

Lappeenranta University of Technology
Faculty of Technology Management
Department of Information Technology

**Platform as a Service – new opportunities for software
development companies**

Examiners: Professor Kari Smolander
M.Sc. Tero Pesonen

Instructor: Professor Kari Smolander

Lappeenranta, May 12, 2010

Vadim Matveev
Punkkerikatu 2 A 6
53850 Lappeenranta
vadim.matveev@lut.fi
Tel. 0466 154885

ABSTRACT

Lappeenranta University of Technology
Faculty of Technology Management
Department of Information Technology

Vadim Matveev

Platform as a Service – new opportunities for software development companies.

Master's Thesis

2010

89 pages, 14 figures and 1 table

Examiners: Professor Kari Smolander, M.Sc. Tero Pesonen

Keywords: cloud computing, Platform as a Service, PaaS, economical estimation, effectiveness, risks, strategy

Today cloud computing is the next stage in development information-oriented society in field of information technologies. Great attention is paid to cloud computing in general, but the lack of scientific consideration to components brings to the problem, that not all aspects are well examined.

This thesis is an attempt to consider Platform as a Service (a technology of providing development environment through the Internet) from divergent angles. Technical characteristics, costs, time, estimation of effectiveness, risks, strategies that can be applied, migration process, advantages and disadvantages and the future of the approach are examined to get the overall picture of cloud platforms.

During the work literature study was used to examine Platform as a Service, characteristics of existent cloud platforms were explored and a model of a typical software development company was developed to create a scenario of migration to cloud technologies.

The research showed that besides all virtues in reducing costs and time, cloud platforms have some significant obstacles in adoption. Privacy, security and insufficient legislation impede the concept to be widespread.

FOREWORD

This master's thesis was written in the Department of Information Technology at Lappeenranta University of Technology during the spring of 2010.

First of all I would like to thank my supervisors Professor Kari Smolander and M.Sc. Tero Pesonen for suggested contemporary and interesting research topic, valuable advices and for supporting and guiding me through the whole process.

Also I want to thank everyone who had helped me during this period in those fields where my competence was insufficient and who stood by my side: Petr Bavin, Lizabeth Parakhina and Daria Volchek. I want to express my gratitude to my family for believing in me as well.

Lappeenranta, May 12, 2010

Vadim Matveev

Table of Contents

1	INTRODUCTION	6
1.1	Background	6
1.2	Objectives and restrictions	7
1.3	Structure of the thesis	7
2	CLOUD COMPUTING APPROACH.....	8
2.1	Cloud computing evolution.....	8
2.2	Software as a Service.....	13
2.3	Infrastructure as a Service.....	17
3	PLATFORM AS A SERVICE	19
3.1	Features	19
3.2	Integrated lifecycle platforms.....	22
3.2.1	Google App Engine	23
3.2.2	Microsoft Azure Platform.....	29
3.3	Anchored lifecycle platforms.....	40
3.3.1	Salesforce.com	40
3.3.2	Intuit.....	42
3.4	Technologies as a platform	44
4	A SCENARIO OF MIGRATION TO CLOUD TECHNOLOGIES	46
5	SECURITY AND PRIVACY	54
5.1	Security	54
5.2	Privacy	57
6	ECONOMIC EVALUATION	62
6.1	Costs	62
6.2	Risks	70
6.3	Strategy	73
7	RESULTS AND DISCUSSIONS.....	78
7.1	Advantages and disadvantages.....	78
7.2	PAAS in the future	79
8	CONCLUSION.....	83
	REFERENCES.....	84

ABBREVIATIONS

API	Application Programming Interface
ASP	Application Service Provider
CLR	Common Language Runtime
CPU	Central Processing Unit
CRM	Customers resource management
CSS	Cascade Style Sheets
CU	Conventional Units
DBMS	Database Management System
DDoS	Distributed Denial of Service
FC	Fixed Costs
HTML	HyperText Markup Language
HTTP	Hypertext Transfer Protocol
IaaS	Infrastructure as a Service
IDE	Integrated Development Environments
IIS	Internet Information Services
ISP	Internet Service Provider
IT	Information Technology
ITIL	IT infrastructure library
JDA	Java Data Objects
JPA	Java Persistence API
JRE	Java SE Runtime Environment
JSP	Java Server Pages
JVM	Java Virtual Machine
NAT	Network Access Translation
PaaS	Platform as a Service
PC	Personal Computer
REST	Representational State Transfer
SaaS	Software as a Service
SDK	Software Development Kit
SLA	Service Level Agreement
SOAP	Simple Object Access Protocol
SQL	Structured Query Language

TCP	Transmission Control Protocol
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
VC	Variable Costs
VM	Virtual Machine
WCF	Windows Communication Foundation
XML	Extensible Markup Language

1 INTRODUCTION

1.1 Background

The gradual process of technologies' development, companies' specialization and wide-spread occurrence of the Internet lead to the result that instead of holding on-site equipment it is possible to lease them from service providers where more facilities are concentrated in one place and can be shared effectively between different organizations all over the world. That concept is not new and the idea that computer resources would be sold as water or electricity was described by Professor John McCarthy in 1961 (Khalid, 2010). Due to transition from centralized data processing to personal under the appearance of personal computers (PC) and a lag of techniques led to the situation that creation of clouds was impossible down to recent times.

Cloud computing provides more flexibility than traditional systems. It has almost unlimited resources in data processing, distributing and storing and companies afford to lease as much resources as required for a specified term when their systems are most demanded and refuse them on the contrary, e.g. at nights. There are attractive points for businesses in getting more flexibility, reducing costs and time; however there are some disadvantages, which put obstacles in adopting the cloud everywhere. Security and privacy are the most significant issues that should be improved.

The cloud approach can be applied for applications' development process by the use of special platforms and environment, which provide an access to remote resources. Platform as a Service offers opportunities for software companies to create applications easier, concentrating on business processes instead of coding and maintenance, reduce costs, associated with hardware and software, anticipate possible problems in scalability and carry out the whole development lifecycle within the same environment.

Platform as a Service will progress in the future, will overcome existent limitations to be the mainstream in developing web-based applications. However,

completely reject traditional approach is unlikely due to the specific of legacy systems and tasks, where high level of abstraction is unacceptable.

1.2 Objectives and restrictions

The primary purpose of the thesis was to examine opportunities which cloud computing provides for software development companies and end users both from technical and economical sides. As a result the complex assessment of the approach can be used to reveal pitfalls and warn potential customers against possible issues in adoption cloud platforms and to estimate reasonability of transition to cloud technologies.

1.3 Structure of the thesis

Section 2 is a survey of the cloud computing approach, describing the features of this contemporary technology and some components, e.g. Software as a Service and Infrastructure as a Service separately. Section 3 covers the technical characteristics of cloud platforms and describes different types of platforms, such as integrated lifecycle platforms, anchored lifecycle platforms and technologies as a platform with investigation of providers' services. Section 4 discusses the aspects of migration process. Section 5 presents the main obstacles in adoption cloud computing overall: privacy and security. Section 6 discloses the economical side of the field, considers costs and effectiveness of using the platforms, evaluates possible risks and describes strategies which a company can develop to gain a competitive edge and make additional profit. Section 7 reveals the results of the work, identifies the advantages and disadvantages for software development companies as compared with the traditional technologies and sees ahead to the future of the approach. Section 8 describes information and results gathered during the work.

2 CLOUD COMPUTING APPROACH

Nowadays cloud computing revolutionizes the way how companies operate with data and applications. Such processes as inventing, developing, deploying, scaling, updating, maintaining and paying for resources undergo the changes (McAlphine, 2010). Cloud computing is not only a term, that refers to “data, processing, or experiences that “live” out there somewhere in the cloud we call the Internet” (Sarrel, 2009). It becomes a rather common word and as all modern tendencies is discussed everywhere in the Internet. Different wordings exist for the term cloud computing, that is why understanding what is cloud computing indeed and defining it is a primary task.

Cloud computing gets its name from representation of the Internet as a cloud in network diagrams. That is something that hides topology, infrastructure, logic, etc. of the net from them who are not interested in and just consume it. Data transferring mechanism from one point to another is common depiction an outline of a cloud and dates back to 1961 (Dupre, 2008). In that year professor John McCarthy presented the idea that computer resources, power and applications would be sold as water or electricity. That idea was very popular during 1960s but hardware, software and communication technologies were not on the required level thus it was discarded. Since 2000s the idea has revived and today turns to cloud computing concept, where computing positions as a public utility. In other words cloud computing is next stage of the internet development, on which all resources and processes, such as computing power, infrastructure, applications, business processes, etc. can be delivered to end users as a service through the Internet.

2.1 Cloud computing evolution

Toffler noticed that civilization goes through three waves: agricultural, industrial and information, each of them has several subwaves (Toffler, 1984). Concerning the following information age many people think that now is the beginning of cloud computing era. A good example was presented in (Carr, 2009), where previously various organizations had their own water wheels and windmills to

serve their needs, but with the appearance of electrification and transition to another level, these companies just plug to the electrical grid. Carr assures that cloud computing will do the same with contemporary information society. Today almost all organizations have their own resources (hardware, software, and staff) to execute their business processes, but in the coming future they will rent it and pay only for resources, they really consume, and that becomes a reality these days. Just plug to cloud and use facilities.

As information wave consists of some subwaves (Mather, Kumaraswamy & Latif, 2009), the evolution process could be shown as presented in the figure 2.1. That era starts with appearance of mainframes and during the progress of information technologies minicomputers, PCs, client-server technologies, IP networks and mobile devices have arisen. Cloud computing is a new subwave where our society is moving due to the technical progress, differentiation of labor and specialization as it took place in ancient society.

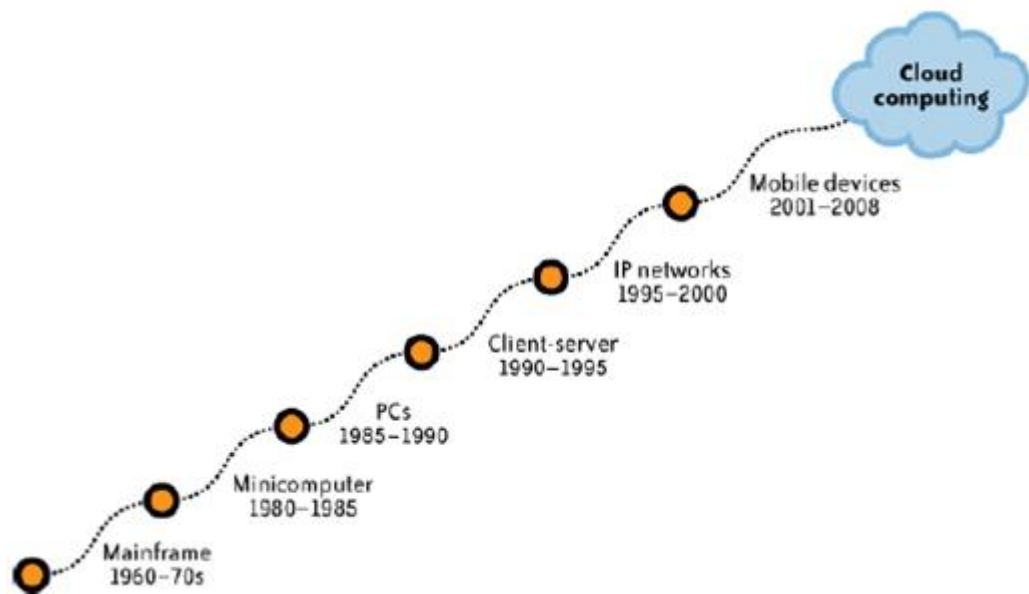


Figure 2.1 – Subwaves within the information age (Mather, Kumaraswamy & Latif, 2009).

The evolution process of cloud computing consists of several phases where Internet Service Providers (ISP) have grown and turned to what we have today. At

the beginning of development ISPs just provided access to the Internet for organizations and users, it was the first generation. Then ISPs started to provide mail and servers access (ISP 2.0). These equipments grew to facilities for hosting organizations' servers with the infrastructure and applications, also known as collocation facilities (ISP 3.0). After that phase appeared application service providers (ASP) which do not just offer computing infrastructure, but provide special applications for organizations (ISP 4.0). ASPs are initial stage of Software as a Service (SaaS) approach, which is one of the parts of cloud computing technology, but the main difference is that ASPs provide applications on the dedicated environment (host or server) while SaaS uses shared infrastructure. Cloud computing (ISP 5.0) is a contemporary phase of ISP development and based on “Everything as a Service” principle. Figure 2.2 illustrates whole the evolution process of cloud computing.

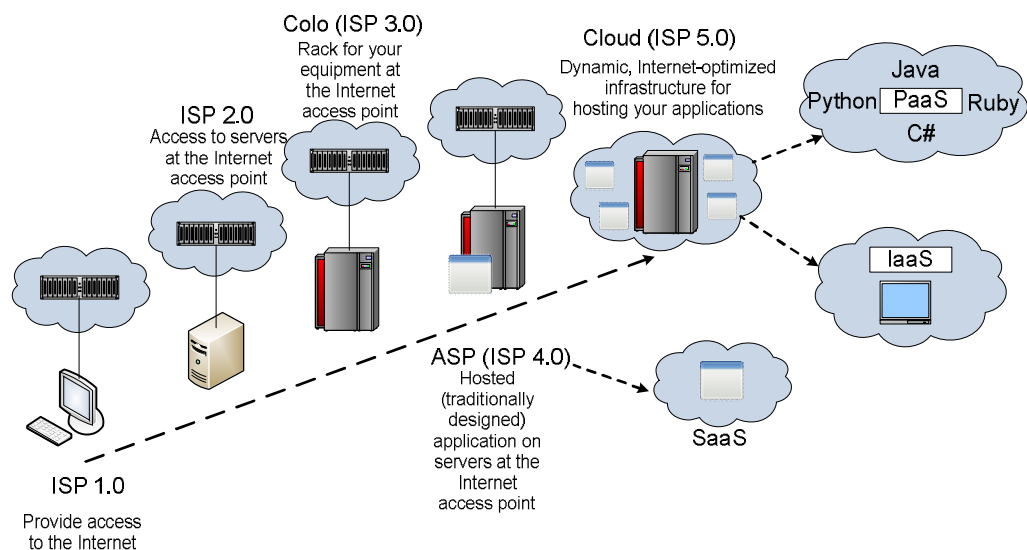


Figure 2.2 – Evolution of cloud computing (Mather, Kumaraswamy & Latif, 2009).

The concept 'as a Service' means few features; they characterize the approach and are appropriate to cloud computing methodology (Velte, Velte & Elsenpeter, 2009):

- Low barriers to entry (computers capable to work with web applications and Internet connection almost all is required)
- Large scalability (additional resources can be easily assigned)
- Multitenancy (shared usage of the same equipment)
- Device independence (different hardware can be used)

Cloud computing is composed of different approaches, which are related with the idea of providing everything as a service. Following directions of cloud computing could be examined:

- Software as a Service (SaaS)
- Platform as a Service (PaaS)
- Infrastructure as a Service (IaaS)

However besides these fields secondary elements like Hardware as a Service, Workplace as a Service, Data as a Service and Security as a Service could be defined, but they are considered to be more or less parts of three major services. There are several deployment models, concerning how these services might be provided: public, hybrid and private. In the figure 2.3 the comprehensive model, describing relations between services, uses and types of clouds, is presented.

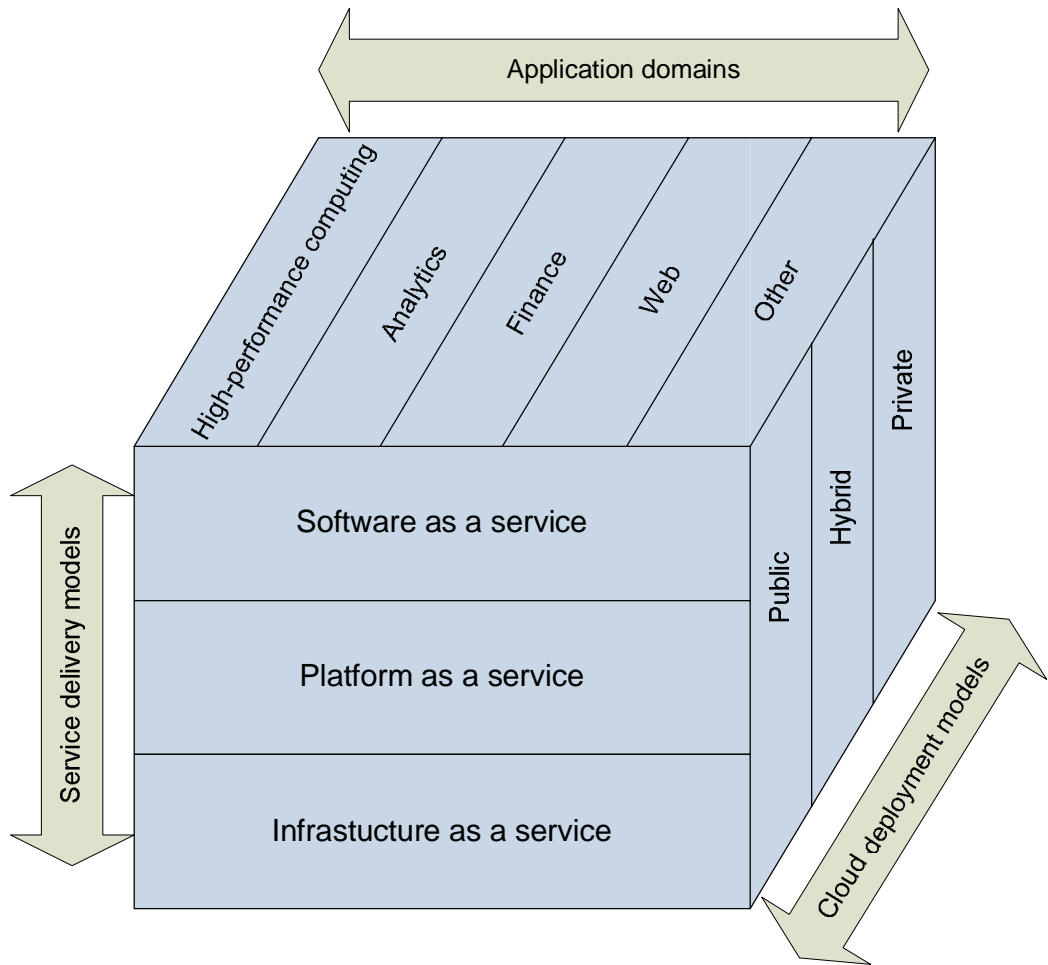


Figure 2.3 – cloud computing service model (Mather, Kumaraswamy & Latif, 2009).

Up-to-day cloud providers have three major services and the cloud computing delivery model consists of three layers, which are presented in the figure 2.4. But that is rather complex for companies to be in every field and as examples illustrate no one is situated on all layers.

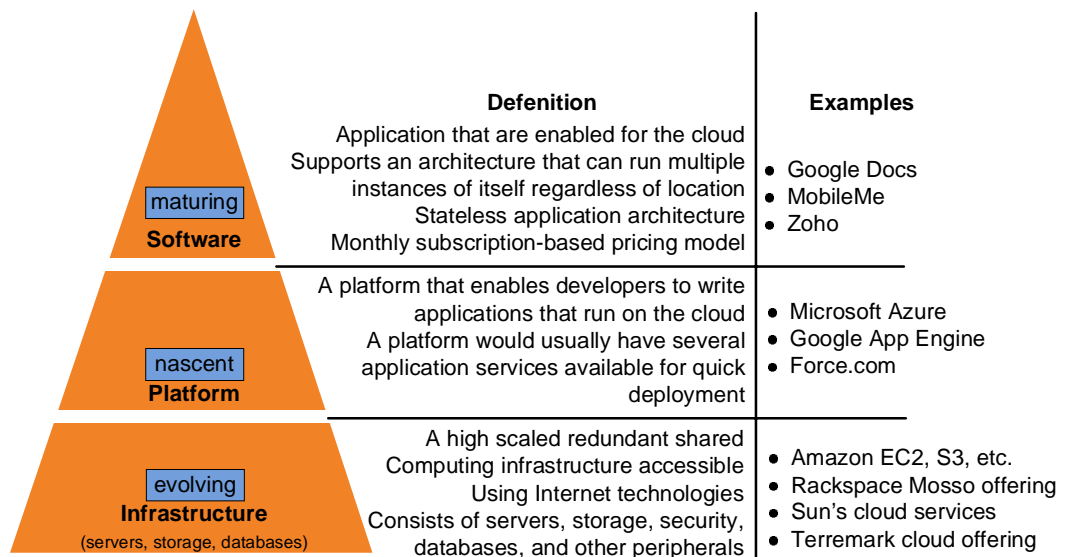


Figure 2.4 – Cloud services delivery model (Mather, Kumaraswamy & Latif, 2009).

2.2 Software as a Service

According to (Levinson, 2009) SaaS “is software that is developed and hosted by the SaaS vendor and which the end user customer accesses over the Internet. Unlike traditional packaged applications that users install on their computers or servers, the SaaS vendor owns the software and runs it on computers in its data center. The customer does not own the software but effectively rents it, usually for a monthly fee. SaaS is sometimes also known as hosted software or by its more marketing-friendly cousin, on-demand”

In this delivery model data is stored on service provider’s servers and cached temporarily on client devices (desktops, tablet computers, notebooks, hand-held devices, mobile phones, etc.) Organizations, which are served by a provider called tenant and this type of agreement is called a multitenant architecture. SaaS is rather developed as compared with PaaS and IaaS which are still in the initial phases. That approach exempts organizations from different duties, like maintaining and supporting, but at the same time changes in software are not now on their control. Companies just use the software as is and are not assumed to adapt applications to own needs. Thus not all applications could be transferred to

SaaS. The following applications' fields are appropriate candidates to be implemented as SaaS solutions (Velte, Velte & Elsenpeter, 2009):

- Accounting
- Customers resource management (CRM)
- Video conferencing
- IT (Information Technology) service management
- Web analytics
- Web content management
- Email software
- Human resources

SaaS provides network-based access to commercial software and is adapted to use web tools, like web-browser, thus using a SaaS application becomes feasible wherever Internet access is.

Providers also update and patch the software and take care of infrastructure, which means that customers do not need to have high performance computing power, servers and staff, maintaining it. Cost policy is rather flexible – customers do not pay for owning the software only for using and they should pay only for resources and software licenses they are really using. That type of licensing is also known as pay-as-you-go subscription licensing model. In traditional model if a company buys software for all organization, but certain user launches it only ones a month for making a monthly report the company loses money. Cloud computing makes it possible to avoid such loses.

SaaS applications' price is usually based on the number of users who have access to the service; extra charge is possible for using of help desk, advanced bandwidth and storage. For vendors, initial revenue of using SaaS instead of traditional license fees is lower, but during following periods, investments bring profit more efficiently.

In accordance with classification given by Microsoft there are four levels of SaaS architecture maturity on basis of attributes, like configuration simplicity, multitenant efficiency and scalability. All of them are presented below:

SaaS Architectural Maturity Level 1 – Ad-Hoc/Custom. This is the first and the basic level, really is no maturity at all. Customers have customized versions of the applications, which have their own instances on the servers. Transferring from the traditional type of software to this level requires minimum development effort and reduces costs, consolidating server hardware and maintenance.

SaaS Architectural Maturity Level 2 – Configurability. This level is characterized by configuration metadata; as a result program flexibility is higher. The same application could be used by great number of customers simultaneously due to usage of detailed configuration options. Updating code base becomes simpler and vendors' maintenance works do not need a lot of efforts.

SaaS Architectural Maturity Level 3 – Multitenant Efficiency. The difference between the current level and previous is added multitenancy. Single program instance is appropriate for all customers. As a result vendors' resources are used more efficient while the end users do not discover any differences. At the same time this level is limited in scalability.

SaaS Architectural Maturity Level 4 – Scalable. At the fourth level multitiered architecture is applied and scalability is grown. The main characteristic feature is possibility to run identical application instances on variable number of servers in accordance with dynamically changing load.

The main SaaS software characteristics (Rittinghouse, Ransome, 2009) are:

- Commercial software is accessed and managed by network from central locations, users work remotely via the Internet.
- One-to-many delivery model: single-instance, multitenant architecture.
- Centralized patching and updating exempt users from downloading and installing anything.

Benefits of using SaaS model are obvious:

- Effective administration. Maintenance functions are executed from the one point in the provider's data center.
- Automated update and patch management. Single administration point allows to apply updates and patches to servers' equipment at a time, as a result changes are available immediately to all end-users.
- Data compatibility within the organization. Software versions are the same for everybody.
- Simplified collaboration. Documents and data are shared between team's members and make possible to work simultaneously on the same things.
- Global accessibility. An access is from everything where the Internet is.
- Saving money comparing with buying application. Rental bill system allow to use exactly what and when is needed.
- Increased reliability. Provider's data centers usually are more reliable than customers' environment.
- World Wide Web type application standards allow users to adapt quickly to the new software. Standardized user interface of applications is easily learned.
- Cut costs of IT staff work due the scale. The main advantage of cloud computing is easy way to scale customer's application, resources consumption may be increased in few clicks.

Besides SaaS benefits there are some obstacles, like:

- Oriented to mass usage. Inappropriate for specific computational needs, SaaS is for the great number of customers with the same needs.
- Impossibility to port application to another vendor. Migration between vendors is not a trivial task: different data formats and computing processes raise obstacles for leaving one vendor and moving to another. Or considerable extra fee for moving can be charged by the provider or third party.
- Can face with open source applications and cheaper hardware. While the migration or leaving certain vendor is sophisticated.

- Security and privacy issues. Are the same as for PaaS and described below in chapter [5 Privacy and Security](#).

2.3 Infrastructure as a Service

Another approach in cloud computing technology is Infrastructure as a Service, where computer infrastructure is delivered as a service. In general, customers use virtualization environment and running software on it. The main points and advantages are the possibilities to offer pay model, depending on usage and service scalability relating to users demand. Beside only hosting applications IaaS vendors offer application support, development and enhancements. IaaS is close to utility computing, where computing services are offered as utilities. Different characteristics like amount of processing power, disk space, etc. are the base for estimation how much the customer should pay and the policy “pay-as-you-go” is fair from customers’ point of view, because they pay exactly for what they consume. Usually customers pay monthly fee for the services, like using utility companies' resources.

Basically IaaS consists of several layers (Rittinghouse, Ransome, 2009):

- Hardware: the physical equipment of a computer system.
- Network: a layer intended for interaction between devices.
- Internet connectivity: provides accessibility from the outside.
- Virtualization environment: an abstract platform, which hides the physical characteristics of the environment from users.
- Service-level agreements: service contracts, defining the level of services.
- Utility computing billing: based upon the usage of resources, service providers bill the customer (typically monthly).

Users do not need to know about the infrastructure where their applications work, they are free from the physical resources, location, data partitioning, scaling, security, backup and other problems.

Benefits of using IaaS are following:

- Preconfigured, secured generally IT infrastructure library (ITIL)-based environment is used. Best practice solutions for infrastructure architecture are offered to customers.
- Flexible scalability is offered. Service-demand peak handling in near-real-time speed, depending on users' requirements.
- Pay-as-you-go type of payment system. Customers pay actually what they consume instead of capital investments.
- Best-of-breed and latest technologies and concentrated IT knowledge are provided. High skilled administrators and experts should be employed, because they are in provider's data centers and solving tasks is not customer's duty more.

Obstacles of using IaaS besides common for cloud computing (security and privacy) are (Turner, 2009):

- Customization. Impossibility to adapt high-security models to a company and customize offered by a provider.
- Availability. Lack of real guarantees that service will be alive 24/7.
- Possibility to be used not in the right way. Virtual machines and almost unlimited processing capacities can be used for illegal activities, e.g. spamming.

Cloud computing is next phase in developing computers and it provides opportunities, which were impossible before. SaaS and IaaS as components of the approach have significant impact to the idea of delivery software and infrastructure, but there are still obstacles in adoption these technologies. The gradual process of developing cloud computing and improving its issues will let to replace more equipment and solutions to the cloud.

3 PLATFORM AS A SERVICE

Platform as a Service is another application delivery concept where resources needed to build applications and services should not to be downloaded and installed, but are accessible through the Internet. It is not so simple to draw a distinct line between PaaS and IaaS, also companies that provide infrastructure services offer platform services as well. Services that PaaS provides are as follows (Velte, Velte & Elsenpeter, 2009):

- Application design
- Development
- Testing
- Deployment
- Hosting
- Team collaboration
- Web service integration
- Database integration
- Security
- Scalability
- Storage
- State management
- Versioning

3.1 Features

In general PaaS approach is characterized by some features (Hurwitz et al, 2009):

- Delivery through the Internet. As all cloud technologies it is accessible from every point in the world, where an Internet connection is.
- Development language availability. The majority of high-level languages are supported depending on the platform and vendor, providing it.

- Customers’ environments are isolated from each other. By means of this safety is achieved, customers’ applications and data cannot be accessible to others as well as changes in one environment do not influence on the rest.
- Whole development lifecycle is supported. Possibilities to deploy, manage, test and maintain applications within the bounds of the safe environment.
- Defined and documented service interfaces are included. Required to support applications and services compositing and orchestration.

It is not obligatory that all features are realized by one vendor. According to a part of functions, implemented by certain provider, companies offer various facilities.

Although software development process develops and changes it still has some problems, that should be solved. There is a sufficient quantity of programs, which are developed in one environment, tested in other and deployed in third. PaaS has a solution: development, testing and deployment operations could be done in the same integrated environment. As a result that excludes costs connected with supporting of separated environments.

PaaS usually provides some tools for creating user interface and as a rule based on HyperText Markup Language (HTML) and JavaScript. It also supports web development interfaces to construct multiple web services with Simple Object Access Protocol (SOAP) and Representational State Transfer (REST), and supports databases and other services (Rittinghouse, Ransome, 2009).

Web service is a “software system designed to support interoperable machine-to-machine interaction over a network. It has an interface described in a machine-processable format. Other systems interact with the Web service in a manner prescribed by its description using SOAP-messages, typically conveyed using Hypertext Transfer Protocol (HTTP) with an Extensible Markup Language (XML) serialization in conjunction with other Web-related standards.” (The World Wide Web Consortium, 2004). Web services use special type of messages – SOAP, that “provides the definition of the XML-based information which can

be used for exchanging structured and typed information between peers in a decentralized, distributed environment” (The World Wide Web Consortium, 2007).

REST is a style of software architecture for distributed hypermedia systems (Fielding, 2000) that “separates a server’s implementation from a client’s perception of resources. It enables transfer of data in streams of unlimited size and type, supports intermediaries as data transformation and caching components, and concentrates the application state within the user agent components. It leverages HTTP and the Uniform Resource Identifier (URI) namespace for all types of applications and allows for independent evolution of clients, servers, and intermediaries” (Ray, Kulchenko, 2002, pg. 238).

Benefits of using PaaS are the same as SaaS, but there are some additional, which are following:

- Multiple sources can be merged to another web services. Supporting SOAP-messages and REST architecture allows applications, based on PaaS technologies, to be easily integrated with different web services.
- Built-in infrastructure simplifies cost savings in security, scalability and failover areas. The cloud computing advantage – processing somewhere allows to easily scale applications, as well as the provider is responsible for availability of systems and data, however there are some pitfalls in security, which are not completely solved.
- Higher-level programming abstractions usage also allows to save costs. Developing process moves to more abstraction level, where simple and ordinary operations are already implemented.

At first sight PaaS offers all services which could be necessary for a development company, but impossibility to change vendor or associated transferring costs are hindrance as well as a risk that provider will go out of business and applications and data will be lost. As an example that happened with Zimki provider in mid-2007 (Velte, Velte & Elsenpeter, 2009, pg. 14)

There are several classifications of systems, associated with PaaS. First of them is on basis of SaaS extensions (Velte, Velte & Elsenpeter, 2009):

- Add-on development facilities: provide SaaS extensions to be customized by developers and users
- Stand-alone environments: are used for general developments and do not depend on any SaaS applications in licensing, technical and financial issues.
- Application delivery-only environments: support hosting services (security, on-demand scalability) without development, debugging or test capabilities.

At the heart of another classification is the level of functions implemented in the lifecycle platforms (Hurwitz et al., 2009):

- Integrated: include all means needed to develop, test and deploy applications without an anchor to any field.
- Anchored: based on business logic at the center of the approach.
- Technologies as a platform: supplementary technology to certain platform.

3.2 Integrated lifecycle platforms

Integrated lifecycle platform is a kind of platforms, which provides users all facilities needed to develop their own applications from operating system to collaborative and version control tools. In general, companies, providing integrated lifecycle platforms include following components:

- An engine: a core that includes logic, needed to operate with provider's environment.
- Development tools: various means, which developers use to create applications.
- Testing tools: special tools, created to test sources and developing systems.
- Third-party tools and services: software and services which supplement existent environment with additional features.

- Databases integration ability: developing applications are able to store information in databases.

That kind of platforms has own middleware and programming interfaces which are offered to customers for developing applications. That abstraction level is probably less functional but provides high-usage tools that satisfy the majority of customers. Therefore tasks that require specific technologies or approach not always could be transferred to cloud.

Google and Microsoft, which are famous for their products during years, understood the advantages of cloud computing technologies and develop it rapidly. Now they offer both facilities to create new applications in clouds (PaaS model) and environment to deploy them directly (IaaS). Relating to integrated lifecycle platforms Google App Engine and Microsoft Azure gained popularity due to benefits for business and IT.

3.2.1 Google App Engine

Google App Engine is a platform that allows creating and running web-applications in Google's infrastructure. These applications are easy to create, maintain and improve in case of traffic or database size increasing. Also developers do not need to think how their application will be supported and maintained, Google will care about it. An application could be published on private domain with Google Apps or used free name on the appspot.com domain. An application could be for general use or only within the company.

Today Google App Engine supports applications written in several programming languages: standard Java technologies, including the Java Virtual Machine (JVM), Java servlets and the Java programming language or other languages, using JVM's interpreter or compiler, e.g. JavaScript and Ruby. In addition Google App Engine provides a special Python runtime environment, including rapid interpreter and the standard Python library.

Another advantage is a payment system, it is very flexible and customers should pay only for what they really use. Also clients do not need to pay for set-up and there are no recurring fees. Using recourses like storage volume and traffic are reasonable evaluated and charged. Applications management is very flexible and allows allocating as much resources for certain application as it really needed, thus it is possible to be in the budget limit. It is free to start using App Engine for applications demanding less than 500 Mb of storage and Central Processing Unit (CPU) resources needed to an effective application, serving 5 million page views monthly. But these limitations could be easily expanded, enabling billing, and fee is charged for using extra resources above free (Google, 2010).

Google App Engine's environment allows creating applications easily, reliable in case of heavy load and storage volume. It contains following functions (Google, 2010):

- Dynamic web serving, with full support for common web technologies
- Persistent storage with queries, sorting and transactions
- Automatic scaling and load balancing
- APIs for authenticating users and sending email using Google Accounts
- A fully featured local development environment that simulates Google App Engine on your computer
- Task queues for performing work outside of the scope of a web request
- Scheduled tasks for triggering events at specified times and regular intervals

An application could be run both in Java and Python environments, which offer standard protocols and technologies for web-applications development.

All customers' applications work in safety environment, providing restricted access to operating system. That kind of restrictions lets Google App Engine to distribute incoming queries to the applications for several servers and start and stop servers in accordance with load and traffic. Testing environment isolates the application in its own safety and reliable environment, independent of equipment, operating system and web-server's location.

Following restrictions are used in Google's reliable test environment (Google, 2010):

- An application can only access other computers on the Internet through the provided Uniform Resource Locator (URL) fetch and email services. Other computers can only connect to the application by making HTTP (or HTTPS) requests on the standard ports.
- An application cannot write to the file system. An app can read files, but only files uploaded with the application code. The app must use the App Engine datastore, memcache or other services for all data that persists between requests.
- Application code only runs in response to a web request, a queued task, or a scheduled task, and must return response data within 30 seconds in any case. A request handler cannot spawn a sub-process or execute code after the response has been sent.

The Java environment.

Using standard Java web-development tools and API an application could be developed in the Java environment. It interacts with the environment by means of Java Servlet standard and can use standard web-application technologies, like Java Server Pages (JSP). The Java environment uses Java 6. Java App Engine Software Development Kit (SDK) supports applications development with both Java 5 and 6. The environment includes Java SE Runtime Environment (JRE) 6 and libraries. Testing environment restrictions are implemented in JVM, so an application can use JVM byte code and libraries within the limits of the testing environment, e.g. the attempt of opening socket or writing a file will invoke an environment exception. Using standard Java API makes it possible to get access to the majority of App Engine services. For storing data App Engine SDK Java has interfaces implementations of the Java Data Objects (JDO) and Java Persistence API (JPA). API JavaMail is used for sending emails by Main App Engine. API HTTP java.net has an access to get data through URL App Engine. In addition for own services App Engine includes low-level API, which realize adapters or could be used directly from the application. Usually to develop web-

application for JVM Java developers use Java and API. Using compatible with JVM compilers and interprets allows creating applications in other languages like JavaScript, Ruby and Scala (Google, 2010).

The Python environment.

Due to the Python App Engine environment it is possible to create application in Python programming language and run them with optimized Python interpreter. App Engine has various API and tools for development Python web-application, including API for rich data modeling, a special web application framework and tools for management and access to applications' data. For development Python web-applications multiple libraries and infrastructures are available, e.g. Django.

The Python environment uses Python 2.5.2, Python 3 is considered to be used in the future. Although the environment uses the standard Python library not all functions are accessible in the testing environment, e.g. the attempt of opening socket or writing a file will invoke an environment exception. For the convenience some standard library modules, which key functions are not supported, are disabled.

Application sources, created for the Python environment should be written in Python exactly. Extensions written in C are not supported. The environment provides powerful Python API for data storing, Google accounts, getting URL and operating with e-mails. App Engine also provides a simple web-application infrastructure – webapp, which simplifies applications creation. With application external libraries could be launched, but they have to be implemented in pure Python and do not require unsupported modules of the standard library (Google, 2010).

The Datastore

App Engine supports a powerful distributed data store service, including the requests and transactions mechanism. A distributed database extension depending on data is the same as extension of distributed web-server's resources at the

expense of traffic. The App Engine data store is not a usual database: data objects (entities) have kinds and properties sets. Using requests it is possible to get entities of certain type, filtered and sorted by properties values. Properties types could be chosen among the supported property values.

Schema is not needed for data storage objects, because data structure is defined in application code. JDO/JPA Java interfaces and Python data storage include functions for applying a structure in applications. An application can get an access to the data store for realizing needed structure part. The data store is coordinated and uses optimistic concurrency control – an entity update occurs in transaction, which executes iteratively several times if other processes try to update the same entity. An application can carry out several operations with data store in one transaction. All these operations will be successful or unsuccessful simultaneously, that provides the data integrity (Google, 2010).

Google accounts

App Engine supports the application integration with Google accounts for users' authentication. New application allows user to sign in with a Google account and access to email and the name, associated with his account. Using this feature makes possible to start using the application immediately without creation new account. Also developers should not create authentication system for their own application.

If an application works under Google Apps, it can use the same functions for organization's members and Google Apps accounts. Users API can notify the application if the current user is the registered application administrator; as a result the administration is simplified.

App Engine services

App Engine provides services that let to perform operations, needed for application management. For these purposes are following API (Roche, Douglas, 2009):

- URL fetch. Applications can get an access to the internet resources, e.g. web-services and other data, using this service. The URL fetch service retrieves web resources in the same way as high performed Google infrastructure retrieves web-pages for other Google products.
- Mail. Applications can send emails by means of App Engine main service, which uses the Google infrastructure.
- Memcache. That service provides high performed memory cache, on the basis of key-value structure. Several application instances get an access to it. Memcache is useful for data, which does not require the constant storing and functions operating with transactions, providing by data store, e.g. temporary data or data, coping from the store to the cache for high speed access.
- Image manipulation. The image service allows an application to work with images, using API it is possible to change size, crop, rotate and flip images in JPEG and PNG formats.

Scheduled tasks and tasks queues

Cron service lets to plan tasks and perform them outside of web requests responding in certain intervals. In addition the application can perform tasks, adding them to a queue by itself. Now that feature is released in the Python environment only and is planned to be implemented in Java soon (Google, 2010).

Development process

The App Engine software development kits (SDKs) for Java and Python include an application on the web-server, simulating the App Engine services on the local computer. Every SDK includes all API and libraries, available in App Engine, besides that the web-server imitates safety testing environment with checks an access to the system resources, which is forbidden in App Engine runtime environment. Each SDK has the tool for adding the application to App Engine. After creating source codes, static and configuration files this tool will ask the email address and password of Google account and after that upload data.

Creating a new release of an application, which is already built on App Engine, developers can upload it as a new version and users work with previous until the new one wouldn't be launched – new release is tested while the old works. The Java SDK runs on every platform with Java 5 and Java 6, available as a zip file. For the Eclipse environment is a special plugin, which creates, tests and uploads App Engine applications. Also the SDK includes command-line tools for starting development server and uploading applications.

The Python SDK is implemented in Python and runs on every platform with Python 2.5, including Windows, Mac OS X and Linux. The SDK is available as a zip file also, for Windows and Mac OS X there are executive files.

The administrative console is a web-based tool for managing applications, working in App Engine. It could be used for creating new applications, domain names configuration, changing the applications working versions, examining an access, logs and browsing an application's data store (Google, 2010).

3.2.2 Microsoft Azure Platform

Azure Services Platform “offers a flexible, familiar environment for developers to create cloud applications and services” (Microsoft, 2010). It is a set of cloud technologies that provide specific services to developers. In figure 3.1 the simple schema of Azure Platform's components' relationship is presented.

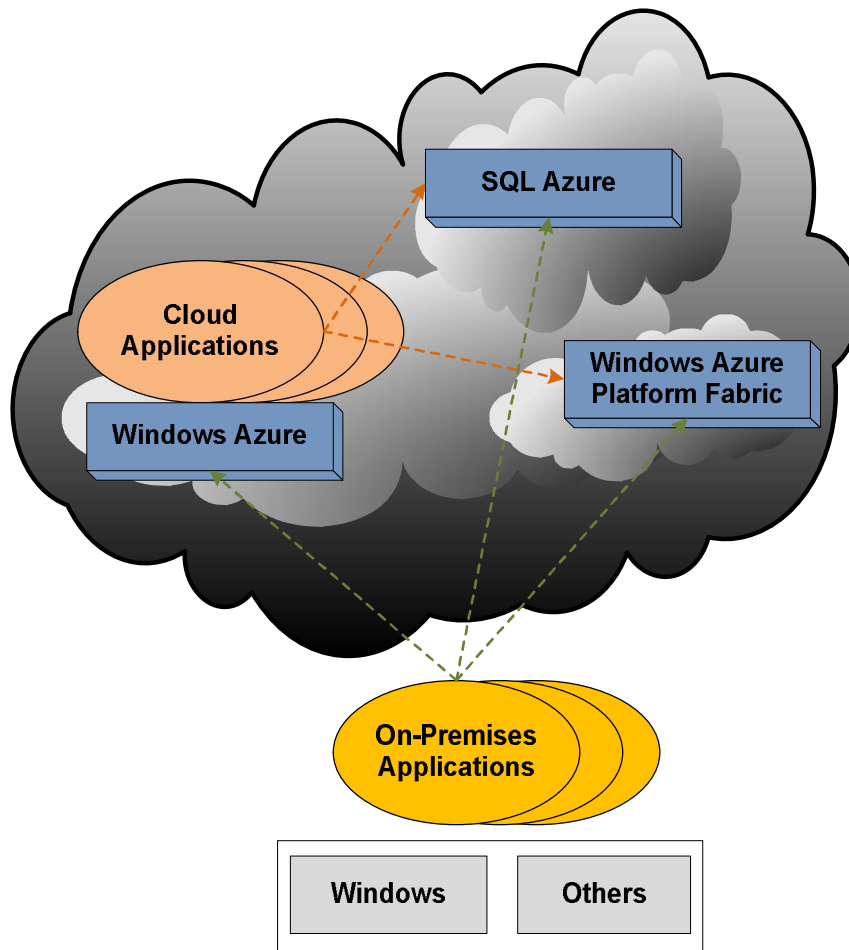


Figure 3.1 – Azure Platform schema (Chappell, 2009).

As is shown it consists of several parts (Chappell, 2009):

- Windows Azure: Provides a Windows-based environment for running applications and storing data on servers in Microsoft data centers.
- SQL Azure: Provides data services in the cloud based on SQL Server.
- Windows Azure platform AppFabric: Provides cloud services for connecting applications running in the cloud or on premises.

Windows Azure

Windows Azure is a platform providing ability for running applications (Compute component) and storing data (Storage service). It includes the environment – servers, which are located in data centers and have special software for managing

and monitoring applications (Fabric). The figure 3.2 shows how an application interacts with components.

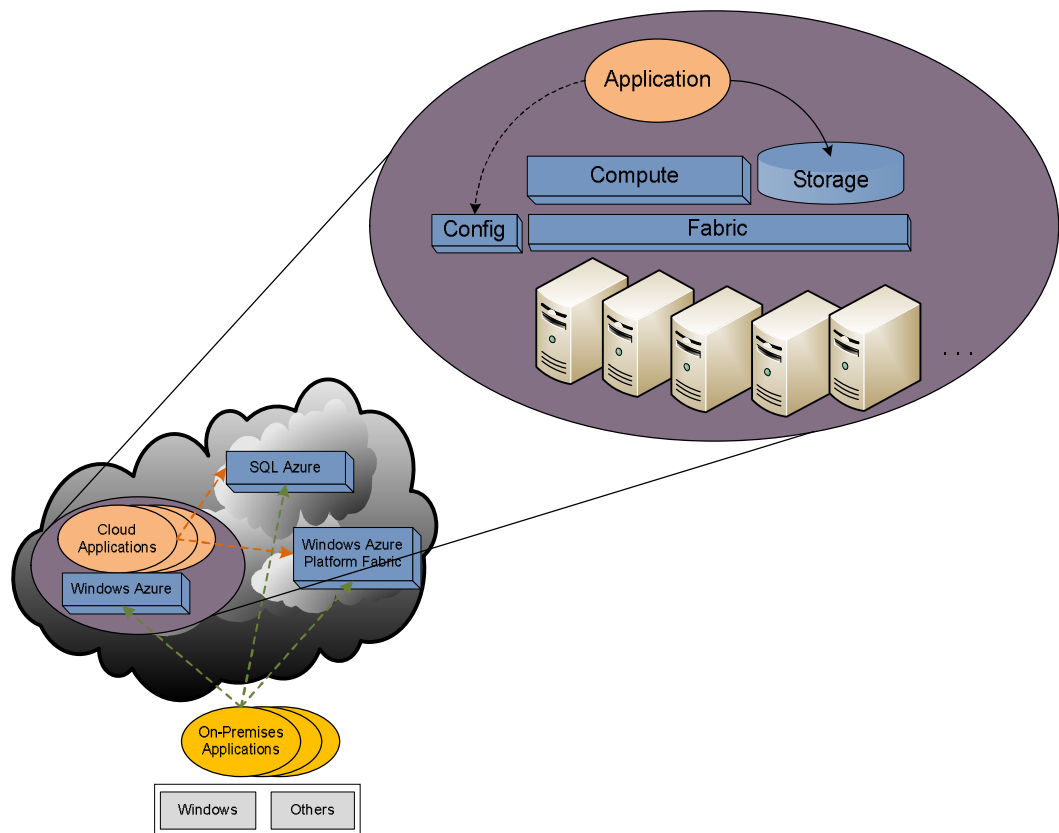


Figure 3.2 – Windows Azure components (Chappell, 2009).

Basically the Windows Azure compute offers Windows technologies for developers and following approaches, like .NET Framework, unmanaged code, some others could be used. C#, Visual Basic, C++, Java, using Visual Studio, are traditional Windows languages and supported by Windows Azure in the first place. At the same time it is possible to create Web applications, using ASP.NET, Windows Communication Foundation (WCF) and PHP.

Each application has multiple instances, which could be a whole or a part of the application's code. An instance runs its own Windows virtual machine (VM). Developers do not care about VMs: creation, maintaining and supplying VM's images are under Windows Azure's control. They should only create an application, using Web roles or Worker role and notify Windows Azure how many instances should be launched. For each instance Windows Azure creates a

VM and runs them automatically. VM has 64-bit Windows Server 2008 operating system with different hardware characteristics (Chappell, 2009):

The implementation of Web role accepts HTTP (HTTPS) requests, using Internet Information Services (IIS) 7. Web role can be realized by means of ASP.NET, WCF or other technologies, compatible with IIS. As it is shown in the figure 3.3 Windows Azure provides the built-in load balancer for requests distribution according to web role instances being the parts of the same application.

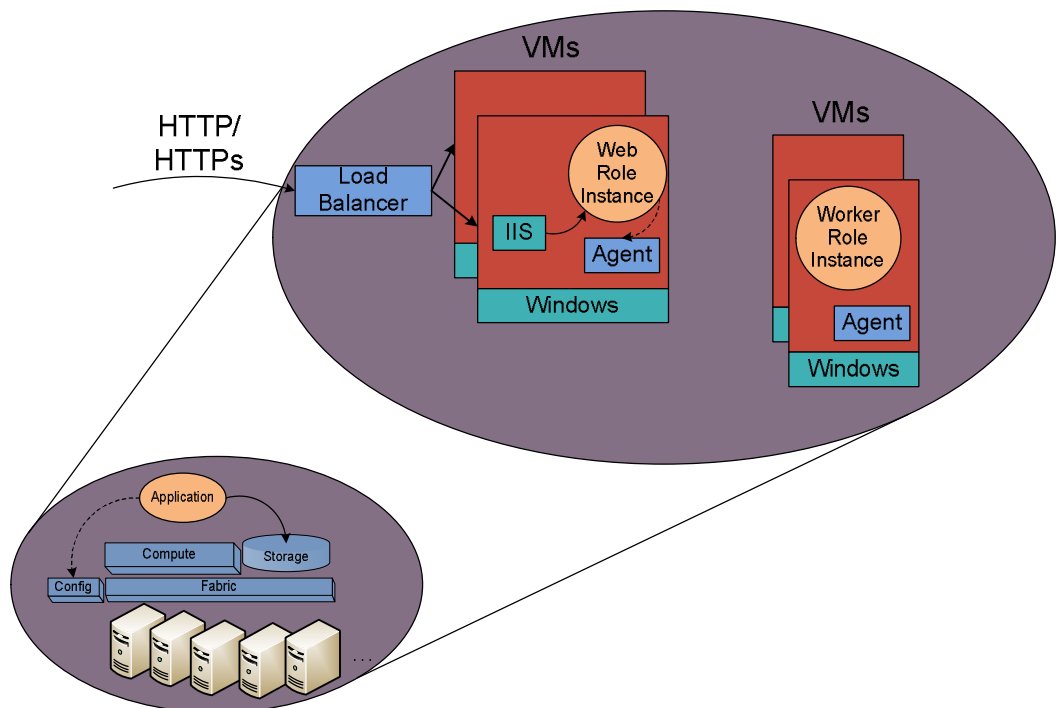


Figure 3.3 – Windows Azure roles (Chappell, 2009).

The worker role is intended to work with non-Web applications; it is not hosted in IIS, but still gets requests from the outside and developers using a Web worker instance can run other Web servers, e.g. Apache. By means of Windows Azure storage queue a Worker role instance communicates with a Web role: a Web role instance puts an item to a queue and a Worker role removes and processes it. Windows Communication Foundation or another technology allows making a direct connection between these roles. To interact with the Windows Azure Fabric structure every VM regardless of a role has a Windows Azure agent.

For managing a load balance an owner should choose how many instances can the application use and according to the number, described in a configuration file. Windows Azure Platform provides API to increase performance, so it is not in automatic mode and the application handles the scalability. But the fabric takes upon itself controlling duties – if a role is failed it starts a new one.

The Web role is stateless, therefore client state should be written somewhere, e.g. to Windows Azure storage or to SQL Azure Database or in a cookie on the client's side. The built-in load balancer also needs client's statelessness, because it is difficult to control that multiple requests from certain user will be served by the same instance.

It is obvious that standard Microsoft technologies are generally used for implementing the applications, even Web and Worker roles are built on them. But to move existing applications, some changes should be done, for example Windows Azure application can run only in user mode. Windows Azure applications development does not vary a lot from creating traditional applications – developers use Microsoft Visual Studio with templates for either roles or combination of them, any Windows programming languages are available. Developers also can install environment (Windows Azure Development Fabric, including Windows Azure storage, agent, etc.) needed to run applications on their own machines. After creating and debugging an application on local machine it is simple to deploy it in the cloud, that option is very useful because some operations are not available in Microsoft Azure, e.g. debug process directly in the cloud provokes writings in a Windows Azure-maintained log. Information about CPU usage, or a bandwidth and storage is also available for developers and could be useful, creating optimized applications.

Using RESTful approach, both Windows Azure and local (on-premises) applications can get an access to Windows Azure Storage Service. But Microsoft SQL Server is not basic data storage. In fact, Windows Azure Storage is not a relational system and operates with non-SQL language. Since this service is destined for applications' support on the basis of Windows Azure, so it provides more scalable and simple storage methods. Consequently it allows to store big

binary large objects (blob), provides queues' creation for interaction between application components and even something like tables with a simple query language. For Windows Azure applications which need a traditional relational storage, Windows Azure Platform provides SQL Azure data base.

Applications' run and storage their data on the Internet have some obvious advantages. For example instead of buying, installing and maintaining own computers a company can entrust it for the internet service provider and customers pay only for computing power, storage and other resources they really use and are not related with servicing big number of servers aimed for peak loads only. If applications are well implemented they can be easily scaled, using huge data centers' resources provided by service provider. But to take these advantages effective management is needed. In Windows Azure each application has a configuration file, which can be modified manually or using a special program. By means of this file an application owner controls different application's aspects, like the number of instances, which should be executed on Windows Azure Platform. Fabric structure monitors for the application to be in necessary state.

Windows Azure provides a portal, available through a web-browser, for creating, configuring and monitoring the application. Windows Live ID is used for access and after authentication a customer should choose whether he creates a hosting account for running applications, a storage account for data storing, or both.

Running applications in a cloud is one of the most important cloud computing features. By means of Windows Azure Microsoft corporation provides a platform for running applications and data storing approach. As interest to cloud computing grows, the creation of more Windows application for this technology is expected.

SQL Azure

One of the most interesting ways in using available through the Internet servers is data handling. The SQL Azure goal is to solve the problem, offering the number of web-services for storing different information and working with it. In the future it is planned that SQL Azure will consist of various opportunities for data,

including reporting, data analysis and so on. First SQL Azure components will be SQL Azure Database and data synchronization facilities Huron.

SQL Azure Database known earlier as SQL Data Service provides the database management system (DBMS). This technology lets local and web applications to store relational and other data types on Microsoft servers and data centers. As with other web technologies, a company pays only for what it really uses, flexible changing the amount of resources (and costs). Cloud database usage changes the nature of capital expenditure: instead of investments to hard disks and software for DBMS operating costs appears.

In contrast to Windows Azure storing service the database SQL Azure is developed on the basis of Microsoft SQL Server, although in initial version it did not provide the traditional relational approach to data. Microsoft decided to change the approach in accordance with customers' desires and now it plans to develop SQL Azure database to support relational data, providing full functionality SQL Server environment in cloud with indexes, views, stored procedures, triggers, etc. An access to these data can be done using ADO.NET and other Windows interfaces to data. Applications that operate today with SQL Server locally will handle with information using SQL Azure database in cloud in the same way. Customers can use local software, e.g. SQL Server Reporting Services, to work with these data in clouds.

Applications which use SQL Azure Database can run on Windows Azure, in corporation computing centers, mobile devices or something else. Wherever applications are launched they access by Tabular Data Stream protocol (TDS), the same protocol as used for local SQL Server database access, thus a SQL Azure Database application can use any existed client library SQL Server, the most significant are ADO.NET services, but ODBC or others also could be used. In figure 3.4 the SQL Azure Database structure and components interaction are presented.

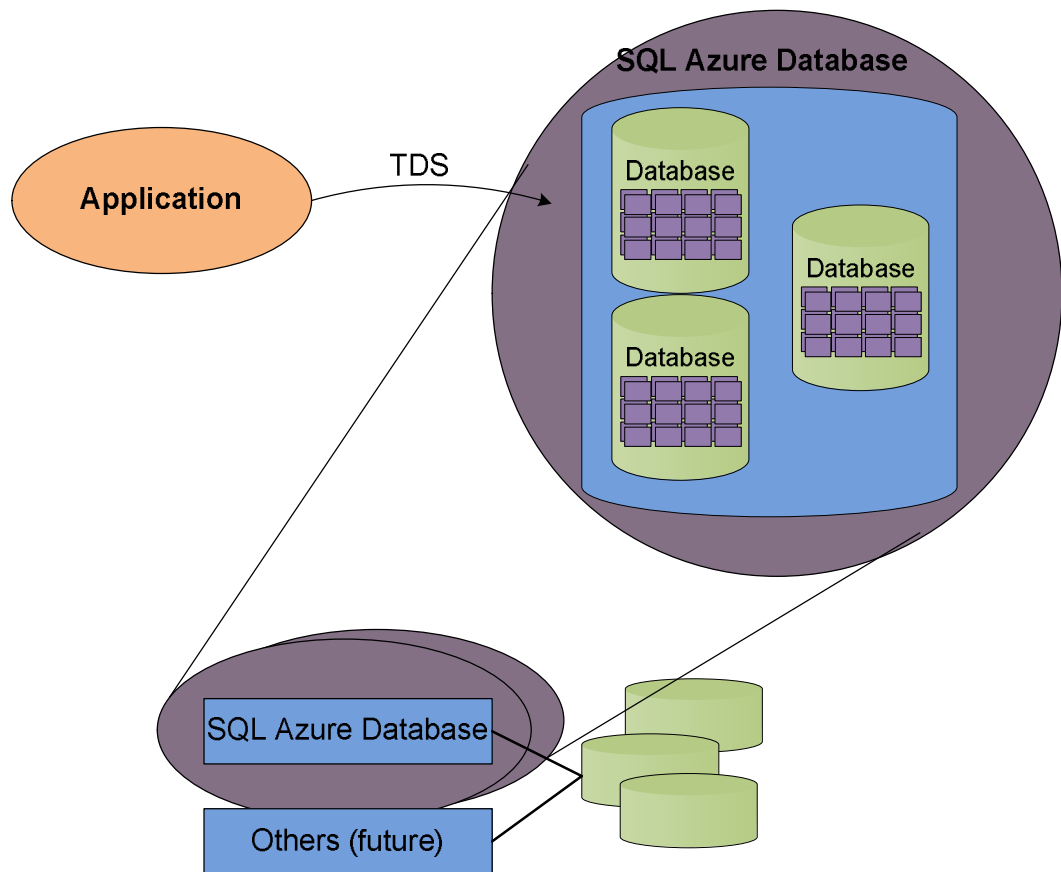


Figure 3.4 – SQL Azure Database.

SQL Azure Database has some differences relating to SQL Server environment, e.g. SQL Common Language Runtime (CLR) is not supported. Also administrative functions are cut through Microsoft administrates the infrastructure itself. Though the environment seems to be usual, it is more reliable than single SQL Server instance. Like in Windows Azure all data in SQL Azure Database is replicated three times and are more consistency. The main task is to provide a familiar, inexpensive and reliable data storage in cases of system or network failures for all applications types.

The maximum SQL Azure Database size is 10 Gb, if an application demands more it can use additional database. As a result if the application uses one database the situation is simple, but in spite of the database provides standard interface the application cannot create a SQL query, which gets an access to all data from all databases. Applications that use several databases should know how that data is split. On the other hand the usage of several databases for applications

with less data amount is justified, because that approach lets to execute parallel queries and performance increases.

Huron

Ideally all data should be in one place, but usually it is not: different companies have copies of the same data, distributed among different databases, remotely located. Synchronization support is difficult, but necessary and Huron cares about it. On the basis of Microsoft Sync Framework and SQL Azure Database it synchronizes data in several databases. Huron will support SQL Server and SQL Server Compact Edition, including SDK for other databases. Anyway Huron first of all synchronizes changes with SQL Azure Database and then for other DBMSs. It provides the graphical interface to define what data in what databases should be synchronized. Also synchronization executes with several masters, it means that changes could be done in any copy and a user should configure this process in case of collision.

Windows Azure applications, data accessibility improvement and synchronization and so on data services in the Internet could be very useful. New technologies in SQL Azure allow organizations to use the Internet for execution more tasks oriented on data handling.

Windows Azure platform AppFabric

Windows Azure platform AppFabric is created for providing services infrastructure on the basis of cloud for local and web applications. It offers functions for solving common infrastructure problems within the creation process of distributed applications. Windows Azure platform AppFabric consists of following components:

- Service Bus
- Access Control

Service Bus: destined for applications to let them provide end points that can be accessible by other applications (local or in the cloud). At first glance could be some problems – how to find these services and how to access them through, for example, firewalls and Network Access Translation (NAT)? Service Bus is meant for getting over these problems. In figure 3.5 the construction of Service Bus is shown.

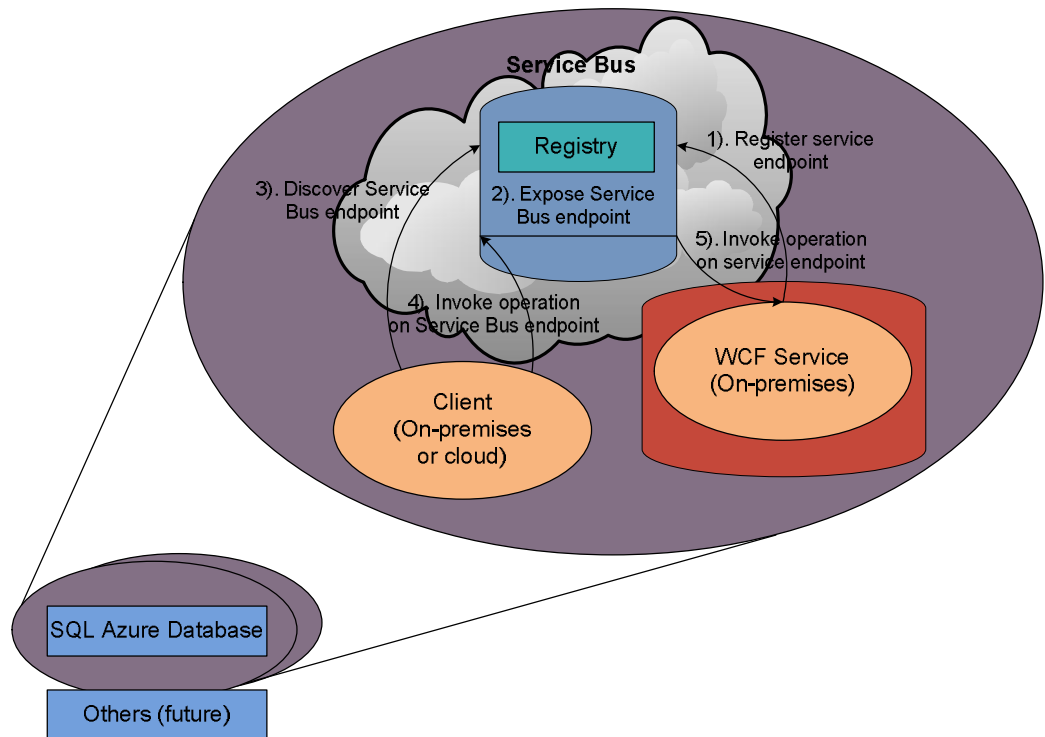


Figure 3.5 – Service Bus’s construction

Firstly WCF service registers service end points within Service Bus; on the second step Service Bus exposes corresponding endpoints. Each end point has a URI, which customers can use to find a service and access it. An application opens a connection to Service Bus for each end point; in the meanwhile Service Bus holds it open, i.e. the problem of translation network addresses and getting through firewalls is solved (traffic is routed to certain application and the connection is initiated inside). On the next step client contacts the Service Bus registry and discovers the endpoint. Using the Atom Publishing Protocol for requests handling Service Bus returns a reference to the endpoint. Then a client

invokes operations on Service Bus end point (step 4) and on service end point (step 5).

It should be noted that a client has a direct connection to the application through Service Bus and communication between them becomes more efficient and security. Even though Service Bus is realized in WCF, clients are not obligatory should be implemented with WCF, e.g. Java implementation is also possible and requests via Transmission Control Protocol (TCP), HTTP or HTTPS are acceptable.

Access Control: RESTful client application is allowed to authenticate and send server application required information. In accordance with a policy server application define what client is allowed or not. The client has a token with identity information to communicate with an application on server, which can contain the organization and application type information. After authentication process is completed, the Access Control server uses this token to define user's rights. At the beginning of it the Access Control Service creates a token for this client, including identity information in human readable view: pairs of names and values. Then the token sends back to the client, this token is signed by the server application's key. After that Server Application verifies signature and determines what client's application can do.

The access control process is quite easy at first sight, but reliable at the same time and provides customers ability to operate with verified data. It also makes application-to-application interactions via REST easier. In the future, is expected, that the Access Control service will include extended set of cloud-based identity services.

3.3 Anchored lifecycle platforms

Anchored lifecycle platforms are another type of platforms, offered by PaaS providers. The main distinction between anchored and integrated platforms is underlying business software. Examples in this PaaS layer deserving consideration due to provided facilities are Salesforce.com and Intuit.

3.3.1 Salesforce.com

Salesforce.com is a company, specialized in SaaS, distributing business software solutions. It provides Customers Relation Management (CRM) products which are delivered over the Internet to the businesses. Force.com is a PaaS product, developed by Salesforce.com to provide facilities for developers to create add-on applications and run them on Salesforce.com infrastructure. Special technologies, such as Apex – Java-like language for the Platform and Visualforce – syntax for creating user interface, based on HTML, AJAX or Flex technologies are used in development process (Salesforce.com, 2010).

By means of Force.com customers develop their own applications without complicated software and use provider's equipment for hosting it. Simplified programming model with high level of abstraction and cloud environment provide facilities to build a business application easier, as a result time and efforts spending for developing process reduce (Narasimhan, 2009). In accordance with PR Newswire Association and IDC (the US research company) "custom application development on Force.com is 4-5 times faster and half the cost of on-premise" (PR Newswire, 2009)

Force.com provides means which can be used to build web-based business applications closely integrated with customers' processes and includes facilities, such as databases, secure infrastructure, workflows, user interface, etc. It has some distinguishing characteristics, for example in the center of all applications is a database (Babcock, 2009). It also has wizards to create databases, with steps to build user's objects, fields and relationships (PR Newswire, 2010). Also the platform generates a user interface, which could be customized by web page

editor – users drag-and-drop objects and modify templates. HTML, JavaScript, Cascade Style Sheets (CSS) are available for developers to create their own user interface (Salesforce.com, 2010).

By means of visual process management tools the process of modeling business processes can be configured via the user interface. After building a workflow, it is available immediately in the system. Interactive wizard allows automate configuring approvals, assigning tasks, alerts and messages sending processes (Salesforce.com, 2010).

With complementary tools and features, the process of creation web sites within the Force.com platform has changed. In addition integration with applications inside a company allows to adapt systems to customer's need and characteristics. However it is still possible to create application with standard web technologies, like HTML, Flash and JavaScript and then configure URL or RSS (Hurwitz, 2009).

Other features, like real-time mobile deployment, integrated content library, real-time analytics, granular security and sharing, the set of existent applications (Salesforce.com, 2010) extend the Force.com facilities. It makes possible to find a particular application for certain requirements or develop own system, based on provided capabilities and extend it with additional functions.

Special language Apex, created by Force.com for developing applications allows to script interactions with other features including user interface (Mathew, Spraeetz, 2009). That language is close to Java and it is possible to write applications in a web browser or using a plug-in to Eclipse.

For user interface development there are several tools: Page Layouts and Visualforce. Page Layouts provides the environment to create a user interface on the basis of data models, using a principle WYSIWYG (“What You See Is What You Get”). It is also possible to make user interfaces without development efforts. Visualforce has facilities to create custom user interface, using special

components – XML markup tags, which format data after returning them from Apex code (Ouellette, 2009).

Applications that planned to be implemented on Force.com should have structured data inherently with strict sharing and security rules. In addition the platform can be easily integrated with other cloud technologies to build high-performance applications (Ouellette, 2009).

3.3.2 Intuit

Intuit is well-known of its product for small and medium businesses called QuickBooks – the application for everyday financial operations. Intuit created a PaaS model to allow developers build their own applications. Federated Applications is a platform providing that facilities. As the company declares it is possible to write code in any programming language, use any databases and other cloud resources and after that it is easy to connect them to the Intuit Partner Platform using configuration files and XML-based interfaces.

Intuit provides services for user interface, billing, account management, etc. and they can be used by developers to create their own systems instead of creating them from scratch. All these services are available and integrated through the Intuit Workplace – a portal, where customers can find, try and buy needed facilities. It also provides automatic integration with the most popular product QuickBooks.

Interaction between customers, developers and Intuit is presented in figure 3.6 (Intuit, 2010)

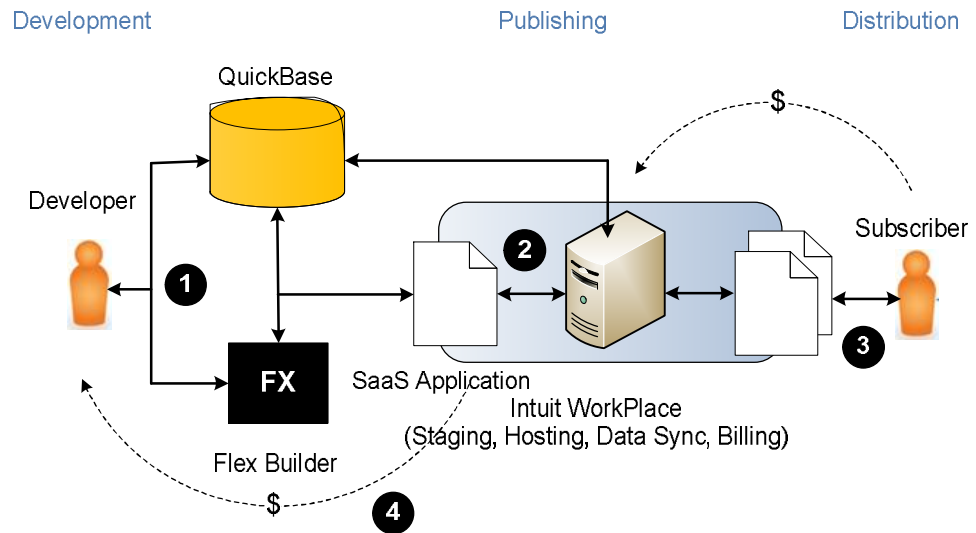


Figure 3.6 – The process of creating and publishing application on the basis of the Intuit platform (Intuit, 2010).

Firstly developers set up a QuickBase database, and using Adobe Flex Builder 3.0 and special IPP SDK develop the application. Then by means of Intuit Workplace create the development environment and publish the application. Customers also use Intuit Workplace to subscribe to applications. On the next step the Workplace’s agent handles the integration process with subscriber’s QuickBooks data: changes in the application are reflected to QuickBooks product and conversely. On the fourth step Intuit cares about billing and developers payments according to a pricing model, which were determined during uploading the application.

The Intuit Platform consists of following components (Intuit, 2010):

- QuickBase – provides Web infrastructure for applications oriented to small businesses. Essentially it is a collaboration platform, including a support for multi-tenancy and a database with team workflow, communications and task management services.
- Workspace – a portal environment, which can be integrated with QuickBooks or used as a stand-alone environment and provides facilities

to observe subscriptions and revenue, in addition allows to control how an application is shown for end-users.

- Federated Applications model – a configuration service, which integrates existing code with the Intuit resources, consists of four methods:
 - Data integration – cloud data schema, describing the relationship between data elements, is used by setting APIs that make data synchronization possible.
 - Login integration – presented by Federated Identity Web API and enables using Intuit Workplace login credentials to get an access to the needed application.
 - User management and permissions integration – realized Web API that has special functionality to operate with a process, like adding more users.
 - Navigation-based integration – common integration is possible, by means of the Intuit Workplace toolbar.

3.4 Technologies as a platform

Some cloud platforms' providers do not offer corresponding tools which are necessary for convenient management of all facilities. For some PaaS types there are technologies which supplement the platform. Different tasks, like testing, service management, integration and configuration, migration from local applications to the cloud can be carried out by outside agency. That advanced tools and technologies are connected to PaaS and they are combined to a special type – technologies as a platform. Following fields which supplement the existing platforms are presented by third parties solutions (Hurwitz et al., 2009).

Testing. There are examples when some platforms do not have their own testing environment, although testing process is very important to make the high-quality application satisfying all requirements. Companies, such as HP, IBM and less known like iTKO and SOASTA provide this service; integrating with a platform they allow testing applications before deploying them to a cloud. Even some vendors have testing tools, relating to the particular platform; customers need the environment which can serve their own code and integration issues.

Service management. The majority of companies, operating in the cloud market have tools to manage applications, created for their platforms, but some of them are not on the necessary level, therefore companies, like RightScale are useful. RightSacle provides the management platform for Amazon.com. HP, IBM, CA, etc. also go ahead in this field with their solutions.

Integration and configuration. Solutions, solving problems of customization for end-user needs, are actual. For example WaveMarket provides a Java cloud-development platform simplifying customization and expansion of Web applications from a browser and makes it possible to integrate legacy systems with a cloud platform.

4 A SCENARIO OF MIGRATION TO CLOUD TECHNOLOGIES

Defining a scenario of migration to cloud technologies it is necessary to find out how to get the sufficient benefits from using cloud platforms, when it is reasonable to migrate, what assets can be transferred to the cloud and what steps should be applied to prevent possible problems. For these reasons a model of typical software development company is developed and on the basis of the model a scenario is described.

It is worthy of note that not all companies will benefit applying offering methods and there are some characteristics which increase the probability of success (Miller, 2008).

Firstly, companies, specialized on web-based applications besides enhanced development process will gain almost unlimited possibilities in computing, storage resources and scalability.

Secondly, distributed teams, using the PaaS approach are able to collaborate more efficiently. Shared access to documents and versioning will help to make clear interaction scheme and as a result save time and money. Before cloud computing people had to send files and edit them sequentially. Now the work is simultaneous. Changes, made by one person, immediately reflect to what others see on their screens.

Thirdly, individual developers or managers access to last versions of a document or code sources in any place – from a workplace, at home, using personal laptop, in a business trip from guest computer, wherever an Internet connection is.

Fourthly, hosting the application in the cloud allows to make changes automatically and after acceptance, they become available to users immediately. Also developing web-applications, programmers do not care about compatibility with different systems and environments (Mac, Linux and Windows), because all is available through a web-browser. As well as using SaaS tools for modeling or

desktop applications they do not have to install and use customers' software to be compatible with them (e.g. the situation, when Microsoft Office 2003 cannot open the formats developed in Microsoft Office 2007).

If these features are important for a company and it develops web-based applications, using PaaS technologies can improve their processes, but that is not compulsory. Here the question of company's management qualification arises. As all new technologies and innovations PaaS adoption requires the appropriate management, who are ready to plan and implement the migration process correctly and then carry out tasks on the basis of PaaS. Well organized process will let a company benefit from using this method (Yapp, 2009).

As Linthicum mentioned in (Linthicum, 2009) there are some features when migration to cloud is reasonable (Linthicum, 2009, pgs. 33-34):

- The processes, applications, and data are largely independent. If they are tightly coupled they will not operate on remote platform.
- The points of integration are well defined. Then they are easy to integrate with applications back in the enterprise.
- A lower level of security will work just fine. Security is an obstacle now in adoption cloud computing.
- The core internal enterprise architecture is healthy. Then it is easy for cloud computing to become part of that architecture.
- The Web is the desired platform. Browser-based application is the solution for user interface.
- Cost is an issue. To build and deploy applications on the cheap is possible with cloud platforms.
- The applications are new. To port existing applications is much more difficult than to create a new application.

As it was described above in [part 2.4](#) PaaS providers offer services that can be used by software development companies, such as application design, development, testing, deployment, hosting, team collaboration, web service

integration, database integration, scalability, storage, state management and versioning. It is possible for them to replace own resources with PaaS solutions.

Software development companies have some specific characteristic, distinguishing themselves from other businesses, e.g. the main – development process, is impossible without IT technologies and there are some tools' types and means, which are obligatory. Particular companies, in need of specific products exist and they require special solutions relating to customer's demands, but they are not under consideration. General organizations, specializing in software development with their needs are examined below.

Tightly integration between PaaS and other cloud technologies, like SaaS (e.g. for using desktop applications in a cloud, like Google Docs) and IaaS (any platform is always based on certain infrastructure) allows organizations to replace the majority of existing resources, by offered in the cloud.

There are several stages intrinsic to the process: planning (requirements analysis), design, implementation, testing, documenting, deployment and maintenance (technical support) (IEEE and EIA, 1998). To create a product on all these stages different facilities are needed. Following capabilities for various participant roles, involved in the process, are required:

- Modeling tools: tools for designing systems and applications.
- Development environment: a special application for software development.
- Database support: facilities to operate with databases.
- Desktop applications: applications intended to work on documents, e.g. office suites.
- Versioning system: a system for the management of files' changes.
- Bug tracking system: an environment for tracking reported software bugs in the work.

Let's consider a typical structure of software development company where several teams work together on different projects.

Before migration to a cloud

The company provides developing, hosting and maintaining services for web applications. In general a company has certain infrastructure to manage business processes. As an example let the company has some branches in three cities: Lappeenranta, Turku and Tampere which are connected remotely to the main office in Helsinki, where the majority of resources is located. The architecture can be as it presented in figure 4.1. Each branch performs projects independently, but some projects need more resources, than can one branch offer.

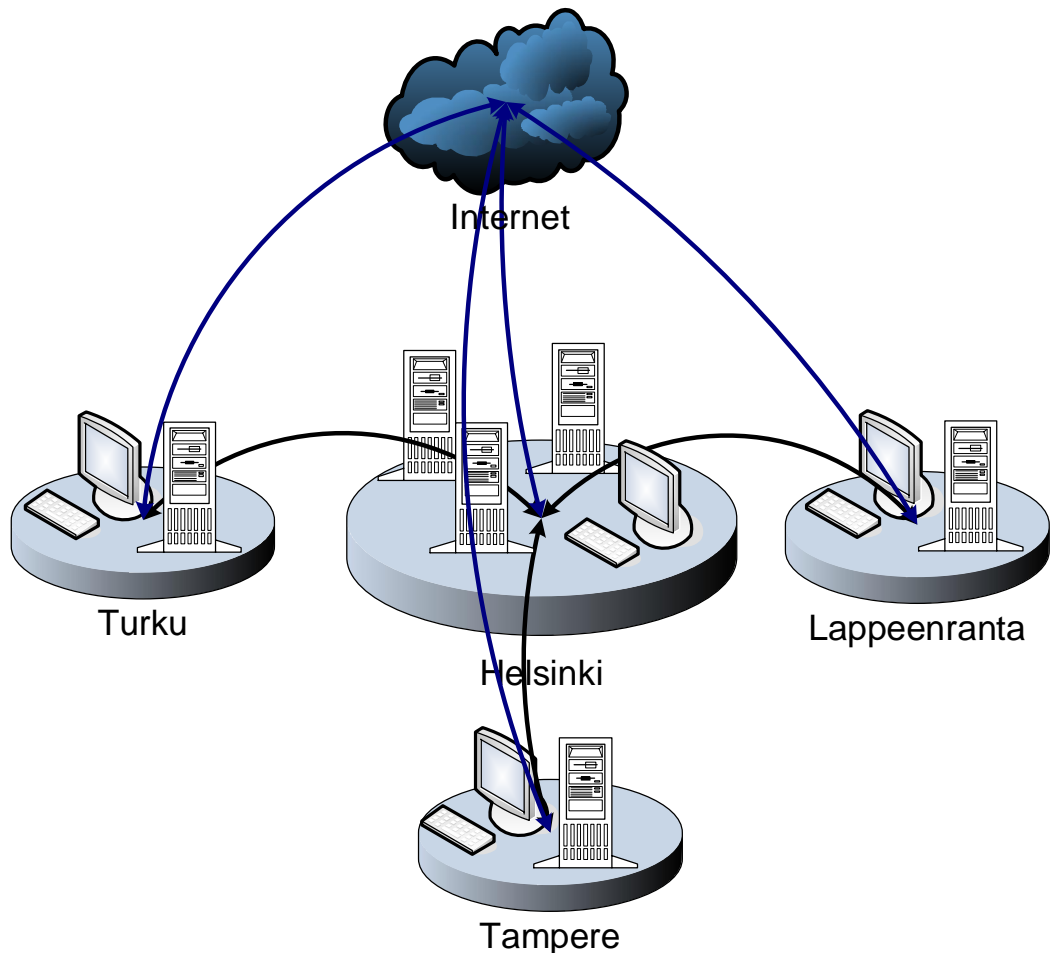


Figure 4.1 – The software development company's structure before the migration to cloud technologies.

Hardware

Physically the remote office consists of desktops' set for developers, testers, managers and few servers, dedicated to serve developing projects. Desktops are up-to-date and capable to handle with tasks, specified for certain process. Besides computers the branch includes network communication hardware to access remote resources that provides an Internet connection and a link to the main office. Bandwidth is sufficient to all operations within the company. As for main office it has both desktops for development and management processes and a data center with servers for business and needs, relating to own infrastructure (like servers for group policies). Security is provided by software and hardware capabilities.

Software

A workstation, according to a person role involved in definite process, is equipped by the set of following software:

- Operating system
- Development environment
- Modeling tools
- Testing tools
- Desktop applications
- Client application for a version control system
- Communication tools
- Client for bug tracking system (usually a web-browser)

Servers' side in the main office and branches consists of undermentioned facilities:

- Server operating system
- Web server
- Data bases
- Versioning system
- Bug tracking system

- Backup system
- Communication system

People

There are system administrators who are needed to maintain servers and workstations and care about system's state, solve occurring problems and support users. In the main office the data center is located and it requires highly experienced administration team to perform all tasks needed for doing business. All administrators report to the chief information officer who renders strategic decisions for the whole company.

Resources that company can transfer into a cloud depend on its characteristics, but in general all of them that were mentioned above potentially can be executed in the cloud. However some limitations exist and managers should discover all possibilities beforehand.

If a company decided to move there are several ways to make it. Firstly, transfer in sequence new projects to cloud technologies, abandoning from own resources and replacing them. In that case two approaches (traditional and cloud) are used simultaneously some time. Another method requires the migration process on the instance that should be well planed and organized. Migration tools are necessary for it and after migration is successful a data center can be deleted before it becomes out-of-date. Anyway it is better to try a pilot project within a branch, ask certain users to test new environment and then apply it to all company.

After migration to a cloud

After migration almost all company's resources can be located in a cloud and leased. The ideal situation when all assets are transferred from own to provider's can be as it is presented in figure 4.2

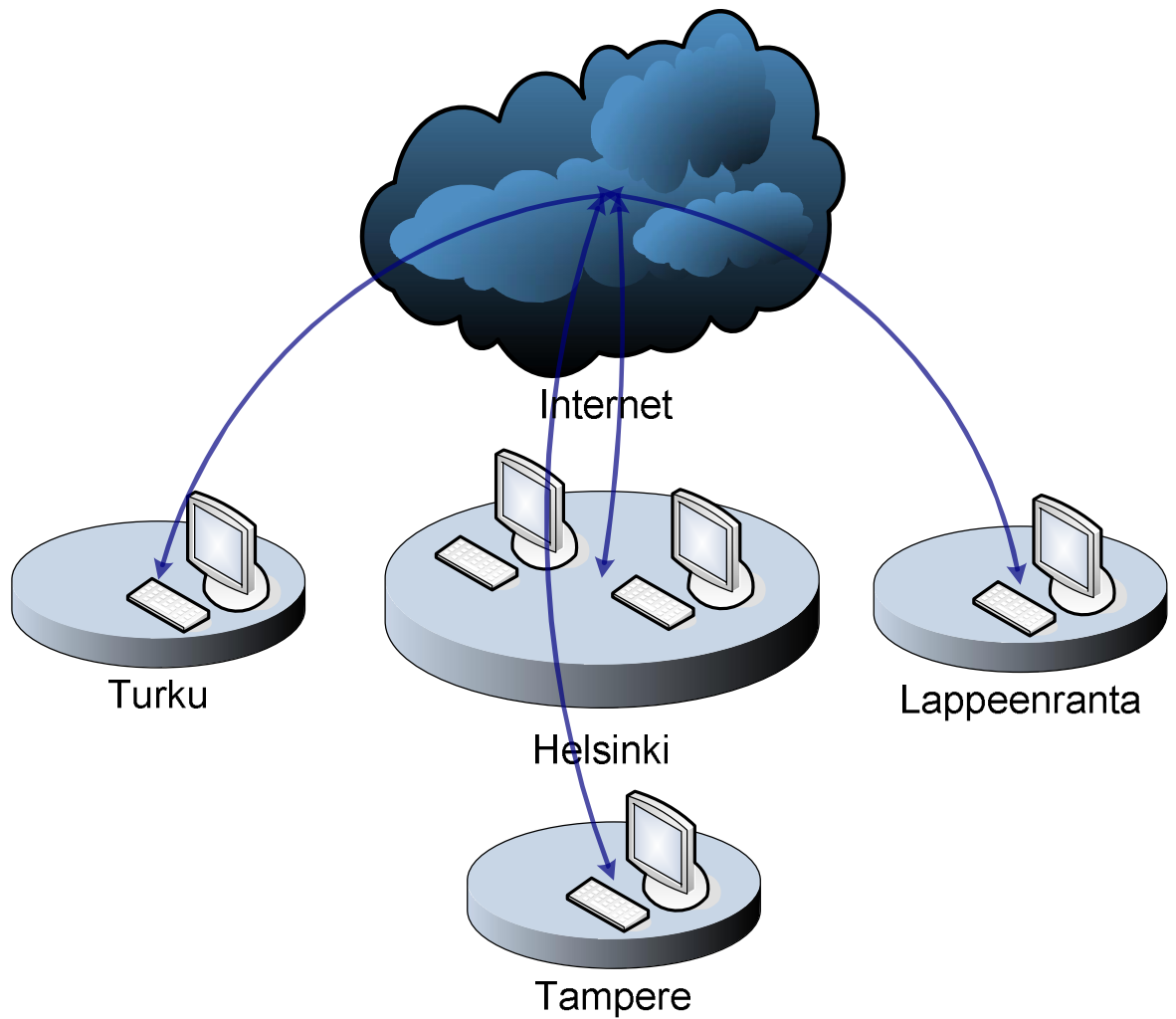


Figure 4.2 – The software development company’s structure after the migration to cloud technologies.

It is enough to have workstations required to convenient work with the Internet through a web browser. That kind of desktops are less demanding of hardware than it is necessary for tasks executed directly on workstations (e.g. compiling, testing, modeling, etc.). As well as servers’ hardware which is rapidly becomes out-of-date requires periodical upgrade process.

If overview the situation after the migration from software viewpoint, the company does not need to invest lump sum to solutions, because it will lease software, the whole environments – platforms running on infrastructure also. Administrative tasks such as installing, maintaining and upgrading become

provider's duties. As a result it also influences on company's staff organization and probably to employ all people who used to be is not reasonable now. Since the quantity of tasks has reduced, part time workers can administrate branches or even on-call maintenance outsourcing company can be involved.

The migration process from traditional to cloud technologies is sophisticated and should be well planned. The first step is estimation if cloud technologies are appropriate for certain software company or its specific does not allow to refuse legacy environment. Then according to Linthicum's the reasonability to migrate should be assessed. The question of qualified people, who are able to face with the cloud issues, plan and implement the process, arose. Discovering what facilities cloud solutions provide is significant to evaluation of possibility to replace in-house resources to cloud.

5 SECURITY AND PRIVACY

Security and privacy in a cloud are complex issues and should be well examined to understand their impact to PaaS adoption. There are different points of view to this problem. As Binnings says, service providers declare that robustness of their products is more, than enterprises can offer (Binnings, 2009). But at the same time there are a lot of customers who are afraid to entrust their data to external company, especially if data is critical, and it is not only psychological barrier, which should be overcome.

5.1 Security

For companies migration to cloud technologies is not a sudden action and usually virtualization is an intermediate step. After estimation all advantages of virtualization an organization decides to replace own resources to provider's. But it is fraught with various pitfalls. The migration process from physical to virtual servers and from virtual to cloud environments is not a simple task and for specialist, who understand security in terms of traditional technologies, such as perimeter defense, network segmentation, intrusions detection and avoidance, antivirus software, etc. can cause difficulties (Rittinghouse, Ransome, 2009).

A traditional data center is a big set of servers, located on the same environment for increasing effectiveness and security. Data center's security can be discovered from two positions: physical and network. Physical is simpler, because it includes the restriction of administrators physical access to servers and network infrastructure. From network security, it means the construction of reliable perimeter defense, including a firewall, intrusion defense. Besides the main function – protection, the firewall provides segmentation of the internal network to different security levels (servers, available from the Internet, servers, available only from the internal network, etc.) (Mather, Kumaraswamy & Latif, 2009).

After migration to virtual and cloud servers and platforms, traditional methods of providing security are limited. The term “perimeter” does not exist more, segmentation is not available and even hardware security equipment cannot be

used, because more traffic move within virtual machines and platforms (Rittinghouse, Ransome, 2009).

Another significant point is updating process, because if a provider creates an update or patch for a platform it is critically important that changes will not influence on customers' applications and systems. For this reason tracing dependences between platform, environment and the customer's security requirements is a necessary task and only work in close collaboration among customers and providers lets secure data and prevent loses. To make collaboration more efficiently, the cloud provider can accommodate customer's security policies.

According to classification, given by Cloud Security Alliance, there are some top-threats related to PaaS services (Cloud Security Alliance, 2010):

Abuse and Nefarious Use of Cloud Computing. Providing almost unlimited compute, network and storage with some times simplified barriers to entry (valid card is enough to create an account and start using it immediately, that favors anonymity) this kind of platforms are subject to abuse usage. Intruders are able to take advantages from resources and be unpunished at the same time. Following activities, like password cracking, distributed denial of service (DDoS), hosting malicious data, spamming, etc. become easy implemented. What is needed – just write program code, that can be executed by provided platform and host it somewhere in the cloud. There are some examples when PaaS and IaaS were used for hacker's attacks and malpractices.

Insecure Interfaces and APIs. Interfaces and APIs are created for interaction with services, therefore the security of cloud services depends on the security of provided APIs. They should be designed as well to protect services from malicious attempts to overreach provider's policies. Other APIs are usually built on the basis of core functionality and dependencies between them are significant to confidentiality, integrity and availability to company's resources.

Malicious Insiders. The cloud providers do not have any special requirements to employees and intruder can easily be in provider's staff. What access certain employee has and how they are controlled is not available information for customers who explore abilities to use particular company as a service provider. To prevent this type of threats is not a simple task, but accurate requirements for human resources applicable to provider's staff can be available for customers. Increased transparency of information security and management will allow customers to understand what company does for detecting and defending insiders and as a result the data, entrusted to provider becomes more reliable and secured.

Data loss or leakage. Data compromise is another significant issue of PaaS security in a cloud. Different ways exist to compromise it: records' deletion and alteration, encoding key loss, unauthorized parties should not gain the access to sensitive data. That type of threats is more important in the cloud due to cloud environment architectural characteristics. Associated losses with vendor's brand, reputation and customer's trust besides competitive and financial problems can bring to breach of agreement and legal outcomes.

Account or Service Hijacking. That kind of threats consists of following attack methods: phishing, fraud, software vulnerabilities exploitation and others. Access gaining to credentials opens up opportunities to eavesdrop activities, transactions, manipulate data, substitute needed information for falsified. Compromised account or service becomes a new base for new attacks and customer's reputation suffers. Breaches result to services' confidentiality, integrity and availability, which can bring to litigations in turn.

Unknown Risk Profile. Information related to sharing infrastructure, network intrusion, redirection attempts is important for company's security estimation. The compliance between declared security features and internal security procedures, configuration hardening, auditing and logging as well as information in case of a security incident, disclosed by vendor face customers with unknown risks, which they do not examine before.

All these problems constitute threats to applications and data, located in clouds. Impossibility to gain the physical access to resources and insufficient control, obscurity, probability to compromise and loss data, human factor, that cannot be traced, create additional barriers for customers to easily adopt cloud technologies and start using them. At the same time privacy aspects which are weak points in legislation in different countries and jurisdictions are obstacles as well.

5.2 Privacy

Privacy is a key business risk and is another pitfall in cloud adoption, because industries and government regulations still impose constraints on data and information even if they are located in the cloud. If, for example, data is processed or stored in the country's territory the government has legal rights to access it and service providers even should not notify customers about this accident (Regulation of Investigatory Powers Act, 2000). Also is possible that data can be available for secondary uses on legal grounds, e.g. service providers can gain an access to customers' data for the purpose of advertisements. And that is common practice for companies to revenue from providing data to third party.

According to developing applications' specific after implementation an application and data are hosted on provider's servers and are not under developers' control. At the same time they are still responsible to customers for quality and reliability.

There are different types of information that can be stored in clouds, for example, customer information and business data. Relating to privacy issues all of them should be carefully protected. Customer information consists of user data and personal data (Rittinghouse J., 2009). User data includes any data that is collected from a customer directly through the application's user interface, data gathered indirectly (e.g. documents' metadata), customer's behavior (e.g. history), data about customer's system (e.g. system configuration). Personal data, such as contact information (e.g. phone number), identification forms (e.g. passport), demographic information (e.g. ethnicity), occupation information (e.g. job title),

health care information (e.g. genetic information) is more important, because it can be used to identify a single individual.

The cloud provider should provide reliable mechanism to protect data, hosted on its resources. That kind of data privacy has solutions, while legislation question is more complex.

It is obvious, that laws in different countries are not the same and have characteristics appropriated for certain region. Some of them depend on the location of company, some on the data center physical location, another on the data subjects' location. According to data privacy laws can vary, the US Federal Rules of Civil Procedure can breach the EU Directive.

In Europe generally personal-identified data can be used but there are restrictions, e.g. that the data subject should be aware about processing and purposes. As regards to sensitive data, such as health and financial data, it requires detailed consent to process that kind of information. At the same time data subjects can refuse using their data for marketing purposes at all or limit location where it can be stored and impose constraints on security. Moreover there are some another limitations and breaches in law, when services do not guarantee that servers where they are running are secured up to snuff, as well as cross-border data transfers can be forbidden if stored data is located in countries with weak privacy laws. In that cases cloud computing can be prohibited to use, e.g. according to the Federal Data Protection Act in Germany storing application and data outside national borders is impossible (Klems, Nimis & Tai, 2009).

Principles of estimation the privacy risks.

There are some principles that allow to estimate fully the privacy risks (Mather, Kumaraswamy & Latif, 2009, pgs. 151-154):

Collection Limitation Principle. It consists of the statement that collection of personal data should be limited to the minimum needed for certain purpose. Such data should be obtained legally and with the data subject's consent. Lack of

specifics on data collection results to appearance of misunderstandings between service provider and organization in the future. There are different conflicting laws and regulations, but universally adopted privacy standard does not exist. An attitude to privacy can vary from customer's and provider's point of view and it is important to discuss all possible issues beforehand. The company can perceive what they do as "the right thing", while it violates the local laws where server provider operates. As well as service level agreements (SLAs) should be defined to make jural relation, where the agreement on the quality level described and in case of precedent has a right to demand compensation.

Use Limitation Principle. This principle prevents data from disclosure, becoming available or using for purposes that the data subject does not allow or violate the law. The problem is that cloud approach places all information in a single location; therefore data governance is needed to control new purposes. For example if a company creates a centralized database new applications can combine the data, but for new purposes the data subject did not withhold his consent. Unexpected usage of the data can be a problem; e.g. a government can ask the provider to report on users' information, their behavior and activities. In addition the provider can be obliged to inform government bodies about users' activities, which can be a trouble for both users and the company.

Security Principle. Specifies that customers' information should be properly protected to avoid risks of losses, modification and disclosure.

Retention and Destruction Principle. This principle specifies that data needed to perform a task should not be retained more than it is required. Data should be destroyed immediately after that period. It is also a rather complex task, because after deleting a file it does not mean that it is impossible to retrieve. After deletion blocks should be overwritten and for example due the U.S. federal government's guidelines at least seven times. Encryption process is also important for data privacy and as a result improves destruction. So it is important to create destruction policies within service provider and describe it to customers, because their data is stored and they are interested to protect it from all possible threats.

Transfer principle. Data transferring should not be to countries where the level of privacy protection isn't the same as the company's location that collected data. Information in cloud is dynamic and located on infrastructure that is shared between different organizations. The possibility to aggregate services brings to situation when sensitive data can leave the service provider's boundaries. That possibility to gain the access by third parties should be prevented. The fact that customers do not know where their data is computing at certain time means that it can be everywhere, but as the European laws require a company have to be informed where the personal data is at every time. The U.S. Safe Harbor Program was created to make compliance with EU requirement, but if the data is processed outside of Europe and the U.S. both of laws is ineffective. Restricting data inside certain area will not solve the problem and reduce flexibility and efficiency only.

Accountability Principle. Specifies that companies are responsible for personal information which they control. Also certain individual or group of people should be designated in there to be accountable for the compliance with other principles. Attaching policies to data and verifying process that they are followed by others who use, store and share the data should be done regardless of jurisdiction where it is processing.

All these principles allow to improve customers' privacy and regulate what activities should be done to achieve best results. Security and privacy issues are the most significant because storing data outside the company is fraught with danger to lose control over them.

During the developing and further deployment processes the management of company should be worried about data security issues. Almost all security aspects are vendor's duties, but before blindly entrust applications and data to the service provider customers should find out how reliable the provider is and what it guarantees. None of following issues: passwords policies (e.g. assignment, protection), the possibility, that data can be accessed by third parties (providers interact with other organizations and that situation can occur), claimed downtime probability, exception monitoring systems, etc. can be omitted. The customer, using provider's resources is responsible for the application quality to users, who

directly exploit it. Independent examination of hosting company, which is used by service provider, will allow the potential customer to get all needed information and to make a right decision (Binnings, 2009).

In addition legislation in different countries mismatches with the cloud approach and there is no definite solution of this problem. The enormous potential of clouds is restricted of governments' laws and they should catch up with the technology.

6 ECONOMIC EVALUATION

Technically PaaS provides some advantages which are essential for software development companies as compared with the traditional approach, but at the same time economical benefits of using cloud platforms are far more considerable than increased computer power and unlimited storage capacity. On the one hand, costs and time savings with pay-as-you-go system, effectiveness of usage and the increased level of risks on the other hand, face customers to carefully weight all pros and cons. These issues are examined below as well as strategies that can be applied to develop a company.

6.1 Costs

From the economic side cloud computing approach has some advantages, comparing favorably with the traditional. Firstly, payment system, so-called “pay as you go”, allows companies to decrease significantly the fixed (capitalizable) costs. In the figure 6.1 the expenses, that are incurred by the companies in case of using the traditional technologies and when ‘moving to a cloud’ are presented. For both situations costs are presented by two different types: variable costs (VC) and fixed costs (FC).

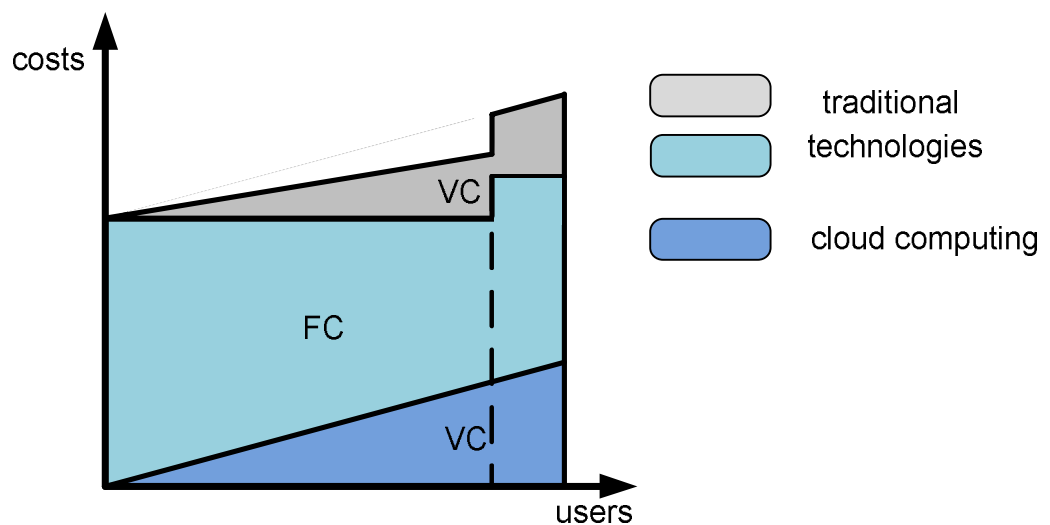


Figure 6.1 – Costs distribution among traditional and cloud technologies.

In the first variant (traditional technologies), in order to set up a company there should be some volume of investments that will be used for amortization of fixed costs, buying servers and software for them, technical support. In case of increasing the number of users variable costs will be added to fixed costs proportionally. At that time when a critical number of users is achieved, figured as a broken line, scalability works will be needed. That requires time, when the users' number will not increase, but can decrease on the contrary.

There is a completely different situation using the cloud computing approach. When starting the company it is not need to buy any server equipment to create infrastructure. Customers just pay more, according to the service agreement, once the number of users is increased.

In the paragraph [4 \(Migration\)](#), software development resources were considered from three sides: hardware, software and people. And their components were examined. In that part it is sensibly to take a look on these resources from economical point of view. The migration to new technologies has for an object to upgrade existent business processes to make them more effective: bearing new profit or reducing costs and risks.

Equipment.

In general, using cloud platforms fixed costs are not equaled to zero, because anyway there are some cases, when investments beforehand should be done. Comparing to the traditional technologies they are negligibly small as for values of up-to-date data centers. For example to make a company connected to the Internet requires funds, and that funds do not depend on the number of users, consuming it. In that field only costs, associated with IT are examined, because if consider expenses in general the company faced with fixed costs in any case (e.g. office premises and equipment purchasing). As concerns desktops for developers, testers, managers they can be referred to variable costs, because the new user addition is not complex and requires only extra computer.

Software.

Besides hardware and network equipment which are compulsory for cloud usage companies planning to use cloud platforms should have certain software resources as well to handle with tasks. For example workstations with installed development environment is required for the development process. As it was considered, for different types of platforms specific Integrated Development Environments (IDE) are used, e.g. programming with Microsoft Azure requires installed Microsoft Visual Studio environment (which is proprietary and demands appropriate operating systems, technologies and tools). Google and Salesforce for developing applications for their platforms provide Eclipse with special plugins and extensions which are free.

It is common that with the developing environments appropriate testing tools are delivered. As for modeling tools today SaaS solutions are in progress, but completely refuse from local software seems to be impossible, while desktop applications, like office suite, for documenting, writing requirements specification are available and widespread. For example, instead of purchasing office suite from Microsoft, Google's SaaS solutions are free and allow to benefit in collaboration (Martin, 2010). Almost all other tools for teamwork (e.g. bug tracking, versioning, communicative clients) have free solutions and can be used instead of proprietary.

Energy.

Evaluating energy needed for data centers brings another advantage of using cloud technologies, because for all high performance servers' power is needed not only for working but for cooling that equipment and when the load increases costs proportionally grows too (Lefevre, 2010). Transferring resources to the cloud reduces costs, associated with energy (Breeding, 2009).

People.

As concerns human resources they can be reduced significantly, because supporting workstations with few necessary programs is much easier than

dedicated servers in each branch and whole data center. Skills demanding from staff are less and the cost of labor decreases notably. Adopting new systems in practice demands administrators' trainings for understanding features for maintenance, as a result extra funds are charged.

Anyway workstations are not so big expense item, while servers' hardware and software require much more funds for purchasing and maintenance. The truth is that all facilities can be transferred to the cloud and a company does not need to buy or lease servers (even virtual) with operating system, required solutions, tools, etc. A platform is provided and it is enough to build, test and immediately deploy the application.

Effectiveness

Another distinguishing characteristic is a payment system; "pay-as-you-go" policy is very flexible and allows to save money and time in case of increased load. At the same time anticipating the possible scalability problems the company can buy equipment beforehand, but if hopes are not realized idling assets still need care, when funds are already allocated. Also the extension of existed resources in that case cannot be done immediately and demands time, efforts and as a result additional money.

Removing the need to buy equipment at a time, while monthly payments are used, results in investing money more reasonable and carefully. This also makes possible to abandon equipment if it is not needed more (e.g. the project is not profitable), while to sell hardware and software license cannot be done suddenly. Funds depreciation is less tangible when resources are rented, as well as the present value of resources is affected on the time value of money.

For some companies it is a problem to allocate money at a time, credit system can help with that, but it also has some limitations and difficulties. For example in (McGarvey, 2010) the situation is described when it was impossible to find money to upgrade computers in schools due to limitation in budget. Some of workstations were out-of-date and even Windows 95 was used on them. Also hard

drives were not working and no enough money to upgrade them all. Kyle Berger decided to enter IBM's cloud computing solutions, which allows to run Windows Vista on these machines remotely (McGarvey, 2010).

Traditional and cloud costs comparison

As it was described above, if the cloud computing approach is applied for development process, a company is out of need to buy expensive equipment (hardware, software), pay for qualified staff, etc. As a result it reduces costs and makes possible to offer more attractive prices for customers by contrast to competitors. As proponents of cloud technologies promise cost reduction will be between 30 to 90 percents (Hardy, 2009). Let's compare two ways – traditional and cloud.

Assume that a company negotiates a contract to develop a system during 2 years, after that for a term of 5 years the company undertakes to support and update the system on the customer's request. In addition management should care of hosting that system.

There are two ways to satisfy requirements: traditional and cloud. If the company decides to adhere traditional approach and enable in-house resources, the purchasing of additional hardware and software is necessary for developing and hosting the system. As well as hiring or training people, administrating that system, if present do not satisfy. With the cloud platform usage that is not compulsory. Let's consider what costs the company will incur on initial phase and annually.

Traditional:

- Prerequisite hardware and software
- Developers, testers, managers wage
- Administrators' wage
- Internet connection
- Electricity for additional servers

Cloud:

- Developers, testers, managers wage
- Staff trainings about using the cloud platform
- Administrators' wage
- Internet connection

In terms of quantity it is difficult to make estimation accurately due to information rapid obsolescence. However the ratio of prices is usually constant. To be free in that limitation let's accept the costs measurement as conventional units (CU).

On the initial phase with traditional approach should be purchased hardware and software; for cloud it is unnecessary, approximate estimation is 200 000 CU (high performance servers with operating systems, databases, etc.). However on this phase for cloud, staff learning and trainings should be included. For current example 2 000 CU is assumed.

In consideration of monthly payments in traditional approach to costs, the company has to pay remuneration to people, involved directly in developing process and administrators who maintain infrastructure and servers, responsible for customer's application, as well as payments for the Internet and electricity. Using the cloud platform company should also pay monthly for the Internet and electricity (which is less due to the servers' number reduction). Relative to people (developers, testers and managers) they are needed as many, but administrators for own infrastructure now should not be as qualified as before and their quantity also can be reduced. During the development phase the company should not pay for consuming resources. Only when they deploy a system on the provider's side monthly payments occur.

For that project is a team, including 7 developers, 3 testers and 2 project managers. In terms of money, salary level is as follows:

Position	Salary (CU)
Project manager	5 000
Developer	3 000
Tester	2 000
Administrator	2 000
Qualified administrator	3 000

Annual amount is 480 000 CU for traditional and 468 000 CU for cloud. Internet connection is the same and costs 500 CU per month (6 000 CU annually) and electricity bills for additional servers is 200 CU (2 400 CU annually). Consumption of cloud resources estimates as 2 000 CU monthly (24 000 CU annually). In the table 6.1 anniversary expenses for every year when the project is developed and supported are presented.

Table 6.1 – Anniversary expenses for traditional and cloud technologies.

Year	Traditional	Cloud
0	200 000	2 000
1	488 400	474 000
2	488 400	474 000
3	44 400	54 000
4	44 400	54 000
5	44 400	54 000
6	44 400	54 000
7	44 400	54 000

As of today, summing these values in consideration of the formula for the future value with negative time (Garger, 2010):

$$C_t = C(1 + i)^{-t} = \frac{C}{(1 + i)^t}$$

where 'i' is a discount rate, taken equal to 0.1. For traditional approach the value 1 186 736,31 CU and for cloud 993 820,24 are obtained. If a discount rate is taken equal to 0.05, then values are 1 282 493,21 and 1 095 416,54 respectively.

The cloud technology lets to reduce costs for 19.4% (17.1% for discount rate equals to 0.05) as a result the profit increases with the same quantity of customer's funds. If keep in mind, that with the cloud platform time of the development process also reduces (the example, when a company developing customer's application based on the cloud platform cut time from 3 years to less, than 6 months (Narasimhan, 2009) this ratio will go up. The assumption is also made that company already has an infrastructure, which satisfies its needs. If a new company or branch is setting up, then on the initial phase additional funds in equipment, including hardware and software, time, allotted for configuring and deploying, should be invested in traditional approach.

Some authors suppose that internal data centers with deployed platforms are cheaper, than cloud computing providers can offer nowadays. For example, McKinsley in his report on cloud computing doubts about reducing costs while replacing own resources to the cloud for large enterprises. He calculates total cost of assets and arrives at the conclusion, that using the cloud is economically inefficient (McKinsey & Company, 2009). At the same time Mrichandani contests McKinsley's estimations in (Mrichandani, 2009).

He notices that the average cost for simple data center environment is \$200 – 1000 per server and according to resources located on provider's side is more expensive. Mrichandani also calculates the value of storing data in the cloud, for one petabyte, located on Amazon's servers with additional fees for transferring data is about \$ 2 million when using internal data center's resources this cost, including the hardware, software, maintenance services is close to the value (Mirchandani, 2009).

After migration to the cloud platform a company should not keep a data center more, even the usage of infrastructure in the cloud is not necessary. A company just uses the platforms and other facilities, like APIs, provided by the service

provider. That is significant for company's costs and is possible for enterprise, medium and even small businesses if they use data centers (Narasimhan, 2009).

In terms of concrete values Narasimhan makes the example of saving over 30% on operating costs for more than 150 customers and as for time-to-market the company achieved 2-3x improvements. Actually, developing application by means of the cloud platform, made it possible to build a customer's application during less than 6 months, while the estimation of the same efforts using the traditional approach was 3 years. As a result the reduction of 50-75% in the time and efforts bring to the positive impression from the Force.com platform, because it was used to develop the application.

Besides that view on costs another pitfall in using cloud platforms exists. The problem of changing vendor faces to extremely high purchases. There can be different situation when moving to another provider is necessary, but the risk that a company can be out of business also should not be omitted. Nowadays it is not simple at all to jump from one platform to another and usually changing a vendor brings to tangible additional costs through simple mechanism of migration does not exist.

The specific is that reducing the quantity of capital expenses is not always a good thing, it is not obligatory that transferring costs from capital to operating will bring to positive effect. As Paul in (Paul, 2009) describes the opinion of Anthony Hill, CIO of Golden Gate University, that it is "often an advantage to bury IT costs in the capital budget and keep it off the P & L (Profit and Loss) for a couple years" (Paul, 2009) this point can be interpreted in two ways. Probably if a company uses that tactic migrating to the cloud will indispose to benefit from it.

6.2 Risks

Besides different views for costs people, who estimate the possibility to migrate, should not blind of economic benefits because risks are rather essential and companies should be aware of security and privacy sides of it. If data no more under customers control possibilities of losing control and accessibility exist.

All possible risks are described below but due to the main pitfall of security and privacy they are carried out in special part [5. Security and privacy](#). Following risks are important and should be estimated during the decision-making if the cloud is remunerative or still too risky (Greenwald, 2010):

- The risk of system failure
- The risk of unauthorized access
- The risk of a lack of control over privacy and data security
- Risks associated with intellectual property and privacy
- The risk of provider goes out of business

The risk of system failure is connected with the problem that provider's equipment is not perfect and even it fails. That situation is possible and although providers have reliable environment and systems, they are not insured of failures, there are some practical examples, for example, Google's cloud service, providing emails, Gmail was down several times (Hardy, 2009) for some hours (Paul, 2009) during 2009 and Amazon had the same problems lasting for several hours in 2008 (MacManus, 2008). And when it occurs, users do not have an access to their own applications.

If a development company entrusts to the cloud platform managers should take into account that fact and not only developing process can be stopped when the system fails, but guaranteeing the quality of the product and time of availability they have to count what real value they can promise. Making Service Level Agreements (SLAs) between the company and vendor may solve that problem, but there are still questions if users need more reliability than vendor promises (Yapp, 2009).

The risk associated with unauthorized access, security and privacy has several points of view. Some experts suppose that having the direct impact to own data is more reliable and compare it with the money keeping analogy "Do I stick my money in a mattress, or do I put it in the bank?" (Korolov, 2009). On the other hand the majority of specialists hold the view point that security issues are not

now on required level. Indeed, there are some essential characteristics associated with security that should be described more.

It is a feature in the cloud that data lifecycle, including collecting, entering, processing, transmitting, storing, reporting and exporting phases is not more in customers' hands (Sarrel, 2009). Therefore all steps should be well protected from internal and external threads, as it was described in part 5. To get information from provider how well it is performing, what methods are used on every phase is important to estimate all possible risks, but in some cases it is unrealizable.

Another difficult point is a result of the cloud approach's advantage – computing data is located somewhere. If a company entrusts provider to keep data and applications it is not obligatory that the company still owns data (Sarrel, 2009). According to a contract and legislations that situation can vary. Law regulation is the most significant weakness today. Service provider cannot say where the data is at a certain time, while it influences on regulations, applying for it. Another feature is that in some cases data cannot leave the country's borders, e.g. in Europe it is prohibited to “move data (via computer) outside the Europe Union without having a legal basis in place to transfer that data to another jurisdiction” (Greenwald, 2010). In addition The Data Protection Act in the US requires to control personal data while processing and storing, which is a problem in case of the cloud.

Intellectual property issues are also difficulties in adopting cloud computing everywhere. When data is stored on provider's equipment the question that who is responsible for legislation contradictory content arises.

The risk that one day a provider goes out of business is essential due to possibility of this situation. As well as customers' applications are developed according to characteristics of certain platform they cannot be easily moved to another technology in few steps. Creating applications for special platform faces to the problem that customer is locked within the environment, including particular frameworks, tools and technologies. If application is created with standard platform it can be hosted inside or on provider's equipment, that allows to change

provider without significant efforts as compared with platforms in the cloud and commitments to customers will be carried out.

Another question is that data types and storing methods are developed with a glance to platform's specific and in case when a customer wants to change provider they are unable to do it without significant expenses to provider or third parties. It is an example when a provider "alleged that the customer breached security and cut the client off from his own data, a situation that took a lawsuit and several years to resolve" (Greenwald, 2010). That can be the possibility that provider will cut customer's service and start dictate terms or customer will not be able to run the business anymore.

All these risks bring to estimation what data can be transferred to the cloud and what should be kept inside. Sensitive, critical, intellectual data should be processed in company because the risk of losing it is rather high, while the value of the data is extremely important. Some providers offer their services "as is" with no warranty and customers are responsible for all issues that can be involved themselves (McAlphine, 2010). Private clouds and SLAs can help to solve these problems but more important is to choose right provider with minimal risks, discover its protection methods carefully and work side by side. Legislation is a weak point and it should be improved according to the contemporary situation. Probably making possible to choose in what country data should be processed will reduce the impact of laws' restrictions (Edwards, 2009) but it requires careful assessment and government's regulations. Providers' audit is intended to help potential customer to estimate how certain company satisfies government and industry requirements. For example SAS 70, developed by the American Institute of Certified Public Accountants makes demands to "data transmission and storage technologies and practices, including network operations, data safeguards and physical security elements" (Edwards, 2009).

6.3 Strategy

Cloud platforms make possible to create new or develop existed company's strategies. Having some specific advantages over traditional technologies they can

be directed toward product and market improvements. Today there are different ways how a company can progress: some of them are general, while others are specific. Cloud platforms offer opportunities for software development companies to achieve competitive edge, reducing costs, development time, anticipate probable problems (e.g. unexpected overload).

In management there is a way, which allows to decide what direction should a company take a ply in consideration of current situation and company's goals. The Ansoff Product-Market Growth Matrix, created in 1957 by Igor Ansoff describes the dependencies between markets and products and provides possible development strategies. In the figure 6.2 his idea is presented (Ansoff, 1957).

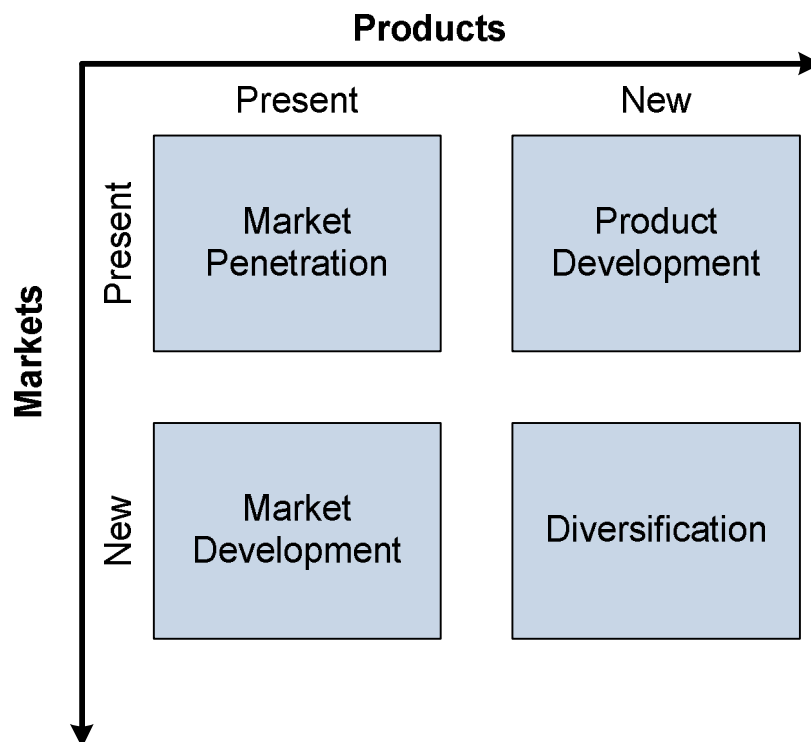


Figure 6.2 – The Ansoff Product-Market Growth Matrix.

Relating to software companies and cloud platforms various directions can be chosen. By means of cloud platforms to invent a really new product seems to be doubtful, while extend basic characteristics is reasonable. Web applications captured headlines and developing existent techniques makes possible to draw new customers.

A company should decide what is better to develop. According to Ansoff's Matrix following directions provide strategies and describe actions, which the company should make to achieve a goal.

Market Penetration. Market penetration is one of obvious strategies for most companies. A company is already on the market and their aim is to push up sells. The main method is competitive recovery; therefore attention should be directed to effectiveness increase of business processes. As a result products' consumption may be grown up by existed customers and due to drawing new. Probable methods are:

- Market-share gain. Reduced costs and time allow to offer competitive prices. At the same time satisfying customers' needs by means of cloud platforms creates additional potential in case of unexpected growth and prevents possible problems. In additional customers are insured of investing significant extra funds if a system should be reorganized.
- Increase frequency of use. For example using loyalty programs – a form of marketing, which aimed to create long-term relationships with customers, allows to analyze consumers needs and provide services that fully satisfy expectations. Methods, like discount system for regular customers create favorable conditions for both parties. Customers are ready to order more frequent, while software companies increase their profit. Cloud platforms have advantages in integration with other services and systems, because they are created generally by titans (Google, Microsoft, Salesforce.com) as supplementary on the basis of their existing technologies. That fact also makes possible to increase frequency of use and achieve the goal of increasing existent market.
- Increase the number of use. Customers either purchase existent products that can satisfy the majority of typical companies or order special applications for their concrete needs if there are no solutions yet. Software companies may create additional systems beforehand which expand the range of products. It makes possible for customers to buy already developed application to get complementary functionality.

Market Development. Market development is second possible solution, where a company tries to adopt existent products for new markets. For that goal new potential customers should be defined for present products. Companies that have developed marketing knowledge can use it efficiently to get required advantages by:

- Geographical market development. With cloud platforms new branches creation in different regions and countries is simpler than it was previously. Collaboration becomes easier and costs for setting up a company are not great, comparing with the traditional approach. Geographically distributed teams can operate together more efficiently and market development, by entrance to new areas in various regions, where that type of applications and systems can lack, results in customers canvassing and earning profit.
- Search of new market segments, which are not customers for that products' group so far. Transferring customers, who use traditional applications that need high scalability and performance to the cloud solutions, will obtain extra funds for the company, developing web-based applications. Opportunity of growth, close integration with various services and business platforms bring to grow of customers' interest to applications in the cloud.

Product Development. Updated and extended products' characteristics that adapt product to the present market is a way how companies can push up sales and gain new profit earners. To achieve that goal following directions can be chosen:

- New properties supplement or creation of a product with increased quality. Cloud platforms provide a big variety of functions which can be used during the development process. Increased resources' reserves that can be applied in few clicks and raised reliability create attractiveness for customers, but applications' advancement should be done with close cooperation with marketing and PR divisions.
- Development of new products generation. Up-to-date technologies are involved in creation cloud applications which make them attractive for

end-users and allow to replace legacy applications to cloud smoothly. Big variety of additional features, applications and functions create possibilities to make new products' type, all the more the majority of corporations have seen the true values and continue to develop the cloud approach.

Cloud platforms provide powerful features and opportunities for software companies. Essential cost saving is the most attractive point in applying cloud platforms, it allows to get competitive edge and based on it develop company's strategies. At the same time a problem, that data is located out of customer's equipment provokes corresponding risks. These risks are associated with strong dependence between customers and providers.

To take advantage of cloud platforms and make effective strategy is important to draw customers, develop markets and gain new profit. Qualified management especially in marketing and PR fields, besides technical leaders, should correctly estimate where the company is today, where it is going and chose the most appropriate methods to achieve aims. Well planed strategy, clear targets and quality people is a half the battle.

7 RESULTS AND DISCUSSIONS

During the work significant results were obtained. As compared with traditional technologies PaaS has essential features that can be useful for software development companies. From technical and economical sides PaaS provides opportunities that can be used by software development companies to get competitive edge. However all characteristics should be well discovered by a management of company to weight pros and cons. That section mentions all advantages and disadvantages, which were discovered in the work. They should be taken into account during the decision making process about the possibility to enable cloud resources. As well as anticipation of PaaS development in the future allows to decide if cloud platforms are suitable for adoption or it is better to wait.

7.1 Advantages and disadvantages

Platform as a Service is still the developing approach. It will be progressed in the future, new technologies, concepts and features will extend existent functionality and developers will get opportunities to use the cloud for their needs. There are still some disadvantages that put obstacles in the way of adoption cloud platforms.

PaaS advantages are as follows:

- Built-in infrastructure simplifies cost and time savings.
- Development phase is free while deployment on provider's side is monthly paid.
- “Pay-as-you-go” system is flexible and effective.
- Geographically distributed teams can collaborate more efficiently.
- Close integration with other technologies and platforms.
- Increased computer power and unlimited storage capacity.
- Whole development lifecycle is supported within the bounds of the safe environment.
- Web based applications are device independent.
- Effective administration.
- Global accessibility.

PaaS disadvantages are still rather significant and companies should take them into account:

- Security and privacy issues.
- Significant quantity of risks.
- Intended to web-based applications only.
- Difficult adapted to specific field.
- Features might be limited.
- High dependence on a provider.

Whether a cloud platform provides enough facilities for developing web-based applications or disadvantages prevail: how sensitive data is and what legislation is applied for applications and data should be estimated beforehand. PaaS opens up opportunities for all types of businesses: small, medium and enterprise, but management should handle with them and only qualitative people are able to plan development strategies, apply it and benefit from the platforms' usage. According to company's specific and characteristics decisions are made in house.

7.2 PAAS in the future

Yapp mentions that users and suppliers need to address the cultural changes it brings (Yapp, 2009). Another prediction was made by IDC – the US research firm that the cloud computing market will reach \$42 billion in 2012 (Oppenheim, 2009).

Developing technologies and applying new concepts, allow the cloud approach to be an attractive way for many development companies, specialized in creating web applications. According to surveys there are different results about present situation in adoption cloud platforms. Many CIOs are interested in the cloud but still wait when more reliable guarantees will occur, Rubenstein in (Rubenstein, 2010) notices that “56% of CIOs are not even familiar with cloud services” and the author shows barrier to cloud services (Rubenstein, 2010):

- Security 68%
- Integration with Internal 61%
- IT applications performance/reliability 58%
- Transparency of costs 47%
- Regulatory and compliance 43%
- Lock-in 37%
- Geographic location 21%

It is important to let down these bars for providers if they want to increase the number of customers among them who still have doubts. The surveys show that software development companies apply cloud platforms to their processes and the cloud approach becomes more popular day by day. In accordance with Wood “Only 8.7% of North American developers are currently using a cloud service as a development environment, but another 18% plan to within the coming year” (Wood, 2010)

In 2008 IDC predicted, that 25 percent of IT spending will be geared toward cloud computing resources, they also surveyed 244 heads of IT companies about cloud computing and as a result three-quarters of respondents answered, that cloud security is one of the most significant obstacles to reaching widespread acceptance (Veen, 2009).

All vendors will try to remove obstacles, which deter now potential customers from adoption clouds. "You do not see a lot of businesses now with spare technicians to migrate their software over [to the cloud] ... Anything with real business value won't be on the cloud for years" (Hardy, 2009).

Another significant obstacle, which requires a lot of efforts, is legislation. Different countries with limiting laws impede computing data to leave the country's borders. That conflict between current laws and the cloud philosophy now cannot be solved.

According to Gartner Inc., cloud computing is in progress at the present time, however it requires several years to become a mainstream IT effort (Velte, Velte

& Elsenpeter, 2009, pgs. 309-311). Gartner also observes that aggressive application development organizations should look to cloud computing for tactical projects through 2011, during which time the market will begin to mature and be dominated by a selected group of vendors. New vendors will appear during that time and more software development companies will pay attention to that approach. By 2015 the cloud applications will be the main direction in developing web based solutions.

Mark Driver, research vice president at Gartner foresees three phases of cloud evolution (Velte, Velte & Elsenpeter, 2009, pgs. 309-311):

Phase 1: 2007 to 2011 – Pioneers and Trailblazers. A market development phase, where technology providers with strongest market “vision” will garner the most success among early adopters. Many vendors will implement rapid applications development process and special features for making these applications attractive for end users. Mainstream IT developers will focus on investing funds to the cloud projects, “where return on investment can be acquired within 18 to 24 months”.

Phase 2: 2010 to 2013 – Market Consolidation. During that time the market will be overcrowded with a set of application from different vendors and weaker will be driven. At the same time more potential adopters will be involved to extend the existent cloud market. The reliance to the cloud will increase and “longer-term strategic investments” will be applied to development the cloud solutions. The term of returning investments will be from three to five years.

Phase 3: 2012 to 2015 and Beyond – Mainstream Critical Mass and Commoditization. In 2013 large providers will prevail on the market and promote proprietary technologies, based on standards created themselves. The market will be stable and competition between open and proprietary platforms will bring to support for one or more open software stacks.

In the future the integration between technical and social sides of the clouds will continue. The Service Cloud – the program created by Salesforce.com, which brings together platforms, like Google, Facebook and Amazon.com to capture

social aspects of the cloud, such as conversations and communities' management. Socialization becomes more and more important in today life, according to Gartner's analyst Michael Maoz "The new generation of consumers trusts content created by peers. This consumer expectation that they can create answers and content as part of a community will lead business and other organizations to adopt similar techniques succeed. Ultimately, organizations will have to change their singular emphasis on tools for agents, to a broader strategy that also supports the role on community experts."

Great attention is paid to cloud computing in general, but the lack of scientific studies of the PaaS technology brings to the problem that some aspects of that approach are not clear and have varied views. During this work the attempt to consider cloud platforms from divergent angles was made. PaaS is new and modern approach, that is why the main weakness of the study was that sources, provided any information about PaaS are more commercial, than scientific and it was difficult to estimate real value of the approach. On the basis of general cloud computing description propositions relating to PaaS were made, but closer look to characteristics and details with field data, gathered from observations of real products, comparing providers, etc. can be done to extend the basic overview of the technology.

In the future works in PaaS field due attention can be given to objective study of the technology. Perhaps it is a good decision to make own independent examination of existent solutions or survey companies, which apply cloud platforms in practice about their way and estimations. This work is based on literature review with corresponding drawbacks and lack of scientific studies due to a novelty of the topic. Following research can be free of these limitations.

8 CONCLUSION

On the basis of the research the literature study method was applied to define characteristics of Platform as a Service, which are important for software development companies.

PaaS provides great opportunities for software companies to improve the development process, to reduce costs and time needed to create high quality, powerful applications, but at the same time it possesses some drawbacks that put obstacles in the way of adoption cloud platforms. The efforts during the work were focused on considering the issue in various aspects: technical, economical, risks and can be useful for estimation real value of the approach.

Besides technical features which are described enough in the press, other sides of PaaS is not well discovered. During the work the model of a typical software development company was examined to understand needs in hardware and software and assume what can be transferred to the cloud. That model also promoted to create a scenario of migration to cloud platforms. The attempt to make evaluation of costs saving and effectiveness in terms of money was done in order to create a model in the future that can be used for complex assessment of PaaS deployment expediency. Probable risks were revealed to notify customers about potential threats. The main obstacles (security and privacy) in adoption PaaS were examined to understand their real importance and influence on the possibility of using cloud platforms in developing web applications.

PaaS opens opportunities for companies and promotes strategic development. The ways, that a company can apply to benefit, according to the goal and current situation, are described as well. In spite of technological advancement, storing data outside the company is fraught with danger to lose control over them. In addition legislation in countries mismatches with the cloud's approach – computing data somewhere. At the end of the work the advantages and disadvantages were identified and anticipation of PaaS development in the future was done.

REFERENCES

Ansoff I. (1957) *Strategies for Diversification*, Harvard Business Review, Vol. 35, pgs 113-124, Harvard.

Babcock C. (2009) *Platform as a Service* InformationWeek 5 Oct. 2009: ABI/INFORM Global, ProQuest.

Binnings D. (2009) *Are hosted services more secure?* Computer Weekly 5/19/2009, p16-18, UK.

Breeding M. (2009) *The Advanced of Computing From the Ground to the Cloud*, *Computers in Libraries*, Vol. 29, pgs 22-25, Computers in Libraries is the property of Information Today Inc.

Carr N. (2009) *The Big Switch: Rewiring the World, from Edison to Google*, W. Norton & Company, US.

Chappel D. (2009) *Introducing Windows Azure, v. 1.2*
Available: <http://go.microsoft.com/?linkid=9682907>
Accessed: 04.05.2010.

Chappel D. (2009) *Windows Azure Platform, v. 1.3*
Available: <http://go.microsoft.com/fwlink/?LinkId=158011>
Accessed: 04.05.2010.

Cloud Security Alliance. (2010) *Top threats*
Available: <http://www.cloudsecurityalliance.org/topthreats/csathreats.v1.0.pdf>
Accessed: 04.05.2010.

Dupre F. (2008) *Utility (Cloud) Computing...Flashback to 1961 Prof. John McCarthy*
Available: <http://computinginthecloud.wordpress.com/2008/09/25/utility-cloud-computingflashback-to-1961-prof-john-mccarthy/>

Accessed: 04.05.2010.

Edwards J. (2009) *Cutting Through the Fog Of Cloud Security*, Computerworld, Vol. 43, pgs 26-29, Computerworld.

Fielding R. (2000) *Architectural Styles and the Design of Network-based Software Architectures*

Available: <http://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm>

Accessed: 04.05.2010.

Garger J. (2010) *The Formula for Calculating the Present and Future Values of an Annuity*

Available: <http://www.brighthub.com/money/personal-finance/articles/17948.aspx>

Accessed: 04.05.2010.

Google (2010) *Google App Engine*

Available: <http://code.google.com/intl/en/appengine/>

Accessed: 04.05.2010.

Hardy Q. (2009) *A Zure Thing?* Forbes Asia, Vol. 5, Forbes Asia, pg. 99.

Hurwitz J., Bloor R., Kaufman M., Halper F. (2009) *Cloud Computing For Dummies*, Wiley Publishing, Inc., Canada.

IEEE and EIA (1998) *Industry Implementation of International Standard ISO/IEC 12207: 1995 (ISO/IEC 12207) Standard for Information Technology — Software life cycle processes.*

Intuit (2010) *Intuit platform*

Available: <https://ipp.developer.intuit.com>

Accessed: 04.05.2010.

Johnston S. (2009) *Cloud Computing Economies*

Available: <http://samj.net/2009/03/cloud-computing-economics-101.html>

Accessed: 04.05.2010.

Khalid A. (2010) *Cloud computing: Applying Issues in Small Business*, 2010 International Conference on Signal Acquisition and Processing, pgs. 278 – 281.

Klems M., Nimis J., Tai S. (2009) *Do Clouds Compute? A Framework for Estimating the Value of Cloud Computing*, Vol 22, Lecture Notes in Business Information Processing, pgs 110-123, Springer Berlin Heidelberg, Germany.

Korolov M. (2009) *Is It Safe In The Clouds?* Securities Industry News, Vol. 21, pg. 14, SourceMedia, Inc.

Lefevre L., Orgerie A. (2010) *Designing and evaluating an energy efficient Cloud*, *Journal of Supercomputing*, Vol. 51, pgs 352-373, INRIA RESO, LIP (UMR CNRS, INRIA, ENS, UCB), Universite de Lyon, Lyon, France.

Levinson M. (2007) *Software as a Service (SaaS) Definition and Solutions*
Available:

http://www.cio.com/article/109704/Software_as_a_Service_SaaS_Definition_and_Solutions

Accessed: 04.05.2010.

Linthicum D. S. (2009) *Cloud Computing and SOA Convergence in Your Enterprise: A Step-by-Step Guide*, Addison-Wesley Professional, US.

MacManus R. (2008) *More Amazon S3 Downtime: How Much is Too Much?*
Available:

http://www.readwriteweb.com/archives/more_amazon_s3_downtime.php

Accessed: 04.05.2010.

Martin J. (2009) *Should You Move Your Business to the Cloud?* PC World, Vol. 28, pgs 29-30, PC World Communications Inc.

Mather T., Kumaraswamy S., Latif S. (2009). *Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance (Theory in Practice)*. O'Reilly Media, US.

Mathew R., Spraetz R. (2009) *Test Automation on a SaaS Platform*, 2009 International Conference on Software Testing Verification and Validation.

McAlpine C. (2009) *Consider the Legal Risks of Cloud computing Baseline*, pg. 32, Ziff Davis Enterprise.

McGarvey R. (2010) *You Want Me To What?* T H E Journal, Vol. 37 Issue 2, pgs 32-38, 5p.

McKinsey & Company (2009) *Clearing the Air on Cloud Computing*
Available: <http://images.cxotoday.com/cxoimages/storyimages/matter101157.pdf>
Accessed: 04.05.2010.

Miller M. (2008) *Cloud Computing: Web-Based Applications That Change the Way You Work and Collaborate Online*, Que Publishing, US.

Mirchandani V. (2009) *McKinsey's Dark Clouds*
Available: http://dealarchitect.typepad.com/deal_architect/2009/04/mckinseys-dark-clouds.html
Accessed: 04.05.2010.

Narasimhan B. (2009) *Cloud Computing Savings – Real or Imaginary?*
Available: <http://blog.appirio.com/2009/04/cloud-computing-savings-real-or.html>
Accessed: 04.05.2010.

Oppenheim R. (2009) *A Match Made in Heaven*, *Searcher*, Vol. 17, pgs 14-18, Information Today Inc.

Ouellette J. (2009) *Development with the Force.com Platform: Building Business Applications in the Cloud*, Addison-Wesley Professional, US.

Parinov O. (2009) *Internet startups with cloud computing*
Available: <http://habrahabr.ru/blogs/Azure/60100/>
Accessed: 04.05.2010.

Paul F. (2008) *Midsized Organization Busts 5 Cloud Computing Myths*

Available:

<http://bmighty.informationweek.com/services/showArticle.jhtml?articleID=211600030>

Accessed: 04.05.2010.

Paul I. (2009) *Gmail Fail: Is the Cloud Reliable?* Macworld Nov2009, Vol. 26, pg. 20, Mac Publishing LLC.

PR Newswire (2009) *Salesforce.com Announces Dramatic Acceleration in Adoption of Force.com Enterprise Cloud Computing Platform - Now More than 135,000 Custom Apps Built*, New York: Nov 19, 2009. ABI/INFORM Dateline, ProQuest.

PR Newswire (2010) *Salesforce.com Raises the Stakes for Enterprise Cloud Computing Platforms With New Force.com Visual Process* PR Newswire 3 February 2010 ABI/INFORM Dateline, ProQuest.

Ray R., Kulchenko P. (2002) *Programming Web Services with Perl*, O'Reilly Media, US.

Regulation of Investigatory Powers Act. (2000)

Available: http://www.opsi.gov.uk/acts/acts2000/ukpga_20000023_en_1

Accessed: 04.05.2010.

Rittinghouse J., Ransome J. (2009) *Cloud Computing: Implementation, Management, and Security*, CRC Press, US.

Roche K., Douglas J. (2009) *Beginning Java Google App Engine*, Apress, US.

Rubenstein R. (2010) *SAFE SETS*, Total Telecom Magazine Feb2010, p28-29.

Salesforce (2010) *Force.com*

Available: <http://www.salesforce.com/platform/cloud-platform/>

Accessed: 04.05.2010.

Sarrel M. (2009) *The Darker Side of Cloud Computing*, PC Magazine, Vol. 28, pg. 1, ZDNet.

The World Wide Web Consortium (2004) *Web Services Glossary*

Available: <http://www.w3.org/TR/ws-gloss/>

Accessed: 04.05.2010.

The World Wide Web Consortium (2007) *SOAP Version 1.2 Part 0: Primer (Second Edition)*

Available: <http://www.w3.org/TR/2007/REC-soap12-part0-20070427/>

Accessed: 04.05.2010.

Toffler A. (1984) *The Third Way*, Bantam Books, US and Canada.

Turner N. (2009) *Cloud Computing: A Brief Summary*, Lucid Communications

Available:

<http://www.lucidcommunications.co.uk/Content/whitePapers/CloudComputing.pdf>

Accessed: 04.05.2010.

Veen C. (2009) *Cloud Computing Survey Shows Growing Acceptance*

Available: <http://www.govtech.com/gt/633071>

Accessed: 04.05.2010.

Velte T., Velte A., Elsenpeter R. (2009). *Cloud Computing, A Practical Approach*, McGraw-Hill Osborne Media, US.

Wood L. (2010) *Research and Markets: North American I.T. Development Survey 2009: New Survey Shows Python Use Has Risen 45% Since Google App Engine Debuted*, Business Wire New York: Feb 3, 2010.

Yapp C. (2009) *Look beyond the technology of cloud*, Computer Weekly 7/7/2009, pg. 12, Reed Elsevier, Inc.