Web Observations:
Analysing Web Data through automated Data Extraction

Inauguraldissertation

zur Erlangung der Würde eines Doktors der Philosophie
vorgelegt der Philosophisch-Naturwissenschaftlichen Fakultät
der Universität Basel

von

Alexander Olivier Gröflin

aus Basel, Schweiz

Basel, 2020

Genehmigt von der Philosophisch-Naturwissenschaftlichen Fakultät

auf Antrag von

Prof. Dr. Helmar Burkhart, Universität Basel
und
Prof. Dr. Liz Bacon, Universität Greenwich

Basel, den 11. Dezember 2018

Prof. Dr. Martin Spiess
Dekan

To my parents, Gertraud and Dr. med. Urs Beat Gröflin-Glück, my sister Lilian and my brothers Stefan and Fabian.

Acknowledgments

Writing a thesis is a lifetime task and due to the strong support of certain people I was able to successfully fulfil it. First and foremost, I want to express my acknowledgements to my supervisors, Prof. Helmar Burkhart and Prof. Liz Bacon. Their suggestions and inputs were crucial for making my thesis the best it can possibly be. Also, I am very grateful for the trust given by Prof. Helmar Burkart over the time I was allowed to be part of his now dissolved research group, "High Performance and Web computing (HPWC)". I wish to express my sincere gratitudes to Prof. Sabine Gless, who did not only provide financial support for the last year of my employment but also provided me with a priceless insight into the law and its growing influence on data and computer science. The confidence she has given me and the experience at the Faculty of Law at the University of Basel was crucial for turning my work into an interdisciplinary thesis.

I would also like to thank my co-workers, friends, and fellow students Danilo Guerrera, Antonio Maffia, Dominic Bosch, Bas Kin, Robert Frank, Yvonne Wegmüller, Patricia Krattiger, Mario Weber, Christian Frei, Joëlle Simonet at the Department of Computer Science and Mathematics and Carl Jauslin, Laura Macula, Dario Stagno, Christine Möhrke-Sobolewski, Nadine Zurkinden, Lia Börlin, Armand Kurath, Quirin Meier, Claudine Abt at the Faculty of Law and Yun Seok Lee at the London School of Economics and Political Science. The constant critical discussions on the various aspects of conducting research for a thesis were very helpful and of great support to me. My special thanks go to Martin Guggisberg, who was particularly influential to the progress and development of the thesis. His constant support and invaluable guidance paved the way to this very thesis. Additionally, the above mentioned have turned from colleagues into dear friends who did not only support me on a working level but have built me up whenever I needed it. This is an experience very dear to me and I am glad I could share these last years with such positive, uplifting and

supportive people. I am also very grateful to Pauline Pfirter and Bernhard Egger for thoroughly reading my thesis and polishing my writing and my structure to perfection.

Additionally, I would like to express my thanks to the Freie Akademische Gesellschaft (FAG) and Prof. Beat Schöneberger who provided me with the financial support required to have this thesis finished. Furthermore, I would like to express my gratitude to the Swiss National Science Foundation for partially funding this project (SNF NRP 75 Big Data 167182 "Legal Challenges in Big Data. Allocating benefits. Averting risks.").

Finally, I could not have finished this project without the immense support from my family and closest friends who would cheer me up and push me forward whenever necessary. Thanks to my mother, Gertraud Gröflin-Glück, who made me go to the very first class of the Computer Science High School, I could obtain a Swiss Federal Certificate of Computer Science which was only the starting point for my academic career. Later on, also thanks to my father, Dr. med. Urs Beat Gröflin, I could continue my education in Switzerland, Italy and the United Kingdom.

Abstract

In this thesis, a generic architecture for Web observations is introduced. Beginning with fundamental data aspects and technologies for building Web observations, requirements and architectural designs are outlined. Because Web observations are basic tools to collect information from any Web resource, legal perspectives are discussed in order to give an understanding of recent regulations, e.g. General Data Protection Regulation (GDPR). The general idea of Web observatories, its concepts, and experiments are presented to identify the best solution for Web data collections and based thereon, visualisation from any kind of Web resource. With the help of several Web observation scenarios, data sets were collected, analysed and eventually published in a machine-readable or visual form for users to be interpreted. The main research goal was to create a Web observation based on an architecture that is able to collect information from any given Web resource to make sense of a broad amount of yet untapped information sources. To find this generally applicable architectural structure, several research projects with different designs have been conducted. Eventually, the container based building block architecture emerged from these initial designs as the most flexible architectural structure. Thanks to these considerations and architectural designs, a flexible and easily adaptable architecture was created that is able to collect data from all kinds of Web resources. Thanks to such broad Web data collections, users can get a more comprehensible understanding and insight of real-life problems, the efficiency and profitability of services as well as gaining valuable information on the changes of a Web resource.

Keywords: Web Observatory, Web Observation, Web Data Collection, Web Architecture, Law & Data, Regulatory

Contents

Introduction

> It doesn't matter how much junk there is out there because: *you don't have to read it*.
>
> — Sir Timothy John Berners-Lee, inventor of the World Wide Web (born June 8, 1955)

About three decades ago, Tim Berners-Lee started a project that has changed the world as we know it today. The World Wide Web is the biggest information and knowledge network ever known by humankind. And while the Web provides content to every single question one could think of, it is at the same time a massive information space to which even more information is added over time.

Now, what if a person is searching for a specific kind or type of information? What if that person finds loads and piles of data on the Web and cannot make sense out of it? What is the difference between data and information? How does that person know what part of this data contains relevant information and what is just chunk and waste?

Like an observatory reaching for the stars, Web scientists are the telescopes that help people filter and extract the "right" information from the structured, semi-structured and mainly unstructured data available on the Web. For example, Web scientists observe and analyse human behaviour and current events. There seems to be a growing need for further research in the area of Web science.

With all this in mind, it is the Web scientist's task to create a technical structure for the collection of relevant information. So-called Web architectures are able to focus on the observation of the Web, specifically on any changes in information of a Web resource. If properly designed, such a Web data collection architecture – academics also call it a Web observation – is able to collect vast amounts of data, changes to the data as well as new additions to the data with

respect to the time scale. The notion of Web observatories in which a system gathers and links data on the Web for the advancement of economic and social prosperity has been outlined by the Web Science Trust. Furthermore, it has been defined as "a system that locates and describes existing data sets [on the Web] around the world". [1]

In contrast, a Web observation architecture wants to harvest and collect data from the Web. However, the proper design of a Web architecture for Web observations is a challenging task. In the design process, the goal is to create a technical sound architecture that is able to observe and collect data from any kind of Web resource. This work outlines purposive collection algorithms, a comprehensive event condition action system that allows to detect and process events on the Web, and a task-specific Web observation system for all sorts of Web data collections. Predominantly, it is a computer scientist's job to consider technical aspects of a Web resource, possible future changes to its structure as well as user interactions and usage analytics of a Web page.

This thesis has taken on the task to create a generally functioning Web architecture that is capable to collect Web data from any given Web resource with little effort. Under the research questions, the author has created several Web architectures to test and find the most adaptive solution for all kinds of Web resources. The final Web architecture consists of a container solution that is flexible enough to observe desired Web resources with the technical means necessary.

This thesis analysed and answered the following research questions:

- **Is it legally allowed to systematically collect Web data without anybody knowing about it?**
- **How to design a flexible architecture that is able to collect data from desired Web resources over a long period of time?**
- **What kind of Web data may be the most interesting for a Web observation?**

To answer these questions, the thesis is structured in six parts including interdisciplinary aspects of data collection, the law, and some influential social science aspects.

First, the fundamentals of data and Web science are outlined. This initial part mainly focuses on the everyday questions raised by the use and application of the Web. These fundamentals sketch out the environment in which this thesis and the research questions are set.

Whilst the early Web was mostly free from national legal regulations besides already existing law, legislators now try to take on the legal questions raised by the Web. These new regulations are currently heavily discussed and account for the growing importance of the Web in the society and our daily life. Especially issues like data privacy, fake news and governmental data collections are current hot topics all around the world. Legislations and judiciaries try to influence the future development of the Web.

In the second part, the methodology of this thesis will be presented in which the research questions are refined according to the research made in the first part.

Therefore, in a third part, recent legislative projects and political issues in regard to data protection and collection, the Web and its future are presented. Over the last couple of months, new legislation on data protection and data collection entered into force in the United States and the European Union. These new laws will extensively influence the processing of Web data access and data collections.

Fourth, the technical issues concerning "how" to create and design data centred Web architectures are discussed. In this part, the architectural approaches behind a Web observations will be discussed prior to giving a broad insight in the three architectural approaches created over the time of the study.

The fifth part will focus on the "what" kind of data can be collected from Web resources. It outlines "where" interesting data is found and "why" it makes sense to collect it while presenting the results of each Web observation project. Furthermore, it clarifies what is meant by "interesting data" mentioned in research question three. From a merely subjective perspective, interesting data can be described as meaningful data that contains value for someone or something, or simply, data that provides an advantage in knowledge by collecting it over time. Based on the results of the data collection, a Web architecture will be proposed, which is able to perform Web observations that collect data from any given resource on the Web. Moreover, a benchmark will indicate the boundaries of the proposed solution.

In the final sixth part, a conclusion will be drawn that highlights the most important findings, lessons learned, and possible future research directions.

Part I

Fundamentals

Chapter 1

Data Aspects

> In those days, there was *different information* on *different computers*, but you had to log on to *different computers* to get at it. [...] Often it was just easier to go and *ask people* when they were having coffee...
>
> — Sir Timothy John Berners-Lee, inventor of the World Wide Web (born June 8, 1955)

This chapter outlines data aspects that are relevant for Web observations and clarifies definitions of major data terms.

1.1 It is all about Data

The term data is ambiguously used in many disciplines all around the world. Diversity of data itself and its usage in the field of Information Technology (IT) fuels this confusion. This chapter will give the fundamental understanding of data and its conception. Ultimately, Web data can be collected through Web observations.

IT digitally facilitates the processing, storing and transportation of data. This enables us to pile up more data every year than in all the previous combined [2]. Shortly after the millennium human kind started to create data at exponential rates. IT paved the pathway for this data explosion into the information age in which our society has become almost completely dependent on

data. Particularly, data exchanges on the Web play a key role in e-commerce for goods and services. This technological change also shows its effects in daily life, ranging from commuting and working to how we spend our leisure time.

Data has become some sort of oil that fuels the digital economy, "an immensely, untapped valuable asset" [3]. Furthermore, it is considered "the most valuable resource" in this economy [4]. However, this comparison is perhaps more than vague, after considering the fact that oil is a finite commodity while data encapsulates infinite economic value that can be unlocked through analysis. Therefore, to compare a raw material with the manifold appearances of digital data is too simple. Oil and data differ not only in attributes but also in the way they are consumed. Oil has an energy value, it is irretrievably consumed and naturally gone while data provides information based on bits and bytes that "can be duplicated, shared and reused". [5] These simplifications of IT have the same notion that Nicholas G. Carr expressed in his article in the Harvard Business Review "IT Doesn't Matter" [6]. Carr portrayed IT as a dominant cost factor and drew parallels to commodities such as electricity and water. But is it really true that we are able to turn IT on and off like the water tap? New technology companies differentiate themselves by "service, product feature, and cost structure" [7]. It is clear that IT-based innovations come with risks, but also hold key advantages. Being the first mover needs courage and justifying these costs against superiors is even more difficult. In the end, the customers decide whether an IT-based product is useful or not. And if more and more customers think it is, then one has gained a competitive advantage over competitors.

1.1.1 Definitions of Data

First, it is necessary to clarify what exactly is meant by the term "data" and how it is defined. The English term "data" can be derived from Latin and stands for the plural of "datum" [8]. After the second world war, "data" has been used for the first time in connection with IT as "transmittable and storable computer information" [9]. In 1954, IBM used the term "data processing" for its IBM 704 computer. It was the first mass-produced computer that was able to operate floating-point arithmetic. [10]

Generally, data is considered to be a superset of information or vice versa, information is a subset of data which consists of values or findings [11]. Checkland and Holwell (1998) define data in relation to information, meaning and learning. They divide disciplines or fields accordingly in which Computer Science, Sociology, Philosophy, and Educational Science are brought into play

with information and meaning [12]. This seems surprisingly interesting since Sociology, Philosophy, and Educational Science are all considered to be social sciences.

According to Schauer (2005), information is the meaning conveyed by a message which is sent from a source to a destination. Information is created from data while data in turn is composed of individual characters, an alphabet and a syntax. Codes facilitate mapping of characters of one alphabet into characters of another alphabet. Ultimately, data can be stored or transmitted with the help of physical signals. Such data signals actually exist in reality, e.g. optical signal, they can be also observed and measured. In contrast to data represented by these signals, information is an abstract interpretation of physical data signals. [13]

With his landmark paper "A Mathematical Theory of Communication", Shannon (1948) established "information theory" as a new research field. Although not even Shannon initially considered his paper to be of such importance. Yet, it soon turned into a major reference and is cited until today. Information theory outlines mathematical methods for measuring characteristics of data. Under information theory the definition of information is not directly related to meaning or knowledge. Information-theoretical methods try to quantify messages in which information is encapsulated. Irrelevant of what code a message contains, the information content depends solely on the probability with which the receiver expects such a message. [14]

However, Hicks (1993) defines data as a detailed elaboration for data transmissions: "A representation of facts, concepts or instructions in a formalised manner suitable for communication, interpretation, or processing by humans or by automatic means." [15]. In essence, Hicks describes data as fit for communication and interpretation by humans and machines.

Many scholars hold the view that the meaning of this term has been broadened in recent years. For Checkland and Holwell (1998), data does not only contain these kinds of communication, data is considered an enclosement of three different viewpoints:

- **Objective view on data**
  Fact records which are considered under the same parameters by every person, e.g. temperature measurement;

- **Subjective view on data**
  Unproven concepts which are considered manifoldly interpretable by individuals, e.g. an opinion;

- **Intersubjective view on data**
  Agreed composition of information ready for communication, e.g. language syntax. [12]

The *objective view on data* enables accurate interpretation of data. Automation of data collections and manipulation thereof can transform it into meaningful information which is also known as data processing [16]. In contrast to the objective view, *subjective view on data* is based on interpretation whereas meaning will be given by a human counterpart. It is considered that subjective data needs to be heavily structured in order to be stored within a data structure, e.g. person A thinks 20 degrees Celsius is cold, while person B thinks it is hot. With the help of pre-shared definitions, the *intersubjective view on data* enables processing of data by either a computer or a human.

Data is Omnipresent

In a broader sense, data might just be all around us. According to a special report on managing information in The Economist (2010), it is believed that the term data stretches much further than just texts, tables, and figures [17]. It just has to be collected and stored in a machine readable way for further interpretation. For example a completed survey form of a survey or a handwritten note can be considered data because it contains the very information for further analysis. In her book "Raw Data Is an Oxymoron", Gitelman (2010) describes data as a matter of disciplines rather than something related to Computer Science. She might be partly right, as long as it can be been assumed that e.g. pen craft is considered an artistic skill set. Yet, even if we are talking about Goethe's "The Sorrows of Young Werther", we are not talking about the data within the masterpiece for obvious reasons. Main topic of conversation is the idea behind the story and the consequent interpretation. One implication thereof is that data must be interpreted in order to make sense of it. Nevertheless, this idea will still continue to evolve across disciplines. Gitelman concludes that "[t]he subject of data is bound to alienate students and scholars in disciplines within the humanities particularly" [18].

Data is Versatile

Data exists in many forms which makes its appearance diverse and versatile. Data may constitute a single entity or a set of entities whereas such entities are also called data sets. Basically, it is a set of values in qualitative or quantitative

means. According to Kitchin (2014), data is mainly used within these three fields:

- **Scientific research**
  E.g. data collection in the field, data cleaning, data visualisation;
- **Businesses and non-governmental organisations**
  E.g. sales data, revenue and profit;
- **Government**
  E.g. unemployment rates, birth rates and age of citizens. [19]

The most basic form of data is data which has been collected automatically or manually from any given source. This kind of data is also called raw or unprocessed data [20]. Logically, information ought to be the output of an information system, in contrast to raw, unprocessed, collected data which can be assessed as the input [12]. Furthermore, it can be said that raw data is considered processed data as soon as it has been cleaned and is in the right format ready for further analysis, e.g. from obvious false or duplicate entries or from entry errors. The following model can be used to identify what type of data is involved:

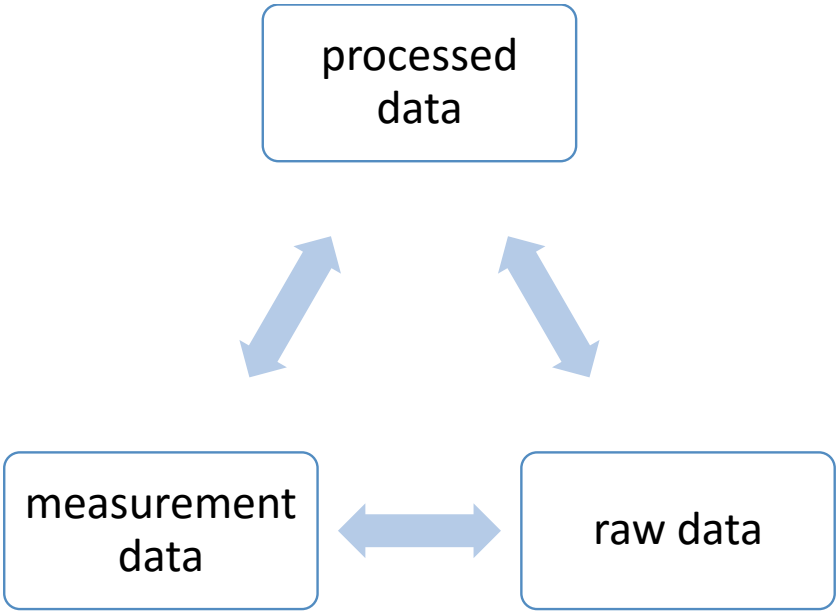

Figure 1.1: *Data must be transformed or processed before interpretation is possible. There is no clear boundary between the entities of measurement, raw, and processed data [18].*

Compared to raw data, measurement data is used in the context of empirical research, e.g. direct and indirect observation or experience [18]. Depending on the scope, data that has been already processed may be either raw or measurement data which explains the ambiguous understanding of these data terms.

Data is Knowledge

Data is the key essence of our modern age; it enables innovation which in turn creates new pathways for value creation. Collection, storage, processing, or interpretation of data has become a dominant role throughout all industries [21]. Rowley (2007) examines the hierarchy of data, information and wisdom and tries to postulate a distinct definition for them. It seems certain that information is defined by data, knowledge by information, and wisdom by knowledge [22]. The key question is whether one is able to produce information from data automatically and whether a significant difference between stored and new data can be noticed [12]. Data holds information or knowledge in a compressed form such as a code that is often easier to process [19]. Many academics have described data as the basis for all further information processing [22, 23]. Ackoff (1989) categorises the human mind into five denominations which outline what the outcome of data can be:

- **Data**
  Consists of actual symbols;
- **Information**
  Processed data that gives answers to "who", "what", "where", and "when" questions;
- **Knowledge**
  Data and information application that answer "how" questions;
- **Understanding**
  Comprehension of "why";
- **Wisdom**
  Evaluated understanding. [24]

Whereas the first four categories are related to the past and what is known, the last category, "wisdom", tries to foresee or to predict the future with the help of past and present data. Thus, to achieve the point of wisdom, every other point has to be thoroughly adhered [24]. This coincidences with the field of

knowledge management which is the process to create, share, use, and manage knowledge and information [25]. In contrast, information management is an activity to get information from a source to those who need it, including the storing and destruction of information [26]. On the basis of these definitions a pyramid with five spare parts can be elaborated in connection with data, information, knowledge, and wisdom:

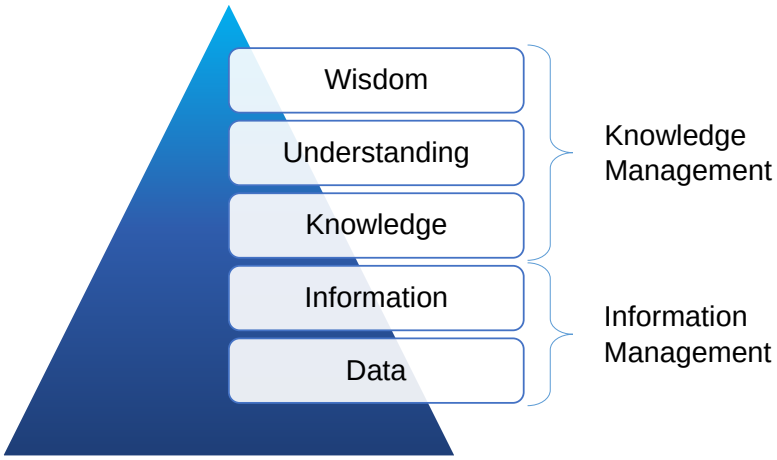

Figure 1.2: *Information is typically defined in terms of data, knowledge in terms of information, and wisdom in terms of knowledge [22, 23, 24].*

Nevertheless, we are constantly bombarded with data. Therefore, we have to distinguish between important and not important data. The Web has become a massive information source and has great potential for information gathering. By removing the "noise" through the application of an architecture, it is possible to harvest and analyse the flow of information in order to gain a quicker understanding. Thereby, recognition of major changes can be achieved almost in real-time by observing what is actually happening on the Web. The combination of several data sources instead of only one makes the removal of noise without losing accuracy a delicate matter. [27]

1.1.2 Web Data

In 1991, the first people outside the European Organisation for Nuclear Research (CERN) were invited to join the Internet [28]. Sir Tim Berners-Lee's original proposal of "Information Management" became reality through the creation of the World Wide Web [29]. From the moment when the public has been invited

to collaborate with the Web community, millions of people came together by using fundamental Web technologies that are still in use today.

As mentioned in section 1.1.1 "Definitions of Data" data comes in many forms and types. The Web encapsulates all kinds of data and enables access to everybody. Whether written text within HTML tags or a linked documents, all linked files accessible through this global data space can therefore be called Web data.

It is clear that the Web has become one of the most influential technological innovations for global economy, education and private life. The first decade of the Web is characterised by a massive proliferation throughout the public and private sectors, including access of individuals. Since the turn of the millennium the Web has transformed into a diverse information space. Proposed service-oriented architectures (SOA) tried to design Web services independent from Web technologies [30]. A selection of various approaches would be Web services such as RESTful HTTP which are based on the Simple Object Access Protocol (SOAP) and Extensible Markup Language (XML) [31]. Automation, customisation or personalisation are Web technologies that play a role in the continuous improvement of the Web and have proven to be commercially successful e.g. auto-fill which completes forms automatically [32].

Observation Data is Unstructured

Besides ordinary HTML pages and text files, Web data provides images, documents and mail messages. Web data is by nature unstructured, meaning it does not appear in predefined pattern. HTML as a markup language is used for rendering or presenting information but it is limited in describing the semantics – meaning and logic – of its content within the HTML elements [33]. Strategies to enhance the meaning of Web content might involve extensive use of meta tags in which data on data is written. Meta tags are not visibly displayed, but make it easier for Web applications to categorise the contents.

Linked documents usually have a predefined structure, however, when these formats are mixed up on the Web together, they still form a vast pool of unstructured Web data [34]. The reason for this lies in predominantly untagged and file-based deployment. In other words, data on the Web most often lacks of proper attributes and self-describing structures. As a result, observation and interpretation of Web data is very difficult to achieve without additional technologies. [35]

Web of Data

Ideas towards more structured and self-describing data on the Web have been put forward, however, these approaches did not spread as quickly as hoped [36]. The World Wide Web Consortium (W3C), founded and led by Tim Berners-Lee [37], formalised a standard called the "Semantic Web" as an extension of the World Wide Web that delivers a "framework that allows data to be shared and reused across application, enterprise, and community boundaries" [38]. The term "Web of Data" is often referred to as the Semantic Web which is based on the concept of transforming the Web from distributed file servers into distributed databases. The basic principle of the Semantic Web includes interoperability and re-usability of available Web data. Although, there is still a lot of data on the Web that is not linked together to a web of data. The Semantic Web claims to link data in other places in order to achieve a web of data instead of just having data available on the Web. [39]

The technology behind this concept is the Resource Description Framework (RDF). Specified by the W3C, RDF "extends the linking structure of the Web" by describing relationships between Web resources. It consists of a *subject*, an *object*, and a *predicate* or *property* which describes the relationship of a resource. These three parts are also called a *triple*: subject, predicate/property, and corresponding URI's. The ability to combine semi-structured or structured Web data with triples enables Web users to create a mixture of data that is suitable for many applications. [40]

Science of the Web

Web science is an emergent research field that studies socio-technical systems such as the Web. Its origin can be found in the article "Science of the Web" by Tim Berners-Lee et al. (2006) in the journal "Science", in which apparently the first time the term "Web Science" was used. The article points out the necessity of an interdisciplinary new field both in engineering and societal means which should study the Web as a socio-technical system. The most significant aspect of Web science is the notion of the Web as a massive information space. It is seen as a technical construct for information and data based on a specific languagese and protocols. [41]

The impact of the Web on our society is very broad since we are all socially embedded in the Web. Implications on commercial, social, political, and legal spheres are profound and may also alter the behaviour of society. Trust or reputation, privacy, and copyright are issues that increasingly undergo pressures from Web technologies. Such technologies are used in everyday communication

and thus store all sorts of information. Web science tries to assess the procedures of Web communities and resources, e.g. correlations and relationships, by observing these interactions. Shneiderman (2007) clarifies Web science as processing of information that is freely available on the Web. It can therefore be assumed that Web observations may also be a part of Web science. [42]

Hendler et al. (2008) have outlined that the Web is changing every day which makes it additionally challenging to make observation. Extensive research is necessary to understand the mechanisms of dynamic and changing Web applications. The Web has become a multi-user and social environment that has a strong repercussion on "social structures, political systems, commercial organizations, and educational institutions". To study these interactions it requires quantitative and qualitative research corresponding to specific disciplines. It is a widely held view that the Web has changed the world forever, and that the world shapes the Web. [43]

Difficulties arise, however, when an attempt is made to study the Web resources and its underlying communities. The Web is in a constant change, contents might disappear, it evaporates. It is somewhere stored but not any longer available on the Web. Therefore constant observation must be applied in order to keep track. On the one hand, Web science deals with the structures and classification of the Web and such also includes its Web resources and contents. On the other hand, Web science is a colourful mix of academic disciplines and is not only limited to natural and social sciences and the humanities. Moreover, it can be seen as an applied science which focuses on the development. As such, it also includes the relationships and interactions among humans through technology, e.g. Web page creators and Web users. This leads to an ever further expansion of the construct. Therefore, Web science involves disciplines such as philosophy, economics, law, sociology, and computer science. Furthermore, it examines development, impact, meaning, and social interactions of the Web in order to produce new findings with means of beneficial methods or models. [41]

1.1.3 Data Sets

Data sets are commonly refered as an aggregation or a combined collection of data in terms of size and amount. They often come in purpose-specific file formats whereas the two most distinct formats are either common data files or data streams. Within data sets delimiters such as commas or rows separate units of data which often resembles a database table. Depending on the intended use of the file contents and its aggregation, data usually comes from many different sources. The content can be either a single table, a data

matrix, values or variables, which each matches a defined entry within the data set [27]. There are numerous formats which are considered to be digitally endurable for future usage. Such formats have to follow guidelines in order to sustain the history of time. Arms and Fleischhauer (2005) call these parameters necessary sustainability factors:

- **Disclosure**
  Degree to which specifications and tools exist and are accessible;
- **Adoption**
  Degree to which the format is used;
- **Transparency**
  Degree to which digital representation is open to direct analysis;
- **Self-documentation**
  Digital objects that contain basic descriptive meta data;
- **External dependencies**
  Degree to which a format depends on particular hardware, operation system, or software;
- **Impact of patents**
  Degree to which a format prevents content from archiving by patents;
- **Technical protection mechanisms**
  Implementation of mechanisms, e.g. encryption that stops the preservation of content. [44]

Due to the valuation of these factors it can be estimated how sustainable a format is and, in particular, whether it can still be read and processed by machines in the near future. Therefore, if data shall be preserved for the public, it makes sense to adhere to digitally endurable formats. Nonetheless, the Open Data Handbook provides brief descriptions and a list of sustainable file formats which form the data structure used for data sets [45].

Naturally, data sets may either contain values and variables or it can loosely be described as a collection of tables of an event or experiment. If data sets become so large that ordinary applications have difficulties to process the actual data, it is viewed as big data [27].

1.1.4 Open Data

Open data is a concept in which data shall be freely available to everyone. It is thought that open data predominantly uses open file formats and no restrictions such as patents or technical hurdles which could hinder its usage [46]. Generally, all data can be transformed into "open" or accessible data; it usually involves larger data sets [47]. The term "open data" has been well established by government initiatives such as https://data.gov.uk. There are many different movements driven by many different stakeholders, e.g. open source, open government, or open science. Open data may hold a fresh perspective for the general public and private sector, research, industry and society: "Open data and content can be freely used, modified, and shared by anyone for any purpose" [48].

Open data may also contain interesting information that can be used for Web observations. Good examples thereof are public weather stations that measure and publish data sets regarding the weather, e.g. temperature, sunshine, humidity, etc. Yet, its content may be used for Web data extraction. Open data already exists in a reusable form that is easy to process and therefore makes the process to extract meaningful information far more convenient. Usually, Web observations create open data as an output in order to have transparent results. Please see also section 4.3.1 "Open Data Access".

The importance of open government data for modern democracies has been outlined by 30 open government advocates [49]. In essence, governments applying these principles are "more effective, transparent, and relevant to our lives" [50]. In 2007, the attendees of the Open Government Working Group defined the original eight open data principles that still have its significance today (see also [51]):

1. Complete: All public data is made available. Public data is data that is not subject to valid privacy, security or privilege limitations.

2. Primary: Data is as collected at the source, with the highest possible level of granularity, not in aggregate or modified forms.

3. Timely: Data is made available as quickly as necessary to preserve the value of the data.

4. Accessible: Data is available to the widest range of users for the widest range of purposes.

5. Machine processable: Data is reasonably structured to allow automated processing.

6. Non-discriminatory: Data is available to anyone, with no requirement of registration.

7. Non-proprietary: Data is available in a format over which no entity has exclusive control.

8. License-free: Data is not subject to any copyright, patent, trademark or trade secret regulation. Reasonable privacy, security and privilege restrictions may be allowed. [50]

There are at least three major benefits from open data. Open data not only makes public actions of a government transparent by its disclosing of data. It also increases the accountability of a government and additionally has economic benefits for the general public. [52]

Linked Data

Linked data is a technique to formalise and publish structured data. Open data is often confused with linked data and/or combined to linked open data. Linked data creates an interlinked information space that can be accessed through meaningful semantic queries. Whereas the term "Semantic Web" commonly refers to formats and technologies, linked data facilitates the automatic finding of related data for Web users and machines [53].

An "openness rating" has been set out by Tim-Berners Lee that advocates linked open data. It indicates how data sets can be transformed into linked open data [54]. The scale is based on a scheme of a maximum of five stars which point out the usefulness of data sets:

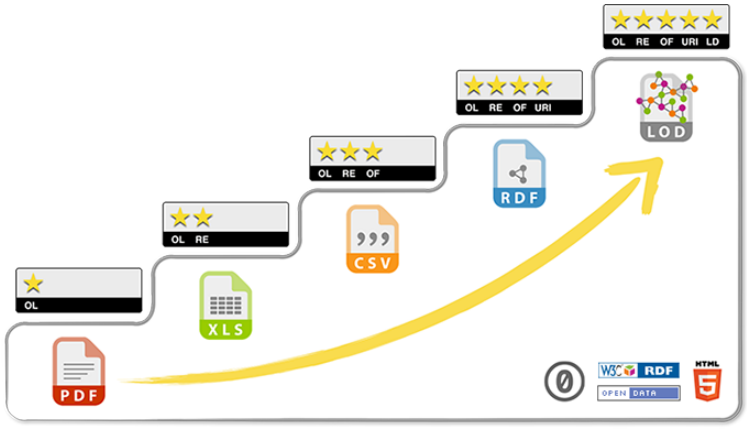

Figure 1.3: *OL: open licence; RE: machine readable; OF: open format; URI: RDF standards; LD: linked data. [55]*

The goal of this scale is to achieve the state Linked Open Data (LOD) that not only allows humans to read the data but enables machines to identify, read, and link data for further analysis. The steps towards LOD is described as follows:

- **Open licence (⋆)**
  Data is available on the Web under an open licence in any format;
- **Machine readable (⋆⋆)**
  Data is available on the Web under an open licence in machine readable means, e.g. Excel file.
- **Open format (⋆ ⋆ ⋆)**
  As of the above plus available data is in a non-proprietary format, e.g. CSV file;
- **RDF standards (⋆ ⋆ ⋆⋆)**
  All of the above plus open standards as defined by the W3C such as RDF in order to identify things;
- **Linked Data (⋆ ⋆ ⋆ ⋆ ⋆)**
  All of the above plus linking of data to other data in order to provide context. [53]

In conclusion, linked data is the utensil to connect to the Semantic Web. Common sense determines whether a link has to be made within the provided five stars scheme. In terms of Web observations, it usually does not matter whether data complies to this scheme. In this thesis, Web observations scenarios gather insights from Web resources that are not labelled open data.

1.1.5 Big Data

The term big data has been coined by John R. Mashey (1998) who predicted an explosion of widely accessible data. Creating, understanding, storing, moving, or else, will put infrastructure under stress – "InfraStress" – or in his words: "Drown in Wave of Infrastructure Stress" [56]. Its meaning has since been transformed to the use of data sets that are too big to process in a traditional IT environment.

Whereas Gartner defines big data as "[...] information assets [...] that enable enhanced insight, decision making, and process automation." [57], McKinsey focuses on the value of big data: "Distilling value and driving productivity from

mountains of data" [58]. Web data may also be a source of untapped value. The sheer size of the Web is very much a big data problem in variety and complexity. In addition, Franks (2015) suggests that Web data is the original big data [59]. As a result, it is reasonable to capture states of the Web and thereafter interpret these data sets. Distilling value from the Web still remains a challenge. A feasible way may be the refinement of Web data into smaller chunks in order to make the content more transparent. Furthermore, ordinary IT tools are not capable to make use of such large data sets. Conclusively, De Mauro et al. (2016) define big data as follows: "Big Data is the Information asset characterized by such a High Volume, Velocity and Variety to require specific Technology and Analytical Methods for its transformation into Value." [60].

According to the extensive literature review of De Mauro et al. the main emphasis of big data are towards *information*, *technology*, *methods* and *impact* [60]. Yet, data dimensions have been firstly introduced by Laney (2001) and have been extended ever since. According to many in the field the foundation of the "three V's" gave rise to the now well established "five V's" of big data:

- **Volume**
  Refers to sheer size of data that makes data to large to store;
- **Velocity**
  Refers to the speed in which data is created or transferred;
- **Variety**
  Refers to the many types of data, e.g. structured and unstructured; [61]
- **Veracity**
  Refers to the reliability or accuracy of data while volume often compensates the lack of quality [62];
- **Value**
  Refers to the value created by the big data analysis [63].

Taken together, these points outline that big data is rich in high *volume*, *velocity*, and *variety*. The relatively recently added points *veracity* and *value* extend the big data understanding in terms of economic value and accuracy of results [62, 63]. The first three V's – "Volume, Velocity, and Variety" – may be grouped together by the characteristics of data or information as it exists. "Veracity" urges technological and analytical methods and means for accurate

and precise results, whereas "Value" is dedicated to the economic utility of having big data transformed into valuable information.

Unfortunately, there are quite a few misconceptions about big data. Much can be understood and the terminology is broad. However, on a regular basis the scientific literature criticised that the expectations put into big data are too high. It seems that big data is often anticipated as a solution for all sorts of problems. The self-appointed big data guru Buchan (2016) put it straight: "I think the biggest misconception is that big data is the answer to everything, and that bigger data will always lead to a better answer." [64]. Another weakness is that there is too much focus on data. The bigger the better is another common misconception of big data. In some cases, more data may indeed reduce the discovery power. That is the reason why it is important that there is an even balance between the amount of data and the information sought [64].

Big data problems in respect to the five V's may also occur with Web observations when harvesting a lot of Web data over time. However, the collection process is limited by the Internet connection and the used hardware. Network latency highlights the importance to collect only necessary data instead of as much as possible data. The ATLAS experiment at CERN with its Large Hadron Collider (LHC) uses an equivalent approach in much bigger dimensions by reducing the amount of data over several layers [65].

Characteristics of Big Data

As outlined before, the initial three V's can be grouped together by its data characteristics. "Volume" clearly addresses the amount of data that is being processed whereas the size is often undefined and rather depends on its usage. Because of its size, data is typically stored on distributed systems and the amounts of bytes reach from terabytes to zettabytes [66]. One terabyte for example is likely to store roughly 1,000 copies of the 2010 edition Encyclopædia Britannica, 32 volumes without images [67].

"Velocity" addresses the input/output speed of data. Due to the massive influx of data in particular IT environments, storage of data is only partially possible. Existing IT infrastructures are at limits and have to reduce the data set in subsets which are easier to work with. Similar to the ATLAS experiment vast amounts of data must be reduced in order to be able to store and process it. Before its reduction, these measurements accumulate enormous amounts of data within seconds although this data is most likely not meant for storage. Therefore, real-time data can also be treated as big data. [65]

In a perfect world, only structured data would exist. Data would always

exist in well organised and well defined structures which seamlessly fit together. This utopian idea is quite contrary to the unstructured data resources which are omnipresent. "Variety" points out the diversity of data in the real world in which the arrangement of this variety of data is a very time consuming task. There are three different types of data:

- **Structured**
  E.g. databases, data warehouses, enterprise systems;
- **Unstructured**
  E.g. text, images, videos;
- **Semi-Structured**
  E.g. JSON, XML, HTML.

It is clear that structured data in tables, rows, and columns which use relational keys are far easier to deal with. Machine structured data is easily processed which simplifies the management of data because of its structured appearance as for example a database. Nevertheless, most of the data in the Web and in organisations is unstructured data, although regularly used. It includes text and multimedia files but also mail messages [68]. Even if they have a file format structure, they are still considered unstructured hence it does not easily integrate with a database architecture. Semi-structured data lies in between structured and unstructured data; it is not stored within a relational database but comes with properties that may be used for the transformation into structured data.

Data Analytics

Basically, analytics makes use of mathematics and statistics. Facts and figures are calculated to make better decisions. There are three distinct subdivisions of analytics:

- **Descriptive analytics**
  Analysis of data in the past;
- **Predictive analytics**
  Analysis of data in the past to infer the future;
- **Prescriptive analytics**
  Analysis of data in order to optimise processes. [69]

According to Klous and Wielaard (2017) we ourselves have become big data on many aspects ranging from society to technology. Although, data may revolutionise businesses, it cannot be 100 percent accurate in identifying the next big market disruptions. Therefore, Klous and Wielaard argue that we, the users of software and hardware, are the drivers of big data. [70] Data driven business models try to make use of data, for example improving data quality is one option to do so. Therefore, reliability is important in order to differentiate between correlation and causality or quantity and quality. However, this leads to the conclusion that the impact of big data is far broader than it might be anticipated by academia and businesses. The main challenge of big data is the solution of multiple issues at the same time. Far stretched big data analytics involves data analysis, data discovery, data mining, data sharing, data storage, and data visualisation. A fairly new to be considered matter of big data are data privacy and corresponding legal perspectives [71]. For a more in-depth analysis of these issues please see section 4 "Legal and Political Perspectives".

Predominantly, managerial pressure is responsible for data-driven business intelligence. Peter Drucker recognised the significance of "knowledge workers" or "knowledge" as the "most valuable asset of a 21st-century institution [...]" [72]. It seems likely that these findings have encouraged executives to get more data based insights about the business itself in order to identify sources of value. Unsurprisingly, data-driven analytics took its course to executive floors. [73]

Public and private companies seek value within available data from many different sources. Even more, analytics offer real-time capabilities with "speed" and "impact" [74]. Mashing up structured, unstructured, and semi-structured data with the help of "Analytics 3.0" methods will perhaps give "[...] radically more [information] about their businesses [...]" [73]. McAfee and Brynjolfsson (2012) have outlined this move as "a management revolution" and emphasise big data analytics with a quote attributed to both W. Edwards Deming and Peter Drucker: "You can't manage what you don't measure". This might be one explanation why big data is so significant for managers. [75]

Small Data

In his book, Lindstrom (2016) casts doubt on the general assumption that big data is always the answer for knowing the next big thing. The alternative to big data is a more human-centric approach for getting desired information. It consists of "small data" observations made by spending time with people in their usual environment to possibly discover the next trends. These individually collected findings hold key information about human behaviour whereas

allegedly insignificant incidents may contain precious information for marketing purposes. [76, 77]

One of the presented cases was about the toy maker Lego where "analytics" took a wrong turn. After the millennium the company was in serious trouble. Sales and profit were in a steady decline and future prospects were poor. A possible explanation for this might be that electronic games are a huge competitor for classic toy makers. Even worse, market data suggested that the average Lego bricks playing time correlates with the decline of sales and profit. This data was interpreted to mean that less playing time of children will lead to the sale of less products. [77]

Therefore, the first action taken was to increase the size of Lego bricks. The idea behind was to enable kids to build Lego worlds with less time. After introducing the bigger blocks, sales fell even more. The management could not explain this additional decrease. Interestingly, the prediction was quite right, but the action taken was wrong and left the brick company Lego in a crisis. In a last effort, Lego management visited children's homes to get to know what the 21st century child requested from a toy maker. The management was baffled by one child's answer, when it was asked to tell them what it was most proud of. The child was most proud of a pair of old sneakers and not as expected the obvious gaming console or mobile device. It gave the management a lesson in regard to the importance of key observations in the real world in comparison to a data analysis conducted in an office. To this end, the management decided to keep its original brick size. Conclusively, strategies to enhance small data observations might also involve big data analysis. [77]

This chapter dealt with the different definitions of data, related terms, and distinguishes what is meant by them. The next chapter "Technologies for Building Web Observatories" will deal with technical solutions behind Web observations.

Chapter 2

Technologies for Building Web Observatories

> Maybe I'm an idiot, but I have no idea what anyone is talking about [*cloud computing*]. What is it? It's complete gibberish. It's insane. When is this idiocy going to stop?
>
> — Larry Ellison, CTO of Oracle Corporation (born August 17, 1944)

This chapter describes the technologies behind Web observations and outlines the different approaches of Web data extraction. Furthermore, it clarifies the differences between Web observatories and Web observations, provides an overview of key Web mining techniques and illustrates how a basic Web observation works.

2.1 Observation of Web Data

Web technologies have been in constant evolution since the early days. The first decade distinguished the proliferation of home and business websites including web-shops, whereas the last decade transformed the Web into a large scale information space. Business services turned out to be economically successful, however, the Web still has unused potential in terms of programmable automation, customisation and personalisation.

Web users often find themselves mashing up data and functionality from different Web resources and services, e.g. manually post a Tweet as a reaction to a mail with certain information. In other words, Web users surf the Web without automation. With human input execution times are considerable longer than with rule-based systems; real-time orchestration of Web services is yet rather limited. Great value would be added for Web users if specific interaction could be automatically achieved, e.g. detecting and reacting to certain information. This would require an instrument for the identification and processing of changes, adoption of time constraints, composition and final deployment of the user's preferred outcome. [78]

A Web observation focuses on specific data resources on the Web to create meaningful data from Web resources. Interesting parameters for observation on the Web are price developments. Without the constant observation, data of a price at a particular date and time would be lost. This phenomenon has been recognised as evaporation of Web data which is also one justification for the importance of conducting Web observations [79].

2.1.1 Web Mashups

Web applications that use Web content from more than just one source are known as mashups. Usually a new service is created by mixing up and displaying content from two or more sources. An example for a Web mashup is Google Maps which is also called a map mashup. [80]

The first ever mashup created in early 2005 was HousingMaps [81]. It collected houses from more than 10 different cities across the United States that had been posted on craigslists and combined it with Google Maps. With the simple idea of combining two Web resources, a new Web service was created. This notion of borrowing pieces from one Web page without asking for permission from existing Web resources inspired many others to follow up with their own mashup. These efforts eventually resulted in the launch of Google Maps API by end of 2005 which still offers mashup functionalities to this day [82].

Nowadays, access to the API such as Google is free to a certain extent. However, after a certain amount of requests API access may become chargeable. [83]. Access to social media APIs, e.g. Twitter and Facebook, is controlled by usage costs depending on which features are accessed. In addition, regulations of usage apply through platform policies which have been further tightened, more recently in regard of privacy concerns and new legislation. [84, 85]

Furthermore, mashups lead to a variety of Web services that tried to achieve individual composition of existing Web capabilities. One example thereof is

the personal customisation and integration of Web content and functionalities "Syndicate". [86]

To achieve reactivity, some mashup platforms enable Web users to automate tedious tasks and to react on certain events in a predefined way. There have been several attempts undertaken to create helpful tools and platforms in order to tackle basic automation [87]. For example, IFTTT (https://ifttt.com/) or Zapier (https://zapier.com) focus on a one to one connections between Web services simultaneously, e.g. change in the weather triggers a Twitter tweet. However, a shortcoming of these platforms is that events, which have occurred in several distributed systems, cannot be detected as part of one rule. It is certainly not possible to create custom event listeners or actions. For additional customisation, these service providers have to set up such a service connection for each Web resource themselves. Furthermore, selecting more than one action on one single trigger is not possible. They rather create a tunnel from one Web service to another, than creating reactivity in between several resources.

A variety of languages for Web service orchestration has been reported in the last years [88, 89]. For example, Web Services Description Language (WSDL) addresses workflows, business collaborations and long running interactions for well-defined processes. [90, 91]

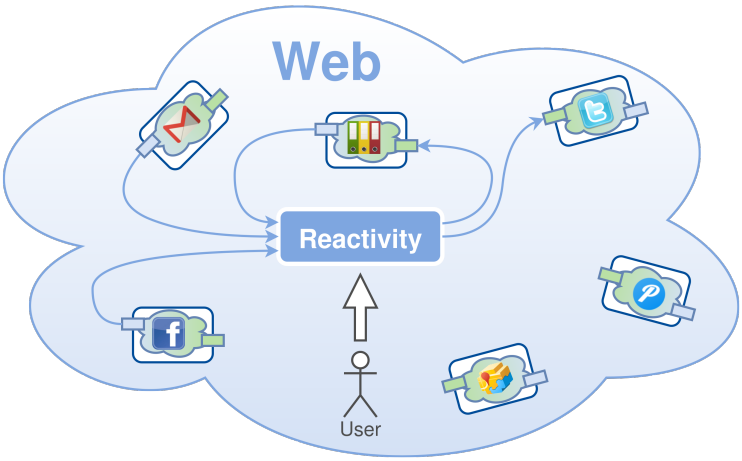

Figure 2.1: *Scheme of a system in which a user aggregates Web resources. Changes originating from Web resources create events that are processed in a reactivity entity. The output is an action that controls Web resources. Personalised settings allow a user-specific orchestration of Web resources, e.g. Facebook, Google Mail, and Twitter etc. [91]*

2.1.2 Data Flow Systems

Another option to guide data through Web services are data flow systems. In recent years, Web services enable Web users not only to create reactive behaviour but let data flow through a predefined setting. One of the earliest attempts was "Yahoo! Pipes" which defines itself as a data mashup tool [92], in contrast to Blackstock and Lea (2014) that categorise such tools as data flow systems [90]. The main goal of Yahoo! Pipes was to aggregate content from many different feeds, whereas desired data sources are selected and represented in a final RSS feed. This also means that the field of application is very limited because it relies on RSS. True mashup tools, for example IFTTT or Zapier, focus on directing the data flow from one Web service to another. By orchestrating Web application programming interfaces (APIs) real reactivity is being created [93]. Such an approach opens up the possibilities of automation in between different Web services. However, the key problem with this concept is that they do not offer generic access to any arbitrary Web resource. Unfortunately, not every Web resource has an accessible API. This means that these platforms have to provide the interface to the desired Web services. Another challenge of Web mashups like IFTTT is the restricted programmability. IFTTT does not allow Boolean operators, which hinders solving expressive tasks. Sometimes it would be useful to react on more complex rules than the simple "If-This-Then-That" scheme. Concluding, data flow systems interconnect existing services and remove the need for any coding. The need for more functionalities inevitably leads to Web systems that want to achieve more than just directing the flow of data. [90, 91]

2.1.3 Condition Action Systems

An increasing amount of literature emerges on reactivity related to events on the Web [94, 95, 96]. Condition Actions systems not only direct data flow but emit actions under certain conditions. The rationale behind these studies is an event-based approach which in turn relies on event condition action (ECA) rules. Especially, business decisions base on such a scheme (see also section 2.3 "Data Mining"). ECA rules consist of three different parts:

- **Event**
  An identifier which detects events;
- **Condition**
  An expression which determines whether an action should be triggered or

not;

- **Action**
  A set of instructions which define the reactive behaviour.

Event-based techniques allow users to make use of loosely coupled Web services, which significantly improve scalability in information exchange and distributed workflows. It semantically decouples space, time and synchronisation between event producer and consumer [97]. Thus, the goal is to remove "explicit dependencies between the interacting participants" [98]. Simple events occur at a single point in time, e.g. a website update. ECA Systems may react on simple events, but also complex situations among different Web resources can be detected. A composition of events reflects a complex event, which is multi-layered and has a distinct duration. It would enable Web users to detect meaningful situations consisting of events and reacting on them.

For having full programmability, experienced users tend to use their self coded scripts or applications. Another option would be open source software such as Huginn which is a tool that performs automated tasks in a programmable environment. Such agents are able to download and recognise data from the Web and detect events in order to take actions. Similar to "Yahoo! Pipes" a graph acts as a screenplay for the orchestration of events. [99, 91]

2.2 Web Observatory or Web Observations

The term "Web observation" first emerged in a research paper by Tiropanis, Hall, Shadbolt, De Roure, Contractor and Hendler (2013). They conclude that "we need to observe the Web at scale across space and time" which makes it possible to "understand and enable the evolution of Web to help address grand societal challenges" [100]. Tinati, Wang, Tiropanis and Hall (2015) describe in their Paper "Building a Real-Time Web Observatory" the notion of a platform that provides data sets and purposeful visualisations for the general public [101].

Open or private instances are both common, however, in the science community it is essential to exchange data and information with the help of repositories, in this context called Web observatories. The openness of research within Web observatories acts as a catalyst for more Web science research. Therefore, all data sets in this thesis – except those with privacy concerns – are available on GitHub [102]. Not only does it promote the development of the Internet, but also outlines the state of the Web with regard to resources, links, and activities. Additionally, it does involve the examination of Web resource uptimes, relation-

ships of Web resources and social graphs. The objective is to present models that have a profound impact in the real world and our society. [103]

It is important to differentiate a Web observatory from Web observations. A Web observatory offers location and description of data sets and applications thereon while a Web observation provides the means of extracting Web data for further analysis. The idea of a Web observatory is to spread the collection of data sets for interested people in order to promote its use in other research areas. A Web observation captures and stores data sets containing meaningful information from the perspective of the observer. It focuses on the needs of the observer who wants to gain more information from one or more Web resources. In contrast to a Web observatory, a Web observation collects all states from a selected Web resource within a given observation time. Selection bias is another potential concern because the selection of such a Web resource depends on the interest of the author. It is about finding a Web resource with potential exciting data which is a subjective matter. Web observations may outline and visualise large data sets from multiple sources for a distinct purpose. In the end, both provide data sets, yet each is specified in a certain matter. The work of this thesis focuses on the needs of the observer who wants to understand the mechanism behind the Web resource. [1]

Harnessing the potentials of Web data remains a challenge. Exponential growth and the fact that Web data distribution is unstructured and chaotic makes useful utilisation more and more sophisticated. If neither data sets nor interfaces are available in any form, e.g. Representational State Transfer (REST), users without technical skills are kept in the dark. In contrast to the concept of Open Definition [104], such Web content evaporates after a few minutes and is irrevocably lost. The idea of conservation would prevent losing such Web data and perhaps would be of great benefit for understanding the nature of the Web.

Extensive research has been conducted into Web intelligence, analytics, and Web 2.0-based crowd-sourcing systems [105, 106]. Usual suspects in the public eye are data sets from companies, industries, products and customers. These data sets can be gathered from the Web with various Web mining techniques and may soon afterwards be visualised, e.g. through Google Analytics [107].

There are a number of Web services that provide Web monitoring for change detection and notifications. The first tool for that purpose introduced in 1996 by the company NetMind was “Mind-it”. It spawned quite a few of tools that wanted to revolutionise information discovery and retrieval of Web users. “Google Alerts” still offers monitoring services of Web content with the help of selected keywords. These tools prove very helpful whenever a change of content must be notified as fast as possible. Usually, a history of these changes is not

available. One key problem with Web resources is that Web users cannot verify the quality of a Web service. With the help of a measurement by observation the quality of a service can be assessed. Such a method could check whether the service gives a response and thus is reachable. Another method would focus on the content and highlight relevant content changes. As a result, a Web observation may also take over random samples and assess the quality of Web services e.g. uptime or response time. Furthermore, other aspects such as accuracy may be evaluated. [108]

The Internet Archive stores plenty of Web data in the form of states. There are also tools available, e.g. HistoryTracker, that make use of this archive in order to observe social phenomena through Web observatories. [109]

2.2.1 Publish and Subscribe Pattern

The publish and subscribe pattern consist of a data communication from a sender to a receiver that is not explicitly programmed. The name of this architecture is derived from the setting; typically, the sender is called publisher and the receiver subscriber. The publisher does not necessarily know how many subscribers there are and therefore publisher and subscribers are decoupled. [98]

Adopted by W3C for recommendation is the WebSub protocol, formerly PubSubHubbub [110], which basically is an open protocol for publish-subscribe transmissions. The protocol's main use are real-time or pushed HTTP notifications. It was originally designed for data feeds such as RSS, however, the WebSub protocol supports all sorts of data types that are on the Web, e.g. HTML, pictures, or audio. It may also make use of webhooks to update other subscribers with content. The key advantage of WebSub is that instead of periodically polling for changes, publisher and subscriber spend less resources in spreading information. Furthermore, WebSub enables reactive real-time systems that trigger and aggregate events. [111]

2.2.2 Push Principle

A push notification's main purpose is to couple things with means of a service worker, a type of Web worker that runs within its own context and allows offline capacity. In most cases, a third party server receives sent or "pushed" messages and distributes them to all available sessions for receiving data. The client receives the message directly from the server and does not have to regularly check for an update. One of the biggest advantages is the real-time ability for fast information distribution which costs less traffic and enables one message

to spread among all receivers. Large messages should not be sent over push services, because the server might not be able to handle the volume for all clients and may exceed traffic limits.

Push notifications are fairly common on mobile devices to keep clients up to date for re-engagement on the service. For Web clients, push notifications are a rather new phenomenon; it allows website creators to inform Web users within an opt-in scheme for in-browser updates. In contrast, mobile devices users are automatically served with push notifications. Depending on the mobile device, these push notifications can be switched on or off. Nevertheless, content can be sent to clients within seconds. [112]

Webhook REST Service

A webhook is a REST service that receives HTTP POST requests. Webhooks do not listen to requests, they accept data from a source in a push manner [113]. Thus, they allow real-time multiplication of events. Existing manifestations of webhooks are available for a server to browser communication, such as Comet [114] or server-sent events, see also: http://dev.w3.org/html5/eventsource/. The instant delivery of data streams as per its availability makes this type of data distribution interesting for real-time notifications.

```
1: https://api.webhook-example.com/v2/data.json?
2: email=XXX@YYY.ZZZ&webhookUrl=
3: http://www.myServer.com/MyServiceUrl&apiKey=XYZ
```

Listing 2.1: *Webhook example.*

By entering the URL in a browser, the website "api-webhook-example.com" would send a JSON data stream to the specified URL "www.myServer.com". In principle, webhooks consist of a URL, which points to a Web resource and a sender. It forwards data as soon as it is available within the publish and subscribe pattern [98]. Whenever an update occurs, new data is delivered via a webhook, which passes it along to the precise URL. Therefore, webhooks use Web services as callbacks on remote Web APIs. This makes webhooks advantageous for event-based data flows, e.g. website update triggers a mail. [115]

2.2.3 Poll Principle

A Poll checks periodically for a Web resource to find out whether an update of that particular resource has occurred. As soon as the "polled" resource has been modified for example in size or its content, the updated Web resource will

be highlighted as such. There are multiple parameters to identify whether a resource has been altered, e.g. attributes of the file such as modification date or file contents such as text or images. This process will produce traffic every time the periodic check takes place, even though no new data has been found. Poll applications are fairly quickly created and are often the very first solution in an existing environment. The key problem with polling is that it produces excessive traffic which in turn could provoke a ban by a "polled" Web resource sooner or later. Another problem is the maintenance of the poll that has to match the specified resource. If the resource changes, the application must be adapted because the resource contents are used as the parameter to extract data. This is raising the question of whether the poll is expedient in making scaling efforts and desirable in large Web architectures. It perhaps makes sense with checks an balances in place.

Depending on its use, polling is predominantly used when no API is available for the interaction with the Web resource. Polling is not really a real-time interaction, however, when one knows the behaviour of a resource, a rough estimation about the time lapse of the interval can be made. A good example would be a price comparison during which once a day a particular product can be polled. So every morning at a defined time, the poll would check several online shops for product prices. [116]

2.2.4 Optimal Time Interval

We are confronted with information all over the Web. Unfortunately, the general user is often unaware of changes made to and new information uploaded on a Web resource. Instead, many Web users spend a long time surfing the Web until they realise that new information is available. The challenge remains to know as quickly as possible about these updates of Web resources. Manual capturing of information implies a massive delay in the reaction. Great value would be added for academics and businesses but also for Web users if desired Web content and its information can be automatically retrieved. It would require a new instrument for detecting and reacting to Web content changes. Such a tool would enable Web users to capture, aggregate, and present information in the users' preferred way and would come close to a real-time environment.

Users may be aware of new content on the Web within seconds. More frequently, updates of resources and new information remains undetected for longer periods of time, e.g. hours if not days. To capture such kind of information, a system may provide mechanisms regarding time intervals with the help of polling (see also section 2.2.3 "Poll Principle").

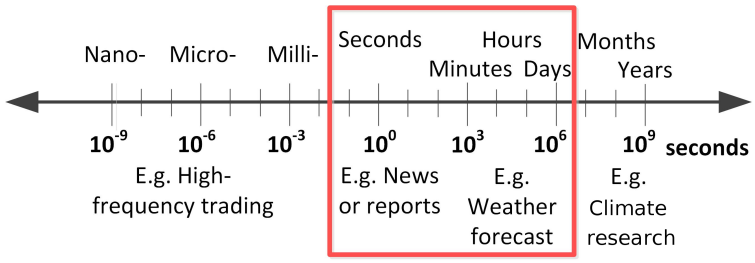

Figure 2.2: *Reaction times of Web observations usually range from seconds to days (red boundary). An interval must be placed for each Web observation. Depending of the area of scope a meaningful interval is given e.g. reports, weather forecast, or press releases.*

Detection Rule

Resources on the Web are in a constant state of flux. Some content might change within minutes, for example news headlines, while other content will be available for a longer period of time such as entries in encyclopaedias like Wikipedia. Content is very much dependent on its owner who is able to make changes at any time. To detect these changes, a Web application must keep track of the document's history. A Web worker monitors particular Web resources and once differences have appeared, an action is triggered. To that extent, it may be only a minor measure to inform about the change, e.g. notification by email. Consequently, a Web user is free to define rules which check whether changes are related to a certain category of interest or not. Moreover, it could highlight certain areas of interest and compare the values that changed. Whenever an article, for example on Wikipedia, has been modified, e.g. more than 10% of its original contents, a notification is being sent by the detection system to inform the person in charge. Such a system is able to directly follow resources on various information spaces.

Therefore, Web detection rules are set in place to monitor Web resources. The idea is to determine whether an information source holds new information. The purpose is to keep track of information on various resources. As soon as an update occurs, users should be informed about new content. Such a detection system can pretty much control the data communication in case new information has been detected.

Web workers operate on the basis of rules that control what shall be observed. As shown in the listing, the rule is encapsulated in a JSON file format. It aims to identify updates and changes on a specific website "URI" containing a specific Document Object Model (DOM) element. When a Web browser loads a Web resource, which contains an HTML file, the browser generates a DOM.

This model creates a tree structure of HTML tags that forms a website. The rule creates the URI and the HTML tag/path or DOM.

```
{
  "id": "Web Detection Rule",
  "event": {
    "DetectWebsiteChange": {
    "uri": "http://www.website.com",
    "dom": "DOM element"
 }
  },
  "actions": {
    "OnWebsiteChange->writemail": {
      "receipient": "mymail@address.com",
      "text": "Content has changed!"
    }
  }
}
```

Listing 2.2: *Example of a Web detection rule in pseudo code.*

In case the Web worker detects a change of content, an email will be sent to the provided mail address. The JSON code instructs Web workers to seek out for new information on any given Web resource. New information is brought forward for the processing of it. In such a case, a predefined action is executed and therefore, a mail will be sent. [91]

2.2.5 Operating-System-Level Virtualisation

Operating-system-level virtualisation, which is also known as containerisation, arranges resources of a computer in isolated instances. These instances are called containers in which additional operating systems (OS) may run from the same libraries as its host computer and makes sharing of a host's resources even possible. Depending on the settings, a container system may share data among other containers for CRUD operations. The vital enabling factor of operating-system-level virtualisation is the operability of many virtual machines in parallel on the very same machine without fully exhausting the machine's resources. For example, if ten containers are running simultaneously, the computer is still able to execute all containers with its resources. Ordinary virtualisation does need all the machines resources even if it is not used. The reason for this big difference is based on the allocated resources to the virtual machine which are most often more isolated and resource-intensive. Container systems try to unravel this isolation by using containers that are using less resources. In practice, hundreds of containers could be run for Web observations without having the problem of too little hardware resources. This gives containerisation

the attributes lightweight and highly scalable. In comparison, a virtual machine needs up to a minute to start up while a container system needs seconds. [117]

Advantages and Disadvantages

Of course there are a few disadvantages to the container system approach. In case full isolation of an OS is required, containers do not make sense. For the isolation of processes containers are reasonable whereas ordinary virtualisation guarantees the assigned resources. In contrast to virtual machines, containers use only those libraries that are necessary for the application and are put together in a container. With the use of an image and commands in the command line the environment is ready. Furthermore, container systems provide basic automation and configuration settings, e.g. Dockerfiles which contain the settings for building images with settings and commands. [118]

A container system can leverage Web observations by its flexibility of choice to mine Web data. Depending on a Web resource a Web observation is more feasible with a certain programming language. Containerisation may hold the key advantage to use the necessary elements for Web observations.

2.3 Data Mining

Data mining is the generic term for the process of discovering patterns or meaningful information. For observing the Web the methods of Web Mining are applied (please see section 2.3.1 "Mining Information from the Web"). Quantitative methods are being used on data which corresponds to the field of knowledge discovery (KD). Most of the methods in data mining are derived from statistics and are generally fitted for the purpose. In this work, the focus lies on experimental methods for practical use on a trail and error basis. In the field of database systems and indexed structures, data mining plays a dominant role in order to reduce complexity, e.g. nearest neighbour search. Another field that benefits from data mining is information retrieval (IR). IR is used for obtaining information from a collection of data, e.g. indexing, document information, and meta-data search. [21]

Many types of data can be obtained from different sources. Data mining not only refers to relational databases, but also to many kinds of versatile data. Good examples for other data types and forms or applications for data mining are sequence data e.g. time series, data streams, continuous sensor data [119], and spatial data such as maps. In economics, a practical use for data mining are sales figures in order to determine how the revenue of a company will be.

Another use for data mining is the stock exchange where stock prices are mined in order to assess when to buy, sell, or hold a certain commercial paper. The idea is to conclude from historical data to the future. The key problem with this explanation is that even if mined data suggests a price increase, many other economic factors are able to turn the tables. Therefore, it is not reasonable to blindly follow historical data. [21]

Data mining involves predefined steps to process data. Depending on the application and availability, each process can be performed in a different order until the desired information emerges, however, the contents of the tasks remain the same:

- **Pre-processing**
  Combination of multiple data sources (data integration) and removal of inconsistent data/noise (data cleaning/cleansing);
- **Selection**
  Retrieval of relevant data of the task for the analysis;
- **Transformation**
  Transformation or consolidation of data into forms appropriate for data mining;
- **Data mining**
  Primary task in which quantitative methods are applied on data in order to extract information;
- **Interpretation or evaluation**
  - **Evaluation of patterns**
    Identification of meaningful patterns or information based on the original data set;
  - **Presentation of information**
    Methods for visualisation and representation of information ready for presentation. [21, 120]

An additional option for the evaluation of patterns is the "exploratory data analysis" (EDA). In the book from 1977, John W. Tukey emphasised the use of data instead of statistical hypothesis testing. EDA possibly will reduce systematic bias by using real data for setting up a hypothesis. [121]

The main objectives of EDA are as follows:

- **Suggestion of hypothesis**
  E.g. observed phenomena;
- **Assessment of assumptions**
  E.g. statistical inference;
- **Support of selection**
  E.g. statistical techniques;
- **Providing foundation**
  E.g. future data collections. [122]

In addition, visualisations still have a strong standing in EDA. It can also be utilised for the formulation and confirmation of data models, the visual assessment of data composition and the identification of occasional rogue results.

The "de facto standard" for practitioners in data mining and KD is the cross-industry standard process for data mining (CRISP-DM) [123]. It basically breaks data mining into six separate tasks in which back and forth iteration is acceptable [124].

2.3.1 Mining Information from the Web

Web mining, often referred to as Web scraping, applies data mining methods in order to discover and to extract meaningful information from the Web. Automated applications are used for the extraction of Web data. As described in section 1.1.2 "Web Data", the vast information space of the Web is fairly unstructured. Therefore, Web mining might give clues about page contents, browser sessions, log-files (browser and server), link structures and other available resources. The mixture of hypertext and media files makes Web mining rather challenging. Text, image, video, and audio files have to be extracted in useful means for individual tasks. Many algorithms are created for the sole purpose of mining the Web. Web observations make use of Web mining techniques to gather states of Web resources at the observation time. [125]

Han, Kamber and Pei (2012) considered Web mining as a future problem that still needs to be addressed, however this work tries to outline an architectural solution for Web observations. Whereas Web mining ought to give answers to the distribution of information on the Web, and the classification of Web pages. Web mining might uncover the mechanics of the Web and tries to

discover behaviour and relationships of "Web pages, users, communities, and activities on the Web". [21]

There are three distinct types of Web mining:

- **Web usage mining**
  Information from interactions, e.g. browser and server log-files;
- **Web content mining**
  Information from unstructured, semi-structured and structured data, e.g. text and hypertext documents;
- **Web structure mining**
  Information from link structure of hypertext documents. [126]

Technical Obstacles

Technical obstacles are in place to prevent automated Web data extraction. To separate humans from machines, it is sometimes acceptable to hinder automation, e.g. to prevent spam. Yet, especially these kinds of mechanisms make the extraction and collection of Web data rather difficult. A popular method to prevent machines from getting to the data is the "Completely Automated Public Turing test to tell Computers and Humans Apart" (CAPTCHA). [127]

After all, these measures do not stop automated extraction but usually cost effort and time until a method is found to get to the desired Web data. Potential circumvention of technical obstacles are for example DOM parsing, natural language processing, or imitation of human-like browsing behaviour.

2.3.2 Extraction Methods

Extracting data from the Web can be undertaken either by human copy-and-paste actions or via software. Most often Web scraping is used in an automated manner with software that is able to access the Web with the help of HTTP. Moreover, it facilitates large-scale Web data extraction. After HTML code and its content have been downloaded, a parser is used for processing the content. Web scraping is often used in gathering mail addresses for mailings, also known as junk or spam. Other applications are for example price comparisons, Web content change detection, and Web mashups. [128]

For example, there are businesses which exclusively exist online. An online shop inevitably results in highly comparable prices, e.g. value of goods and services. When looking for the best deal, it has become rather difficult to

follow all offers manually. With the help of a monitoring system, a potential customer may have the necessary support for having the best offer.

There are several ways to access Web data. In most cases text is encapsulated within HTML tags, however, Web pages are created for Web users and not for software by design. Therefore, Web scraping software makes it possible to extract desired Web content for another purpose. Domain-specific solutions are fairly widespread in Science and Technology. Yet, a different approach is used by regular expressions that use a sequence of characters to define a particular search pattern. This allows automated data extraction without human interaction. Many methods make use of the structure of resources in order to identify useful information, while other approaches derive from the field of machine learning with supervised or semi-supervised learning abilities. [107]

According to Ferrara et al. (2014), Web data extraction techniques can be divided in three major methods:

- **Path/graph/tree-based**
  Structure is known, e.g. XPath or DOM parsing;
- **Web wrappers**
  Part of content is known, e.g. regular expressions;
- **Hybrid systems**
  Mixture of above, e.g. template-based matching. [128]

Perhaps the most serious disadvantage of Web wrappers are that they are tightly attached to the Web content. Whenever websites change the layout significantly Web wrappers will no longer work. A possible solution for this problem may be rule or learning-based approaches that are able to extract desired data independently.

Nevertheless, Web data extraction software should be able to get, store, analyse, transform and visualise Web data. The idea thereof is to set up a collection of data, facts and figures which must still be comprehensible, even over a long period of time. Periodically extracting one simple value from a website may not be a big challenge. Yet, problems may appear regarding more distributed data sources or Web resources that do not have API capabilities. [79, 129]

2.3.3 Web Mining Algorithms

To access the data of Web resources, many Web mining algorithms cab be used. Web mining techniques are key components within the constraint of a

programming language or library for this task.

Basic Web Observation

A server with access to the Internet can be used for all kinds of Web observations. It is a shift towards server side applications for having the flexibility of different environments and manifold tasks. The final choice of components is up to the creator: programming language (Code), database software (Storage), file format, e.g. RDF, (Data Sets), and libraries for graphical representation (Visualisation). The Web client plays an essential role in exploring a Web resource. The goal is to find meaningful or interesting pieces of information. In many cases it might be data that is in a constant flux and has a certain value to the viewer. Web observations want to make use of content from the Web.

As follows a feasible basic architecture for a Web observation:

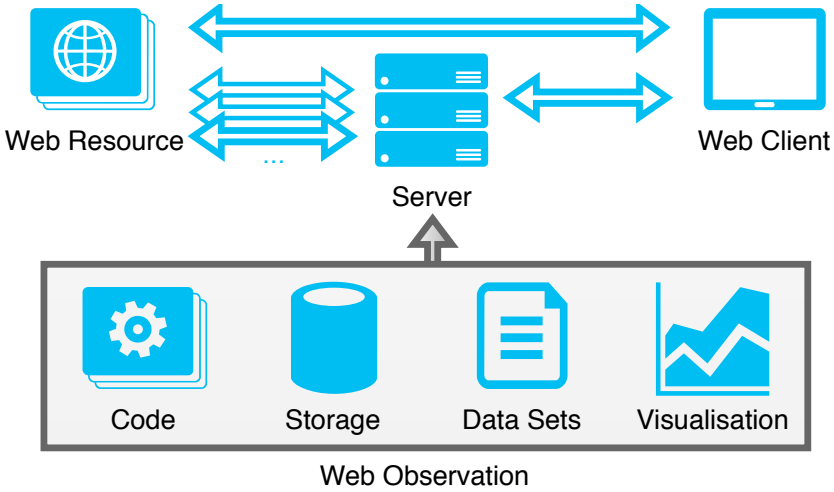

Figure 2.3: *Basic Web observation architecture in which all components are stored on a server. [79]*

However, the server hosts the key components that define a Web observation. It may consist of actual programming code (Code), a database system (Storage), a file format (Data Sets), and graphical representation (Visualisation). The code contains the actual Web observation which will then store its information in a database. From this database, data sets may be automatically generated which in turn can be accessed from a Web client for further analysis of data.

It is common sense to run a Web mining application from a server that sorts out computing resources, e.g. database storage, run times, and other matters in order to properly collect Web data. The idea is to include all the necessary

means on a single server; programming code, storage solution, data sets and visualisations thereon.

The implementation part of a Web mining algorithm varies considerably depending on the used language and libraries. Already available libraries and existing code snippets make programming tasks far easier than starting from scratch. Although, used functionalities may differ, the result remains the same. However, it may be that special characters might escape. Each library uses different dependencies of other libraries that have to be pre-installed.

BeautifulSoup in Python

The lines of code below provide a basic Web mining algorithm that outlines the functionality in Python:

```
1:  #Python
2:
3:  #import library 'urllib2'
4:  import urllib2
5:  #import the Beautiful Soup functions for parsing
6:  from bs4 import BeautifulSoup
7:  #specified url stored in variable 'wikiurl'
8:  wikiurl = 'https://en.wikipedia.org/wiki/Liz_Bacon'
9:  #query the wikiurl and return the HTML code to the variable 'html'
10: html = urllib2.urlopen(wikiurl)
11: #parse the data returned from the website to the variable 'parser'
12: parser = BeautifulSoup(html,'html.parser')
13: #parse 'html' variable and store it in Beautiful Soup format
14: soup = BeautifulSoup(html)
15: #print the result of Beautiful Soup
16: print soup.prettify()
```

Listing 2.3: *Pyhton Web mining example.*

These few lines of Python code facilitate Web mining of Wikipedia contents. Output of the variable "html" is the whole HTML code of a specified Wikipedia page. Additional code such as "list(soup.children)" could parse the content and would simplify the selection of desired content.

Long-time Web observations are seeking for information of high significance. Collecting data that points to facts in the real world might be a good idea. Meta data such as moving geolocations are particularly interesting to observe. It is therefore a good idea to capture movements of things in the real world. These movements are often stored within HTML or XML source codes and have to be sought and found. The library "Beautiful Soup", or short "Soup" makes this task rather easy; it composes precise content strings from long HTML codes according to given parameters. Following the code snippet for accessing child elements "geo" including "latitude" and "longitude":

```
1: #Python
2:
3: #get all elements 'geo' and return them to the variable 'geo'
4: geo = soup.find('geo')
5: #get latitude from 'geo' elements and print them out
6: print geo.latitude.get_text(strip=True)
7: #get longitude from 'geo' elements and print them out
8: print geo.longitude.get_text(strip=True)
```

Listing 2.4: *Access child elements 'geo', 'latitude', and 'longitude' in Python.*

Soup is also able to eliminate whitespaces from the beginning and end of each text. Therefore, the method is extended with "(strip=True)". This Python example outlines which section of the Wikipedia content should be the focus of the data collection.

Needle & cheerio in JavaScript

There are a number of important differences between Python and JavaScript. JavaScript may have a similar syntax, however, they are distinctive in regard to event driven and asynchronous capabilities. Therefore, code executes further lines even if a function has not yet returned [130]. Node.js creates a powerful runtime engine for JavaScript which fits perfectly for Web observation tasks. As follows a basic Web mining script that outlines the functionality in JavaScript:

```
1: //JavaScript
2:
3: //import libraries 'needle' and 'cheerio'
4: var needle = require('needle');
5: var cheerio = require('cheerio');
6: //store Google url in variable 'googleurl'
7: var googleurl = 'https://www.google.com/search?num=100&q='
8: //specified search string stored in variable 'searchString'
9: var searchString = 'Alexander Groeflin';
10: //query to Google plus searchString with the library 'needle'
11: //using callback
12: needle.get(googleurl + searchString, function(error, response) {
13:   //if HTTP status code '200' OK
14:   if (!error && response.statusCode == 200) {
15:     //return HTML body 'response.body'
16:     console.log(response.body);
17:   }
18: });
19:
20: doSomething();
```

Listing 2.5: *JavaScript Web mining example with needle callback.*

The function "doSomething()" on line 20 does not have to wait for results from the callback function "needle.get()" on line 12. Moreover, asynchronous

JavaScript is very common in Web environments. There are many advantages in terms of expensive and/or time-consuming operations. In contrast to synchronous code, however, asynchronous code keeps on running further lines of code and thus possibly will not make a Web server unresponsive. It simply does not have to wait for the return of a called function and proceeds with the execution of code. [130]

A useful library for Web mining tasks is needle. In the example above, needle requests the particular string "Alexander Groeflin" in the search engine Google. For that reason the string is used in the needle callback method "needle.get()". Needle facilitates HTTP requests by using the Node.js environment, e.g. for API, data streams, and event handling. With a callback, the response body will be buffered and written to "response.body"; the function will only start when all data has been collected and properly processed. No callbacks are used for specific file retrieval tasks. In the case of documents and files, a stream of data is the preferred choice in which no buffering functionalities are necessary. [131, 132]

As follows a needle algorithm using stream functionalities:

```
1:  //JavaScript
2:
3:  //name of file 'easyjet_BSL-LGW.json' stored in variable 'fileout'
4:  var fileout = fs.createWriteStream('easyjet_BSL-LGW.json');
5:  //query to GitHub with the library 'needle'
6:  //using streams
7:  needle.get('https://github.com/WebObservatoryUnibas/lab/raw/master
        /easyjet_BSL-LGW.json').pipe(fileout).on('end', function() {
8:    //signalling end of task
9:    console.log('stream finished!');
10: });
```

Listing 2.6: *JavaScript needle stream.*

As described before, parsing libraries such as BeautifulSoup makes a programmer's life easier. The library Cheerio offers parsing functionalities for the composition of markup language structures. To achieve the objectives of the Web mining task, simple tag selectors such as CSS selectors, CSS path, or XPath can be very helpful. Selectors are capable to filter out content of a Web resource for example the headlines of a Google search and thus are able to observe the page rank. The code snippet is as follows:

```
1:  //JavaScript
2:
3:  //query to Google incl. searchString with the library 'needle'
4:  //using callback
5:  needle.get(googleurl + searchString, function(error, response) {
6:    //if HTTP status code '200' OK
7:    if (!error && response.statusCode == 200) {
```

```
8:      //load HTML body 'response.body' in cherio
9:      //return it to variable 'html'
10:     var html = cheerio.load(response.body);
11:     //select all tags 'h3 > a' and return them to the variable '
            searchResultsHeadlines'
12:     var searchResultsHeadlines = html('h3 > a').text();
13:     //print out variable 'searchResultsHeadlines'
14:     console.log(searchResultsHeadlines);
15:   }
16: });
```

Listing 2.7: *JavaScript needle and cheerio combination.*

These kind of algorithms run in the command line/terminal once it is executed. A repetitious execution of such a Web mining task would be exactly what a Web observation wants to achieve. Consequently, a resilient method must be used for the collection and storage. One available option is the method setInterval() which calls a function at specified intervals (in milliseconds). The method will continue to run as long as the method clearInterval() is called. Concluding, JavaScript facilitates event timing with the help of time-intervals as follows:

```
1: //JavaScript
2:
3: //function
4: doSomething(
5:     //code is executed on startup and once every 4 hours
6:   );
7:
8: //milliseconds: once every 4 hours
9: setInterval(doSomething, 1000*60*60*4); //14400000 milliseconds
```

Listing 2.8: *JavaScript setInterval() method.*

As shown above, the method "clearInterval()" is not part of the code. This means that the code would run continuously until further user input is given. For the use of interval methods it is required to have an ever running command prompt, e.g. Linux environments. A major problem with this kind of application is unexpected errors in the Web mining process which may crash the whole program, for example a JavaScript TypeError. Sometimes different data types are collected, e.g. a string instead of number data type, which may also stop the program and result in an error "NaN", Not-a-Number. These likely events may accidentally invalidate a Web observation because of a lack of data. Such errors have to be handled explicitly.

The JavaScript method for intervals is a simple option and proven method to repeat Web mining tasks. Account should be taken into the maximum workload of a collection algorithm. Larger data collection problems that occur in Web observations with many different values should not be handled with these kind

of methods. Complex applications with many lines of source code will make it hard to manage.

Cron Jobs

A good alternative is cron jobs which is a utility for Unix operating systems. It schedules commands and collection algorithms within the operating system which makes maintenance less time expensive. Cron jobs are usually used for periodic execution but also on fixed events such as a reboot. The JavaScript library "cron" enhances the node.js environment with cron job functionalities [133].

This chapter described technologies behind Web observations and clarified the difference between Web observations and Web observatories. The next chapter "Research Plan" will provide the applied research methodologies.

Part II

Methodology

Chapter 3

Research Plan

> Plans are of *little importance*, but planning is *essential*.
>
> — The Right Honourable Sir Winston Churchill, former British Prime Minister, (born November 20, 1874 – died January 24, 1965)

This chapter provides an overview of the research methodologies applied in this thesis. After highlighting the research contributions a refined research methodology including research problem and research questions is given.

3.1 Research Contribution

The objective of the author's research is to develop Web observations that gather data of interest from the Web. Moreover, an architecture shall be created that gives the user a reusable structure for conducting Web observations. In addition, a focal point is the actual Web observation task which should have the flexibility to switch between programming languages, software environment and operating system (OS). It is important to allow this flexibility of choice because of the diversity of the Web with its manifold appearances. From the operating system to the actual programming code components should be in a self-defined order. The architecture should be able to bind software together like building blocks.

For this reason the process of examining the actual Web resource of interest shall be described and outlined. Furthermore, implemented architectural approaches will be outlined in order to determine what possibilities are fitting best to the given research questions. Main planned research contributions of this thesis include:

- Literature review of data aspects, technologies for Web observations, legal and political perspectives, as outlined in the chapters before;
- Definition of inspection tasks undertaken before a Web observation may actually begin;
- Recommendation for an architecture for Web observations.

The goal is to create an architecture that groups together software packages and programming languages based on a modular approach. For collecting and storing data from Web resources, there is a need for a toolbox with reusable parts. No general algorithm currently exists to collect data from any given Web resource. Therefore, Web mining tasks have to be adaptable to the Web resource. Essentially, the intention is to facilitate Web observations by providing reusable components for the process of collecting Web data which allows to measure and monitor events on the Web.

3.2 Research Problem

The objective of this thesis is to facilitate Web observations of Web resources. After having assessed available literature, it can be said that current state of the art services and tools are not able to collect Web data in all cases. Various Web services and systems are available but did not have the ability to gather desired Web data. Current services do not fully deliver this functionality for Web data extraction. Available Web mashups did enable some Web observations, but by growing complexity they cannot deliver data extraction capabilities. Condition action systems are a promising part that must be tackled with this work. That is the reason why the author was forced to conduct his own research and development in this area. Thus, the author defined a non-exhaustive list on the kind of data a Web observation might be useful to collect:

- Web service events or Web change detection;
- Measurement of values, e.g. number of movements, data, and availability;

- Verification of data and prediction, e.g. verify weather forecast;
- Web resource uptime, e.g. meta data and service performance.

For a technical solution that is able to collect all these different kinds of data from the Web, a flexible building block architecture must be developed. Based thereon, meaningful data possibly will be collected over a longer period of time. Real-life data such as temperatures or geo location based services are of particular interest. However so far, no Web resource has been selected as a useful source. Therefore, it is necessary to find a fitting head start for a Web observation. Numerous data sources on the Web will make it a difficult choice to identify use cases for Web observations. The criteria of using Web observations are often unclear. Initially, it was the preparation for a conference that provided the author with a useful first scenario for knowing exactly when deadlines have been updated. With this in mind Web observations gained momentum for this particular use case. While monitoring flight tickets for a cost effective offer, the author is determined to apply Web observations to other Web resources. However, interesting Web resources have yet to be found which is also an important aspect to solve.

3.3 Final Research Questions

Based on the literature review outlined beforehand, the original three research questions illustrated in the introduction are still considered highly relevant. However, the last research question did not meet expectations in terms of precision: *"What kind of Web data may be the most interesting for a Web observation?"*. It is extremely vague to clearly determine what is considered "most interesting". The question has been adjusted in order to find scenarios that are particularly suitable for Web observations. As follows the final research questions including further remarks:

- **Is it legally allowed to systematically collect Web data without anybody knowing about it?**
- **How to design a flexible architecture that is able to collect data from desired Web resources over a long period of time?**
- **What kind of Web resource scenarios are particularly suitable for Web observations?**

The first research questions deals with the legal challenges arising with data collections. The current situation regarding international data movements and data ownership will be raised. Furthermore, the new legal framework GDPR will be introduced into the subject of Web observations.

The second research question outlines the development of an architecture that is able to give a choice between different software packages. This can be answered by describing an essential tasks for the extraction of Web data. Moreover, an architecture has to work for all sorts of Web resources on the Web. A flexible architecture is able to extract data from all sorts of environments. It is clear that a Web mining programming code must be adjusted to the needs of the environment. In the end, a robust Web observation solution must be able to work over a longer period of time, so it can gather different states of a Web resource and its changes.

The third research question points out scenarios that must be identified and conducted for Web observations. For example the Web changes every millisecond, but which events and values are particularly interesting for the given research? These different scenarios have to be developed, tested, and described to answer this questions. By designing an architecture, it is essential to test and evaluate it. Part of such an evaluation could be an in-depth analysis of the results to prove that it really can provide useful data. Therefore, Web observation architectures must be used for real scenarios. The scenarios in turn collect and create data sets that may give insights of what happened to a Web resource over time and perhaps explains what mechanisms play a role within a Web resource. Furthermore, collected data sets may hold valuable information for Web users.

This chapter described the research methodologies applied and expected research contribution, the research problem and final research questions. The next chapter “Legal and Political Perspectives” will provide an overview of legal issues and challenges.

Part III

Law & Data

Chapter 4

Legal and Political Perspectives

> Any big *Internet companies* in Europe? No, why? They *worry too much*. Oh, privacy, oh, security, oh, rules and laws. Before they do that, they just bring all the worries inside. [...] But you never *solve the problem* and then *go do it*...
>
> — Jack Ma, Founder and Executive Chairman of Alibaba Group (born September 10, 1964)

This chapter gives an overview of legal issues and challenges in Switzerland, the European Union, the United States, and the United Kingdom. From an economic perspective, the European Union and the United States are very important trade partners for Switzerland. The United Kingdom already started to implement regulations deviating from the European Union which will be of interest as soon as the United Kingdom evolves into an independent trade partner. Moreover, it outlines political issues such as open data access and predictive policing. Perhaps even more interesting, this chapter tries to proclaim a general recommendation in a field that has much trouble understanding technical issues but still wants to govern them.

4.1 Introduction

Modern day legislation is faced with the difficult task to fit fast changing technological concepts into legislative acts or ordinances. However, both national legislators and courts are often faced with questions that are difficult to understand for lawyers or other non IT professionals but still need to be decided or governed. Therefore, several national and international legislative bodies are working on revisions or new acts to regulate data, its processing and transfers or the Internet as such. Yet, the legislative approaches widely vary from nation to nation. Often, private law issues are differently handled than data collections in a criminal investigation. Additionally, with companies growing bigger and bigger, it is not only the general protection of data that worries legislators. There are big conglomerates and multi-national companies without clear structure that make data protection even more challenging, considering the growing difficulties in anti-trust regulations. [134]

In the following, a short overview on current political positions, legislative projects and case law from Switzerland, the European Union, the United States, and the United Kingdom shall give an overview on these different solutions chosen for a problem that is still difficult to grasp for legal experts.

4.2 Legal Issues

Modern data protection laws have to deal with two major but diverging interests. While the protection of sensitive personal data is the main goal of protection settings in both private and public law, there is a strong interest of investigatory agencies to know who can be held responsible for harmful actions. Countries have chosen different approaches to deal with these issues, while the European Union has a strong focus on securing a person's private information through governmental regulations, the United States highly value their privacy and people prefer to be left alone rather than having governmental interventions imposed on their private life. Considering that, cross-border data transfers can not only lead to a violation of a national law, it could also cause international issues due to different standards in regard to handling of data. Such discrepancies should have been avoided through the so called "Safe Harbour" practise between the European Union and the United States. However, after 15 years in which American companies could register to be considered as properly applying data protection regulations also recognised in the European Union, the European Court of Justice invalidated this agreement. [135]

Although a new EU-US Privacy Shield agreement had been established shortly thereafter, the European General Data Protection Regulation (GDPR, Regulation (EU) 2016/679) as well as two major cases the United States Supreme Court (Carpenter v. United States and United States v. Microsoft) will definitely influence the future international handling of data transfers and data processing. [136]

4.2.1 Switzerland

In Swiss law, there is no clear definition of what sort of information falls under "data". According to the Federal Act on Data Protection (FADP, SR 235.1), data is considered to include all personal information of an identified or identifiable person (Art. 2 lit. a). Yet, data cannot only be found in situations where a private or an administrative body is processing information (Art. 1). Data is also a crucial part of criminal proceedings and while disclosure of data collected in a private law setting may be unpleasant for that respective person. In a criminal investigation the access to a person's data can – if disclosed – cause even more harm than just a personal discomfort. [137]

The technological changes of recent years gave way to several revision projects by the Swiss legislator. In 2011, the very first Federal act on Criminal Procedure was introduced in Switzerland. Beforehand and thanks to the federal system of the country, each and every canton have had its own Criminal Procedure Code. After having realised that modern day law enforcement issues can partially be banned through uniform procedural legislation, the Swiss Criminal Procedure Code (CrimPC, SR 312.0) was enacted. However, like with any new legislative project, only practice could tell the Code's strengths and weaknesses. Considering the latest global events – specifically the strong use of digital means in crimes like drug trafficking, human trafficking (mainly done over the Darknet) and terrorism (e.g. propaganda on social media) – the CrimPC has recently been revised. [138]

As of March 1, 2018, the CrimPC was expanded with specific regulations on how and when access to a cell phone or a computer may be conducted by law enforcement to conduct covert surveillance. According to these new Articles 269bis, 269ter and 269quater surveillance is now allowed through special technical devices or specific observation software to collect both content and meta data. More precisely, law enforcement agencies may now broadly listen and record conversations or intercept and recover data such as location information. However, these rules are considered to be the ultima ratio of an investigation. Only in very specific cases, law enforcement may go as far as fully observe a

suspect under these new regulations. Therefore, these measures may only be ordered if (i) all other investigative techniques have failed so far, (ii) there is probable cause against the suspect, (iii) the crime to be investigated is a major offence and (iv) police was issued a written warrant specifying the duration and scope of each respective interception. [139]

Yet, the additions to the CrimPC were not the only revisions done by the Swiss legislator to adopt the technological changes into its laws. Also, as of March 1, 2018, the fully revised Federal Act on the Surveillance of Postal and Telecommunication Services entered into force (SR 780.1). As a complementation of the CrimPC, the Federal Act on Surveillance of Telecommunication Services and its respective ordinance list all the acceptable technical means that can be used for the surveillance of postal and telecommunication services. Strikingly, while there are about forty different means to intercept a telecommunications device, the act only offers three acceptable means for the surveillance of postal services. [140]

The new regulations mainly govern the acceptable way of requests for disclosure from third party service providers, real-time and retrospective surveillance techniques, distress searches and tracing. Especially the new rules on real-time surveillance are now adopting certain standards set by the Council of Europe in its Cybercrime Convention. [141]

Also, the new rules include a right to request disclosure of information on participants in an open WiFi. However, for a service provider to know who participated in its service, open WiFis will be regulated in the future by registration requirements that allow the service provider to collect at least some information on its participants. Interestingly enough, if a private person offers access to his or her open WiFi to their friends, they might also be considered a third-party service provider that would need to disclose information on the network's participants. Yet, only time will tell whether a private person will ever be in a position to disclose such data. Also, due to the relatively short period of time since the enforcement of these new legislative acts, it is too early to tell how important the single regulations will be in the daily business of law enforcement and judicial agencies. [142]

4.2.2 European Union

Fast technological changes compared with new threats of privacy intrusion and cybercrime were just some reasons for the vast legislative actions on data by the European Union. In 2016, the European Union enacted several legislative acts to generally improve data protection and facilitate the transfer of data for

police and judicial agencies in the Union.

Specifically, (i) the Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation); (ii) the Directive (EU) 2016/680 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data by competent authorities for the purposes of prevention, investigation, detection or prosecution of criminal offences or the execution of criminal penalties, and on the free movement of such data; and (iii) the Directive (EU) 2016/681 of the European Parliament and of the Council of 27 April 2016 on the use of passenger name record (PNR) data for the prevention, detection, investigation and prosecution of terrorist offences and serious crimes shall constitute the new legislative framework on data handling inside and outside the European Union. Each of these legislative acts will be presented in the following.

General Data Protection Regulation

May 25, 2018 is an important date in international data protection legislation. From this day on, the General Data Protection Regulation (GDPR) of the European Union (EU 2016/679) is applicable to all private bodies inside and outside the European Union that collect, hold, transfer and/or process data. With this new Regulation, the European Union has created legislation which reaches outside the Union's borders. Therefore, in the future, it will not matter anymore whether a (legal or natural) person is located inside the European Union to fall under the GDPR. Rather, the Regulation will target anybody who – if even only remotely – deals with data located inside the Union. This mainly includes companies that trade with customers in the European Union.

That is the reason why many companies have started to update their privacy policy and/or terms of service in 2018. A cascade of updates has been recently noticed by many Web service users:

```
---------- Forwarded message ----------
From: The Turnitin Team <communications-eu@turnitin.com>
Date: 2 May 2018 at 19:31
Subject: Important updates to our Privacy Centre and Terms of
    Service
To: alexander.groeflin@unibas.ch

We've updated our privacy policy

To support upcoming changes to European data protection law,
```

```
we've focused our efforts on refreshing our processes around
how we use your data.
[...]
```

Listing 4.1: *Mail from Turnitin. [143]*

Another example for this cascade of customer contact in relation with the GDPR is the airline company easyJet. It sent out an mailing to all customers that have had booked a flight in the near future:

```
---------- Forwarded message ----------
From: easyJet <generationeasyJet@email.easyjet.com>
Date: 3 May 2018 at 19:49
Subject: Your booking *******: Alexander, you're always in control
     - on your upcoming trip to [.] and beyond
To: alexander.groeflin@unibas.ch

SAY YES TO OUR PROMISE, NOW AND ALWAYS

Hi Alexander,

At easyJet we believe in being upfront with our customers.

We use your personal data to keep your flights safe, easy and
affordable. We collect and store it securely, and we'll only
use it with your permission.

[...]
```

Listing 4.2: *Mail from easyJet. [144]*

And while the Regulation not only introduces an extraterritorial reach, it also fully regulates any action taken by a targeted person, no matter whether that action is connected to the handling of data from inside and outside the European Union. [145, 146]

To clarify the above said, a practical example from a pharmaceutical company in Switzerland shall shed light on the Regulation's scope.

For example, a pharmaceutical company normally deals with a lot of data that falls under the category of sensitive personal data, meaning all data that can be used to create a personality profile and that includes information such as a person's state of physical and mental health, wealth and daily habits. In general, no such personalised sensitive data may be collected or processed without the consent of the respective person. A consent to the collection and handling is only valid, if the respective person has been properly informed about issues like "why", "how much" and "where to" of the data collection and if based thereon the person has freely consented to the disclosure of the data (so called "informed consent"). If the data has been lawfully collected, it may then be further processed and transferred.

This means that a person acting as a trial subject does not only need to be fully informed about what pharmaceutical treatment he or she will receive. The person will also have to be informed about what kind of data from that respective person will be collected, where that data will be transferred to and what will happen to the data. Mainly, these questions will be part of the consent form that the trial subject will sign before any study. Still, such informed consent will be of utmost importance for future proper data handling under the GDPR. [147]

Considering that the pharmaceutical company complied with the Regulation's rules on proper data collection and transfer, it further needs to ensure that the data will be properly handled after its correct collection. This includes organisational changes like the creation of the position of a Data Protection Officer (DPO) in every processing body, the general safeguarding of any sensitive personal data as well as the introduction of systems that ensure data safety and the immediate discovery of any data breach.

As mentioned above, if a company falls under the GDPR due to having customers or other trading relationships with the European Union, this also means that the pharmaceutical company in our example has to generally ensure that its SOPs, contractual relationships and the actual daily business are always in line with the European data protection rules. Therefore, the company will also need to ensure that their relations to any American company are in compliance with the GDPR. If a company is in violation of the GDPR – be this e.g. through improper handling of data or a general data breach – this can have severe consequences for that company. Depending on the severity of the breach, fines of either (i) €10 Mio. or 2 percent of the global annual revenue or (ii) €20 Mio. or 4 percent of the global annual revenue can be imposed, whatever amount will be higher.

While this new Regulation puts a lot of pressure on companies to be compliant, it gives a single person new rights. Based on the GDPR, a person may request the disclosure on any and all data that was collected by a data collector (be this a legal or natural person). Additionally, a person may always request that any data collected shall be corrected or deleted, meaning that there shall be a "right to be forgotten".

In general, the European Union tried to create a broad scope for data protection which is specifically supported by civil rights advocates all over the world arguing that only under the rules of the GDPR full security of personal data can be ensured. However, the very strict rules and especially the high fines did make the implementation of the Regulation difficult and will be a constant future threat in regard to the upholding of compliance. [148, 149, 150, 151]

A privacy campaigner, Max Schrems and his lobby noyb "None of Your Business", filed complains against Google, Facebook, and their European subsidiaries on the starting day of the applicability of the GDPR. Schrems who already filed suits against the Irish data protection commission which led to the revocation of the Safe Harbour rules between the European Union and the United States is now eager to take on big tech firms under the GDPR. His claims were filed in Austria, Belgium, and Germany but will eventually be transferred to Ireland due to the GDPR's rule on so called "one stop shop mechanisms" which create jurisdiction at the European seat of incorporation of a company. His activism will test European interaction and the value of the GDPR from its very beginning. [152, 153]

Influence on Swiss Legislative Projects

The European Union, additionally to just physically surrounding Switzerland, is the biggest trading partner of the Swiss Confederation. Therefore, it is of utmost importance to the Swiss government to be considered as a state with a comparable level of data protection regulations. In accordance thereto and also in consideration of the enormous technological changes of recent years, the Swiss legislator took on a full revision of the Federal Act on Data Protection.

The revision considers both the GDPR and the Directive (EU) 2016/680. Specifically, similar rules in regard to transparency in data collection shall be implemented. However, while an alignment of rules and the provision of clear information on any data collection is of importance for future trading relations with the European Union, the Swiss legislator did not include fines at the same amount as the GDPR. Violations of the new Federal Act on Data Protection might still be expensive for some organisations, but with a maximum amount of CHF 250,000.– the fines are far away of the millions of euros that can be imposed under the GDPR.

In September 2017, the Federal Council published the official draft of the revised Federal Act on Data Protection with its official communication. Next, the draft will soon be discussed in the Swiss Parliament. However, there is no date yet set for the discussions. Although the GDPR is now in full force and applicable, the Swiss alignment – also known as "Swiss finish" – will still need some time to enter into force. [154, 155]

Influence on Swiss Companies

As discussed in the practical example in regard to the extraterritorial reach, Swiss companies are especially under pressure to align their course of business

with the data protection rules in the GDPR.

A Swiss entity incorporated in Switzerland will need to review whether its SOPs are sufficient to fulfil the requirements set by the GDPR. While this might not be of utmost importance to a small business in Switzerland solely working and trading locally, almost any company that is located close to the border irrespective of its size can possibly fall under the GDPR. It is already sufficient that an employee is a cross-border commuter, because already then, the company will collect and process information from a party living inside the European Union. Swiss companies will similarly need to create the position of a data protection officer as well as review all their legal and working relationships to be in line with the GDPR. And while the revised Federal Act on Data Protection might still be years away from enforcement, the GDPR and its enormous fines are sufficient to make companies want to comply. Switzerland is therefore the perfect example of how the European Union extended its legislation across its borders and into foreign territory ignoring all basic international principles and rules of state sovereignty and territorial powers. It might be difficult for a small jurisdiction like Switzerland to do anything else than comply – the fear of being fined is just good enough as an incentive – it is still questionable whether the future shall be governed solely by big jurisdictions no matter how good their intentions. [156]

Directives (EU) 2016/680 and (EU) 2016/681

While the GDPR is part of every major headline at the moment, there is another piece of European legislation, that needs to be mentioned in regard to the future handling of data. The Directive (EU) 2016/680 regulates the handling of a natural person's data in regard to data processing by authorities in the investigation, detection and prevention of crimes. [157]

In comparison to the GDPR, which was enacted in 2016 in the form of a regulation – meaning the rules therein are directly applicable in any Member State – here the form of a directive was chosen. This means, that the rules formulated therein have to be transferred first into national law by each Member State to be applicable while the Directive itself only provides the framework.

A directive leaves room for national considerations, which in regard to legislation that affects police as the main enforcing power of a state is of utmost importance. The Directive (EU) 2016/680 was created as a public law regulation and in answer to the Paris terrorist attacks in 2015 and while it shall provide a general European Standard of how data shall be handled in police investigations, it shall still allow the Member States to keep their respective

procedures within their law enforcement agencies.

The Directive tries to implement latest technological changes into the data handling done by law enforcement. Therefore, it shall mainly facilitate the free movement of data processed in the prevention, detection and investigation of crimes. Furthermore, in comparison to the GDPR, the Directive does not allow the same broad access to collected data. While there is a legitimate interest to allow people full access to their collected data in a private law setting, there is also a legitimate interest to keep people from fully accessing data that is handled by law enforcement agencies in investigative matters.

In regard to general transfers of data between Member States and non-EU states, such transfers are allowed to be done under the Directive. However, transfers can only be made to states that are accepted by the European Commission to provide the same level of protection to sensitive personal data as provided under European law. This consideration is a direct outflow of the decision of the European Court of Justice in the case of Schrems v. Data Protection Commissioner (Case C-362/14). Another addition to a general European framework for data handling by law enforcement is provided in the Directive on European Passenger Name Records (PNR). According to this Directive (EU) 2016/681 data must be transferred from air carriers flying from outside the European Union into the European Union to the respective Member State. [158]

Although the Directives are considered to create a general framework for data protection in law enforcement, experts are anxious to yet confirm such an outcome. While the adoption of a directive can be postponed to a certain extent, it should further be noted that the Directives include only police and justice actions. Any data transfers done by an administrative body or by the Union's institutions might possibly fall neither under the GDPR nor under the Directive (EU) 2016/680. [159]

4.2.3 United States

Currently, the US Supreme Court has to decide in two cases that will be leading in regard to how data will be handled in future investigatory processes, be this either in national or international investigations.

Carpenter v. United States

The first case, Carpenter v. United States, is considered by some scholars as the most important case to be decided in this decade. Carpenter was a criminal who could be connected to several armed robberies in two different states thanks to

the location data from his cell phone. Investigatory agencies had served warrants to several cell phone service providers to hand over location data of Carpenter's cell phone for the specific days on which the robberies had been committed. Thanks to the data disclosed by the service providers the investigatory agencies could prove that Carpenter had been present on each day of a robbery in each of the cities and always close to the crime scene. Based on this data and other evidence, Carpenter had been convicted and given a prison sentence of 116 years. Carpenter appealed his case all the way up to the US Supreme Court arguing that the service provider should not have disclosed his data because it should be considered part of his personal papers which would be protected by the Fourth Amendment. Therefore, his Constitutional right to be free of any unlawful search and seizure had been violated. According to Carpenter, cell phones are crucial to take part in modern day society, therefore, the meta data collected by his service provider cannot be considered as voluntarily given to the service provider. Thus, the service provider should have asked for Carpenter's consent before handing over his location data to the investigatory agencies. Based on this argument, the location data should have been excluded from his initial process and no conviction should have been made. [160]

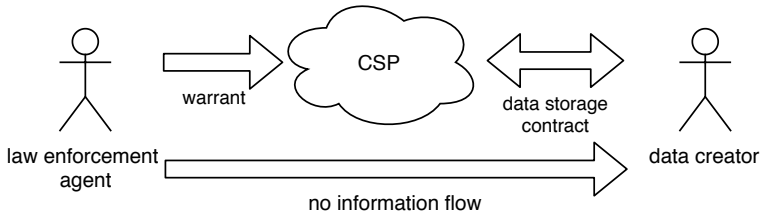

Figure 4.1: *Law enforcement agent requests data disclosure from cloud service provider (CSP) safekeeping information of its subscriber in its electronic communication or remote computing service.*

While Carpenter argues for a violation of the Fourth Amendment, the Solicitor General of the United States considers this case to be in accordance with legal precedents. Specifically, the Solicitor General points out that according to Smith v. Maryland (1979), meta data collected from a service provider with a pen trap does not consider a violation of a person's Fourth Amendment rights, if the service provider had been served a proper warrant. Back in the days, the US Supreme Court held that pen traps might be able to collect information such as the phone number a person had been calling from and the number it had been calling to as well as the time and length of a call. However, because no content data could be collected by a pen trap, the collection of meta data was considered to be in line with the Fourth Amendment. The Court had also

pointed out that a person was generally aware that a telephone company would be collecting a certain amount of data and that employees sitting on the switch board of a service provider could even listen to phone calls.

While the US Solicitor General might be right with his understanding of Carpenter's case, it is noteworthy that several Justices during oral arguments in November 2017 had asked their questions in a way to keep open the possibility for a narrow decision of the case. The US Supreme Court would then follow his latest case in regard to searches of electronic devices, Riley v. California (2014). In Riley, the Court had to decide whether the search of a cell phone found in a car was acceptable under precedents which would allow searches without warrant of containers in cars. In 2014, the Court decided to take a narrow decision supporting Riley in his arguments that cell phones provide much more information on a person than a grocery bag stored in a car. Therefore, the search of a cell phone found in a car cannot be conducted without having a proper warrant for that search. If done so anyway, this is considered a violation of the cell phone owner's Fourth Amendment rights. [161]

Carpenter v. United States is a case that could go either way. The Justices have not given hints whether they wanted to take a narrow decision similar to Riley v. California or whether they wanted to stick to the Court's own present, Smith v. Maryland. A narrow decision would only give a legal answer to the case in question but would not provide a rule of law to dictate future cases. The decision is expected by the end of 2018 and no matter how the Court will decide, it will definitely influence how data is collected by US investigatory agencies in the future.

United States v. Microsoft

While the decision in Carpenter will mainly affect how information collection from third parties will be conducted in the future, United States v. Microsoft is all about data collections abroad. In 2013, Microsoft was served with a warrant to disclose data on its users. Included in this warrant was a request for its Ireland affiliate to also disclose data on its servers. Microsoft got the warrant quashed and appealed against the Government's actions.

Beside Carpenter, this case is the second major question on data handling that should have been decided by the US Supreme Court in this term. The main question the Court should have decided is the legality of a US warrant served to a subject located abroad. This case was of special interest to international companies that have their headquarters in the United States but have many affiliates with big data centres all over the world, e.g. Microsoft owns 100 data

centres in 40 different countries. About twenty amici briefs from all over the world have been entered in support of Microsoft and proved the international scope of this case.

In general, a country may not act through its law enforcement agencies outside its own territory. This would constitute a major violation of international law because according to the general principle of sovereignty, every nation is free to act within its own borders but may not conduct any actions on foreign territory. The only exception thereof is an acceptable reaction to an act of war. Now, a US warrant issued by a US investigatory agency is considered a legal action by a government that is acceptable if served within its own territory. If the United States would have wanted to access information from a subject located outside its borders, such as Microsoft Ireland that is located on the green island, the official way of mutual legal assistance in investigatory matters should have been chosen. Based on a Mutual Legal Assistance Treaty (MLAT), a country can officially request a state action from another country within that other country's territory. Because the United States did not choose to follow this official path, the 2d Circuit Court in New York considered the actions taken against Microsoft as illegal and therefore Microsoft was not required to disclose any data that was located outside the United States. However, the Government did not take this decision well and therefore appealed to the US Supreme Court. [162]

Yet, very recently, Congress took on this issue of overseas access to information. While the decision of the US Supreme Court could go either way – oral arguments and questions by the Justices have not given a clear hint in regard to whether the Justices wanted to support the 2d Circuit in its reasoning or not – a law issued by Congress can be made to fit the issue. They have done just that with the so called CLOUD Act (Clarifying Lawful Overseas Use of Data Act). Based thereon, US investigatory agencies shall be able to access user data of US companies anywhere in the world. At the same time, the act opens the possibility of access by foreign countries to user data located in the United States after entering into a bilateral agreement. While this new act is the most favourable solution for the United States government, it could question the applicability of the EU-US privacy shield in regard to cross-border data transfers. Also, the CLOUD Act is a general bypass to the mutual legal assistance process.

Nevertheless, on April 17, 2018, the US Supreme Court declared the case moot due to Microsoft having accepted that a new warrant under the CLOUD Act should be issued and that based thereon Microsoft will disclose the requested data. Therefore, the main legal question of the case became irrelevant and the Court decided based on it's long established practice in such cases not to take a

final decision but give it back to the 2d Circuit Court for remand to also declare the appeal moot. [163]

When the CLOUD Act was signed into law, several technology companies including Microsoft were in support thereof. This may sound contradicting considering the fact that Microsoft had appealed against the Government because they did not want to disclose information.

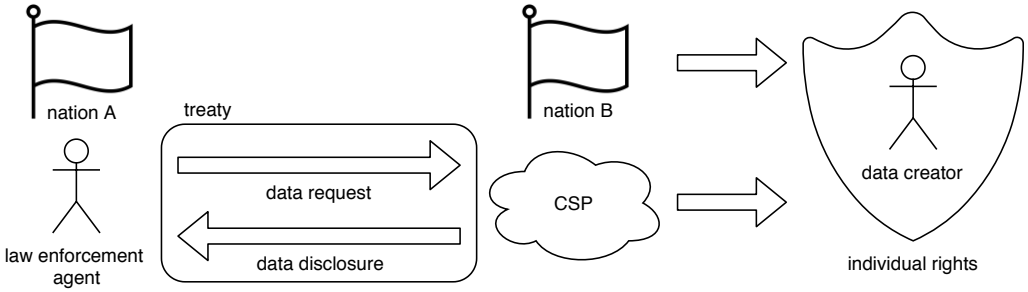

Figure 4.2: *Law enforcement agent requests access to data located in another country from a cloud service provider (CSP). Personal data is protected by human rights and the rule of law required by mutual legal assistance treaties. The CLOUD Act shall facilitate the extraterritorial access to data located in a foreign country.*

However, regarding the fact that under a law there will be clear cut rules on the "how" and "what" to be disclosed, it is the safest way to go rather than hoping that the Supreme Court might decide in favour of one's question. Especially in regard to the current more conservative composition of the Court, chances might have been that the Court would have supported the Government nonetheless. [164]

4.2.4 United Kingdom

In regard to the future of data protection Regulation, the United Kingdom may not be left unmentioned. Whether their current position on data and its handling is influenced by Brexit or whether this too can be considered an opposing opinion shall be left open.

However, it is noteworthy that from all the different official bodies that entered amici briefs in the case United States v. Microsoft, the one entered by the Government of the United Kingdom of Great Britain and Northern Ireland, was written in support of neither Microsoft nor the United States. Yet, in several arguments the British government followed arguments by the United States government and gave at least some support to national security reasoning that

should allow access to data independent of its location in the world. With its reasoning, the British government was the only non-US writer for this case that did not strongly oppose the argument of the United States. [165]

Also, the United States and the United Kingdom already started negotiations for a bilateral agreement allowing the United Kingdom the access to user data located in the United States under the CLOUD Act. As stated by a Downing Street spokesperson, the CLOUD Act as well as any bilateral agreement between the US and the UK will be in favour of any investigations of major crimes such as terrorism, human trafficking and sexual abuse of children. [166]

There is perhaps one big winner in the aftermath of Brexit in regard to the legislative and business development of data. While the UK tends to support a US position on data privacy and access, Ireland is a strong supporter of the European Union's GDPR. Irish businesses are working hard on the implementation of the GDPR. This is done in the hope of gaining business from all those companies that will be leaving the United Kingdom due to Brexit, but which still want to be part of the trading community and have close ties to Continental Europe. Already for years, Ireland has been the most successful place for data centres in Europe (please see also United States v. Microsoft Corp.). Latest news reports suggest that Ireland will keep this position for years to come. [167]

4.3 Policy and Political Issues

Not only the legislators have to consider questions on how to handle data. Also, companies have to decide on the quality and quantity of data they want to produce, use and publish while still being in compliance with national and international legal standards and creating a gain out of the data. In comparison thereto, law enforcement agencies around the globe have started to use the immense amount of data available in their data bases to cope with increased criminality, new forms of crimes on one side and the lack of personnel on the other side.

4.3.1 Open Data Access

While some organisations make data available to everybody, some do not. That is why first, it needs to be considered what reasons stand behind such publications or non-disclosures of data sets. One reason might be the wish to protect one's own information against the use through a competitor. Also, not all companies see an obvious benefit in publishing their data. However, especially monopolistic public companies tend to keep their data secret although they

would not need to fear a competitor. Nonetheless, publicly owned companies might consider full transparency unfortunate for their services. Because, if a citizen were able to analyse such company data, predictions and evaluations of the effectiveness of a service could be made.

On the contrary, academics often consider transparency as a positive factor of IT in the public sector [168, 169]. Thanks to the increasing use of technical means, the public sector is offered a chance to create more transparent working processes. Bovens and Zouridis (2002) define transparency as an "important principle of the constitutional state in the information society". Mainly, they underline the change of procedures from street-level via screen-level to system-level bureaucracies. According to Bovens and Zouridis (2002), clear-cut rules, observable decision-making and access to information are major requirements for citizens to verify the accountability and efficiency of public sector actions. Considering this, transparency is the key factor for citizens to assess the performance of public sector work and based thereon the proper use of tax money. For a proper observation of public work, the publication of data is therefore of utmost importance. Without this measure, citizens cannot verify and measure services. Applying the above-said to a public transportation system would offer citizens the possibility to assess the operational capability of the system. Especially, a user could identify whether a delay is based on an isolated incident or whether the transportation service provider does not put its best efforts in fulfilling its duties. A Web observatory might be able to get to this data.

Without doubt open source access can sometimes be very beneficial for the private sector. Companies can identify many drivers for open source adaptations in organisations; it has commonly been assumed that innovation is the key driver. Concluding, open source data may advance the technological development and progress society. [170]

Eventually, publishing data of a public sector organisation depends only on the internal or external political will. No matter how simple the technology to publish data might be, without the political will to do so, there will be no transparency in public sector work. However, Bertot et al. (2010) indicate that ICT "can in fact create an atmosphere of openness that identifies and stems corrupt behaviour". Nevertheless, it seems very unlikely that transparency develops spontaneously. Besides constitutional legality, transparency has become a new constitutional ideal which basically changes how citizens perceive the public sector [168]. [79]

4.3.2 Predictive Policing

Already as early as 1994, the New York Police Commissioner at that time had realised how valuable information and the processing thereof can be for the investigation and detection of crimes. William Bratton, a high-profile police man who had served in Boston had gone on to reform not only the NYPD (1994-1996 and 2014-2016), but had also directed the LAPD into a less violent and less corrupt future (2002-2009). One of Bratton's main achievements was the introduction of the use of data and statistics in everyday policing. [171]

CompStat (short for "Compare Statistics") is a program used by the NYPD since the early 90s. Already in the first year after its introduction, thanks to the use of current and historic data on crimes in New York City, the murder rate had dropped to such a low percentage it was absolutely unknown to the city. CompStat started off in the form of weekly meetings between different sections of the NYPD and transit police. Thanks to the analysis of their current numbers of incidences, crimes and arrests, predictions of future incidents could be made. Specifically, resources could be better directed and additionally to a very strict deterrence of even a misdemeanour, a general improvement in law obedience could be achieved. It was the beginning of predictive policing, where numbers were key to create an early prevention system against criminal activities.

While CompStat is often mistaken to be a software, it is mainly the procedure of maintaining a regular meeting schedule to compare and analyse numbers and crime statistics. However, the use of software was one key factor in the evaluation of crime statistics. After the successful implementation of CompStat procedures in New York, other cities in the United States followed suit, e.g. Los Angeles who implemented similar procedures while Bratton was serving as Police Chief of the LAPD. [172, 173]

Currently, Chicago is implementing the newest form of predictive policing. Based on an algorithm, scores are awarded to Chicago's citizens indicating whether they are likely to be involved in a crime or not. Thanks to the excessive amount of data available in Chicago's own data bases as well as other national crime data bases, the algorithm determines the chances of being either victim or perpetrator in gun violence. While presumably a person's former criminal history is one major factor to be considered by the algorithm, so are gang memberships, gang relations of family members, place of residence and employment status. The so called "Heat List" was widely criticised. Numbers yet need to prove that the Heat List supports the prevention and reduction of crime. What is already clear to many Chicago residents: Either the algorithm or the data fed to the algorithm is racially biased. Black citizens have been awarded disproportionally

more often a high score on the Heat List than their fellow white citizens. This may be connected to the fact that black people are more often to be stopped by the police than white citizens independent of being a criminal or not. Which would then support the argument that the data is generally biased. However, the Chicago police refuses to publish the algorithm to prove critics wrong. [174, 175]

While the effective use of big data in policing may still be at the beginning, the future will make police more and more rely on predictive data analysis. The pressure to cut costs and the reductions of police squads will leave law enforcement no other option than predicting their work based on big data, if they still want to cope with the increased threats to modern day society. [176] Similar projects are currently evaluated in other countries such as Germany and the United Kingdom. [177, 178, 179]

Another development of big data analytics is a social credit system. This sort of mass surveillance has been recently brought to attention by the Chinese government. It still remains a mystery what the scores and impact of such a system run by the Chinese government will be on its citizens. However, it may be a first step towards restricting personal freedom of citizens with low scores, e.g. access to public goods or Internet usage. [180]

4.4 Conclusion and Outlook

Even legal experts at the moment are still uncertain about the future of data protection and access to data. One main issue which might lead to difficulties in international relations in the near future are the two very different approaches taken by the United States and the European Union. Whereas the United States is currently propagating an extraterritorial reach for data collection, the European Union is very much in favour of an extraterritorial reach for data protection. While both legislative acts are fairly new, it is considered that these differences will complicate trading and relationships between these two big jurisdictions. Considering this, other countries such as Switzerland might also be influenced through these conflicting regulations, especially because compliance with both the European Union and the United States might be difficult to implement. [147]

Additionally, Gless (2018) outlines that there is yet no clear definition on the scope of data crimes or cybercrimes. Although the legislators are working hard to include as much undesirable conduct in their criminal provisions, constant technological changes make it difficult for the non-experts to define

what should be criminalised and how. Also, the strong influence and feasibility of data collection and analysis of private players might possibly lead to closer working relationships between law enforcement and/or administrative agencies and private companies to collect relevant information for prevention, investigation and detention of whatever will be considered a data crime, cybercrime or information crime by then. [181]

At this point in time, a prediction of the impact of the legislative projects undertaken is difficult to make, no matter how well written the algorithm might be. Only practical use and rulings of national supreme courts can provide answers to the proper applicability of these acts. However, we can be sure of one thing: By the time, a supreme court has decided a controversial question on the national or international relation to data and its use, technology will not have stopped evolving and any judicial "solution" to a problem might be outdated.

There might be clashes with the law specifically in regard to privacy laws when using a Web observation. If a person is observed through a Web observation or data is collected that can be put together into a personality profile, for example meta data, this may raise privacy concerns. However, such infringement can be circumvented if a person is informed on the purpose and the scope of a Web observation and that person gives their informed consent.

4.4.1 Recommendations

The current legal situation is difficult from the point of view of a computer scientist. First, it is challenging to give a clear forecast on the outcome of legal issues. In law, it always "depends" on the specific circumstances, there are very few universally applicable formulas. Second, the current legislative changes in the US and the EU fully redraft the data protection landscape. Therefore, it is even more challenging to give a prognosis in any way.

In general, anyone who plans to process personal data is always better off asking for consent in regard to his or her collective actions. Yet, it is the courts that will have the last word on the applicability of the new data laws to a case. If data is transferred across national borders things can become even more complex. A violation of a data protection act may also cause a violation of human rights or even cause a friction of international relations.

However, scientists have more freedom in comparison to private persons and legal entities when conducting scientific research. Many national constitutions as well as international conventions like the UN Charter and European Convention on Human Rights include provisions on the freedom of science. But until

now there are no court decisions in regard to the acceptable scope of data collections through a computer scientist under the freedom of science. Therefore, also a scientist is better off asking for consent when collecting personal data for research purposes. Concluding the above the author is unable to offer a universal formula to his colleagues due to the variability of the law. It always depends on the specific details of a case in which the facts must be weighed against each other.

This chapter described the the legal issues and challenges in Switzerland, the European Union, the United States, and the United Kingdom. It also tries to answer the first research question. The next chapter "Considerations for Web Observations" will discuss the process of examining Web resources.

Part IV

Creating a Web Observation

Chapter 5

Considerations for Web Observations

> Being able to talk to people over long distances, to transmit images, flying, accessing vast amounts of data like an oracle. These are all things that would have been *considered magic a few hundred years ago*.
>
> — Elon Musk, CEO of SpaceX, Tesla, Inc., Neuralink, and The Boring Company (born June 28, 1971)

This chapter discusses the awareness of Web observations and clarifies the states that are caught with the help of Web mining algorithms. Furthermore, it points out the thorough process of examining a Web resource before the actual Web observation begins.

5.1 Observation of States

In the ancient times astronomers observed celestial bodies and their movements. For getting the understanding of what happens in the heavens, they have extensively collected data of their observations. More and more observatories equipped with at the time state of the art telescopes were created trying to understand the system of the stars. With the data sets of Tycho Brahe, Kepler was able to formulate in the 18^{th} century the Kepler's laws of planetary motion. Kepler could interpret the data sets and could elaborate basic laws of physics. [182]

In a classical observatory, the research subject is a physical phenomenon. In contrast to Web observatories, the research subject is Web data (cyber) and a real life phenomenon. Web observatories are a collection of Web observations that collect data from the Web. In contrast to astronomers of the past, Web observations can automatically collect data from any given Web resource. Please see also section 2.3.1 "Mining Information from the Web". The idea of a Web observation is to collect data from Web resources that represent a state at a given time within the process. It is not a physical state but it preserves values at a certain time of measurement. Nevertheless, it remains a challenge to find a balance of effort spent and information gained. Some data sources comprise information value which only unravels after a certain investment. Decisions on the respective software architecture used for collection are crucial and hard to change in a later stage of the process. [183]

The analysis of Web resources over time may involve methods that focus on information value only. This means an application has to funnel information in which new information is identified and stored. Therefore, information must be checked according to its novelty value. The novelty value of information is the worthiness through new and previously unknown information. A time interval Δ_t in which information is collected may be an indicator but is not a final assessment of the novelty value. Whereas s_n is a state, a copy of a Web resource. As shown in the figure below information is repeatedly collected:

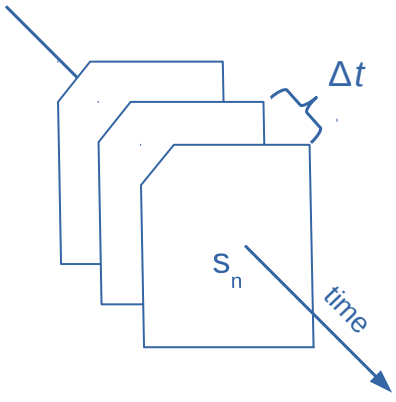

Figure 5.1: *Observing the Web needs an architecture that is capable of collecting and storing data over time. It must recognise whether s_n is new information. Δ_t determines the time interval of the observation depending on the content.*

Periodic collection of data alone is not very efficient. Information gains are created when new information is stored. However, saving all states would be a waste of data storage and Web traffic, therefore, only the information that differs is actually stored, e.g. s_1, s_2, ..., s_n.

Web observations should seek out for new information. Freshly collected information can be matched against already collected information (data sets s_n) in terms of attributes and content. In case attributes or content are the same as found in the already collected data set, the information can be considered as old information. Therefore, the information should not be stored in the data collection again. Similar information must be dropped in order to avoid duplicates and redundant data sets (incremental update strategy). In contrast to information that cannot be found in the data collection, the application treats such information as new information and stores it into the data collection. For this purpose new information must be processed as follows:

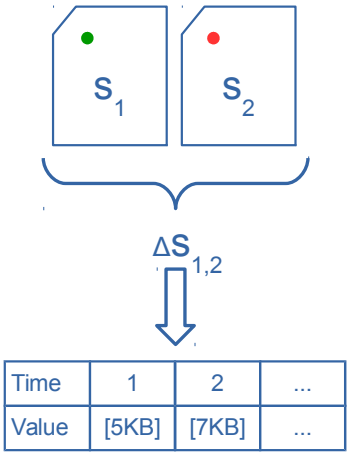

Figure 5.2: *Web observations need an architecture that is capable to collect and store data over time and thus must recognise whether s_n is new information.*

Information s_1 (green) is compared with s_2 (red) and attributes and content are matched accordingly. With an incremental update strategy, only dissimilar and therefore new information is then saved and stored in the data set, in this illustration within a tree structure. This ensures that new information, s_2 (red), is saved and stored within the data set. With this method all different states are collected and stored (Δ_s). Furthermore, the data set provides information in machine-readable means what ensures the automatic processing. With this kind of differentiation scheme it is rather easy to classify information according its novelty value.

5.2 Awareness of Data Collections

Web observations make it possible to collect states of information from the Web. For example, when a business sells goods on the Web, its specifications and prices are automatically retrievable. Ordinary Web users do not have the means of assessing structures behind Web resources, they usually see what the Web resource owner wants them to see. For non-technical people collecting states of Web resources is a manual task. However, skilled computer scientists have the abilities for automatic collection of Web data. An initial effort is needed for getting to interesting information of a Web resource. The effort starts with the first inspection of Web traffic and ends with an automatic collection method. Such a consideration can take minutes to several hours, more complex Web observations even days, which in the end determine the outcome of the whole observation application.

Usually website owners upload data on Web resources in order to show Web users content. Every Web resource owner is aware of this copy of data because it was deliberately uploaded to a Web server. Unlike the awareness of possible information gathering from Web observations, the consequences are not perceived by the data owner. Weber (2017) has introduced an awareness model which outlines the levels of awareness of Web resource owners. An adaptation of the model can be represented as follows:

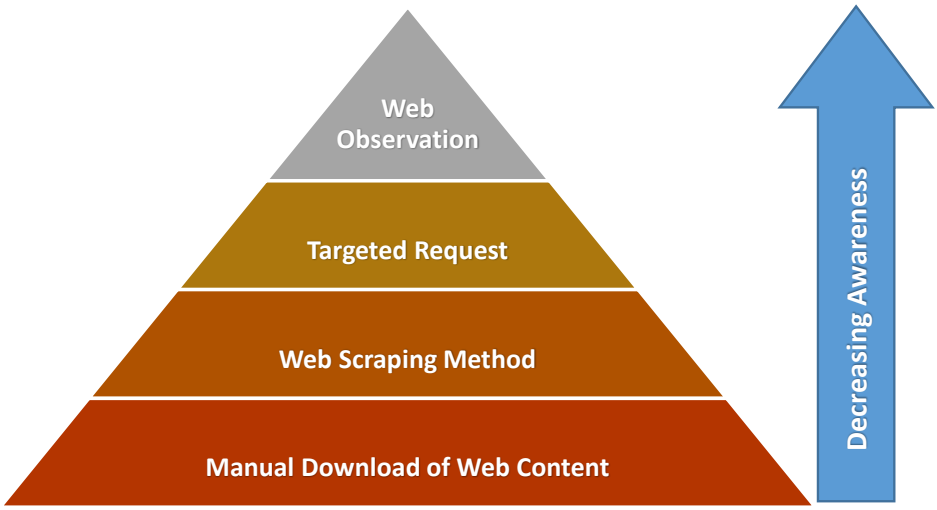

Figure 5.3: *The pyramid symbolises the decreasing awareness of website owners. The overall awareness of data collections decrease from bottom with each step towards the top. While website owners understand the concept of manual download of Web content, automated Web observations and its information gain is perhaps not known. [184]*

The first bottom layer of the pyramid "Manual Download of Web Content" represents the understanding that content is accessible. And there are many ways for getting to the Web content of a Web resource, e.g. right click and "save as". The next layer "Web Scraping Method" focuses on data collection of desired content of a Web page. It should only retrieve specific data, e.g. file or parts of the Web resources' content. More sophisticated measures are "targeted requests" whereas the Web resource is sent manipulated requests e.g. HTTP in order to get data from it. This can significantly increase the information value from a Web resource. At the top, most likely Web owners are completely unaware whether someone started to accumulate information about the Web resource with a "Web Observation". Automated intervals and content states including meta data help to determine how the Web resource evolved. Web resources that are under scrutiny of a Web observation can reveal more about their content than it is meant to be. The pyramid describes the idea that the vast majority of Web resource owners are unaware of Web observations and the ability to systematically collect Web data.

5.3 Examination of Web Resources

There is a need for a certain process that clarifies the goal of a Web observation. A Web observation only makes sense when a definition is in place in regard to what information should be collected. Functionalities such as sorting function, data cleansing, and visualisations help to analyse the collected Web data.

Such clarifications include a comprehensive analysis of the Web resource in order to find a method for Web data mining. Conclusively, it entails the specification of *conceptual and technical design*, *input/output*, and final *data structure*.

5.3.1 Conceptual and Technical Design

Web resources provide content in various ways. A concept of clarifications describes where the important data is located and which parts are used for the Web observation. The technical design outlines how desired data can be retrieved. For example, targeted requests can be used for the purpose of data retrieval which depends on the technologies used by the Web server. The conceptual and technical design should also emphasise the feasibility. The analysis comprises of finding a method to automatically mine desired content over a longer period of time. The following list is part of the conceptual and technical design of a Web observation.

- **Exact data location**
  Common places for desired Web data can be within the Web resource, e.g. an HTML file. It can also be stored in other files and places. Minimum parameters are the URI and the section of content of the Web resource. It may also be that content is spread over multiple Web resources.

- **Targeted requests**
  Web data is often only revealed when a request has been sent. Imitation of website behaviour often involves HTTP requests but also database requests for the purpose of Web data mining e.g. an API or HTML forms.

- **Time and costs**
  Most of the data on the Web can be collected with a Web observation. Some websites are forearmed against automated data access which detect bots and block them before data can be gathered. From a technical perspective, all content that is being displayed in a Web browser can

also be mined. It can be said that the higher the difficulty to mine Web data the higher the costs. Available libraries can help to tackle these kind of problems but also are time consuming. Other cost factors are server environment, computing times, and Web traffic. Care should be given to balance time requirements and costs in order to avoid a growing mismatch.

5.3.2 Definition of Input/Output

After desired content is identified and located the definition of input and output of a Web observation must be specified. The specifications from the conceptual and technical design in regard to the contents should be considered as input. It might be useful to repeatedly examine a resource, in case there is hidden information within the source code. The definition of input data makes it clear for what data the observation is looking for. A brief look at the Web resource indicates the data sought for data collections. The output is particularly important because it will be used for fulfilling the goal of the Web observation. Usually, the output corresponds to the input which is further processed for the preferred data structure.

- **Syntax analysis/parsing**
  Analysis of a syntax that consists of consecutive strings and symbols is also called parsing. Parsing makes use of defined rules and formal grammar to dismantle content into its components. The outcome often consists of parsing a tree that may outline syntax relations or semantics. Often libraries are used for parsing tasks that make a programmer's life much easier.

- **Data processing**
  Input and output data must be processed for the actual need of the observation. Create, read, update, and delete (CRUD) operations are executed in order to process data according to its needs [185]. Also the final display of data must be considered and certain visualisations must use a specific file format.

- **Long-term performance**
  The hardware and software environment has to endure a long runtime. It is very much depended on the infrastructure, e.g. server software, free hard disk space, bandwidth. Long-term Web observations should also consider proper programming standards in order to avoid unwanted effects such

as memory leaks. A memory leak is a particularly unwelcomed bug in an application because it consumes more and more memory while it does not release memory which finally results in a crash of the application.

5.3.3 Selection of Data Structure

As described in the section before, output data corresponds with input data. For the purpose of Web observation it makes sense to adopt the data structure which indicates the observation architecture. Ideally, the data structure is suitable for the Web observation and does not need a lot of support effort. The processing of large data has to be considered when a Web observation will run over longer periods of time.

- **File format and data structure**
  There are many file formats and corresponding data structures that can be used for Web observations. Data should be well arranged for storing valuable information e.g. file format or database. To prevent data loss and increase accuracy, adequate data formats should be used more frequently. Time formats like "2018-05-01 08:00 AM" have an underlying convention, e.g. ISO 8601 for date and time displays. For the transformation of data into written language, habits, time zones and standards must be taken into account. Please see also section 1.1.3 "Data Sets" for sustainable file formats.

There are multiple aspects to consider before starting a Web observation. The most important points are the before mentioned, however, this list is non exhaustive. Depending on the observation goals, it can be further escalated to create even larger Web data collections. However, this comprehensive analysis illustrates essential requirements for most Web observations.

This chapter discussed the awareness of data collections and clarifies the states of data collections. Furthermore, it pointed out the process of examining a Web resource. The next chapter "Architectural Designs" will outline two architectural approaches.

Chapter 6

Architectural Designs

> If you're doing *anything interesting* in the world, you're going to have critics. If you absolutely *can't tolerate* critics, then don't do *anything new or interesting*.
>
> — Jeffrey Preston Bezos, Chairman, President & CEO, Amazon.com, Inc. (born January 12, 1964)

This chapter outlines two distinct architectures for Web observations that have been developed in order to answer the second research question. Starting with reusable Web observation scripts, the "Event Condition Action (ECA) System" is predominantly used for reactive Web applications and orchestration of events on the Web. It may also be combined with all sorts of Web servers as well as Web observations. While the ECA System was specifically built for event-driven interaction, the "Building Block Architecture" emerged for the sole purpose of Web observations. In turn the building block architecture can be composed with different software packages and programming languages. Each block can be equipped to the user's needs in order to create all sorts of Web observations. Both systems will be further discussed in this chapter.

6.1 Event Condition Action (ECA) System

The Event Condition Action (ECA) System is an application framework that provides a standard structure of software, designed to support software programmers in core functionalities such as data and session management. The use of framework applications drastically reduces the amount of time for the creation of Web applications, however, strongly varies from its purpose of use. [186]

Bosch (2014) originally outlines the idea of an application framework with reactive features that is considered an ECA System. His work "Towards Reactive Information Systems and their Services" led to the creation of an event-based application framework programmed in JavaScript [187]. This notion has been further improved with the focus on real-time reaction, which uses inbuilt coding functionality for the reactivity in between Web applications and resources. The goal is to know as fast as possible when an event has occurred. However, the architectural design must be flexible enough to handle real-time data.

The development of such a prototype fulfils the needs for reactive actions on the Web. Its focus is not primarily on Web observations but for all sorts of API interactions. Naturally, Web communication is latency-driven, a certain delay before the actual transfer of data begins followed by transfer instructions. Thus, asynchronous communication and scalability mechanisms are core features of a Web application framework. Another key aspect is the architectural model that enables the programmability of events. It enables inbuilt coding in which JavaScript can be used. The underlying programming code is shown within the user interface and can be directly modified.

6.1.1 Architecture

The basic architecture of the ECA System is based on either an event or data flow. Both data (blue) and events (orange) can be retrieved from a "Web Service". The system keeps feeding itself through the "Event Trigger" and "Event Listener", in which "Webhooks" can be used. According to the external input sources events can also be aggregated. For further actions, stored "ECA Rules" are being applied in order to process the two input streams in the "Rule Engine". Afterwards, actions are dispatched in the "Action Dispatcher" whereas the action itself may also be used as input for another "Web Service". After all, the ECA System is a Web application framework that accepts all sorts of requests in order to provide all possible interactions. It foremost wants to achieve reactivity on the Web.

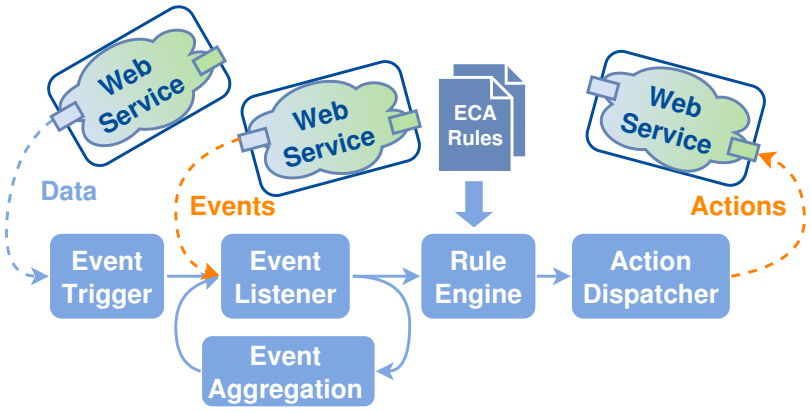

Figure 6.1: *ECA System interconnects Web services and Web resources. Web services often deliver data through an API in which the "Event Trigger" anticipates the data. It may also directly interact through events in the "Event Listener". The "Rule Engine" works according to the stored "ECA Rules" and instructs the "Action Dispatcher". [91]*

In detail, the ECA System consists of five main modules. The Web is used as a resource for events and actions thereon which enables access to data and events on the Web. For the purpose of processing information, three core entities control the flow of events and actions. In the top layer the "Poller" (active events) detects and polls Web data for desired information while the "Event Listener" is in place for passive retrieval of events from Web services. The "Rule Engine" dispatches predefined actions according to a rule set which can be adjusted to the user's needs. For Web observations the ruleset is in place to gather new information and drop old information. Rules are elaborated by an administrator for certain types of events, for example a Web service. In the middle layer, databases ensure the storage and execution of user input. The lower part of the system architecture consists of the system management where CRUD functions can be assessed.

The full ECA System architecture outlines as follows:

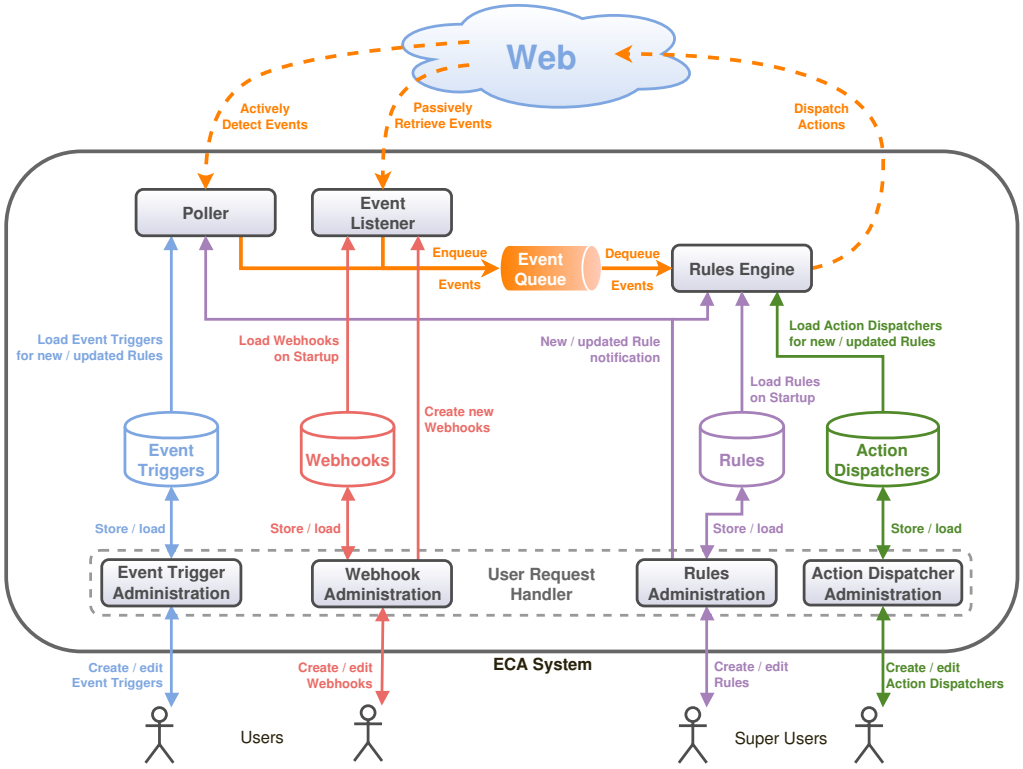

Figure 6.2: *Core functionalities are shown in rectangles and data storages in cylinders. "Poller" and "Event Listener" forward events to the "Event Queue" whereas the "Rules Engine" evaluates events for actions. Configuration settings are stored within 4 databases that consist of the "User Request Handler". This kind of user input is managed via the user interface. [91]*

If an event or a data flow of interest occurs, the information is forwarded to the "Event Queue". Whenever an event rule matches the input, the "Poller" and the "Event Listener" enqueue their information into the "Event Queue" in which events are stored for the "Rules Engine" to be processed. Upon the ECA rules actions are dispatched.

The main functions within the ECA System architecture can be explained as follows:

- **Poller**
  The "Poller" loads stored "Event Triggers" for new and updated rule entries and enqueues them to the "Event Queue" in case the detection rule matches; thereof an event will be dispatched. "Event Triggers" may actively detect or poll changes in the Web which are transformed into events;

- **Event Listener**
  "Event Listener" listens for data input on APIs for the "passive retrieval of Events" or loads "Webhooks" on start-up including the related URI. Events are forwarded to the "Event Queue" if the detection rule is matching;

- **Event Queue**
  The "Event Queue" acts as a queue for events from "Poller" or "Event Listener";

- **Rules Engine**
  Events are processed in the "Rules Engine" from the "Event Queue" when a match with a detection rule is found;

- **User Request Handler**
  Interaction with users and administrators are undertaken in the "User Request Handler" which builds 4 different interfaces for CRUD actions: Event Triggers, Webhooks, Rules and Action Dispatchers. [91]

All these functionalities are based on action and event objects. Action objects target URIs with the help of detection rule configurations while event objects consist of a fixed time and a source attribute. Whenever a condition is met in both cases, an action will be processed and an event is created upon the given parameters.

6.1.2 Conclusion

The ECA System is mainly built for detecting and dispatching events on the Web. With so many functionalities such as the event listener, event poller, and sophisticated rule engine for the orchestration of Web events, it is not an ideal system for conducting Web observations. However, the architecture of the ECA System seems to be very helpful in reacting to all kinds of events on the Web. Therefore, the system ought to be very valuable in orchestrating tedious tasks of Web users. The ECA System wants to encourage Web users to program their own applications with their preferred language to provide a more reactive Web.

With this hand-coded architecture, the goal was to make use of Web events with the following options:

- **Inbuilt programming window**
  A mask for programmers that allows direct input but also "hot swap" or "hot deploy" of programming code. With this concept code is taken into the application without the need to reload the whole page;

- **Usage of event rules**
  The use of already available event rules for removing tedious tasks. For that reason, regular Web users can make use of the ECA System.

Existing Web applications and services in this field are limited to the offered functionalities while the created ECA System is extendible without limitations.

Another issue encountered was the maintenance of Web observations in the ECA System. The creators of the ECA System did not focus on the use of the collected data from Web observations, because new information must be processed immediately and more emphasis was taken on event detection on the Web. For that reason the value of novelties was valued higher than data structures. Under best circumstances, real-time updates are the preferred communication means as value of novelties degrade by time. Web change detection was the main practical use case in which the system proofed its usefulness. The reactivity is generated for example every time Web users are notified by mail or when a tweet is posted on Twitter. Output emerges from Web changes which is to some extent a reactive behaviour. This may help Web users orchestrating Web services and underlying events for their purposes.

Difficulties arise, however, when an attempt is made to store data by using the ECA System's architecture. The system does not support data file formats, only strings are passed along. In addition, most of the system users were actually not deploying code from the ECA System but from their own command line. Conclusively, the system did fulfil its purpose but did not serve full Web observation capabilities and was therefore phased-out end of 2016. [91]

6.2 Building Block Architecture

An architecture consisting of building blocks is the logical consequence for having all necessary functionalities combined in order to make powerful Web observations. Weber (2017) introduced the concept of an architecture that consists of customisable and reusable blocks. His work "Component Based Web-Scraping Strategies" was the starting point of this architecture [184]. By adapting and enhancing this notion and incorporate it into Web observations, this work elaborates a valuable Web observation architecture that is flexible thanks to the numerous choices for each building block.

By splitting the actual Web observation tasks into small building blocks it makes this architecture exceptionally flexible. The Web observation process remains the same as well as the framework, requests and individual parsing. However, the programming code can be reused and a problem-specific solution can be found. This enables the collection of Web data from any Web resource, no matter what Web technology has been used. The idea is to run all processes in a chain of blocks that can be customised for the Web observation task just like a construction kit.

6.2.1 Building Blocks

Building blocks consist of a well-defined system boundary which solves one particular problem of the Web observation task. A building block has to be created once whereas the Web observation application uses the best fitting blocks for the observation. One building block is a basic solution component that will be used for many different problems.

What follows is a full outline of all used building blocks that could be chosen and exchanged for the purpose of the Web observation:

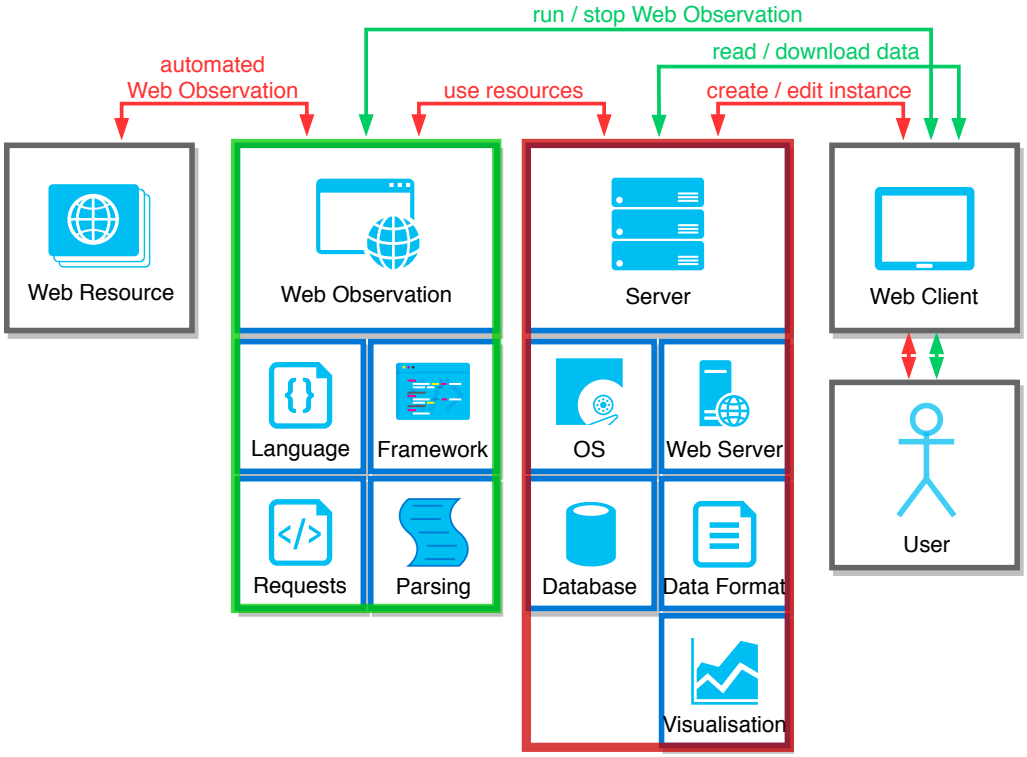

Figure 6.3: *Single building blocks form the basic components of the building block architecture. Each block outlines the interchangeability of each design decision. One block contains many different design choices that must be adjustable for Web observations.*

The "Web Resource" and the "Web Client" as such are not part of the overall architecture (grey). Both represent the target and the end user that wants to create a Web observation. By means of using each building block with its best fitting feature for a given Web observation task a choice can be made for each of the components (green and red).

For example after careful examination, a Web observation can be defined according to the needs of the Web resource. In case, seek time is critical when querying data, an in-memory database would be the preferred choice. For that reason the "Database" determines which "OS" should be used. From the "Database", data sets can be extracted in a particular "Data Format". The format often depends on the "Database". For visualising data of a Web observation a "Web Server" displays the data that is also depending on which libraries are used in the "Visualisation". It often comes down to the decision

of which libraries are used for the visualisation, e.g. D3.js. The flexibility of choice takes all available software, frameworks, libraries, etc. and not just one particular in consideration.

The same is applicable to the "Web Observation" block. The programming "Language" that best fits to the Web observation task is taken which comes with a "Framework", e.g. Node.js, including packages. With the help of the programming "Language" and the "Framework", automated targeted "Requests" can be created to the desired "Web Resource". The resulting data is then further processed for the "Server" building block. With the "Parsing" the desired data of a Web observation is stored into a format that meets the volume of data which is usually stored in a "Database".

- **Web Client**
  With the help of a "Web Client" the best approach for gathering the data of a "Web Resource" can be identified. It is the first point of contact with the "Web Resource" in which a user is able to obtain the structure of a website. Depending on this examination, the conceptual and technical design (server resources), input/output definition and data structure selection (Web observation creation) can been undertaken.

- **Server**
  The server environment contains distinct features such as operating system "OS", a software for the "Web Server", a data storage solution which consists most often of a "Database", which also correlates with a pre-defined "Data Format", and thereof resulting "Visualisation" which is highly depended on the building blocks in the "Web Observation". A choice of each building block must be well thought through according to the requirements. It must be taken into consideration that large amounts of complex data sets might be collected.

- **Web Observation**
  The Web observation should be the best solution for collecting data of a specific "Web Resource". The Web resource has a big impact on how the Web observation is constructed. The setup of building blocks has to be adjusted to the needs of the Web resource. The application code is determined by the chosen programming "Language". Some languages work within or make use of a "Framework". Different HTTP "Requests" are sent to the "Web Resource" in order to get to the encapsulated data. Finally, a parser for proper "Parsing" is used to store data for the "Server" building block.

- **Web Resource**
  The "Web Resource" is the actual target for the data collection process of the Web observation. Whether it is HTML, XML or another file format, the "Web Resource" must be thoroughly analysed for the best solution of the "Web Observation". After an analysis, a "Web Resource" may have an open API or a another data source to make use of.

The outlined modules in each block represents the flexibility and large extent of options which in turn depend on the Web observation task in order to collect data of a Web resource.

Container Architecture

The resulting final container architecture facilitates the solution that delivers Web observations for all sorts of Web resources. For each Web resource the blocks are chosen by the developer that fit best for a Web observation task. While all the containers run on one single host server it enables the communication in between each choice. Moreover, the linking of containers facilitates the sharing of containers for different Web observation solutions. Linked components are isolated but can also be made available for other containers for further processing. Web observations can use different containers for their tasks. A composition of containers in combination with a Web mining programming code makes up a Web observation. The host server consists of an Ubuntu 16.04 that has the container software "Docker Community Edition 17.12.0" pre-installed and running.

The modularity of this system allows greatest flexibility for the implementation of Web observations. As follows the template of pseudo code that uses modular blocks for a Web observation task:

```
1:  //pseudo code
2:
3:  //server block
4:
5:  //defines container with operating system, database, webserver,
        and visualisation library
6:  use container_A -OS -database -webserver -library_visualisation
7:
8:  //web observation block
9:
10: //library import
11: import library_A
12: import library_B
13:
14: //settings
15: webresource = "https://www.web-resource.com/data_API"
16: databaseAccess = {
17:   username: "username"
18:   password: "password"
19: }
20:
21: //web observation interval
22: timeInterval = every 2 hours
23:
24: //data collection
25: function collectData() {
26:   //request to webresource
27:   response = doRequest(webresource)
28:
29:   //parsing process
30:   parsedData = doParsing(response)
31:
32:   //database access and data storage
33:   connectDatabase(databaseAccess)
34:   storeData(parsedData)
35: }
36:
37: //interval
38: startWebObservation(timeInterval, collectData)
```

Listing 6.1: *Pseudo code template of a Web observation of the building block architecture. The server block defines the operating system, database, additional Web server software and libraries for visualisations. The Web observation block consists of libraries for the actual Web observation application including settings for data storage, observation interval, and the actual target Web resource.*

As follows the figure that illustrates the final architecture:

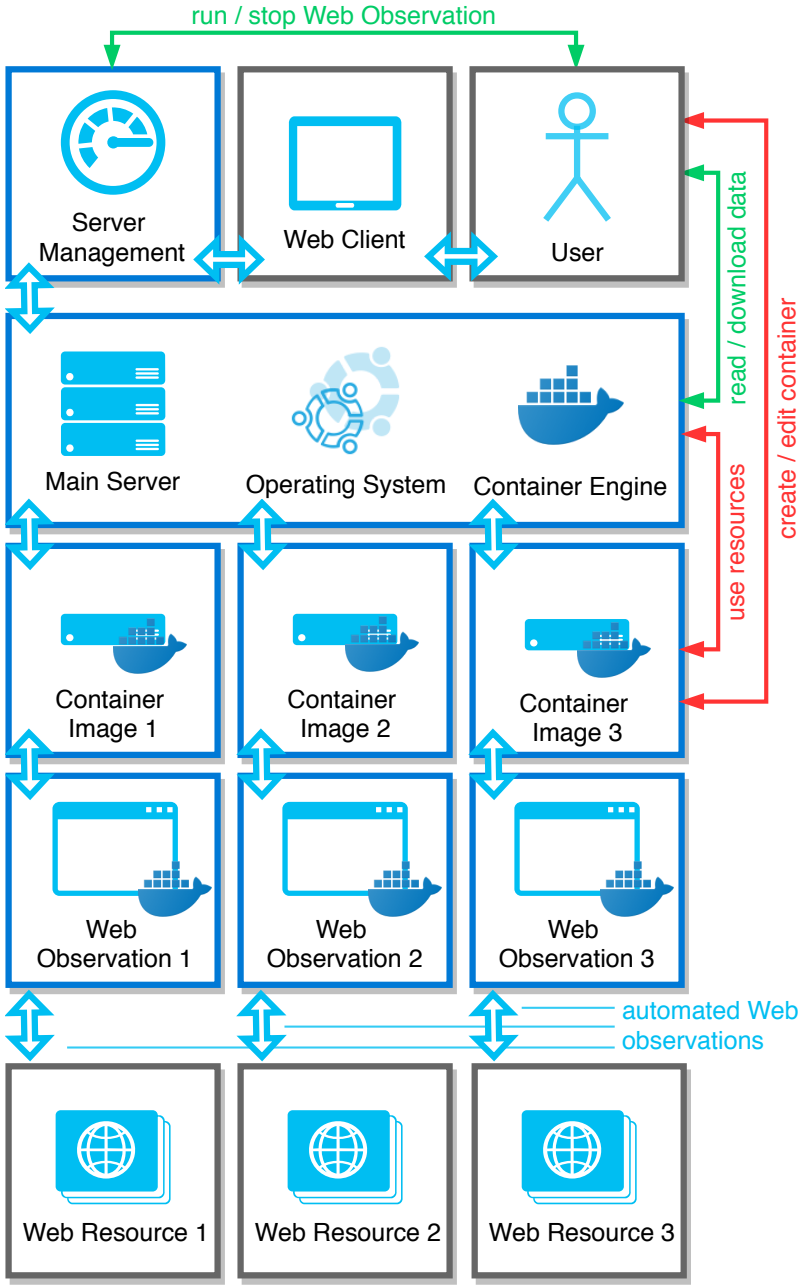

Figure 6.4: *Final container architecture using a container software.*

A "Server Management" building block is set in place to streamline administration tasks such as the orchestration of containers and Web observations. Essential for this building block architecture is the enabling factor of a container software. It paves the way for an environment that provides greatest flexibility and links all building blocks together.

6.2.2 Web Resource and Web Client Blocks

The main activity to find a suitable Web data extraction algorithm includes analysing the "Web Resource" with the help of a "Web Client". The outcome of this setting has implications for all other blocks of the Web observation. However, the data source has to be revealed in order to set up a proper Web mining process. Any difficulties must be set aside within these blocks, e.g. data cannot be automatically retrieved, otherwise no proper solution can be achieved. The challenge is to find the right and relevant data for the Web observation within the "Web Resource" which can be a time-consuming task. Unless the costs are too high for getting to the data, a prototype solution shall be the outcome. Conclusively, the "Web Resource" is the actual data source, the "Web Client" or Web browser is used for the purpose of reverse engineering. Both form two external building blocks that are static parts of the architecture whereas the Web observation needs only one accessible Web resource.

Web Development Tools

"Web Clients" provide the functionality of web development tools such as code debugging. A so called Web developer tool does not foster the creation of websites, it actually gives the developer the necessary means for testing. Development tools are usually built-in in modern Web browsers. Web development tools pave the way for versatile Web applications that browsers have to make use of. The most basic feature is the code inspector that allows an instant look into the website code. Another important feature for detecting data flows of a "Web Resource" is the network tab that highlights all sorts of network traffic. Moreover, it outlines the full communication in between the "Web Resource" and the "Web Client", for example XMLHttpRequest (XHR), JavaScript (JS), and Cascading Style Sheets (CSS). [188]

6.2.3 Web Observation Block

The Web observation consists of many blocks that make out the actual data observatory. In essence, the Web data collection task is an automated solution for the data collection. For each Web resource on the Web, a specified solution has to be set up in order to fulfil the Web observation task. There is no silver bullet solution that would achieve in every situation the perfect data collection. That is the reason why a building block solution is perhaps the best solution to facilitate a Web observation. The choice of programming "Language" is determined by the structure of the "Web Resource" which also inflicts the

"Framework" used. In the end, it is the personal preference of the programmer that determines the selection. While "Requests" stay the same, the code for creating these requests and "Parsing" meaningful information to the "Server" may differ from solution to solution.

Language, Framework, Requests and Parsing Blocks

The choice of the programming "Language" depends on the "Web Resource", however the basic components may consist of equal parts. It is clear that the most basic code snippets can be reused. Thus, each programming language has its own coding syntax and "Frameworks" in place. All these programming languages must be able to emit requests to "Web Resources" which then are parsed in a machine-readable "Data Format". Upon these decisions storage of data "Database" and the resulting data sets "Data Format" have to be dealt with. Time intervals play also a key role for the maintenance of a Web observation.

6.2.4 Server Block

The server block contains all server specific software that is used for the Web observation. The server block must allow multiple databases, Web servers but also operating systems.

Operating System (OS) Block

The OS block provides different operation systems with different functionalities for the "Web Observation", always depending on what "Database" and which "Web Server" is running for the representation of the data in the "Visualisation". Another option would be the use of cloud services that have similar capabilities. Yet, operation costs on the cloud services can hinder the operation. A common option as the base OS are Linux operating systems.

Database and Data Format Blocks

The variety of database software is diverse. Generally, there are three types of databases which can be classified as follows: First, by the type of content, second, by the application area, and third, by the technical aspects such as their structure. Thus, there are both advantages and disadvantages by using one particular database software. Nevertheless, it must be able to store data from the Web observation task. Predominantly, the "Web Resource" determines the "Database" software choice which is derived from the mined "Data

Format". Perhaps complex requests in a Web observation result in a more complex database. NoSQL databases perform better on large data sets but relational databases have more functionalities to unravel data with complex queries (see also chapter 1.1.3 "Data Sets").

Visualisation Block

Data sets do not speak for themselves; it is necessary to visualise data in a certain way to express its meaning. Especially, data that has valuable new information must be filtered out in order to be visually recognised. Dashboards in which data is seen in real-time are common practise in the business. The "Data Format" for visualisations have to match the programming language, e.g. a JSON file format is expected.

Visualisations may also need a "Web Server" for live data that is directly shown on a URL. This has also implications on what "Web Server" software must be installed in order to achieve real-time applications. Another advantage is the direct access to the visualisations for any Web user equipped with a browser.

6.2.5 Container System for Linking Building Blocks

The final challenge is the linking of each building block with its unique solution for a particular problem. A generic architecture must be found for the interconnection of blocks and for making quick changes of components in each block. This notion of having the choice between different software, programming languages, and frameworks, etc. and the ability to take the best solution for the Web observation task. A solution must be found that allows to have multiple software choices with different configurations.

A possible solution for coupling all building blocks into a flexible set up may be provided by a Container System e.g. Docker. Docker is using virtualisation of operating systems and creates containers thereof. Containers are in an isolated state but share parts of the operating system, bins and libraries. Docker itself advertises its containers as "freedom of choice" for "agile operations". It is the flexibility to make use of different software and even legacy systems that makes Docker particularly interesting. In contrast to classical virtual machines, Docker does not need a fresh installation of an operating system. It rather isolates necessary configurations from the operating system for applications. So most parts of a container run separate from its host. [118]

For the building block architecture it is useful to run multiple containers with pre-defined images. For a properly working container image the following points must be respected:

1. **Definition of building block**
   The definition of the building block helps to identify what should be achieved by the container. In case the Web observation receives a JSON file format, it makes sense to use a database with a similar document scheme, e.g. MongoDB.

2. **Official images**
   For common databases there are usually official images available. The hyperlink "https://hub.docker.com//mongo/" provides a list of all available MongoDB images. Perhaps more importantly, there is no need to always update the Web observation because of software updates; legacy software can be easily used.

3. **Adjustment of settings file**
   The settings file "Dockerfile" can be adjusted to the needs of each Web observation.

6.2.6 Conclusion

The architecture provides a general solution for Web observations. Considerable emphasis was given to the flexibility for all sorts of Web observation tasks. A building block architecture provides the extensibility needed for such a tool. While the ECA System was focussing on detecting and dispatching events, the building block architecture is a viable solution for Web observations.

By dividing the problems into building blocks, solutions can be reached bit by bit. It even makes it possible to add a new building block with distinct tasks. While design decisions still are important, the outlined architecture possibly will be more adaptive than other rock-solid solutions. As discussed in section 5.3 "Examination of Web Resources", it all comes down to finding the exact location of data within a Web resource. HTTP is the enabler to get to the data whereas the cost factors, time and level of difficulty determine the overall effort for a Web observation.

The variety of Web technologies make the choices even harder. Therefore, container-based and linked building blocks provide the flexibility of Web programming. Cost-benefit of the infrastructure is another economic factor;

open source software contributes to a cost-efficient architecture. Finally, the introduced architecture has to stand the test in further Web observations.

This chapter described two distinct architectural approaches; the ECA System and the building block architecture. It also answers the second research question. The next chapter "Measurements" will indicate limits of the building block architecture in terms of thresholds and ideal interval frequencies.

Part V

Results from Web Observations

Chapter 7

Measurements

> [...] when you *can measure* what you are speaking about, and express it in numbers, *you know something about it*; but when you *cannot measure* it, when you cannot express it in numbers, *your knowledge is of a meagre and unsatisfactory kind*;"
>
> — Lord, 1st Baron William Thomson Kelvin, Scots-Irish mathematical physicist and engineer (born June 26, 1824 – died December 17, 1907)

This chapter evaluates the building block architecture in terms of thresholds and interval frequencies of Web observations. These benchmark measurements will indicate the limit values of the building block system. Furthermore, the evaluation will indicate the maximum interval frequency of the Web and its interval frequency boundaries with the help of measurements from both the Intranet and Internet.

7.1 Purpose of Web Observations

By using the outlined architectures data sets can be obtained from the Web. Depending on the architectural design and the need of the Web observation, machine readable data is generated for further analysis. From data sets, it is possible to derive conclusions into the real world that have significant importance to people. Web observations by themselves do not automatically produce

new knowledge, it must first be extracted from the collected data. The final goal of every Web observation is to discover new information from the respective collected Web data. Furthermore, it is adequate to publish these data sets as long as no infringement to third-party rights occurs, e.g. privacy intrusion or personality violations (see also chapter 4 "Legal and Political Perspectives"). Based on the data set and its extracted knowledge, interested parties can draw their own conclusions.

Specifically, Web observations can be used for the independent assessment of a Web service. One key problem that remains in any service is the inability of customers to verify the quality of a service. However, such verification can be provided by a Web observation. By searching for predefined HTTP responses, e.g. "200 OK", exceptions within a service can be caught and also stored in the data set. In addition, a response time-out can easily be configured in order to make sure the website is reachable. Request tools provide these functions such as "response_timeout". As a result, the performance of a service can be measured by the availability and transmission delay determining service uptime.

Since many Web resources change over time, it can sometimes be important to catch a snapshot of the current state of the Web resource. With these saved states of a Web resource it is possible to outline the development of a Web resource and which part of the information is actual new information. Some information changes in a way that Web users do not understand and do not have the means to investigate the Web resource's behaviour. For example, news portals want to hook you to their website and therefore specifically publish news articles sequentially. Getting more people to read the follow up articles would possibly maximise the advertisement revenue generated from the additional page visits. Another example of collected information changes through Web observations would be the development of an encyclopaedia entry that is of particular interest. Thanks to the Web observation, changes in the content can be gathered and thus create a message with the changes. Consequently, by observing and making snapshots of the Web and its content, analysis of these data sets may hold valuable insights about a Web resource and its progression.

7.1.1 Latency-Driven Web

For Web observations, it does not matter how often they have to iterate in order to gather Web data. The frequency is specifically adapted to the respective Web resource to get maximum output with minimal input. The actual time interval must be estimated before a frequency can be set. The minimum frequency of a Web observation is usually determined by the Internet connection, its latency

from the Web client to the actual Web server. It logically crosses multiple router hops and is also protected against distributed denial-of-service (DDoS) attacks. This protection and other aspects of the Internet such as the current flow of data may hinder a Web observation through low intervals from successful performance. While in the Intranet, it is possible with very low networking hops to have Web observations with very low intervals.

7.2 Experiments

Given the latency of the Web, the interval frequency will reach the limits of the Internet. Nevertheless, reaction times of the building block architecture remain unclear. Different interval frequencies should cause varying reaction times of Web observations. For that reason, two experiments have been created to obtain the necessary measurements. The first experiment is set in the Intranet with one switch between the Web resource and the Web observation that uses the building block architecture. The second experiment takes place on the Internet between a Web resource and the Web observation as well as before with the building block architecture.

7.2.1 Setup

For the experiments, a Web observation and a HTTP server (both in Node.js) have been created and made available on the Intranet and the Internet. The scenario has been chosen to have control over both sides; the server and the Web observation. Issues on the server side such as blocked requests were excluded, so the experiment could focus on the speed of transport routes. The Web observation uses the building block architecture to perform a fixed amount of 550 HTTP requests with various interval frequencies. The fixed number of 550 requests per Web observation has been selected for practical reasons because all interval frequencies (500 to 0.05 milliseconds) can be handled in under 10 minutes. Initial tests have revealed that this fixed number did not significantly alter the results. By using the building block Web observation architecture, request and response times are added up and measured together.

The two experiments are illustrated as follows:

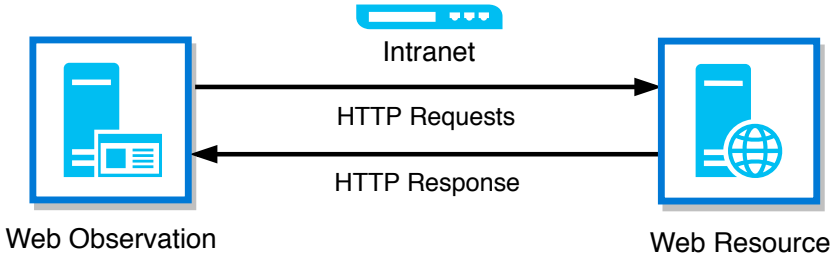

Figure 7.1: *Experiment 1, Web observation over the Intranet.*

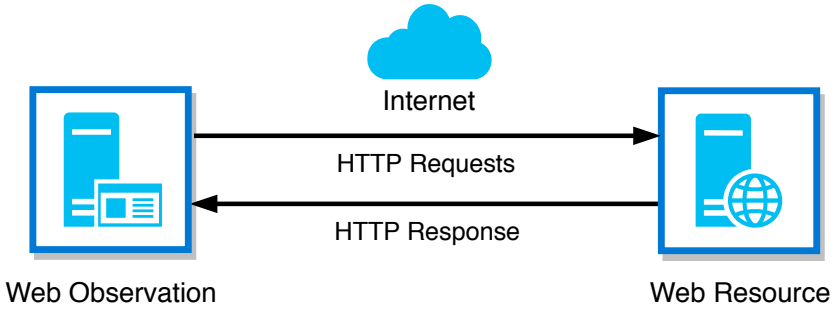

Figure 7.2: *Experiment 2, Web observation over the Internet.*

For both experiments the building block architecture is used for the evaluation. In a best case scenario over the Intranet with almost no traffic and a real-life scenario over the Internet with traffic and several routes the speed can be benchmarked. The output from these measurements will be a limit value or threshold in which a system boundary can be derived.

The building blocks are structured as follows (see also section 6.2 "Building Block Architecture"):

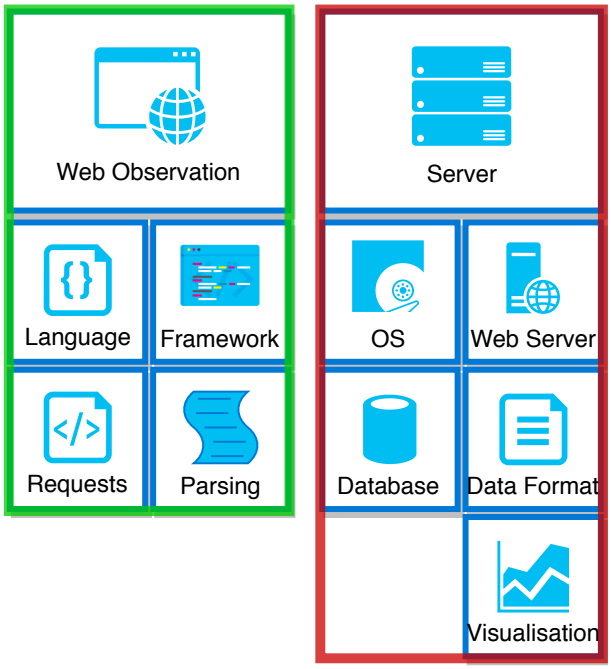

Figure 7.3: *Detailed view of building block architecture which represents the actual Web observation (green) and the Web resources on the server (red).*

For the evaluation, various Web observation interval frequencies indicate at what interval frequency the building block architecture will be still able handle requests and responses. These measurements will outline the performance of the system on the Intranet, but also on the Internet with many possible network routes and hops. This would indicate the performance boundaries of the system.

All kinds of interval frequencies from 500 to 0.05 milliseconds will be tested and measured. The resulting time corresponds to the duration from start to end of the Web observation. In both experiments the whole process is measured from the initial gathering of information from the Web resource (HTTP request) to the storage of that respective information on the local machine (HTTP response).

As follows a code snippet of the JavaScript application implemented within the building block architecture in both experiments:

```
1:  //JavaScript code snippet
2:
3:  //counter variable
4:  var i = 0;
5:
6:  //store current system time in variable 't1'
7:  var currenttime = new Date();
8:  var t1 = currenttime;
9:
10: //HTTP request
11: needle.get(url, function(error, response) {
12:
13:   //if HTTP status code '200' OK
14:   if (!error && response.statusCode == 200) {
15:
16:     //load HTML body 'response.body' in cherio and
17:     //return it to variable 'htmlContent'
18:     var htmlContent = cheerio.load(response.body);
19:
20:     //Web observation data
21:     var WebObservationData = htmlContent('body').text().trim();
22:
23:     //if WebObservationData is not empty
24:     if (WebObservationData === "") {
25:
26:       //store current system time in variable 't2'
27:       var aftertime = new Date();
28:       var t2 = aftertime;
29:
30:     } else {
31:
32:       //store empty variable iteration number
33:       WebObservationData == 'empty: '+i;
34:     }
35:
36:     //output the time difference
37:     console.log(t2-t1);
38:
39:     //exit after 550 iterations
40:     if(i > 550) {
41:       exit;
42:     }
43:
44:     //increase counter variable
45:     i++;
46:   }
47: }
```

Listing 7.1: *Code snippet of the Web observation for the measurements.*

The application is measuring the time difference at the moment a request is issued (t_1) until a response is received (t_2). This results in the actual time costs until Web data can be gathered through the building block architecture.

7.2.2 Results

From the given measurements, no matter what time interval has been set out, excessive requests in the Intranet do not extend the time of requests and responses. Measured times keep up the pace with higher interval frequencies.

As outlined in the figure below interval frequencies 250, 90, and 1 milliseconds show the following result in the box plot (maximum, third quartile, median, first quartile, minimum):

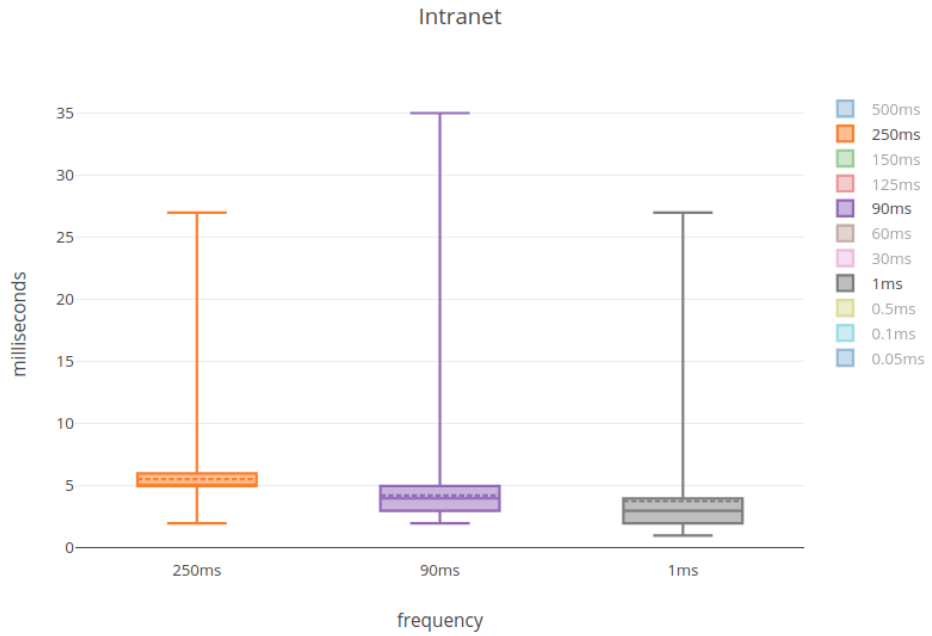

Figure 7.4: *Web observation interval frequencies 250, 90, and 1 milliseconds on the Intranet. The measurement outlines a time range between 2 to 35 milliseconds.*

By testing all kinds of interval frequencies from 500 to 0.05 milliseconds in the Intranet, it can be stated that no significant change in the times could be observed.

The Web observation task performed similarly, even if the Web observation interval is set to 0.05 milliseconds:

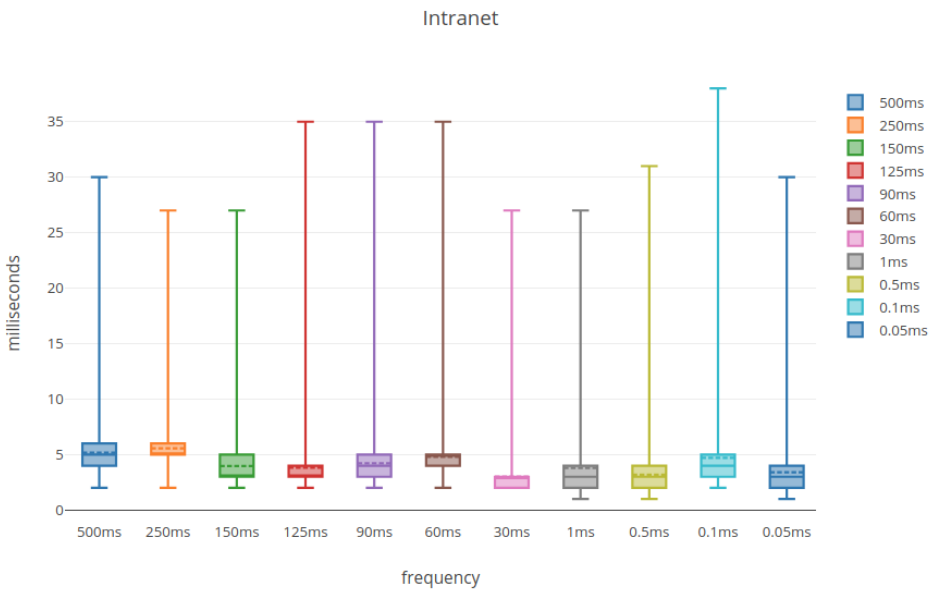

Figure 7.5: *Web observation interval frequencies from 500 to 0.05 milliseconds on the Intranet. All frequencies could finish their Web observation task and could successfully deploy and process 550 requests.*

In the Intranet, duration of a Web observation varies from 2 to 38 milliseconds. Interestingly, all requests were successfully processed. Responses could be triggered even with an interval frequency as low as 0.05 milliseconds. Yet, considering the fact that Web observations are generally thought to be applied to a Web resource on the World Wide Web, it is obvious that results on the Internet must be different.

Logically, a Web observation which is conducted with the building block architecture on the Internet performs with significantly higher request and response times than over the Intranet.

As outlined in the following figure intervals below 90 milliseconds significantly increase the total time consumption of the Web observation:

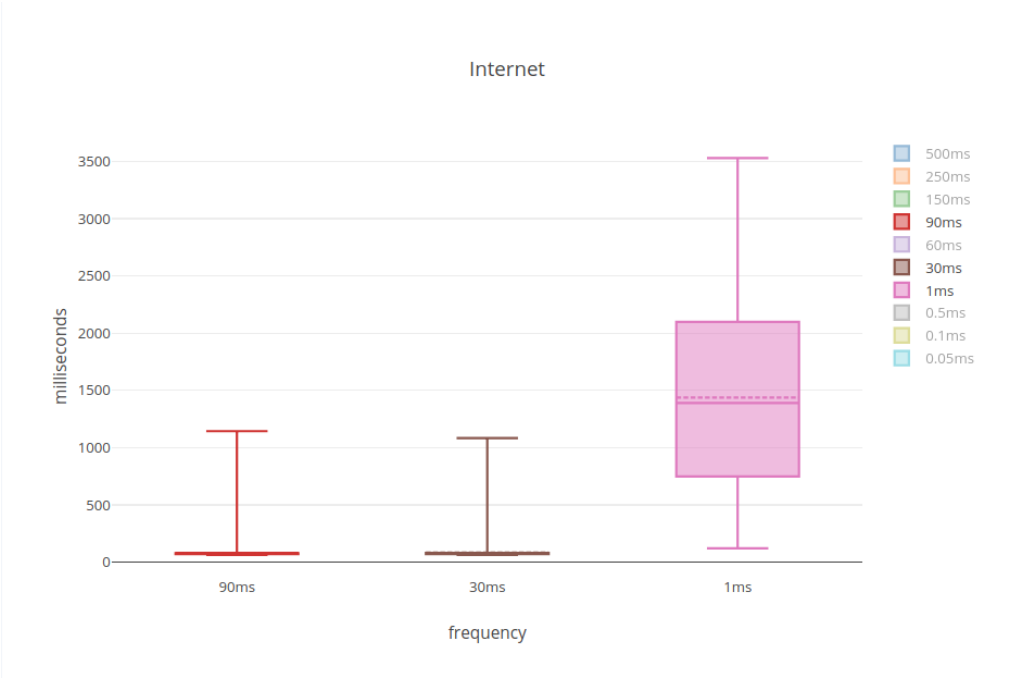

Figure 7.6: *Web observation interval frequencies from 90, 30, and 1 milliseconds on the Internet. A high jump between 30 milliseconds and 1 millisecond can be observed.*

One possible reason for this jump in times could be the latency of the Web which includes additional Web traffic. In contrast to the Intranet experiment, interval frequencies are not alike in the lower interval frequencies. From these measurements, it can be inferred that after a certain interval frequency, measured times increase or finish without proper results. The HTTP request has been sent but no HTTP response is received (time-out). It seems that high request frequencies are blocked on many network devices of the Internet for obvious reasons, e.g. DDoS. A sharp drop of performance in total time consumption can be observed from 30 to 1 milliseconds.

As shown in the figure below, the Web observation task runs well up to an interval frequency of 30 milliseconds:

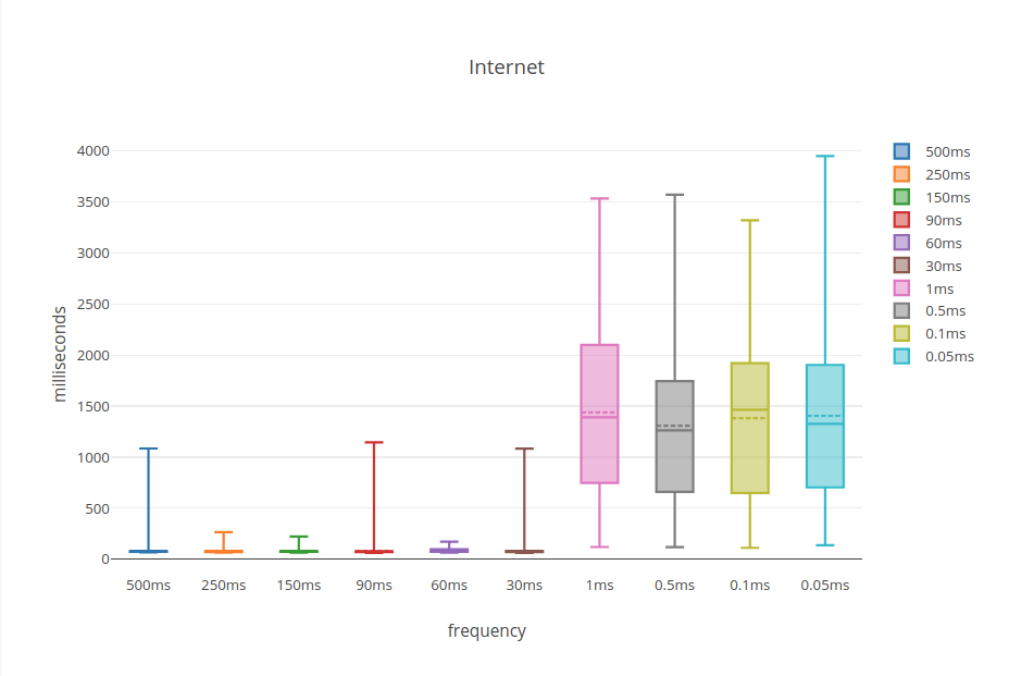

Figure 7.7: *Web observation interval frequencies from 500 to 0.05 milliseconds on the Internet. The frequencies of 1, 0.5, 0.1, and 0.05 milliseconds could not finish the task and timed out without notice before 550 requests had been deployed.*

Interval frequencies below 30 milliseconds will increase total time consumption to 3500 milliseconds. Even worse, most of the requests time out. Most importantly, the architecture of the Internet prevents interval frequencies below 30 milliseconds. Such tasks time out without any error message from the Web resource itself. One could expect a HTTP Status Code 429 "Too Many Requests" but this did not happen in the test environment. Most likely network devices block excessive requests which prevents a HTTP response. Therefore, no data could be collected. It seems that high interval frequencies are instantly blocked by the Web server or the network devices when using the building block architecture over the Internet.

7.2.3 Conclusion

From the results of these benchmarks, it can be deducted that the interval for Web observations must be higher than 30 milliseconds. By adding a safety margin of 20 milliseconds, a frequency of 50 milliseconds is the maximum for a

Web observation interval frequency to be successfully deployed with the building block architecture. In a perfect environment on the Intranet better results can be achieved. The delay in the data flow between the Web server and the Web observation, also called latency, and other factors of the Internet do not allow more requests.

It is clear that multi-player online games are not in the scope of Web observations. Players do feel distracted by network latency between client and server above 25 milliseconds. Therefore, the Web observation building block system is not suitable for supporting measures in online games.

This chapter outlined an evaluation of the building block architecture with the intention to indicate limits or maximum interval frequencies. The next chapter "Observation Scenarios" will illustrate five Web observation scenarios.

Chapter 8

Observation Scenarios

> It's not about *charisma and personality*, it's about *results and products* and those very bedrock things that are why people [...] are getting more excited about the company and what Apple *stands for* and what its *potential* is to contribute to the industry.
>
> — Steve Jobs, Co-founder, Chairman, and CEO of Apple Inc. (born February 24, 1955 – died October 5, 2011)

This chapter illustrates five different Web observation scenarios in order to evaluate the flexibility of either the ECA System or the building block architecture. The scenarios were specifically selected in which meaningful data could be collected, e.g. movement of geo location, time differences in schedules, and new information. The diversity of scenarios ensures the flexibility of the architectures under scrutiny. The scenarios outline gathered Web data that would be otherwise irrevocably lost. They are perfect examples of what kind of information a Web observation can collect. Many other observations have been undertaken, this selection consists of the most promising projects. First, data of a free floating car sharing service is presented. Second, a public transport operator's performance is shown. Third, news articles and comments thereon are in scope of the Web observation. Fourth, price changes of an airline over the time until the flight day are of interest. Fifth, the usage of an online messenger might give far more insights into a personal life than expected, all solely based on Web

data that would otherwise evaporate.

8.1 Catch a Car

Catch a Car is a free-floating car sharing service that was launched in Basel in the summer of 2014. Since November 2016, Catch a Car is also available in Geneva. The idea of the service is to provide cars within a specified zone of a city or region for car sharing purposes. The term "free-floating" defines cars that can be left at any given end point of journey (in a car park) if that position is within the pre-defined perimeter. [189]

Infobox

Architecture:	Building Block System
Data format:	XML
Filtering:	Yes
Frequency:	7.5 min
Login:	No
Uptime since:	May 19, 2016
Volume:	7,315.2 KB per day
# of requests:	192 per day

For the Web observation of this service data first steps included the identification of the Web resource (www.catch-a-car.ch) from which information should be collected and the kind of data that should be gathered. A thorough analysis of Web resource as described in chapter 5 "Considerations for Web Observations" will most likely lead to the data sources relevant to the Web observation. Fortunately, Frei (2016) developed a method to extract data from the Web resource of Catch a Car which has been adapted by the author to fit this Web observation into the building block architecture. [190]

8.1.1 Web Observation Task

The eye-catching characteristic of the Catch a Car service is the provision of a map with all available cars visibly attached. For the user-base this feature is beneficial because it signals all cars that are ready for rental. Thanks to this broad disclosure of location information it is technically possible to record all movements of cars and determine the usage rate of the service.

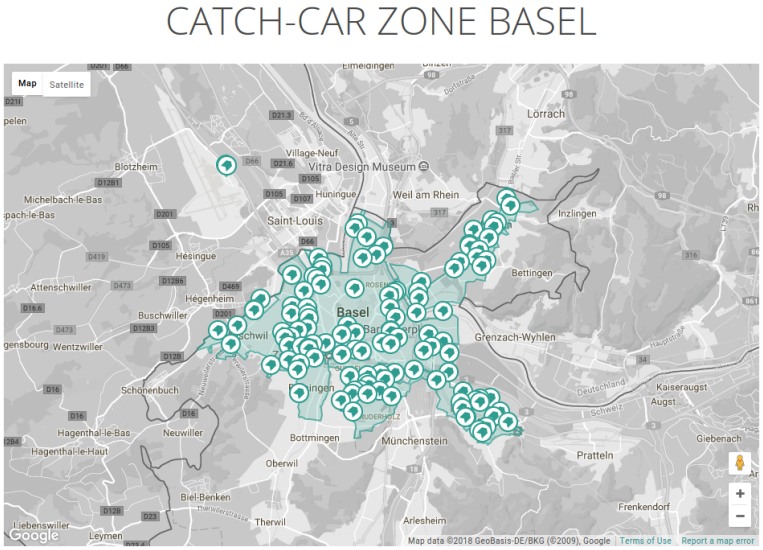

Figure 8.1: *Screenshot "www.catch-a-car.ch", the map outlines the perimeter in which cars can be parked on public ground. Car rentals must start and end within this zone. [191]*

This is an example in which data is shared for a user experience, however, the shared data may tell much more than desired. An even more detailed view can be found by clicking on a specific car on the map. This reveals not only the exact street name and number where the car has been parked, but also the serial number of the car and its remaining level of gasoline. These are useful data fields to get a better understanding of the service.

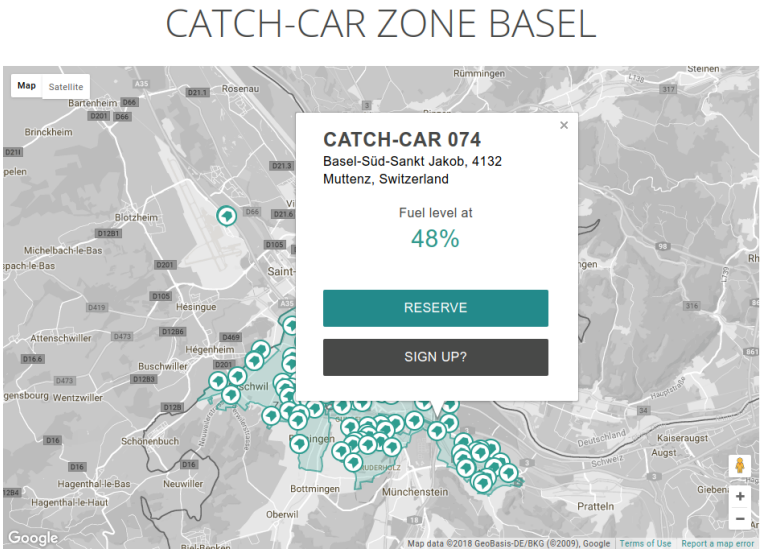

Figure 8.2: *Screenshot "www.catch-a-car.ch", detailed view of one car of the service. [191]*

From the picture before the fuel level can be determined which is indicated at "48%". From the specifications, it can be learned that the car model "VW up!" has a standard 35 litre fuel tank [192]. A simple calculation reveals that the car has 16.8 litres in its tanks and 18.2 litres have already been consumed by car rental customers.

However, more questions arise in regard to the service usage and its efficiency and profitability. A Web observation shall give answers to the following aspects:

- **Usage of car sharing;**
- **Number of car rentals;**
- **Time used by each car rental;**
- **Full and single distance driven by each car rental;**
- **Fuel consumption of single rental and all rentals taken together;**
- **Total car rental time;**
- **Distribution of rental cars within the perimeter;**
- **Hot and cold car rental spots.**

Data of Catch a Car is freely accessible on the Web with a Web client. Access to these data sets can be managed with technical understanding within the HTML code of Catch a Car. As long as data is shown in the Web client, data can be collected and stored for other purposes. This allows it to gather states of the service in a data set from which the actual usage of Catch a Car may be interpreted.

8.1.2 Verifying the Results

A Web observation may collect all data shown in the browser. By assuming that the collected data set is correct calculations can be made about the car sharing service. It is thus possible to provide real-time data in the respective latest data collection, e.g. number of current car rentals, available rental cars and total number of available rental cars.

Geographic Coordination System

In case the rental cars are not visible on the map, they are currently used for rentals or are most likely in maintenance. With the help of the geographic coordination system, geolocations of each car and their exact parking location can be made visible. Based on two precise values, a location can be determined. The coordinates 47°33'35.4" N, 7°35'58.8" E point to the city of Basel and are collected from one of the Catch a Car data sets. The author could verify the data point to an actual car parked in the city: "X: 47.559833, Y: 7.599665". To figure out what threshold value is relevant for the decimals within the coordinates, a visualisation might be helpful:

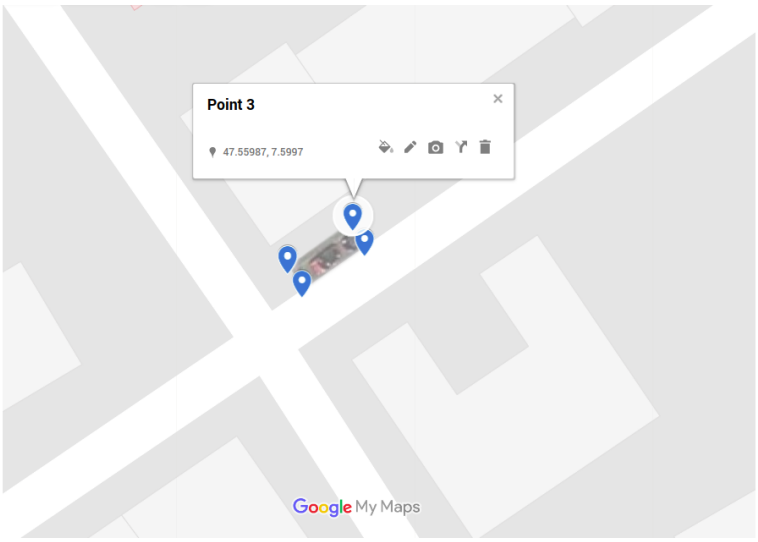

Figure 8.3: *Examination of geolocations. A detailed view of a car park overlayed with the satellite view. [193]*

By overlaying the satellite view on the coordinates in the screenshot above, a parked car is indicated in red. Based on the coordinates, it can be said that it is actually a car park and therefore seems to be a correct value of the Web observation. Since the cars are spread across the whole city of Basel, it matters where exactly cars are parked. On examining the coordinates supplied, the following points corner the car in the car park as follows:

1. **Point**
   47.5598, 7.59962;
2. **Point**
   47.55983, 7.5996;

3. **Point**
   47.55987, 7.5997;

4. **Point**
   47.55984, 7.59971.

The X and Y values of the coordinates outline that it might be reasonable to round the decimal digits. This measure clearly identifies a vehicle within a car park. While looking at the coordinates of the X values, they all have the same four decimal digits: "47.5598". Thus, it might be reasonable to round the X coordinates to four decimal digits. Unlike the Y values, the fourth decimal has already two different digits as choice. However, in any case by using the Y values "7.5996" or "7.5997" the car park would still be exactly identifiable. If absolute certainty is preferred, the Y coordinates can be rounded to the fifth decimal digit.

	ROUND(GeoX,4)	ROUND(GeoY,5)	KID	ID	Name	Time
0	46.1406	6.14601	3199945	329	Catch-Car 329	2017-01-24 13:00:06
1	46.1665	6.11421	2070893	371	Catch-Car 371	2016-11-27 11:15:06
2	46.1690	6.18794	1864897	360	Catch-Car 360	2016-11-16 17:15:06
3	46.1700	6.13161	2521373	309	Catch-Car 309	2016-12-20 19:30:06
4	46.1700	6.13164	3070820	339	Catch-Car 339	2017-01-17 20:00:07
5	46.1700	6.13165	3093112	339	Catch-Car 339	2017-01-18 23:00:06
6	46.1700	6.13185	3340913	320	Catch-Car 320	2017-01-31 23:00:06
7	46.1701	6.13172	3132879	333	Catch-Car 333	2017-01-20 23:45:06
8	46.1703	6.13198	2892550	334	Catch-Car 334	2017-01-08 19:00:06
9	46.1703	6.13201	2714157	338	Catch-Car 338	2016-12-30 18:30:06
10	46.1703	6.13203	4745561	372	Catch-Car 372	2017-04-13 17:30:06

Figure 8.4: *Rounded data set. X values rounded to 4 decimals and Y values to 5 decimals.*

From this examination, it can be deduced that all available cars of the car sharing service correspond to the collected Web observation data sets and a physical car park. The coordinates rounded to 4 decimals on the X coordinates and to 5 decimals on the Y coordinates give sufficient information for identifying the location: "X: 47.55983321, Y: 7.59966535".

Test Drives

The resulting data set of the Web observation, must be validated. In this particular case it is possible to rent a car and search the data set for the entry. This verifies whether the Web observation collects all states of the car sharing service. In self experiments carried out on August 10, 2017, and thereafter the

author undertook several test drives identifying both the rental car used and the specific ride afterwards in the data set:

Figure 8.5: *Self experiment of the Web Observation with the car number "050" on August 10, 2017.*

Figure 8.6: *Rental drive from A to B of car number "050" on August 10, 2017.*

The Web observation can be stated complete only if test drives are actually stored in the collected data sets. More than 20 test drives have been conducted and all of them could be matched to information found in the data sets collected by the Web observation. Therefore, the Web observation did catch all the states necessary of the free-floating car sharing service "catch-a-car.ch". Collections of observation states are made four times per hour: minute 0, 15, 30, and 45.

Hence, it is likely that some rental drives are not collected due to the observation interval frequency of 15 minutes. In this case two drives are collected as one. For example, in case a rental drive ends at the end of an hour on minute 58 and the car is taken over for a new rental drive on minute 59, these two rental times would still be collected as one drive. It would take another 15 minutes to recognise that the car has been moved but neglects the fact that there were two rental drives. However, since all performed test drives could be observed in the collected data, the likelihood of such an occurrence is fairly small. A first look at the data shows that the cars' rest period is much longer. Therefore, even if a rental drive would only take 10 minutes, it takes other customers at least 5 minutes to get to know that a new car is available in their area and additionally a considerable amount of time to walk to the car if needed. So 15 minutes seems to be a reasonable interval frequency for this kind of service.

In addition, the mileage driven did actually correspond by a discrepancy of approximately 0.3 miles (0.5 kilometres). The reason for this difference is that the shortest way from point A to B is given by the Google Maps API. Yet, the actual path chosen also includes a turn around the block for placing the rental car in a car park. This leads to the discrepancy in test drives observed by the author. The real-world values must therefore be carefully assessed and may only serve as an indicator.

8.1.3 Interpretation of Data

To make sense of the collected data by the building block architecture from the Web resource of Catch a Car, several methods to analyse the data may be be applied. After having randomly tested several rides, it can be assumed that the data set is correct. It should be noted that the data set is composed of the start (A) and end point (B) of a rental car. So, it basically describes the shortest path. From this data, it cannot be clearly identified which roads the driver took or who used the rental car at the time. However, the gasoline indicates the usage and allows assumptions on such statistical outliers and an overall calculation of the car rental business can be made. The number of kilometres driven should correspond to the gasoline usage. Users are free to fill up the car but they do not have to do it on their own.

From the data set, it is clear that we have to identify all cars for the Basel section of the service. The service is also used in Geneva based on the same architecture, thus the Web observation also collected data from the service in Geneva.

By mapping the geolocations of Basel, the service made use of a total number of 123 cars. Plotting one month of the car number "001" results in the following figure:

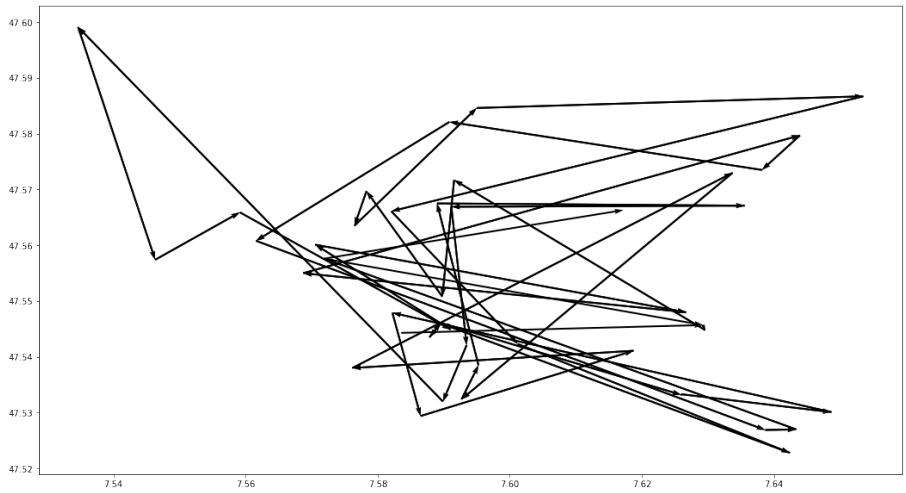

Figure 8.7: *All 44 rental drives in January 2017, car number "001". X-axis corresponds to longitude and y-axis to latitude.*

One limitation to the figure above is that it does not highlight the actual position of a car in relation to the map. By adding a Google Maps layer, it significantly increases readability. The system perimeter of Catch a Car (in turquoise) is very helpful for getting a first insight of the car rental movements:

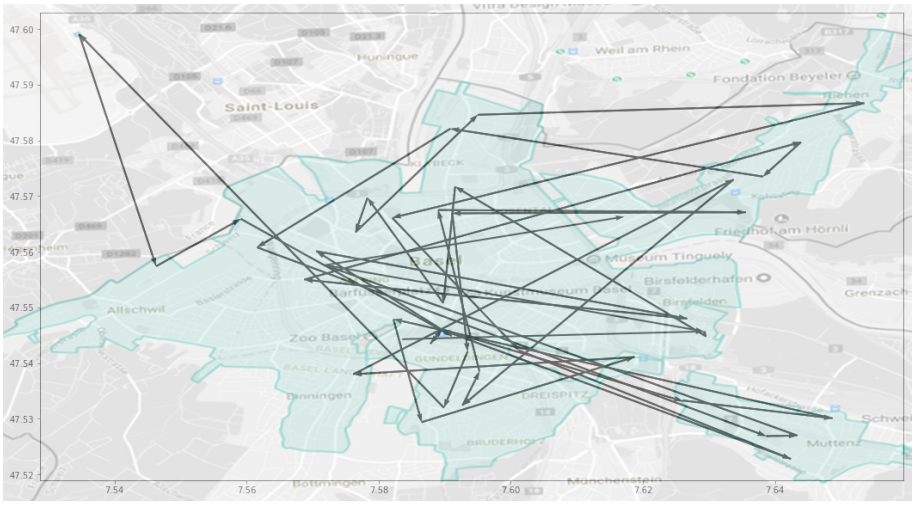

Figure 8.8: *All 44 rental drives in January 2017, car number "001", overlaid map layer. X-axis corresponds to longitude and y-axis to latitude.*

This sample of the rental car "001" shows the start and end point of the

car's journeys. A more sophisticated calculation of the data will be outlined throughout the whole year of 2017. This will provide further information on how the service is actually used over a longer period of time. Since the data set contains a start and endpoint, the number of entries has to be divided by the factor 2:

```
#Python

#number of Total Car Sharing Rides
RentalCount = 0
for i in CarID:
    if len(df_car[i]) == 0:
        print('Number of Car Rentals ' + i)
        print(round(len(df_car[i])/2))
    else:
        RentalCount += round(len(df_car[i])/2)
```

Listing 8.1: *Calculating total car rentals.*

In 2017, facts and figures of the Web Observation can be calculated from all rental drives. The total rental time is the time during which a car was rented. Given the observation interval of 15 minutes, calculations indicate an estimated 40,451 hours of total rental time for all cars in Basel. The value is an estimate based on the observation interval.

```
Total Number of Car Rentals: 55,473
Average rentals per car:        451
Total km driven:             79,937 km
Average km per car:             649.89 km
Average km per ride:              1.44 km
Total fuel consumption:       4,796.2 l
Total gasoline costs:         8,393.40 CHF
Total rental time:        2,427,063 minutes
Total rental revenue min:   857,225 CHF
Total rental revenue max:   995,095 CHF
```

Listing 8.2: *Main output of the analysis.*

While the total amount of kilometres driven also reduces the fuel tank which is also stored in the data set, it additionally correlates with the data field "EstimatedDistance" that has been created by the car sharing service itself. This cross-check makes the author more certain than the calculation is accurate. Based on all car rentals, it can be said that cars are predominantly used for short distances. Perhaps Catch a Car uses their information to assign two car rentals in a row by the same person with the same car as only one car rental drive. This could possibly explain to some extent the mismatch of the officially stated (7 km) and from the Web observation collected values (1.5 km) per rental drive [194]. However, this explanation does not fully close the gap of 5.5 km. Therefore, it would be beneficial if the service Catch a Car would publish

their data to explain this difference. Overall, the statement of Catch a Car raises serious doubts with regards to the average distance of each drive. Moreover, these values allow to calculate an estimated revenue. Based on these values and given assumptions, a maximal revenue of CHF 1 Million can be achieved.

The 55,473 movements build a strong result of the Web observation. From the movements, an average rental time per car can be calculated. This results in an average usage of approximately 30 minutes per day.

```
Average rental time per car:
  2,427,063min/123 = 19,732min; 328.87h
  328.87h/649.89km = 0.50h; 0.50h; 30.36min
```

Listing 8.3: *Average rental time per car.*

The analysis does not exclude maintenance and fuelling by Catch a Car – this will be recorded as rentals. It might be possible to identify refuelling with the help of the fuel level by excluding drives that have 100% of fuel in the tank after the ride.

Additionally, the data also shows at what time the service is actually being used. The measured values clearly highlight service peaks between 6 and 9 pm and indicate the nature of use:

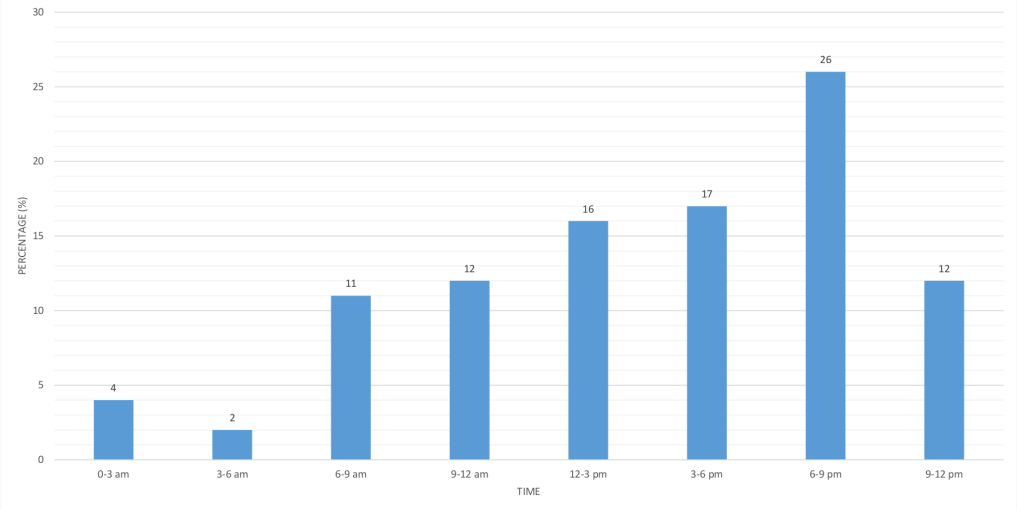

Figure 8.9: *Average usage of the service during the day.*

These percentages outline the usage of the car rental service throughout a day. It highlights that the service tends to be used more often in the usual leisure time (6-12 pm) by more than a third of all car rentals.

By plotting all parking times on top of each other in a heat map, the colouring indicates different periods in parking time per car. If a car is parked for more than a day it will be coloured in red and if it is parked less than a

day it will be coloured green. Interestingly, vehicles parked in the outskirts of the city are much more often to be left for more than a day compared to cars parked closer to the city centre.

Figure 8.10: *All rental drives in 2017, coloured by the lowest standing period (green) and longest standing period (red).*

The data set may give even more analysis and answers to the usage of the free-floating car rental service. It is evident that the data set may hold even more valuable information. The full data sets are available for download on the author's GitHub repository [102].

The data gained from Catch a Car is very insightful. Not only does it provide the movements of cars but also enables assumption on the course of business, e.g. estimated total revenue. The key for analysing this service is the long-term data gathering. Moreover, from this very example real-time data such as moving rental cars can be highlighted. Thanks to the cron jobs which help schedule commands on the server and the building block architecture, the Web observation was able to run smoothly for over a year.

8.2 Public Transportation Data

Infobox	
Architecture:	Building Block System
Data format:	HTML
Filtering:	No
Frequency:	30 sec
Login:	No
Uptime:	Mar 31 - May 18, 2017
Volume:	20,736 KB per day
# of requests:	2,880 per day

Transport operators all over the world started to publish data of their services in real-time on the Web, e.g. the San Francisco Metropolitan Transportation Commission [195] or Transport for London [196]. These data sets include arrival, departure and delay times on all stations and hold useful insights for a better understanding on how transportation systems work. In other words, it may hold valuable information for commuting within the public transport system. Unfortunately, not all transportation operators openly publish their data, even if they are publicly funded. Although sometimes unstructured data is available on the Web, making use of it tends to be more sophisticated. If neither data sets nor interfaces are available, users without technical skills are kept in the dark. Such Web content evaporates after a few minutes and is irrevocably lost. To prevent losing such data, a Web observation may help to preserve data from transport operators from evaporation.

8.2.1 Web Observation Task

The local transport operator in Basel, Basler Verkehrsbetriebe (BVB), is a perfect example for a public company having a Web service that does not provide collected data sets, but an online mask for customers with live data only. The operator serves a small dense transportation network of 113 miles (181 km) if all buses and trams are taken together [197]. The Web resource of the transport operator outlines information about all stations including the information boards of all buses and trams that move in and out of a station. However, it would take an enormous amount of effort to check all the information manually. No history or statistics are being kept available for the public; departure times of connections are gone as soon as time has passed. Ultimately, data, in a metaphorical sense, evaporates and is lost without a possibility to be obtained from the Web page at a later stage. Thus, it would be beneficial to observe delay times of trams.

8.2.2 Verifying the Results

A Web observation may collect all data shown in the browser. By assuming the collected data set is complete, statements can be made about the transport operator's service. It is thus evident that real-time data is provided from the latest data collection, e.g. current delays at a station, performance of the network and perhaps optimisation of schedule.

For verification reasons, approximately 100 tram passings were personally observed by the author at the station "Barfüsserplatz" and compared with the data collected by the Web observation. Based on this verification, it is possible to deduce that the Web observations matches the actual station board by an accuracy of 95%. 5% are missed and are due to the collection time interval set between 1 to 3 minutes. Time differences of less than a minute are not caught, while five trams did not even match the station board's indication, one tram was actually departing earlier than recorded. Each arriving tram at the station "Barfüsserplatz" that has at least one minute delay is observable as a state. Therefore, it will be collected and stored as the actual Web observation data set. For the duration of approximately three weeks, the Web observation extensively stored states from trams passing this particular station.

Criticism

The Web observation collects data from the local transport operater BVB that is also shown to the passengers on station boards but is also available online. Therefore, this might be weaknesses within this method built in this Web observation application. Even if a delay of 2 minutes is indicated on the station board or on the Web, it may actually be a delay higher than 1.5 minutes which was rounded for practical reasons by the operater. Since the threshold is unknown, a rounding error margin may distort the result.

Yet, the transport operator BVB has kindly disclosed their own data for the Web observation period which enables the comparison between our and the official data (April 26 - May 18, 2017). There are two separate systems for measurement of incidents. The transport operator does provide information on station boards on trams passing to inform passengers in the station. Data from the station boards bases on rounded minutes. Additionally, an on-board information station in each vehicle provides measurements on the tram's journey which is far more precise and collects information on a second basis. Data from the station boards bases on rounded minutes. The transport operator defines delays in case more than 2 minutes have passed from the scheduled operation.

For an exact on-board measurement, the station "Barfüsserplatz" has four distinct measurement points [198]:

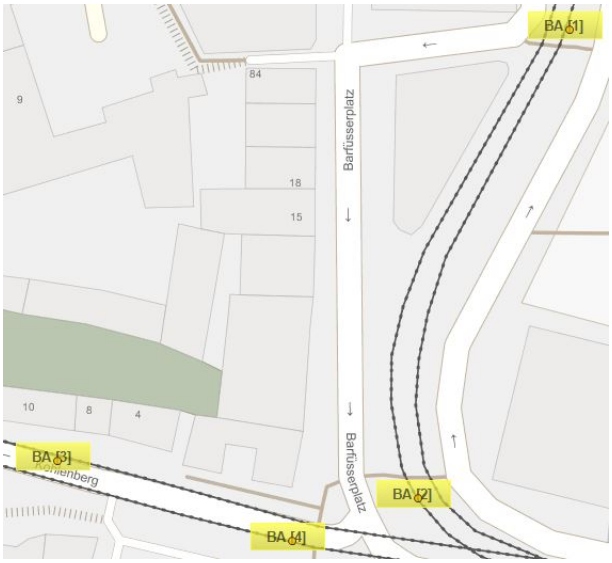

Figure 8.11: *BVB measuring points of station boards in yellow at the station "Barfüsserplatz".*

The measure points determine the direction of the tram: BA [1] towards "Marketplace", BA [2] towards "Heuwaage", BA [3] towards "University of Basel", and BA [4] towards "Bankverein". The data set that has been disclosed reveals the measurement points of the tram line and also indicates the time delays. By using a delay threshold time of higher than one minute (> 1 min), a total of 576 hours of delay results from the raw data. In contrast to the measurement of 437 hours of delayed trams within the measurement time, the Web observation is more reluctant to observe incidents. From the information disclosed by the BVB, it is not clear whether the station boards or the on-board measurement is correct. However, by assuming that the on-board measurement is correct, the Web observation is distorted by more than 10 percent. However, the Web observation falls significantly shorter than the internal data of the transport operator BVB. Therefore, one can assume that the magnitude of incidents is correct, but it rather fails to store longer delay times. Even by defining a delay higher than 2 minutes (> 2 min), the delays measured on board of a tram would still be about 380 hours. [198]

8.2.3 Interpretation of Data

With the help of a Web observation, a data set with states from station board sensors including arrival and departure times could be obtained. From the data, an interpretation can be drawn e.g. whether a tram line has more incidents in comparison to other lines. Conclusions can be drawn from this station board to the entire network. In case the trams are delayed due to congestion, the chosen station "Basel, Barfüsserplatz" as one of the major junction in the city will most likely be affected. Unsurprisingly, about half of all passing trams were registered with at least one minute delay.

As shown in table below, we could extract 16,417 passing trams over a time period of three weeks:

tram line	# delayed trams	%
1	12	0.1
3	1,870	12
6	3,530	22
8	2,163	14
11	1,930	12
14	2,611	17
15	1,238	8
16	1,700	11
17	723	4

Table 8.1: *Total delayed trams at the station "Basel, Barfüsserplatz".*

Data for tram line 8 is conspicuous. It has the third highest incident rate with fairly long delay times. It repeatedly shows delays of more than 15 minutes. These would pile up in the afternoon and on Friday and Saturday. Tram line 8 is used for cross-border transportation to Germany, delays of the tram line may therefore be based on the high frequency of people using this line for cross-border shopping especially on weekends. It might be stated that when too many Swiss residents decide to do their shopping in Germany, the tram line 8 is most likely delayed, be it both because of the amount of customers to be transported as well as having to share the road with people using their cars to cross the border. This could be further investigated by evaluating trams that crossed the border against trams that switched direction and stayed in Switzerland. Additionally to sharing the road, the tram line 8 must cross the road twice at the border crossing "Weil am Rhein", Germany. This results in delays due to congestion and is the most plausible reason why this particular tram line has massive delays, e.g. 55 minutes.

The longest delay observed was registered for the line number 14, which had been delayed for 51 minutes when heading towards the outskirts of Basel. This is due to an accident on the same day that has blocked the tram tracks. Interestingly, tram line 6 had the most incidents with more than 3,500 delays. Together with the tram line 14, both lines are prone to frequent minor delays. Regarding the low delay percentages of tram lines number 1 and 17, an explanation may be given based on the return of trams to the depot in case of line number 1 (it only stops at the station when returning to the depot) and rush hour only operation in case of line number 17. Overall, tram lines 3, 11, 15 and 16 have fairly good performances.

All tram line delay entries taken together amounted up to 26,253 minutes in three weeks only or almost 437 hours of delay time which is the equivalent of about 40% of all trams operating on the transportation network of Basel during the observation period. On the contrary, this leads to the conclusion that 60% of all passing trams are on time and therefore are not delayed.

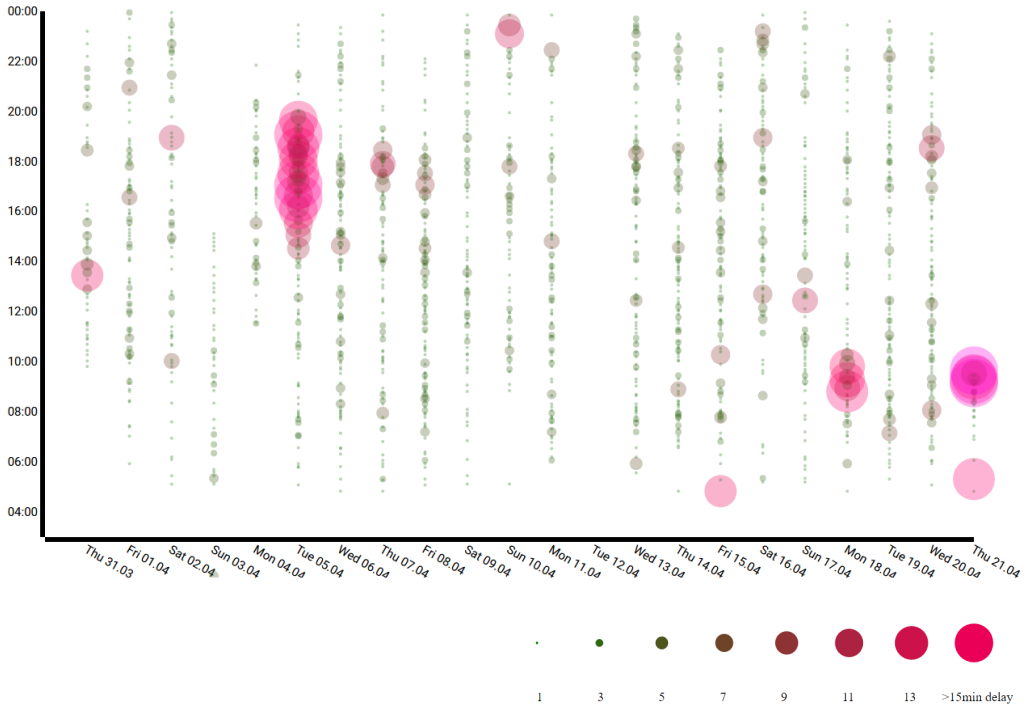

Figure 8.12: *The scatter-plot outlines incidents of tram lines in the observation period (March 31 – April 21). A major disturbance can be identified on April 5 (Tue 05.04). Depending on the selected line, performance is highlighted with a delay bubble. Best case scenario would be no colours at all. Colour-scale from green (delay = 1min) to pink (delay >15min).*

Disturbances of the network traffic of line 3 are visualised for 21 consecutive days from 0:00 to 23:59 o'clock. The plot accumulated delays at different day times and weekdays. The reasons for those delays tend to be rush hour related. A detailed monitoring of delays may give schedule planers the possibility to directly measure the outcome of their work. [79]

Data of the local transport operator is of high interest to citizens. It helps to understand current traffic movements on the public transport network. Due to the renewal of the Web resource the Web observation had to be adjusted which resulted in data gaps. One finding from this research project is, that it is vital for long-term measurements to have a stable system in place that makes sure, data is collected and otherwise alerts the user if collection of Web data fails.

8.3 News Article Comments

IT has a strong influence on the media, news are increasingly consumed online and the print media is losing readers. A Web observation could possibility get an insight on how journalists update their stories on news portals. Based on these changes in the personal news consumption from print to online, online news portals are among the most frequented Swiss websites [199].

Infobox

Architecture:	ECA System
Data format:	HTML
Filtering:	No
Frequency:	2 min
Login:	No
Uptime:	Sept 4 - 18, 2015
Volume:	∅ 43,200 KB per article
# of requests:	720 per day

Online newspapers encourage readers to leave a comment. This option of expressing thoughts and opinions is widely used and can lead to online discussions. It can commonly be assumed that the reader leaves a comment on a topic that seems important to him or her or arouses his/her interest. These comments of Web users could serve as a basis to identify hot topics that predominantly occupy readers in newspapers.

8.3.1 Web Observation Task

News portals provide commenting functions in which users may leave a comment. Since these comments are available in clear text, a Web observation might observe comments as soon as they are published. Questions arise on

whether there is a publishing pattern. Therefore, a successful Web observation gives answer to:

- **Number of headline stories;**
- **Number of comments per news section;**
- **Number of comments per article;**
- **Number of articles blocked from commenting;**
- **Total number of articles;**
- **Topics of interest.**

The comments may also serve as a basis to find the topics that predominantly employ the readers of a paper. Simonet (2015) has already assessed a method to extract data from the Web resource of a Swiss news portal which has been adapted and used for the author's Web observation [200].

8.3.2 Verifying the Results

Verifying the results is a fairly easy task. A simple alignment of the data set with the news portal reveals whether all comments have been collected. More interestingly the Web observation gathered more comments than that are actually available. One explanation might be that controversial comments will be removed from time to time even though they have been published before.

8.3.3 Interpretation of Data

In a first step, a Web observation was set in place to observe headlines of 6 news portals in Switzerland. By using a GitHub colour scheme the level of headline changes could be visualised in the following figure:

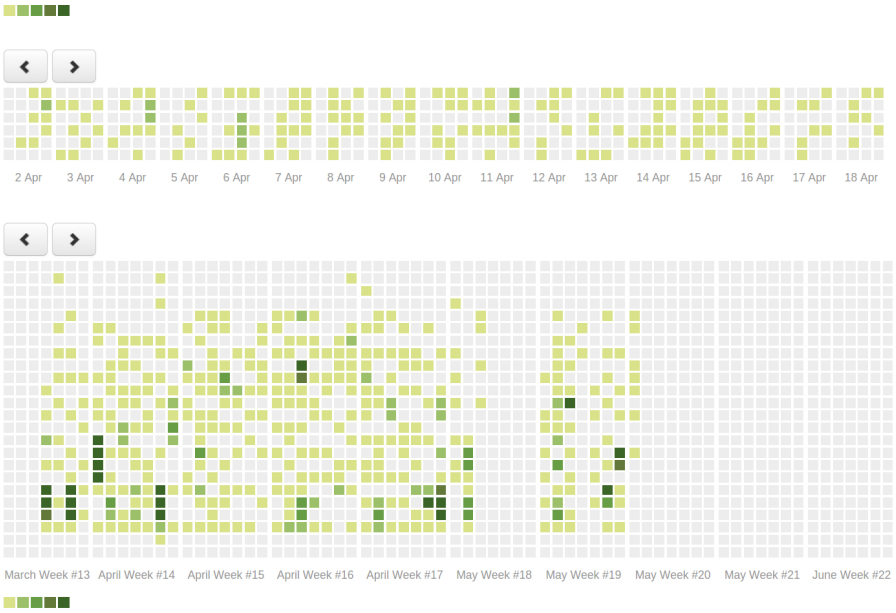

Figure 8.13: *Headline changes of the Swiss newspaper "Tagesanzeiger". Top figure corresponds to the hours 0-23 per square starting from left to bottom, bottom figure corresponds to the hours 0-23 per square starting from top to bottom. Colours: less than 2 items; 2 and 4 items; 4 and 6 items; 6 and 8 items; more than 8 items.*

Although the figure does indicate hours with a lot of headline changes, the individual headline contents is not yet certain. These need to be examined to find out what has led to the accumulation of headlines. The content of the headlines makes it possible to explain the figure as follows:

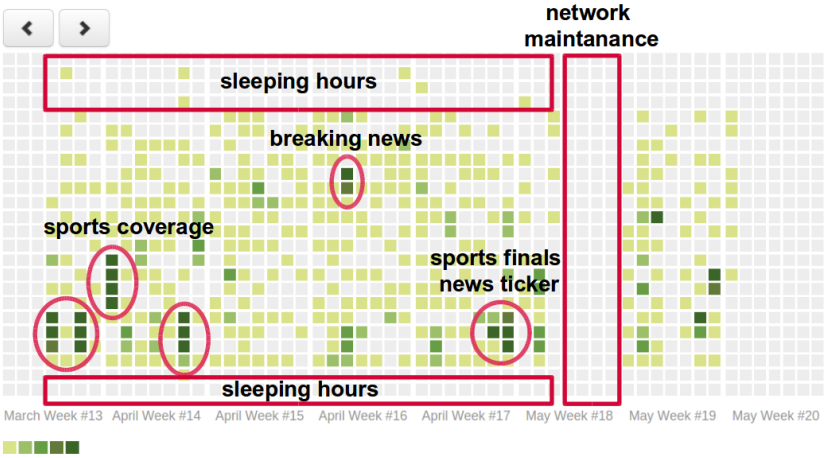

Figure 8.14: *Hours 0-23 per square starting from top to bottom. Colours: less than 2 items; 2 and 4 items; 4 and 6 items; 6 and 8 items; more than 8 items.*

Over a period of 20 days, comments have been collected from the news portal "20 Minutes". In all sections mentioned, the news portal published more than 1,000 articles. For 594 articles the commenting function was activated. In total, around 50,000 comments have been published by active users.

Section	# articles	comments allowed	# comments	∅ comments/article
Switzerland	253	124	24,378	197
Foreign affairs	176	19	1,281	67
Economy	123	68	6,854	101
Sports	148	137	7,871	58
People	68	37	1,315	36
Entertainment	92	50	2,326	47
Digital	74	67	2,574	38
Knowledge	83	64	2,182	34
Life	53	28	1,070	15
Total	**1,070**	**594**	**49,851**	**65**

Table 8.2: *Comments collected from categories on the news portal "20 Minutes". [200]*

Most of the articles have been published in the section "Switzerland". Followed by the sections "Foreign affairs", "Sports", and "Economy" while less than 100 articles were published in the remaining sections. Interestingly, articles that have commenting functions available show an unexpected result. In the section "Switzerland" about 50% of all articles are open for comments while only about 10% of the articles in the section "Foreign affairs" were open for comments. People also seem to comment less on foreign affairs. In the sports section "Sports" almost all articles were open for comments. In the remaining sections, the percentage of articles with activated commenting functions is higher than 50%, remarkably high in the section "Digital" in which 90% of the articles were open for comments.

About half of the almost 50,000 comments were written on articles in the Switzerland section. The sections "Sports" and "Economy" follow with 7,871 and 6,854 comments respectively. The remaining 20% of comments are distributed among the six other categories with numbers between 1,070 and 2,326. Thus, the Switzerland section with an average of about 200 comments per article with activated comment function is clearly the most commented category and therefore probably also the most read section. Here too, the categories "Economy", "Foreign affairs" and "Sports" follow with around 100, 67 and 58 comments per article. Very few comments are published on articles in the

"Lifestyle" section. On average, the category only reaches about 15 comments per article.

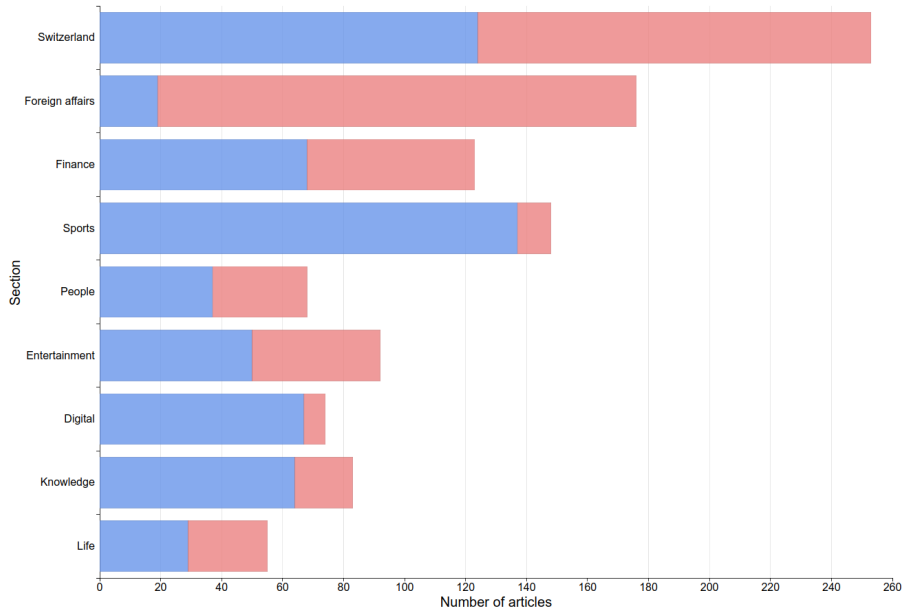

Figure 8.15: *Articles per section; commenting function on (blue), commenting function off (red). [200]*

The Switzerland section is far ahead in terms of numbers. Almost 25,000 comments were saved in the category. The most commented article with 953 comments is the article about commuting with the Swiss Federal Railways becoming more expensive in the future. The article describes a proposal for a new cost concept designed to lighten public transport during rush hours. One possibility is the participation of many commuters during their commute while reading the newspaper on their smartphone. Also popular are articles about "drinking and driving", "home ownership", "popularity of the Swiss Armed Forces", and the weather forecast in general.

Only about one fifth of the articles that can be commented are assigned to the section "Switzerland". Another fifth of the articles open to comments belongs to the section "Sports" which counts almost 8,000 comments. The third most commented area is the section "Economy" with a total of about 7,000 comments. Accordingly, these three categories make up the most comments over the time of the Web observation.

For the Sports section it has to be considered that the articles with the most comments collected during the time of the observation dealt with a scandal of the football association FIFA. Incidents of this kind are not commonplace and are not associated with traditional sports journalism. The results could therefore

differ greatly from the usual numbers and should be treated with caution. In all other sections, one can assume that the numbers correspond to the usual user behaviour.

From the collected data, three peak times per day can be identified in which many comments are published. The first peak time is roughly between 6 and 9 o'clock in the morning, followed by a second peak time at noon and a third peak time in the late afternoon and evening. Most of the comments are written in the first four hours after the article is published. Thereafter, the commenting activities flatten and remain at a constant level until 24 hours after article publication. In the following twelve hours, the number decreases or in some cases the commenting function is even turned off by the news portal. After 36 hours, only a few new comments can be expected. Yet, the various discussions about the football association FIFA tend to be long-lasting. After about 20 hours, new comments have still been published. Perhaps the high participation in the discussion is a consequence of comments made on comments.

Since a comment is monitored before the news portal publishes it, timestamps on the comment do not correspond to the actual time it was available on the news portal. Generally, the comments are published after a short period of time. Around 65% of the comments were published within an hour. Another 20% will be published between one and six hours. The remaining 15% of the comments will be published up to 24 hours after the comment was written.

On average, a comment is published after two hours and 40 minutes. For certain topics, however, comments seem to be published at a faster rate, for example in sections "Switzerland" and "Economy". Comments are normally published after two hours. In the sections "Digital", "Lifestyle", "Entertainment", "People", and "Knowledge", it takes the news portal more than five hours to process the comments for publication.

The results of the evaluation outline that the majority of the readers comments articles in the section "Switzerland". The majority of the comments are written within first hours after publication of articles. After two days, the commenting function is deactivated.

Conclusively, the collection of comments generates a higher amount of data than expected. It is not only the comment that has to be stored, there are a lot of flags within the comment to store as well. Likes, dislikes, or comments on comments are just a few of additional data fields to store from which a meaningful insight can be deducted. However, comments often allow various interpretation possibilities which makes it hard to have a non-biased point of view.

8.4 Price Observation of EasyJet

EasyJet is an airline company flying to many destinations in Europe. The promoted price model of the company bases on the principle that the earlier a booking is made the cheaper the tickets will be. But is that really so? If a Web observation is conducted, it should prove that prices increase the closer the flight date is.

Infobox

Architecture:	ECA System
Data format:	HTML
Filtering:	Yes
Frequency:	6 h
Login:	Yes
Uptime:	July 10 - Dec 17, 2015
Volume:	256 KB per day
# of requests:	4 per day

8.4.1 Web Observation Task

Holidays or business trips are usually planned months ahead of the actual flight. Therefore, it seems beneficial to know at what time it is best to book flights; best in terms of cheapest. The examined study object EasyJet only has an online presence, they do not provide any kind of physical travel agency. Often, during bookings it is highlighted that the ticket price will increase if "only a few seats are left".

8.4.2 Verifying the Results

The Web observation may collect all data shown in the browser. It is assumed that the collected data set is complete, therefore, calculations can be made about the EasyJet price model. However, the prices collected cannot be verified from the service since prices are constantly changing. Although EasyJet provides a tool to find low fares online, this tool only focuses on prospective flights and thus cannot be used for dates that lie in the past.

8.4.3 Interpretation of Data

The data set collected by the Web observation is based on several flights selected prior to the collection. The Web observation was focused on flights being sold between July 10, 2015, and December 17, 2015. By plotting all prices of flights per day the following figure results:

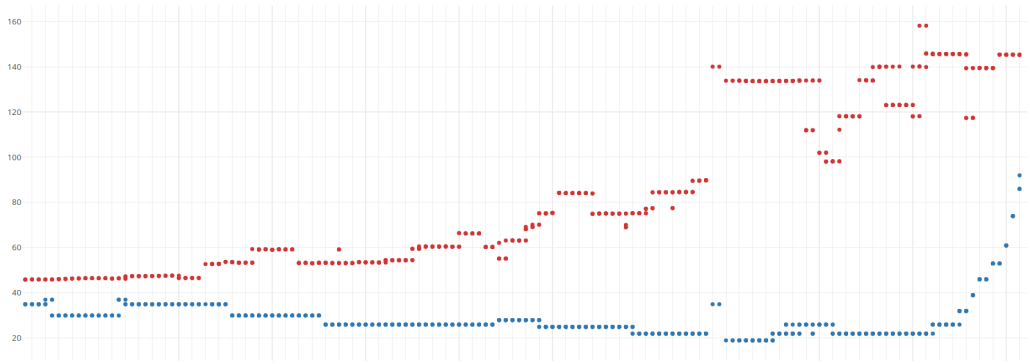

Figure 8.16: *Observed prices from July 10 to December 17 (flight date). Red: flight Basel (BSL) to London Gatwick (LGW), blue: flight London Gatwick (LGW) to Basel (BSL). X-axis corresponds to the time while Y-axis corresponds to price in CHF.*

Interestingly, there is no constant rising tangent for the flight Basel to London (red). The price development seemed arbitrary and although certain trends could be made out in the data, prices did fall back several times before rising again. A key observation made is that flights particularly increased in price on weekends and on Fridays. This may be explained by higher sales/clicks during these days, however, from the observation itself no explanation can be given. Even more interestingly, prices tend to fall on Mondays and Tuesdays to a certain level. The reason for this behaviour is not yet clear, but an explanation might be a manual reset for more sales. In contrast to the flight back to Basel (blue), prices fell until December 5, when a sharp increase could be observed. In the days just before a flight, strongly increasing prices could be observed. The same observation could be made for several other flights.

EasyJet warns customers that prices may change in two separate warnings: "price alert" or "available seat warning". One would expect that after such warnings prices would increase. Yet, in a monthly observation, a total of 426 price warnings have been posted, but in 30 cases a price decrease could be measured. This raises questions about the actual price model of EasyJet and whether these warnings are in place to drive sales.

Nevertheless, it is an instrument to guide customers into buying tickets and it provides no actual information on the true amount of seats taken on a specific flight. Ticket prices have a tendency to rise substantially the closer it comes to the travel date. Additionally, and based on the observed fluctuations in the data, it can be concluded that prices on weekends and Fridays are rising significantly higher than on other days of the week. This gives an insight into Easyjet's pricing model but may also reflect user habits.

8.5 WhatsApp Meta Data

The idea for taking a step towards communication applications was inspired by the book "Mining the Social Web" [201]. In case of the messenger service WhatsApp, the only data available of other users are status messages such as "last seen", "typing...", and "online". Otherwise, the information shared over WhatsApp is encrypted end to end between two respective users.

Infobox

Architecture:	Building Block System
Data format:	HTML
Filtering:	Yes
Frequency:	1 min
Login:	Yes
Uptime:	May 7 - 28, 2018
Volume:	94,320 KB per day
# of requests:	1,440 per day

To make use of this the service, only a valid phone number and the minimum age of 16 is required (which is an outgrowth of the European Union's GDPR but can easily be circumvented by entering another birth date).

As follows a screenshot of the WhatsApp Web page:

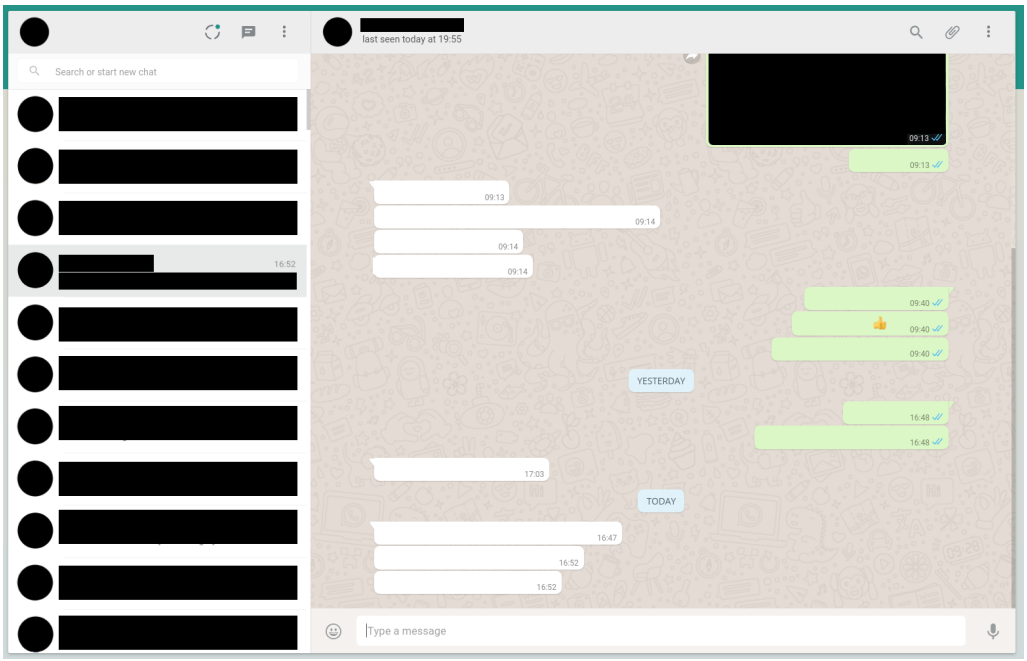

Figure 8.17: *Screenshot "web.whatsapp.com", WhatsApp Web interface.*

8.5.1 Web Observation Task

WhatsApp provides a Web interface for sending and receiving messages. Since it is on the Web, a Web observation may collect data from the service. Questions

arise in regard to the conclusions that can be drawn from the data available on everybody who has a WhatsApp account to a real-life person. Perhaps the service may have privacy issues in terms of personal data. Can a Web observation obtain sleeping times, daily routines and deviations thereafter and perhaps identify communication patterns and other users?

As follows a screenshot of the authentication Web page of WhatsApp:

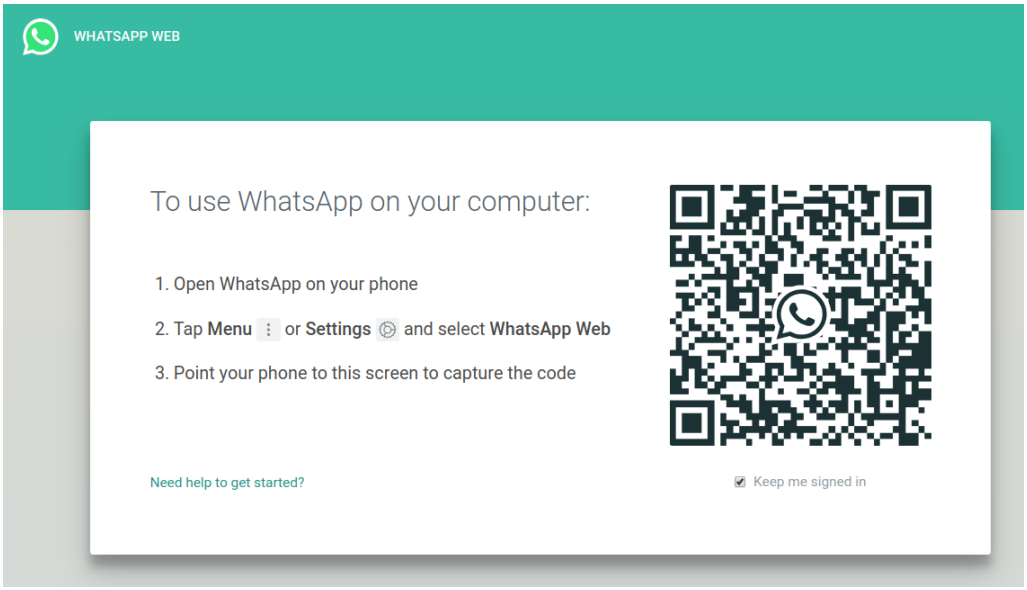

Figure 8.18: *Screenshot "web.whatsapp.com", WhatsApp Web login page.*

8.5.2 Verifying the Results

Without the informed consent of a subject in regard to the collection of its messenger data, it is difficult to identify a plausible verification of the collected data. Since this Web observation shall not be considered a privacy infringement, the author observed only his personal WhatsApp account. With this knowing, it is fairly easy to assess the quality of collected data. For that purpose a prepaid card with a Swiss mobile number was used to observe the author states such as "last seen", "typing...", and "online". By knowingly collecting meta data of the author's account it needs to be considered that it consciously or unconsciously affects the author's behaviour. In regard to privacy, this choice has been made to avoid conflicts with the law. However, no obvious pattern change could be observed which also led to the conclusion not to disclose this kind of data to the public in great detail.

However, while observing one's own account, at the same time the author also observed the communication flow between himself and his communication

partners which would also allow the indirect observation of all his communication partners. If such communication partner data was included in the collection, it would not matter whether he or she used the Web surface or the smartphone application.

8.5.3 Interpretation of Data

The collected data set might reveal online times and other behaviour, depending on the personal usage of the messenger. From the plain data it can be derived that the author used the service throughout the day with a slight increase in the evenings. In regard to time invested in the usage of the application, an assumption can be made that the author spent more time with the messenger service after he left work which would then further support the assumption that the author is employed in an office job and does not work in shifts. Approximate sleeping times could be clearly determined during night times when no activity could be observed.

Moreover, no other communication partner was actively sending messages anymore. However, this approximation fails to exactly identify the sleeping hours of the author. It indicates that the person is not active during these times. Nonetheless, daily routines and deviations thereof correlate with the usage. Even if the security setting "hide last online status" is enabled, the current "online" status is still visible. By monitoring the online states, the status "last online" can easily be reproduced with observation timestamps. [202, 203]

Due to limitted access to meta data, it is difficult to detect communication partners, but if the observer has a presumption in regard to their online behaviour, an inference might be possible. Since the author and any other person with technical understanding could use this method to extract meta data on their own or another person's choice, questions about privacy arise.

As follows the average of the author's WhatsApp usage:

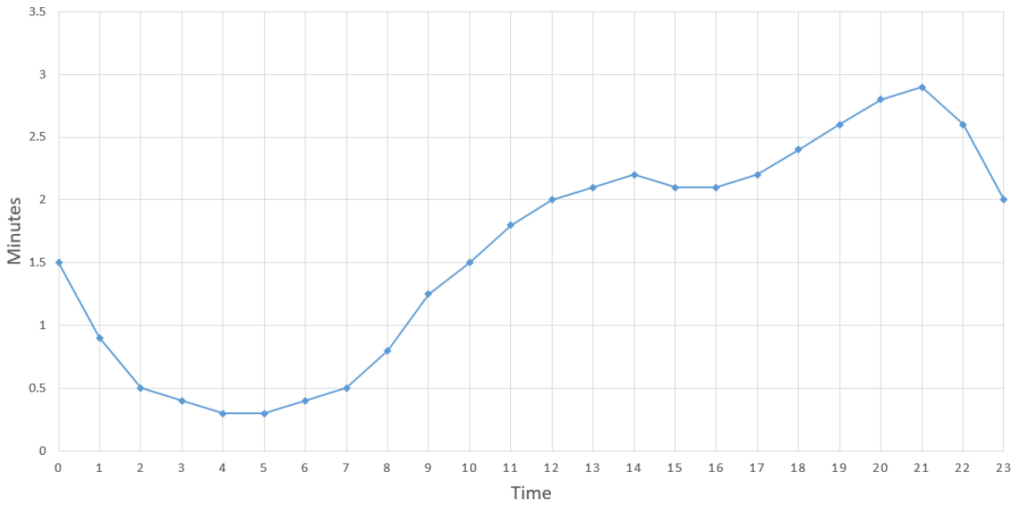

Figure 8.19: *Average usage per minutes of a person per hour of the day. The Web observation has been conducted from May 7 – 28, 2018.*

Daily behaviours or certain routines could easily be determined from this very Web observation. An observation period of less than a month provided much more information than initially expected. Only from analysing this very little meta data the author could start to outline his own behaviour patterns. If this Web observation was applied on a broader scale, even more would be possible. Basic personality profiles could be determined without much effort, yet publication thereof could lead to severe privacy issues, e.g. intrusion into private behaviour. Therefore, the author refrains from fully disclosing the data sets collected through his Web observation and fully limits his review to this written reproduction of the work.

To create a reliable daily schedule and personality profile of a user's habits, a particularly long-term Web observation has to be carried out, if all fluctuations due to weekends and holidays shall be included in a way it does not falsify data. This allows to connect the online times to the schedule and determines whether there are deviations. Conclusively, the measurement must collect data over a longer period of time. Interruptions would weaken the results.

This chapter illustrated five Web observation scenarios and proved the applicability to different scenarios. Moreover, it answered the third research question. The next chapter "Conclusion and Outlook" will outline the contributions of this work and will draw a final conclusion.

Part VI

Conclusions & Outlook

Chapter 9

Conclusion and Outlook

> Basically, our goal is to *organise the world's information* and to make it *universally accessible* and useful.
>
> Lawrence Edward Page, Co-founder and CEO of Google, (born March 26, 1973)

In this final chapter the answers to the research questions and relevant contributions will be presented. Finally, a conclusion, limitations thereto and future work will be discussed.

9.1 Contributions

This thesis presents research undertaken to facilitate Web observations. Chapter 1 and 2 provide a literature review of the topic such as data aspects and technologies which are made use of. The first two chapters form the theoretical and practical foundation of this thesis.

While chapter 3 defines the research contribution and research problem, it moreover refines the research questions based on the findings of the literature review.

Chapter 4 outlines legal and political perspectives of data in order to answer the first research question. Furthermore, this thesis outlined the most important current legal and political perspectives that play a crucial role in the age of the Internet. Data collections have recently come to the attention of the European

Union which led to the introduction of the GDPR. Thanks to its extraterritorial reach, it is also to be regarded by Switzerland and other Non-Member States of the EU. Which is one of the reasons why many websites now openly provide an information on cookies to its users.

Chapter 5 discusses the issues with Web observations before data can be collected. The collection of states of Web resources is one of the key aspects of a Web observation. The steps towards a flexible and generally applicable Web observation are defined in an awareness model. As outlined in this chapter, the most focal point is the examination of the Web resource. With this in mind a thorough understanding can be gained what must be known and taken into consideration from the Web resource when programming the actual Web observation in order to collect useful data.

In chapter 6 two architectural and technical solutions for Web observations are presented including the programming code of two distinct systems. First, the Event Condition Action (ECA) System that wants to achieve a reactive event detection system but also includes Web observation capabilities. Second, the ECA System enabled the programmability of events for planned reactions by using event rules, e.g. push notification to a Web service or poll of Web resources. From the ECA System it could be learned that storage of data was a major problem and it was tried to tackle both Web services with APIs and ordinary Web resources. With the intention of having a flexible architecture for the sole purpose of Web observations, there was a need for a new architectural approach. Container based software enables software packages from different operating systems to run together and gives the flexibility of choice that other environments cannot achieve. Eventually, this led to the development of the building block system based on containerisation which indeed had the flexibility to deal with versatile Web resources. Conclusively, a recommendation for the building block system is outlined with the most flexible solution to solve versatile problems. Eventually, chapter 6 answers the second research question.

Chapter 7 presents benchmark measurements for revealing the limits of the building block system. These measurements showed the latency of the Web and proved that the Intranet can handle higher observation interval frequencies than the Internet. Moreover, it indicates the maximal interval frequency of the Web.

Chapter 8 presents five distinct examples of Web observations, its collected data sets and visualisations.

The following five scenarios are presented in detail:

1. Free floating car sharing service and the movements of cars;
2. Public transportation data of vehicles;
3. News article commenting of users;
4. Price observation of flights;
5. Meta data of the online messenger WhatsApp.

These scenarios highlight and describe data that is in constant flux and changing. It is therefore interesting for a certain audience to sift through this data. Nonetheless, the scenarios represent fine examples for suitable Web observations. Meaningful insights can be gained from them and in some cases adapt its usage to identified patterns, e.g. booking of flights. Thus, the conclusion in chapter 9 answers the third research question. Finally, this thesis deals with current developments of the law and computer science until August 2018.

9.1.1 Summary of Contributions

Summarised, the main research contributions of this thesis comprise of as follows:

- A literature review of data aspects, technologies for Web observations, legal and political perspectives;
- A definition of analysis tasks for the examination of Web resources that should be undertaken before a Web observation may actually begin;
- An overview of architectural and technical solutions used for various Web observations;
- A recommendation for an architecture for Web observations applicable over a longer period of time;
- A benchmark measurement of Web observations in order to get to know the limits;
- A diverse set of Web observation scenarios based on two architectural approaches;

- Results from Web observations and its data sets, interpretation and visual representations thereof;
- A list of suitable Web resource scenarios for Web observations.

9.2 Conclusion for Web Observations

The creation of a Web observation involves many architectural designs that need to be considered for a proper use. Several research projects with different designs have been conducted. Considering all these decisions taken, the building block architecture was created as a final research product. Such an architecture also symbolises the exploratory use from algorithms to the ECA System to a Web observation system and additionally outlines the path towards the final flexible building block architecture. Several scenarios have been tested that gathered data from many different Web resources. Many data sets have been collected, evaluated and visualised to extract a meaning. In the end, the data sets were published on GitHub [102].

Based on all these steps taken over the time of the study, the author was finally able to present a flexible and versatile architecture for Web observations that is able to perform data extractions from Web resources. The author considers the container based architecture as the most reliable and flexible design form to conduct Web observatories in contrast to single scripts and the ECA System.

The main benefit of a container based architecture is its composition of single building blocks that allow an easy exchange of each block in regard to the Web observation project. This architecture has been used in the research project on public transportation and its collected data sets were published [79]. During this project, the author could easily adapt the building block architecture to the Web resource. This flexible adaptation of the architecture which allows to switch from a software to another, led to the conclusion that a container based building block architecture is the most suitable for a flexible and versatile Web observation.

Yet, using the container based Web observation, several results can be drawn from the data collected. Thus, before data sets and results can be published, they have to be verified in regard to the quality. The container architecture is able to collect data from any given Web resource. In short, the building block architecture collects states from Web resources in which only the content creator can explain why updates occurred.

9.2.1 Web Data Evaporation

During the time of the study, the author had encountered on several occasions that the collection of data by a Web resource might be rather unwelcome. For example Catch a Car does not want to publish any kind of data at all claiming this to be internal business information. In the case of WhatsApp collected data might possibly infringe the privacy rights of a person that is being observed by a Web observation. That is the reason why the author decided against the publication of data sets that could infringe privacy rights. Nonetheless, from the outlined scenarios, it can be stated that many more calculations can be made with and conclusions on business and people be drawn from the collected data.

Web technologies empowers Web users to collect data from the Web. The awareness of people for data collectors is still quite low. However, the collected data had not been available for the general public before neither was it shown in real-time. With the container based system the Web observation is now easily able to extract and collect a broader amount of Web data than possibly ever before. Moreover, the author was able to collect (real-time) Web data before its evaporation. On many Web resources data is not included in a state to last, but will evaporate over time and is irrevocably lost for the Web users. These scenarios are particularly suitable for Web observations and therefore answer the third research question. A non-exhaustive list can be found below:

- **Price developments (up, down, stable)**
  E.g. hotels, flights, goods and services;
- **Position based on geo location**
  E.g. car sharing, flight tracker, ship traffic;
- **Public transportation / traffic flow**
  E.g. train, bus, tram;
- **Online Messengers**
  E.g. Facebook, Skype, WhatsApp;
- **Page rank positions**
  E.g. Google, Bing, DuckDuckGo.

From all these data sources a container based architecture could collect numerous data sets that provide a broad insight into the efficiency and workability of these services. While a Web observation can be an interesting working

tool for a data scientist, the collected knowledge may help a user to identify meaningful information and services which can support everyday choices.

9.3 Limitations and Future Work

This work can be meaningful for researchers, citizens, media, etc. that want to understand data sources on the Web. The provided architectures illustrate numerous applications of Web observations that were able to collect meaningful data and therefore give interesting insights into the content provider of the Web resource. Nevertheless, the building block architecture must be further utilised on scenarios in order to further test the architectural design.

However, this work does come with several limitations. A high amount of effort has been given to a scenario for the evaluation of price movements of goods in several online shops. Depending on the Web resource the price movements were behaving very differently. Some online shops obtain all kinds of information from the browser and use it to determine whether a Web user is likely to have more buying power than others (higher price). This work could not provide hard evidence for such kind of factors. The only thing that could be established was some sort of an assumption that online shop operators seemed to adjust their prices to their customers. However, it might be interesting for further research to analyse customised pricing strategies in depth. Such an analysis could unravel the mechanisms of the pricing models for adapted usage behaviour.

Another limitation encountered is that the building block architecture needs a certain programming skill-set in order to be able to use the system. The programming of code for a Web observation in order to collect meaningful information from a Web resource still is a manual task. Code can be reused, but it has to be specifically written for a Web resource. The user focus is therefore on IT professionals and not on the general public. However, this could be mitigated by introducing pre-existing code that could be easily applied within a graphical user interface. This could be implemented in the future as a next milestone.

Finally, it is realistic to say that in the near future, such Web observatories will be available either for free or as a pay service for everybody on the Web. Therefore, regulation of data collection will become more and more important in the future. A data provider will have to consider its users and the data's privacy and safekeeping and what kind of information is shared for what purpose. Perhaps soon, such data sets will be analysed by artificial intelligence without

any further need for a specific Web observation to be set up. With more data, a prediction model might be applied for some scenarios to improve the system. Also, with more resources available more data collections could be executed. From the already vast amounts of data existing on the Web, the same amount could be recreated through a Web observation. One of the main advantages from this immense collection will be to make use of untapped potential of knowledge which would otherwise be lost or undetectable. Therefore, data scientists who help to make sense of the collected knowledge will be of utmost importance.

Bibliography

[1] WebScience Trust, "Introduction - WebScience Trust", The Web Observatory: A global data resource for the advancement of economic & social prosperity. [Online]. Available: http://www.webscience.org/web-observatory/about/introduction/. [Accessed: 15-Aug-2018]. [cited at p. 2, 29]

[2] M. Hilbert and P. López, "The World's Technological Capacity to Store, Communicate, and Compute Information", Science, vol. 332, no. 6025, pp. 60-65, Apr. 2011. [cited at p. 5]

[3] J. Toonders, "Data Is the New Oil of the Digital Economy", WIRED, 01-Aug-2014. [Online]. Available: https://www.wired.com/insights/2014/07/data-new-oil-digital-economy/. [Accessed: 15-Sep-2017]. [cited at p. 6]

[4] J. Vanian, "Data Is The New Oil — Fortune.com", Data Is The New Oil — Fortune.com, 11-Jul-2016. [Online]. Available: http://fortune.com/2016/07/11/data-oil-brainstorm-tech/. [Accessed: 25-Sep-2017]. [cited at p. 6]

[5] M. Mandel, "The Economic Impact of Data: Why Data Is Not Like Oil", Progressive Policy Institute, 11-Jul-2017. [Online]. Available: http://www.progressivepolicy.org/publications/economic-impact-data-data-not-like-oil/. [Accessed: 25-Sep-2017]. [cited at p. 6]

[6] N. G. Carr, "IT Doesn't Matter", Harvard Business Review, 01-May-2003. [Online]. Available: https://hbr.org/2003/05/it-doesnt-matter. [Accessed: 25-Sep-2017]. [cited at p. 6]

[7] F. W. McFarlan and R. L. Nolan, "Why IT Does Matter", HBS Working Knowledge, 25-Aug-2003. [Online]. Available:

http://hbswk.hbs.edu/item/why-it-does-matter. [Accessed: 29-Sep-2017]. [cited at p. 6]

[8] "data — Definition of data in English by Oxford Dictionaries", Oxford Dictionaries — English. [Online]. Available: https://en.oxforddictionaries.com/definition/data. [Accessed: 10-Oct-2017]. [cited at p. 6]

[9] "data — Origin and meaning of data by Online Etymology Dictionary". [Online]. Available: http://www.etymonline.com/word/data. [Accessed: 12-Oct-2017]. [cited at p. 6]

[10] IBM, "IBM Archives: 704 Data Processing System", 23-Jan-2003. [Online]. Available: http://www-03.ibm.com/ibm/history/exhibits/mainframe/mainframe_PP704.html. [Accessed: 22-May-2018]. [cited at p. 6]

[11] I. Becerra-Fernandez and R. Sabherwal, "Knowledge management: systems and processes", Armonk, N.Y: M.E. Sharpe, 2010. [cited at p. 6]

[12] P. Checkland and S. Holwell, "Information, Systems and Information Systems: Making Sense of the Field", John Wiley & Sons, Inc., New York, NY, USA, 1998. [cited at p. 7, 8, 9, 10]

[13] H. Schauer, "Information – Basic Concepts", 03-Nov-2005. [Online]. Available: http://www.ifi.uzh.ch/ee/fileadmin/user_upload/teaching/hs09/L1_Information_Text.pdf. [Accessed: 22-May-2018]. [cited at p. 7]

[14] C. Shannon and W. Weaver, "The Mathematical Theory of Communication", Aspects of information theory, Sept. 1949. [cited at p. 7]

[15] J. O. Hicks, "Management Information Systems: A User Perspective", Subsequent edition, Minneapolis/St. Paul: West Group, 1993. [cited at p. 7]

[16] C. French, "Data Processing and Information Technology", Cengage Learning EMEA, 1996. [cited at p. 8]

[17] "Data, data everywhere", A special report on managing information, The Economist, 25-Feb-2010. [cited at p. 8]

[18] L. Gitelman, "Raw Data Is an Oxymoron", MIT Press, 2013. [cited at p. 8, 9, 10, 171]

[19] R. Kitchin, "The Data Revolution: Big Data, Open Data, Data Infrastructures and Their Consequences", 1st edition. Los Angeles, Calif.: SAGE Publications Ltd, 2014. [cited at p. 9, 10]

[20] J. J. Hox and H. R. Boeije, "Data collection, primary versus secondary", 2005. [Online]. Available: http://dspace.library.uu.nl/handle/1874/23634. [Accessed: 10-Oct-2017]. [cited at p. 9]

[21] J. Han, M. Kamber, and J. Pei, "Data Mining: Concepts and Techniques", in Data Mining (Third Edition), Third Edition, Boston: Morgan Kaufmann, pp. 1-38, 2012. [cited at p. 10, 35, 36, 38]

[22] J. Rowley, "The wisdom hierarchy: representations of the DIKW hierarchy", Journal of Information Science, vol. 33, no. 2, pp. 163-180, Apr. 2007. [cited at p. 10, 11, 171]

[23] C. Zins, "Conceptual approaches for defining data, information, and knowledge", Journal of the American Society for Information Science and Technology, vol. 58, no. 4, pp. 479-493, Feb. 2007. [cited at p. 10, 11, 171]

[24] R. L. Ackoff, "From data to wisdom", Journal of applied systems analysis, vol. 16, no. 1, pp. 3-9, 1989. [cited at p. 10, 11, 171]

[25] J. Girard and J. Girard, "Defining knowledge management: Toward an applied compendium", Online Journal of Applied Knowledge Management, vol. 3, no. 1, pp. 1-20, 2015. [cited at p. 11]

[26] A. Bytheway, "Investing in Information: The Information Management Body of Knowledge". Springer, 2014. [cited at p. 11]

[27] C. Snijders, U. Matzat, and U.-D. Reips, "'Big Data': Big Gaps of Knowledge in the Field of Internet Science", International Journal of Internet Science, vol. 7, no. 1, pp. 1-5, 2012. [cited at p. 11, 15]

[28] "History of the Web", World Wide Web Foundation. [Online]. Available: https://webfoundation.org/about/vision/history-of-the-web/. [Accessed: 09-Jan-2018]. [cited at p. 11, 179]

[29] T. J. Berners-Lee, "Information management: A proposal", 1989. [cited at p. 11]

[30] R. Perrey and M. Lycett, "Service-oriented architecture", in Applications and the Internet Workshops, Proceedings, 2003, pp. 116-119. [cited at p. 12]

[31] E. Newcomer and G. Lomow, "Understanding SOA with Web services", Addison-Wesley, 2005. [cited at p. 12]

[32] B. Mobasher, R. Cooley, and J. Srivastava, "Automatic Personalization Based on Web Usage Mining", Commun. ACM, vol. 43, no. 8, pp. 142-151, Aug. 2000. [cited at p. 12]

[33] I. H. Witten, M. Gori, and T. Numerico, "Web Dragons: Inside the Myths of Search Engine Technology". Elsevier, 2010. [cited at p. 12]

[34] EMC News, "New Digital Universe Study Reveals Big Data Gap: Less Than 1% of World's Data is Analyzed", Dec-2012. [Online]. Available: https://www.emc.com/about/news/press/2012/20121211-01.htm. [Accessed: 25-Jan-2018]. [cited at p. 12]

[35] T. Heath and C. Bizer, "Linked Data: Evolving the Web into a Global Data Space", Synthesis Lectures on the Semantic Web: Theory and Technology, vol. 1, no. 1, pp. 1-136, Feb. 2011. [cited at p. 12]

[36] T. Berners-Lee, J. Hendler, and O. Lassila, "The semantic web", Scientific american, vol. 284, no. 5, pp. 28-37, 2001. [cited at p. 13]

[37] "Facts About W3C". [Online]. Available: https://www.w3.org/Consortium/facts. [Accessed: 26-Jan-2018]. [cited at p. 13]

[38] "W3C Semantic Web Activity Homepage". [Online]. Available: https://www.w3.org/2001/sw/. [Accessed: 26-Jan-2018]. [cited at p. 13]

[39] N. Spivack, "The Semantic Web, Collective Intelligence and Hyperdata — Nova Spivack", 18-Sep-2007. [Online]. Available: http://www.novaspivack.com/technology/the-semantic-web-collective-intelligence-and-hyperdata. [Accessed: 26-Jan-2018]. [cited at p. 13]

[40] "RDF - Semantic Web Standards". [Online]. Available: https://www.w3.org/RDF/. [Accessed: 26-Jan-2018]. [cited at p. 13]

[41] T. Berners-Lee, W. Hall, J. Hendler, N. Shadbolt, and D. J. Weitzner, "Creating a Science of the Web", Science, vol. 313, no. 5788, pp. 769-771, Aug. 2006. [cited at p. 13, 14]

[42] B. Shneiderman, "Web Science: A Provocative Invitation to Computer Science", Commun. ACM, vol. 50, no. 6, pp. 25-27, Jun. 2007. [cited at p. 14]

[43] J. Hendler, N. Shadbolt, W. Hall, T. Berners-Lee, and D. Weitzner, "Web Science: An Interdisciplinary Approach to Understanding the Web", Commun. ACM, vol. 51, no. 7, pp. 60-69, Jul. 2008. [cited at p. 14]

[44] C. Arms and C. Fleischhauer, "Digital Formats: Factors for Sustainability, Functionality, and Quality", Archiving Conference, vol. 2005, no. 1, pp. 222-227, Jan. 2005. [cited at p. 15]

[45] Open Data Handbook, "File Formats". [Online]. Available: http://opendatahandbook.org/guide/en/appendices/file-formats/. [Accessed: 26-Jan-2018]. [cited at p. 15, 178]

[46] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives, "DBpedia: A Nucleus for a Web of Open Data", in The Semantic Web, Springer, Berlin, Heidelberg, 2007, pp. 722-735. [cited at p. 16]

[47] D. Kyburz, "Open Data", Explora. [Online]. Available: https://www.explora.ethz.ch/en/s/open-data/. [Accessed: 02-Feb-2018]. [cited at p. 16]

[48] "Open Knowledge International". [Online]. Available: https://okfn.org. [Accessed: 02-Feb-2018]. [cited at p. 16]

[49] "Memorandum", Open Government Working Group, 22-Oct-2007. [Online]. Available: https://public.resource.org/open_government_meeting.html. [Accessed: 27-Apr-2018]. [cited at p. 16]

[50] "8 Principles of Open Government Data", Request for Comments, Open Government Data Principles, 07-Dec-2007. [Online]. Available: https://public.resource.org/8_principles.html. [Accessed: 27-Apr-2018]. [cited at p. 16, 17]

[51] J. Tauberer, "Open Government Data: The Book", 2nd Edition, 6-Oct-2014. [cited at p. 16]

[52] M. Chui, D. Farrell, and K. Jackson, "How government can promote open data and help unleash over $3 trillion in economic value", open data, Oct. 2013. [cited at p. 17]

[53] T. Berners-Lee, "Linked Data - Design Issues", Linked Data, 18-Jun-2009. [Online]. Available: https://www.w3.org/DesignIssues/LinkedData.html. [Accessed: 03-Feb-2018]. [cited at p. 17, 18]

[54] "5 Star Linked Data - Government Linked Data (GLD) Working Group Wiki", 15-Mar-2013. [Online]. Available: https://www.w3.org/2011/gld/wiki/5_Star_Linked_Data. [Accessed: 03-Feb-2018]. [cited at p. 17]

[55] M. Hausenblas, "5-star Open Data", 31-Aug-2015. [Online]. Available: http://5stardata.info/en/. [Accessed: 02-Feb-2018]. [cited at p. 17, 171]

[56] J. R. Mashey, "Big Data... and the Next Wave of InfraStress", Apr. 1998. [cited at p. 18]

[57] "What Is Big Data? - Gartner IT Glossary - Big Data". [Online]. Available: https://www.gartner.com/it-glossary/big-data/. [Accessed: 10-Feb-2018]. [cited at p. 18]

[58] J. Manyika, M. Chui, B. Brown, J. Bughin, C. Roxburgh, and A. Hung Byers, "Big data: The next frontier for innovation, competition, and productivity — McKinsey & Company", May-2011. [Online]. Available: https://www.mckinsey.com/business-functions/digital-mckinsey/our-insights/big-data-the-next-frontier-for-innovation. [Accessed: 10-Feb-2018]. [cited at p. 19]

[59] B. Franks, "Web Data: The Original Big Data", in Taming the Big Data Tidal Wave, Wiley-Blackwell, pp. 2951, 03-10-2015. [cited at p. 19]

[60] A. De Mauro, M. Greco, and M. Grimaldi, "A formal definition of Big Data based on its essential features", Library Review, vol. 65, no. 3, pp. 122-135, Apr. 2016. [cited at p. 19]

[61] D. Laney, "3D Data Management: Controlling Data Volume, Velocity, and Variety", META Group, Feb. 2001. [Online]. Available: http://blogs.gartner.com/doug-laney/files/2012/01/ad949-3D-Data-Management-Controlling-Data-Volume-Velocity-and-Variety.pdf. [Accessed: 25-Nov-2017]. [cited at p. 19]

[62] M. Schroeck, R. Shockley, J. Smart, D. Romero-Morales, and P. Tufano, "Analytics: The Real-World Use of Big Data", IBM Institute for Business Value, Said Business School, New York, NY, Oct. 2012. [cited at p. 19]

[63] J.-P. Dijcks, "Oracle: Big Data for the Enterprise", An Oracle White Paper, Oracle Corporation, Redwood Shores, CA. Jun. 2013. [cited at p. 19]

[64] D. Kalra, I. Buchan, and N. Paton, "Three Gurus of Big Data", Nov-2016. [Online]. Available: https://thetranslationalscientist.com/issues/0816/three-gurus-of-big-data/. [Accessed: 10-Feb-2018]. [cited at p. 20]

[65] ATLAS Experiment, CERN, "ATLAS Fact Sheet: To raise awareness of the ATLAS detector and collaboration on the LHC", 2010. [cited at p. 20]

[66] IBM, "Big Data Analytics — IBM Analytics". [Online]. Available: https://www.ibm.com/analytics/hadoop/big-data-analytics. [Accessed: 06-Feb-2018]. [cited at p. 20]

[67] J. Bosman, "After 244 Years, Encyclopædia Britannica Stops the Presses", Media Decoder Blog, 1331675662. [Online]. Available: //mediadecoder.blogs.nytimes.com/2012/03/13/after-244-years-encyclopaedia-britannica-stops-the-presses/. [Accessed: 06-Feb-2018]. [cited at p. 20]

[68] J. Madhavan, A. Y. Halevy, S. Cohen, X. L. Dong, S. R. Jeffery, D. Ko, and C. Yu, "Structured data meets the Web: a few observations.", IEEE Data Eng. Bull., vol. 29, no. 4, pp. 19-26, 2006. [cited at p. 21]

[69] S. Ransbotham, D. Kiron, and P. K. Prentice, "Minding the analytics gap", MIT Sloan Management Review, vol. 56, no. 3, p. 63, 2015. [cited at p. 21]

[70] S. Klous and N. Wielaard, "We are Big Data: The Future of the Information Society". Atlantis Press, 2016. [cited at p. 22]

[71] P. Jain, M. Gyanchandani, and N. Khare, "Big data privacy: a technological perspective and review", Journal of Big Data, vol. 3, p. 25, Nov. 2016. [cited at p. 22]

[72] P. F. Drucker, "The Landmarks of Tomorrow", 1st edition. New York, Harper and Row, 1959. [cited at p. 22]

[73] A. McAfee and E. Brynjolfsson, "Big Data: The Management Revolution", Harvard Business Review, Oct. 2012. [Online]. Available: https://hbr.org/2012/10/big-data-the-management-revolution. [Accessed: 25-Aug-2016]. [cited at p. 22]

[74] T. H. Davenport, "Big data. The management revolution.", Harvard Bus Rev.Harvard Bus Rev, 90(10), pp. 61-67, Oct. 2012. [cited at p. 22]

[75] W. E. Deming, "The new economics: for industry, government, education". Cambridge, Mass.: MIT Press, 2000. [cited at p. 22]

[76] IBM, "Taming Big Data: Small Data vs. Big Data — IBM Big Data & Analytics Hub", Aug. 2013. [Online]. Available: http://www.ibmbigdatahub.com/infographic/taming-big-data-small-data-vs-big-data. [Accessed: 25-Aug-2017]. [cited at p. 23]

[77] M. Lindstrom, "Small Data: The Tiny Clues That Uncover Huge Trends", John Murray Learning, 2016. [cited at p. 23]

[78] P. J. W. Windley, "The Live Web: Building Event-Based Connections in the Cloud", Cengage Learning, 2012. [cited at p. 25]

[79] A. Gröflin, M. Weber, M. Guggisberg, and H. Burkhart, "Traffic flow measurement of a public transport system through automated Web observation", in 2017 11th International Conference on Research Challenges in Information Science (RCIS), pp. 156-161, 2017. [cited at p. 25, 39, 40, 67, 127, 143, 171]

[80] "What is a Mashup? - Definition from Techopedia", Techopedia.com. [Online]. Available: https://www.techopedia.com/definition/5373/mashup. [Accessed: 20-Mar-2018]. [cited at p. 25]

[81] P. Rademacher, "HousingMaps". [Online]. Available: http://www.housingmaps.com/. [Accessed: 29-May-2018]. [cited at p. 25]

[82] W. Roush, "Killer Maps – Google, Microsoft, and Yahoo are racing to transform online maps into full-blown browsers, organizing information – and, of course, ads – according to geography. The likely winner? You.", MIT Technology Review, 01-Oct-2005. [Online]. Available: https://www.technologyreview.com/s/404705/killer-maps/. [Accessed: 30-May-2018]. [cited at p. 25]

[83] Google, Inc., "FAQ — Google Maps Platform", Google Developers. [Online]. Available: https://developers.google.com/maps/faq. [Accessed: 29-May-2018]. [cited at p. 25]

[84] Twitter, Inc., "Pricing". [Online]. Available: https://developer.twitter.com/en/pricing.html. [Accessed: 29-May-2018]. [cited at p. 25]

[85] Facebook, Inc., "Platform Policy", Facebook for Developers. [Online]. Available: https://developers.facebook.com/policy/. [Accessed: 29-May-2018]. [cited at p. 25]

[86] S. Rizzotti, "Syndicate – Individual Service Composition in the Web-Age". Jan. 2008. [cited at p. 26]

[87] Z. Akbar, J. M. Garca, I. Toma, and D. Fensel, "On Using Semantically-Aware Rules for Efficient Online Communication", in Rules on the Web, From Theory to Applications, vol. 8620, Eds. Cham: Springer International Publishing, pp. 37-51, 2014. [cited at p. 26]

[88] R. M. Dijkman, M. Dumas, and C. Ouyang, "Semantics and Analysis of Business Process Models in BPMN", Inf. Softw. Technol., vol. 50, no. 12, pp. 1281-1294, Nov. 2008. [cited at p. 26]

[89] P. Wohed, W. M. P. van der Aalst, M. Dumas, and A. H. M. ter Hofstede, "Analysis of Web Services Composition Languages: The Case of BPEL4WS", in Conceptual Modeling - ER 2003, I.-Y. Song, S. W. Liddle, T.-W. Ling, and P. Scheuermann, Eds. Springer Berlin Heidelberg, pp. 200-215, 2003. [cited at p. 26]

[90] M. Blackstock and R. Lea, "Toward a Distributed Data Flow Platform for the Web of Things (Distributed Node-RED)", in Proceedings of the 5th International Workshop on Web of Things, New York, NY, USA, pp. 34-39, 2014. [cited at p. 26, 27]

[91] A. Gröflin, D. Bosch, M. Guggisberg, and H. Burkhart, "Facilitating the Reactive Web – A Condition Action System using Node.js", presented at the 11th International Conference on Web Information Systems and Technologies, pp. 89-95, 2015. [cited at p. 26, 27, 28, 34, 83, 84, 85, 86, 171, 172]

[92] "Yahoo Pipes Blog - Pipes End-of-life Announcement", 04-Jun-2015. [Online]. Available: https://web.archive.org/web/20150604181928/http://pipes.yqlblog.net/post/120705592639/pipes-end-of-life-announcement. [Accessed: 20-Mar-2018]. [cited at p. 27]

[93] S. Ovadia, "Automate the Internet With 'If This Then That' (IFTTT)", Behavioral & Social Sciences Librarian, vol. 33, no. 4, pp. 208-211, Oct. 2014. [cited at p. 27]

[94] A. Paschke, "Reaction RuleML 1.0 for Rules, Events and Actions in Semantic Complex Event Processing", in Rules on the Web. From Theory to Applications, pp. 1-21, 2014. [cited at p. 27]

[95] S. Hausmann and F. Bry, "Towards Complex Actions for Complex Event Processing", in Proceedings of the 7th ACM International Conference on Distributed Event-based Systems, New York, NY, USA, pp. 135-146, 2013. [cited at p. 27]

[96] A. Paschke, H. Boley, Z. Zhao, K. Teymourian, T. Athan, "Reaction RuleML 1.0: Standardized Semantic Reaction Rules". in A. Bikakis, A. Giurca (eds) Rules on the Web: Research and Applications. RuleML. Lecture Notes in Computer Science, vol 7438. Springer, Berlin, Heidelberg, 2012. [cited at p. 27]

[97] S. Hasan, S. O'Riain, and E. Curry, "Approximate Semantic Matching of Heterogeneous Events", in Proceedings of the 6th ACM International Conference on Distributed Event-Based Systems, New York, NY, USA, pp. 252-263, 2012. [cited at p. 28]

[98] P. T. Eugster, P. A. Felber, R. Guerraoui, and A.-M. Kermarrec, "The Many Faces of Publish/Subscribe", ACM Comput. Surv., vol. 35, no. 2, pp. 114-131, Jun. 2003. [cited at p. 28, 30, 31]

[99] "huginn: Create agents that monitor and act on your behalf. Your agents are standing by!", 20-Mar-2018. [Online]. Available: https://github.com/huginn/huginn. [Accessed: 20-Mar-2018]. [cited at p. 28]

[100] T. Tiropanis, W. Hall, N. Shadbolt, D. De Roure, N. Contractor, and J. Hendler, "The web science observatory", IEEE Intelligent Systems, vol. 28, no. 2, pp. 100-104, 2013. [cited at p. 28]

[101] R. Tinati, X. Wang, T. Tiropanis, and W. Hall, "Building a Real-Time Web Observatory", IEEE Internet Computing, vol. 19, no. 6, pp. 36-45, Nov. 2015. [cited at p. 28]

[102] A. Gröflin, "Web Observatory University of Basel", GitHub repository, 17-May-2018. [Online]. Available: https://github.com/WebObservatoryUnibas/lab. [Accessed: 01-Jun-2018]. [cited at p. 28, 121, 143]

[103] S. Price, W. Hall, G. Earl, T. Tiropanis, R. Tinati, X. Wang, E. Gandolfi, J. Gatewood, R. Boateng, D. Denemark, A. Groflin, B. Loader, M. Schmidt, M. Billings, G. Spanakis, H. Suleman, K. Tsoi, B. Wessels, J. Xu, and M. Birkin, "Worldwide Universities Network (WUN) Web Observatory: Applying Lessons from the Web to Transform the Research Data Ecosystem", in

Proceedings of the 26th International Conference on World Wide Web Companion, Republic and Canton of Geneva, Switzerland, pp. 1665-1667, 2017. [cited at p. 29]

[104] Open Knowledge International, "Open Definition 2.1". [Online]. Available: http://opendefinition.org/od/2.1/en/. [Accessed: 10-Jan-2017]. [cited at p. 29]

[105] A. Doan, R. Ramakrishnan, and A. Y. Halevy, "Crowdsourcing systems on the World-Wide Web", Communications of the ACM, vol. 54, no. 4, p. 86, Apr. 2011. [cited at p. 29]

[106] T. O'Reilly, "What Is Web 2.0? Design Patterns and Business Models for the Next Generation of Software", Sep. 2005. [Online]. Available: http://www.oreilly.com/pub/a/web2/archive/what-is-web-20.html. [Accessed: 25-Aug-2016]. [cited at p. 29]

[107] H. Chen, R. H. Chiang, and V. C. Storey, "Business Intelligence and Analytics: From Big Data to Big Impact.", MIS quarterly, vol. 36, no. 4, pp. 1165-1188, 2012. [cited at p. 29, 39]

[108] A. C. M. Fong, S. C. Hui, and H. L. Vu, "Effective techniques for automatic extraction of Web publications", Online Information Review, vol. 26, no. 1, pp. 4-18, Feb. 2002. [cited at p. 30]

[109] M. S. Weber, "Observing the Web by Understanding the Past: Archival Internet Research", in Proceedings of the 23rd International Conference on World Wide Web, New York, NY, USA, pp. 1031-1036, 2014. [cited at p. 30]

[110] PubSubHubbub, "The PubSubHubbub protocol specification", 29-Mar-2018. [Online]. Available: https://github.com/pubsubhubbub/PubSubHubbub. [Accessed: 03-Apr-2018]. [cited at p. 30]

[111] J. Genestoux, B. Fitzpatrick, B. Slatkin, and M. Atkins, "WebSub", 19-Dec-2017. [Online]. Available: https://www.w3.org/TR/websub/. [Accessed: 03-Apr-2018]. [cited at p. 30]

[112] M. Thomson, E. Damaggio, and B. Raymor, "Generic Event Delivery Using HTTP Push", draft-ietf-webpush-protocol-12, Internet Engineering Task Force (IETF), 25-Apr-2017. [Online]. Available: https://tools.ietf.org/html/draft-ietf-webpush-protocol-12. [Accessed: 06-Apr-2018]. [cited at p. 31]

[113] V. Trifa, D. Guinard, V. Davidovski, A. Kamilaris, and I. Delchev, "Web Messaging for Open and Scalable Distributed Sensing Applications", in Web Engineering, pp. 129-143, 2010. [cited at p. 31]

[114] S. Duquennoy, G. Grimaud, and J. J. Vandewalle, "Consistency and scalability in event notification for embedded Web applications", in 2009 11th IEEE International Symposium on Web Systems Evolution, pp. 89-98, 2009. [cited at p. 31]

[115] D. Benslimane, S. Dustdar, and A. Sheth, "Services Mashups: The New Generation of Web Applications", IEEE Internet Computing, vol. 12, no. 5, pp. 13-15, Sep. 2008. [cited at p. 31]

[116] V. Pimentel and B. G. Nickerson, "Communicating and Displaying Real-Time Data with WebSocket", IEEE Internet Computing, vol. 16, no. 4, pp. 45-53, Jul. 2012. [cited at p. 32]

[117] S. Hogg, "Software Containers: Used More Frequently than Most Realize", Network World, 26-May-2014. [Online]. Available: https://www.networkworld.com/article/2226996/cisco-subnet/software-containers–used-more-frequently-than-most-realize.html. [Accessed: 10-Jul-2018]. [cited at p. 35]

[118] Docker Inc., "Why Docker?", 05-Mar-2018. [Online]. Available: https://www.docker.com/enterprise-edition. [Accessed: 08-May-2018]. [cited at p. 35, 95]

[119] A. Bifet, J. Zhang, W. Fan, C. He, J. Zhang, J. Qian, G. Holmes, and B. Pfahringer, "Extremely Fast Decision Tree Mining for Evolving Data Streams", in Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, New York, NY, USA, pp. 1733-1742, 2017. [cited at p. 35]

[120] U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth, "From Data Mining to Knowledge Discovery in Databases", American Association for Artificial Intelligence, AI MAGAZINE, p. 18, Jul. 1997. [cited at p. 36]

[121] J. W. Tukey, "Exploratory Data Analysis", Addison-Wesley Publishing Company, 1977. [cited at p. 36]

[122] J. T. Behrens, "Principles and Procedures of Exploratory Data Analysis", Psychological Methods, vol. 2, no. 2, pp. 131-160, 1997. [cited at p. 37]

[123] O. Marban, G. Mariscal, and J. Segovia, "A Data Mining & Knowledge Discovery Process Model", in Data Mining and Knowledge Discovery in Real Life Applications, InTech, Rijek, Crotia, pp. 1-16, Jan. 2009. [cited at p. 37]

[124] C. Shearer, "The CRISP-DM Model: The New Blueprint for Data Mining", Journal of Data Warehousing, vol. 5, no. 4, pp. 13-22, 2000. [cited at p. 37]

[125] B. Liu, "Web Data Mining: Exploring Hyperlinks, Contents, and Usage Data", Second Edition, Berlin, Heidelberg: Springer, 2011. [cited at p. 37]

[126] B. A. Galitsky, G. Dobrocsi, J. L. de la Rosa, and S. O. Kuznetsov, "Using Generalization of Syntactic Parse Trees for Taxonomy Capture on the Web", in Conceptual Structures for Discovering Knowledge, pp. 104-117, 2011. [cited at p. 38]

[127] L. V. Ahn, M. Blum, N. J. Hopper, and J. Langford, "CAPTCHA: Using Hard AI Problems for Security", in Proceedings of the 22nd International Conference on Theory and Applications of Cryptographic Techniques, Berlin, Heidelberg, 2003, pp. 294-311. [cited at p. 38]

[128] E. Ferrara, P. De Meo, G. Fiumara, and R. Baumgartner, "Web data extraction, applications and techniques: A survey", Knowledge-Based Systems, vol. 70, pp. 301-323, Nov. 2014. [cited at p. 38, 39]

[129] M. Beyer, "Gartner Says Solving 'Big Data' Challenge Involves More Than Just Managing Volumes of Data" Gartner, 2011. [Online]. Available: http://www.gartner.com/newsroom/id/1731916. [Accessed: 25-Aug-2016]. [cited at p. 39]

[130] D. Parker, "JavaScript with Promises: Managing Asynchronous Code", Second Release, O'Reilly Media, Inc., 17-July-2015. [cited at p. 42, 43]

[131] "needle", npm. [Online]. Available: https://www.npmjs.com/package/needle. [Accessed: 15-Apr-2018]. [cited at p. 43, 184]

[132] "cheerio: Fast, flexible, and lean implementation of core jQuery designed specifically for the server", 03-Sep-2018. [Online]. Available: https://github.com/cheeriojs/cheerio. [Accessed: 15-Apr-2018]. [cited at p. 43, 184]

[133] L. Merancia, "node-cron: Cron for NodeJS", node-cron - npm, 04-May-2018. [Online]. Available: https://github.com/kelektiv/node-cron. [Accessed: 05-May-2018]. [cited at p. 45]

[134] Leaders Section, The data economy demands a new approach to antitrust rules: "The world's most valuable resource is no longer oil, but data", The Economist. [Online]. Available: http://www.economist.com/news/leaders/21721656-data-economy-demands-new-approach-antitrust-rules-worlds-most-valuable-resource. [Accessed: 15-Sep-2017]. [cited at p. 53]

[135] "Maximillian Schrems v Data Protection Commissioner", C362/14, EUR-Lex, 6-Oct-2015. [Online]. Available: https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX%3A62014CJ0362. [Accessed: 15-Apr-2018]. [cited at p. 53]

[136] M. A. Weiss and K. Archick, "U.S.-EU Data Privacy: From Safe Harbor to Privacy Shield", Congressional Research Service, 19-May-2016. [cited at p. 54]

[137] The Federal Council, "CC 235.1 Federal Act of 19 June 1992 on Data Protection (FADP)", 01-Jan-2014. [Online]. Available: https://www.admin.ch/opc/en/classified-compilation/19920153/index.html. [Accessed: 30-Apr-2018]. [cited at p. 54]

[138] The Federal Assembly, "CC 312.0 Swiss Criminal Procedure Code of 5 October 2007 (Criminal Procedure Code, CPC)". [Online]. Available: https://www.admin.ch/opc/en/classified-compilation/20052319/index.html. [Accessed: 30-Apr-2018]. [cited at p. 54]

[139] The Federal Council, "SR 780.1 Bundesgesetz vom 18. März 2016 betreffend die Überwachung des Post- und Fernmeldeverkehrs (BÜPF)". [Online]. Available: https://www.admin.ch/opc/de/classified-compilation/20122728/index.html. [Accessed: 30-Apr-2018]. [cited at p. 55]

[140] "Die neuen Auskunfts- und Überwachungstypen", 10-Apr-2018. [Online]. Available: https://www.li.admin.ch/sites/default/files/2018-04/Die%20neuen%20Auskunfts-%20und%20%C3%9Cberwachungstypen.pdf. [Accessed: 30-Apr-2018]. [cited at p. 55]

[141] Coucil of Europe, "Convention on Cybercrime", Details of Treaty No.185, 23-Nov-2001. [Online]. Available: https://www.coe.int/en/web/conventions/full-list. [Accessed: 30-Apr-2018]. [cited at p. 55]

[142] The Federal Council, "SR 780.11 Verordnung vom 15. November 2017 über die Überwachung des Post- und Fernmeldeverkehrs (VÜPF)", 02-

May-2018. [Online]. Available: https://www.admin.ch/opc/de/classified-compilation/20172173/index.html. [Accessed: 04-May-2018]. [cited at p. 55]

[143] communications-eu@turnitin.com, The Turnitin Team, "Important updates to our Privacy Centre and Terms of Service", Turnitin UK, 7.31 PM, 02-May-2018. [cited at p. 57]

[144] generationeasyJet@email.easyjet.com, easyJet, "Your booking *******: Alexander, you're always in control - on your upcoming trip to [.] and beyond", easyJet Airline Company Limited, 7.49 PM, 03-May-2018. [cited at p. 57]

[145] European Union, "EUR-Lex - 32016R0679 - EN - EUR-Lex", 27-Apr-2016. [Online]. Available: https://eur-lex.europa.eu/eli/reg/2016/679/oj. [Accessed: 29-Apr-2018]. [cited at p. 57]

[146] European Commission, "EU GDPR Information Portal", EU GDPR Portal. [Online]. Available: http://eugdpr.org/eugdpr.org.html. [Accessed: 29-Apr-2018]. [cited at p. 57]

[147] P. Pfirter, "Modern communication in the investigatory process: What kind of data can the government collect without violation of procedural and privacy laws?", Accountability of Criminal Justice Systems, Pre-Conference on the 10th Anniversary of Future of Adversarial and Inquisitorial Systems 2018, Faculty of Law, University of Basel, Switzerland, 25-April-2018. [cited at p. 58, 69]

[148] W. Malcolm, "Getting ready for Europe's new data protection rules", Google, 08-Aug-2017. [Online]. Available: https://www.blog.google/topics/google-europe/gdpr-europe-data-protection-rules/. [Accessed: 29-Apr-2018]. [cited at p. 58]

[149] C. Coleman, "Are you ready for a data privacy shake-up?", BBC News, 20-Apr-2018. [Online]. Available: http://www.bbc.com/news/technology-43657546. [Accessed: 29-Apr-2018]. [cited at p. 58]

[150] O. Solon, "How Europe's 'breakthrough' privacy law takes on Facebook and Google", The Guardian, 19-Apr-2018. [Online]. Available: http://www.theguardian.com/technology/2018/apr/19/gdpr-facebook-google-amazon-data-privacy-regulation. [Accessed: 29-Apr-2018]. [cited at p. 58]

[151] J. E. Webber, D. Stephens, "Google vs the right to be forgotten: Chips with Everything podcast", The Guardian, 20-Apr-2018. [Online]. Available: http://www.theguardian.com/technology/audio/2018/apr/20/google-vs-the-right-to-be-forgotten-chips-with-everything-podcast. [Accessed: 29-Apr-2018]. [cited at p. 58]

[152] noyb, "GDPR: noyb.eu filed four complaints over 'forced consent' against Google, Instagram, WhatsApp and Facebook", 25-May-2018. [Online]. Available: https://noyb.eu/wp-content/uploads/2018/05/pa_forcedconsent_en.pdf. [Accessed: 27-May-2018]. [cited at p. 59]

[153] C. Foxx, "Google and Facebook face GDPR complaints", BBC News, 25-May-2018. [Online]. Available: http://www.bbc.com/news/technology-44252327. [Accessed: 27-May-2018]. [cited at p. 59]

[154] Bundesamt für Justiz, "Stärkung des Datenschutzes", 15-Sep-2017. [Online]. Available: https://www.bj.admin.ch/bj/de/home/staat/gesetzgebung/datenschutzstaerkung.html. [Accessed: 29-Apr-2018]. [cited at p. 59]

[155] The Federal Assembly - The Swiss Parliament, "17.059 — Datenschutzgesetz. Totalrevision und Änderung weiterer Erlasse zum Datenschutz — Geschäft — Das Schweizer Parlament", 15-Sep-2017. [Online]. Available: https://www.parlament.ch/de/ratsbetrieb/suche-curia-vista/geschaeft?AffairId=20170059. [Accessed: 29-Apr-2018]. [cited at p. 59]

[156] G. V. Müller, "Schweizer Firmen sind gezwungen, Daten ihrer Kunden besser zu schützen – sonst drohen drakonische Strafen — NZZ", Neue Zürcher Zeitung, 14-Jul-2017. [Online]. Available: https://www.nzz.ch/wirtschaft/in-knapp-einem-jahr-gilt-es-ernst-die-schweiz-wird-eu-datenschutz-konform-ld.1305383. [Accessed: 29-Apr-2018]. [cited at p. 60]

[157] European Union, "Directive (EU) 2016/680 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data by competent authorities for the purposes of the prevention, investigation, detection or prosecution of criminal offences or the execution of criminal penalties, and on the free movement of such data, and repealing Council Framework Decision", 2008/977/JHA, vol. OJ L. 2016. [cited at p. 60]

[158] European Union, "EUR-Lex - 32016L0681 - EN - EUR-Lex". [Online]. Available: https://eur-lex.europa.eu/eli/dir/2016/681/oj. [Accessed: 29-Apr-2018]. [cited at p. 61]

[159] C. Di Francesco Maesa, "Eurojus ≫ Balance between Security and Fundamental Rights Protection: An Analysis of the Directive 2016/680 for data protection in the police and justice sectors and the Directive 2016/681 on the use of passenger name record (PNR)", Giurisprudenza di Diritto dell'Unione Europea – Casi Scelti – seconda edizione, 24-May-2016. [Online]. Available: http://rivista.eurojus.it/balance-between-security-and-fundamental-rights-protection-an-analysis-of-the-directive-2016680-for-data-protection-in-the-police-and-justice-sectors-and-the-directive-2016681-on-the-use-of-passen/. [Accessed: 29-Apr-2018]. [cited at p. 61]

[160] "Carpenter v. United States", SCOTUSblog. [Online]. Available: http://www.scotusblog.com/case-files/cases/carpenter-v-united-states-2/. [Accessed: 21-Apr-2018]. [cited at p. 62]

[161] A. D. Sorkin, "In Carpenter Case, Justice Sotomayor Tries to Picture the Smartphone Future", The New Yorker, 30-Nov-2017. [Online]. Available: https://www.newyorker.com/news/our-columnists/carpenter-justice-sotomayor-tries-to-picture-smartphone-future. [Accessed: 21-Apr-2018]. [cited at p. 63]

[162] Harvard Law Review, "Microsoft Corp. v. United States", 09-Dec-2016. [Online]. Available: https://harvardlawreview.org/2016/12/microsoft-corp-v-united-states/. [Accessed: 22-Apr-2018]. [cited at p. 64]

[163] "17-2 United States v. Microsoft Corp.", Per Curiam, p. 3, 17-Apr-2018. [cited at p. 65]

[164] B. Smith, "Microsoft statement on the inclusion of the CLOUD Act in the Omnibus funding bill", Microsoft on the Issues, 21-Mar-2018. [Online]. Available: https://blogs.microsoft.com/on-the-issues/2018/03/21/microsoft-statement-on-the-inclusion-of-the-cloud-act-in-the-omnibus-funding-bill/. [Accessed: 22-Apr-2018]. [cited at p. 65]

[165] Sir I. Macleod and D. I. Baker, "Brief amici curiae of Government of the United Kingdom of Great Britain and Northern Ireland in support of neither party filed.", 13-Dec-2017. [cited at p. 66]

[166] L. Bell, "Theresa May supports Trump's CLOUD Act to extend US power over overseas data — V3", http://www.v3.co.uk, 07-Feb-2018. [Online]. Available: https://www.v3.co.uk/v3-uk/news/3026215/theresa-may-supports-trumps-cloud-act-to-extend-us-power-over-overseas-data. [Accessed: 22-Apr-2018]. [cited at p. 66]

[167] "Ireland's Brexit Challenge, Global Business - BBC World Service", BBC, 14-Apr-2018. [Online]. Available: http://www.bbc.co.uk/programmes/w3cswjxk. [Accessed: 15-Apr-2018]. [cited at p. 66]

[168] M. Bovens and S. Zouridis, "From Street-Level to System-Level Bureaucracies: How Information and Communication Technology is Transforming Administrative Discretion and Constitutional Control", Public Administration Review, vol. 62, no. 2, pp. 174-184, Jan. 2002. [cited at p. 67]

[169] J. C. Bertot, P. T. Jaeger, and J. M. Grimes, "Using ICTs to create a culture of transparency: E-government and social media as openness and anti-corruption tools for societies", Government Information Quarterly, vol. 27, no. 3, pp. 264-271, Jul. 2010. [cited at p. 67]

[170] Raconteur, "Open source: sharing patents to speed up innovation", Business / Intellectual Property 2018, 20-Apr-2018. [Online]. Available: https://www.raconteur.net/business/open-source-sharing-patents-speed-innovation. [Accessed: 04-May-2018]. [cited at p. 67]

[171] J. D. Goodman and A. Baker, "William Bratton, New York's Influential Police Commissioner, Is Stepping Down", The New York Times, 02-Aug-2016. [Online]. Available: https://www.nytimes.com/2016/08/03/nyregion/bill-bratton-nypd-commissioner.html. [Accessed: 30-Apr-2018]. [cited at p. 68]

[172] J. H. Burch, K. Rose, and W. Bratton, "Perspectives in Law Enforcement – The Concept of Predictive Policing: An Interview With Chief William Bratton", First Predictive Policing Symposium, National Institute of Justice, Los Angeles, California, USA, 20-Nov-2009. [cited at p. 68]

[173] W. Perry, B. McInnis, C. Price, S. Smith, and J. Hollywood, "Predictive Policing: The Role of Crime Forecasting in Law Enforcement Operations", RAND Corporation, 2013. [cited at p. 68]

[174] A. G. Ferguson, "The Police Are Using Computer Algorithms to Tell if You're a Threat", Time, 03-Oct-2017. [Online]. Available: http://time.com/4966125/police-departments-algorithms-chicago/. [Accessed: 30-Apr-2018]. [cited at p. 69]

[175] A. G. Ferguson, "The Rise of Big Data Policing: Surveillance, Race, and the Future of Law Enforcement", NYU Press, 3-Oct-2017. [cited at p. 69]

[176] Z. Friend, "Predictive Policing: Using Technology to Reduce Crime", FBI: Law Enforcement Bulletin, 09-Apr-2013. [Online]. Available: https://leb.fbi.gov/articles/featured-articles/predictive-policing-using-technology-to-reduce-crime. [Accessed: 04-May-2018]. [cited at p. 69]

[177] D. Gerstner and H. Straub, "Predictive Policing", Evaluation of the Pilot Project P4 in Baden-Württemberg/Germany, Max Planck Institute for Foreign and International Criminal Law, Freiburg i. Br., Germany, 03-Jan-2018. [Online]. Available: https://www.mpicc.de/en/forschung/forschungsarbeit/kriminologie/predictive_policing_p4.html. [Accessed: 04-May-2018]. [cited at p. 69]

[178] L. Dearden, "How big data can now be used to predict where crime will happen", The Independent, London, United Kingdom, 23-Sep-2017. [Online]. Available: http://www.independent.co.uk/news/uk/home-news/police-big-data-technology-predict-crime-hotspot-mapping-rusi-report-research-minority-report-a7963706.html. [Accessed: 04-May-2018]. [cited at p. 69]

[179] H. Chloe, "Can Computers Predict Crimes That Haven't Happened Yet?", The Inquiry – BBC World Service, BBC, 31-May-2018. [Online]. Available: https://www.bbc.co.uk/programmes/w3cswqt9. [Accessed: 01-Jun-2018]. [cited at p. 69]

[180] C. Hatton, "China sets up huge social 'credit system'", BBC News, 26-Oct-2015. [Online]. Available: https://www.bbc.com/news/world-asia-china-34592186. [Accessed: 12-Sep-2018]. [cited at p. 69]

[181] S. Gless, "Closing Remarks", Conference on the 10th Anniversary of Future of Adversarial and Inquisitorial Systems 2018, Faculty of Law, University of Basel, Switzerland, 27-April-2018. [cited at p. 70]

[182] J. Kepler, T. Brahe, emperor of G. Rudolph II, J. S. Bute, and donor D. Burndy Library, "Astronomia nova aitiologetos" [romanized]: sev phys-

ica coelestis, tradita commentariis de motibvs stellæ Martis, ex observationibus G. V. Tychonis Brahe; jussu & sumptibus Rvdolphi II. [Heidelberg: G. Voegelinus], 1609. [cited at p. 73]

[183] M. Fowler, "Design – Who needs an architect?", IEEE Software, vol. 20, no. 5, pp. 11-13, Sep. 2003. [cited at p. 74]

[184] M. Weber, "Component Based Web-Scraping Strategies", Master Thesis, University of Basel, 7-June-2017. [cited at p. 77, 87, 172]

[185] J. Martin, "Managing the Data Base Environment", Pearson Education, Limited, 1983. [cited at p. 79]

[186] M. Schwartz, "Web application framework – DocForge", 23-Jul-2015. [Online]. Available: https://web.archive.org/web/20150723163302/http://docforge.com/wiki/Web_application_framework. [Accessed: 05-May-2018]. [cited at p. 82]

[187] D. Bosch, "Towards Reactive Information Systems and their Services", Master Thesis, University of Basel, 20-June-2014. [cited at p. 82]

[188] Mozilla, "Developer Tools & Resources — Mozilla", Developer tools, resources, videos and more. [Online]. Available: https://www.mozilla.org/en-US/developer/. [Accessed: 08-May-2018]. [cited at p. 93]

[189] Catch a Car, "About". [Online]. Available: https://www.catch-a-car.ch/en/about/. [Accessed: 13-May-2018]. [cited at p. 111]

[190] C. P. G. Frei, "Track a Car: Untersuchung von Methoden zur Gewinnung von Echtzeitdaten am Beispiel Catch a Car", Bachelor Thesis, University of Basel, 02-Mai-2016. [cited at p. 111]

[191] Catch a Car, "Basel", Map screenshot. [Online]. Available: https://www.catch-a-car.ch/en/basel/. [Accessed: 13-May-2018]. [cited at p. 112, 173]

[192] Volkswagen, "VW up! — Volkswagen Deutschland", 2018. [Online]. Available: https://www.volkswagen.de/content/vw_pkw/importers/de/de/models/up.html. [Accessed: 13-May-2018]. [cited at p. 113]

[193] Google Inc., "One Car", Screenshot, Google My Maps. [Online]. Available: https://www.google.com/maps/d/edit?mid=1HfsjHZBN0hn4OTY12lO3_Gv9VV4. [Accessed: 15-May-2018]. [cited at p. 114, 173]

[194] M. Regenass, "Die Kunden wünschen Elektroautos – Ab Herbst bietet das Carsharing-Unternehmen Catch a Car zusätzliche 30 Stromer an", Basler Zeitung, Basel, p. 19, 02-May-2018. [cited at p. 119]

[195] "511 SF BAY – Home". [Online]. Available: http://511.org/. [Accessed: 01-June-2018]. [cited at p. 122]

[196] "Transport for London — Keeping London moving". [Online]. Available: https://www.tfl.gov.uk/. [Accessed: 01-June-2018]. [cited at p. 122]

[197] Basler Verkehrs-Betriebe (BVB), "All about us – Basler Verkehrs-Betriebe" [Online]. Available: https://www.bvb.ch/en/unternehmen/portraet. [Accessed: 25-May-2018]. [cited at p. 122]

[198] S. Eichkorn, "Verspätungen bei den BVB – Jedes zweite Tram hat am Barfüsserplatz Verspätung", Eine neue Untersuchung der Universität Basel zeigt, dass Fahrgäste in der Basler Innenstadt viele Verspätungen hinnehmen müssen, Schweizer Radio und Fernsehen (SRF), 26-Jul-2017. [Online]. Available: https://www.srf.ch/news/regional/basel-baselland/jedes-zweite-tram-hat-am-barfuesserplatz-verspaetung. [Accessed: 01-Jun-2018]. [cited at p. 124]

[199] Alexa Internet, Inc., "Top Sites in Switzerland – Alexa". [Online]. Available: https://www.alexa.com/topsites/countries/CH. [Accessed: 17-May-2018]. [cited at p. 127]

[200] J. Simonet, "Leserkommentare des Newsportals 20 Minuten – Analyse von Webinhalten mithilfe des Condition Action Tools WebAPI ECA-Engine", Bachelor Thesis, University of Basel, 2015. [cited at p. 128, 130, 131, 174]

[201] M. A. Russell, "Mining the Social Web: Data Mining Facebook, Twitter, LinkedIn, Google+, GitHub, and More", OReilly Media, Inc., 2013. [cited at p. 135]

[202] S. Schrittwieser, P. Fruhwirt, P. Kieseberg, M. Leithner, M. Mulazzani, M. Huber, E. Weippl, "Guess Who's Texting You? Evaluating the Security of Smartphone Messaging Applications", 2012. [cited at p. 137]

[203] Y. Cheng, L. Ying, S. Jiao, P. Su, and D. Feng, "Bind Your Phone Number with Caution: Automated User Profiling Through Address Book Matching on Smartphone", in Proceedings of the 8th ACM SIGSAC Symposium on Information, Computer and Communications Security, New York, NY, USA, pp. 335340, 2013. [cited at p. 137]

[204] Y. Shafranovich, "Common Format and MIME Type for Comma-Separated Values (CSV) Files", RFC 4180, Internet Engineering Task Force (IETF), Oct. 2005. [Online]. Available: https://tools.ietf.org/html/rfc4180. [Accessed: 26-Jan-2018]. [cited at p. 176]

[205] S. Faulkner, A. Eicholz, T. Leithead, A. Danilo, and S. Moon, "HTML 5.2", 14-Dec-2017. [Online]. Available: https://www.w3.org/TR/html/. [Accessed: 02-Feb-2018]. [cited at p. 176]

[206] T. Bray, "The JavaScript Object Notation (JSON) Data Interchange Format", RFC 7159, Internet Engineering Task Force (IETF), Mar. 2015. [Online]. Available: https://tools.ietf.org/html/rfc7159. [Accessed: 28-Jan-2018]. [cited at p. 177]

[207] Open Definition, "Open Definition 2.1 - Open Definition - Defining Open in Open Data, Open Content and Open Knowledge". [Online]. Available: http://opendefinition.org/od/2.1/en/. [Accessed: 02-Feb-2018]. [cited at p. 177]

[208] N. B. Dale and J. Lewis, "Computer Science Illuminated", Jones & Bartlett Learning, 2007. [cited at p. 177]

[209] Stanford University, "Best practices for file formats", Stanford Libraries. [Online]. Available: http://library.stanford.edu/research/data-management-services/data-best-practices/best-practices-file-formats. [Accessed: 02-Feb-2018]. [cited at p. 178]

[210] O. Lassila and R. R. Swick, "Resource Description Framework (RDF) Model and Syntax Specification", 05-Jan-1999. [Online]. Available: https://www.w3.org/TR/PR-rdf-syntax/. [Accessed: 31-Jan-2018]. [cited at p. 178]

[211] "data.gov.uk". [Online]. Available: https://data.gov.uk/. [Accessed: 31-Jan-2018]. [cited at p. 178]

[212] archiveteam.org, "Scientific Data formats - Just Solve the File Format Problem", 27-Jan-2018. [Online]. Available: http://fileformats.archiveteam.org/wiki/Scientific_Data_formats. [Accessed: 02-Feb-2018]. [cited at p. 178]

[213] T. Reschenhofer and F. Matthes, "An Empirical Study on Spreadsheet Shortcomings from an Information Systems Perspective", in Business Information Systems, 2015, pp. 50-61. [cited at p. 178]

[214] T. Bray, J. Paoli, C. M. Sperberg-McQueen, E. Maler, and F. Yergeau, "Extensible Markup Language (XML) 1.0 (Fifth Edition)", 26-Nov-2008. [Online]. Available: https://www.w3.org/TR/REC-xml/. [Accessed: 31-Jan-2018]. [cited at p. 179]

[215] T. Berners-Lee, R. Fielding, U. C. Irvine, and L. Masinter, "Uniform Resource Identifiers (URI): Generic Syntax", Aug. 1998. [Online]. Available: https://tools.ietf.org/html/rfc2396. [Accessed: 02-Mar-2018]. [cited at p. 179]

[216] T. Berners-Lee, R. T. Fielding, and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax", Jan-2005. [Online]. Available: https://tools.ietf.org/html/rfc3986#section-3.2. [Accessed: 09-Mar-2018]. [cited at p. 180]

[217] S. Lohr, "The Web's Inventor Regrets One Small Thing", Bits Blog, 12-Oct-2009. [Online]. Available: //bits.blogs.nytimes.com/2009/10/12/the-webs-inventor-regrets-one-small-thing/. [Accessed: 09-Mar-2018]. [cited at p. 180]

[218] B. Lavoie and H. F. Nielsen, "Web Characterization Terminology & Definitions Sheet", W3C Working Draft, 24-05-1999. [Online]. Available: https://www.w3.org/1999/05/WCA-terms/#Resource3. [Accessed: 02-Mar-2018]. [cited at p. 180, 181, 182]

[219] I. Jacobs and N. Walsh, "Architecture of the World Wide Web, Volume One", W3C Working Draft, 15-Dec-2004. [Online]. Available: https://www.w3.org/TR/webarch/#id-resources. [Accessed: 06-Mar-2018]. [cited at p. 181]

[220] R. Fielding and J. Reschke, "Hypertext Transfer Protocol (HTTP/1.1): Message Syntax and Routing", June 2014. [Online]. Available: https://tools.ietf.org/html/rfc7230. [Accessed: 28-May-2018]. [cited at p. 182]

[221] M. Arlitt, "Characterizing Web User Sessions", SIGMETRICS Perform. Eval. Rev., vol. 28, no. 2, pp. 50-63, Sep. 2000. [cited at p. 182]

[222] A. Soltani, A. Peterson, and B. Gellman, "NSA uses Google cookies to pinpoint targets for hacking", Washington Post, 10-Dec-2013. Available: https://www.washingtonpost.com/news/the-switch/wp/2013/12/10/nsa-uses-google-cookies-to-pinpoint-targets-for-hacking/. [Accessed: 08-Mar-2018]. [cited at p. 182]

[223] GitHub, "GitHub Octoverse 2017", GitHub Octoverse 2017 — Highlights from the last twelve months. [Online]. Available: https://octoverse.github.com/. [Accessed: 15-Apr-2018]. [cited at p. 183]

[224] Mozilla, "JavaScript", JavaScript — MDN Web Docs, 2016. [Online]. Available: https://developer.mozilla.org/bm/docs/Web/JavaScript. [Accessed: 15-Apr-2018]. [cited at p. 183]

[225] Node.js Foundation, "About — Node.js". [Online]. Available: https://nodejs.org/en/about/. [Accessed: 15-Apr-2018]. [cited at p. 183]

[226] "Welcome to Python.org", Python.org. [Online]. Available: https://www.python.org/about/. [Accessed: 15-Apr-2018]. [cited at p. 184]

[227] "PEP 20 – The Zen of Python — Python.org", Python.org, 19-Aug-2004. [Online]. Available: https://www.python.org/dev/peps/pep-0020/. [Accessed: 15-Apr-2018]. [cited at p. 184]

[228] R. Prediger and R. Winzinger, "Node.js: Professionell hochperformante Software entwickeln", Carl Hanser Verlag GmbH Co KG, 2015. [cited at p. 184]

[229] "pyquery: A jquery-like library for python", 01-Nov-2018. [Online]. Available: https://github.com/gawel/pyquery. [Accessed: 15-Apr-2018]. [cited at p. 184]

[230] "node-cron", npm. [Online]. Available: https://www.npmjs.com/package/node-cron. [Accessed: 15-Apr-2018]. [cited at p. 184]

List of Figures

Appendices

Appendix A

Glossar

File Formats

Comma-Separated Files

The Comma-Separated Value (CSV) file format is widely used for the transfer of large and homogeneous data. It is believed that data saved in the CSV format is futile without comprehensible documentation. That is the reason why it is important to create a documentation of the file contents and data fields. The documentation intends to make understanding and use contained data more comprehensible. According to specifications, the structure of the file must be well respected, if not, reading errors will most likely occur. This is due to the strict syntax processing. [204]

HTML

Vast amount of data is encapsulated in the HTML format on many Web pages. HTML may be used for stable content with limited area of application. The modification of HTML is sometimes difficult to achieve, however, it still is a convenient way to display data or to refer to a Web page that contains data [205].

JSON

JavaScript Object Notation (JSON) is a text-based format for language-independent data interchanges. According to the corresponding "Request for Comments" (RFC) document series, JSON is a lightweight file format based on ECMAScript Programming Language Standard. It uses a small set of formatting rules in or-

der to transfer data arrays and objects [206]. JSON provides two types of compositions:

- **Name and value pairs**
  E.g. object, record, hash table;
- **Ordered list of values**
  E.g. array, list, or sequence. [206]

Furthermore, it is a widely held view that JSON offers humans an easy to read structure while machines are still able to process it.

Open File Formats

Even if data is stored in machine readable means, issues with the chosen file format may still remain. Open file formats or open formats resolve the most common proprietary issues. Open file formats or open formats are defined by the Open Definition 2.1 as follows: "An open format is one which places *no restrictions*, monetary or otherwise, upon its use and can be fully processed with at least one free/libre/open-source software tool." [207]. In contrast to a "closed" file format, specifications for an open file format must be available for free to everybody without limitations on re-use. Logically, closed file formats do not publicly outline its specifications and therefore limit its re-use.

Plain Text

Plain text files (TXT) involve structured sequences of lines for storage of electronic text. There are usually two denominations: either *text files* or *binary files*. A text file stores bytes of data as characters by using the American Standard Code for Information Interchange (ASCII) or for bigger character sets the Unicode. In contrast to the text file, a binary file needs to have specific interpretation of each bit for the data within the file. No meta data can be written into a text document, however, string operations for parsing may be used for interpretations. [208]

Proprietary File Formats

File formats with secret encoding-schemes are considered proprietary. Often designed by a company or organisation, proprietary file formats are only usable with the help of particular software or hardware. Therefore, the licence holder controls the specifications of data encoding and perhaps its patents. These

restrictions apply to hinder third parties to use the format for their purpose. For sustainability reasons, data should be stored in non-proprietary (open) file formats whenever possible. Otherwise data loss may occur when a software manufacturer is not supporting the file format any longer [209].

RDF

The Resource Description Framework (RDF) provides the basis for processing meta data and plays a key role in the Semantic Web. It enables multiple data sources to be combined through representation of data and facilitates the automatic processing of Web resources. It further enhances the interoperability between Web applications [210]. For RDF storage JSON and XML file formats may be used. Open Data initiatives such as the one of the British government are using RDF for interconnecting open data among each other [211].

Scientific File Formats

Scientific Data formats such as Hierarchical Data Format (HDF) are commonly used to store large amounts of data. There are many different file formats available for specific scientific applications. HDF for example is supported by applications e.g. MATLAB, Python and Julia. Depending on the actual science field many different formats can be applied. Many commercial and non-commercial formats are in use for research purposes [212].

Spreadsheet

The spreadsheet is still believed to be the most common form of information medium in both governmental agencies and businesses [213]. Rows, columns, inherent descriptions, and sheets qualify most often for publication because data is available immediately. Perhaps macros and formulas make it more difficult to comprehend undertaken calculations. According to the Open Data Handbook it is recommended to document formulas or macros for better guidance [45].

Text Document

Text document file formats such as DOC, ODF, or PDF are sometimes good enough to store various types of data. Typically stored within text documents is data which is more or less stable, e.g. mailing lists [45]. A major downside of text documents is the lacking support of consistent structure. Depending on what text document software is used for editing, difficulties arise when data

must be automatically processed. Besides, it is a widely held view that DOC format should not be used, if data exists in a different format.

XML

The goal of the Extensible Markup Language (XML) is to design a simple, general and widely supported file format for data exchanges. It enables to keep the structure in the data and at the same time allows to expand the documentation of data. [214]

Technologies

Resource

According to the URI specification, a resource "can be anything that has identity" and thus it is not limited to other resources such as a book which is not accessible trough the Web [215]. The term resource has become a common term for things that have an identity whether they are on the Web or not, e.g. Web pages or human beings. The focus lies on a subset associated with the Web. That turns into Web resources which forms the actual Web content. Uniform Resource Identifier (URI) are most often used for Web addresses where a specific type of URI refers to an Uniform Resource Locator (URL). It is a unique string of characters to identify resources. Uniform Resource Identifier (URI) are used to identify resources of all sorts. [28]

The notion of addressable documents and files has been changed ever since. The understanding is far broader nowadays and Web resources are considered all things that can be obtained through the Internet. In contrast to resources governed by the operating system, Web resources simultaneously constitute Web content.

Web Page or Web Site

A Web page is nothing less than a Web document which can be accessed with a browser from the Internet but also within an Intranet. Web pages usually come with structured or unstructured text in which multimedia elements and images are put in place. Web pages are the main sources of information and are identifiable through one particular URL which are interconnected with hyperlinks.

Authority and Path

A host page identifies a page with a distinct Web address, it may also contain "authority" and "path". The access authority is the clearly defined Web address that can be enlarged with a path or a directory. It may also be empty without any character. The authority starts right after "http:" and a double slash [216]. Interestingly, Tim Berners-Lee considers the double slash a mistake that he used for no purpose. He would certainly not implement a double slash again [217]. To count how many people have accessed a Web page, a page view is concluded when a page has completely loaded. Page views are a good measuring unit for quantifying Web client traffic. [218]

Request

For example, the HyperText Transfer Protocol (HTTP) enables the retrieval of linked hypertext documents from across the Web. HTTP requests are being used by HTTP clients such as the browser to trigger a response from an HTTP server. An HTTP request consists of a method, URL and the request header. Each request is initiated by a specific method to inform the server what to do with it. The most common HTTP methods are "GET" and "POST". Following the method, syntax demands the URL and HTTP version used. [218]

A Web request is a subset of a request which is mainly used by Web clients. There are two distinct forms of Web requests:

- **Explicit request**
  Manual user initiated request;
- **Implicit request**
  Initiated by the Web client. [218]

While explicit and implicit Web requests are initiated by Web users or by Web clients respectively, there are two differences in terms of Web resource interactions to be clarified:

- **Embedded request**
  Encapsulated request within a Web resource, e.g. hyperlink;
- **User-Input request**
  Direct request to the Web client from the Web user, e.g. click on bookmarked hyperlink.

When a Web user clicks on a hyperlink in an HTML document it is an explicit but embedded Web request, whereas the manual input of a Web address out of an advertisement is considered an explicit user-input Web request. [218]

Response

For example, the HTTP response is the answer of an HTTP server to the client. The HTTP response consists of the HTTP version, the status code of responses and the plain text message of the status code. Following the status code, syntax demands the header and a HyperText Markup Language (HTML) file. HTML is the standard markup/formatting language for the Web. [218]

Web Client

HTTP or Web clients facilitate access to Web resources by sending requests and processing responses that involve the rendering of Web resources. Clients control the process of sending requests to Web servers and receiving responses. A Web client is also called a browser. They come in two distinct designs known as obsolete line browsers and state of the art graphical browsers. Graphical browsers are considered as one of the enabling factors of the success of the Internet. Hyperlinks are part of this technology which refer to other Web resources. A hyperlink points to other Web resources (for example an HTML document) and enhances the navigation, e.g. by a click. [218]

Thus, Web clients send requests to a Web server and a Web server sends back an HTML with the help of HTTP. Web clients interpret HTML instructions from web server responses and present it to the user. [219]

Web Server

A HTTP server or a Web server provide access to Web resources with the help of Web clients. A Web server sends a Web response to a Web client. Within the response is the Web response header which compromises information about the Web server. The actual data displayed in the browser is in the Web response body. [218]

User Session

In the Web, communication is not a persistent flow between the client and a server connection. Suitable information for the identification of unique visitors is necessary. User sessions help Web clients and Web servers to associate data to a specific a Web user. For that reason, sessions must be implemented on

the application layer because an IP address or a client ID may be the same for two Web users. Once a user is online, the session identifies his or her activities. [218]

Since HTTP is a stateless communication protocol, the sender of packets does not expect acknowledgement of receipt by that protocol. This means that the internal state of the server is not known and information about the session itself is not kept on both sides. Requests can only be associated if there is further information available. Therefore, it is sometimes necessary to include additional information in requests that has to be interpreted by the server. [220]

A new session is started every time a new visitor who was not yet assigned a session, accesses a respective Web page. Depending on the transmission method of the session, a session may be terminated without a logout. Methods for sessions are enclosed by means of a cookie in the HTTP header (client side) or server side. Client side sessions are more comfortable for the user if sessions should not expire for a duration of several days or weeks. Such information on a session is stored either on the client or server side and is regularly used by "shopping carts" of online shops. That's why in certain online shops, a returning visitor can still see the items put into the shopping cart, although he or she might not have proceeded to check out before exiting the session. [221]

For Web observations, sessions play a vital role in the process of collecting Web data. Depending on the structure of a Web page, sometimes a session ID – most often a string – or a cookie is attached to the Web client. The client needs this information to get a proper response from the server. By imitation of a request without session ID or cookie no answer from the server can be expected.

Cookie

As mentioned before cookies can be described as unique session identifiers. They store the user session on the client or server side in order to remember the clients interactions in further requests. Cookies store essential information about a Web user interaction, e.g. selection of goods in an online shop. Cookies may also be used for caching purposes, e.g. if a mobile connection to the server is not always given. [218]

Tracking cookies are used for "tagging" visitors to recognise Web users. Ordinary cookies are used on Web sites where the visitor must log in with his credentials. Username, password and email address possibly will identify a user for a session. Nevertheless, tracking cookies are used to observe a user's browsing behaviour over a longer period of time. [222]

Appendix B

Programming Languages

There are numerous programming languages with plenty of libraries that offer functionalities for creating a Web observation. In the agony of choice, it might be a good idea to take a look at the web hosting service GitHub. The service is the home for many open source projects and is used by many programmers for storing programming code of their own projects. According to GitHub, the most commonly used programming language is JavaScript, followed by Python, having displaced Java from the second to the third place in 2017. [223]

From this given fact, it may be inferred that the use of JavaScript and Python is fairly widespread. The larger the programming language community, the more code is available for similar reference work. Moreover, the likelihood that someone else bumped into a similar problem is pretty high. However, the choice of a programming language may also be the result of personal taste. As follows, the two most used programming languages on GitHub are:

- **JavaScript**
  JavaScript or short JS is a high-level programming language which tends to be easier to use in comparison to a lower-level language. Besides many characteristics such as asynchronous, dynamic, and interpreted programming language, it supports event-driven functionalities in order to perform actions in response to e.g. user input. JavaScript has the power of modifying the HTML DOM structure, e.g. elements, attributes, and styles. Furthermore, it can interact with HTML events of a website. [224]
  Node.js, an open-source JavaScript run-time environment, has turned JavaScript from client-side to server-side scripting in order to create dynamic websites and thus enables scalable Web applications. [225]

- **Python**
  Python is also a high-level programming language and is used for general-purpose, a wide variety of application domains [226]. The advantage of Python is its philosophy of code readability by using off-side rules which manifests the indentation blocks [227].

Both programming languages are platform independent and can be run on multiple operating systems.

Libraries

A comprehensive range of libraries are available for many purposes. Libraries are reusable collections of configuration data, templates, pre-written code, subroutines, classes, values, and types. There are static and dynamic libraries; static libraries merge together with program code and are usually faster, in contrast to dynamic libraries which are linked to the code on start-up of the program and thus reusable. Through the use of these libraries, the difficulty of programming is drastically reduced. [228]

Well-known JavaScript libraries on the Web include:

- **Cheerio**
  Cheerio parses markup language, e.g. HTML, for manipulating data structure of a Web resource. It uses jQuery to select tags of the DOM structure of a document. String operations such as *split(), substring(), or replace()*, would be extremely time-consuming. jQuery selectors reduce the length of code to a single line. In the event of a change of the Web resource structure, the selector must also be updated. [132] A similar approach to jQuery in JavaScript for Python is pyQuery [229].

- **Needle**
  Needle provides HTTP/HTTPS request in Node.js for the interaction of Web data. It may also include authentication, file uploads, streaming, XML and JSON parsing. Needle basically enables the communication in between Web servers. [131]

- **Cron**
  Cron jobs are important for repetitious tasks such as a Web observation. Implementing similar functionality in JavaScript may encounter problems in asynchronous functions, time-outs and callbacks, which are sometimes difficult to avoid. Cron reduces the complexity by executing applications on a defined time schedule. [230]

Alexander Olivier Gröflin

Particulars

Address	PO box 156, 4009 Basel, Switzerland
Date of birth	May 25, 1985
Nationality	Swiss

Education

12.12.2018	PhD examination, University of Basel, Switzerland
09.2013 – 12.2018	PhD candidate in Computer Science, University of Basel, Switzerland
10.2012 – 09.2013	MSc in Management of Information Systems and Innovation, London School of Economics and Political Science (LSE), United Kingdom
09.2011 – 09.2012	MSc in Computer Science, University of Camerino (UNICAM), Italy
09.2007 – 09.2012	MSc and BSc in Business Information Systems, University of Applied Sciences Northwestern Switzerland (FHNW)
08.2002 – 08.2006	Professional Maturity, Swiss Federal certificate in Computer Science, High School for Information Technology, WG/WMS Basel, Switzerland

Publications

A. Gröflin, M. Weber, M. Guggisberg, and H. Burkhart (2017). Traffic Flow Measurement of a Public Transport System through automated Web Observation, IEEE Eleventh Int. Conf. on Research Challenges in Information Science (RCIS), Brighton, UK, 2017, 156161.
DOI: 10.1109/RCIS.2017.7956532

S. Price, W. Hall , G. Earl, [and 17 others, including A. Gröflin] (2017). Worldwide Universities Network (WUN) Web Observatory: Applying Lessons from the Web to Transform the Research Data Ecosystem. In Proceedings of the 26th International Conference on World Wide Web Companion (WWW 17 Companion), pages 1665-1667.
DOI: 10.1145/3041021.3051691

A. Gröflin (2016). Poster session, Real-Time Web Observations: Data Science meets HPC, poster presentation for the 45th SPEEDUP workshop.

A. Gröflin, D. Bosch, M. Guggisberg. and H. Burkhart (2015). Facilitating the Reactive Web – A Condition Action System using Node.js. In Proceedings of the 11th International Conference on Web Information Systems and Technologies, ISBN 978-989-758-106-9, pages 89-95. DOI: 10.5220/0005446800890095

Talks

A. Gröflin (September 26, 2018). Web Observationen – Automatische Sammlung von Echtzeit-Daten aus dem Internet zur quantitativen und qualitativen Analyse, Talk at connect Dreiländereck, Datensicherheit und Dateneigentum in Zeiten vernetzter Systeme, Meine Daten gehören mir!, Lörrach, Deutschland.

A. Gröflin (November 17, 2017). Web Observatory – Zooming in or Spying out? Collecting Web Data, PhD Seminar, organised by the Faculty of Law at the University of Basel and the European Criminal Law Academic Network (ECLAN), Basel, Switzerland.

A. Gröflin (February 16, 2017). Architecture of a Real-Time Web Observation, 3rd International Winter School on Big Data "BigData 2017", Università degli Studi di Bari Aldo Moro & Universitat Rovira i Virgili, Bari, Italy.

A. Gröflin & D. Bosch (October 27, 2016). Real-time visualisations for criminal data analysis. Talk at the conference of cantonal prosecution authorities, Staatsanwaltschaft Basel-Stadt, Switzerland.

A. Gröflin, C. Frei, and D. Bosch (September 23, 2016). Daten über das Auto – Was kann ein Web Observatory liefern?, Intelligenter Verkehr – Rechtsfragen im Kontext, Landgut Castelen, Augst, Switzerland.

A. Gröflin (October 2015). Web data extraction at the University of Basel. Talk at World Universities Network (WUN) for the Web Observatory Project, University of Southampton, United Kingdom.

Thesis Co-Advisor

M. Weber (2017). Component Based Web-Scraping Strategies, Master Thesis.

C. Frei (2016). Track a Car: Untersuchung von Methoden zur Gewinnung von Echtzeitdaten am Beispiel Catch a Car, Bachelor Thesis.

M. Weber (2015). Visualisation of Real-Time Web Data [of a local transport provider], Bachelor Thesis.

J. Simonet (2015). Leserkommentare des Newsportals 20 Minuten; Analyse von Webinhalten mithilfe des Condition Action Tools WebAPI, Bachelor Thesis.

E. Hodzic (2015). Programmierbare Webassistenten: Technologien und Nutzungsszenarien, Bachelor Thesis.

Teaching Assistant and Examiner

10851-01 – Vorlesung: Werkzeuge der Informatik, Prof. Dr. Helmar Burkhart (HS 2014, HS 2015).

10906-01 – Vorlesung: Algorithmen und Datenstrukturen, Prof. Dr. Helmar Burkhart (FS 2014, FS 2015, FS 2016).

28518-01 – Vorlesung: Web Data Management, Prof. Dr. Helmar Burkhart & Prof. Dr. Heiko Schuldt (HS 2016).