UNIVERSITY OF VAASA

FACULTY OF TECHNOLOGY

SOFTWARE ENGINEERING

Anna Katajamäki

APPLICATION REFERENCE ARCHITECTURE IN PUBLIC CLOUDS

Master´s thesis for the degree of Master of Science in Technology submitted for inspection, Vaasa, 30 September, 2014.

Supervisor	Prof. Jouni Lampinen
Instructor	D.Sc. (Tech.) Sakari Palko, D.Sc. (Econ.) Jari Töyli

FOREWORD

This thesis is made for Nordea Bank Finland Plc's unit called Midas & Infrastructure. I want to thank Nordea Bank Finland Plc and especially Teppo Jansson for the opportunity to do my master thesis.

I want to express special thanks to my instructor Sakari Palko from Nordea for finding this subject for this thesis. I want to thank Sakari for all the help and guidance he gave me throughout the thesis work when it was needed and also for his attitude that motivated me to finish the work.

I want to thank Jari Töyli and Jouni Lampinen from University of Vaasa for supervising this thesis.

I want to thank my family for their support during my studies and especially when I was writing this thesis. I want to also thank all my friends for making all these years as a student memorable.

Espoo, 24.9.2014

Anna Katajamäki

TABLE OF CONTENTS **page**

ABBREVIATIONS

API – Application Programming Interface

AWS – Amazon Web Service

CDN – Content Delivery Network

EC2 – Elastic Cloud Compute

IaaS – Infrastructure as a Service

NIST – National Institute of Standard and Technology

PaaS – Platform as a Service

RDS – Relational Database Services

SaaS – Software as a Service

VM – Virtual Machine

LIST OF FIGURES **page**

LIST OF TABLES **page**

VAASAN YLIOPISTO

Teknillinen tiedekunta
Tekijä: Anna Katajamäki
Diplomityön nimi: Application reference architecture in public clouds
Valvoja: Prof. Jouni Lampinen
Ohjaaja: TkT Sakari Palko, KTT Jari Töyli
Tutkinto: Diplomi-insinööri
Koulutusohjelma: Tietotekniikan koulutusohjelma
Suunta: Ohjelmistotekniikka
Opintojen aloitusvuosi: 2008
Diplomityön valmistumisvuosi: 2014 **Sivumäärä:** 87

TIIVISTELMÄ:

Pilvipalveluista kiinnostuminen ja niiden käyttäminen yrityksien toiminnassa on lisääntynyt viime vuosina. Pilvipalvelut ovatkin jo tuttu ilmiö tavallisten kuluttajien keskuudessa ja tavallinen kuluttaja törmää niihin melkein päivittäin esimerkiksi käyttämällä Googlen palveluita. Yrityksien kohdalla on kuitenkin vielä monta ongelmaa ratkaistavana sekä ennakkoluuloja siitä, sopivatko pilvipalvelut yrityksen toimintaan.

Tämän diplomityön tarkoituksena on rakentaa sovelluksen referenssiarkkitehtuuri julkiseen pilveen. Työn tutkimuskysymyksinä käsitellään sitä, mitä tulee ottaa huomioon kun sovelluksia siirretään julkiseen pilveen. Lisäksi keskitytään perusperiaatteisiin ja standardeihin ja niiden mahdollisiin muutoksiin. Tutkimus suoritettiin tekemällä markkina-analyysi, johon otettiin mukaan tarkempaan tarkasteluun kolme alan isointa toimijaa. Lisäksi kolmeen pienempään toimijaan tutustuttiin pintapuolisesti. Toimijoista kerättiin tietoa eri lähteistä, joihin kuului muun muassa internetsivut, artikkelit ja uutiset.

Tutkimuksessa selvisi, että nykyisten sovelluksien ja pilviympäristön vaatimuksien välillä on eroja. Moni näistä eroista liittyy palveluntarjoajaan ja siihen, että palveluntarjoaja on vastuusta pilvialustasta kun taas asiakas vain itse ohjelmistosta. Tämä johtaa moneen erilaiseen ongelmaan, joka ei aina ole täysin asiakkaan käsissä. Erot tulisivat ottaa huomioon ennen kuin ohjelmistoja siirretään pilveen. Pilvipalveluiden standardointi on vasta aluillaan ja se kaipaisikin pikaisesti toimia. Tällä hetkellä markkinoilla ei ole yhtä yhteistä standardikokoelmaa, jota kaikki noudattaisivat. Standardien olemassa olo auttaisi muun muassa siinä, että asiakkaat voisivat vaihtaa palveluntarjoajaa helposti ja vaivatta kun kaikki palveluntarjoajat noudattavat samoja standardeja. Standardien tulisikin siis muuttua paljon tämän hetkisestä tilasta.

AVAINSANAT: pilvipalvelut, julkinen pilvi, sovelluksen referenssiarkkitehtuuri

UNIVERSITY OF VAASA
Faculty of Technology
Author: Anna Katajamäki

Topic of the Thesis:	Application reference architecture in public clouds
Supervisor:	Prof. Jouni Lampinen
Instructor:	D.Sc. (Tech.) Sakari Palko, D.Sc. (Econ.) Jari Töyli
Degree:	Master of Science in Technology
Degree Programme:	Degree Programme in Information Technology
Major:	Software Engineering
Year of Entering the University:	2008
Year of Completing the Thesis:	2014 **Pages:** 87

ABSTRACT:

Interest towards cloud computing and its usage in companies core processes has been rising during the last decade. Cloud computing is already a common phenomenon among ordinary consumers and one can run into them almost daily for example by using Google's services. But when it comes to companies, there are still many challenges and prejudices of cloud computing to be solved before companies can utilize it.

The aim of this thesis is to build application reference architecture in public cloud. The research questions are investigating what one has to take into consideration when moving applications to public cloud. The other question where this thesis concentrates is standards and principles in cloud computing and if there are any changes or challenges regarding them. The research was based on market analysis that consisted of three biggest service providers in the market that were investigated in detailed way. Three smaller providers were also introduced only superficially. Using different sources like websites, articles and news, the information of service providers was collected.

As a result, gaps between the current applications and requirements for the cloud environment were found. Many of them were related to the fact that service provider is responsible for the platform and customer only for the application. These gaps should be solved before moving applications to cloud. Standardization is also needed since at the moment there is not one single source that everyone is following. It will also help customers for example to change cloud service provider easier when all of them are following the same standards. At the moment, the standards have to change quite a lot from the current state.

KEYWORDS: cloud computing, public cloud, application reference architecture

1. INTRODUCTION

1.1 Purpose and background of the study

Cloud computing and clouds have been a new trend throughout the 21st century. Many consumers are already using cloud services and we can see the number of them rising all the time. For many ordinary consumers the first touch to cloud computing and cloud services comes from using different services from Google, Apple and Dropbox. However businesses have been more unwilling to take advantage of them and commit their core business process applications to it. This is because cloud services are seen to have problems with security, data protection and risks of service interruption. (White 2014: 30; Marks & Lozano 2010: 27.)

Forrester research has forecast how cloud computing will grow. The global cloud market is estimated to grow from $40.1 billion to $241 billion in 2020 and especially the public cloud computing usage will increase. The market is forecast to grow from $26 billion in 2012 to $160 billion in 2020. (Columbus 2012.)

Traditionally businesses are not so used to in using cloud computing in their core business processes. In new start-up companies the situation is totally opposite and they are fully relying on clouds since cloud can help them grow and scale quickly. But for example financial organizations in EU are still worried about using public cloud especially regarding the uncertainty of regulatory obligations. (Business Insider 2014; Jacobs 2013.)

Cloud is becoming to be an important way to store, share and seek different kind of data in a fast way regardless where the user is located physically. In 2014 IBM moved its supply chain and its applications to the cloud. One important thing that they learned was that applications should be built on the cloud and not just move the old applications there. They think that one should not try to retrofit them. (Ward & Gopal 2014: 27.)

In 2012 Dutch bank ING Group informed that they are going to build a hybrid cloud data centers, combination of public and private cloud. ING Group believes that working in a cloud environment will bring new products faster to market and create cost savings. This means that company should use more cloud services and less legacy applications. It would also bring technology and business closer together. Lots of financial organizations are having legacy in their environment which means that for example they can't respond to demand as fast as they would like to or move quickly as the environment and situations changes. Cloud would help the companies to be more agile and respond to the demand and its changes in a better way. In 2012 ING said that they run already 5% of their functions in the cloud. In 2013 Bank of Australia moved almost all information that was in websites to Amazon Web Services. (Jones 2012; Jacobs 2013.)

In 2013 Dutch banking regulator De Nederlandsche Bank (DNB) announced that financial and banking organizations can use Amazon Web Services, especially public cloud services. It can be used for example across retail banking, websites and also mobile applications. DNB sees cloud computing as one option to outsource. As a result for example many Dutch banks started to run their retail banking platforms in cloud. (Essers 2013.)

Traditionally the IT systems have been managed internally but cloud help companies to reduce cost by outsourcing this to cloud service provider. They can manage and access for example the infrastructure over the Internet. This is not the only benefit that companies can get from moving to the cloud. The increase in workforce mobility and productivity are one of the big benefits along the fact that companies will have savings in IT hardware and real estate costs. (BCS The Chartered Institute for IT 2012: 19.) This way companies no longer need to have their own servers and server rooms when the service provider will take care of that. "*The availability of IT services and the pricing model based on the usage makes cloud attractive for companies* (Choudhary & Vithayathil 2013: 68)."

Also the delivery of applications is a major part of why companies would benefit from cloud computing. Nowadays companies have different ways to deliver applications and this can be a huge problem for them and of course costly. With standardized cloud computing services, the delivery of applications can be simplified. Integration and other delivery problems that companies might face now can be reduced with the use of cloud computing.

Since cloud computing is a trend in IT world with many benefits and one can see it in one's everyday life, it is important to look into the matter and see what companies need to take into consideration when moving its applications to cloud. Many companies have old applications that they are using in their everyday work tasks. In order to fully take advantage of cloud computing and its benefits, companies need to take a look on what they need to do with the applications in order for them to work in cloud. It is important to have an image of what is going to happen if an application is moved to the cloud, for example what is possibly going to break down and what changes should be done in the application.

1.2 Research Challenge and Problem Definition

The aim of this thesis is to create application reference architecture in public clouds and create recommendations and best practices on what one should take into consideration in the application point of view when moving applications to cloud. On definition says that reference architecture

> "*is, in essence, a predefined architectural pattern, or set of patterns, possibly partially or completely instantiated, designed, and proven for use in particular business and technical contexts, together with supporting artifacts to enable their use. Often, these artifacts are harvested from previous projects* (Reed 2002)."

Research questions are the following.

1) What has to be taken in consideration when moving applications to public cloud? What consequences does the moving have in the applications?

2) Does something change in the basic architecture principles and what kind of effect it has? What has to be taken into consideration regarding standards?

The first research question examines what happens to the applications when they are moved to the cloud. Is something going to break down or what else could happen? The second research question concentrates more on the architecture and standards and what kind of effect the moving will have for them.

1.3 Structure of the Study

This thesis consists of six chapters: Introduction, Cloud computing, Market analysis, Current application architecture gap, Recommendations and best practices and Conclusions. In this chapter the topic of the thesis is introduced and research method is explained. In the next chapter cloud computing is defined and its benefits and challenges are introduced. The architecture of cloud computing is also introduced and the situation with standards are presented. In the chapter three the market analysis is done. Market analysis will focus on the main providers of cloud platforms and the reference architecture available.

In the fourth chapter the current application architecture gap will be discussed based on the theory and findings. In chapter five the recommendations and best practices are given based on the market analysis and the findings in chapter four. The last chapter is for conclusion

and summary part. In this chapter the important findings and issues are discussed and suggestions for future research are provided.

This thesis work has been done to Nordea Bank Finland Plc's unit called Midas & Infrastructure. The work itself is not Nordea specific and the study has been conducted in general level.

1.4 Research methods

This thesis is based on qualitative research. The research is based mostly on previous literary researches and other literature about the topic. The research method used in this thesis is market analysis. Information from different companies and their services are gathered from companies' websites and other publications that are available. Information of different service providers will be collected regarding cloud solutions and they are compared between each other.

2. CLOUD COMPUTING

2.1 Definition and history

According to National Institute of Standards and Technology's (NIST) definition created by Timothy Grance and Peter Mell (2011: 2) cloud computing is defined as following:

> "*Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources, that are for example networks and servers, that can be rapidly provisioned and released with minimal management effort or service provider interaction* (Grance & Mell 2011: 2)."

We can say that cloud computing has five different characteristics. We can also divide it into three different service models. Lastly, it also has four different deployment models. (Grance et al. 2011: 2.) These can be seen in figure 1.

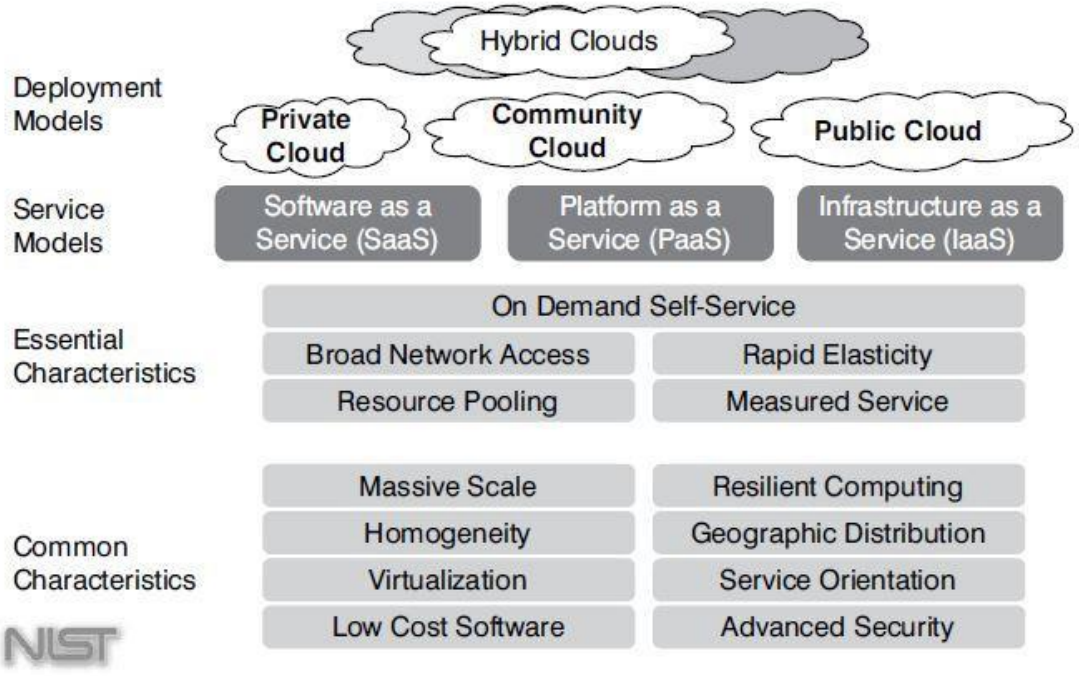

Figure 1. The NIST cloud computing definition framework (Williams 2010: 8).

In figure 1 cloud computing definition framework is seen as it is defined by National Institute of Standards and Technology. The figure gives an idea of what cloud computing is and what the different models and characteristics are. In this work we are going to concentrate only in the PaaS model from the three different service models. From the different cloud types we will only concentrate in public cloud as a deployment model even though there are few other cloud types such as private and hybrid cloud. (Grance et al. 2011: 3.)

NIST has created a standard definition for cloud computing but multiple industry forums have created other definitions as well. For example Gartner is defining cloud computing

> "*as a style of computing in which scalable and elastic IT-enabled capabilities are delivered as a service to external customers using Internet technologies* (Gartner 2009)".

The five attributes Gartner presents are service-based, scalable and elastic, shared, metered by use and uses Internet technologies. One other example is Open Cloud Manifesto that is created by many companies in the industry like IBM. However, Amazon and Google were not part of it even though they are big actors in cloud computing industry. The Open Cloud Manifesto points out characteristics and challenges of cloud computing and also presents six principles that should be followed. (Gartner 2009; Urquhart 2009; The Open Cloud Manifesto 2014.) However in this thesis the definition created by NIST is used as a definition for cloud computing.

In the table 1 below the development of IT industry is linked to cloud computing's features. In the development column the current trend in companies is presented and in the last column the answer on how the trend is feasible with cloud computing is introduced.

Table 1. IT development in companies linked to cloud computing's features. (Heino 2010: 29.)

Development	Situation	Why cloud computing?
The increasing complexity of technical environments	Maintenance costs are high and from time to time there are expensive and huge investments. The automated data center might be located far away and it might be difficult to get there.	The idea behind cloud operating model is buying the service. This means that the need for the right knowledge and development of technical environment are smaller, maybe not even needed. One doesn't need to have high investment in the beginning.
Making supply more towards commodity products	Technical environments are developing in a way that they get more and more similar to each other. Since customers are favoring big providers the demand is directed to fewer parties.	Cloud computing is commodity product and by producing it with high volume the benefits are met. These benefits the customer itself could not get.
The emergency of economic thinking in IT	Management hopes that all fixed costs are reduced but IT doesn't have tools or procurement model for it. Saving electricity has been discussion topic for some time but in the absence of incentives it hasn't proceed to real actions.	For the cloud provider the cost of one server is so small that services can be sold without any fixed costs. The cloud provider has a real incentive to reduce the use of electricity.
Participation	Products are becoming more and more intangible and to be able to sell them there has to be new tools and different way of thinking in IT area.	Cloud computing doesn't require investments so IT function can offer multiple applications to smaller user groups. Analytical decision-making systems and observational data can be purchased from the cloud.
Networking of physical world	Most of the equipment, raw material reserves and domains are going to be connected to a network. This generates new businesses. Companies are hoping that they'll have same kind of operating model as cloud computing.	Cloud computing has potential. The investment money will go to its development.

Public cloud

Public cloud is a type of cloud where more than one organization shares the same infrastructure and where the cloud service provider sells shared computing resources. The cloud exists on the premises of the vendor and it can be accessed through Internet. Public cloud is a shared resource between the users. Users will locate their applications on the same infrastructure and even though they use the same infrastructure, the applications are separate and they are protected from each other. The company who buys the service won't have any control over the physical environment and this has raised a question if the service provider could access company data without permission to do so. (McDonald 2010: 20; Gendron 2014: 54.)

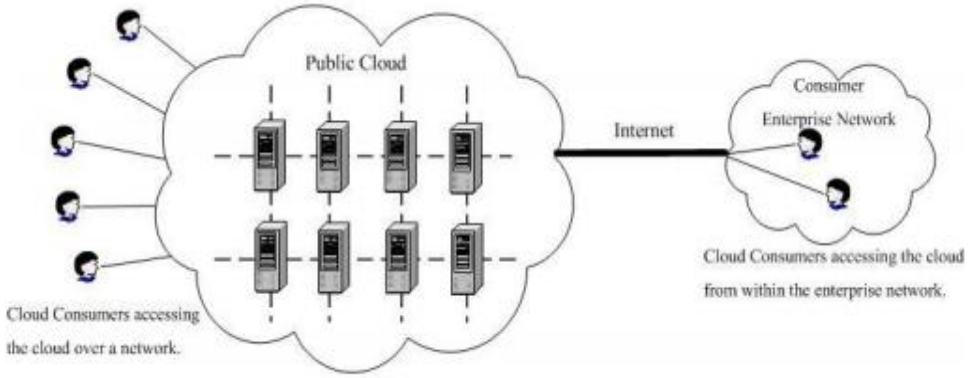

Figure 2. Public cloud (Liu; Tong; Mao; Bohn; Messina; Badger & Leaf 2011: 10).

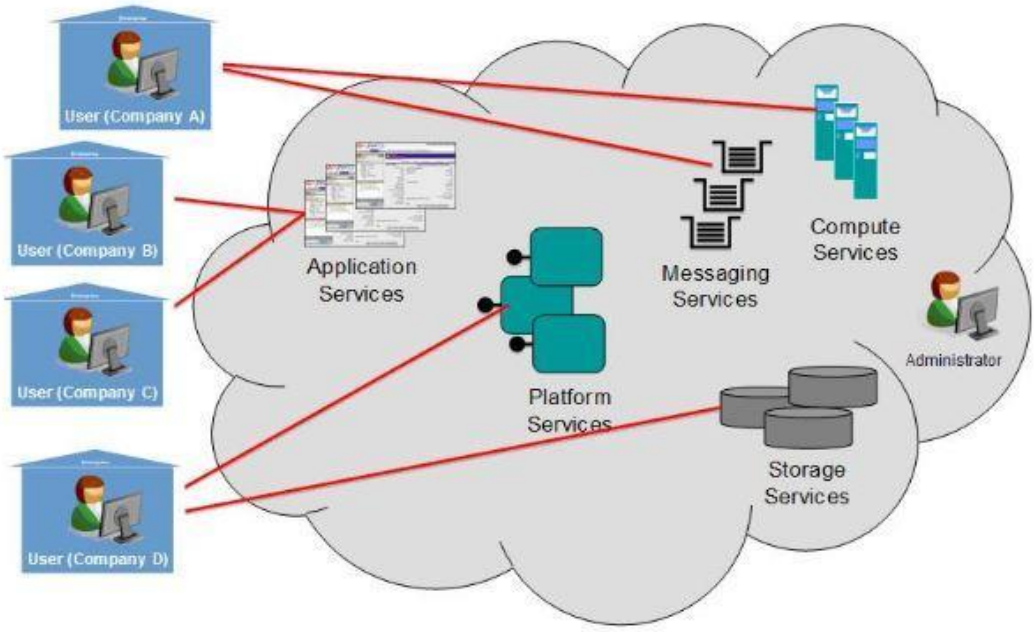

Figure 3. Public cloud (IBM 2010).

Figure 2 shows the idea of a public cloud. Cloud customer, the consumer, can access the cloud from company's network through Internet. At the same time many other consumers can also access the cloud. In figure 3 the public cloud is presented in a more detailed way. One can clearly see that one of the key characteristics is multi-tenancy. (IBM 2010.)

According to National Institute of Standards and Technology (NIST) (Grance et al. 2011: 2–3) cloud computing includes three different service models: Software as a Service (SaaS), Platform as a Service (PaaS) and Infrastructure as a Service (IaaS). In this work, we are only concentrating in the Platform as a Service –model.

Platform as a Service – PaaS

Platform as a Service is a model where the service provider has virtual server environment from where the provider allocates services to the customer according to the demand. This

means that the resources are cleaved into standard size blocks that the customers can utilize. The customer uses the capacity and tools of the PaaS cloud machine through the programming interface API. The customer can also make applications itself which utilize cloud machines or have them done by some application provider. (Heino 2010: 51.)

PaaS model is the most useful for the customer who can itself build up the applications. The interfaces in PaaS are well known and lightweight. Customer's user interface is software development tool and some kind of control console that is in the service. The end user can access the application through browser. Google, Microsoft and Salesforce have the best PaaS implementation. Google has its own development environment where one can create applications that can be run in Google AppEngine service. In Microsoft's Azure the applications are built in Microsoft's Visual Studio. (Heino 2010: 51–52.)

PaaS model offers a platform on which one can develop applications and one can test and maintain the application there. The development work will be simpler when one doesn't have to take care of the infrastructure and huge amount of the functionalities are available in ready modules and software interfaces. When using platforms the development work will become faster, cost-effective and the end result is scalable to the massive numbers of users without any extra work. Some disadvantages in PaaS model are that the user will get stuck with the same provider, there are new competence requirements in development work and maintenance and of course there are concerns about information security. (Salo 2012: 24.) It would be good for large companies to have also a second alternative for the service provider just in case something happens.

However the problem around being stuck with one provider has been noticed. This problem could even make the cloud adoption more difficult if there is not a way to integrate data and applications between different clouds. Standardization and interoperability would have a significant impact on that. With standardization customer can change the vendor whenever needed. Situations were one has to always stay with the same vendor disappears. There are

many cloud standardization projects with different focuses. Some focus on authentication and data access and some how the different parts could work together. CloudAudit and Cloud Computing Interoperability Forum are examples of these projects. For example the latter concentrates on building a framework for cloud platforms. (Parameswaran & Chaddha 2009: 10, 25; Lewis 2012: 5.)

Platform as a Service will also change the way of thinking in software development. The agile software development methods are easily applicable when reliability, scalability and concerns in platform's maintenance and updating are the responsibility of the provider and the company is responsible only for creating the code. Also the development work and implementing new ideas are easy and fast. (Salo 2012: 25.)

> "*PaaS systems typically include some or all of the following features:*
>
> - *browser-based development environment for creating databases and editing application code – either directly or through visual, point-and-click tools;*
> - *built-in scalability, security, access control and web service interfaces;*
> - *easy integration with other applications on the same platform;*
> - *tools for connecting to applications outside the platform's cloud;*
> - *tools for designing web forms, defining business rules and creating workflows* (Williams 2010: 12–14)."

In the figure 4 the different models are presented and how the responsibility is divided between the customer and the service provider. When looking at the PaaS model we can see that customer is then locked into a particular system that can be for example Google AppEngine, Microsoft Azure or Force.com where the customer has handed out the flexibility and control to provider. (Rutland 2012: 9.) The figure 4 illustrates that the service provider is responsible of everything that has to do with the infrastructure of the cloud and everything that has to do with the platform. Customer is only responsible for the application and logic behind it.

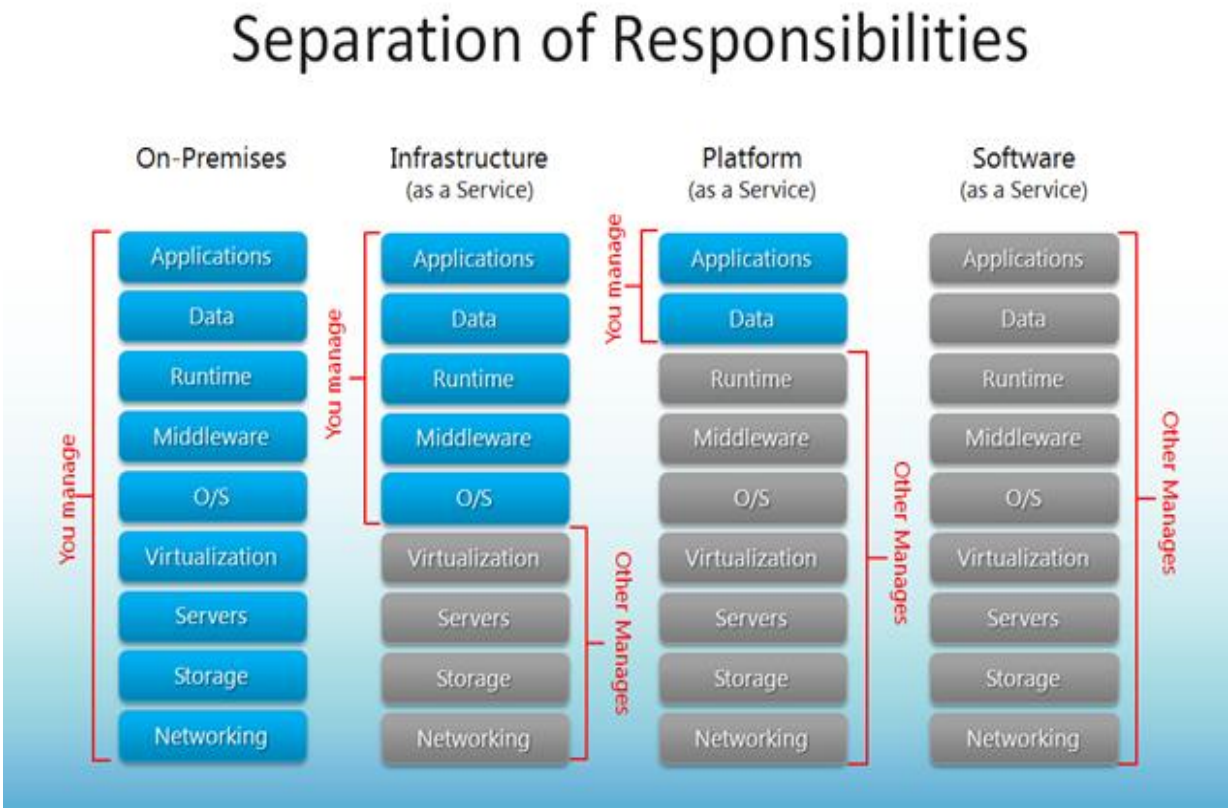

Figure 4. Separation of responsibilities between different cloud computing models. (Remde 2011.)

2.2 Characteristics

NIST (Grance et al. 2011: 2) describes cloud computing with the following characteristics: on-demand self-service, broad network access, resource pooling, rapid elasticity and measured service. Next each characteristic is explained in a more detailed way.

On-demand self-service

On-demand self-service means that one can get computer resources in use if needed and stop using them whenever without having to inform service provider. Resources like

capacity, platform or application are available when they are needed and they are not creating cost when they are not used. This allows the user to decide when to use different resources and also what resources are needed and how they are used. (Salo 2012: 17.)

Broad network access

Broad network access means that the services provided by certain vendor are available over the network and user can access them through standard mechanism. These can be a normal table computer or some kind of mobile device such as tablet. The services adapt to the equipment that are used and not in the other way around. (Gendron 2014: 34; Salo 2012: 17.)

Resource pooling

In resource pooling the user doesn't have to know in which way and where the services are implemented. Usually this information is not available for user. All the users using the same provider's service will use the same hardware and software capacity without being aware of each other and regardless of each other. Sharing the service and its capacity will intensify maintenance but at the same time brings up some challenges like isolating users from each other and delimit one user's damaging activity from others in a way that they can use the service without any changes. (Salo 2012: 17.)

Rapid elasticity

Cloud services scale easily and fast both upwards and downwards because of rapid elasticity. From customer's point of view, there is no limit in the capacity and the development and implementation of new applications will become faster. To increase storage, telecommunications and computing capacity for a need that is not planned is possible almost immediately as the need arise. (Salo 2012: 17.)

Measured service

Different pricing plans can be created and they can be for example pay as you go, subscriptions and fixed plans. The payment has to be based on the usage and not for example how much the different equipment cost. The usage can be measured by hours, data transfer or using other attributes where the attribute is something based on use. The measurability can mean storage capacity, the use of server's CPU or the main memory or different kind of performance data in telecommunications. The measurement data can be used to control, manage and develop the environment. (Gendron 2014: 33; Heino 2010: 48.) In companies this means that the control will be done against enterprise policies services. In companies the use of services is transparent. At the moment one can allocate cost per usage but not an average cost for all.

2.2.1 Advantages

Cost savings

Since the resources are acquired as a service from the service provider, it means that they can be bought on a pay-per-use basis. PaaS costs increase if the customer is using resources more and decreasing if usage is less. Also the size of the application will have an effect to the cost: increasing it when it is growing and vice versa. When buying cloud computing resources from the vendor, they are seen as operational expenditure which means that the company can take the cost straight from profit and it won't increase the company's taxes as it would if they would be considered as a normal hardware that the business is buying. (Williams 2010: 26–27.)

Own IT infrastructure will have lots of up-front costs when the company needs to buy software and hardware and maintain them. Also, if there is a need for example for more capacity since the company has grown, they need to take care of it themselves and do

changes in the whole IT infrastructure. However, with cloud computing and PaaS model the systems will be just the way the company needs them to be at each time. IT hardware and software licenses will also be utilized more in cloud computing. When using cloud services, the customer doesn't have to do the maintaining which means that the management costs will disappear. (Williams 2010: 26.)

Flexibility

Since the market is changing rapidly, companies need the ability to adapt to changing situations. Cloud services provide flexibility in relation to information technology. When the resources are bought as a service, flexibility is a great benefit that comes from it which allows very rapid adaptation to changing situations. If the company for example lays off people for certain period of time, the flexible resources are available as needed. Also when the business is growing both hardware and software resources don't impose significant restrictions on the growth. (Salo 2010: 80.)

Different PaaS solutions provide platform for software development and usually they have modules ready to use. Developers don't have to fight with limited resources or wait for them to be free to use. The development and testing can be executed in the platform quickly and smoothly. Because the development platform is operating over the network, this approach is usable for companies' development teams that are spread out all over the world geographically. One can develop new applications without having to worry about things like servers and firewalls. (Salo 2010: 80; Williams 2010: 28.)

Access anywhere

Workers should have an access to tools and data they need regardless of location. Cloud computing will enable that any portable terminal can perform computationally intensive

tasks by using group of servers. Because cloud services are web-based, the users only need to have Internet access. (Salo 2010: 81; Williams 2010: 28–29.)

Scalability and use-base billing

Upfront investments are not necessary when using cloud services because resources are available the amount needed without time delay or any other problems. Companies don't have to worry about the over capacity since it is easy to get rid of the capacity that is not needed. When the need for more capacity comes, it is as easy to get more of it as to get rid of. For example Amazon Web Services has created a service called Reserved Instance Marketplace. The idea of it is that the customer who has purchased too much capacity and haven't used it for 30 days can sell it to other customers. (Salo 2010: 81; Amazon 2014a.)

The costs of used services are based on the actual use and that's why the costs are usually changing. Also measuring systems, which enables this use-base billing produce lots of data of who has been using the resources, when they have been used and where and how the resources have been used. This information is produced for both service provider and also for the company who is the customer. This helps allocating costs and efficiency and productivity assessments. (Salo 2010: 81.)

Application can be scaled horizontally or vertically. Horizontal scaling is used when the application has clustered architecture. It has either a gateway or master node that is used for dividing the requests to different virtual machines. Extra nodes will be added to the cluster if the amount of requests and work increase. When they decrease the extra nodes are being removed. Most clouds support this. Vertical scaling means that extra resources are added to or removed from virtual machines. This is used in data centers for stabilization. (Dutta, Gera, Verma & Viswanathan 2012: 221.)

Ease of deployment

Self-service and a high degree of automation mean that when adapting cloud services the need for communication between persons has been minimized. This will bring savings in costs and use of time. In the perfect situation, the customer can enable or disable the use of resources easily and fast. (Salo 2010: 81.) This will be possible because of the standardized delivery model.

Quality

When the company itself is responsible for maintaining the available resources, it will give the sense of control, which will be lost when moving to cloud services. The fact is though that maintaining own resources comes with a price that is often lack of flexibility and slowness in regeneration, which can be a bad thing in a rapidly developing and changing market. Many companies still use old and insecure solutions and because of lack of knowledge and complex cross-couplings it can be very expensive and difficult to get rid of them. Cloud service providers are focused on delivering ICT-services and that is their core business and knowledge. For companies, whose focus is somewhere else, support systems might end up a little side tracked and sometimes they get even forgotten. Cloud service providers have an interest in maintaining and developing the quality of their services and keep them updated all the time. With PaaS model customer will get the latest software and updates automatically. (Salo 2010: 81–82; Williams 2010: 30.)

Document sharing and collaboration

Cloud services are changing the way people are acting and working. Cloud service model offers an option for accessing and using applications and documents with the independence of terminal and place. Cloud services provide a model, which enables access to applications

and information wherever and when. This will reduce for example the problems related to finding the correct information and the disappearance of information. (Salo 2010: 82–83.)

Support green agenda

As companies move their services to cloud, they have been noticed that their electricity consumption has been reduced not just a little but a significantly. This is because companies don't have to have their own data centers, which mean that they would also need to have for example cooling down services all the time. In cloud service the consumption and the cost of it will be shared with all the customers using certain vendor's services. (Mongan 2011: 60.) Since the environmental factors are quite important in today's world, it is good to see that using cloud services companies can really make an effort and show that this is a better way to handle things when thinking about the environment.

Cloud computing will also decrease the travelling costs that appear when company's employees have to commute from home to work when they need to use certain applications and data. When the cloud services are in use, employees can work for example from home and other business travelling will be reduced. Company doesn't have to pay either for employees to go to the data center to maintain and update the servers. (Williams 2010: 34–35.)

2.2.2 Challenges

Access control

Access control is one of the main challenges in cloud. It is really important to identify users accessing the cloud with different device. One scenario is how to identify the user if the device is lent or sold to some other person after the access has been granted. Users might

also use different systems and therefore they might need to use different access methods. (Amin, Bib Abu Bakar & Al-Hashimi 2013: 243.)

Security and privacy

For companies, data security is one of the most important things. Data and data storage are one area where security concerns arise. Companies want that the data is in a safe place and that service provider's own personnel or external parties don't have access to it. There is also usually a need to store the data because of regulatory and other reasons. For example in banking industry one needs to store certain information for certain period of time. Legislation may have an effect on how the data is stored and where. For example one doesn't want to store personal data outside Finland or EU countries or company doesn't want to store research and development documents unencrypted outside the organization. Examples like Wikileaks have shown that data protection of many traditional systems is not hundred percent reliable and in most cases people itself are the weakest links with weak passwords or morals. (Salo 2012: 37.) If someone is unsatisfied and knows what to do, that is all needed for them to do what they want unless there is a tracking system to track for example logins, which help one to react faster to this kind of cases.

If the cloud service provider's data center contains lots of data with good quality, it is more attractive target where vandals or corporate spies try to attack. Company who is concerned on the data security should assess how business-critical they are and what would happen if they would end up in wrong hands before transferring data to cloud. (Salo 2012: 38.)

Cloud service provider might store and keep customer's data in many different countries where for example electricity and personnel costs are lower. This might lead to a situation where there are different laws regarding data protection in the country where the data is generated compared to the country where the data is kept and stored. This can create data protection concerns for companies. (Dutta, Peng & Choudhary 2013: 44.)

In Europe the new EU data protection rules are expected to become binding in 2015 or latest in 2016. The old regulation is from 1995 and the world has changed after that a lot. People are nowadays buying things from Internet and using social media services. It is important to update the data protection rules across the whole EU. This will of course have a direct link to cloud computing where the data is stored in cloud. The fine for breaking these rules are said to increase up to 1 million or be 2% of company's revenue. (Travers Smith 2013.)

Performance concerns / availability

Availability and stability of the cloud service is one challenge. If one can't access the cloud, one doesn't have access to the information that is located there. For example Amazon's data center in the United States was down in 2011 almost for one day and services that were relying on it even more. Another example is also from Amazon. Amazon's data center in Ireland got struck by a lightning which led in a situation where some of the servers were not working. Even though interruptions are rare, it might be costly from company's point of view if the user would need the information right away. (Salo 2012: 39.)

It is good to think what the value of the data is and what the level of certainty that it must be available in is. If the data has to be accessible all the time, then cloud is not the right place to store it. It is useful maybe for storing the backup but not the primary location for storage. However one could use the idea of decentralization and store the same date in many different cloud providers' services. Or one could spread the information to cloud provider's different data centers. Amazon is one provider who provides it. (Salo 2012: 39.)

Researchers have found out that applications are running 20 % more effectively when the data they are using is located near. If there are many applications running on the same time on the same core, they will run more slowly compared to if the data is running on another

core. Distance will affect in latency and the longer the data has to travel the worse the latency will be. Cloud performance can also be slow because of congestion. This can be seen for example in the slowness of an online meeting. (Fogarty 2013; Rubens 2012; Plamondon 2011.)

There haven't been many problems with stability. There are examples of service breaks or losses of data for one day but almost every time the data has been restored. For example Amazon Web Services and Microsoft's Windows Azure store the data three times, so there are three copies of the data and they are stored decentralized in many places. (Salo 2012: 39.)

Vendor viability and maturity

Some companies are worried about the size and maturity of the cloud service vendor. Larger cloud service vendors are seen as better and safer options to protect customer's data as well their own data than little start-ups in cloud computing business. (Jaeger 2010: 56.) This refers to a fact that bigger service providers have already their own infrastructure and people working on it. Little start-ups are just beginning and they might not have the same offerings as larger service providers.

2.3 Standards

Cloud computing's standardization is still on going and there is not one certain group of standards that everyone follows. One can say that at the moment smaller companies are following the big cloud service providers like Amazon on this matter.

Like said, there are many different standards regarding cloud computing. For example IEEE has working groups for two different cloud computing standards: IEEE P2301 and

IEEE P2302. In IEEE P2301 the purpose is to have guide for cloud portability and interoperability. It will have different instructions for cloud attendants like provider and customer. The areas this standard should cover are for example application, portability and management interface. IEEE P2302 creates guide for Intercloud Interoperability and Federation. This includes for example topology and functions. (IEEE 2014a; IEEE 2014b.)

In the table 2 below the most important standardization areas have been gathered and their status is presented. The elements have been collected from ETSI's report. ETSI stands for European Telecommunications Standards Institute.

Table 2. Cloud computing standards and their status. (ETSI 2013: 20–26.)

Standard / Summary of the topic	**Status of the standard**	**Remarks**
Terminology and metrics	Draft version is available but the development of this standard is still on going and needed.	
Requirements specification	Nothing exists at the moment and there is a need for this standard.	There are other standards that can help for example in comparing different service providers.
Security and privacy requirements	Standard needed.	There are many different non-cloud security standards. For privacy there are for example different legislation in different countries.
Service assessment	Is available.	
Standards expression of SLA	In process.	
Enabling interoperability	Is available.	
Enabling data portability	Is available.	
Integration of cloud solution with legacy systems	Nothing exists at the moment and there is a need for this standard.	
continues (on next page)…		

…continues (on previous page)		
Administration of users, identities and authorizations.	Need to be developed further.	Non-cloud standard exists.
Independent monitoring of SLA	Is available.	
Reporting SLA infringements	Is available. Should be developed further.	Trust and automatisation can be seen as challenges.
Responding to SLA infringements	Nothing exists at the moment and there is a need for this standard.	
Creation of a VM image for the public cloud	Is available.	
Monitoring: Availability	Is available.	
Monitoring: Incident management	Nothing exists at the moment and there is a need for this standard.	
Monitoring: Storage performance	Nothing exists at the moment and there is a need for this standard.	
Monitoring: Processing performance	Nothing exists at the moment and there is a need for this standard.	
Monitoring: Networking performance	Nothing exists at the moment and there is a need for this standard.	
Monitoring: Access security event information	Nothing exists at the moment and there is a need for this standard.	
Monitoring: uptime	Nothing exists at the moment and there is a need for this standard.	
Contract termination	Is available.	
Providing an evaluation report	Nothing exists at the moment and there is a need for this standard.	This could be used to compare service providers.

As one can easily see from table 2, that there are lots of standards that are still missing but the need for them has been noticed. There are also many standards available from for example security but the problem is that they are not made for cloud environments. This is why they can't be used as they are but instead they should be changed to fit cloud environment first.

2.4 Architecture of cloud computing

The complexity of the concept of cloud computing is one reason why it might be difficult to understand cloud computing. Market and technical innovations are combined in cloud computing which makes it difficult to understand the whole picture. NIST published its presentation about cloud computing architecture, which aims to describe the cloud service market as a whole, and it will not to take stand on implementation technologies or providers. (Salo 2012: 29.)

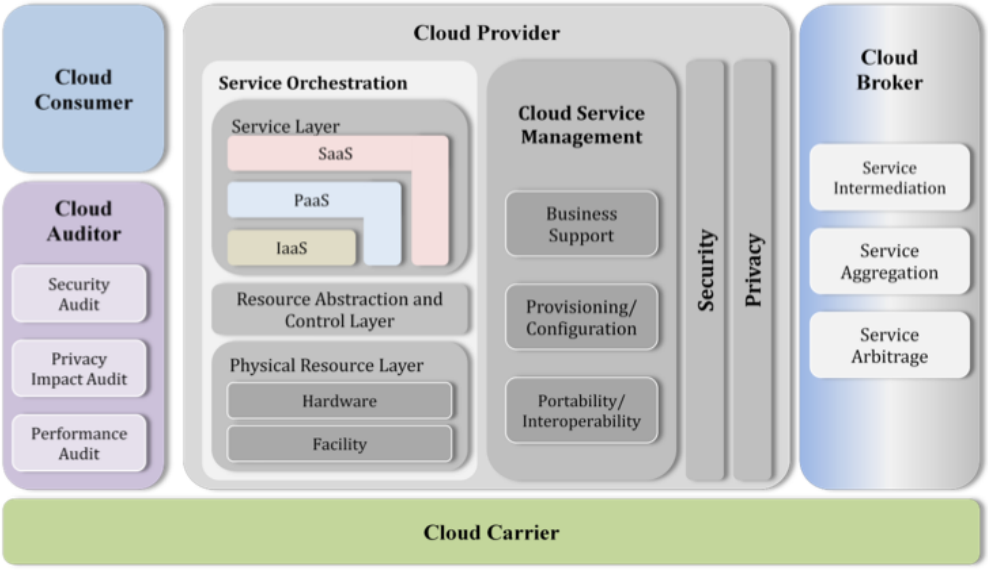

Figure 5. The Conceptual Reference Model (Liu et al. 2011: 3).

In figure 5 we can see an overview of the NIST cloud computing reference architecture. The reference architecture identifies the following things: major actors and their activities and functions. The different actors are cloud carrier, cloud provider, cloud consumer, cloud broker and cloud auditor. Each of them is an entity that can be for example an organization. All of these actors are part of transaction or process that happens and/or executes a certain jobs in cloud computing. The overview helps to understand for example cloud computing's characteristics and standards. (Liu et al. 2011: 3.)

Table 3. Actors in Cloud Computing (Liu et al. 2011: 4).

Actor	**Definition**
Cloud Consumer	A person or organization that maintains a business relationship with and uses service from *Cloud Providers*.
Cloud Provider	A person, organization, or entity responsible for making a service available to interested parties.
Cloud Auditor	A party that can conduct independent assessment of cloud services, information system operations, performance and security of the cloud implementation.
Cloud Broker	An entity that manages the use, performance and delivery of cloud services, and negotiates relationships between *Cloud Providers* and *Cloud Consumers*.
Cloud Carrier	An intermediary that provides connectivity and transport of cloud services from *Cloud Providers* to *Cloud Consumers*.

In table 3 the different actors in cloud computing are presented. Cloud Consumers are a group of different users: private persons and organizations. Organization can be a small single person company or a huge company with thousands of employees. One can easily

see that cloud services have a broad customer base. Many potential users of cloud services have already used SaaS services even though they might not have had any idea of it. (Salo 2012: 30–31.)

Cloud Providers are the ones who actually provide the service as IaaS, PaaS or SaaS service. They can directly sell the service to end customer or do it through intermediaries. These can be cloud brokers or carriers. Cloud Auditors are market players who try to give answers to problems such as information security, legal questions, risk management and service reliability. Cloud auditors also ensure the quality of the service. Usually the normal customer doesn't have a chance to visit the cloud provider's premises or get more information on how the provider has implemented the services. The cloud auditor can look into these matters and give information to customers. (Salo 2012: 30–31.)

In addition to the service it already offers Cloud Broker offers value-added services, which decrease implementation and usage costs, speed up processes, decrease or manage the risk, give valuable information to the customer or other services that benefits the customer. Cloud Carriers are the ones who take care of the customer interface. These carriers are specialized in customer relationships, brand expertise, sales, marketing and customer service. With their skills they are able to find customers for cloud service providers. (Salo 2012: 31.)

2.5 Cloudstep

In many companies, there have been problems when they have been trying to migrate the already existing legacy applications to different public clouds.

> "*Cloud migration decisions are inherently complex since they are influenced by multiple, possibly conflicting factors, such as cost, performance, security and legal concerns* (Beserra, Camarra, Ximenes, Albuquerque & Mendonça 2012: 7)."

Also the technical limitations that might stop the whole migration have to be taken into consideration. These limitations can be for example that the moving applications might do some damage to the environmental constraints determined by the vendor. Step by step process of typical cloud migration has been developed because of these problems companies have been facing. The name for it is Cloudstep and it supports companies with decisions regarding clouds and application migration. (Beserra et al. 2012: 7.)

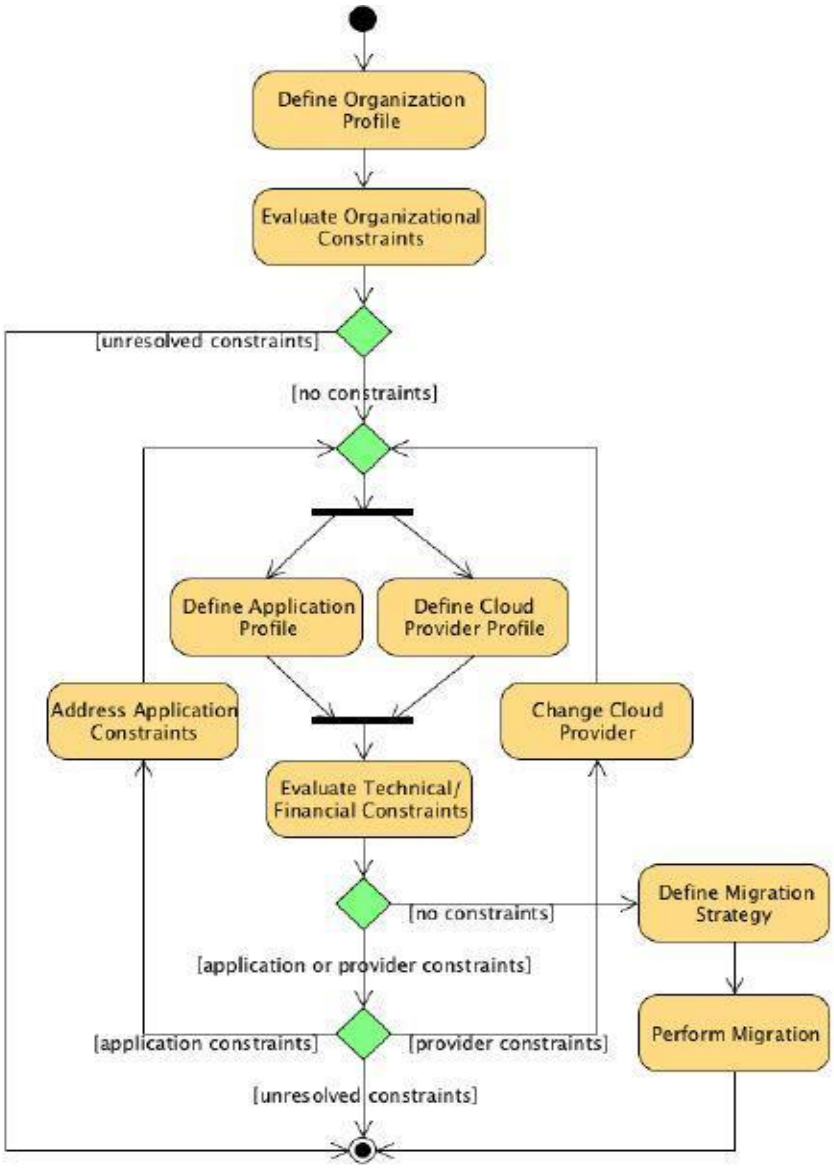

Figure 6. The Cloudstep workflow (Beserra et al. 2012: 8).

Cloudstep's workflow is presented in figure 6. In the figure you can see that the whole workflow has nine different activities. Next these activities are described.

Define Organization Profile

The first activity in the workflow is called "Define Organization Profile". The goal is to create a profile of the company where the information is created regarding legal and administrative features that are seen relevant when thinking about the migration. These characteristics are for example policies and guidelines or other practices in the company. The idea of creating the profile is to get an image if there are some restrictions that might have an effect on the cloud migration and the decision to do it or not. (Beserra et al. 2012: 8.)

Evaluate Organizational Constraints

In the second activity one

> "*should conduct preliminary assessment of potential organizational constraints regarding the adoption of a particular cloud model, based on information available in the organization profile* (Beserra et al. 2012: 9)".

In this way the possible critical aspects of the organization are noticed. This is also an important input for the decision-making process along the technical and financial factors. (Beserra et al. 2012: 9.)

Define Application Profile

In the third activity the profile for the application that one wants to migrate is created. In the profile all important characteristics that might have influence when doing the migration to cloud should be mentioned. This activity has two sub-activities: identifying usage

characteristics and identifying technical characteristics. In the first one the characteristics of the application's use and operation are identified. In the latter one the characteristics of technologies used are identified and also any other essential technical information regarding applications use. (Beserra et al. 2012: 9.)

Define Cloud Provider Profile

In the fourth activity a profile for different cloud vendors is created. This way one can compare different vendors and see if they fulfill the required limitations identified for the application and the company. (Beserra et al. 2012: 10.)

Evaluate Technical and/or Financial Constraints

> "*The goal of this activity is to assess the conformance between the organization profile, the application profile and the profile of the candidate cloud provider* (Beserra et al. 2012: 10)."

Seven main types of constraints have been recognized. They are financial, organizational, security, communication, performance, availability and suitability constraints. All of them should be assessed within the same context. After the evaluation, there are three different actions that can be chosen. First one is to do something for the constraints that are breaking the application. Second one is to select another service provider and the third and last action is to stop the whole process of application migration. (Beserra et al. 2012: 10–11.)

Address Application Constraints

In this activity application constraints should be resolved. This can be done in different ways. One option is to change the whole application that would mean recoding some parts of it. Other option is to decrease or increase the migration scope. This could mean for example that only certain application components are moved or that one has to move other

components and systems in the cloud in order for the application to run. (Beserra et al. 2012: 11.)

Change Cloud Provider

In this activity, the constraints evaluated in the last two activities are compared to different cloud providers in order to see who can solve the problems with their service. Different selection criteria can be used and companies can decide what the most important criteria are for them. An example can be that constraints relating to a physical location of the data can be solved with provider whose data center is situated in a certain area that is accepted by the company. If a new provider is selected one has to create a new profile for it and start the constraints evaluation cycle from the beginning. (Beserra et al. 2012: 11.)

Define Migration Strategy

This activity can be carried out when the cloud provider is selected and there are no more constraints that would prevent the migration. The migration strategy should be designed with using the information and issues that has been raised during the evaluation process. It is suggested to consider many different migration scenarios since the cloud environment is rapidly changing. (Beserra etl al. 2012: 11–12.)

Perform Migration

The ninth and last activity is the actual migration that the company will do after they have evaluated all the different factors before. This activity includes all the actions required to perform the migration. This means for example moving application data and its components to cloud provider's service and testing that the application is running smoothly in the cloud and there are no problems in it. (Beserra 2012: 12.)

3. MARKET ANALYSIS

3.1 Main platform clouds

Cloud computing market has been growing and there are plenty of different vendors to choose from. In this work the following vendors and their services will be examined in more detailed way: Amazon, Microsoft Azure and Google. Oracle would be a good example as well but they are still at very early stage with their solutions.

3.1.1 Amazon Web Services

Amazon has been utilizing decentralized IT infrastructure already for a long time. It launched Amazon Web Services, also known as AWS, that offers IT infrastructure services in the form of cloud computing. AWS was launched in 2006. Earlier they were called services in the form of web services. It provides infrastructure platform in the cloud for many thousands customers in 190 countries all over the world. Its data centers are located in the USA, Europe, Brazil, Singapore, Japan and Australia. (Varia & Mathew 2014: 4; Amazon Web Services 2014b.)

AWS has different infrastructure services that they are providing. It offers power, storage, content delivery and other functionalities (Varia et al. 2014: 10). These services can be seen in figure 7. From these, the following services are described better: Amazon Elastic Compute Cloud (Amazon EC2), Amazon S3 and Amazon Relational Database Service.

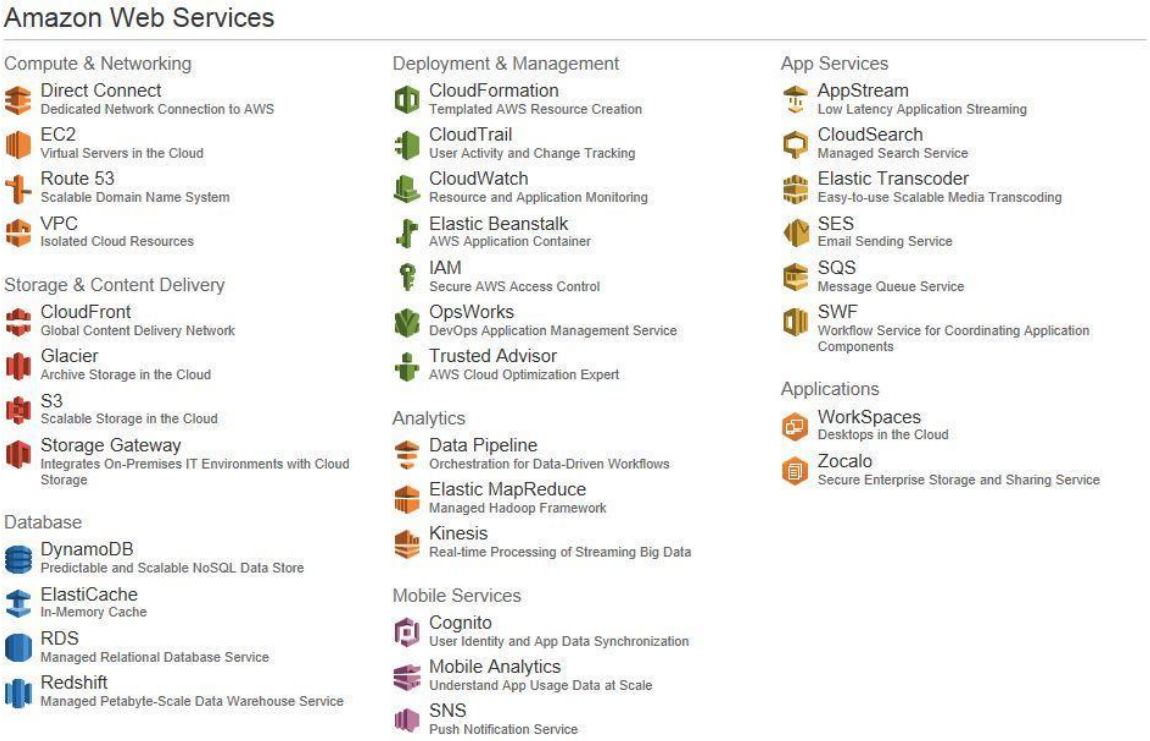

Figure 7. Amazon Web Services (Amazon 2014c).

Amazon Elastic Compute Cloud, also known as Amazon EC2, is offering resizable computing capacity in the cloud. It is a web service that allows the user to get and configure capacity with low number of conflicts. When customers computing requirements changes it is easy to increase or decrease the capacity with Amazon EC2. Amazon EC2 also offers tools to build applications that can handle failures and avoid common failures. (Amazon Web Services 2014d.)

Amazon EC2 has many different features. *Amazon Elastic Block Store*, EBS, provides persistent storage for the instances that are used by Amazon EC2. The volumes of EBS are network-attached and last as long as the instance is up. It is possible that some of the Amazon EC2 instances can be launched as *EBS-Optimized Instances*. They enable those "*EC2 instances use the IOPS provisioned on an EBS volume* (Amazon Web Services

2014c)". Instances can be placed in *multiple locations*. The locations can be Regions or Availability Zones. Availability Zones are separate locations that have been designed in a way that they won't get distracted from failures in other Availability Zones. All Availability Zones in the same Region are connected. When instances are placed in different Availability Zone, one can protect the application from failures in certain location. Regions contain one or more Availability Zones and all of the Regions are places differently geographically. (Amazon Web Services 2014e.)

*Elastic IP addresse*s are one of the features. They are static IP addresses. This means that the address is associated with user's account and the user can control it until decides to let it go. (Amazon Web Services 2014e.)

> "*Unlike traditional static IP addresses, however, Elastic IP addresses allow you to mask instance or Availability Zone failures by programmatically remapping your public IP addresses to any instance in your account* (Amazon Web Services 2014c)."

Amazon CloudWatch is a service that offers a possibility to follow resources and applications in AWS. With this service, one can have information of utilization of resources, operational performance and all different metric data that can be followed. Examples from this are disk readings and writings and the CPU usage. The information can be in different forms; basic statistics or graphs and user can set alarms for monitoring certain metric data. (Amazon Web Services 2014e.)

Auto Scaling allows automatically scaling capacity up and down depending of the need. With Auto Scaling it is possible to make sure that the right instances are scaling up when the demand is rising to maintain the performance and vice versa. This is well-suited service especially for applications that face variety in usage for example hourly or daily. *Elastic Load Balancing* is dividing the incoming applications to different Amazon EC2 instances. This can notice the unhealthy instances and stop sending applications to it. The applications

will be sent to working instances until the unhealthy one has been restored. (Amazon Web Services 2014e.)

There are still some features, but they are not going to be explained in this work. Below is the list of the rest of the features that Amazon EC2 has.

- “ *Amazon Virtual Private Cloud*
- *High Performance Computing (HPC) Clusters*
- *GPU Instances*
- *High I/O Instances*
- *High Storage Instances*
- *VM Import/Export*
- *AWS Marketplace*
- *Enhanced Networking (Amazon Web Services 2014e).*”

Amazon S3 is a data storage service. It offers a web-services interface. The interface can be used to store and seek data whenever and from anywhere on the web. User can store different kind of data in Amazon S3, from web applications to media files. (Amazon Web Services 2014f.)

Amazon Relational Database Service, also known as Amazon RDS, is a web service that offers an easy way to create, run and scale database. It is cost-efficient and resizable when it comes to capacity. “*Amazon RDS gives you access to the capabilities of a familiar MySQL, Oracle, SQL Server or PostgreSQL database* (Varia et al. 2014: 14).” So when one is using Amazon RDS, the tools and applications that are already used in today’s databases can still be used. Amazon RDS also repairs any failures in the database software and does automatically back-ups. The backups are stored for a period of time which the

customer can decide and it is possible to do recovery in certain point-in-time. (Varia et al. 2014:14.)

AWS Management Console is a web-based user interface that is used to manage AWS. With the console user can work for example with Amazon S3 buckets and launch and connect to Amazon EC2 instances. The console includes all the different services, which makes it easy to manage all of them from one place. User can also find information about billing from there. (AWS Documentation 2014.)

AWS Management Portal for vCenter is another tool that customer can use in order to manage AWS resources by using VMware vCenter. It is developed especially for enterprise customers and to manage EC2 instances. It also has support from VMware. It has single sign-on support, which will allow the customer to use some identity provider to manage users. It has Role-Based Access Controls which means that one can give permissions to certain resources that user can edit, view or create. The administrators can also define templates where the instance types and other resources are defined and how the end user can use them for example when creating EC2 instances. (Amazon Web Services 21014g.)

3.1.2 Windows Azure

Windows azure is Microsoft’s cloud platform that was announced in 2008 and became commercially available in 2010. Windows Azure platform consist of three different components: Windows Azure, SQL Azure and Windows Azure AppFabric. Windows Azure offers computing and storing services, SQL Azure offers relational database that is cloud-based and Windows Azure AppFabric offers cloud-based infrastructure services. (Anderson 2012: 16; Chappell 2010: 3.)

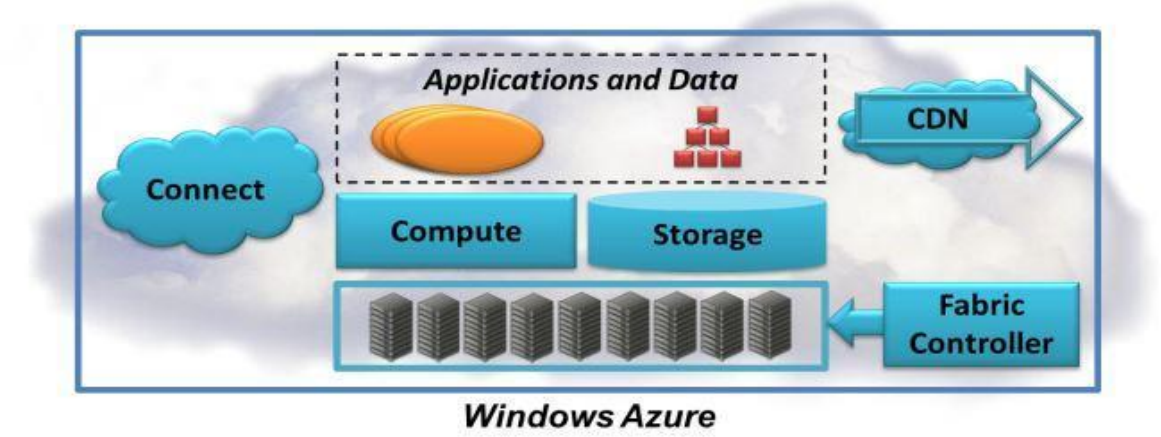

Figure 8. Windows Azure has five main parts: Compute, Storage, the Fabric Controller, the CDN, and Connect (Chappell 2010: 3).

As seen in figure 8 Windows Azure has five main parts. Compute is the place where the applications are run. Storage will store binary and structured data. Fabric Controller is a part that places, manages and tracks the applications. It is also responsible for updating the system software in the platform. CDN means Content Delivery Network and it manages the global access to data in the storage by maintaining cached copies of it all over the places. The last one, connect, will generate IP-level connections between physical computers and Windows Azure applications. (Chappell 2010: 3.)

Azure Cloud Services is one of three compute options Windows Azure offers. The cloud Services is a service based on PaaS model. This service supports applications that are scalable, reliable and cheap to run. With this service, developers don't have to take care of maintaining the platform and they can concentrate on their applications. (Microsoft Azure 2014a.) This is presented in figure 9.

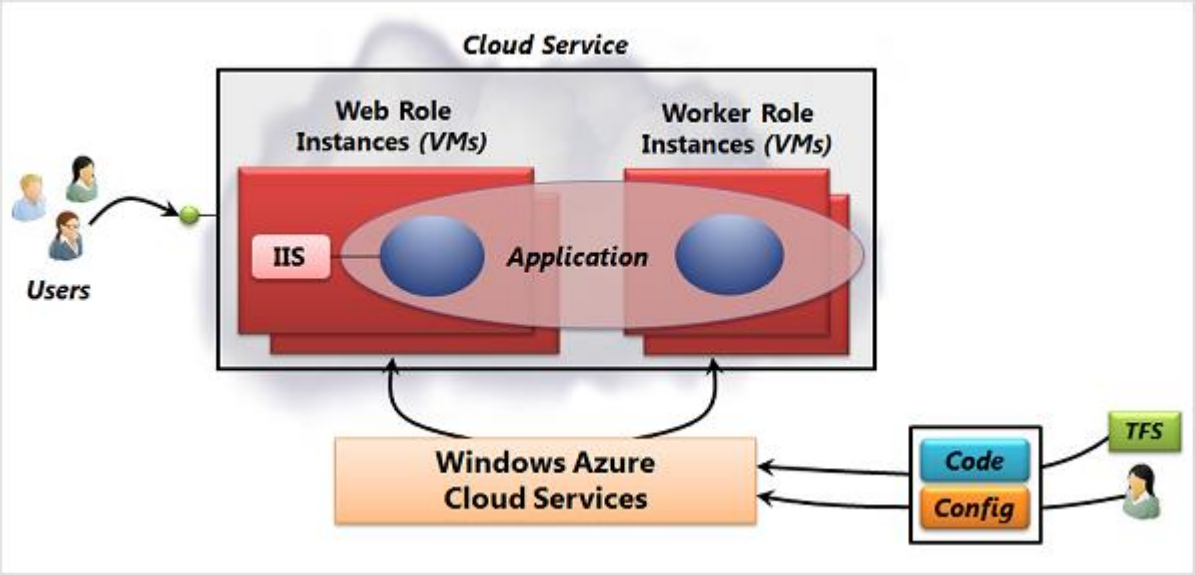

Figure 9. Azure Cloud Services provides Platform as a Service (Microsoft Azure 2014a).

Azure is relying on virtual machines, VMs. As seen in figure 9 all the virtual machines in a single application operates in the same cloud service. (Microsoft Azure 2014a.)

> *"Because of this, users access the application through a single public IP address, with requests automatically load balanced across the application's VMs. And as with cloud services created using Azure Virtual Machines, the platform will deploy the VMs in a Cloud Services application in a way that avoids a single point of hardware failure* (Microsoft Azure 2014a)."

When using Cloud Services, one doesn't have to create VMs. There are two different VM options: Web role instances and Worker role instances. Cloud Services are relying on the combination of them both. Web role is for web application programming and worker role is useful for generalized development. Worker role can run background processing for the web role. As an example, a single application could use web role when a complex application could use web roles to handle all the requests coming from users and then transfer those to a worker role, which would then process them. (Microsoft Azure 2014a; 2014b & 2014c.)

The needed amount of web roles and worker roles has to be informed to Cloud Services. If the application in question requires a greater load, then one can inquire for more VMs and then Azure will generate those instances. This works also the other way around. If the needed load decreases then one can shut those instances down itself. (Microsoft Azure 2014a.)

Cloud Service has monitoring as a part of it service. It can for example detect if a server has failed and start again the VMs that were running in that server on a new machine. Hardware failures are not the only failures Cloud Services can detect; alongside the VM problems it can detect problems with applications. Cloud Service can start again them because it has an agent inside every web and worker role. (Microsoft Azure 2014a.)

When using Cloud Services, one has to notice that the applications used in this service should run correctly even if any one of the web or worker roles stops working. In order for this to work, the information of the state of an application shouldn't be kept in the own virtual machine. Instead of that all the states should be stored to SQL database, tables or into some other external place. If the applications are built this way it will make it easier to scale them and they will be durable against to failure. (Microsoft Azure 2014a.)

Windows Azure application looks like building any other Windows application. The platform supports .NET applications and also applications that are created by using unmanaged code. One huge difference between cloud and physical computers is that the applications in cloud won't run locally and this could make the development difficult. Microsoft offers development fabric as help, which is a certain version of the environment where Windows Azure is running that can be used in other machines. It operates on a single desktop or server machine. (Chappell 2010: 17.)

Azure Websites

Azure Websites is a PaaS solution that is built on the Azure Cloud Services and developed to run web applications. It is running on a single VM, which can contain many websites that are created by different users. This can be seen in figure 10. VM's are resources that are managed by Azure Websites and because of this they are reliable and they have high fault tolerance. Applications and its copies can be running in different VM's at the same time and Azure Websites will balance the requests between all the VM's. Azure Websites supports development tools like WebMatrix and Visual Studio. Applications can be created by using for example languages ASP.NET, PHP and Python. For example publishing code Git, GitHub and Dropbox are also supported by Websites. (Microsoft Azure 2014a.)

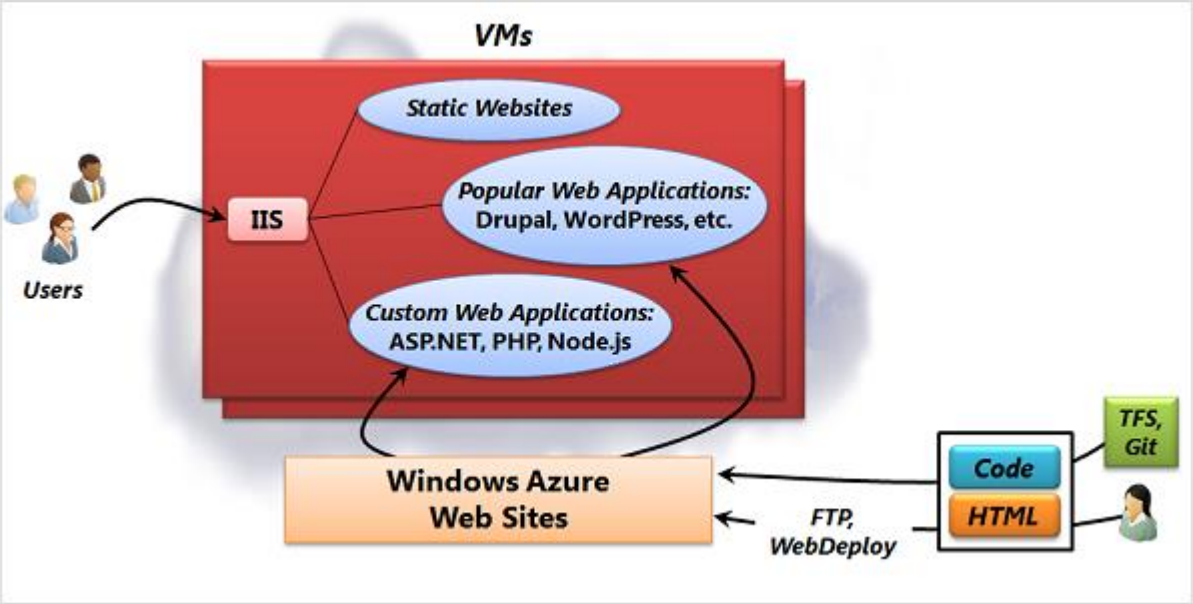

Figure 10. Azure Websites. (Microsoft Azure 2014a.)

SQL Azure

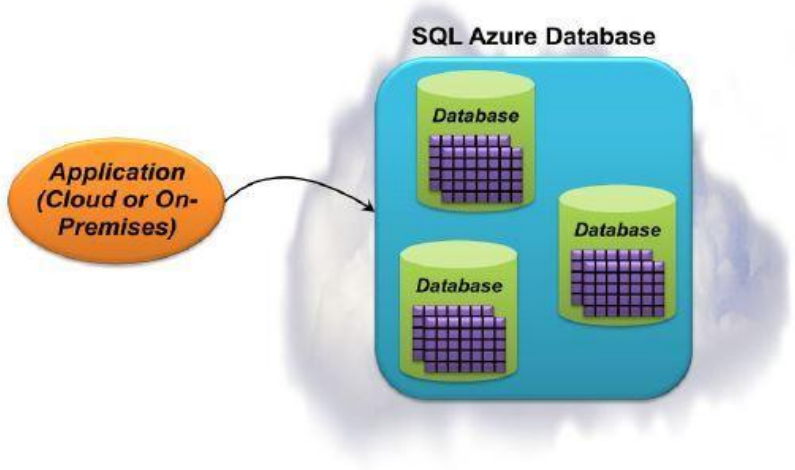

Figure 11. SQL Azure Database provides cloud-based relational storage for applications running on-premises or in the cloud (Chappell 2010: 6).

In figure 11 the basic idea of SQL Azure is presented. It consists of databases and application. The application that is using SQL Azure Database can be running in the cloud or on premises. No matter where the application is running, it accesses the cloud service like it would access SQL server. Basically it offers a SQL Server environment with for example complete indexes and views. If the application is now accessing the server locally it will work in the same way when the data is in the Azure's database. Though there are some differences when talking about the cloud and on premises databases. As everything is shared in the cloud, a query can be running for a certain time since a single request can't use more resources than what has been defined before. There is also a restriction that says that limit for single database is 50 GB. If the application has more data, it needs to create multiple databases and divide the data between them. (Chappell 2010: 6.)

Windows Azure AppFabric is used for infrastructure services. It can be divided into two components: Service Bus and Access Control. Access Control offers identity services. It also controls the access to Service Bus. (Chappell 2010: 6–7.)

> "*Service Bus provides a secure way for an on-premises application to expose services built using Windows Communication Foundation (WCF) to clients outside the firewall. Each exposed service establishes a connection with Service Bus, through which clients can access the service* (Chappell 2010: 6)."

3.1.3 Google App Engine

App Engine is Google's service that is based on PaaS. It supports applications that have been written in Java, Python, PHP and Go. The applications are running in a safe environment where the requests are sent to many different servers and there are enough scaling servers to take care of the changing need in capacity. App Engine supports common development tools like Eclipse and Git. Customer can choose from many different storage options that are for example MySQL database by using Cloud SQL or object storage in Cloud Storage. (Google Developers 2014a; Google 2014.)

Google App Engine includes the following features:

- "*Persistent storage with queries, sorting, and transactions.*
- *Automatic scaling and load balancing.*
- *Asynchronous task queues for performing work outside the scope of a request.*
- *Scheduled tasks for triggering events at specified times or regular intervals.*
- *Integration with other Google cloud services and APIs* (Google Developers 2014a)."

AppEngine's instances can be managed by using Admin Console. Instances are the basic building blocks that are for example language runtime, application's code and memory. Each instance has security layer which assures that different instances can't affect each other for example by damaging one another. There are two types of instances. Dynamic instance will start up and shut down when needed. Resident instance is running all the time and it can lead to a better performance. When the application is configured, one needs to decide also how the modules scale and how much time one instance can use when taking care of the requests. (Google Developers 2014b.)

There is a request queue for each instance and App Engine monitors this. If it notices that the waiting time for certain instance is starting to get too long, it will automatically create a new instance of the application, which then takes care of part of the requests. Instances can be scaled down also when the volume decreases. If some application is not used at all, App Engine will turn it off but as soon as the application is needed again, it will reload it. Latency has a huge impact on how many requests can be handled and in other hand how many instances are needed. Single-threaded instances, for example Java, can handle one request at one time. Multi-threaded instances can handle many at the same time. (Google Developers 2014b.)

3.1.4 Summary of different vendors

In table 4 a summary of different vendors can be seen. It also has some information that was not yet presented in the previous chapters. As one can notice all of the three main vendors have similar features. For example language framework is the one clear difference between all of them. Security standard SAS 70 was replaces in 2011 with SSAE 16. SSAE states for Statement on Standards for Attestation Engagements. It is a guide for executing service auditor's examination. (SAS 70 2014.)

Table 4. Key features of cloud provider offerings (Louridas 2010: 10).

Feature	Amazon Web Services	Google App Engine	Microsoft Azure
Computing architecture	Elastic Compute Cloud (EC2) supports virtual-machine image uploads and gives client APIs to instantiate and manage them	Google's distributed architecture	Hosted in Microsoft data centers, providing an OS and developer services that can be used individually or together
Virtualization management	OS-level running on Xen hypervisor	Applications run on instances of Java VM or Python runtime	Hypervisor (based on Hyper-V)
Service	IaaS	PaaS	PaaS
Load balancing	Elastic load balancing	Automatic scaling and load balancing	Built-in load balancing
Fault tolerance	Placing EC2 instances in multiple "availability zones"	Automatically pushed to a number of fault-tolerant servers, App Engine's Cron service	Application instances are replicated in independent fault domains; all data is replicated three times
Interoperability	Published API for interacting with instances; hosted applications can be written in standard programming languages running in the virtual machines (VMs) and accessed through the interfaces they provide	Hosted applications written in Java and Python and accessed by the interfaces they provide; access to Internet resources from code and App Engine's Fetch service	Hosted applications run in MS-Windows and accessed by the interfaces they provide; applications can also use a simple API to access the Azure fabric
Storage	Simple Storage Service (S3), Amazon Elastic Block Storage (EBS), Amazon SimpleDB, Amazon Relational Database Service (RDS)	App Engine datastore (not relational, built on Bigtable); objects with properties are stored without a schema, supporting transactions	Persistent data stored in non-relational blobs, tables, and queues; SQL storage offered by SQL Azure
Security	SAS 70 Type II Certification, firewall, X.509 certificate, SSL-protected API, access control list	SAS 70 Type II Certification, secure access to intranet via Google's Secure Data Connector	SAS 70 Type II Certification, applications run on 64-bit MS-Windows Server 2008
Programming framework	Amazon Machine Image (AMI), MapReduce	Java and Python, scheduled tasks and queues, access to services such as URL fetch, mail, Memcache, image manipulation	.NET and unmanaged code as long as it runs on MS-Windows

3.2 Other vendors in the market

3.2.1 Rackspace

Rackspace was established in 1998. It offers different kind of cloud services, from private cloud to hybrid cloud. Rackspace is the founder of OpenStack, the cloud operating system. It has nine datacenters in four different continents. Rackspace Cloud Sites is Rackspace's PaaS solution. (Rackspace 2014a.)

Cloud Sites is running on both Windows and Linux servers. Windows based pages are running with clusters that are built especially for Windows. The same is with Linux pages. Cloud Sites is using Load-Balancing technologies which tell automatically what technology is used and based on that knows to which servers it belongs to. This way customer doesn't' have to make decision between Windows and Linux and can use them both. (Rackspace 2014b.)

3.2.2 OpenStack

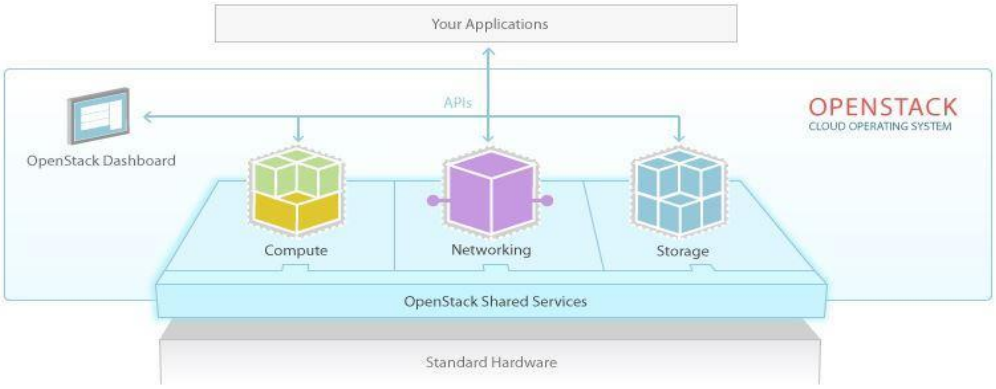

Figure 12. OpenStack: The Open Source Cloud Operating System (OpenStack 2014a).

OpenStack is a cloud operating system founded in 2010 that includes three different parts; compute, storage and network. (OpenStack 2014a; Rackspace 2014a.) This can be seen in figure 12. OpenStack offers an opportunity for providers to offer on-demand computing resources. This is possible since OpenStack is having and maintaining lots of networks of virtual machines. These resources can be accessed via APIs for the developers and via web interfaces for the end users and webmasters. These resources can be scaled horizontally. (OpenStack 2014b.)

When it comes to storage, OpenStack has got support for Object Storage and Block Storage. Object Storage offers distributed storage platform that is accessible via API. It can be combined straight into applications and it can also be used for backing up and archiving the data. Block Storage can reveal block devices and connect them to compute instances in order to for example increase the storing size and achieve greater performance. (OpenStack 2014c.)

OpenStack Networking is a system that manages networks and IP addresses. This can be used to increase the datacenter resources just like in the other parts of this cloud system. For example, this will make sure that the network won't end up being a bottleneck in the development of a cloud. OpenStack Dashboard is a tool, which is used to manage cloud-based resources. It is only one way to manage the resources and other like native OpenStack API or EC2 compatibility API can be used for it too. (OpenStack 2014d; OpenStack 2014e.)

3.2.3 VMware

VMware has the industry's first open PaaS Solution called Cloud Foundry. It is an open source project and it has huge community behind developing the service. Cloud Foundry has support for example for Spring for Java, Sinatra for Ruby and Node.js. Since the architecture is open, other programming languages will be supported as well. After some

time it will also support MySQL. Cloud Foundry is not tied to a certain cloud environment and it can use other than VMware's infrastructure for example environments that are offered by VMware vCloud partners. Even support for AWS has been demonstrated. (VMware 2011; Cloud Foundry 2014.)

3.3 Examples

3.3.1 Zynga

Zynga is known for different games that user can play with their phones or via web browser. Zynga uses Amazon Web Services' public cloud and as a private cloud they have their own zCloud. zCloud works in the same way as AWS but it offers better performance, scale and reliability. Public cloud is used when they launch a new game and need to have flexibility and greater storage. When it is possible to predict for example how many people will play the game per hour and how much storage it would need, the workload would be transferred to private data centers and the game would run in the private cloud. (Zynga Engineering 2012.)

3.3.2 Netflix

Netflix started as a DVD Shipping service in USA in 1998. In 2007 it changed to streaming video first in USA and after that in other countries too including Canada, Latin America and Nordics. (Cockcroft 2013.) How Netflix works is presented in figure 13. It is using AWS Cloud Services and it has everything in the cloud concerning the service itself as one can see in the figure 13. The end-user has its own device that can be TV or smartphone or something else. CDN stands for content delivery network and OpenConnect is Netflix's own solution and it is responsible of delivering and accessing the data.

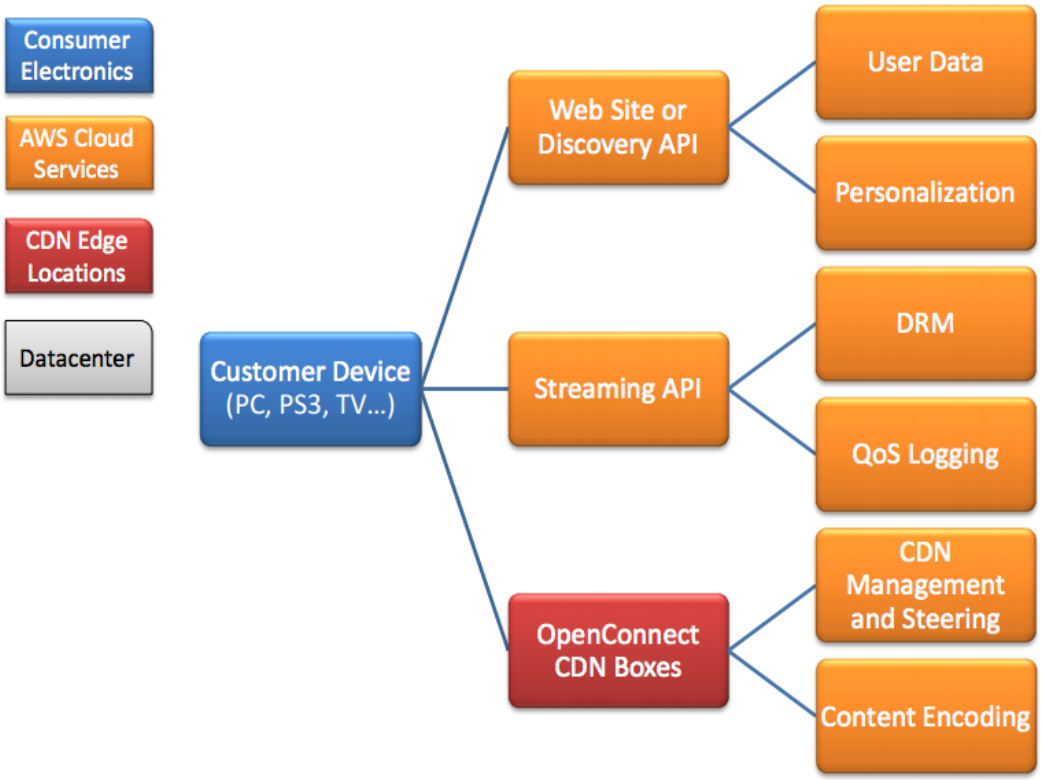

Figure 13. How Netflix streaming works today (Cockcroft: 2013).

3.4 Reference architectures

In this chapter the Amazon Web Services' reference architectures will be used as an example. Amazon offers different reference architectures from where it is easy to start building up own architecture. Amazon offers reference architectures for example for web application hosting, content and media serving, online games and media sharing. (Amazon Web Services 2014h.) Next web application hosting and media sharing will be presented. Web application hosting reference architecture is a good base when creating application that is accessible through web browser. Media sharing reference architecture could be a starting point when creating video streaming services similar to Netflix.

Web Application Hosting

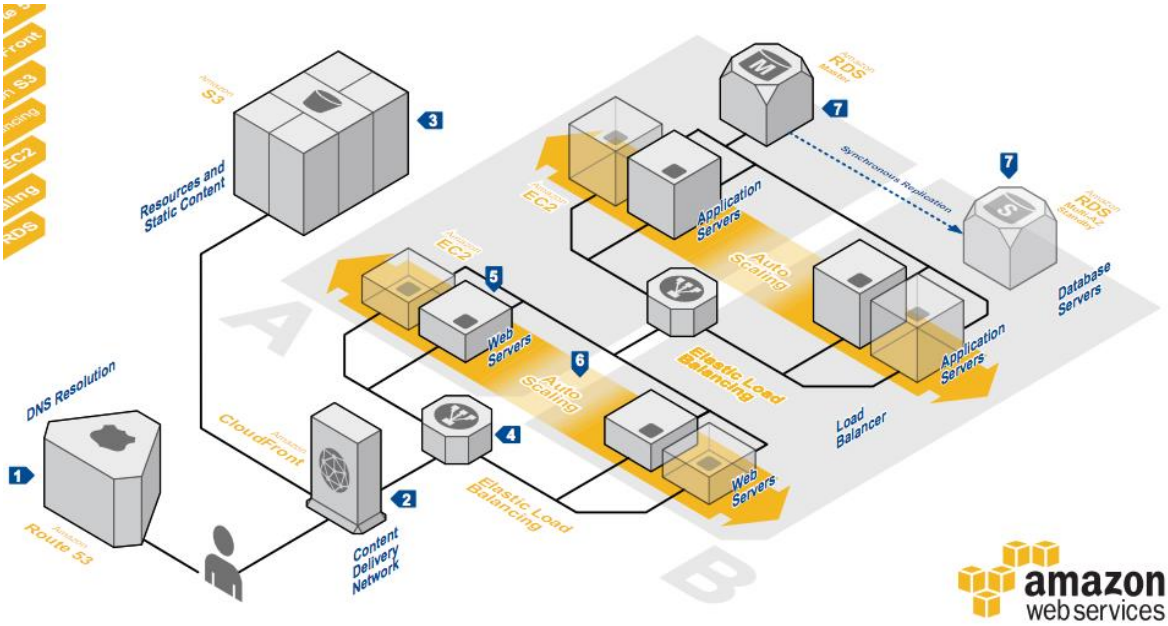

Figure 14. Web Application Hosting. (Amazon Web Services 2014i.)

Web application's architecture in the cloud is presented in figure 14. It has seven different elements. The first element is Domain Name System Resolution where user's requests are handled by Amazon Route S3. The second element is Content Delivery Network where CloudFront is used. All the content is delivered by using it and it will send the requests to the closest location securing the good performance. Element number three is the storage of resources and static content that the application uses by using Amazon S3. Element four is responsible by handling requests and distributing them to Amazon EC2 in different availability zones. Element five is web servers that are placed on Amazon EC2 instances. Element six is auto scaling that will automatically scale capacity up and down according the defined rules. Element seven is Amazon RDS, which includes the databases where the

data is saved. It is good to notice that the elements five, six and seven are all in both availability zones. (Amazon Web Services 2014i.)

It is good to notice that the elements five, six and seven are all in both availability zones since they are critical for the applications to be able to run. As these instances are in different zones, it is likely that the application won't get disturbed even if some of the instances in other zone will stop working. The application can just continue running in other zone. In figure 15 example of a classic web application is presented in a more detailed way.

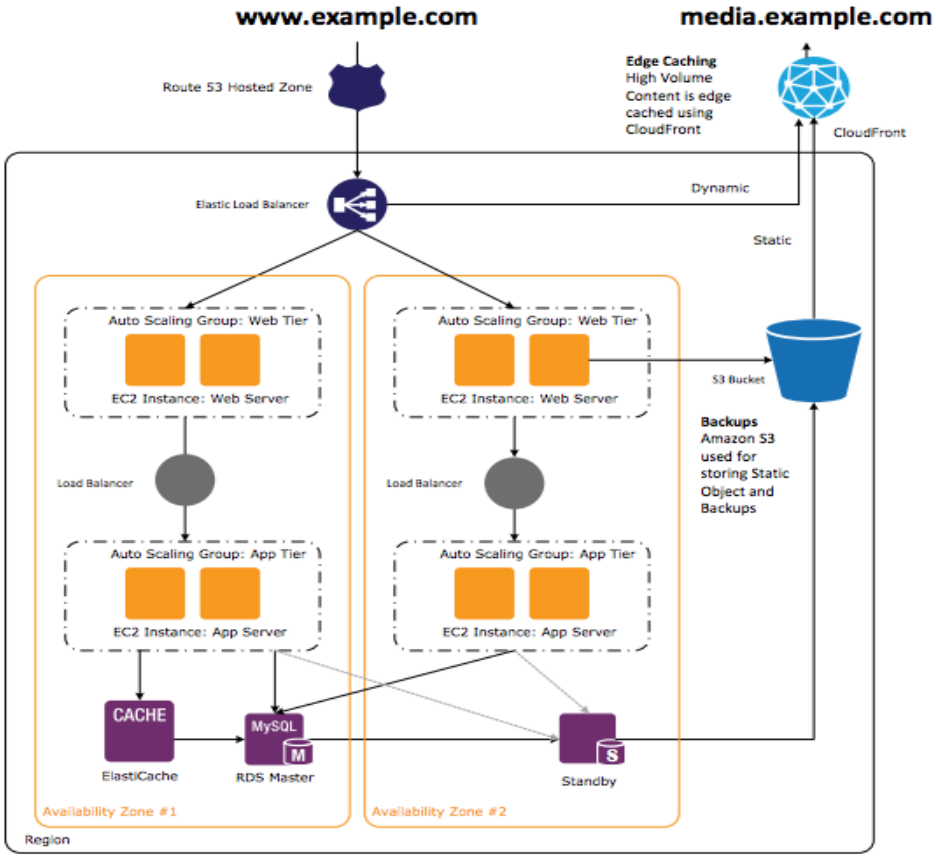

Figure 15. Example of a Web Hosting Architecture on AWS (Amazon Web Services 2012: 6).

Media sharing

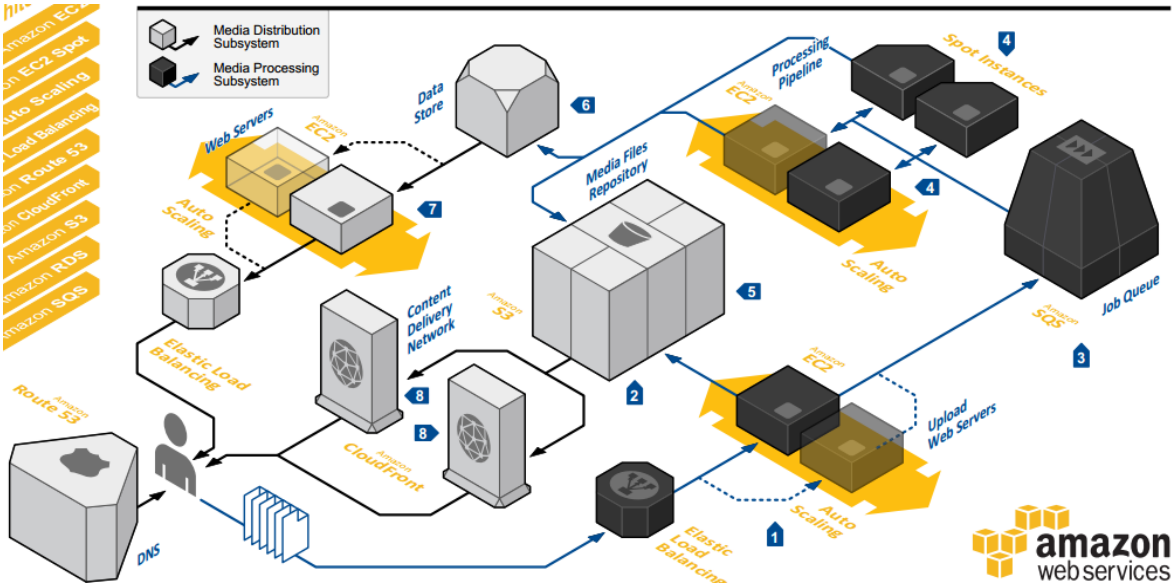

Figure 16. Media sharing. (Amazon Web Services 2014j.)

In figure 16 AWS's reference architecture for media sharing is presented. When compared to web application hosting architecture most of the elements are the same. There are the basic elements like EC2 instances as web servers, auto scaling, balancing and a way for storing data. The only difference is that they are used in a different way in order to perform the service that is intended to do.

4. CURRENT APPLICATION ARCHITECTURE GAP

> "*Moving an application to the cloud is a rearrangement of the application's deployment topology in which component dependencies are captured* (Leymann, Fehling, Mietzner & Nowak 2011: 308). "

The application architecture gaps that have been arisen in this thesis have been collected in table 5. The gaps have been explained and their impact described. In the last column the importance of the gap has been pointed out. The rates are high, medium and low. High being the most important to solve before moving to cloud and low the least important.

Table 5. Current application architecture gap.

Gap	Description	Impact	Importance
Multi-tenancy	Multi-tenancy means that multiple customers, companies, are using the same application and its databases but all of them have separate instances.	Applications used to be before used by only one company and with fewer users. In the cloud the amount of users can be many times bigger than before. Shared application and instances might lead to a security concerns.	High
Scaling model	Scaling instances can be done automatically according to the need. Some vendor's offer services where customer can itself control the need manually.	If scaling is not working, enough instances are not in use and this might lead to for example slowness of the service. Not all of the vendors have the possibility for endless scaling which means that they have a certain limit where the scaling ends. This is typical with old technologies that are not suitable for clouds since they are not designed for them.	High
continues (on next page)…			

…continues (on previous page)			
Access control	Users have different access rights and the customer who is in control of the whole application manages them.	How can one assure that no one else won't be able to access for example application's data when the cloud is shared with other users as well.	High
Security	Customer is responsible for the security if for example data leak happens. Even though they are responsible of this they are not in control of the service itself. New EU law is in preparation regarding security and privacy and it will be available in 2015 or 2016.	Biggest fear is that the other company's users can by accident or some other way access one's data and misuse it in worst case.	High
Availability	Companies used to maintain and control servers and instances. In cloud computing the responsibility is now the service provider's which can affect the availability of the service.	If service provider doesn't have a good back up service or they are not fixing problems fast enough, problems with service availability might occur and users might not be able to access important data or applications.	High
Language framework	Applications are created with using certain programming language.	Service providers accept only certain programming languages which might narrow the list of possible service providers that company can use. Worst scenario is that service providers don't accept the language that is used to code the application in question.	Medium
continues (on next page)…			

…continues (on previous page)

Company policies	Resources are used to maintain by the company and they have had certain policies on them.	Service provider is responsible for maintaining infrastructure that means that those applications and their delivery changes and they are more independent. Enterprise's own policies are harder to follow in these circumstances though it is not impossible.	Low
Middleware technologies	Middleware technologies are based on old technology. They have an impact on how resources can be scaled.	Scaling of middleware technologies might be difficult since they are not designed for cloud. Nowadays many vendors are using Open source technologies that include easier licenses and technologies scalable for cloud.	High
Deployment	Deployment (also known as delivery) models are different in cloud.	Companies might have tens of ways to deliver applications. Since cloud has its own model, companies need to standardize their ways. Some applications might be difficult to change.	Medium
Development	Service providers are responsible for development of the platform in PaaS model. Customer takes care only of the application.	If service provider develops the platform customer has to make sure they also make updates on their application so that it won't stop working. Service provider has to inform customers early about the changes or otherwise they might not be able to update the application.	Medium
Temporary services	Temporary services are needed only for a limited amount of time. The data needed is also temporary.	Since the need is only for certain time, data should be located in the application itself which would lead to greater performance and latency.	Medium

continues (on next page)…

…continues (on previous page)			
Persistent services	Persistent services are not changing often and they use the static data that stays the same.	Data has to be located somewhere where it is easy to store it, since it is not going to be changed. Data storages located near would be good solution or otherwise there might be problems with performance and latency. For example persistent service can be separate instance with data backed up, but offers the data to other dataless running instances.	High
Data storage	Data has to be easy to access and it has to be available whenever needed.	Data storage is important since without it users can't utilize it and applications. Worst scenario is that other processes will stop because right data is not available. Reasons for that regarding data storage might be for example that the location is far away, the storage is down and doesn't have backup resources. Data might be located in many different geographical regions. Legislation might limit where the data can be stored. One has to make sure that the data is consistent in different places.	High
Standardization	Cloud computing's standardization have only begun. Before it, problems with for example trying to change cloud service provider are more difficult or impossible.	Application portability is difficult to implement without standardization. Customer can't move its services to other clouds since they are designed different way. Customer should have a second alternative for cloud services and this is must especially for companies.	High
continues (in next page)…			

…continues (on previous page)			
Platform's infrastructure	Customer doesn't know how the service provider has designed and built the platform's infrastructure.	Customer doesn't know how application and the platform work together. It is difficult to know if there are going to be some problems. Updating the application might have effect on the platform and vice versa.	Medium
Application development	Application developer is responsible to make applications scalable and design them to for failure.	Even though cloud service provider offers scalable instances they are not working if the application itself is not scalable. Also if the application is only designed to work in certain circumstances, it will stop working if a new accident occurs. Developer will also have more time to focus on the coding and developing instead of building everything from scratch.	High
Quality of code	Application can be as good as the developer's code is.	Bad code will lead to a bad quality of the application.	High

In table 6 the different service providers are compared when it comes to certain aspects that have an effect to the application and its architecture. These might create architecture gaps also.

Table 6. Cloud service provider comparison. (Amazon Web Services 2014k; Microsoft Azure 2014d; Google 2014.)

	AWS	Microsoft Azure	Google App Engine
Virtual machines	Yes	Yes	Yes
Storage	Yes	Yes	Yes
Database			
- MySQL	Yes	Yes	Yes
- Microsoft SQL Server	Yes	Yes	No
- Oracle	Yes	No	No
User control	Yes	Yes	Yes
Language			
- PHP	Yes	Yes	Yes
- Java	Yes	Yes	Yes
- Go	No	No	Yes
- Python	Yes	Yes	Yes
- . NET	Yes	Yes	No
- Node.js	No	Yes	No
- Ruby	Yes	No	No
Auto scaling	Yes	Yes	Yes

5. RECOMMENDATIONS AND BEST PRACTICES

Moving current applications to the cloud it's not so easy task since one needs to achieve a state where the application and the cloud environment are compatible with each other. In this chapter the recommendations and best practices of it will be presented.

Cloudstep diagram contains many relevant items about the process of moving applications to the cloud and it should be used as a plan for the migration. Though, it doesn't contain application architecture on substance level. When one starts to think about moving applications to cloud one should first do a plan based on Cloudstep. First it is good to identify company policies that affect the applications and data in it. Second is to identify critical financial and technical aspects that have an effect in the moving process. In the third phase the application itself comes along. Its important characteristics are defined that should be able to transfer to cloud.

After that the possible vendors are examined based on what is needed. After that application should be modified so that it can work in the vendor's cloud that is chosen. This phase should include testing as well. If the test results are not good then one should change the vendor. One should create migration strategy where possible problems arise. This way it would be easy to test certain situations and ensure that nothing is ignored. In the previous chapter, these problems were pointed out and they should be noted in the strategy as problems and also what are their consequences. The last step of the whole process is the implementation of application move to cloud.

Application size matters in the cloud. Before applications were able to be huge and still work well. When the application is moved to cloud it is not completely sure that all of the components will work in a way they are intended to. In this situation the application should be divided in smaller components and these components should be moved to cloud

separately. It is also easier to identify the problems when the application is divided into smaller parts than when one needs to try to identify the problem from one giant application. Applications should be built in a way that they are easy and possible to move from one cloud provider's service to another. This might mean that the usage of for example Microsoft's Azure is not clever since it only has support to Windows applications. It is clever to prepare applications for this since this should be possible in the future when the standardization is in better state.

The language framework used in the applications should be chosen in a way so that many service providers already support it. It is also important that operation team and design team will work in tight collaboration. This way one can ensure that the application will work in the cloud environment but also that is it designed for the right use.

Applications and data should be in two different services provided by different service providers. The first is the primary provider and the second is kind of a backup service. This has to be done just in case something major happens in the primary service provider's cloud. In extreme situation when the cloud is for example down and it can't be fixed right away, company can continue using the services from the second service provider's cloud. This is why it is also important that the language framework used is suitable for many cloud environments so that company doesn't have to do changes to application separately because of the different environments and language compatibility.

Application developers could also start sharing their codes for applications that are running in the cloud environment. Since there are many old applications that probably have to be recoded, sharing already working code would decrease the time used to fix the applications to fit the new environment. This would help the whole industry to develop more and faster. It is not necessary to invent the same methods all over again when one can share and use the already invented one.

Applications should be built on scalable architecture since one element of cloud computing is scalability. Scalable architecture enables scaling instances up and down according to the current need. This has to be taken into consideration when developing applications to fit the cloud environment. Since the cloud environment is multi-tenant, the scalability has to be taken into consideration from that point of view also.

It is important to keep in mind that multiple users use cloud services. The application should be protected so that other users can't access it accidentally for example through some instance. The access control has to be strict since the application's owner is responsible if for example the data is lost.

Data storing is a big question and it can be divided in confidential data and other data. If the data is confidential is shouldn't be stored in the cloud just because other users have also access to the public cloud platform and when the security issues are not solved, it is not wise to store this kind of data in cloud. For example in banking customer specific data shouldn't be stored there. Other kind of data can easily be stored in cloud. In general it doesn't matter where the data is geographically but depending on the legislation it might be strict. For example banks might have strict rules where they can store certain data and this can be for example only in Finland and not outside Finland at all. Then one needs to make sure from service provider that one's data is stored in the certain place and nowhere else.

In table 7 the elements for suitable and not suitable applications are presented. This can be used as a recommendation when thinking about what kind of applications could be moved. It is important to notice that this listing is only made in a general level and there are more elements that make applications suitable for moving to cloud.

Table 7. Application Candidates for Migration to Cloud Computing (Cloud Standards Customer Council 2013: 7).

Suitable Candidates for Cloud	Less Suitable Candidates for Cloud
• Applications that are used by a group of mobile workers to manage their time and activity, and that contribute only limited information to the company's broad management information databases. • Applications that are run infrequently but require significant computing resources when they run. • Applications that are run in a time zone different from that where your company's IT personnel are located. • Development, testing and prototyping of application changes, even if the final applications will be run on your own infrastructure. • Service Oriented Architecture (SOA) applications.	• Applications that involve extremely sensitive data, particularly where there is a regulatory or legal risk involved in any disclosure. These will at minimum require special treatment if they are to be run in a cloud service. • Applications now being run on the company's private network and that are very performance-sensitive. • Applications that require frequent and/or voluminous transactions against an on-premises database that cannot be migrated to cloud computing. • Applications that run on legacy platforms that are typically not supported (or may not be supported in the long run) by cloud providers.

Applications and data that are moved in the cloud need to be that kind of that it is not a major problem if there is a disorder in the availability. If the application needs to be accessed all the time no matter what, then it is not clever to move it in the cloud and it is better to keep the application in company's own network as said in the table 7. But for example data that is not needed all the time can easily be there.

One important thing to notice from the table 7 is that applications that require lots of computing resources are the ones that could be moved to cloud. This is one reason why cloud computing is a good thing and why it should be utilized. Since the need for resources changes between applications, it is better to use them as much needed when they are needed. This way the service provider will take care of the resources and their availability.

The applications should be able to handle all different problems and incidents that might occur in cloud. So when developing them, these should be taken into consideration. The following things might be helpful when these situations occur:

> "*1. Have a coherent backup and restore strategy for your data and automate it*
>
> *2. Build process threads that resume on reboot*
>
> *3. Allow the state of the system to re-sync by reloading messages from queues*
>
> *4. Keep pre-configured and pre-optimized virtual images to support (2) and (3) on launch/boot*
>
> *5. Avoid in-memory sessions or stateful user context, move that to data stores* (Varia 2011: 11)."

The different components should be designed in a way that they are not dependable from others. This means that for example if some element is sleeping, busy or shuts down and dies for some reason, the others will stay alive and work even though part of the system is not working. One should also implement elasticity in the applications. This means that there should be scaling that occurs after certain time for example weekly, scaling is happening just before a big event is about to occur or the scaling is based on demand. (Varia 2011: 12–14.)

Data should be located as close as possible to the computer or processing elements in order to decrease latency. If the data that is needed is located far away but the amount of it is big, it might be wise to first move the data to the cloud and after that run computation. Data that is not going to change often should locate closer to the end-user since it will decrease access latency. (Varia 2011: 16.)

Security should be implemented in every layer of the cloud application architecture (Varia 2011: 17). Customer is responsible of application and network security. If there is going to be confidential data transferring between browser and web service, there should be SSL defined on the server instance. If one is going to store confidential data in the cloud it is good to first encrypt it. It is important that one can manage users and grant them permissions. (Varia 2011: 17–19.)

Cloud Security Alliance has published research that lists the top threats to cloud computing. In 2013 there were nine threats that were defined as the major ones and they all can appear in services based on PaaS model. First threat is data breaches. Researchers say that even if one of the customer's application have a vulnerability that can lead to a situation where attacker can access the application's data in question, it can also get access in to the other user's data in the same public cloud. This is why multi-tenant clouds have to be designed so that these situations wouldn't occur. The second threat is data loss. Data can be lost because of many different reasons like service provider can unintentionally delete it, there can be a physical catastrophe and even customer can forget own encryption key. (Cloud Security Alliance 2013: 8–9.)

The third threat is account or service traffic hijacking. This is something that has been already existing before cloud computing. The fourth threat is insecure interfaces and APIs. The fifth threat is denial of service. Denial of service attack is one form of attack. It is meant to prevent the users to access the applications or data that are in the cloud. The sixth threat is malicious insiders which can be for example current or old workers or even business partners. They have had access to company's network and data before and they are misusing it in order to cause problems with for example availability of company's cloud services. (Cloud Security Alliance 2013: 10–16.)

The seventh threat is abuse of cloud services. Customers can access a large amounts of computing power by using cloud computing. This can be also used in a wrong way and attackers could use this for example to spread malware. The eight threat is insufficient due diligence. Companies are excited of cloud computing because of its promises of cost reduction and operational efficiency. If the customer doesn't know the new cloud environment, new risks might occur since customers might not understand all the risks. The ninth and last threat is shared technology vulnerabilities. (Cloud Security Alliance 2013: 18–21.)

A question why can come up after realizing all the current gaps that exist at the moment. After all the research, it seems that before applications are ready to move to cloud one needs to make lots of changes to them. Even though it seems to require lots of work at the moment, the benefits are greater than what they would be if this is not done. Cloud solution brings costs savings and flexibility. Flexibility can be seen in the ways of working, in the usage of instances and storage, they are all based on the need at each time. One can be wherever in the world and still access the cloud and the applications and data that are located in the cloud. In companies many problems like not having a certain application on your own computer will disappear.

Companies will also benefit for the fact that they don't have to be the one maintaining and controlling the platform itself where the applications are running. This will release more time for the company to concentrate on the work itself what is important for them and leave the control for the service provider. In this way the collaboration between workers might increase also and lead to improvements in company's everyday business. In a long run this investment will pay back all the development work that is needed to do in the beginning.

The challenges that exist in cloud computing need to be solved in order to fully utilize it. In public multi-tenant cloud access control and security and privacy are the main topics that need to be developed more. Companies can utilize the services already even though these challenges still exists but they limit the use of cloud computing.

There are many benefits but also many challenges and because of them one could raise a question asking if it is wise to even move applications to cloud. The situation should be measured and thought through with every single application alone. Some applications are more beneficial to move to cloud than others. This needs to be thought through and if the situation is that in order to move the application many expensive changes has to be made, then one should consider not to do it. In those cases it would be even wiser to create completely new application straight to the cloud and maybe stop using the old one if one

wants to use the cloud environment. Companies should make prioritization lists, which would show applications that are most important for the company to use. That list could be used to figure out the applications that will be moved to cloud even though there might be many changes to the application before it is ready for the moving.

Cloud computing standardization is an important area that needs to be developed more. At the moment there are not standards that all of the service providers and companies are following. Without having commonly used standards things like changing cloud provider can turn out to be difficult. But with having and using them, it is easy. The need for this has been noticed and the process is ongoing but this has to be taken into consideration also when moving applications to cloud. It is important that the service provider is using commonly used standards and not its own.

It is though noticed that the standardization has to change from the old and completely new standards needs to be created and for some old standards the cloud environment needs to be added. Since there are many different forums and groups developing the standards, it might be good to form one bigger group and combine knowledge and create those standards together. Or then someone should gather all the different standards in one place. NIST could be right for that job.

6. CONCLUSIONS

The main objective of this thesis work was to create application reference architecture and recommendations and best practices when it comes to moving an application to cloud environment. Part of the main objective was to find out what will happen to the application architecture. Other research area was the standards and how they might get affected.

It’s not yet an easy task to move applications to cloud. One reason is that there are many applications that are built by using old technology that is not designed for cloud environments. That’s why there are many gaps between the application and cloud environment. These gaps should be solved before the application is moved to cloud. There is a model called Cloudstep that has all the steps necessary in order to perform the migration. One should use it as a plan for the application migration.

Many of the gaps that were found are related to the fact that service provider is responsible for the platform and service itself and customer only for the application. This leads to many situations where the customer doesn't have any control over the cloud. This is understandable since the service model investigated is PaaS. But this is also the reason why at the moment there might be many difficulties to move application to cloud. Also the customer has to be aware all the time if the service provider making updates to the platforms infrastructure since it has also a high impact on the applications performance and availability.

There are many threats that cloud computing are facing. It is important that when creating applications suitable for the cloud environment, these threats are being taken into consideration and methods to prevent them are applied. This is important to notice already when for example coding the application.

Cloud computing's standardization has just begun. There are many different standards but one commonly used source is still missing. At the moment the trend has been that smaller cloud service providers have followed the bigger ones like Amazon. In the future when standardization takes one step forward, cloud computing might have one commonly used standards source which would help customers for example move between clouds that it basically not possible at the moment. It will also make the service more reliable and create competition that always is good since it will lead to improvements and new solutions.

As a conclusion current application architecture gaps have been identified in this work and they need to be fixed in order for even thinking about moving applications to cloud. Some of the gaps can be fixed easily but for example the security and privacy parts might not be easily solved. Also standards have to be created and some of the existing ones need to be changed so that they meet the all the time changing demand.

This research leaves many options for future research. The future research should include implementation phase where a test application would be tested in different cloud environments. The test should include all different scenarios of what is going to happen in the application. The gaps that were found in this thesis should be taken into consideration when creating test cases. Also research regarding cloud security is one of the things that could be investigated more since that still concerns user. It couldn't be examined in a detailed way in this thesis but it would be important to investigate and develop it more in future research.

REFERENCES

Amazon Web Services (2014a). Amazon EC2 Reserved Instance Marketplace [online]. [cited 14.8.2014]. Available from World Wide Web: <http://aws.amazon.com/ec2/purchasing-options/reserved-instances/marketplace/>.

Amazon Web Services (2014b). About AWS [online]. [cited 24.6.2014]. Available from World Wide Web: <URL:http://aws.amazon.com/about-aws/>.

Amazon Web Services (2014c). Amazon Web Services [online]. [cited 14.8.2014]. Available from World Wide Web: <URL:https://console.aws.amazon.com/console/home?&region=us-west-2>.

Amazon Web Services (2014d). Amazon EC2 [online]. [cited 24.6.2014]. Available from World Wide Web: <URL:http://aws.amazon.com/ec2/>.

Amazon Web Services (2014e). Amazon EC2 Product Details [online]. [cited 24.6.2014]. Available from World Wide Web: <URL:http://aws.amazon.com/ec2/details/>.

Amazon Web Services (2014f). Amazon S3 [online]. [cited 24.6.2014]. Available from World Wide Web: <URL:http://aws.amazon.com/s3/?nc1=h_l2_sc>.

Amazon Web Services (2014g). AWS Management Portal for vCenter [online]. [cited 15.8.2014]. Available from World Wide Web: <URL: http://aws.amazon.com/ec2/vcenter-portal/>.

Amazon Web Services (2014h). AWS Architecture Center [online]. [cited 15.8.2014]. Available from World Wide Web: <URL:http://aws.amazon.com/architecture/?nc1=h_l3_cc>.

Amazon Web Services (2014i). Web Application Hosting [online]. [cited 15.8.2014]. Available from World Wide Web: <URL:http://media.amazonwebservices.com/architecturecenter/AWS_ac_ra_web_01.pdf>.

Amazon Web Services (2014j). Media sharing. [online]. [cited 15.8.2014]. Available from World Wide Web: <URL:http://media.amazonwebservices.com/architecturecenter/AWS_ac_ra_mediasharing_09.pdf>.

Amazon Web Services (2014k). Amazon Web Services [online]. [cited 21.8.2014]. Available from World Wide Web: <URL:http://aws.amazon.com/>.

Amazon Web Services (2012). Web Application Hosting in the AWS Cloud - best practices [online]. [cited 15.8.2014]. Available from World Wide Web: <URL:http://media.amazonwebservices.com/AWS_Web_Hosting_Best_Practices.pdf>.

Amin, Mohammed Arif; Kamalrulnizam Bin Abu Bakar & Haider Al-Hashimi (2013). A Review of Mobile Cloud Computing Architecture and Challenges to Enterprise Users. *7th IEEE GCC Conference and Exhibition (GCC), 17th – 20th November 2013, Doha (Qatar)* [online] [cited 1.5.2014], 240–244.

Anderson, Tim (2012). Windows Moves Towards Infrastructure on Demand. *Computer Weekly* [online] [cited 3.7.2014], 16–18.

AWS Documentation (2014). Getting Started with the AWS Management Console [online]. [cited 15.8.2014]. Available from World Wide Web: <URL:http://docs.aws.amazon.com/awsconsolehelpdocs/latest/gsg/getting-started.html#get-billinginfo>.

BCS The Chartered Institute for IT (2012). *Cloud Computing: Moving IT out of the Office.* Swindon, GBR: British Informatics Society. p. 59.

Beserra, P.V., A. Camara, R. Ximenes, A.B. Albuquerqu & N.C. Mendonca (2012). Cloudstep: A step-by-step decision process to support legacy application migration to the cloud. *IEEE 6th International Workshop on the Maintenance and Evolution of Service-Oriented and Cloud-Based Systems (MESOCA), 24th September 2012, Trento (Italy)* [online] [cited 1.6.2014], 7–16.

Business Insider (2014). Five free options for start-ups to get started with cloud computing [online]. [cited 26.8.2014]. Available from World Wide Web: <URL: http://economictimes.indiatimes.com/magazines/panache/five-free-options-for-start-ups-to-get-started-with-cloud-computing/articleshow/38532936.cms>.

Chappell, David (2010). Introducing Windows Azure [online]. [cited 2.7.2014]. Available from World Wide Web: <URL: http://www.utdallas.edu/~muratk/courses/cloud11f_files/Introducing%20Windows%20Azure,%20Final%20PDC10.pdf>.

Choudhary, Vidyanand & Joseph Vithayathil (2013). The Impact of Cloud Computing: Should the IT Department Be Organized as a Cost Center or a Profit Center. *Journal of Management Information Systems* [online] 30:2 [cited 14.6.2014], 67–100.

Cloud Foundry (2014). About [online]. [cited 15.8.2014]. Available from World Wide Web: <URL:http://www.cloudfoundry.org/about/index.html>.

Cloud Security Alliance (2013). The Notorious Nine Cloud Computing Top Threats in 2013 [online]. [cited 28.8.2014]. Available from World Wide Web: <URL:https://downloads.cloudsecurityalliance.org/initiatives/top_threats/The_Notorious_Nine_Cloud_Computing_Top_Threats_in_2013.pdf>.

Cockcroft, Adrian (2013). Cloud Native Architecture at Netflix [online]. [cited 21.8.2014]. Available from World Wide Web: <URL:http://yowconference.com.au/slides/yow2013/Cockcroft-CloudNativeArchitectureNetflix.pdf>.

Columbus, Louis (2012). Roundup of Cloud Forecasts and Market Estimates, 2012 [online]. [cited 20.5.2014]. Available from World Wide Web: <URL:http://www.cloudcomputing-news.net/news/2012/jan/18/roundup-of-cloud-computing-forecasts-and-market-estimates-2012/>.

Cloud Standards Customer Council (2013). Migrating Applications to Public Cloud Services: Roadmap for Success [online]. [cited 21.8.2014]. Available from World Wide Web: <URL:http://www.cloudstandardscustomercouncil.org/Migrating-Apps-to-the-Cloud-Final.pdf>.

Dutta, A., G.C. Peng and A. Choudhary (2013). Risks in enterprise cloud computing: the perspective of IT experts. *Journal of Computer Information Systems* [online] 53:4 [cited 13.6.2014], 39–48.

Dutta, Sourav, Sankalp Gera, Akshat Verma & Balaji Viswanathan (2012). SmartScale: Automatic Application Scaling in Enterprise Clouds. *IEEE Fifth International Conference on Cloud Computing, 24th – 29th June 2012, Honolulu (Hawaii)* [online][cited 28.8.2014], 221–228.

ETSI (2013). Cloud Standards Coordination – Final report [online]. [cited 30.8.2014]. Available from World Wide Web: <URL:http://www.etsi.org/images/files/Events/2013/2013_CSC_Delivery_WS/CSC-Final_report-013-CSC_Final_report_v1_0_PDF_format-.PDF>.

Gartner (2009). Gartner Highlights Five Attributes of Cloud Computing [online]. [cited 15.8.2014]. Available from World Wide Web: <URL:http://www.gartner.com/newsroom/id/1035013>.

Gendron, Michael S. (2014). *Wiley and SAS Business Series : Business Intelligence and the Cloud : Strategic Implementation Guide.* New Jersey: Wiley. p. 242.

Google (2014). App Engine [online]. [cited 15.8.2014]. Available from World Wide Web: <URL:https://cloud.google.com/products/app-engine/>.

Google Developers (2014a). What is Google App Engine? [online]. [cited 25.6.2014]. Available from World Wide Web: <URL:https://developers.google.com/appengine/docs/whatisgoogleappengine?_ga=1.77134930.631693961.1404892761>.

Google Developers (2014b). Instances in the Admin Console [online]. [cited 15.8.2014]. Available from World Wide Web: <URL:https://developers.google.com/appengine/docs/adminconsole/instances>.

Grance, Timothy & Peter Mell (2011). The NIST Definition of Cloud Computing [online]. [cited 25.10.2013]. Available from World Wide Web: <URL:http://csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf>.

Essers, Loek (2013). Dutch banks can use Amazon cloud tech [online]. [cited 1.8.2014]. Available from World Wide Web: <URL:http://www.computerworld.com/s/article/9241175/Dutch_banks_can_use_Amazon_cloud_tech>.

Fogarty, Kevin (2013). In the Cloud, Distance Matters for Compute Efficiency. *Networking Computing - Online* [online] [cited 14.8.2014].

Heino, Petteri (2010). *Pilvipalvelut - Cloud Computing*. Talentum Media Oy. p. 276.

IBM (2010). Chapter 3 – Cloud Deployment and Delivery Models [online]. [cited 1.8.2014]. Available from World Wide Web: <URL:https://www.ibm.com/developerworks/community/blogs/c2028fdc-41fe-4493-8257-33a59069fa04/entry/september_19_2010_1_45_pm7?lang=en>.

IEEE (2014a). IEEE P2301 Working Group[online]. [cited 31.8.2014]. Available from World Wide Web: <URL: http://www.cloudcomputing.ieee.org/standards/standards-guidance-p2301>.

IEEE (2014b). IEEE P2302 Working Group [online]. [cited 31.8.2014]. Available from World Wide Web: <URL: http://www.cloudcomputing.ieee.org/standards/standards-guidance-p2302>.

Jacobs, Edwin (2013). Cloud Computing and EU Financial Services [online]. [cited 26.8.2014]. Available from World Wide Web: <URL:http://www.timelex.eu/en/blog/detail/cloud-computing-and-eu-financial-services>.

Jaeger, Jaclyn (2010). Weighing Risks, Benefits of Cloud Computing & SaaS. *Compliance week* [online] 7:83 [cited 5.11.2014], 56–57.

Jones, Penny (2012). ING Group commits to the Cloud [online]. [cited 1.8.2014]. Available from World Wide Web: <URL:http://www.datacenterdynamics.com/focus/archive/2012/06/ing-group-commits-cloud>.

Leymann, Frank, Christopg Fehling, Ralph Mietzner & Alexander Nowak (2011). Moving Applications to the Cloud: An approach based on application model enrichment. *International Journal of Cooperative Information Systems* [online] 20:3 [vited 1.7.2014], 307–356.

Lewis, Grace A. (2012). The Role of Standards in Cloud-Computing Interoperability [online]. [cited 14.8.2014]. Available from World Wide Web: <URL:http://resources.sei.cmu.edu/asset_files/TechnicalNote/2012_004_001_28143.pdf>.

Liu, Fang, Jin Tong, Jian Mao, Robert Bohn, John Messina, Lee Badger & Dawn Leaf (2011). NIST Cloud Computing Reference Architecture [online]. [cited 26.10.2013]. Available from World Wide Web: <URL:http://www.nist.gov/customcf/get_pdf.cfm?pub_id=909505>.

Louridas, Panos (2010). Up in the Air: Moving your Applications to the Cloud. *IEEE Software* [online] 27:4 [cited 16.8.2014], 6–11.

McDonald, Kevin (2010). Above the Clouds: Managing Risk in the World of Cloud Computing. Gambs , GBR: IT Governance Publishing. p. 166.

Marks, Eric A & Bob Lozano (2010). *Cloud Computing*. New Jersey: John Wiley & Sons. p. 302.

Microsoft Azure (2014a). Azure Execution Models [online]. [cited 1.7.2014]. Available from World Wide Web: <URL:https://azure.microsoft.com/en-us/documentation/articles/fundamentals-application-models/>.

Microsoft Azure (2014b). WorkerRole Schema. [online]. [cited 1.7.2014]. Available from World Wide Web: <URL:http://msdn.microsoft.com/en-us/library/azure/gg557552.aspx>.

Microsoft Azure (2014c). WebRole Schema [online]. [cited 1.7.2014]. Available from World Wide Web: <URL:http://msdn.microsoft.com/en-us/library/azure/gg557553.aspx>.

Microsoft Azure (2014d). Microsoft Azure Schema [online]. [cited 21.8.2014]. Available from World Wide Web: <URL:https://azure.microsoft.com/en-us/>.

Mongan, Kieran (2011). Cloud Computing - The Storm Is Coming. *Accountancy Ireland* [online] 43:3 [cited 4.4.2014], 58–60.

OpenStack (2014a). OpenStack: The Open Source Cloud Operating System [online]. [cited 6.7.2014]. Available from World Wide Web: <URL:http://www.openstack.org/software/>.

OpenStack (2014b). OpenStack Compute [online]. [cited 6.7.2014]. Available from World Wide Web: <URL:http://www.openstack.org/software/openstack-compute/>.

OpenStack (2014c). OpenStack Storage [online]. [cited 6.7.2014]. Available from World Wide Web: <URL:http://www.openstack.org/software/openstack-storage/>.

OpenStack (2014d). OpenStack Networking [online]. [cited 6.7.2014]. Available from World Wide Web: <URL:http://www.openstack.org/software/openstack-networking/>.

OpenStack (2014e). OpenStack Dashboard [online]. [cited 15.8.2014]. Available from World Wide Web: <URL:http://www.openstack.org/software/openstack-dashboard/>.

Parameswaran, A. V. & Asheesh Chaddha (2009). Cloud Interoperability and Standardization [online]. [cited 14.8.2014]. Available from World Wide Web: <URL:http://www.infosys.com/infosys-labs/publications/infosyslabs-briefings/documents/cloud-interoperability-standardization.pdf>.

Plamondon, Scott (2011). Cloud Performance: Crucial New Facts They Forgot to Mention [online]. [cited 14.8.2014]. Available from World Wide Web: <URL:http://www.riverbednews.com/2011/08/cloud-performance-crucial-new-facts-they-forgot-to-mention/>.

Rackspace (2014a). About Rackspace [online]. [cited 2.8.2014]. Available from World Wide Web: <URL:http://www.rackspace.com/about/>.

Rackspace (2014b). Rackspace Cloud Sites Essentials: Cloud Sites Technologies [online]. [cited 14.8.2014]. Available from World Wide Web: <URL:http://www.rackspace.com/knowledge_center/article/rackspace-cloud-sites-essentials-cloud-sites-technologies>.

Reed, Paul (2002). Reference Architecture: The best of best practices [online]. [cited 2.8.2014]. Available from World Wide Web: <URL:http://www.ibm.com/developerworks/rational/library/2774.html>.

Remde, Kevin (2011). SaaS, PaaS, and IaaS.. Oh my! ("Cloudy April" - Part 3) [online]. [cited 1.8.2014]. Available from World Wide Web: <URL:http://blogs.technet.com/b/kevinremde/archive/2011/04/03/saas-paas-and-iaas-oh-my-quot-cloudy-april-quot-part-3.aspx>.

Rubens, Paul (2012). Could Latency Kill the Cloud? [online]. [cited 14.8.2014]. Available from World Wide Web: <URL:http://www.enterprisenetworkingplanet.com/netsysm/could-latency-kill-the-cloud.html>.

Rutland, Duncan (2012). Clouds: Economics [online]. [cited 2.8.2014]. Available from World Wide Web: <URL:http://c1776742.cdn.cloudfiles.rackspacecloud.com/downloads/pdfs/WhitePaper_CloudEconomics.pdf>:

Salo, Immo (2010). *Cloud computing - palvelut verkossa.* Jyväskylä: Docendo. p.168.

Salo, Immo (2012). *Hyötyä pilvipalveluista.* Jyväskylä: Docendo.. p. 191.

SAS 70 (2014). SAS 70 Overview [online]. [cited 25.8.2014]. Available from World Wide Web: <URL: http://sas70.com/sas70_overview.html>.

The Open Cloud Manifesto (2014). The Open Cloud Manifesto: A call to action for the worldwide cloud community [online]. [cited 15.8.2014]. Available from World Wide Web: <URL:http://gevaperry.typepad.com/Open%20Cloud%20Manifesto%20v1.0.9.pdf>.

Travers Smith (2013). New EU data protection law: time to start preparing [online]. [cited 14.8.2014]. Available from World Wide Web: <URL:http://www.traverssmith.com/media/1305907/ts4-_20639303-v1-new_eu_data_protection_law_briefing_may_2013_-_final_pdf.pdf>.

Urquhart, James (2009). Open Cloud Manifesto now signed and delivered [online]. [cited 15.8.2014]. Available from World Wide Web: <URL:http://www.cnet.com/news/open-cloud-manifesto-now-signed-and-delivered/>.

Varia, Jinesh & Sajee Mathew (2014). Overview of Amazon Web Services - January 2014 [online]. [cited 1.7.2014]. Available from World Wide Web: <URL:https://d36cz9buwru1tt.cloudfront.net/AWS_Overview.pdf>.

Varia, Jinesh (2011). Architecting for the Cloud: Best Practices - January 2011[online]. [cited 21.8.2014]. Available from World Wide Web: <URL:http://media.amazonwebservices.com/AWS_Cloud_Best_Practices.pdf>.

VMWare (2011). VMware Delivers Cloud Foundry, The Industry's First Open PaaS [online]. [cited 15.8.2014]. Available from World Wide Web: <URL:http://www.vmware.com/company/news/releases/cloud-foundry-apr2011>.

Ward, Thomas & Vasanthi Gopal (2014). How They Did It: Moving IBM's Smarter Supply Chain to the Cloud. *Supply Chain Management Review* [online] 18:2 [cited 1.6.2014], 26-31.

White, Nick (2014). A cloud over business. *Intermedia (0309118X) Spring 2014* [online] 41:1 [cited 1.6.2014], 29–31.

Williams, Mark (2010). *A Quick Start Guide to Cloud Computing - Moving your business into the cloud.* London: Kogan Page. 1st ed. p. 139.

Zynga Engineering (2014). The Evolution of zCloud [online]. [cited 15.8.2014]. Available from World Wide Web: <URL:http://code.zynga.com/2012/02/the-evolution-of-zcloud/>.