FACULTY
OF INFORMATION
TECHNOLOGY
CTU IN PRAGUE

ASSIGNMENT OF BACHELOR'S THESIS

Title:	Feasibility Study of Metheologic SW in Environment of Cloud Computing II
Student:	Ondrej Hudcovič
Supervisor:	Ing. Martin Koval, Ph.D.
Study Programme:	Informatics
Study Branch:	Information Systems and Management
Department:	Department of Software Engineering
Validity:	Until the end of summer semester 2018/19

Instructions

The aim of the thesis is to analyze possibilities of implementation of legal metrological SW (MSW) in Cloud Computing (CC) environment. The WELMEC Guide 7.2 (WG 7.2) defines the structure and parameters of MSW in the categories long-term storage, communication, download SW, and SW separation. The output will be the methodology used for the implementation of the MSW according to WG 7.2 requirements in CC.
Follow next steps:
1.Specify the properties of the measuring device according to WG 7.2.
2.Realize an overview of CC platforms and choose a proper provider to implement the thesis. Choose another platform than in a parallel thesis.
3.Design an appropriate application architecture (SW) which represents the measuring device.
4.Analyze possibilities of security and storage for the MSW in CC in according to WG 7.2.
5. Develop a methodology for the implementation of the legal MSW of measuring device for the chosen CC. The methodology must respect all requirements according to WG 7.2.

References

Will be provided by the supervisor.

Ing. Michal Valenta, Ph.D.
Head of Department

doc. RNDr. Ing. Marcel Jiřina, Ph.D.
Dean

Prague February 5, 2018

FACULTY
OF INFORMATION
TECHNOLOGY
CTU IN PRAGUE

Bachelor's thesis

Feasibility Study of Metrologic SW in Environment of Cloud Computing

Ondrej Hudcovič

Department of Software engineering
Supervisor: Ing. Martin Koval, Ph.D.

May 15, 2018

Acknowledgements

I wish to express sincere thanks to my supervisor Ing. Martin Koval, Ph.D. for helping me with writing this thesis and providing valuable information about the topic. I am also grateful to my family and friends, who were very helpful during the time I spent at the university.

Declaration

I hereby declare that the presented thesis is my own work and that I have cited all sources of information in accordance with the Guideline for adhering to ethical principles when elaborating an academic final thesis.

I acknowledge that my thesis is subject to the rights and obligations stipulated by the Act No. 121/2000 Coll., the Copyright Act, as amended, in particular that the Czech Technical University in Prague has the right to conclude a license agreement on the utilization of this thesis as school work under the provisions of Article 60(1) of the Act.

In Prague on May 15, 2018 ………………

Citation of this thesis

Hudcovič, Ondrej. *Feasibility Study of Metrologic SW in Environment of Cloud Computing*. Bachelor's thesis. Czech Technical University in Prague, Faculty of Information Technology, 2018.

Abstrakt

Cieľom tejto práce je analyzovať možnosti implementácie softvéru legálnej metrológie v prostredí Cloud Computing.

Príručka WELMEC Guide 7.2 definuje štruktúru a parametre metrologického softvéru v kategóriách dlhodobého ukladania dát, komunikácie, sťahovania softvéru a softvérovej separácie. Výstupom práce bude prípadová štúdia a metodológia použitá na implementáciu metrologického softvéru v prostredí Cloud Computing podľa požiadavok, ktoré sú spísané v príručke WELMEC Guide 7.2. Na túto prácu bude použitá Cloud Computing platforma Microsoft Azure.

Klíčová slova WELMEC Guide 7.2, Cloud Computing, legálna metrológia, metrológia, Microsoft Azure, Internet vecí

Abstract

The aim of the thesis is to analyze possibilities of implementation of legal metrological software in Cloud Computing environment.

The WELMEC Guide 7.2 defines the structure and parameters of metrological software in the categories long-term storage, communication, download software, and software separation. The output will be a feasibility study and the methodology used for the implementation of the metrological software according to WELMEC Guide 7.2 requirements in Cloud Computing. The Cloud Computing platform used for purposes of this thesis is Microsoft Azure.

Keywords WELMEC Guide 7.2, Cloud Computing, metrology, legal metrology, Microsoft Azure, Internet of Things

Contents

List of Figures

List of Tables

Introduction

Metrology plays a very important role in our lives. Measurement instruments are present in almost every building and, at the moment, data manipulation in this field is impractical. The solution to current issues such as data storage and data access appears to be cloud computing.

Cloud computing is *"a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction"* [11, p. 2]. The main goal of this information technology paradigm is to cut costs and to allow the user to focus on the core of his business without the need to understand technologies behind it.

WELMEC Guide 7.2 *"provides technical guidance for the application of the requirements set by MID [5], especially for software-equipped measuring instruments concerning legal metrology. It addresses all those who are interested in the technical understanding of software-related requirements of the MID"* [20, p. 6].

In this thesis, some possibilities of metrological software implementation in cloud computing in accordance with WELMEC Guide 7.2 are discussed. The first part of the paper consists of facts about metrology and cloud computing in general, as it is important for the reader to have basic knowledge of the topic.

After obtaining fundamental information, the next chapter focuses on legal metrology and its usage according to WELMEC Guide 7.2. This guide includes structure and parameters of metrological software in several categories, such as long-term storage, communication, download software and software separation.

Decision related to platform choice is very important in this topic, so some advantages and disadvantages are offered in case of the solution — Microsoft Azure Cloud Computing Platform & Services — chosen for this thesis in comparison with other CC vendors.

Next, the design and architecture are presented. It is necessary to analyze options when it comes to security and storage for the metrological software in cloud computing according to WELMEC Guide 7.2. Knowing this, the chapter also discusses possible threats associated with cloud computing.

The thesis also contains a methodology of a possible SW CC implementation in accordance with WG 7.2.

Last chapter describes basic economic aspects of the designed solution. Ecological impacts are also considered.

Chapter 1

Metrology

1.1 Overview

BIPM defines metrology as *"the science of measurement, embracing both experimental and theoretical determinations at any level of uncertainty in any field of science and technology"* [22, p. 1]. Metrology in general can be divided into categories:

- scientific,
- industrial,
- legal.

In the next couple of lines, first two categories will be elucidated to the reader. Legal metrology, however, requires separate chapter, as it is main focus of this thesis.

1.2 Scientific and industrial metrology

While scientific metrology focuses on creating and maintaining measurement standards, main purpose of industrial metrology is the application of measurement devices into our lives, industry and academic research. Metrological traceability is defined as *"a property of a measurement result whereby the subject result can be related to a reference through a documented unbroken chain of calibrations, each contributing to the measurement uncertainty"* [10, p. 29].

1.3 Metrology in Czech republic

In Czechia, all metrology system is controlled by MPO. National metrology system, which is a detailed overview of metrological infrastructure in Czech republic, can be seen in the figure below. Its main purpose is to establish uniformity of measurements and measurement devices in particular state, mainly through regulations, laws, etc.

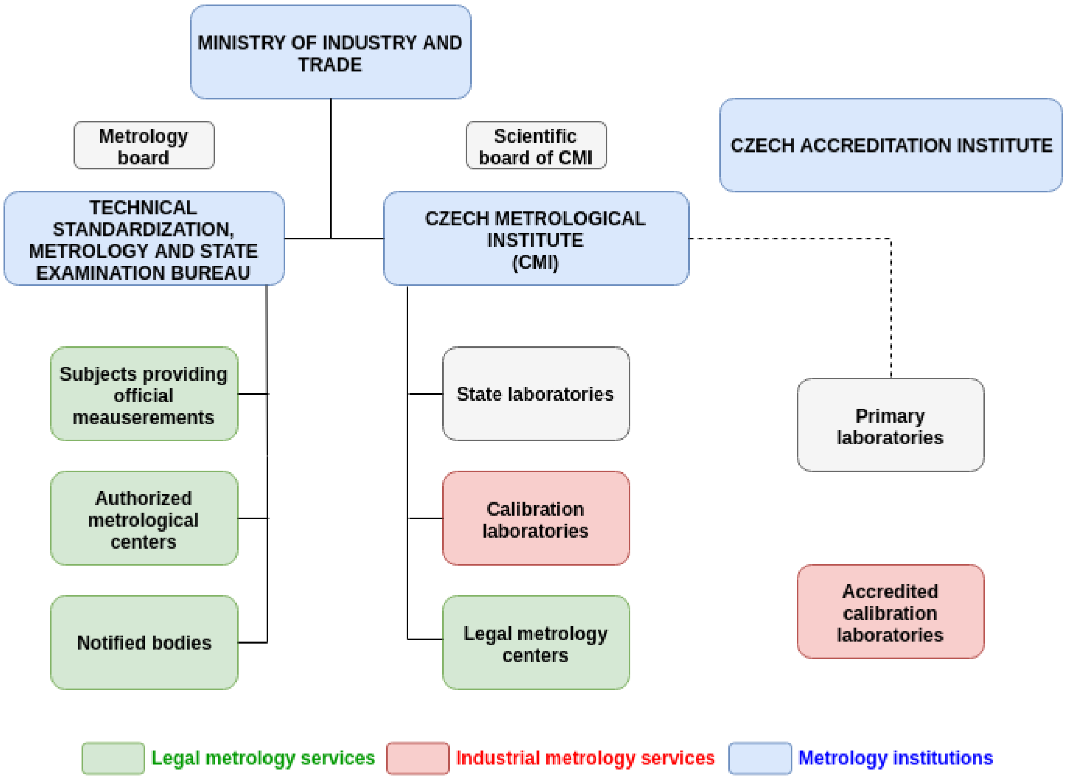

Figure 1.1: Overview of the national metrology system in Czech republic [16]

As mentioned before, the central entity in this system is MPO. Rest of the institutions are directly or indirectly controlled by cooperate with it.

Main function of ÚNMZ is following:

- setting and execution of program of state metrology,
- representing Czech republic in international metrology organizations,
- authorizing subjects to perform tasks in the field of state metrological control of devices and official measurement,
- empowering authorized subjects to keep state metrological standards,
- performing activity inspection of Czech metrology institute,
- monitoring a compliance with the law,
- notifying the authorities of the European Communities of the bodies responsible for the type-approval of measuring instruments and for the verification of measuring instruments.

ČMI represents a national metrology institute. It operates in fields of fundamental metrology, unit transfer and legal metrology. ČMI, together with ÚNMZ, cooperates with their european peers in fullfilling strategic objectives, such as systematic acquisition of information related to the field, division of labor (internationally), participation in the development of metrology regulations, etc. Main purpose of ČMI is:

- performing metrological research,
- keeping state metrological standards,
- providing certification of reference materials,
- performing state metrological inspection of measurement devices,
- authorizing entities that perform reparation of these devices,
- performing state metrological supervision,
- many more.

The last important entity, ČIA, offers services in judging a competence of other entities in the field of metrology. Among these entities are notified bodies, authorized metrology centers, manufacturers of measurement devices, etc. [16] [19]

CHAPTER 2

Legal metrology

Legal metrology is concerned with devices which require some form of verification before their usage due to their impact on financial actions.

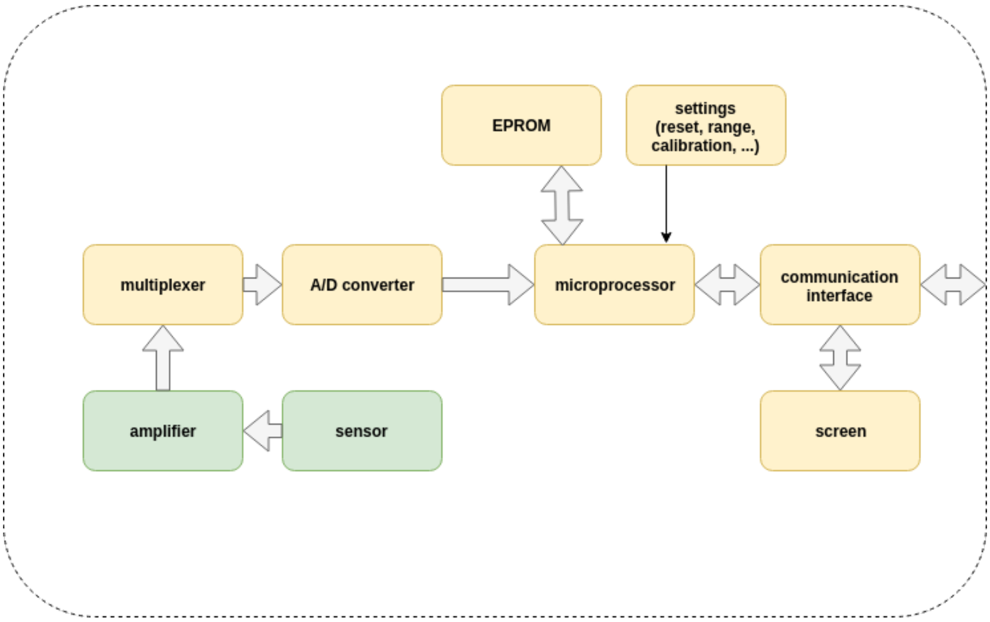

Figure 2.1: Measurement instrument scheme

2.1 WELMEC Guide 7.2

WG 7.2 is considered to be the interpretation of requirements of the MID when it comes to SW in measurement instruments. It defines requirements for several types of devices. These measuring instruments differ in an IT system implemented in them. Also, requirements for extensions relevant to SW and data issues (e.g., data downloading, SW separation) are described. The most important information concerning WG 7.2 is written in the next paragraphs.

2.1.1 Structure

Structure of the WG 7.2 can be generally summarized into three categories. Figure below offers basic overview.

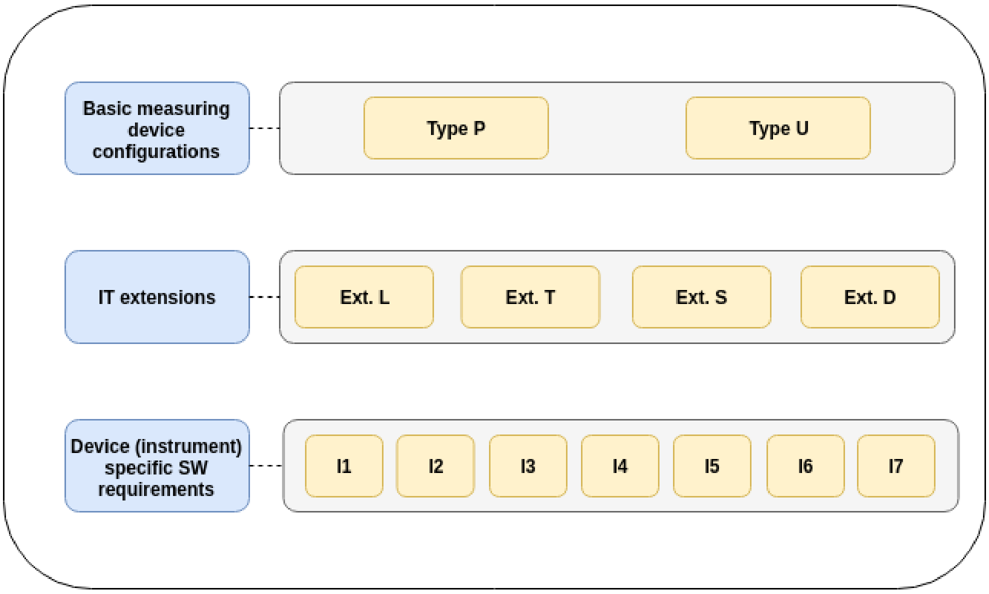

Figure 2.2: WG 7.2 structure

The last category — instrument specific SW requirements — is not relevant for the purposes of the thesis, so basic explanation of the particular instruments is sufficient:

- I1 - Water Meters,
- I2 - Gas Meters and Volume Conversion Devices,
- I3 - Active Electrical Energy Meters,
- I4 - Thermal Energy Meters,
- I5 - Measuring Systems for the Continuous and Dynamic Measurement of Quantities of Liquids Other than Water,
- I6 - Weighing Instruments,
- I7 - Taximeters.

First two categories — device configurations and extensions — are explained in detail in this chapter. WG 7.2 recognizes two types of devices — type P and type U. With that being said, all of the extensions apply to these two types.

2.1.2 Type P device

This type of device is defined as *"a measuring instrument with an embedded IT system (e.g., a microprocessor or microcontroller based system)."* [20, p. 17]. To be considered a type P instrument, its embedded IT system needs to meet requirements below. First of all, SW as a whole, including UI and additional functions, is created specifically for measuring purposes. If the device uses an OS, it can not allow any changes in legally relevant SW and all communication must be part of legally relevant SW. As this is device with embedded system, SW environment as a whole is fixed. SW download is allowed only if specific requirements of extension D are met. [20]

2.1.3 Type U device

Type U means it is a device with universal computer, as opposed to IT system specifically built for particular device (type P). When it comes to HW aspects of this type of device, in addition to universal computer, it usually consists of fixed storage (e.g., HDD) or removable one (e.g., USB). Also, UI of this system may offer functions that are not part of legally relevant SW. Measurement components are usually externally connected to computer via communication link. From SW point of view, an OS is usually used, with optional multitasking. In addition, if some device can not be considered type P device (some requirements are not met), it is considered to be type U device.

As this type of device is universally built, it is less secure. It is therefore necessary to enhance the protection of the SW integrity. If risk classes are considered, low level of conformity is insufficient. Class C is therefore lowest possible one applicable to the type U devices. [20]

2.1.4 Extension D

This extension represents downloading of legally relevant SW. It is used only if SW download is possible without compromising its parts. Two configurations for SW download are considered: SW configuration and HW configuration. In HW configuration, measurement instrument can be, as mentioned before, of both type P and type U. Here, SW can be transmitted either through direct communication link (such as USB, SD card), closed network (e.g. LAN) or via Internet.

SW configuration allows the SW as a whole to be downloaded as legally relevant only, or combination of (separated) legally relevant and legally non-relevant parts. With that being said, legally non-relevant SW is not restricted or inspected in any way. [20]

2.1.5 Extension L

This extension describes requirements for long-term storage of measured data. According to WG 7.2, long-term storage is defined as *" the time from when a measurement is physically completed to the point in time when all processes to be done by the legally relevant software are finished"* [20, p. 34]. There are three ways of storing measuring data: integrated storage, storage for universal computer and removable or remote storage. Description can be seen in the table below. [20]

integrated storage	simple device, not possible to change data, storage is integrated for data or parameters (e.g., RAM, HDD)
storage for universal computer	computer with OS and GUI, removable storage, can be copied into another computer
removable/remote storage	universal computer or simple device, removable storage (USB, flash card, database)

Table 2.1: Technical configurations for long-term storage of measurement data

2.1.6 Extension T

Extension T represents a subsequent information (and requirements) about measuring instruments and it is related to transmission of measurement data via communication networks. However, it only applies when the data is further used for legally relevant purposes. Table below shows relevant information about two different network configurations. [20]

closed network	limited number of devices can be connected, information about these devices (location, functionality, identity) must be known
open network	device with arbitrary functions can be connected, information about these devices may be unknown

Table 2.2: Communication network configurations for transmission of measurement data

2.1.7 Extension S

This extension is related to SW separation. SW separation is, according to WG 7.2, defined as *"an optional design method that allows to separate legally relevant SW from legally non-relevant SW"*. [20, p. 51]

SW in measurement devices is usually complex and consist of both the legally relevant and legally non-relevant parts. Manufacturer then decides if it is necessary to separate these parts or not. There are two way top separate legally relevant SW and legally non-relevant SW.

In the low level SW separation, it is typical to merge SW units *"on the level of the programming language or merging parts of a programme (i.e. subroutines, procedures, functions, classes) to form the legally relevant part of the programme. The rest of the programme is the legally non-relevant part"* [20, p. 51]. High level SW separation, on the other hand, merges *"all parts of the SW to one object that is identifiable by the OS (a programme, a DLL, etc). The rest of the SW is the legally non-relevant part"* [20, p. 51].

Typically, the former option is used in both the type P and the type U devices, while the latter one is used in the type U devices.

Chapter 3

Cloud computing

Cloud computing is identified with five essentials characteristics:

- On-Demand Self-service,
- Broad network access,
- Rapid elasticity,
- Measured service,
- Resource pooling.

On-demand self-service means that CC concept provides its resources on demand of a consumer. This is possible due to the fact some services are fully automated and therefore, from the provider's perspective, do not require human intervention.

Main advantage of CC services is the fact that a consumer does not have to take care of computing power. As a result of this, it is possible for computing power to be located elsewhere. Broad network access is therefore needed.

Rapid elasticity allows a user to scale the space in the cloud. In comparison to older systems, this seems like, from a consumer' side, the space available on the cloud is unlimited.

Another key characteristic of CC systems is measured service, which means the cloud provider can control and monitor the service usage. The provider can then use the data in favor of both himself and the consumer.

Resource pooling is a concept which allows the provider to provide a service to several customer at the same time using multi-tenant model [11].

3.1 Service Models

CC services and cloud infrastructure can be provided in three forms. Software as a Service, Platform as a Service and Infrastructure as a Service.

NIST defines a cloud infrastructure as *the collection of hardware and software that enables the five essential characteristics of cloud computing* [11, p. 2]. This infrastructure can contain both physical layer, which is basically HW essential for the cloud to function and SW built on physical layer called abstraction layer. Below, in Figure 3.1, CC stack layer arrangement is shown, with Saas as the highest level paradigm.

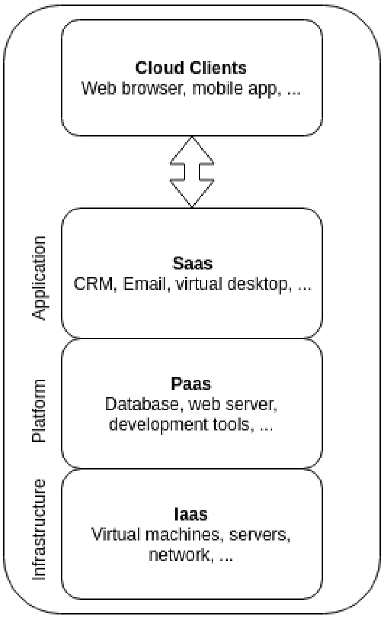

Figure 3.1: Cloud computing service models arranged as layers in a stack

[3]

3.1.1 Infrastructure as a Service

IaaS is lowest-level and probably the most important CC service paradigm of the three. A customer is provided with servers, storage, networking and other low-level services, but is required to build operating environment, on which a final application is deployed.

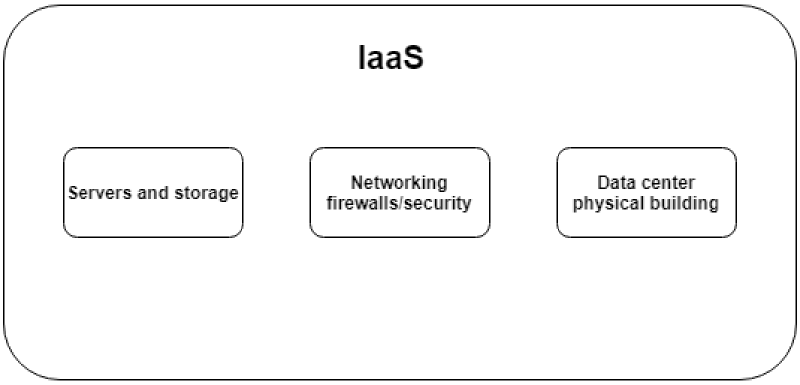

Figure 3.2: IaaS infrastructure

[12]

3.1.2 Platform as a Service

A customer is provided with a cloud infrastructure including operating environment, on which final application is built by him. Therefore, he does not manage the low-level cloud infrastructure which includes storage, operating systems, etc. This form of service is used by developers, as their main task is deploying applications.

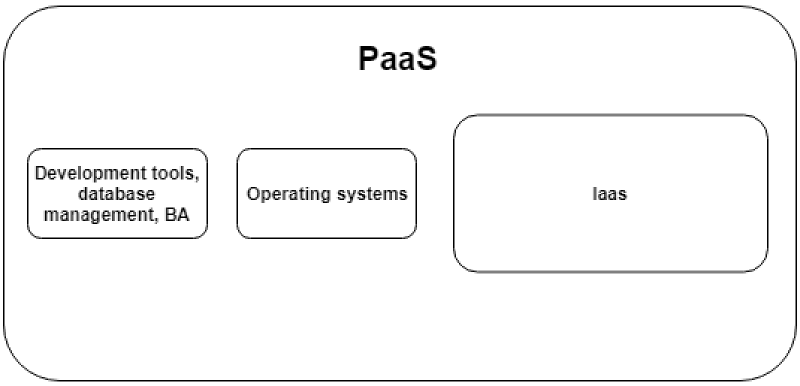

Figure 3.3: PaaS infrastructure

[12]

3.1.3 Software as a Service

Saas represents cloud application services. Operating environment is largely irrelevant, because a consumer is provided with fully functioning application which runs on cloud infrastructure. Application is mostly deployed through web browser, so end user does not need to install and run it on his computer.

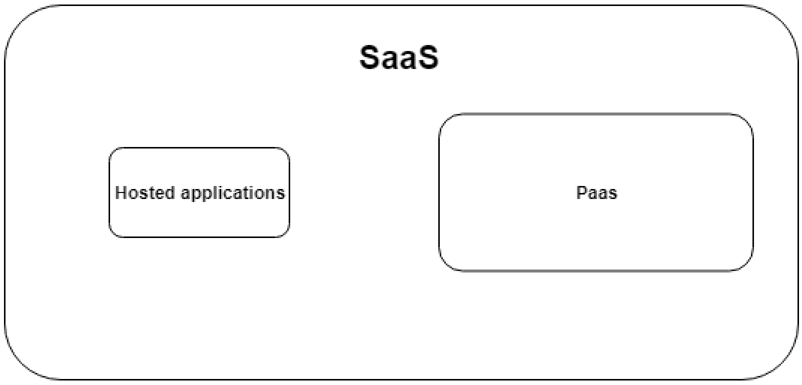

Figure 3.4: SaaS infrastructure

[12]

3.2 Deployment Models

Cloud services are typically divided on the basis of ownership, accessibility, size, etc. These are called cloud deployment models. Each of the models offers different kind of advantages and disadvantages in categories like security, cost effectiveness or scalability.

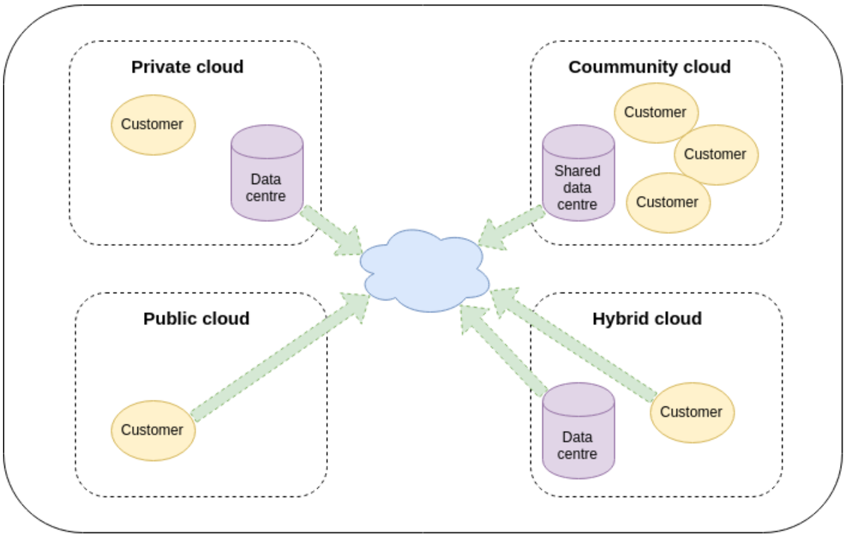

Figure 3.5: Deployment models

[18]

The figure above offers an overview on cloud deployment models and their relationship with the consumer.

3.2.1 Private cloud

Private cloud is usually used by companies for critical tasks, such as data manipulation and data storage. The main advantage is its highly secured nature. On the other hand, this model tends to be more costly. This is because of more advanced security elements.

3.2.2 Community cloud

Community cloud model is usually infrastructure, in which cloud services are shared among institutions with common concerns, usually managed by third-party organizations. Typical example of this are banks or other financial institutions.

3.2.3 Public cloud

This is by far the most popular deployment model. All of the cloud users have the same access and use conditions. Usually, for end users, its usage is free of charge. Most known examples are Facebook or Google.

3.2.4 Hybrid cloud

This model of cloud consists of combination of two or more different cloud deployment models from the three listed above. Some companies use it because sensitive tasks such as company data manipulation can be done through private cloud and other, trivial tasks, through public cloud. Each cloud infrastructure remains separate entity. Main advantage of this model is its cost effectiveness. For example, cloud bursting, is a hybrid cloud deployment model that runs in a private cloud, but, when needed (e.g. company needs more computing resources), can ”burst” into a public cloud. This way, the company pays for additional computing power only when it actually uses it [11].

3.3 CC platforms overview

This overview contains comparison of the three biggest CC vendors: AWS, Microsoft Azure and Google Cloud. For new customers, all three companies offer 12 month trial accounts that contain limited products. These are either free for a limited time period or always. Below are few examples of many more products, that are part of the accounts (all are on monthly basis and free):

product	availability
750 hours of VM for Microsoft Windows Server/Linux	12 months
128 GB of Managed Disks (two 64 GB SSD)	12 months
250 GB of SQL Database	12 months
50 virtual networks free with Azure Virtual Network	always
10 apps with Azure App Service with 1 GB storage	always

Table 3.1: Microsoft Azure

product	availability
750 hours of Amazon EC2 Linux/Windows Server usage	12 months
20 GB of database storage	12 months
15 GB of data transfer out and 1GB of local data transfer	12 months
25 GB + 25 Units of Read/Write Capacity - DynamoDB	always
20,000 free requests with AWS Key Management Service	always

Table 3.2: Amazon Web Services

product	availability
1 TB of querying and 10 GB of storage with BigQuery	always
60 minutes of Google Cloud Speech API	always
1 GB of NoSQL database with Cloud Datastore	always
1,000 units with Cloud Vision API	always
60 minutes with Google Cloud Speech API	always

Table 3.3: Google Cloud

To perform an overview, it is important to divide cloud services into key categories. This topic is written in next paragraphs.

3.3.1 Compute

Microsoft's main compute service is known as Virtual Machines. It offers variety of virtual machines (e.g., Linux, Windows Server, SQL Server), so it is applicable to virtually any computing solution. Main advantage over its competitors is the environment. Majority of enterprises run on Windows, therefore it is more convenient to apply Microsoft's cloud services.

Amazon EC2 is, according to Amazon, *"a web service that provides secure, resizeable compute capacity in the cloud"* [1]. It offers elastic web-scale computing, which allows the consumer to modify the capacity that is needed in the matter of minutes.

Unlike the other two vendors, Google Cloud does not possess as much products in this category. The main attention focuses on Compute Engine, which supports machine customization and flexible pricing for specific needs of the consumer. It is important to point out that this VM is leader in the field of local SSD performance.

AWS	Azure	Google Cloud
Amazon EC2	Virtual Machine	Compute Engine
Elastic Container Service	Azure Container Service	Kubernetes Engine
AWS Lambda	App Service	App Engine
AWS Batch	Batch	Cloud Functions

Table 3.4: Products related to computing

3.3.2 Storage

Azure's basics in terms of storage services contain a wide range of options. These include:

- large-volume cloud services for app communication,
- storage for unstructured data,
- secure cloud file shares,
- scalable storage for VMs,
- cheap storage for rarely accessed data,
- storage for unstructured data.

In this field, Azure offers largely scalable, cost-flexible and secure services, which is its main advantage. For defining storage resources (such as storage type), consumer can use storage account. It is therefore not necessary for him to do it at the service level [6].

Flagship among Amazon's storage services is Amazon S3. It is an *"object storage with a simple web service interface to store and retrieve any amount of data from anywhere on the web"* [14, p. 14]. Amazon guarantees 99.999999999% durability and availability of object up to 99.99% over a year. Security is not an issue, it supports SSL data transfer and automatic data encryption. Advantage of the product is its integration with other Amazon services, such as AWS Lambda, Amazon RDS or Amazon RDS. Amazon also offers some other storage products, out of which those worth mentioning are displayed in the table below.

Google Cloud Storage represents a universal medium with a number of use cases. Customer can store data, use it for analytics, machine learning and also for data backups and archives. Google offers similar approach in storing data — type-specific service — to Microsoft Azure. However, Google Cloud Storage does not have a high level operational layer [8].

AWS	Azure	Google Cloud
Amazon S3	Blob Storage	Cloud Storage
Amazon EBS	Queue Storage	Persistent Disk
Amazon EFS	Disk Storage	Cloud Memorystore
Amazon Glacier	Data Lake Store	Cloud Spanner

Table 3.5: Products related to storage

3.3.3 Networking

Microsoft's Virtual Network allows a user to fully control everything related to networks, from choosing IP addresses to general network configuration. It provides isolated virtual network, same as Google Cloud Platform.

Majority of AWS products are dependant on EC2 and when it comes to networking, it is not different. Primary networking product is Amazon VPC, which is a networking stack. Its easily customizable, for example, it allows a customer to create and connect separate subnets, one being accessible online, the other one without an internet access.

Google chose another point of view. For them, networking is a separate feature, which expands to all other services. In contrast with the other two companies, Google also offers a networking product called Shared VPC, which allows multiple projects to use one common virtual network with shared resources [7].Most important products in this category are, again, displayed below.

AWS	Azure	Google Cloud
Amazon VPC	Virtual Network	Virtual Private Cloud
Amazon CloudFront	Azure DNS	Cloud Load Balancing
Amazon Route 53	ExpressRoute	Cloud CDN
Elastic Load Balancing	Load Balancer	Cloud Interconnect

Table 3.6: Products related to networking

3.3.4 Databases

In terms of database options, Microsoft is a bit ahead of the other vendors, offering multiple SQL-based possibilities. Other crucial products and services are mentioned below. Also, it is important to mention Microsoft is the only one to offer backup and archive services.

Out of Amazon's database options, Aurora is the most important. It relatively cost-effective, fast and highly available (up to 99.99%) database engine, which is compatible with both MySQL and PostgreSQL. Amazon offers also NoSQL and relational database services, some of which are mentioned in the table below. Main missing link here is, in contrast with Azure, the lack of backup services.

In this category, Google lacks the extensivity both AWS and Azure possess. Customer is offered with couple SQL and NoSQL services and also mission-critical relational database (Cloud Spanner). As mentioned above, Google does not offer neither of backup and archive services [9].

AWS	Azure	Google Cloud
Amazon Aurora	Azure Cosmos DB	Cloud MySQL
Amazon RDS	Azure Database for MySQL	Cloud PostgreSQL
Amazon DynamoDB	SQL Server on Virtual Machines	Cloud Datastore
Amazon ElastiCache	SQL DWH	Cloud Bigtable

Table 3.7: Products related to databases

3.3.5 Provider choice

All three companies offer also other products related to developer and management tools, AI, machine learning, etc. However, the market focuses mainly on the four categories mentioned above and it is the best way to compare the competitors. Therefore, the thesis contains only description of compute, database, networking and storage services.

Amazon is still a leader of CC field, which is its major advantage on the market right now. It offers vast range of options and is more than suitable for large enterprises. However, it is not as cost-flexible as competitors, mainly Microsoft Azure. Google does not possess with as many different services as AWS or Azure and it focuses mainly on machine learning and big data. This is not the main focus of metrology. The solution will be based on the concept of Internet of Things and in this category, Microsoft Azure is a bit ahead of its competitors. It offers its platform called IoT Suite, which gathers other services for data storage, data analysis and data visualization. Particular tools will be mentioned in the next chapter.

When it comes to measuring instruments in legal metrology, Microsoft Azure appears to be the better solution of the three and, for purposes of the thesis, it is chosen as a CC vendor.

Chapter 4

SW design and architecture

4.1 Currently possible design

The solution assumes the use of VPN as an intermediary between a device and the cloud. The system uses smart devices that are connected to the Azure Cloud infrastructure (via VPN). In this case, a smart device is a measuring instrument which is connected to the cloud and it meets following conditions:

- real-time capturing of consumption (e.g., water, electricity),
- long-distance control of the device is possible (for example enabling/disabling energy supply),
- long-distance display of measured data and device status is possible.

The figure below displays a possible solution in the current situation of legal metrology while using Microsoft Azure services.

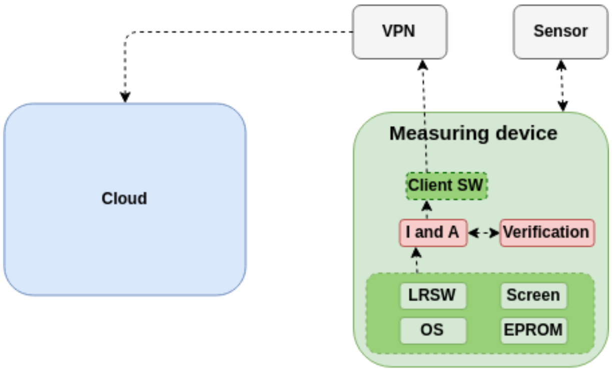

Figure 4.1: Design of a possible solution in current legal metrology situation

[17] [13]

Measuring instruments in this architecture, as mentioned before, need to undergo some changes in comparison with the regular ones. Vital parts, such as sensor, legally relevant SW or operating system, are still part of a measuring instrument.

On top of the three conditions above the Figure 4.1, there are other aspects that are necessary to be added. In contrast with a regular measuring instrument, measured data needs to be verified for its integrity and authenticity (tab 'I and A' in the Figure 4.1) before going any further. After verification, the information is sent from Client SW (which is also part of the smart device) via VPN to data collection and processing service, which is a part of the Azure Infrastructure. Azure Infrastructure is basically built from several products that Microsoft offers and it is called Azure IoT Suite.

The best answer for storage purposes is Azure Cosmos DB. It is a NoSQL key-value store for semi-structured data. This is ideal for flexible datasets, such as data from measuring instruments. It is highly scalable, available and very suitable for Iot solutions.

The highest level of this architecture would be user interface and administrator interface. User interface communicates with the table storage and displays its contents. It contains all the relevant data obtained from a measuring instrument. The customer can check his consumption online without the need to physically access the measuring instrument.

In case of metrology, it is important to separate the two interfaces, because the user has to have restricted access to this information in comparison with the administrator (i.e. water or energy provider).

[2]

4.2 Final design

For this part of the thesis, the measuring instrument will be based on type U device from WG 7.2. It will be built as a smart device in the Figure 4.3, which is part of IoT platform. However, its base is — same as type P device — a built-for-purpose microcontroller, which is specifically programmed only to send telemetry data to the cloud. Of course, applying CC concept automatically means that universal computer will be part of the architecture. WG 7.2 claims, that *" a type U system shall also be assumed if the characteristics of a type P instrument are not completely fulfilled."* [20, p. 24]. In this case, parts of the system, i.e. GUI, are not *"dedicated to the measuring purpose"* [20, p. 17], which implies that this is a type U device. Vital parts of the instrument have to stay, for example EPROM. In case of blackout, the instrument needs to be able to store the measured data.

To sum up the information about the new measuring instrument, it is capable of measuring, temporarily storing telemetry data and then sending it through VPN to the Azure Infrastructure.

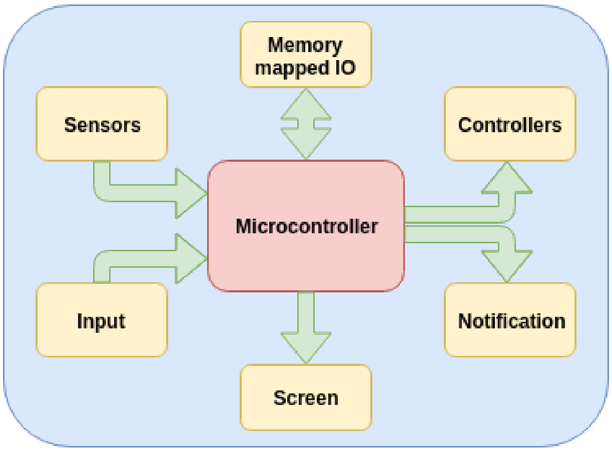

Figure 4.2: Smart device with an embedded system

[15]

The previous design allowed verification of data in the device. This part can be easily eliminated with the verification process shifting to the Data collection and processing services. Figure 4.3 assumes usage of a measuring instrument from Figure 4.2. The biggest change is missing GUI on the device. It is no longer necessary, as it will be accessible via computer.

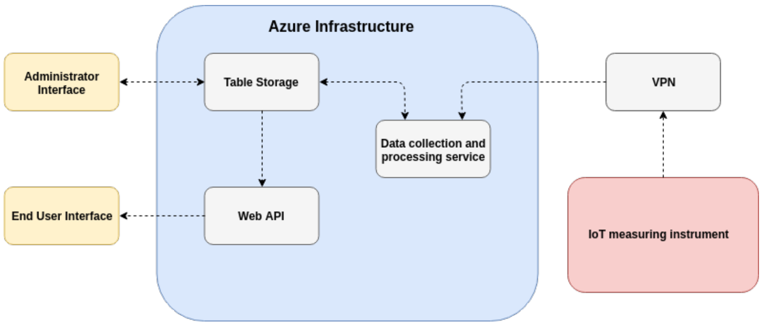

Figure 4.3: Final design

[13]

Figure 4.4 displays an example of a data collection and processing service. IoT Hub is the first element of this scheme. It behaves as a gateway that establishes connection with smart devices. It is also responsible for authentication of said devices by securing both device-to-cloud and cloud-to-device messaging, thus being an important part when it comes to security.

Another part of data collection and processing is Azure Event Hub. It is a streaming platform, which is able to process a large amount of events per second. In this case, Event Hub obtains telemetry data, which is later processed, transformed and stored in the database.

For data transformation and analysis, the solution uses Azure Stream Analytics. This analytics service retrieves archived data from measuring instruments and allows a real-time analysis and processing of the data. The data is subsequently sent to the database. This intermediary is a necessity, because a raw measurement data retrieved from a measurement instrument can contain redundancies or other anomalies.

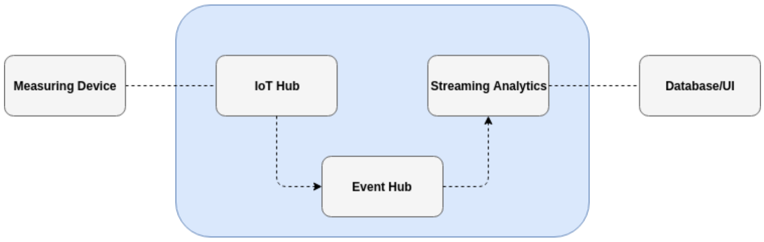

Figure 4.4: Data collection and processing scheme

The design could be extended with addition of Machine Learning tools that Azure offers in its Azure Iot Suite. This would introduce additional data processing methods, such as data clustering, data mining or data forecasting, which would further improve these services. This way, the user has endless possibilities to further process and analyse his data.

4.3 Possible threats

Threats related to the solution can be divided into two points of view:

- Iot threats,
- CC threats.

4.3.1 CC point of view

Couple of problems can occur, although Microsoft Azure (along with other CC vendors) offers solutions for majority of them. For example, data loss can be prevented by using Azure Backup service. It can still be considered a threat, though. However, current measuring instruments are not very reliable either. The only way to ensure security of legally relevant SW is to set a metal seal around the instrument to prevent it from being misused. This is very easily removable and replaceable, so compromising a legally relevant SW or measured data is not entirely impossible.

From the Azure point of view, there is always a probability of employee failure or deliberate malicious activity. On the other hand, from the outside, DDoS attacks are the most popular ways to prevent users to access the network.

Azure, however, mitigates this problem by centralizing data gathering and analysis, using layer protection with Azure Application Gateway Web Application Firewall and continuously evolving to prevent new types of these attacks. The last relevant threat can be insecurity of APIs. By definition, API is accessed externally, which leaves a room for vulnerabilities. Azure also offers a solution in this category — Azure Multi-Factor Authentication. The most used method is Mobile app verification code. A 6-digit verification code is displayed on a smart phone, changing every 30 seconds. When signing in his profile, user enters the code displayed on the phone. Currently, there does not exist an algorithm that can break this protection in such a short time. [4]

4.3.2 IoT point of view

Main concern in this case is a large amount and variety of devices connected to the network. It is therefore harder to retain the integrity of the network without compromising it. Such an infrastructure usually uses DNS, which is quite an imperfect naming system. It should be at least replaced with Domain Name Service Security Extension, which ensures integrity and authenticity of records while also carrying cryptographic public keys. In the best case scenario, because both aforementioned systems are vulnerable to several types of attacks, new naming system should be introduced to the field of IoT. One example seems to be Named Data Networking, which has, as opposed to the other two naming systems, data-centric security. This means every data packet has to be encrypted before it is sent. Current system assumes security is assured at particular endpoints, not in data itself.

Privacy of the user is another problem, because concept of IoT is based on an every day usage of smart devices. This means collecting sensitive data and potentially misusing it. The challenges of privacy issues can be divided into two categories.

- data collection,
- data anonymization.

The main purpose of the former category is to restrict the amount and type of data that is stored.

On the other hand, for data anonymization, it is necessary to ensure cryptographic protection and to remove direct relations between the user and data itself. Main challenge in cryptography protection in IoT is variety of devices that are part of it. Preferably, different approaches can be used for different devices. Making the relations between data and user anonymous can be done by the use of homomorphic encryption. It can be used for computation of encrypted data without having information about unencrypted data (in this case, data-user relations). [23]

The figure below describes an example of homomorphic encryption used in this solution.

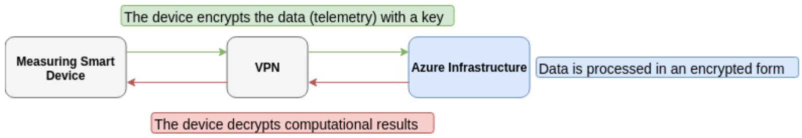

Figure 4.5: Example of a homomorphic encryption

To conclude, the solution brings several disadvantages when it comes to security. On the other hand, as mentioned before, at present, there are also some possibilities to compromise telemetry data.

Chapter 5

Methodology

This chapter suggests a methodology of SW implementation for CC and IoT solutions. In the previous chapter, it was stated that the solution demands the usage of a universal computer. From now on, specific SW requirements are assumed for the type U device only. Tables 5.1 to 5.5 consist of individual requirements set by WG 7.2 for the type U device (with additional extensions) and the information whether it is possible to apply these requirements for the new measuring architecture.

5.1 Device requirements

requirement	description	conditions
U1	documentation	met
U2	SW identification	met
U3	UI influence	not met
U4	communication interface influence	met
U5	unintentional changes protection	met
U6	intentional changes protection	met
U7	parameter protection	met
U8	measurement data presentation	partially met
U9	influence of other SW	met

Table 5.1: Specific requirements for the new device

Documentation requirements are quite straight-forward. Inclusion of SW and HW components' descriptions of an instrument is necessary. In general, this category does not need a change, WG 7.2 covers all the important issues.

SW identification must be *"permanently presented by the instrument, presented on command or during operation"*. This is achievable simply by displaying it in the user and admin interface.

In addition, the measuring device must also have its own identification. Otherwise, changes in comparison to WG 7.2 are not necessary.

According to the solution and WG 7.2, measuring instrument does not have to have a UI. This is a problem, because MID states that whether or not *"a measuring instrument intended for utility measurement purposes can be remotely read it shall in any case be fitted with a metrologically controlled display accessible without tools to the consumer. The reading of this display is the measurement result that serves as the basis for the price to pay"* [5, L 96/175]. This implies that a change in the directive is needed that does not require a measuring instrument to have a screen. Other than that, categories U3 and U4 do not need to change, because the cloud architecture would not allow an influence of a LRSW via user or communication interface.

Categories U5 and U6 are a similar case, protection against changes in LRSW is provided by Microsoft Azure architecture.

Parameter protection is also shifted to the cloud, as device-specific parameter is *"legally relevant parameter with a value that depends on the individual instrument"* [20, p. 31]. That means it is part of a LRSW, therefore protection is provided by Microsoft Azure architecture.

Measurement data is presented via user or administrator interface. Authenticity of the data is guaranteed by Azure architecture provided physical measurements are correct. An addition of remote checking of physical parts of the instrument is recommended.

Finally, protection against an influence of other SW is implicit within Azure policy.

5.2 SW requirements for long-term storage

requirement	description	conditions
L1	completeness of measurement data stored	met
L2	unintentional changes protection	met
L3	integrity of data	met
L4	authenticity of measurement data stored	met
L5	confidentiality of keys	met
L6	stored data retrieval/verification/indication	met
L7	automatic storing	met
L8	storage capacity and continuity	met

Table 5.2: Specific requirements for long-term data storage

This section does not require any changes in comparison to the requirements set by WG 7.2. Azure provides a secure, long-term storage of data. Integrity and authenticity of data is secured. Measured data can be accessed by both the user and the provider, which meets the criteria of category L6 and, in case of long-term storage, L8 too.

5.3 SW requirements for data transmission

requirement	description	conditions
T1	transmitted data completeness	met
T2	unintentional changes protection	met
T3	data integrity	met
T4	authenticity of transmitted data	met
T5	confidentiality of keys	met
T6	handling of corrupted data	met
T7	transmission delay	met
T8	transmission services availability	met

Table 5.3: Specific requirements for data transmission

Data has to be protected against loss, intentional and unintentional changes or any other malicious activities thanks to the fact that transmission is executed through VPN. Again, provided data is physically measured correctly (see measurement data presentation in the section 5.1), it is secure, as all communication is homomorphically encrypted.

Protection against issues related to categories T6 and T7 is easily deployed as part of the cloud infrastructure, because Azure also offers these kinds of services. An addition of filter is recommended. The filter would recognize corrupted and not corrupted data, so that only relevant data is taken into account.

Category T8 — availability of transmission services — forbids any data loss. The measuring instrument shall have reasonably-sized storage to prevent data loss from happening. The solution takes this fact into account.

5.4 Requirements for SW separation

requirement	description	conditions
S1	realisation of SW separation	met
S2	mixed indication	met
S3	protective SW interface	met

Table 5.4: Specific requirements for SW separation

Similar to previous section, SW separation is fully provided by the cloud architecture. Both user and admin interfaces (which can be considered a legally non-relevant SW) are separated from the section responsible for data transmission, processing and storing.

5.5 SW requirements for LRSW download

requirement	description	conditions
D1	download mechanism	met
D2	transmitted SW authentication	met
D3	downloaded SW integrity	met
D4	traceability of LRSW download	met

Table 5.5: Specific requirements for download of LRSW

The only addition to existing requirements is the ability to remote updating and download of LRSW. Measuring instrument can be accessed remotely by notified body or SW manufacturer. [20]

Chapter 6

Economic aspects summary

The designed architecture brings several economic advantages.

Firstly, current measuring instruments are quite big and therefore have a bit high production costs. With reduction of some components, the device and its production costs would shrink significantly, as the only parts necessary to stay are a screen and a sensor.

Another important issue to discuss is the case of an update of LRSW. Nowadays, it is necessary for notified body to physically locate the instrument and then update the SW. It is very impractical and it comes with a great deal of slow administration processes.

The solution should be considered within the concept of smart cities. This would have huge economic and also ecological impacts. All the data related to consumption, be it water or energy, would be stored in database. This would allow predictions on account of how much resources is necessary to deliver for particular area in the city. That means more economically advanced attitude towards a consumption of all the temporary resources used in households.

When it comes to financial aspects, specific difference in price is hard to calculate. Lot of variables come to equation, such as production costs of a measuring instrument, consumption reduction and cloud pricing. Without vital parameters, for example area size, for which the solution would be applied, it is impossible to make a specific cost prediction.

Conclusion

The goals of this thesis were:

- specify the properties of the measuring device according to WG 7.2,
- perform an overview of CC platforms and choose a proper provider to implement the thesis,
- design an appropriate application architecture (SW) which represents the measuring device,
- analyse possibilities of security and storage for the metrological SW in CC in according to WG 7.2,
- develop a methodology for the implementation of the legal metrological SW of measuring device for the chosen CC while respecting all requirements according to WG 7.2.

The reader is now familiar with WG 7.2 and its requirements for measuring instruments. Also, current metrology hierarchy and situation has been introduced. After defining and describing Cloud Computing as a concept, an overview of the biggest CC platforms has been performed. The chapter comprises of key products comparison. Out of the three discussed vendors, Microsoft Azure has been chosen to complete the design because of its advanced IoT solutions.

Next chapter contains the aforementioned solution. It was necessary to introduce smart devices to the architecture, as all the communication between measuring instruments and the cloud has to be remote.

Upon displaying the offered solution, the thesis describes the most recurring threats related to both Cloud Computing and Internet of Things along with probable negatives in current state. Majority of the threats already have a solution, therefore the counteractions that Azure offers were also discussed in the chapter.

Next in order is a developed methodology for SW implementation related to CC and IoT. The thesis follows the requirements set by WG 7.2 and it can be used as an instruction guide within this topic.

At the end, there is a brief economic section. It points out financial and ecological advantages of the cloud solution in comparison with current situation in legal metrology and consequences of such a solution.

Bibliography

[1] Amazon EC2. In: Amazon Web Services [online]. [cit. 2018-04-15]. Available from: https://aws.amazon.com/ec2/?hp=tile&so-exp=below

[2] Build your Internet of Things Solution with Azure IoT Suite. In: *Saviant* [online]. [cit. 2018-05-05]. Available from: http://www.saviantconsulting.com/blog/build-your-IoT-solution-with-azure-iot-suite.aspx

[3] Cloud computing service models arranged as layers in a stack. *Wikipedia, the free encyclopedia* [online]. [cit. 2018-03-27]. Available from: https://en.wikipedia.org/wiki/Cloud_computing#Service_models

[4] C. M. R. da Silva et al., Systematic Mapping Study On Security Threats in Cloud Computing (IJCSIS) International Journal of Computer Science and Information Security,Vol. 11, No. 3, March 2013 [cit. 2018-04-18]. Available from https://arxiv.org/ftp/arxiv/papers/1303/1303.6782.pdf

[5] Directive 2004/32/EU of the European Parliament and of the Council of 26 February 2014 on the harmonisation of the laws of the Member States relating to the making available on the market of measuring instruments (recast). Official Journal of the European Union L 96/149, 29.3.2014 [cit. 2018-04-19]. Available from: https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32014L0032&from=EN

[6] Find the storage product you need. In: Microsoft Azure [online]. [cit. 2018-04-16]. Available from: https://azure.microsoft.com/en-us/product-categories/storage/

[7] Google Cloud Platform for Azure Professionals: Networking. In: Google Cloud [online]. July 18, 2017 [cit. 2018-04-15]. Available from: https://cloud.google.com/docs/compare/azure/networking#virtual_network

[8] Google Cloud Platform for Azure Professionals: Storage. In: Google Cloud [online]. March 15, 2018 [cit. 2018-04-16]. Available from: https://cloud.google.com/docs/compare/azure/storage#service_model_comparison

[9] HARVEY, Cynthia, PATRIZIO, Andy. AWS vs. Azure vs. Google: Cloud Comparison. In: *Datamotion* [online]. December 20, 2017 [cit. 2018-04-10]. Available from: https://www.datamation.com/cloud-computing/aws-vs.-azure-vs.-google-cloud-comparison.html

[10] *Metrological traceability* [online]. Bureau International des Poids et Mesures. [cit. 2018-03-27]. Available from: https://www.bipm.org/en/bipm-services/calibrations/traceability.html

[11] MELL, Peter, et al. The NIST definition of cloud computing. In: National Institute of Standards and Technology [online]. 2011 [cit. 2018-04-12]. Available from: https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-145.pdf

[12] Microsoft Azure. What is PaaS?. [cit. 2018-03-28]. Available from: https://azure.microsoft.com/en-us/overview/what-is-paas/

[13] OPPERMANN, Alexander; SEIFERT, Jean-Pierre; THIEL, Florian. Secure Cloud Reference Architectures for Measuring Instruments under Legal Control. In: CLOSER (1) [online]. 2016 [cit. 2018-04-02]. p. 289-294. Available from: https://pdfs.semanticscholar.org/49bb/eb20a7b68a0b82e852d8f7c894eb31825036.pdf

[14] Overview of Amazon Web Services: AWS Whitepaper. In: Amazon Web Services [online]. April 2017 [cit. 2018-04-15]. Available from: https://docs.aws.amazon.com/aws-technical-content/latest/aws-overview/aws-overview.pdf?icmpid=link_from_whitepapers_page

[15] Smart device with an embedded system. In: *Slideshare* [online]. November 17, 2015 [cit. 2018-05-13]. Available from: https://www.slideshare.net/scato18/iot-and-management-systems-new-dimensions-for-research-and-didactics-towards-sustainability-55495201

[16] Struktura NMS ČR. In: *Úřad pro technickou normalizaci, metrologii a státní zkušebnictví* [online]. [cit. 2018-04-27]. Available from: http://www.unmz.cz/urad/narodni-metrologicky-system-ceske-republiky

[17] THIEL, F., et al. Cloud Computing in Legal Metrology [online]. In: 17th International Congress of Metrology. EDP Sciences, 2015 [cit. 2018-04-14]. p. 16001. Available from: https://cfmetrologie.edpsciences.org/articles/metrology/pdf/2015/01/metrology_metr2015_16001.pdf

[18] VOLD, Nils. Cloud basics – Deployment models. In: *VISMA, Corporate Blog* [online]. March 12, 2012 [cit. 2018-04-06]. Available from: https://www.visma.com/blog/cloud-basics-deployment-models/

[19] Vše o ČMI. In: Český metrologický institut [online]. [cit. 2018-04-27]. Available from: https://www.cmi.cz/vse_o_cmi

[20] WELMEC, Guide. 7.2: Software Guide (Measuring Instruments Directive 2014/32/EU). 2015. Available from: https://www.welmec.org/fileadmin/user_files/publications/WG_07/Guide_7.2_2015__Software.pdf

[21] *What is metrology?* [online]. Bureau International des Poids et Mesures. [cit. 2018-03-27]. Available from: https://www.bipm.org/en/worldwide-metrology/

[22] Worldwide metrology [online]. In: Bureau International des Poids et Mesures. [cit. 2018-03-28]. Available from: https://www.bipm.org/en/worldwide-metrology/

[23] ZHANG, Zhi-Kai, et al. IoT Security: Ongoing Challenges and Research Opportunities. In: IEEE Xplore, Digital Library [online]. 2014 [cit. 2018-05-06]. Available from: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6978614

Appendix **A**

Acronyms

API Application programming interface

AWS Amazon Web Services

BIPM The International Bureau of Weights and Measures

CC Cloud Computing

CLI Command line interface

ČIA Czech authorization institute

ČMI Czech metrology institute

DB Database

DDoS Distributed denial of service

DNS Domain name system

DWH Data Warehouse

EC2 Elastic Compute Cloud

EBS Elastic Block Store

EFS Elastic File System

EPROM Erasable programmable read-only memory

HW Hardware

IaaS Infrastructure as a Service

IoT Internet of Things

IT Information technology

LRSW Legally relevant software

MID Measuring Instruments Directive

MPO Ministry of Industry and Trade

NIST National Institute of Standards and Technology

NoSQL Not only SQL

OS Operating system

PaaS Platform as a Service

RDS Relational Database Service

S3 Simple Storage Service

SaaS Software as a Service

SQL Structured Query Language

SSD Solid state drive

SW Software

ÚNMZ Technical standardization, metrology and state examination bureau

VM Virtual machine

VPC Virtual Private Cloud

VPN Virtual Private Network

WG 7.2 WELMEC Guide 7.2

Appendix B

Contents of enclosed CD

- readme.txt .... the file with CD contents description
- src .... the directory of source codes
  - thesis .... the directory of LaTeX source codes of the thesis
- text .... the thesis text directory
  - thesis.pdf .... the thesis text in PDF format