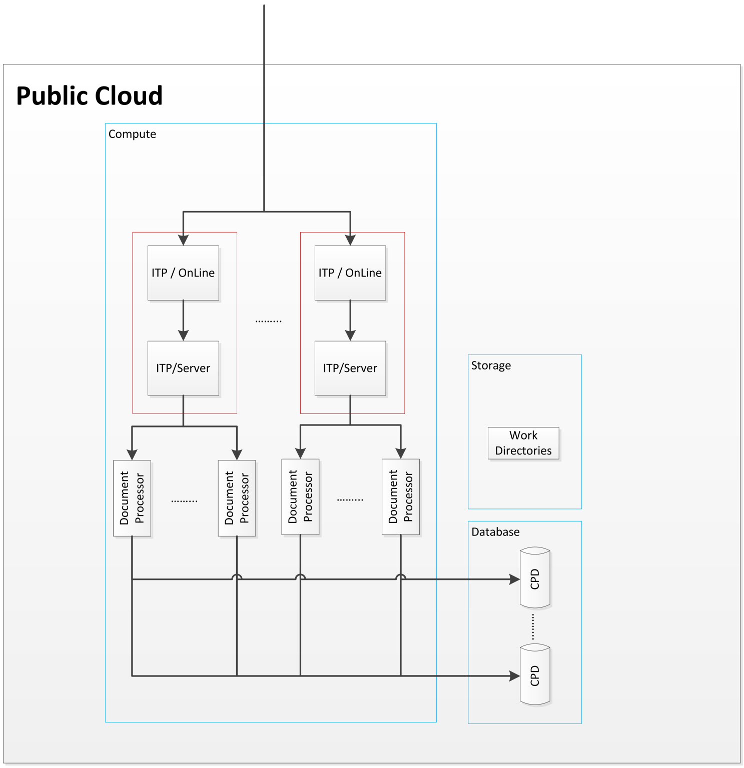

Figure 14: Combining ITP/OnLine and ITP/Server

Figure 14 shows a design in which ITP/OnLine and ITP/Server are combined in one virtual machine. Document processors are placed in separate virtual machines. ITP/OnLine and ITP/Server can be placed in the same VM since they don't need that many resources. The heavy resource usage is placed at the document processor level. In this situation heavy resource usage of a document processor will not affect the responsiveness of ITP/OnLine. Again, requests of one session have to return to the same ITP/Server, but now this can be done at the upper level. In this situation there will be more ITP/Servers than necessary, but this will not influence the performance and price since the work load of ITP/Server is almost nothing. This solution makes it relatively easy to send request to the correct instance of ITP/Server since load balancing (and traffic management) is done at the upper level. Nevertheless this solution is not possible with the current version of ITP (4.2) so there have to be some adjustments made. Another possible downside in this solution is that when an extra ITP/OnLine instance is needed, there will also have to be some document processor instances created because otherwise there will be no document processor that is connected to the ITP/Server that comes with the ITP/OnLine virtual machine.

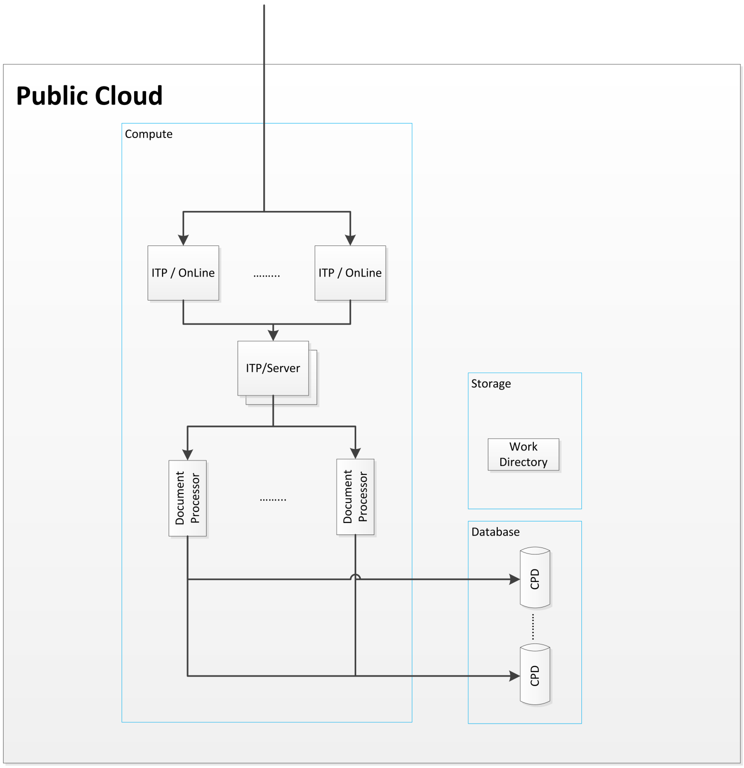

Figure 15: Separating all components

Figure 15 shows a picture in which ITP/OnLine, ITP/Server and the document processors are in separate virtual machines. In order to guarantee availability, there have to be at least two instances of ITP/Server. Therefore in this situation there will be a kind of failover mechanism for ITP/Server. If one Server goes down, a new one will replace the first one. The downside of this solution is that there will be a limited number of document processors and there is time needed to detect the failure of ITP/Server. Furthermore, ITP/Server and ITP/OnLine are both processes that don't need that many resources so this solution will have some additional costs because there are some extra compute instances needed for ITP/Server and ITP/OnLine. Because of this it might be better to combine these two components into a single virtual machine. Again there have to be made some adjustments to the current version of ITP, in order to let this solution work. When the load in the ITP installation gets really big, it will be necessary to run a second system with this architecture in order to handle the load (because of a maximal capacity of ITP/Server).

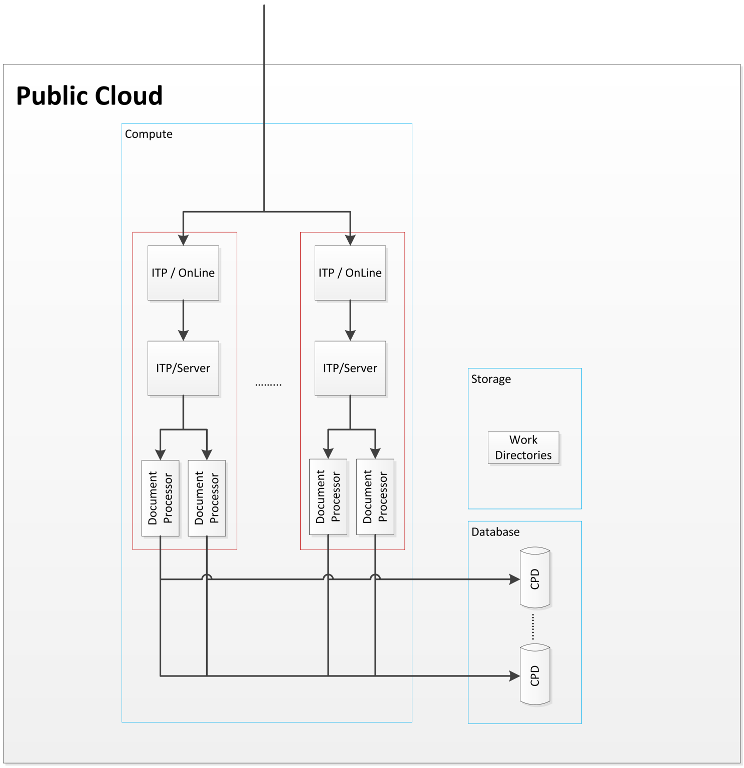

Figure 16: Combining all components into a single VM

In figure 16 a solution is shown in which a complete ITP installation exists in one virtual machine. This means that scaling occurs by adding a complete new installation of ITP. This seems to be the simplest way to go since this is definitely possible with the current version of ITP, but there will be the risk that heavy resource usage of the document processor, will influence the performance of the other components. Again this solution has a single point of load balancing/traffic management. Furthermore scaling has to be done with large compute instances, unless it is possible to change the size on demand. Just like the architecture in which ITP/Server and the document processors are combined (figure 13).

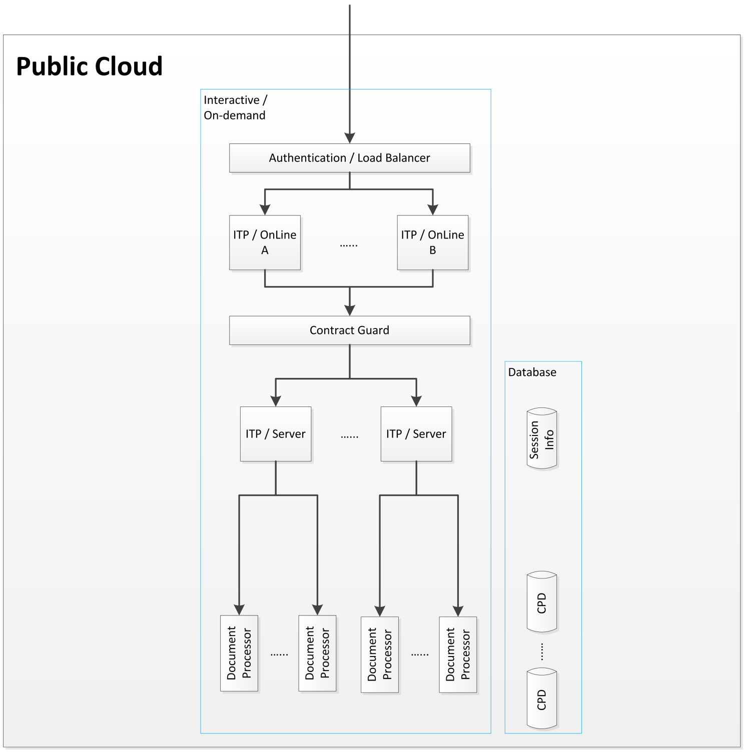

Figure 17: Multiple shared ITP/Servers

Figure 17 shows a more customer specific architecture. In this architecture there will be a specific ITP/OnLine installation for every customer. By doing this it is possible to have a lot of customizations for the customer. A “contract guard” component can be added to check whether a customer has already reached his limits for which he has paid. Furthermore this component can be used to load the configuration for ITP/Server that is needed for a specific job. In this architecture ITP/Server and the document processors will be shared by all customers. This has some implications for ITP/Server. Because of the shared character, scripts and session information has to be stored outside ITP/Server. In order to prevent several ITP/Servers from accessing the same session information, this information will be stored in a database that provides a locking mechanism. Scripts will be stored outside ITP/Server because they are very customer specific.

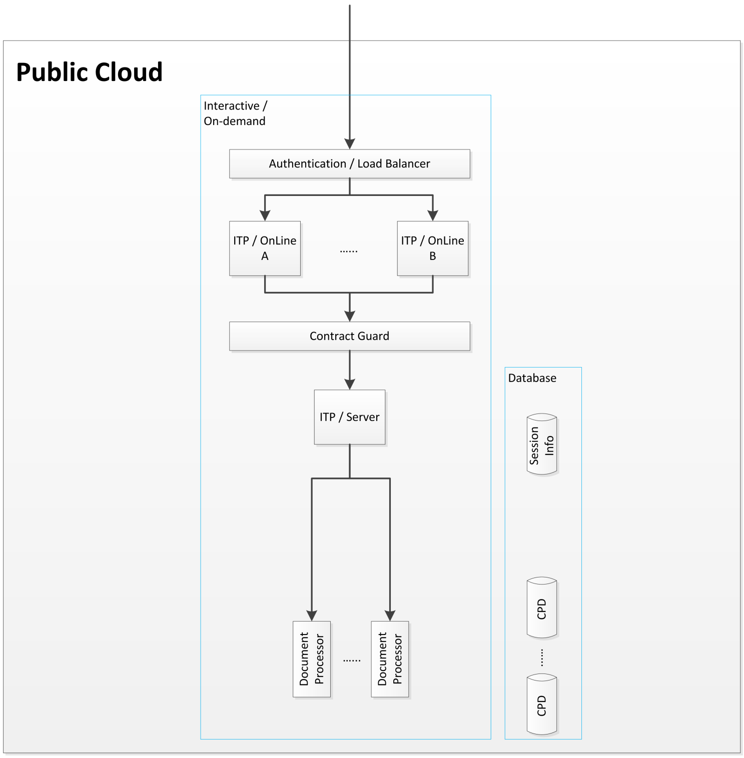

Figure 18: Single shared ITP/Server

Figure 18 contains an architecture that is in essence the same as the architecture in figure 17. The only difference is that there is only one ITP/Server. This means that session information can be stored in a database, but this is not necessary so there will be fewer changes in the software required. Once the limit of ITP/Server is reached, there should be a second installation that looks the same. Once this installation is ready, a part of the customers can be ported to the new installation in order to divide the work load over multiple installations.

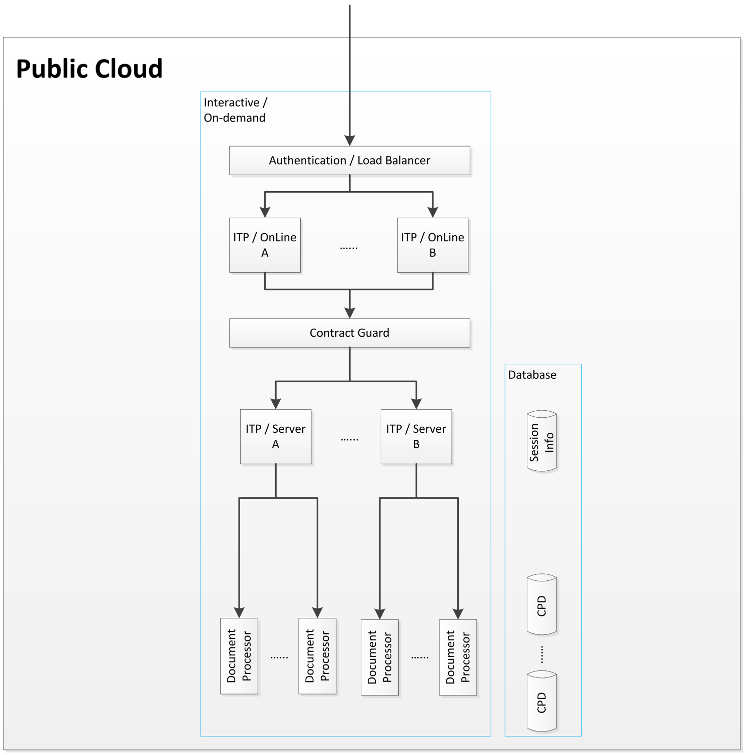

Figure 19: Contract guard after ITP/OnLine

Figure 19 shows an architecture that looks a lot like the architecture in figure 17. There is only one important difference that has some consequences. In this architecture ITP/Server will be customer specific as well. This means that there are more ITP/Server installations needed. Since ITP/Server doesn't use a lot of resources, it is possible to install multiple ITP/Servers in a single virtual machine. The other downside of this architecture is that also the document processors will be customer specific since they belong to only one ITP/Server. This will result in a less efficient use of the document processors this customers cannot use capacity that is not being used by other customers.

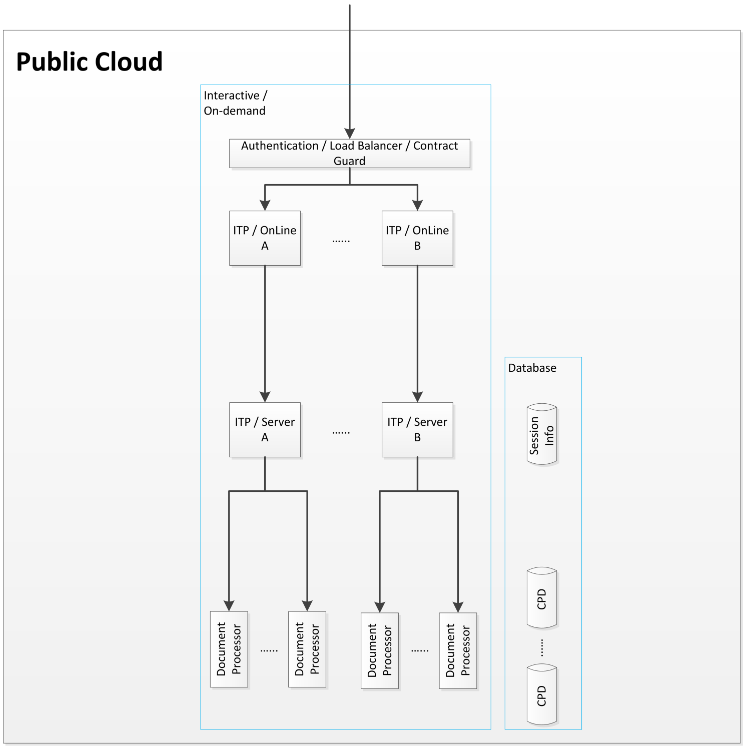

Figure 20: Contract guard at load balancer level

In the architecture in figure 20 ITP/OnLine and ITP/Server are customer specific as well. But in this architecture the authentication part and the contract guard are placed at the top level. This means that ITP/OnLine and ITP/Server could be place into a single virtual machine. The downside is again that document processors will be customer specific again (because of ITP/Server being customer specific).

9.5.2 Final Architecture

In chapter 6 there are shown some general architectures for a cloud based solution. Because the intended business model suggests that every customer has a fixed number of document processors available, ITP/Server can be assigned to a customer as well. ITP/OnLine will also be customer specific which means that the architecture of figure 5, in which every customer has his own installation, will be the best solution. Every customer having his own installation doesn't mean that they cannot run on the same hardware. It also doesn't mean that the installations have to differ a lot. Furthermore there can be some kind of load balancing component above all installation that distributes the requests to the correct instance. The architecture shown in figure 6, in which there are several product versions, could also be useful, but this solution will be more useful when all customers are sharing all components. When the support for the old version is stopped, the architecture of figure 6 is actually the same as the architecture in figure 7. The solution in figure 5 can also handle different version.

The architecture in figure 7, in which there is only one installation of the product, is not suitable for Aia. Since there are destructive updates possible for ITP (like the update from ITP 3.5 to ITP 4.2) this will be a deal breaker for using this solution. Otherwise this will result in the fact that after an update, customers cannot use ITP anymore.

In the previous section there were described a lot of possible architectures. These architectures were discussed with the management of Aia and finally one architecture was chosen that had to be extended. Figure 12 to figure 16 were evaluated but it was decided that the business model will be based on the number of document processors. Furthermore the application should be able to do more than just producing pdf. Because of this, more customizations are needed, which results in an installation in which there are customer specific components. Once this was known, figure 17 to figure 20 were created. At first the intention was to create an architecture in which document processors would be scaled on demand. After some discussion it was determined that once a customer pays for 3 document processors, there should be always 3 document processors for that customer. Because of this decision, ITP/Server could be assigned to a customer as well because the workload of ITP/Server is not that high and it doesn't make sense to divide the work of one customer over different ITP/Servers. This means that the architecture in figure 19 and figure 20 were left. Since it is better for the entire ITP product to remove customer specific configuration from ITP/Server, it is decided that this will be done. This means that in the figure, every customer has its own ITP/Server, but these ITP/Servers will be clones. In this case a clone means that they will share the installation base. The ITP/Server installations will differ from each other in the work directory. This work directory contains scripts that can be executed and other specific settings like the host and the port on which that specific ITP/Server is running.

The architecture will be designed in such a way that the ITP Document Platform becomes a kind of black box. The customer will send in a request via ITP/OnLine or his application and gets a document out. At first this request will be an XML message and the result document will have the doc format. The customer will be allowed to modify the settings of ITP as less as possible. The architecture for 1 version of ITP can be found in figure 21 and 22. When another version of ITP is added, for a specific customer the installation can be replaced by the new installation. There will be no connection between the different versions. In order to get the software easily scalable, all document processors must be able to communicate with the different content publication databases of the customers. Since every customer has his own content, these databases have to be separated. When there are too few document processors there will be an option to add (temporarily) an extra document processor. In the beginning this has to be done by someone within Aia Software.

In the architecture there will be running two systems. One system is responsible for the interactive and on-demand jobs, and the other system will be responsible for the batch jobs. The distinction between the two systems is made to guarantee availability because otherwise there would be the risk that a series of batch jobs, blocks all other jobs. This is in particular true when one ITP installation is shared by different customers. So at the start when every customer has his own document processors, the batch part could be combined with the interactive part. The only thing that has to be done in this case, is setting priorities for interactive and on-demand jobs. For the interactive part there will be a limit for the time that a job can run in the document processor. When this time gets exceeded the job will be killed because interactive jobs should be executed quickly. If the execution takes too long, the customer has to change its model. This measure is also implemented for guaranteeing response times.

Since ITP will be accessed over the Internet it must be prevented that everybody can start using ITP. Therefore there will be an authentication component in front of all the ITP/OnLine components. This component can also handle the maximum number of users that is allowed to use the ITP document platform. The business model has quite some impact on the architecture. Since the business model will be heavily based on the maximum number of document processors that a customer can use. This means that it has to be prevented that a customer exceeds the processing capacity for which he has paid. This level of capacity (and other agreements that are made) is described in a SLA. Because of this there has to be a component in the architecture that checks this SLA and decides which actions a user can do. Therefore in the architecture there is the "contract guard" component. This component can also be used to load user specific configurations. The contract guard and authentication layer can be combined in a single component. Since different customers use for example different scripts and constants, this should be removed from ITP/Server and be stored externally. Since ITP/Server should have this configuration at some point, it is a good place to gather this information just before ITP/Server is used. When such a component is added to the current on-premise version of ITP, this version will also benefit from this solution since the components are more generic without losing functionality. This will help to improve the installation process. This is actually part of the standardization phase that was discussed in chapter 6.5. In this phase the software should be standardized for every customer, which means that there should be no customer specific information in the software components. Another aspect that can be handled by the contract guard is identity and access management. Since the contract guard is intended to route requests to the correct installation, this will be the right place to check whether the requester is allowed to perform the action on that particular installation. So the contract guard must not only check whether the limits of a contract are reached, but it must also do the access management to the ITP/Servers. Of course a part of the access management has to be performed in the authentication layer. The authentication layer also has to do the identity management. The most important identity for the ITP installation that has to be determined, is the organization from which a request is coming. Based on the organization, an ITP installation can be chosen.

Another important difference with the current version of ITP is that session information, scripts and constants of ITP/Server will be stored in a database. This is done because this provides us a locking mechanism that guarantees that the session information is used by only one instance of ITP/Server. Previous research within Aia Software has shown that this wasn't possible on the regularly file system. The other information is stored outside ITP/Server because this creates an ITP/Server that can be used by any customer, as long as all the configuration data is in the database. This is important when an ITP/Server installation will be shared across multiple customers. This is not the initial idea of the standard version for ITP in the cloud but this might be the solution when ISVs are going the use ITP in the cloud.

9.6 Implications

Putting ITP in the cloud has some implications that have to be considered. The advantage of the product in the cloud is that the newest version is always available for the customers. This might also bring some problems when the software is being updated. Updates that do not need to change the software at the clients side can be done automatically and for everybody, but updates for which the client also has to change some things cannot be done automatically since this will break the software for clients that have not done those changes. This problem could be solved by having two versions in the cloud. Since this is not a desired situation, the time that there are two different versions in the cloud, should be limited. For example: support for the old version could be given for 2 years after a new version is released. Somewhere within this time the customer must switch to the new version or the contract could be ended automatically.

In the current situation ITP can gather its data by accessing the databases at the client side. In the cloud solution this cannot be longer the case. Since the functionality of ITP Express is the first version that will be in the cloud, this not a problem since information can only be send to ITP via a XML message. Once more functionality is added to the cloud solution Aia should keep in mind that it is not possible anymore to access the database at the customer side. All the data that is needed should be send to ITP via push messages.

The location of the data and the data processing might be important for some customers. Some customers (including foreign customers) are not allowed to bring their data outside the country borders. When it comes to foreign countries, the legislation in these countries still has to be considered.

The performance and the costs for the customer will be related to the requests that are being sent to the cloud version of ITP. In the current situation there is a lot of information in the request that is possibly used by ITP. This means that there is a lot of information that wasn't necessary to send with the request. Especially in the cloud this generates a lot of traffic for which has to be paid. It is up to the customer to pay this extra amount of traffic. This is something that should be told to the customer. This might change the way in which they will send requests to ITP. Another cost aspect to which the customer might have some influence is the overall time that ITP is used. By splitting large batch jobs, there will be several document processors that each require some start-up time. This means that not splitting batch jobs can result in lower costs. Nevertheless, this isn't always true, so this should be done with caution. When large documents are being created, it might be more advantageous to send smaller jobs because of heavy resource usage. The definition of a large document depends on image format, image size, number of pages, number of styles, etc. So when sending a request to ITP this is something to which the customer has to pay attention. At some point, there will be moment at which it will be more advantageous to cut a batch job into smaller parts again.

The ITP Document Platform is also capable of running batch tasks. This task is not included in ITP Express but there are some issues that have to be handled. Since batch tasks can need a lot of time to complete, there are some problems when running batch jobs in a shared environment. Since interactive jobs need user interaction and have a short time to completion, it is unacceptable that a user has to wait for a batch job to complete. Therefore the batch jobs and the interactive jobs should run in a separate instance. When this is done, not all the problems are solved yet. There is still the risk that one customer creates so many batch jobs that all document processors are being used. In this situation another customer has to wait till new instances are started. For this problem there will be several options. The first option is using the internal mechanism of ITP/Server that reserves a specified number of document processors for interactive and on-demand jobs only. Document processors that can handle batch jobs, can handle interactive and on-demand jobs as well (the other way around is not possible). When doing this, research should be done to the right division between document processors for interactive/on-demand jobs and document processor that can handle all jobs. As another option there could be document processors assigned to a specific customer, which means that those document processors will only process jobs of a specific customer. This also means that a customer, who needs an extra document processor, has to wait till a new instance is started because he cannot use a document processor that might be free because that one is not assigned to him. Another possibility is to create a new load balancing mechanism that keeps track of a maximum number of document processors that can be used by a customer. In this situation there will still be the risk that a customer has to wait till his jobs start. The best solution in my opinion is to create document processors that listen to a queue for jobs from a specific customer. This is the only situation in which it can be guaranteed that one customer doesn't have to wait for the other customer to complete his jobs or for starting up a new instance. For batch jobs this will be less important because it is very likely that nobody is waiting directly for the documents being produced. When running an interactive job a user will be waiting to enter new input and to receive the result document. So for interactive jobs the response times have to be very low but for batch jobs it can be justifiable to have some additional start up time.

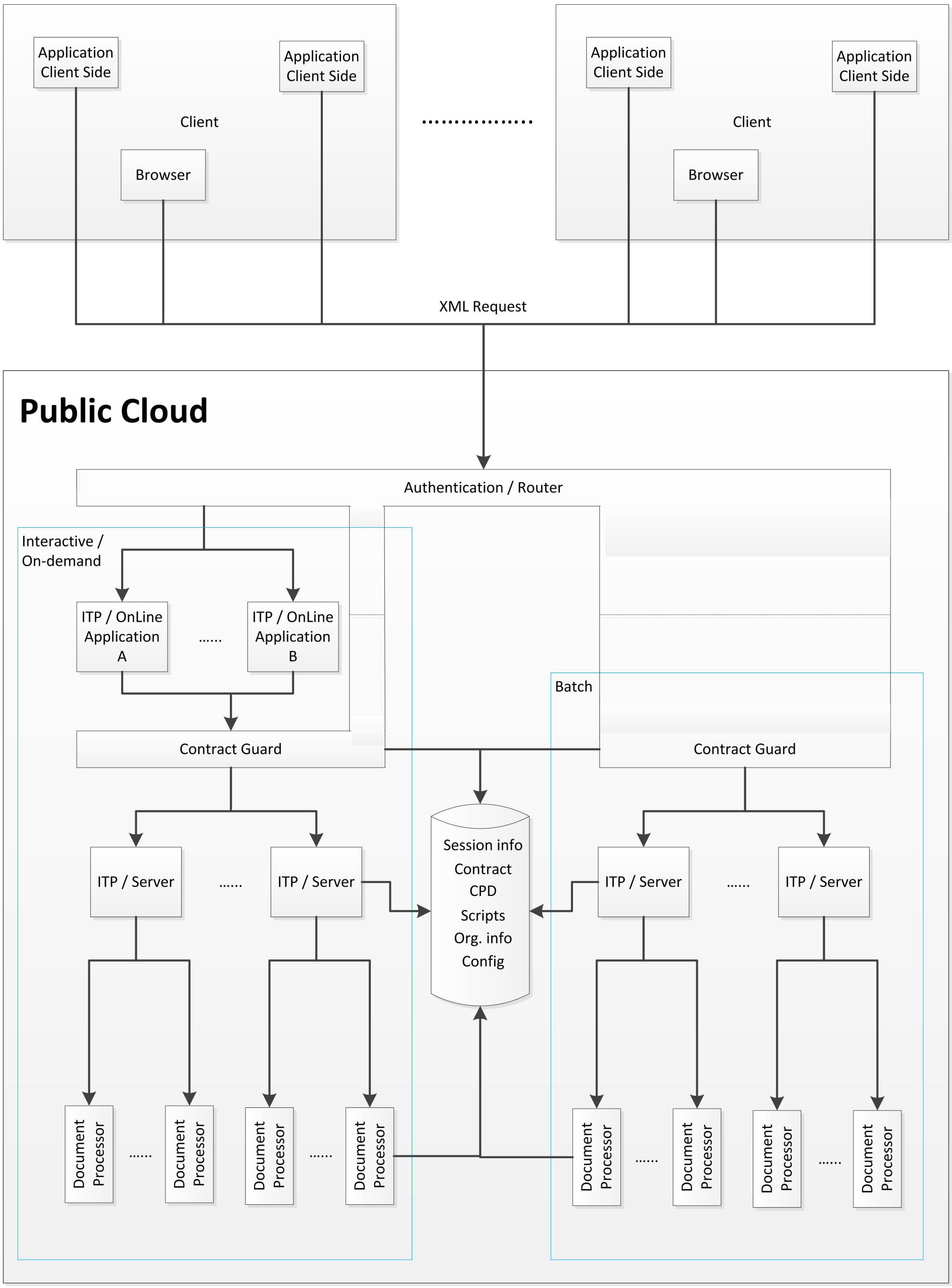

Figure 21: The architecture of the public solution

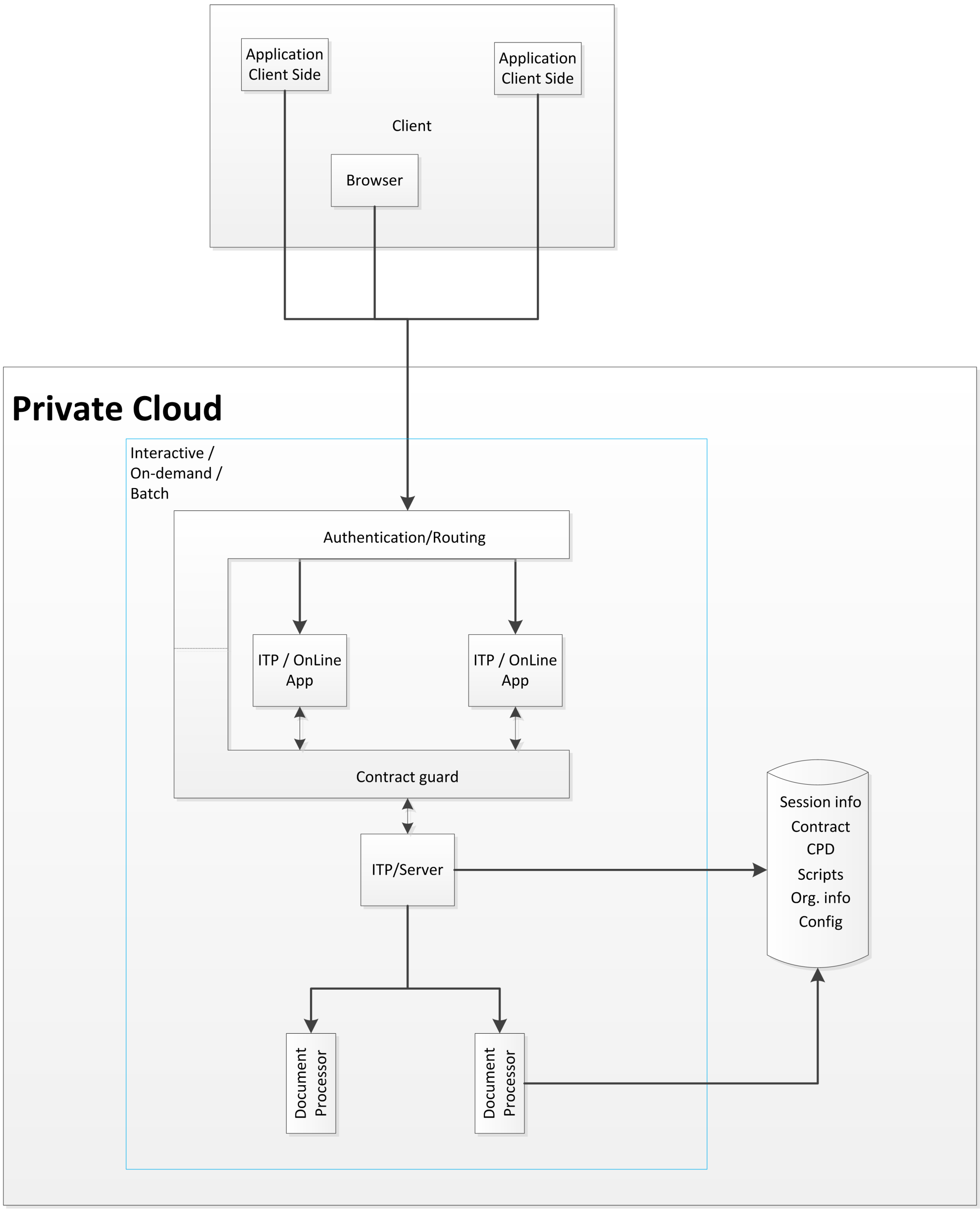

Figure 22: The architecture of the private solution

10 Proof of Concept

10.1 Testing limits

As already mentioned in chapter 2, cloud computing offers the possibility to use hardware more efficiently. Because of this, Aia wants to put multiple customers on the same hardware and in the same virtual machine. In order to find out what the limits are, some tests were performed. In table 4 the memory usage of a number of ITP/Server installations (including 1 DP each) is shown. The column "after 1 run" indicates the total memory usage of the all installations after 1 doc to pdf conversion. This conversion is done because it uses Microsoft Word. The conversion consists of a Word document that was located on the hard disk which was converted to a pdf file. After this conversion all document processor have their own Word instance up and running which of course uses memory as well. Based on the last column the conclusion can be that an ITP/Server installation with 1 document processor (including a running Word instance) uses approximately 26 MB of memory. One note to make is that in the set up with 290 installations, the second run failed with a Word Automation Server error. Furthermore there were some set ups with up to 500 ITP installations, but those installations failed once a doc to pdf conversion was done because of the Word Automation Server error. When only a simple model was created (this excludes the use of Word) by the installations, no problems occurred. The column "All installations" contains the memory usage of all installation together that are running (including a Word instance per installation). This value is calculated by subtracting the value in the column "Before" from the value in the column "After 1 run". The value per

Amount	Before	After start	After 1 run	All installations	Installation
1	1169	1181	1222	53	53
10	1178	1292	1470	292	29.2
20	1208	1410	1753	545	27.25
50	1249	1683	2512	1263	25.26
100	1263	2153	3781	2518	25.18
200	1304	3757	6878	5574	27.87
250	2114	4293	8670	6556	26.22
280	1884	4201	8950	7066	25.24
290	1927	4338	9666	7739	26.69

Table 4: Memory usage synchronous jobs (MB)

installation is calculated by dividing the “All installations” value by the “Amount” value.

10.2 Minimal solution

In the previous chapter the architecture of the ideal solution was described. Since this solution is a too big for a proof of concept, the key concepts are chosen and put in a minimal solution. The design of the proof of concept is shown in figure 23. Another reason for starting with a minimal solution is the time to market. When a smaller solution has to be created, it will be finished earlier and therefore can be used earlier. This also supports the idea that it is better to put a cloud product into the market in phases (once it is decided to start offering cloud services). This also fits the principle of “release early, release often”, which means that a release of a small version should be done as soon as possible and afterwards additional functionality can be added to the product. In order to finance the addition of new functionality the revenues of the initial version can be used.

The minimal solution does not include a customer database that contains the configuration of an ITP/Server installation. Nevertheless it contains a customer specific database for the content. The configuration for ITP/Server will be stored in ITP/Server itself. This is relatively simple because in the minimal solution every customer will have access to a fixed set of scripts and a fixed configuration.

Since every customer has its own ITP/Server installation it is also possible to store session information in the virtual machine as well.

It is also possible to remove the contract guard from the minimal solution because every customer will have its own ITP/Server and ITP/OnLine installation. Therefore ITP/OnLine knows to which ITP/Server the requests have to be sent. In fact, ITP/OnLine will be adapted to search the corresponding ITP/Server once a message has to be sent. In the proof of concept ITP/OnLine will get its ITP/Server location out of a database. A difference with the old situation is that ITP/Server can be configured at the ITP/OnLine application level. This means that every application can send requests to a different ITP/Server. In the previous situation the ITP/Server could be configured for all applications at once only.

Requests will be send to a single web service which does identification of the customer and gets the location of the corresponding ITP/Server web services interface out of a database. The request will be send to the corresponding web service interface of the ITP/Server installation for customer that is identified. For interactive jobs there is only one possibility to start. Since all applications are running in secure mode, there has to be a session created before the ITP/OnLine application can be started. This means that ITP/Server has to prepare a session first. The result of this is that all requests have to start at the general web services interface. In figure 23 this is shown by the black lines. The path of the black arrows is the path that will be followed in every situation. For interactive jobs this path will be followed as well. After this run, a session id will be given back to the customer's application. Now it is possible to follow the green arrow to ITP/OnLine, by adding the session id as a parameter to the URL. Different applications in ITP/OnLine have different URLs in IIS (Internet Information Services). Based on those URLs the requests will be routed to the different ITP/OnLine applications. Every customer has its own ITP/Server installation, this ITP/Server installation has its own web services interface which has the host and the port of ITP/Server configured in a configuration file. Those web service interfaces are exposed as applications in IIS with their own URL. Again the requests of a specific customer are send to the corresponding URL. By using ITP/OnLine in combination with the web services interface of ITP/Server, the only ports of the virtual machine that have to be exposed are port 443 for SSL connections (HTTPS) and a port for remote desktop (for ITP administrators only). In the proof of concept, authentication is based on Windows Authentication and Basic Authentication of IIS. By doing this, one customer cannot use the ITP installation of another customer. Once someone tries to access an installation that doesn't belong to him, there will an authentication error or there will be no response. Of course this is only secure when the credentials are unknown for other people than the customer.

The authentication mechanism as implemented in the proof of concept is only used to show that customers can be separated from each other. It is also undesirable to use Windows Authentication in the final solution.

The requirements mentioned in chapter 9, are met as much as possible. The availability requirement is something that has to be discussed with the provider since the provider manages the hardware on which the solution will run. The new code is written in `C#` and Java in order to meet the requirements. Furthermore there are no vendor specific elements implemented which prevents vendor lock-in. PDF generation is implemented and tested by installing Word on the Windows Server 2008 virtual machine. Data leakage is prevented by giving every customer its own installation and therefore its own ITP/Server work directory. By doing this, there can be no accidental data leakage. Sentia furthermore offers persistent storage within virtual machines so data loss will be prevented as well.

10.2.1 Security

As mentioned in previous chapters, security can be one of the biggest problems in cloud computing. Therefore this aspect is also taken into account when designing the proof of concept. As already mentioned, the authentication layer and contract guard are both responsible for identity and access management. Since in the proof of concept, the contract guard is integrated in ITP/OnLine, there had to be a slightly different solution for the authentication. In the proof of concept, the authentication layer is provided by IIS. A request of a customer is send (over a SSL connection) to a URL that is customer specific. This solves the routing problem since the URL belongs to only one ITP installation. Once the request is arrived, the credentials of that user will be checked. The URL points to a directory on the hard disk for which the user and the ITP administrator are the only persons that have access to it. These access rights are set in IIS. After the credentials are validated, the access to the system is granted or denied. The same mechanism is used for the ITP/Server web services interface. This interface is exposed as an IIS application as well. Again the only people who can use this interface, are the customer and the ITP administrator. So by setting access on the directories behind the URLs the customers can be prevented from (accidentally) accessing the installation of another customer. In the proof of concept this authentication method relies on Windows Authentication and IIS Basic Authentication. Because of this, every time a new customer is added, a user on the Windows Server is created. The remote desktop right for the user is disabled, so the only way to access the system is via port 443 at which the IIS authentication is running. By doing this the attack surface of the ITP platform is being reduced. Since authentication will be done when arriving at the port, the check at the gate principle is satisfied as well. The SSL connection guarantees confidentiality and integrity of the information that is send to the ITP Document Platform. In the proof of concept, a self-signed certificate is used which will not be seen as secure by web browsers. Because of this a “real” certificate has to be requested. Authentication is done when the system is entered, but it is also done when ITP/OnLine will be used. In the current version this is necessary because a session identifier has to be specified in the ITP/OnLine URL once it has to be used. Without this identifier, ITP/OnLine cannot be used. Although all requests are being sent over a secure connection, the URL will still be visible from outside and since this URL contains the session id, it is possible for a hacker to use this URL to enter the system. This can be solved by performing a post operation with the session id, but this also involves some changes at the customer’s side.

For every customer a repository can be installed. During this installation there will be a database for every customer created. This database has a customer specific login which means that customers cannot (accidentally) access another customer’s database. Within the repository client, a customer can continue creating users and giving them permissions by assigning a role to them. The only people having access to the database are the customer and the ITP administrator.

The proof of concept uses a single user for every customer. In such a situation the customer has to handle authorization in the application that is calling ITP services. Furthermore the customer account is not connected to customer systems and therefore there is no possibility for a SSO mechanism.

10.3 Steps to ideal solution

In the minimal solution SQL Server is included in the installation. Since SQL Server has relatively high license costs it can be attractive to use MySQL on a Linux machine instead. By creating such a VM, the license costs for SQL Server and Windows Server can be avoided. Sentia furthermore includes regular maintenance of such a VM in the hosting price.

In order create an ITP/Server installation which can work for every customer, all the configuration has to be put in a database or shared storage device. Once this is done, a script should be created that loads the correct customer configuration into ITP/Server at the moment when it is necessary. By doing this, an ITP installation can be temporarily customer specific which increases the level of multi-tenancy. Once a more dynamic solution is needed, there should be some kind of mechanism that tells ITP/OnLine to which ITP/Server it has to connect. Another possibility is to use the proxy (which was created for the proof of concept) that routes request to the correct ITP/Server installation. Telling ITP/OnLine which ITP/Server should be used, can be done by executing the batch file of the proof of concept. Nevertheless, there is still some monitoring mechanism necessary which automatically executes the batch file once it is needed. In the ideal situation it might happen that different users need different permissions on the ITP/OnLine applications. In order to accomplish this, there should be a role based IAM mechanism implemented. By doing this users, can be given a role and these roles can have permissions.

Another part in the proof of concept that has to be changed is the authentication mechanism. In the proof of concept, this authentication mechanism is based on Windows Authentication and IIS Basic Authentication. This has to be replaced by another mechanism that could integrate with the user management systems of a customer. By doing this, a customer can have a single sign-on mechanism instead of an extra account for ITP.

Something that has to be changed before the launch of the first version, is access to the local file system. In the current product it is possible move files over the local file system by using ITP/Server scripts, but also by using specific commands in the modeling language. ITP/Server scripts can be removed before putting an installation in the cloud but the modeling language has to be adapted before the first version of ITP can be placed in the cloud. All the commands that do something with the local file system should be removed from the cloud version of the modeling language.

For ISVs there have to be a few changes made to the solution. Since in the ISV situation, different customers will use the same ITP instance, the organization identifier is mapped to an environment in ITP/Server. By doing this, a repository can be set in every environment and therefore every group of users can have its own content.

In the situation when everything has to be put in a single database per customer, the ITP/Server web services interface has to be extended with an organization identifier (or the code has to be adapted that it can be extracted from another part). This has to be done because in this situation the scripts will be in the database as well, so when the organization is put in the script, it should already have the knowledge about the database that has to be used. Therefore the identifier should be placed in the request as a first class citizen.

When an ITP/Server installation will be shared by multiple customers, it is undesirable that it has to be restarted once some changes in the configuration are made. A solution for this problem is to create a new ITP/Server instance that contains the changes in configuration. Once this instance is created, the customers can be routed to the new instance. In this situation attention has to be paid to the jobs that were already running on the “old” ITP/Server instance.

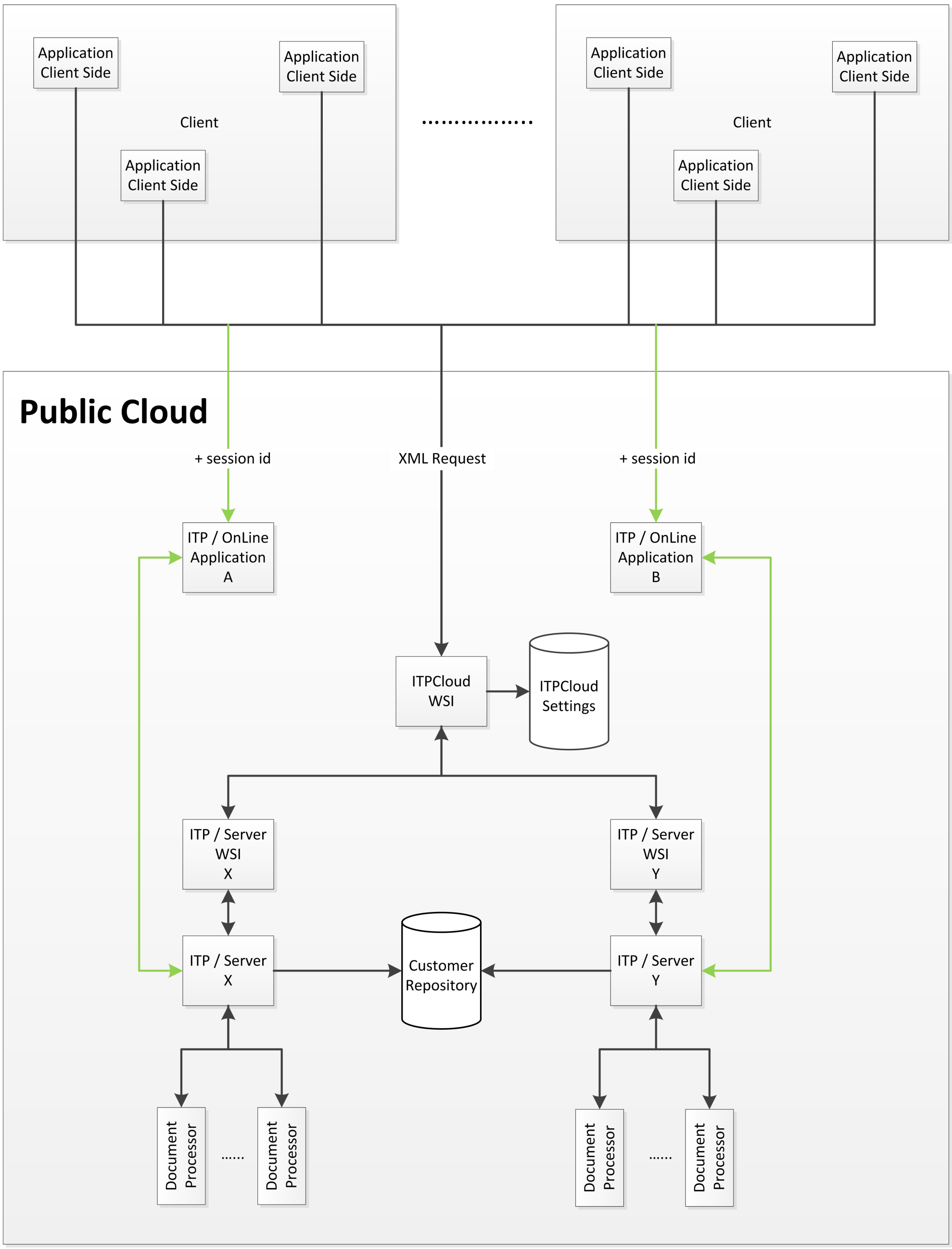

Figure 23: Design Proof of Concept

Part IV
Conclusion and Future Work

11 Conclusion

People often ask how they should implement cloud computing. The answer to this question is not that simple because it really depends from the business in which an organization is operating. There are a lot of models for implementing cloud computing but most often it is not enough to pick one. The best solution often consists of a combination of different models. This is exactly what makes it difficult to give an answer to the question how cloud computing should be implemented. It really depends on the business, size and requirements of the organization. One thing that is clear: an organization must have some arguments, other than because everybody does it, why it would start to use or offer (or both) cloud services.

Another conclusion that can be derived is that an organization should not want to have everything in the cloud. Not every application is suitable to run in the cloud. Sometimes it is just better to run applications on-premise, close to the other applications and databases. When a lot of interaction with other applications and databases in needed, it might be wise to stick to on-premise software. The reason for this will be that when a lot of interaction with other applications is needed, there is a high chance of getting too much latency. This means that the components of the total system should be loosely coupled.

Since an application in the cloud is supposed to be scalable, the architecture of the application must support this scalability. Based on the literature study and the case study, it is found that existing applications often have to be changed before they are suitable to be put in the cloud. One important aspect to get an application scalable is by making it stateless. Once an application is stateless, the advantages of the cloud will be present.

During the case study it appeared that licensing can be a real problem when putting products in the cloud. So when an organization wants to put its products in the cloud, it should really pay attention to the existing licenses that are necessary for running the installed software in a hosted environment. Very often other licenses are needed which has an impact on the costs. Most of the times it is not allowed to use the regular licenses in a hosted environment.

For a small organization it can be more interesting to start using cloud services because they often don't have their own IT department. This means that a large potential market exists of smaller organizations. This doesn't mean that large organizations aren't interested in a cloud solution, but for small organization the benefits might be even better. Large organizations often have to perform maintenance to their IT systems, but actually they don't want to do this. By using a cloud based solution, this problem will be solved.

When an application has to be able to handle peak loads, it will be a good idea to put this application or a part of this application (hybrid cloud) in the cloud. By doing this the hardware expenses for the customer/user can decrease dramatically.

12 Future work

Since there are some changes coming in the legislation it will be interesting to see how these changes will affect the world of cloud computing. The DPD as mentioned in chapter 4 is supposed to change in the near future. Furthermore different countries have different legislation about this topic, so a specific implementation can have different consequences in different countries. It will be interesting to find out what these consequences are. It would also be interesting to find out what happens in case of a conflict between Europe and the United States.

Existing cloud service providers are expanding their offerings rapidly. The result of this is that these offerings of different providers start to be pretty much the same. It will be interesting to follow this movement en keep track of the different offerings. In combination with the legislation it will be interesting to see whether there will be more and more European providers.

Since cloud computing is relatively new, it should be investigated whether the advantages and disadvantages can be confirmed. Especially the availability and security concerns should be removed. This doesn't mean that the technology is new, but cloud computing brings some new interesting challenges like multi-tenancy and identity and access management. Another shift can be found in the business models of an organization that starts to offer cloud services.

The biggest part of this thesis focuses on technology aspects and aspects that are interesting for a cloud service provider. Of course customers should also be willing to use a cloud service. So the customers should be prepared to use a cloud service. This is something that is not investigated within this thesis. The concerns in this thesis are based on literature only so it will be interesting to do some research within the "real" world.

For Aia Software it will be really interesting to do some further research concerning the liability in case something goes wrong. Once personal data will be accidentally leaked out of the ITP Document Platform, it has to be crystal clear for which damage Aia Software will be liable. When this is not the case it can turn into a disaster because there will be a risk of having to pay a lot of claims.

The proof of concept shows that the design will work in practice but it is far from a complete solution. The most important thing for Aia to start with, is implementing a secure authentication mechanism instead of Windows Authentication. The scripts that are created for the proof of concept, can be used by Aia when the final version is created.

13 Academic Reflection

13.1 Process

The process of writing this thesis took approximately 6 months. This is exactly the time that was planned for it. Going to the office every day really helped me to stay on schedule because there is less distraction from other things.

Cloud computing is a very trendy topic, this results in the fact that there are a lot of new articles written every week. It is very tempting to read all the new articles but at some point it was necessary to continue with the rest of the project. The same holds for the discussions, during the discussions a lot of new ideas and concepts were born. Nevertheless it is important to keep the scope of the assignment in mind because otherwise it will result in a never ending thesis since new tasks will be added continuously.

During the process of writing the thesis I have learned a lot about cloud computing. Since the thesis was written at Aia Software, I have also learned a lot about the questions that companies have when adopting cloud computing. It was really interesting to hear the ideas and opinions of my supervisor and management at Aia Software. These ideas and opinions really helped me to create a solution that fitted into the scope of the project.

The monthly meetings with my supervisors at the university and the weekly meetings with my supervisor at Aia Software were really useful for controlling the progress. Because of this frequency, small misunderstandings were noticed in an early stage. All this resulted in the fact that I didn't have a lot of problems to complete my thesis in time.

Furthermore it was interesting to see that there is not a simple solution for a single organization. While designing a solution for Aia Software, I discovered that there were a lot of solutions possible. The reason for choosing the one or the other is often not based on technical decisions, but on business decisions.

The purpose of the proof of concept was to show that my solution for Aia Software could really work. During the process of creating the proof of concept some small changes were made to the solution. This was mostly done to avoid complex programming. These changes only involved some shifts of functionality from additional components into existing components.

13.2 Product

This research is intended to contribute to the process of choosing a cloud solution. Once an organization has chosen to start using or offering a cloud solution, a lot of questions have to be kept in mind. This thesis should help in getting answers to these questions. Every organization has its own needs to which the solution has to be adapted. This thesis should give an organization information about specific requirements for the solution under specific circumstances. The case study provides an example of the process of adopting cloud computing (to offer services) within a specific company. All the considerations during the process of developing a cloud offering are described in the case study. Based on this information another person can create some idea about how to build a cloud solution for its own organization. Since the research was done within one organization, a critical note can be that some data is based on only one example. It could be that for other organizations other information is important and therefore could lead to other conclusions. In particular this holds for the information presented in the case study. In the other parts of the thesis I have tried to avoid this.

For the legal aspects one should know that it is purely based on literature. Since there are different opinions about this topic, I have tried to describe the most common opinion but it is not founded on existing legislation (since it doesn't exist).

While designing a solution for Aia Software, most of the theoretical part was applicable to the case study. Nevertheless it turned out that there were some aspects that were of less importance because of business values. An example is the dynamic scalability. A cloud application often adjusts to the work load of it and once there is less work load some resources will be shut down. This is something that is not done in the solution for Aia because the management decided that the customer should always have the capacity available for which he is paying.

With the proof of concept I have tried to show that my idea for Aia Software could really work. This proof of concept shows the working of the ITP Document Platform when a customer has its own software components running on shared hardware. Once all the software components are shared among different customers, there might possibly be some other undiscovered problems. Nevertheless the proof of concept is based on the solution that was described in the paper.

I think the solution for Aia Software that is presented in this thesis is pretty much a general solution. The application will be made stateless and afterwards the components will be loosely coupled to provide scalability. A general approach will be to put a new authentication layer in front of the application and route customers to the corresponding instances. This is exactly what happened in the proof of concept.

References

Scientific

[1] Abdulaziz Aljabre. Cloud computing for increased business value. *International Journal of Business and Social Science*, 3:234–239, 2012.

[2] Business Software Alliance. Bsa global cloud computing scorecard - a blueprint for economic opportunity. Technical report, BSA, 2012.

[3] Wanlei Zhou Alessio Bonti Ashley Chonka, Yang Xiang. Cloud security defence to protect cloud computing against http-dos and xml-dos attacks. *Journal of Network and Computer Applications*, 34:1097–1107, 2011.

[4] Unknown author. Being smart about cloud security. *Technology Review*, 114(6):75 – 76, 2011.

[5] Jantine de Jong. Patriot act maakt cloudopslag onzeker. *Automatiseringgids*, 10:22, 2012.

[6] Dave Durkee. Why cloud computing will never be free. *Communications of the ACM*, 53:62–69, 2010.

[7] Zach Hill, Jie Li, Ming Mao, Arkaitz Ruiz-Alvarez, and Marty Humphrey. Early observations on the performance of windows azure. In *Proceedings of the 19th ACM International Symposium on High Performance Distributed Computing*, pages 367–376. ACM, 2010.

[8] G. Kiewiet. Technische impact van hybride cloud computing op iam. Master's thesis, Open Universiteit Nederland, 2011.

[9] Qian Wang Kui Ren, Cong Wang. Security challenges for the public cloud. *Internet Computing, IEEE*, 16:69 –73, 2012.

[10] Andrew McAfee. What every ceo needs to know about the cloud. *Harvard Business Review*, 89:124–132, 2011.

[11] Rean Griffith Anthony D. Joseph Randy Katz Andy Konwinski Gunho Lee David Patterson Ariel Rabkin Ion Stoica Matei Zaharia Michael Armbrust, Armando Fox. A view of cloud computing. *Communications of the ACM*, 53:50–58, 2010.

[12] Siani Pearson. Taking account of privacy when designing cloud computing services. *Cloud '09*, May 2009.

[13] Stefan Ried, Holger Kisker, and Pascal Matzke. The evolution of cloud computing markets. Technical report, Forrester Research, 2010.

[14] Joep Ruiter. The relationship between privacy and information security in cloud computing technologies, 2009.

[15] Joep Ruiter and Martijn Warnier. Privacy regulations for cloud computing: Compliance and implementation in theory and practice. In *Computers, Privacy and Data Protections: an Element of Choice*. Springer Netherlands, 2011.

[16] Louis Jonker Ruud Leether, Elisabeth Thole. Usa patriot act haaks op privacywet eu. *Automatiseringgids*, 10:20–21, 2012.

[17] Subhajyoti Bandyopadhyay Juheng Zhang Anand Ghalsasi Sean Marston, Zhi Li. Cloud computing - the business perspective. *Decision Support Systems*, 51:176–189, 2011.

[18] Christopher Millard W Kuan Hon. Data export in cloud computing - how can personal data be transferred outside the eea?, October 2011.

[19] Christopher Millard W Kuan Hon, Julia Hörnle. Data protection jurisdiction and cloud computing - when are cloud users and providers subject to eu data protection law?, February 2012.

[20] Ian Walden W Kuan Hon, Christopher Millard. The problem of ‘personal data’ in cloud computing - what information is regulated?, April 2011.

[21] Ian Walden W Kuan Hon, Christopher Millard. Who is responsible for ‘personal data’ in cloud computing?, March 2011.

[22] Lizhe Wang, Gregor von Laszewski, Andrew Younge, Xi He, Marcel Kunze, Jie Tao, and Cheng Fu. Cloud computing: a perspective study. *New Generation Computing*, 28:137–146, 2010.

[23] Randy H. Katz Yanpei Chen, Vern Paxson. What’s new about cloud computing security. Technical report, Electrical Engineering and Computer Sciences - University of California, 2010.

[24] Dimitrios Zissis and Dimitrios Lekkas. Addressing cloud computing security issues. *Future Generation Computer Systems*, 28(3):583 – 592, 2012.

Books

[25] Tom Jenkins. *Managing Content in the Cloud.* Open Text Corporation, 2010.

[26] Erik van Ommeren and Martin van den Berg. *Seize the Cloud.* LINE UP boek en media bv, 2011.

Non-scientific

[27] Cloud computing, fundament op orde, 2012.

[28] *Oracle Cloud Conference*, 2012.

[29] Ruud Alaerds. Groeistuipen van cloud computing. Technical report, Heliview Consultancy, March 2012.

[30] Edwin Brok. De windwakken van de cloud-sector. *Computable*, 4, February 2012.

[31] David Chappell. The windows azure platform and isvs, July 2010.

[32] Mike Chung. Cloud compliceert toegangscontrole. *Automatiseringgids*, 7, April 2012.

[33] Cloud Security Alliance (CSA). Domain12: Guidance for identity and access management v2.1, April 2010.

[34] Cloud Security Alliance (CSA). Top threats to cloud computing, March 2010.

[35] ENISA. Procure secure, 2012.

[36] Brian Garvey. 2012 wordt echt het jaar van de cloud! *Computable*, 4, February 2012.

[37] Jonathan Gershater. Patriot act is not the first (nor likely) last law of its kind, January 2012.

[38] Sander Hulsman. Kosten stimuleren overstap naar cloud-software. *Computable*, 4, February 2012.

[39] Sander Hulsman. Overheid vreest voor veiligheid in de cloud. *Computable*, 4, February 2012.

[40] Mark Nicolett Jay Heiser. Assessing the security risks of cloud computing, 2008.

[41] Ellen Mesmer. Worries over patriot act drives ndp to cloud encryption, September 2011.

[42] Aad Offerman. Werken in de amazon ec2-cloud, September 2010.

[43] Aad Offerman. Google app engine: gevangen in de cloud, May 2011.

[44] Aad Offerman. Windows azure: huwelijk tussen desktop en cloud, July 2011.

[45] Hewlett Packard. Start small, grow tall: Why cloud now, May 2011.

[46] Ewald Roodenrijs. Private versus public cloud. *Computable*, March 2011.

[47] René Schoemaker. Cloudconflict tussen eu en vs op scherp gezet, January 2012.

[48] Marco van der Drift. Cloud computing verandert landschap voor ict-bedrijven, January 2012.

[49] Maurice van der Woude. De cloud is blind voor eindgebruikers. *Computable*, 4, February 2012.

[50] Erik Westhovens. Het einde van de traditionele werkplek. *Computable*, 4, February 2012.

Websites

[51] Amazon. `http://aws.amazon.com/ec2`, March 2012.

[52] Aia Software BV. `http://www.aia-itp.com/en/top/aboutaia.html`.

[53] Cisco. `http://newsroom.cisco.com/press-release-content?type=webcontent&articleId=574021`, March 2012.

[54] Microsoft. `http://msdn.microsoft.com/en-us/library/windowsazure/5229dd1c-5a91-4869-8522-bed8597d9cf5#BKMK_LoadBalancing`, April 2012.

[55] Microsoft. `http://www.windowsazure.com`, March 2012.

[56] Mark Russinovich. `http://channel9.msdn.com/Events/BUILD/BUILD2011/SAC-852F`, April 2012.

[57] SETI. `http://setiathome.berkeley.edu/`, June 2012.

[58] Wikipedia. http://en.wikipedia.org/wiki/Amazon_Elastic_Compute_Cloud, March 2012.

[59] Wikipedia. http://en.wikipedia.org/wiki/Azure_Services_Platform, March 2012.

[60] Wikipedia. http://en.wikipedia.org/wiki/Force.com, March 2012.

[61] Wikipedia. http://en.wikipedia.org/wiki/Google_App_Engine, March 2012.

[62] Wikipedia. http://en.wikipedia.org/wiki/IBM_cloud_computing, March 2012.

[63] Wikipedia. http://en.wikipedia.org/wiki/Shard_(database_architecture), July 2012.

Part V
Appendices

A Windows Azure Load Balancing

In this appendix some load balancing techniques will be explained [54].

A.1 Performance load balancing

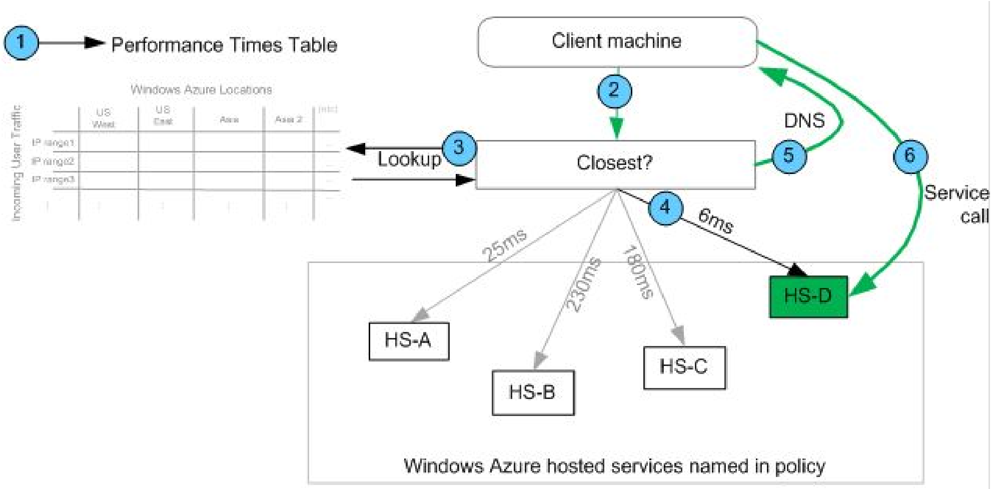

Figure 24: Overview of Performance load balancing [54]

The following numbered steps correspond to the numbers in figure 24.

1. The traffic manager infrastructure runs tests to determine the round trip times between different points in the world and the Windows Azure data centers which run hosted services. These tests are run at the discretion of the Windows Azure system.

2. Your Traffic Manager domain receives an incoming request from a client computer.

3. Traffic Manager looks up the round trip time between the location of the incoming request and the hosted services that are part of your policy using the table created in step 1.

4. Traffic Manager determines the location of the hosted service with the best time. In this example, that is HS-D.

5. Traffic manager returns the DNS name of hosted service D to the client machine.

6. The client computer resolves the DNS name to the IP address and calls the hosted service.

A.2 Failover load balancing

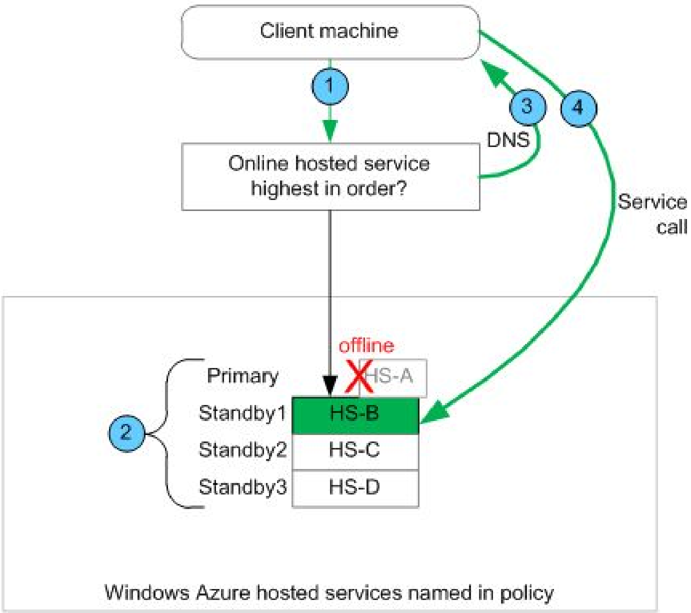

Figure 25: Overview of Failover load balancing [54]

The following numbered steps correspond to the numbers in figure 25.

1. Your Traffic Manager domain receives an incoming request from a client.

2. Your policy contains an ordered list of hosted services. Traffic Manager checks which hosted service is first in the list. It verifies that the hosted service is online. If the hosted service is unavailable, it proceeds to the next online hosted service. In this case HS-A is unavailable, but HS-B is available.

3. Traffic Manager returns the DNS entry to the client. The DNS entry points to the IP address of HS-B.

4. The client calls HS-B.

A.3 Round Robin load balancing

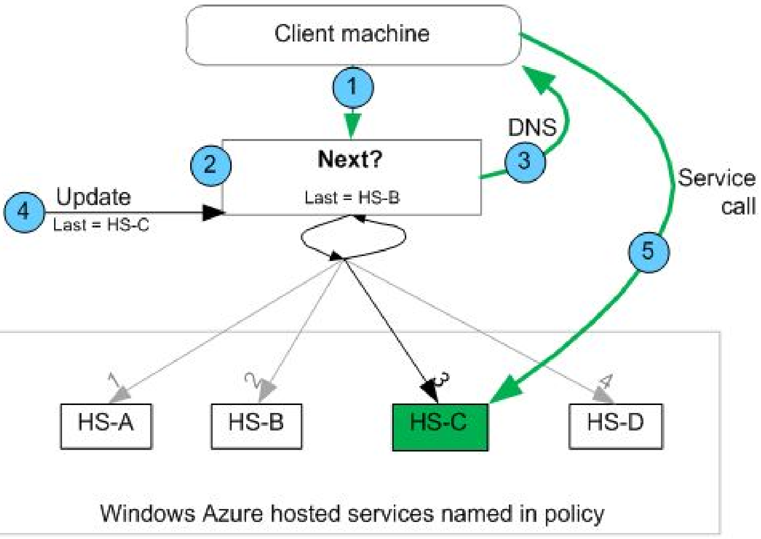

Figure 26: Overview of Round Robin load balancing [54]

The following numbered steps correspond to the numbers in figure 26.

1. Your traffic manager domain receives an incoming request from a client.
2. Your policy contains an ordered list of hosted services. Traffic Manager knows which service received the last request.
3. The Traffic Manager sends the DNS entry that points the next hosted service in the list back to the client computer. In this example, this is hosted service C.
4. The Traffic Manager updates itself so that it knows the last traffic went to hosted service C.
5. The client computer uses the DNS entry and calls hosted service C.

B Interviews/Conversations

B.1 Erik Meijer (10-2-2012)

After the Aia Awards I have had a short talk to Erik Meijer (head of the Cloud Programmability Team at Microsoft) about the cloud computing topic. This was just an informal chat to introduce myself. During this chat he told me that Aia was too much concerned about security. He told this because the evening before he had dinner with Paul Dirven (CEO Aia Software) and Erik thought that Paul was talking way too much about the security aspects of cloud computing. The best way to handle security is to define the security requirements first and then start thinking about how to implement it, instead of the other way around.
Erik also told me that it would be wise to use some existing clouds (Microsoft, Amazon, etc.) instead of designing a new cloud especially for Aia Software. The reason for this was that large companies like Microsoft and Amazon really think about where to put their data centers. Those data centers will be in places where there is little risk for earthquakes and floods. Furthermore those companies also really think about security so there is little risk about your data leaking to other parties. Since those large companies can buy storage, computation power and bandwidth in large volumes, those companies can offer a cloud at a much lower price than Aia can do when designing a new cloud.
A last remark of Erik was that I should have to total image of the solution in my mind. He said this because security might be more expensive in the cloud solution compared to the current situation but you can earn this back because some other costs can be cut.
I asked Erik if I could contact him when I had some questions and this is possible.

B.2 Erik Poll (10-2-2012)

Erik Poll is an associate professor in the Digital Security group at the Radboud University Nijmegen. I have had a really short chat with Erik Poll about my graduation topic. Erik told me that there are some people at the university that have knowledge about IT legislation.

- Mireille Hildebrandt
- Merel Koning
- Ronald Leenes

B.3 Mireille Hildebrandt (27-3-2012)

I have had some e-mail contact with Prof. dr. Mireille Hildebrandt about the legislation that is related to cloud computing. She holds the chair of Smart Environments, Data Protection and the Rule of Law at the Institute for Computing and Information Sciences (iCIS) at Radboud University Nijmegen. I have asked her a few questions to which she has tried to give an answer:

When is the Patriot Act applicable to a company?
The problem is that the Patriot Act is applicable to American organizations, even when they store their data outside the US. So when a government stores its data at Microsoft in the cloud with the guarantee that they don't leave Europe, the American government can still access this data. [1] [2]

What information can be asked by the government?
The government can as IP-addresses, identification, traffic data and possibly the content of communication. Under which conditions this will be possible depends on the jurisdiction. The conditions are concerned with the instance that can ask the data (police, prosecution), the type of "crime", whether there is a court order and the duration (real time or not).

Is there European legislation that prevents this?
Relevant is article 8: European Convention on Human Rights (the right for privacy), the Data Protection Directive, the Data Retention Directive (which requires the government to store traffic data for a period of 6 months to 2 years to give justice the possibility to ask for this data). Extremely relevant is the recently published Data Protection Directive.

Furthermore Mireille has send me a paper about the legal framework for cloud computing.

B.4 Mireille Hildebrandt + Merel Koning (30-3-2012)

After my weekly meeting with Marko I went to Mireille to ask here some additional questions based on the previous interview. After 20 minutes she took me to Merel Koning who is her PhD student to help me further. During the discussion it became clear that Aia Software BV currently does not fall under the Patriot Act despite of having a sister in the US. Merel expected that US government could ask data from Microsoft and therefore from Aia and its customers as well.

[1] http://webwereld.nl/nieuws/108909/cloudklant-dumpt-microsoft-vanwege-patriot-act.html

[2] http://blog.fpweb.net/usa-patriot-act-cloud-hosting

When a company is under suspicion they don't have to hand over their data because someone does not have to cooperate at its own conviction. When a person inside the company is being researched a company will have to hand over its data. Once the Dutch government starts helping the US government, all Dutch companies will have to comply.

B.5 Jean-Claude Grattery (26-4-2012)

In a phone call with Jean-Claude Grattery from Microsoft I have discussed some functionality from Windows Azure. The results are in this section.
In Windows Azure a web and worker role can be combined in a single compute instance but this is not done very often.
When an instance of a specific role is added, a new compute instance will be attached to it. The starting time of an instance in Windows Azure is 2 minutes.
It is not automatically possible to run the code from the cloud solution in a private data center. The biggest part can be re-used but there will be some modifications necessary since there exists no such things as blob storage, queues and tables for an on-premise solution. Furthermore it will be necessary to install SQL Server when SQL Azure is used.
A request in Windows Azure will be handled only once. The visibility time-out of a task in the queue can be adjusted at runtime.
Windows Azure does not contain anything to convert doc files to pdf files.
In a VM role it is not possible to store persistent information so it is not a good idea to install for example SQL Server in a VM role. A VM role can be used when a lot of additional packages have to be installed in order to avoid a lot of installing when an instance is started. In the future persistent storage will be added to the functionality. The customer has to take care of licenses that are necessary.
The use of cache can generate a lot of performance improvement.
Load balancing is done automatically in the entire application except from the load on SQL Azure databases. For databases always the primary database will be chosen. When this primary database is not available, the secondary will be chosen and when that one is not available the third copy will be used.

B.6 Ian Zein (19-6-2012)

I have had e-mail contact with Ian Zein from the hosting provider Sentia. I have asked him some questions about the offering and working of Sentia. This information was necessary to compare the Sentia offering to the offerings of other providers.

Remark
First a good thing to remember: we work as well with our own private cloud (the

Sentia Cloud) as well with the clouds of Amazon and Rackspace. Depending on the specific requirements we select the one or the other. Most of our customers run mission critical applications and therefore the choice will often be our private cloud. We simply have much more control over the capacity and how other customers can pull down the performance. Amazon and Rackspace have advantages when it comes to global scaling, but when it comes to a problem (which can mean completely down, or poor performance) it is very difficult to have some influence on the solution. To prevent this, there has to be such a large infrastructure that the costs will not be interesting anymore. Since as well the Amazon cloud as the Rackspace cloud are in the news (because of failures) on a regularly basis, we are a bit cautious to put everything there.

Can Aia add additional capacity on its own?
We offer fully managed hosting so this is not comparable to a cloud provider at which everything has to be managed by the customer. We believe that it is our management that adds a lot of quality to our service, that's why changes in capacity will be made by us.

How fast can additional capacity be added?
This can be done very quickly. In most cases this can be done during the same day as the request. Of course this depends on the exact specification of the expansion.

Is there a limit to the size of a server?
Yes, 128 GB of memory and 8 vCPUs.

How will back-ups be managed?
A VM image will run on a RAID-10 array which gives quite some redundancy. Furthermore there will be a daily off site backup and a quarterly offset snapshot.

Is data in a VM stored persistently?
Yes (in contradiction to Amazon).

How do database servers and corresponding licenses look like?
Software licenses are a special thing. We are renting licenses when a customer doesn't have them by itself. When an organization is renting services, it is not allowed to use regular (buy) licenses. You have to start using SPLA licenses. Since Aia wants to offer a SaaS solution, SPLA licenses will be involved. A full chapter could be written about this topic.

Is there any additional functionality available?

Load balancing will be offered by using a central solution. Auto scaling has to be provided by the application/infrastructure that will be created for Aia.