jamk.fi

IoT service platform architecture

Jukka Kalsi

Master's thesis
June 2017
Technology, communication and transport
Degree Programme in Information Technology

Jyväskylän ammattikorkeakoulu
JAMK University of Applied Sciences

jamk.fi

Description

Author(s)	Type of publication	Date
Kalsi, Jukka	Master's thesis	June 2017
		Language of publication: English
	Number of pages 40	Permission for web publication: x

Title of publication
IoT service platform architecture

Degree programme
Master´s Degree Programme in Information Technology

Supervisor(s)
Kotikoski, Sampo, Karjalainen, Mika

Assigned by
Telia Finland Oyj

Abstract

Telia Finland had some old IoT kind of services such as automatic meter reading for electricity companies as well as alerting and monitoring services. These services based on old legacy technology developed internally, making it difficult to reuse them to build new services. There was a need to renew those old service platforms and take one common platform into use to enable the development of Telia's new IoT services and also to offer a plain platform for partners and customers.

In accordance with need, the basic task was to evaluate different ready platform alternatives, select one that fit the needs best and then implement the new service platform. After vendor selection, the plan was to build up a running test environment, run the company's own internal testing there and start implementing old services and needed internal processes.

Very soon it was found out that there is no ready platform that would meet all the needs of the old services, and the scope was then changed into building up a platform only for new IoT needs and migrate the existing services there gradually if possible. The platform vendor was selected, and the new platform was technically built up. The company's own testing and development was carried out with some Proof of Concept customer projects and the first few new IoT services are about to be launched in public.

It was found out that the old legacy platforms and services are very hard or impossible to replace with new ready-made platform solutions without massive extra work on migration and integration. On the other hand, the new platform was designed and built up according to the new scope. There are still several issues in need of further development when customer use increases.

Keywords/tags (subjects)
IoT, Internet of Things, Industrial Internet

Miscellaneous

jamk.fi

Kuvailulehti

Tekijä(t)	Julkaisun laji	Päivämäärä
Kalsi, Jukka	Opinnäytetyö, ylempi AMK	Kesäkuu 2017
	Sivumäärä 40	Julkaisun kieli Englanti
		Verkkojulkaisulupa myönnetty: x

Työn nimi
IoT service platform architecture

Tutkinto-ohjelma
Master´s Degree Programme in Information Technology

Työn ohjaaja(t)
Sampo Kotikoski, Mika Karjalainen

Toimeksiantaja(t)
Telia Finland Oyj

Tiivistelmä

Telia Finlandilla on joitakin vanhoja IoT:n kaltaisia palveluita kuten automaattinen mittarinlukupalvelu sähköyhtiöille ja hälytys- ja valvontapalvelu. Nämä palvelut pohjautuvat vanhaan teknologiaan, jota on kehitetty talon sisällä ja jota ei ole helppo käyttää uusien palvelujen kehittämiseen. Vanhojen alustojen uusiminen tuli ajankohtaiseksi, samoin yhteisen alustan, joka mahdollistaisi myös uusien omien IoT-palveluiden kehittämisen ja alustan tarjoamisen yhteistyökumppaneille ja asiakkaille.

Tarpeen mukaisesti tehtävä oli arvioida valmiita alustavaihtoehtoja, valita parhaiten omiin tarpeisiin sopiva ja ottaa uusi palvelualusta käyttöön. Toimittajavalinnan jälkeen suunnitelma oli rakentaa toimiva testiympäristö ja tehdä siellä omaa testausta, minkä jälkeen aloittaa vanhojen palvelujen ja tarvittavien prosessien käyttöönotto.

Hyvin pian kuitenkin tuli selväksi että sellaista valmista alustaa ei löydy, joka täyttäisi kaikki vanhojen palvelujen vaatimukset. Tavoite muutettiin uuden alustan perustamiseen IoT-tarpeeseen ja vanhat palvelut siirrettäisiin mahdollisuuksien mukaan osissa. Sovellustoimittaja saatiin valittua ja alusta teknisesti toteutettua. Oma testaaminen ja kehitystyö tehtiin asiakasprojektien avulla ja ensimmäisiä uusia IoT-palveluita ollaan juuri julkaisemassa markkinoille.

Selkeä havainto oli, että vanhoja alustoja ja palveluita on erittäin vaikeaa tai jopa mahdotonta korvata valmiilla alustaratkaisulla ilman, että joudutaan tekemään paljon ylimääräistä työtä palvelun siirrossa ja integraatiossa. Toisaalta uusi alusta saatiin suunniteltua ja toteutettua uuden tavoitteen mukaisesti. Edelleen palvelussa on paljon kehitettävää asiakasmäärän kasvaessa.

Avainsanat (asiasanat)
IoT, Internet of Things, Industrial Internet

Muut tiedot

Contents

Figures

Tables

Acronyms

AEP	Application Enablement Platform
AMR	Automatic Meter Reading
API	Application Programming Interface
DM	Device Management
DNS	Domain Name System
GDPR	General Data Protection Regulation
HA	High Availability
HTTP	Hyper Text Transfer Protocol
HTTPS	Hyper Text Transfer Protocol Secure
IoT	Internet of Things Internet of Everything Industrial Internet
IIoT	Industrial IoT Industrial Internet of Things
ITU-T	International Telecommunication Union – Telecommunications Standardization Sector
LAN	Local Area Network
M2M	Machine to Machine
PoC	Proof of Concept
REST	Representational State Transfer
SOA	Service Oriented Architecture
SPOF	Single Point of Failure
UI	User Interface
WSN	Wireless Sensor Network

1 Introduction

“A global infrastructure for the information society, enabling advanced services by interconnecting (physical and virtual) things based on existing and evolving interoperable information and communication technologies.” (ITU-T Y.4000/Y.2060 (06/2012), 1)

That is how ITU-T defines Internet of Things.

IoT, Internet of Things is a hot topic today. People want to be on the internet and they want everything to be on the internet as well. Companies are pushing network capabilities on their devices to meet the customer requirements. Many new IoT services and IoT capable devices are coming to the market all the time. Who could have imagined 10 years ago that kitchen devices can be controlled or fridge contents can be checked with one’s mobile phone.This is reality today and more and more complicated use cases and solutions are released every day.

On the other hand there are no clear standards covering all interfaces so every manufacturer and service provider offers their own solutions which creates the problem that devices and services from different providers cannot communicate with each other. Devices also collect massive amounts of data which needs to be accessed but in some cases one has no access to that data. If one has access to data generated by the device it is very often located in service providers or device manufacturers environment.

Telia wants to be one player on that IoT playground and offer their own new IoT services as well as their development platform for partner and customer use. So far there is no existing platform for that kind of use which is why a new platform is needed.

2 Background

2.1 Research method

The research question set for this research is:”Is the platform to be implemented useable as a service platform for Telia IoT services?”

“There are two processes in design research: development work that is targeted, for example, at a process, product, service or action and research that results in thesis.” (Kananen 2013, 50)

The definition of design research state that this method combines the development and research and it always aims at a change. In addition, when discussing change, the direction should always be towards better. (Kananen 2015, 33)

The development in design research repeats the cycles of planning, action, observation and follow up as shown in the Figure 1. In companies, such development work is a continuous process. (Kananen 2013, 41-42)

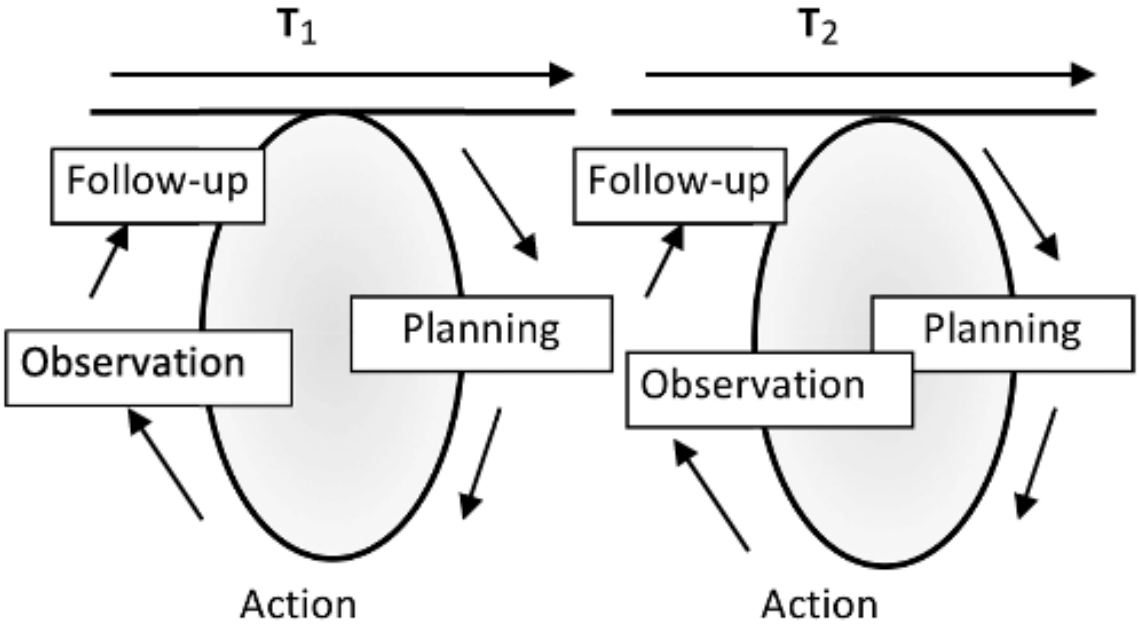

Figure 1. Cycles of the development work (Kananen 2015, 61)

Design research was chosen with the qualitative approach as the research meets the attributes Kananen states. The aim of the research is to develop a new service platform and to evaluate whether it can be used as an IoT service platform. The method chosen was seen as the most practical way to build up a working environment and run the own development and research side by side as well. The implementation of a new platform and its usability assessment cannot be measured by quantitative methods, therefore a qualitative method was chosen.

The cyclic development in design research is well suited to the sprint-based Agile development model used in this project.

The research material is mostly be collected by direct observation and discussion in development team.

2.2 IoT

IoT cannot be an unambiguously described, since it is not a one simple system but a higher concept and a collection of different technologies. The idea of computers everywhere was already inrtoduced in the 1980s and the term IoT was taken into use around the turn of the millenium. At first, the term IoT was more related to RFID; however, it was later broadened also to include other technologies. IoT itself is closely related to a bunch of other terms and abbreviations such as M2M and WSN just to name few. (Zhou 2013, Chapter 1.2)

Industrial Internet, Industrie 4.0 or IIoT, these all terms refer to the same Industrial Internet. That can be often interpreted as an internet service for the manufacturing industry; however, it is not just that. It is related to all industry level internet of the devices for also within another industry lines such as electricity and health care. These terms are more related to business services when the plain IoT term is more related to services offered to privat customers. (Collin & Saarelainen 2016, 29-35)

Both writers above can be agreed, IoT is not anything brand new and many different terms refer to same one thing. IoT is certainly an umbrella term for many different technologies under that and it cannot be handled as a one system. IoT has many variations and every technology area turns term into their own form or introduce a totally new one.

Gartner has released in February 2017 forecast that by the end of the year 2017 there would be around 8.4 billion connected IoT units in use and the number has increased by about 2 billion units in a year. The majority of devices, about 63 per cents, are consumer devices, and this is estimated to be the fastest growing category also in the future as can be seen in the following Table 1. (Gartner 2017)

Table 1. IoT Units Installed Base by Category (Millions of Units) (Gartner 2017)

Category	**2016**	**2017**	**2018**	**2020**
Consumer	3,963.0	5,244.3	7,036.3	12,863.0
Business: Cross-Industry	1,102.1	1,501.0	2,132.6	4,381.4
Business: Vertical-Specific	1,316.6	1,635.4	2,027.7	3,171.0
Grand Total	**6,381.8**	**8,380.6**	**11,196.6**	**20,415.4**

The direction Gartner presents cannot be disagreed with and at the moment the number of consumer units is about to grow the fastest. One thing that cannot be fully agreed with is the presented ratio between consumer and business units in the future as the number of business units seems rather low. Of course there can be very big differences in different parts of the world.

2.2.1 IoT Service

IoT service is not a solid block of software but it is divided into multiple layers with different functions and components from ean nd device to a user interface and everything between these. On a higher level, the service can be divided into three or four layers depending on the actual service. Layers are defined as the Perception layer including sensors, gateways and all the end devices. The next layer is the Network layer reponsible for connecting devices to the backend system. The last common layer is the Application layer which holds the actual intelligence and UI of the service. The four-layer model differs from the three-layer model and contains an extra Supporting layer taken apart from the Application layer as can be seen in Figures 2 and 3. That extra layer is responsible for all data management and computing capabilities. There are also other models introduced with different numbers of layers where the main layers are split into more detailed pieces. (Ning 2013, Chapter 2.3)

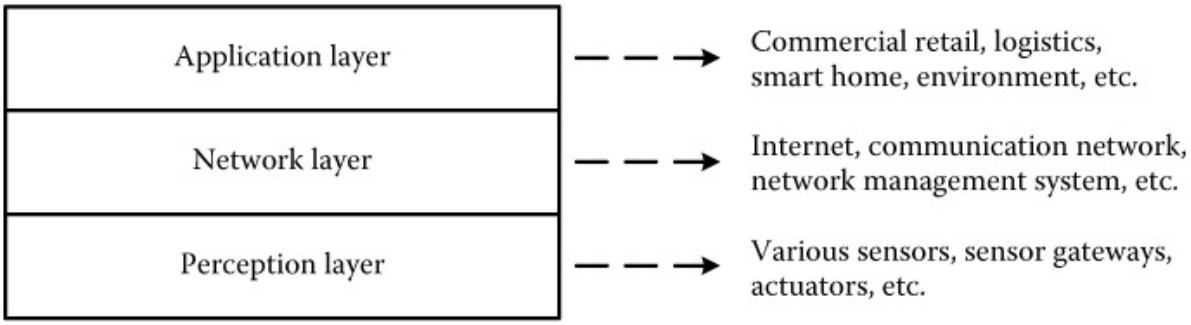

Figure 2. Three-layer IoT service model (Ning 2013, Chapter 2.3)

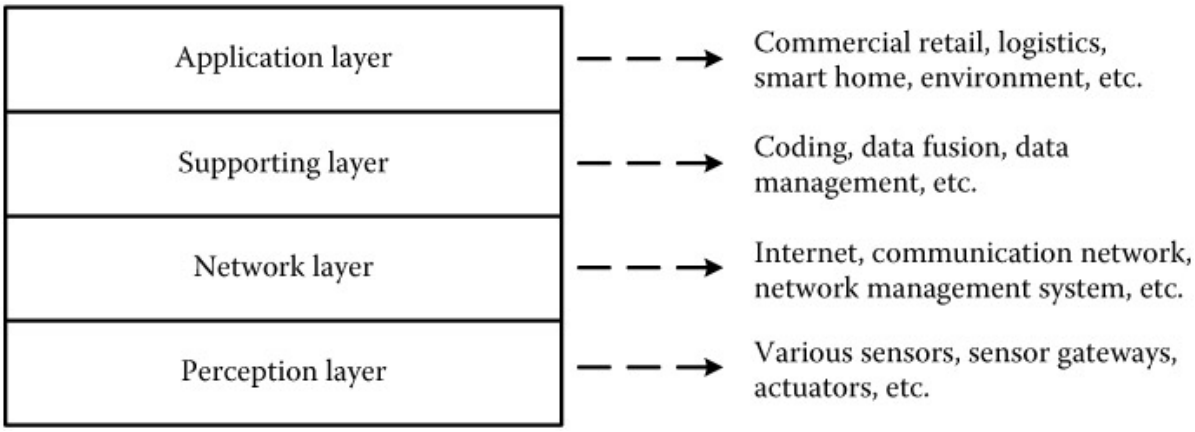

Figure 3. Four-layer IoT service model (Ning 2013, Chapter 2.3)

A four to six-layer technology stack is a common way to represent an IoT service. One good example of the multilayered model is a six-layer model as shown in Figure 4. In this model, the lowest layer is called Sensors and it contains sensors and other devices. The next layer, Communication, is responsible for the device connections. Storage is a layer where the device data is stored. Over Storage there is Analytics, which is the layer where the device data is managed and processed. Application layer is the layer that provides end user applications and UIs. The highest layer, Digital Service, is a commercial layer with business features and it cannot be seen as a technical layer. Layers from four to six can be combined depending on the capabilities of the used platform. (Collin & Saarelainen 2016, 142-144)

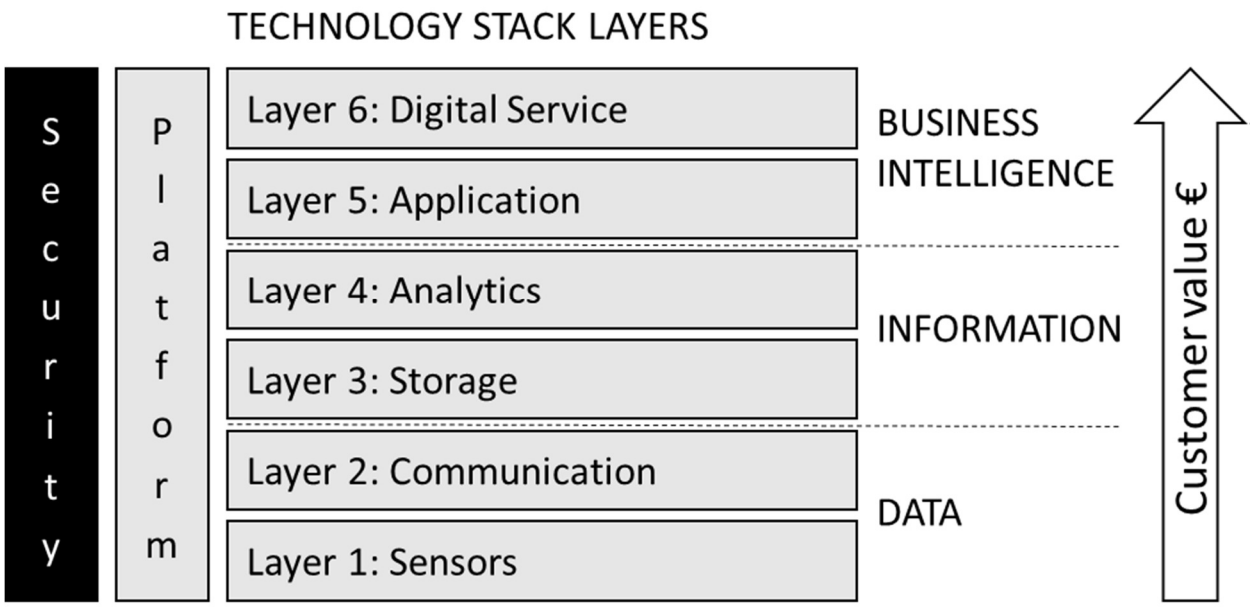

Figure 4. Technology stack (Collin & Saarelainen 2016, 143)

From another point of view the IoT service can be divided into four asset layers in a slightly different way as shown in Figure 5. In this model the end point device is separated from the gateway, the network layer is still responsible for connectivity and the data center layer holds all the platform intelligence. (Macaulay 2017, Chapter 2)

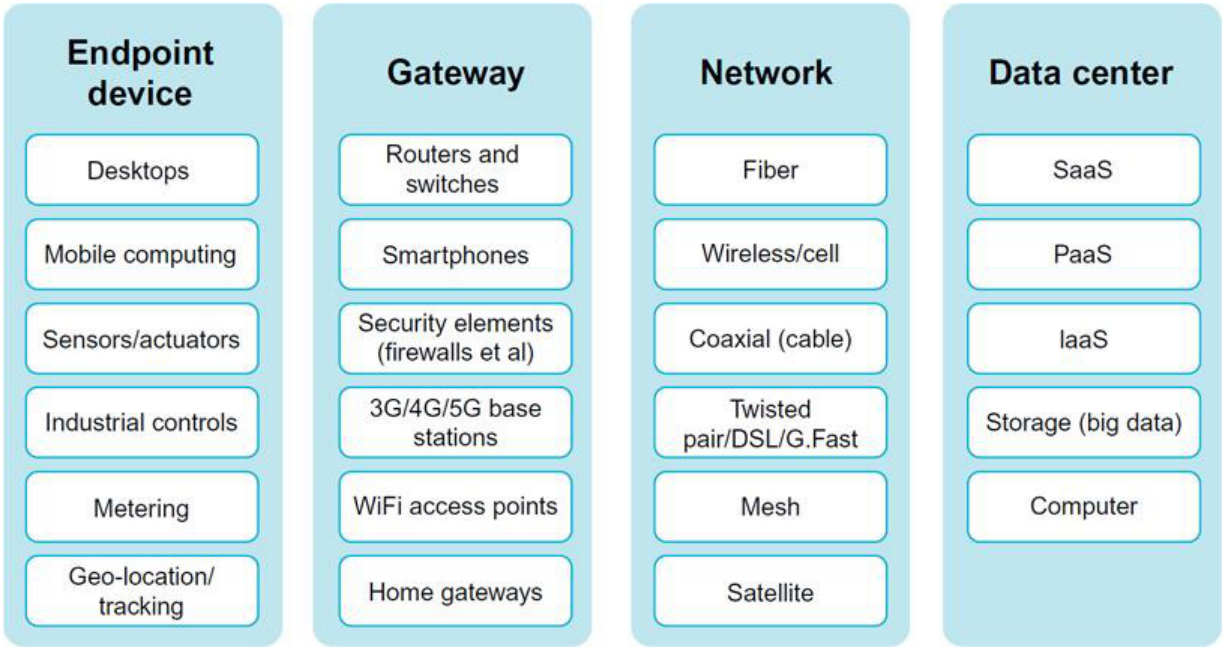

Figure 5. SoA for Iot (Macaulay 2017, Chapter 2)

Ning's model with three or four layers suits well on a platform studied in this research. Depends on the service in which the platform is being used, whether an

external system is needed as a fourth layer. The platform studied in this research has capabilities to be used on both Supporting layer and Application layer. On the other hand, external systems can be connected to the system and used at different levels of the service. Collin's model with six layers can be used as well; the top layers may, however, be combined together.

Sensors and gateways can be considered as one group and do not need to be separated as Macaulay suggests, as there are devices that can directly connect to the system and some that connect using special gateway. The network layer is always needed for devices to connect to a backend system. The backend system can be divided into two parts if the system cannot be used as the end user platform and some external system is needed.

2.2.2 Devices and gateways

IoT end devices can be divided into two different categories, sensors and actuators, based on their use on the system.

Sensors are units used to collect data such as temperature, humidity or velocity, they are linked to the rest of the system with wired or wireless connection.

Actuators are units used to perform actions such as adjust valves or control switches. Actuators receive their commands from the system and perform actions based on commands. (Ning 2013, Chapter 3.2)

Devices are the smallest part of the system at the farthest end of the whole service. They are the components that carry out the actual measurements or perform actions that the service has been developed for.

There are various reasons why gateways are used to communicate with endpoint devices such as sensors and actuators. Sensors can use some industry standard communication that cannot be used directly with the backend system. Sensors can use some short range wireless system as well and again cannot directly connect to the backend system. (Zhou 2013, Chapter 4.3)

Gateways are used to connect devices to a backend system in a situation where devices cannot directly connect to it. The gateways often has more computing

resources, better power supplies and a connection to the backend system. Therefore, they are used to collect data from sensors and to provide it forward to the backend system. (Macaulay 2017, Chapter 2)

Many wired sensors use different industry standards and cannot directly be connected to the backend system. On the other hand, wireless sensors always use some sort of batteries and to save energy, often use some short-range radio technology that cannot be used directly with the backend system. Gateways are used as bridges to connect these different technologies into one system.

2.2.3 Platform architecture

The platform architecture can be designed using many different aspects. For security reasons it is advised to use multi-tier architecture for server architecture. In that kind of architecture the servers with same security requirements can be placed in a shared LAN and restricted from others using firewalls as shown in Figure 6. (Arregoces & Portolani 2003, Chapter 4)

Another aspect on the architecture design is service availability. To ensure service availability, different technologies can be used to share load between servers running the same role and processes. (Arregoces & Portolani 2003, Chapter 3)

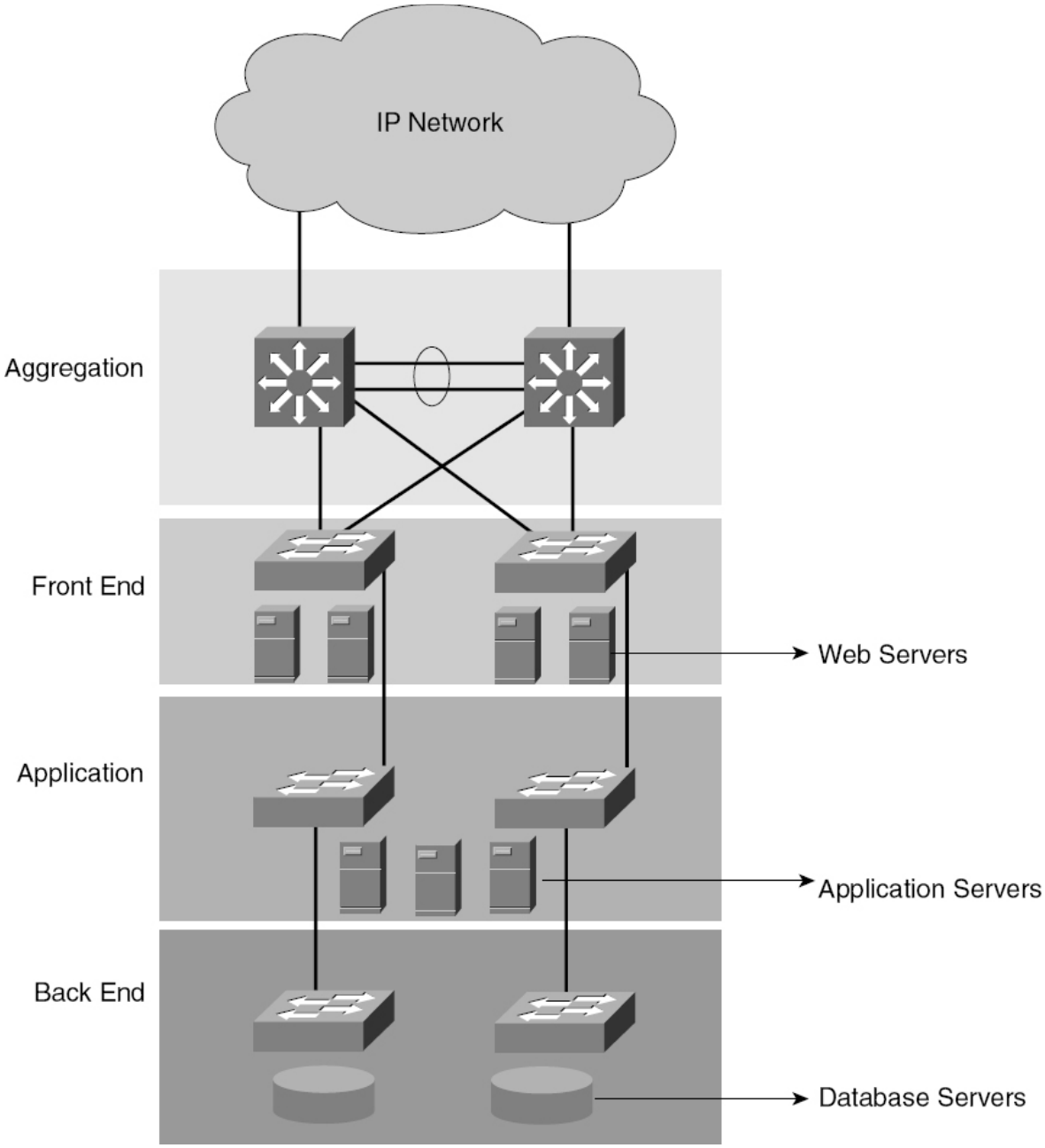

Figure 6. Secure server architecture example (Arregoces & Portolani 2003, Chapter 4)

The basic architecture of the IoT service platform can be seen in the same way as any web-based service with separate front end, application and back end servers. All in all, such an architecture is a good approach to limit access to one system level in such a situation that someone erroneously get access to the service platform. Usually the system can also be scaled at a certain level if necessary.

The system must be designed in such a way that the best possible availability can be guaranteed as well. To ensure system availability, the system should not have any SPOFs.

2.2.4 IoT security

As the number of IoT devices increases, also the risk of misuse increases. During the past few years more and more incidents have popped up where different devices have been used as a tool for attacks. (Macaulay 2017, Chapter 1)

There is no single way in which systems are misused and no single way in which systems have been attacked.

In general, especially different consumer products suffer from poor security and in addition, average consumer does not pay attention to security on one's home network. These two facts combined together can cause new security issues and open the doors for attacks.

Because of the nature of the IoT systems and their complexity, there might appear more vulnerabilities. In Table 2 there are ten most potential vulnerabilities named in the different parts of the whole IoT system. (Li & Xu 2017, Chapter 1.1.3)

In Table 2, system is divided into four layers with a slightly different naming. In this case, the Sensing layer is equal to the Perception layer of the Ning's model, the Service layer is equal to the Supporting layer and the Interface layer is equal to the Application layer.

Table 2. Top ten vulnerabilities in IoT system (Li & Xu 2017, Chapter 1.1.3)

Security Concerns	**Interface Layer**	**Service Layer**	**Network Layer**	**Sensing Layer**
Insecure web interface	√	√	√	
Insufficient authentication/authorization	√	√	√	√
Insecure network services		√	√	
Lack of transport encryption		√	√	
Privacy concerns		√	√	√
Insecure Cloud interface	√			
Insecure mobile interface	√		√	√
Insecure security configuration	√	√	√	
Insecure software/firmware	√		√	
Poor physical security			√	√

As end devices and gateways can be located practically anywhere they might be in a risk to be physically accessed in an undesired way and can that way cause a big risk. Also the size and capacity of the devices set a challenge on implementing them in a secure enough way. Proper authentication and authorization are needed to make sure only allowed devices can access the system. (Li & Xu 2017, Chapter 3.1-3.2)

Poor security can be partly explained by small resources in devices but very often it may be a lack of knowledge or just a way to reduce costs. The end device may be the easiest part of the system to get in contact with and poor design can open doors to the entire system. The end device plays a very important role as well as it is the one making measurements or carrying out the given tasks, and thus the security of the end device should be considered much more important.

Network connection is always a risk on an internet service. To secure the network connections sufficient encryption is needed on all connections available. (Li, Xu 2017, Chapter 3.3)

The network connection plays a very important role in IoT, as there is usually no service if there is no network connection. In addition to the network connection, the security of the connection between the gateway and the sensor should be adequate ensured.

The platform and its overall safety is one of the biggest parts in securing the whole service. IoT service is no exception and can be secured as any other internet service. (Li, Xu 2017, Chapter 3.4)

As stated, IoT service does not differ much from the modern internet service on a platform level. The same security requirements can be set for IoT services and for all web-based services. A good architecture from security point of view is presented in pervious chapter 2.1.2 Platform architecture.

One important and timely issue in data security is new EU regulation, GDPR. EU Parliament approved General Data Protection Regulation in May 2016 and it will be in force in May 2018 after a transition period of two years. It is a regulation for protecting EU citizens from privacy and data breaches. GDPR defines what data can be seen as personal data, how it should be handled and what the rights of the data

subject are. It clearly defines that new regulation applies on all companies processing or holding the personal data of data subjects residing in the European Union. (Key Changes with the General Data Protection Regulation 2016)

GDPR affects all IoT services and service providers since all the data generated by IoT devices can be considered personal data and must be treated in the system as such. This sets great demands on the systems of the service providers and may cause a great deal of work on systems as such a requirement cannot be anticipated in the development of the system.

2.2.5 IoT applications

In different countries such as United States, China, Japan and many European Union countries, different actions have started to support the research and the development of IoT applications. (Ning 2013, Chapter 1.4)

IoT applications can be segmented into different technology sectors in many ways, one segmentation model can be seen in Figure 7. (Zhou 2013, Chapter 2.1)

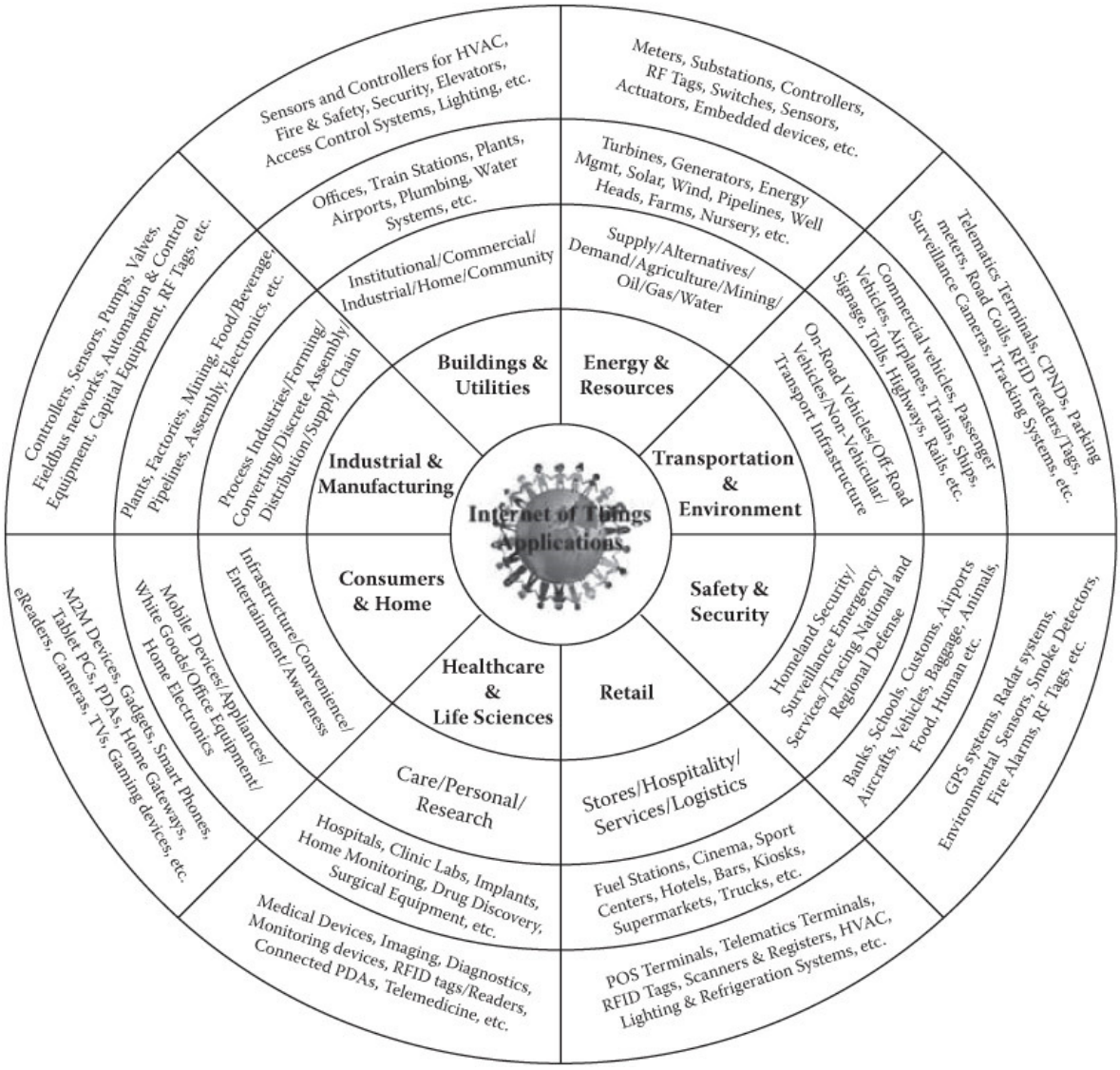

Figure 7. Different IoT application on different technology sectors (Zhou 2013, Chapter 2.1)

There are already a number of services developed for different sectors of technology and society today. Air conditioning and ventilation systems can be monitored and automated. Surveillance systems and automatic fire and burglar alarm systems can be remotely controlled. Asset tracking systems allow one to find out where one's container or parcel is located in real-time. Electricity consumption can be monitored and, if necessary, controlled. Self-service cashiers have become more common. These are just to name a few services already implemented.

2.3 Telia and scope

Telia Company today has a long history in Sweden and Finland from being a government agency to the present Telia. The main part of the Telia was formed in a merge between Telia in Sweden and Sonera in Finland in December 2002 and after that company has acquired operators mostly from other Nordic and Baltic countries. Today the company has activities in Nordic and Baltic countries and also in Eurasia. Since 2016 it has used the name Telia Company. (Markets and Brands | Telia Company 2017; TeliaSonera History 2017)

Telia has a long history as such, however, it also has a long history and a great deal of experience with IoT kind of services. Two good examples of IoT kind of services are Telia AMR and Alerta services.

For over ten years Telia has offered automatic meter reading services for electricity companies and is one of the biggest service providers in that business area in Finland. The system connects daily to every single electricity meter and remotely reads the use of electricity as well as some other parameters and then delivers the results to electricity companies. The old electricity meter generation is remotely manageable; however, it still uses old communication technologies and is very limited in other capabilities than electricity usage metering. These old devices can be seen as the first generation IoT devices. Newer meters are much more capable and can be used or at least expanded to be used also for totally other purposes.

Another example of IoT kind of services is Telia Alerta, alerting and monitoring service that Telia has also been offering for years. The service can connect different kinds of alarm systems such as fire or burglar alarm to itself and then handle and forward the incoming alarm messages based on customer specific rules. Telia's special gateway devices are used to connect to the system.

Both old services are closed systems and are based on old legacy technology. They are running both on their own service platforms which requires own operating personnel. It is also challenging to reuse existing system resources and develop new services on old platforms.

2.3.1 Original scope

Few years ago a new project was started with the idea to renew the platform and move the existing integration and message handling services to one common and modern service platform which also can be used to develop new services. The project was started with evaluating some software providers that can offer that kind of platform installation. One requirement was that the platform can be installed in the company's own servers which limited the alternatives to few. In that phase, platforms from Accenture, PTC, Ericsson, Comarch and Cumulocity were evaluated; however, during the process some were dropped out for different reasons. At first the platforms were evaluated based on technical system documentation and not the live systems.

2.3.2 Changed scope

Soon it became clear that all services that were thought to be migrated cannot be easily moved to a new platform. In addition, some of the services were outsourced during the evaluation and the scope needed to be changed. A new scope for the project was agreed: to implement a new platform which can be used to develop new IoT services and that can be provided as a plain platform for partners and customers. It was agreed to migrate the old existing services partly and in smaller pieces. Cumulocity was chosen as the platform vendor and the development project could be started. The scope of this thesis was restricted to discuss the platform development and its usability as a service platform, which means that no other parts of the services were studied.

2.4 Cumulocity platform

Cumulocity is German based company that has developed their own IoT platform software. The Software AG acquired Cumulocity on March 2017. (Cumulocity | Company 2017)

2.4.1 Overview

Cumulocity can provide a full IoT platform with the base on Inventory, which holds all master data. All actions such as Measurements, Operations and different kind of Events are linked to devices or other assets on Inventory as shown on Figure 8. (Cumulocity | Cumulocity's domain model 2017)

Figure 8. Overview on Cumulocity system (Cumulocity | Cumulocity's domain model 2017)

2.4.2 Interfaces

Cumulocity system uses the same APIs for applications and for device communication, which means all APIs are available for everyone with the same functions both on system applications and devices. HTTP, HTTPS and REST are used for communication as shown in Figure 9. (Cumulocity | Introduction to Cumulocity 2017)

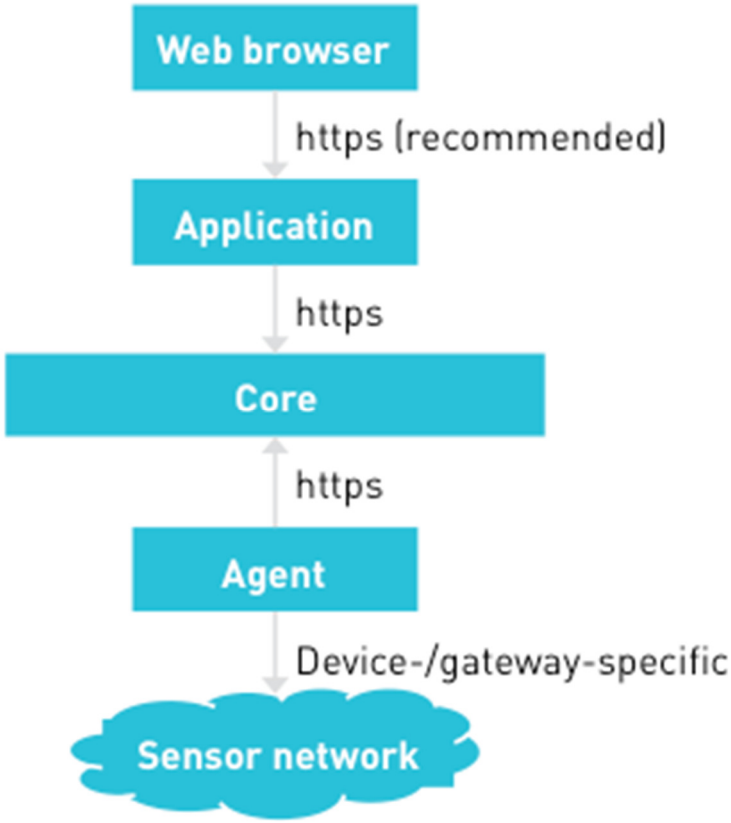

Figure 9. System communication (Cumulocity | Security aspects 2017)

The connection is never opened from the core system but the device always opens connection to the system which means there is no need to open any ports on the device side. (Cumulocity | Security aspects 2017)

Use of same APIs on both device and application communication has advantages and disadvantages. The same APIs make it easier for developers to write software as the same technology can be used in both applications and devices. A clear disadvantage is that all APIs are available on both devices and applications, although they are not needed and the use cannot be restricted.

2.4.3 Agents

As devices and other systems are rarely able to communicate directly with each other, some agent software is usually needed. In practice, there are two kind of agents, server- and device-side agents depending on where the agent software is installed as shown on Figure 10. No matter where the agent is installed, it is

responsible for the communication between Cumulocity platform and the device. (Cumulocity | Interfacing devices. 2017.)

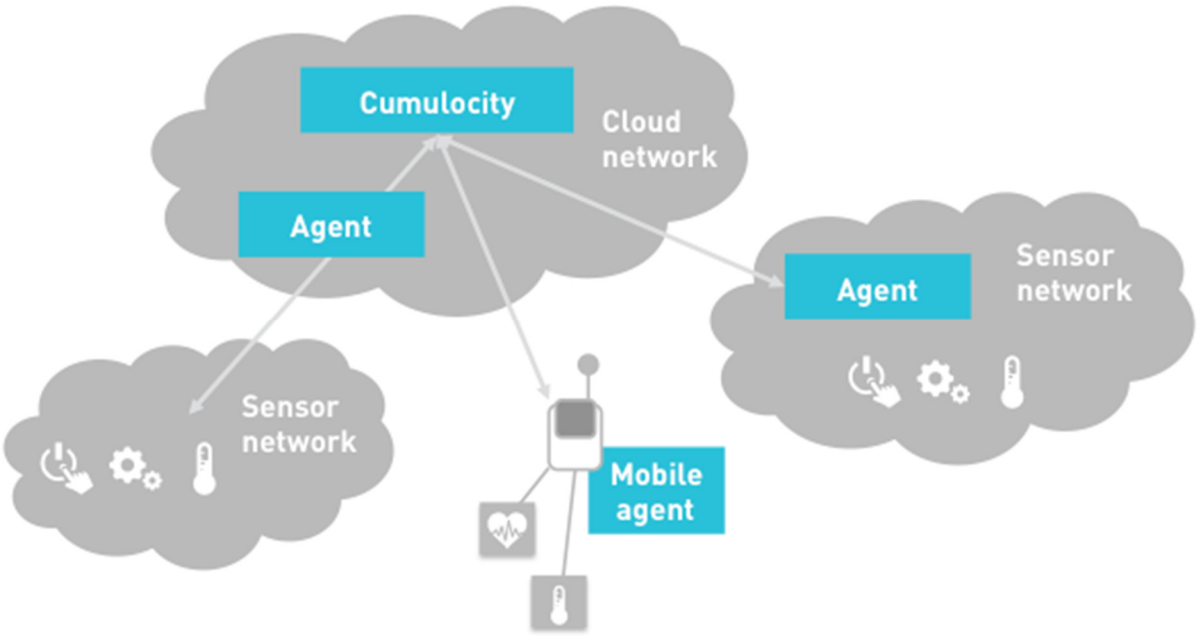

Figure 10. Agent variations (Cumulocity | Interfacing devices. 2017.)

2.4.4 MachNation's 2016 IoT Application Enablement Platform ScoreCard

There are quite a few players on the platform market today with different kind of services on different level of maturity. MachNation has ranked 35 IoT platform vendors with four different rating categories and fourteen sub-requirements. Different platform vendors included in the evaluation are listed in Table 3.

Table 3. MachNation AEP ScoreCard 2016 platform vendors (MachNation's 2016 IoT Application Enablement Platform ScoreCard 2016)

VENDOR	VENDOR	VENDOR	VENDOR
Aeris	Carriots	Gemalto	SAP
Afero	ClearBlade	HPE	Scriptr.io
Altizon	Connio	IBM	SiteWhere
Amazon	Cumulocity	Kaa	Software AG
Amplia	Davra Networks	MachineShop	Telit
AT&T	DevicePilot	Microsoft	TheThings.iO
Ayla Networks	Electric Imp	MODE	Waylay
Bosch	Ericsson	PTC	Yaler
C3 IoT	Exosite	Relayr	

Table 4. MachNation IoT AEP ScoreCard Overall Ratings (MachNation's 2016 IoT Application Enablement Platform ScoreCard 2016)

VENDOR	OVERALL SCORE	FLEXIBLE & SCALABLE DEPLOYMENT MODEL	FOCUS ON THE DEVELOPER PERSONA	OPERATIONAL SOPHISTICATION	WELL-EXECUTED PARTNER STRATEGY
Cumulocity	84	◓	●	◓	○
Vendor 22	77	●	●	○	○
Vendor 1	72	○	●	○	◓
Vendor 18	72	●	○	○	◓
Vendor 14	72	◓	◓	○	○
Vendor 9	71	◓	●	○	○
Vendor 32	71	●	◓	○	●
Vendor 10	71	◓	◓	◒	◓
Vendor 24	70	○	○	◒	◓
Vendor 11	70	◓	○	○	◓
Vendor 34	69	◓	◓	○	○
Vendor 19	69	●	◒	◓	○
Vendor 7	69	◓	○	○	○
Vendor 28	69	○	●	○	○
Vendor 8	68	◓	◓	○	○
Vendor 3	45	○	◓	●	●
Vendor 23	44	◒	◒	○	●
Vendor 15	41	○	●	◒	●
Vendor 35	39	○	○	●	●

● LEADING ◓ ABOVE AVERAGE ○ AVERAGE ◒ BELOW AVERAGE ● TRAILING

Cumulocity platform was ranked on the first place as a leading Application Enablement Platform vendor on 2016 scorecard with three out of four categories valued over average as can be seen on Table 4. (MachNation's 2016 IoT Application Enablement Platform ScoreCard 2016)

The MachNation ScoreCard supports platform vendor evaluation and selection of the Telia's IoT service platform carried out at the beginning of the project. There are

some same players on the list as in the evaluation phase and some new vendors that were not available at the time of the evaluation.

3 Platform development

It was decided to carry out the platform and service development using Agile methods with one-month sprints. Agile methods have usually been used in customer projects with subcontractors; however, not internally in company's development projects so this method of working required some extra attention at the beginning. Each sprint was planned with the whole project team, and the results achieved were evaluated at the end of the sprint.

3.1 First installation

Companys's own platform development and testing started in March 2016 by installing the whole system from scratch with the help from Cumulocity. Before installation, it was already decided to make the installation on company's own public cloud service, which was also in a development phase. The use of that cloud service was a good idea as during the first installation it was discovered that some servers were missing and some were misconfigured. The cloud service made it possible to create new servers in few minutes; when using other internal virtual services that would have required at least some days. There are also some downsides when using a platform under development. This time there were some major problems with network connections, and as the system installation required a working internet connection, it took about a day to make all needed preparations to even start the installation.

3.1.1 Installation

Installation itself was mostly automated and was carried out with special installation automation software without major problems; few servers still needed some special manual configurations; however, that was known beforehand. The first system was up and running in two days from the start and the installation of another environment took one more day when all the problems were already known and

solved on the installation of the first environment. For Telia personnel these first installations were more about learning the system basics and doing without understanding what is done and why.

3.1.2 Network architecture

The first iteration of the system architecture was to we put load balancer alone in own its network towards the internet and all other nodes to their own backend network as they need no access from the internet. The load balancer in this case runs as a software service on one server and is not a special hardware unit. The first system architecture version is shown in Figure 11.

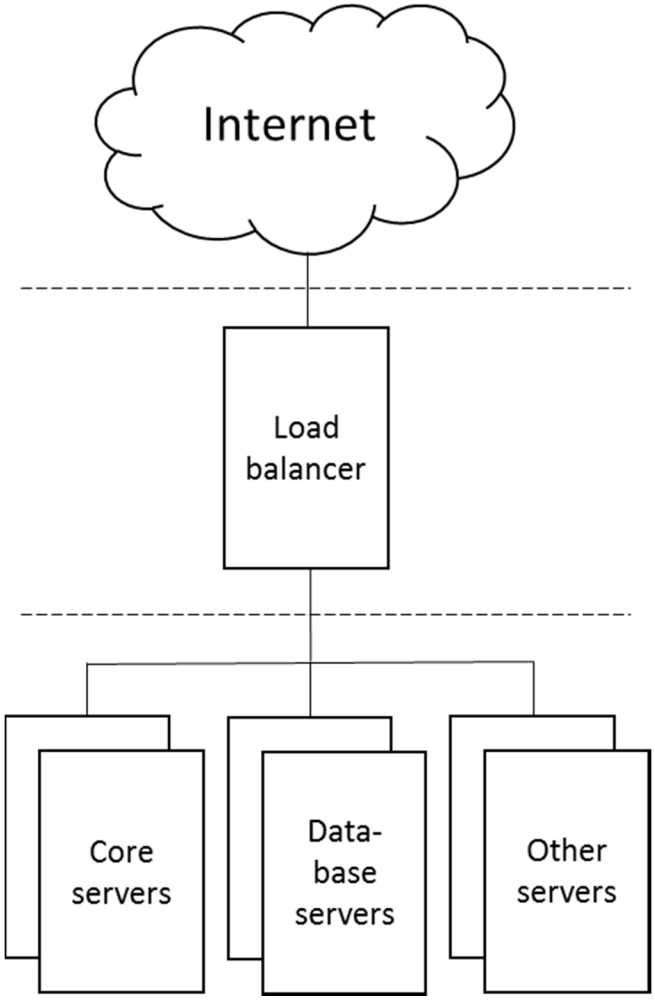

Figure 11. First architecture design

This network architecture caused no problems from Cumulocity point of view and the first installation was executed with that configuration. However, during the installation it was already discussed and agreed that the management access should

not go through the same load balancer server as all production traffic and a special jump server is needed. Later after installations, the jump server was added on the same network as the load balancer and configured to run the proxy software as well.

3.1.3 Studying and testing

After installation working environment was ready and it was possible to start testing and studying the system deeper. Platform operations required more studying because the automation system was unfamiliar to the entire project team. However, understanding the automation system is essential, since most of the maintenance work of the platform is carried out using the automation system.

The device testing was started with some basic devices such as Raspberry Pi to which Cumulocity offers a ready reference agent and also some sensor support.

3.1.4 System Update

Every second week Cumulocity releases a new system version, which is taken in use into their own environments. For customers running their own environments such as Telia it makes no sense to run updates that often. To reduce needed updates Cumulocity releases every fourth version as a bigger update for customers running their own system. That means updated version for Telia roughly every two months. The first system update was executed about a month after the initial installation as a part of operational training Cumulocity arranged for Telia. The system updates are also carried out using installation automation software and the update turned out to be a quite straight forward process with no visible downtime on customer if done in a correct way.

3.1.5 Challenges

As Telia was using a public cloud service, there was no special management network available to access servers. Because of that, the users were forced to access the servers through the internet and the only way was to use the only internet exposed server on the platform, load balancer, to access others. The special jump server was then added to the system to move the management connections away from the load balancer.

As the load balancer was on its own network but also had an interface to backend network, it required some manual configuration to enable it to use multiple network interfaces and to make those work in the needed way.

As the core system servers were on their own backend network with private IP range and no DNS in use, hosts-files had to be created and updated to be able to use some understandable names instead of plain IP addresses.

There was also a need to access the web interfaces on the backend network; however, those should not be exposed on the internet. An extra proxy on load balancer was configured to allow access to web interfaces on the backend servers. This proxy was later moved to the jump server to move all management traffic away from the load balancer.

Update usually means some changes in the core software but also on the automation scripts. If the existing scripts are manually modified for one's own use, all needed changes are in danger to be overwritten if the same modifications are not implemented on new scripts as well.

During testing, it was realized that as the system UI does not use sessions they cannot expire, which is not a real problem but some security issues anyway if one can stay logged in and server never logs one out. This issue has now been recognized and corrective actions are being investigated.

3.2 Second installation

The first environment had been run over the summer, running internal testing, implementing processes and designing the service. Then it was announced that there would be changes on the cloud platform that was used in the project and the easiest way to move to the new platform would be reinstallation. The planning system installation and data migration had to be started as there already were some PoC customers and their data on the system.

3.2.1 New network design

As some downsides on the existing network design had been found out and some more had also been learnt about the used cloud environment, it was decided to change the architecture slightly. As shown in Figure 12, it was decided to move the load balancer inside the same network as the other system core nodes and access restrictions were done with access rules on the cloud service. These changes were seen to ease the management of the servers and also simplify the connections.

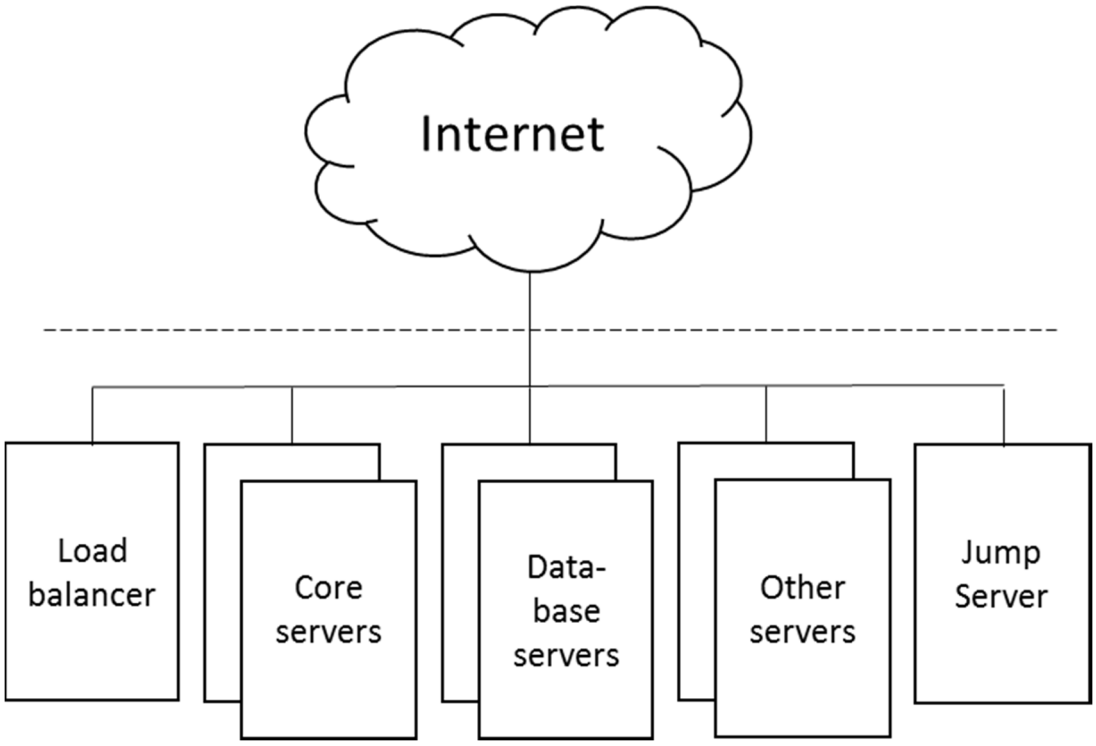

Figure 12. Second architecture design

3.2.2 Installation

The new installation was decided to be carried out by the project team as a test of the readiness for production. The installation process was changed and automated slightly more and no manual work was supposed to be needed after the preliminary server configuration. However, the installation scripts were not totally in order and some manual work was still needed to get the system up and running. The installation itself took a day; however, some more time was needed to make all configurations match the existing system.

3.2.3 Data migration

It was agreed to migrate the data from the old environment by taking full dumps from databases and restoring those on the new environment. The backup restore process of the databases had been under design during testing and everything was in order with the test environment. However, the restore did not go correct at first and needed few retries until the issue was fixed. After that, the backup restore process has been developed further and in future same kind of problems should not occur.

3.2.4 Challenges

During the second installation, the setup scripts provided by Cumulocity did not work as expected and extra manual work was needed. This has later been fixed and should not be an issue in the future.

Database restoration did not work at first, which caused some delay on the system switch over. The backup and restoration processes have been studied since that and this should be in order now.

3.3 Other changes in platform

3.3.1 Virtual server resizing

Currently, the used cloud platform does not support virtual server resizing. During development there have been situations where used servers have been found to be underpowered and need resizing. Lack of resizing feature means that the project team has been unable to add resources such as memory or CPU on the existing server and the server had to be recreated with more resources. Of course, this has been a good real life test for server recovery; however, it feels like unnecessary work as resizing should be a default feature of the virtual platform.

3.3.2 Domain name change

As TeliaSonera Finland Oyj name was changed to Telia Finland Oyj, all old Sonera references needed to be removed everywhere and changed to Telia. That name change also required some extra work on the platform as the system name and

domain had to be changed. The changes were well planned and tested on the test environment and in production everything went as expected. The known challenge on name change was that all devices might not be able to connect to the new address without a restart, which could not to be initiated remotely in all cases.

4 Results

The initial result of the study and development project was to set up a new platform and evaluate its validity to be used as a service platform for new IoT services. At first, the system itself was running well; however, it was not ready to be sold out as Telia service as it only was an empty platform. Useability and achieved results were evaluated with the project team at the sprint planning meetings.

A good example of the cycles in the development process is server architecture. First iteration was planned based on the technical documentation and it was implemented according to the initial plan. During the installation it was already found out that some extra servers are needed. The architecture was deeper observed and agreed that architecture needs to be slightly changed for simplier connections between the servers and better manageability. New changes were planned and architecture changed to simplier. The second iteration was implemented and it was found out to meet the set requirements for the connections and manageability. On follow-up phase it was realized that some new changes are still needed for enhanced security. New architecture was planned again and it is now waiting for implementation. All the challenges during the project were handled according to the same model and some issues were left open for the further development.

The lack of the needed platform features in the first versions has been corrected in multiple version upgrades since the initial installation. The future upgrade versions are also promised to meet many of the required issues that are still open. There has been discussion with some PoC customers that the service cannot provide all data the customers need, which cannot be seen as a problem on the platform itself but a shortage on service implementation.

There have also been some performance challenges on the platform after changes on the cloud platform. These problems are directly related to the cloud platform and they are not seen as a reason not to approve the own service platform solution.

The own device end development has been so slow that the first released services use the devices from 3rd party providers who can offer gateways and a wide variety of sensors for measuring the environmental conditions. A working implementation of the 3rd party system proves that the platform can now be used as a service platform.

From the organization point of view, the service platform has already moved to production and the resposibility has been transferred to the production team. That means all quality and functionality requirements from the organization have been met.

Now the service platform is useable with the needed basic features for the first services to be released, which means that the set target has been reached and the platform itself is ready to be used as service platform. There are known limitations that can be tolerated and permanent solutions on these will be developed in the near future.

As a result, the platform architecture design should still be changed to support multitier architecture as can be seen in Figure 13. All servers are most likely kept in one existing network; however, for enhanced security the access rules on the cloud service will be changed to restrict rules so that access is only allowed for the needed services.

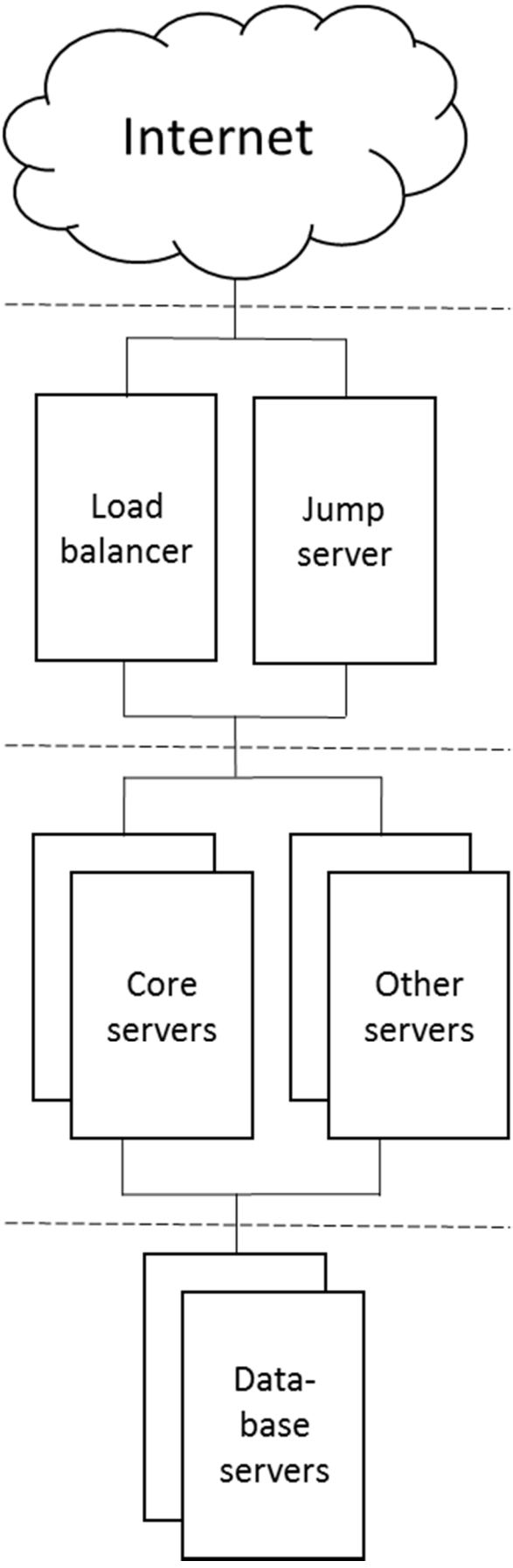

Figure 13. Target architecture design

5 Conclusions

It was found out at a very early phase of the development project that it is very hard and not reasonable to replace internally developed old service platforms with a new of-the-self system and move the existing services there. The project scope was changed and the system itself is now built up and running with some PoC customers and the first new services are just about to be launched in public. From that point of view, the target was reached and the platform can provide the needed features. Now it is just a question of customer needs and how new platform can meet those forthcoming requirements. When looking from customer requirements point of wiev, the platform is never ready, and some development is always needed in technical or process aspects.

During the development and testing some unexpected challenges were faced. Some of them were already solved during the development and some were addressed to the platform vendor as development requests. However, the remaining challenges are not seen to be blocking the service release.

From the UI point of view, it was found out that the default UI serves well for the adminstration use and it can be used internally: however, for the end users there is too much they can see and do and the UI itself is far too complex for daily use. Because of these facts, an own UI application was developed for the first launched services and it is planned to be further developed. On the other hand UI solution for the future services still requires some discussion and a decision on what technology is used.

The system was installed on the own new cloud platform which also was in development phase. That caused some extra work when the cloud platform was changed during the testing phase. From some point of view, that was good as the team was forced to test the installation from scratch as well as the system restore and migration from old environment to new. On the other hand, that extra work caused some delay on the own system testing and development, and it is still under consideration if the service later will be moved on some other platform or not.

As the environment was installed on a cloud service that is designed for external customers, it became clear that the system cannot be fully integrated on internal processes. That caused the need to define the own processes for many issues that normally would have been taken care of by normal internal production processes such as operating system level maintenance, system monitoring and backups.

One major challenge is prsented by the device connections, and there is a need for change in thinking as only the device can open the connection and not the server. This works just on the contrary compared to the old services where the whole system is based on the server being also able to open the connection to the device when needed. On the old services, on the other hand, also devices are managed and it is know exactly what is in the network; however, it will not be like that in the future.

It is an accepted risk that there are still some SPOFs in the system because of the limitations based on the software used. It is also known that the software vendor is working with these SPOFs and most of such challenges will be resolved in the future software releases.

In virtual environments, the system snapshot is often used for taking backups. At the moment this cannot be used in this system because of a mismatch in the cloud platform and virtual server operating system version. This is not a major problem; however, it has forced the team to use traditional database dumps as a backup method. This problem is assumed to be solved in the future when the operating system can be changed to newer one. On the other hand, when snapshots are available, there is still some development needed, as well as testing and creating new processes.

6 Discussion

The whole development project starting from scracth has been very interesting. The technology used is new and there has not been anyone saying how things must be implemented but we have had free hands on developing and setting up the whole system. Now the system is running, the platform itself is useable and meets the

initial needs; however, new development is still needed to keep up with the development in the future.

6.1 Current state

The current cloud environment might not be the correct place to run production as that platform is still under development. Platform is also outside the normal internal production processes such as platform user and access management, monitoring and backups. I think that in the development phase the use of as flexible environment as possible is well reasoned and fastens the problem solving. However, when going to a production environment that is stabile and need no big changes there is no reason to run the system at the same kind of platform as in the development phase. Flexibility is not essential anymore but more inportant is that all internal processes and resources that are available and can support the system maintenance can be used and the chosen platform should not set limitations on that. Now the system can be managed but requires special arrangements and extra work from production personnel compared to the old services.

The development project itself from the very beginning to the first released service has been quite long. The release date was planned to be already earlier but it was delayed due to delays on the cloud platform development. Also, the used resources have caused some delays as all resurces have not been fully useable for developent work.

6.2 Further development

During the development work, some issues were faced and considered less important and left to the further development. Some of these issues have already been evaluated and some are just ideas.

6.2.1 Automation

To make it easier to install possible new instances of the platform as well as gain faster disaster recovery and easier scalability, automation in node deployment should be enhanced. The system installation process is handled by iisntallation

automation software and it is automated; however, some preliminary preparations are needed on the new servers before automation can be used. To get the most benefit from the automation system and to use it best way for all the platform maintenance needs as well, it needs to be studied deeper.

6.2.2 Version management

On every system upgrade the installation scripts change and all internally made modifications are overwritten with the default scripts received from Cumulocity. Some reasonable way to ensure that one's own modifications are taken into use after the upgrade is needed to be agreed upon. Now one have to keep a record of the modifications and in addition the scripts must be checked and fixed manually, which increases the risk of a configuration error.

6.2.3 HA load balancer

There are still some SPOFs in the system; one of them is the load balancer. To assure that the service is available if the load balancer for some reason fails there is need to find a feasible way to implement a duplication of that node.

6.2.4 API management

At the moment, APIs are all open to the internet and there is no easy way to restrict the use to some specific APIs or usage volume in general. With a special API management solution, the use of APIs can be managed outside the base system. This needs some development and integration work in the near future.

6.2.5 Data analysis

Data analysis is one of the main features in an IoT service. At the moment data analysis possibilities are very restricted on the platform and some external data analysis solution is needed. Current services do not require much data analysis and core system meets the requirements. However, in the future when more complicated solutions are developed, core system is not enough anymore.

6.2.6 Device management

From the old experience it has been found out that when the number of devices increases, the importance of the device management solution is increasing. Device management capabilities on the base platform are quite restricted and possibilities do not meet all the future needs. There is some preliminary work already done; however, separate device management solution still needs to be studied.

6.2.7 GDPR

GDPR will be in force in May 2018. This means all systems collecting personal data have to meet requirements before that. In practice, this means at least the validation of the system if there are no necessary changes on the system.

References

Arregoces, M., Portolani M. 2003. *Data Center Fundamentals - Understand Data Center Network Design and Infrastructure Architecture, Including Load Balancing, SSL, and Security.* Indianapolis: Cisco Press.

Collin, J., Saarelainen, A. 2016. *Teollinen internet.*Helsinki: Talentum

Cumulocity | Company. 2017. Page on Cumulocity's website. Accessed on 27 May 2017. Retrieved from http://cumulocity.com/about/

Cumulocity | Cumulocity's domain model. 2017. Page on Cumulocity's website. Accessed on 27 May 2017. Retrieved from http://cumulocity.com/guides/concepts/domain-model/

Cumulocity | Interfacing devices. 2017. Page on Cumulocity's website. Accessed on 27 May 2017. Retrieved from http://cumulocity.com/guides/concepts/interfacing-devices/

Cumulocity | Introduction to Cumulocity. 2017. Page on Cumulocity's website. Accessed on 27 May 2017. Retrieved from http://cumulocity.com/guides/concepts/introduction/

Cumulocity | Security aspects. 2017. Page on Cumulocity's website. Accessed on 27 May 2017. Retrieved from http://cumulocity.com/guides/concepts/security/

Gartner Says 8.4 Billion Connected "Things" Will Be in Use in 2017, Up 31 Percent From 2016. 2017. Page on Gartner's website. Accessed on 30 May 2017. Retrieved from http://www.gartner.com/newsroom/id/3598917

Kananen, J. 2013. *Design Research (Applied Action Researc) as Thesis Report.*Tampere: Suomen Yliopistopaino Oy – Juvenes Print.

Kananen, J. 2015. *Kehittämistutkimuksen kirjoittamisen käytännön opas – Miten kirjoitan kehittämistutkimuksen vaihe vaiheelta. Jyväskylän ammattikorkeakoulu.* Tampere: Suomen Yliopistopaino Oy – Juvenes Print.

Key Changes with the General Data Protection Regulation. 2016. Page on EUGDPR website. Accessd on 5 June 2017. Retrieved from http://www.eugdpr.org/key-changes.html

Li, S., Xu, L.D. 2017. *Securing the Internet of Things.* Cambridge: Elsevier.

Macaulay, T. 2017. *RIoT Control—Understanding and Managing Risks and the Internet of Things.* Cambridge: Elsevier.

MachNation's 2016 IoT Application Enablement Platform ScoreCard. 2016. MachNation.

Markets and Brands - Telia Company. 2017. Page on Telia Company's website. Accessed on 26 May 2017. Retrieved from https://www.teliacompany.com/en/about-the-company/markets-and-brands/.

Ning, H. 2013. *Unit and Ubiquitous Internet of Things*. Boca Raton: CRC Press, Taylor & Francis Group.

Recommendation ITU-T Y.2060. Overview of the Internet of things. Geneva. ITU-T. Approved 15.6.2012. Referenced 31.5.2017. http://www.itu.int/ITU-T/recommendations/rec.aspx?rec=y.2060

Start - TeliaSonera History. 2017. Page on Telia Company's website. Accessed on 26 May 2017. Retrieved from http://www.teliacompanyhistory.com.

Zhou, H. 2013. *The Internet of Things in the Cloud: A Middleware Perspective*. Boca Raton: CRC Press, Taylor & Francis Group.