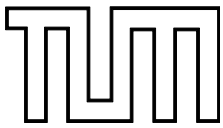
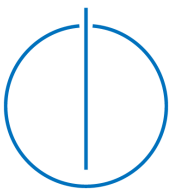


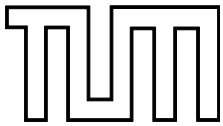
DEPARTMENT OF INFORMATICS
OF THE TECHNICAL UNIVERSITY OF MUNICH

Bachelor's Thesis in Information Systems

Design of Big Data Reference Architectures for Use Cases in the Insurance Sector

Vladimir Elvov



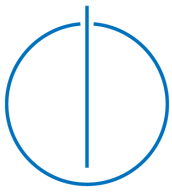


DEPARTMENT OF INFORMATICS
OF THE TECHNICAL UNIVERSITY OF MUNICH

Bachelor's Thesis in Information Systems

Design of Big Data Reference Architectures for Use Cases in the Insurance Sector

Author:	Vladimir Elvov
Supervisor:	Prof. Dr. Florian Matthes
Advisor:	Martin Kleehaus
Date:	15.03.2018



I hereby assure that the thesis submitted is my own unaided work. All direct or indirect sources are acknowledged as references.

Munich, March 15th 2018

_____

Vladimir Elvov

Abstract

In spite of Big Data and the corresponding technologies being already widespread, many Big Data projects do fail or do not deliver the promised results. This is also the case in the insurance sector. One reason is the approach to the respective projects: It doesn't take into account or analyze the business value of a specific Big Data Use Case or its implementation complexity and feasibility.

This thesis uses a different approach by analyzing the business aspects of Use Cases in the insurance sector at the beginning. First, a number of possible Use Cases in insurance is derived from a literature study. Then these Use Cases are evaluated in expert interviews in order to find out, which ones do have the highest potential.

Eventually, based on these requirements and a comparison of existing Reference Architectures for Big Data in common, a new Reference Architecture is designed for Big Data in insurance. The final goal is to use this Reference Architecture as a blueprint for deriving components in order to implement Big Data Use Cases in insurance. The Reference Architecture is tested in a case study by designing a Solution Architecture for two specific Use Cases.

Keywords: Big Data, Insurance, Reference Architecture, Solution Architecture, Requirements Engineering, Requirements Analysis, Big Data Use Cases, Streaming Architecture, Machine Learning, NoSQL, Data Lake, Apache Kafka, Google Cloud Platform, Microsoft Azure.

Outline

1. Introduction 1

1.1 Motivation 1

1.2 Research Questions 2

1.3 Approach 3

2. Foundations 6

2.1 Big Data 6

2.1.1 Definition 6

2.1.2 Machine Learning in Big Data 9

2.1.3 Other Important Technical Terms 12

2.1.4 Big Data in Insurance 16

2.1.5 Big Data and Privacy 19

2.2 Requirements Engineering 22

2.2.1 Common Process 22

2.2.2 Operationalizing Use Cases Based on Requirements 23

2.3 Reference Architecture 27

2.3.1 Definition 27

2.3.2 Analysis of Existing Big Data Reference Architectures 29

2.3.3 Deriving Common Components 36

2.4 Related Work 39

2.4.1 Marr (2015) 39

2.4.2 National Institute of Standards and Technologies (2015) 39

2.4.3 Fox et al. (2014) 40

2.4.4 Lanquillon et al. (2015) 40

3. Big Data Use Cases in Insurance 41

3.1 Use Case Description Methodology 41

3.2 Use Case Overview 43

3.2.1 Customer Analytics 44

3.2.2 Internal Processes 48

3.2.3 IoT in Property and Casualty 53

3.2.4 Smart Health and Smart Life 58

3.3 Use Case Evaluation Methodology and Results 64

4. Requirements for Operationalizing Big Data Use Cases 67

4.1 Use Case specific Requirements 67

4.1.1 Customer Analytics 67

4.1.2 Internal Processes 73

4.1.3 IoT in Property & Casualty 83

4.1.4 Smart Health and Smart Life ... 90
4.2 Generic Requirements ... 100
4.3 Comparison of Big Data Reference Architectures ... 107
5. Big Data Reference Architecture for the Insurance Sector ... 111
5.1 Mapping Requirements to Architecture Components ... 111
5.2 Top-Level Big Data Reference Architecture for the Insurance Sector ... 112
5.2.1 Big Data Reference Architecture - Level 1 ... 112
5.2.2 Big Data Reference Architecture – Level 2 ... 113
5.3 Case Study: Big Data Solution Architecture for Selected Use Cases ... 118
5.3.1 Notation Definition ... 119
5.3.2 Big Data Platform Approach ... 119
5.3.3 Case Study – Claims Settlement ... 121
6. Evaluation ... 125
7. Conclusion ... 128
7.1 Summary ... 128
7.2 Outlook ... 129
Bibliography ... 131

List of Abbreviations

ACID	Atomicity, Consistency, Isolation and Durability
AI	Artificial Intelligence
AWS	Amazon Web Services
BDSG	Bundesdatenschutzgesetz
BI	Business Intelligence
CRM	Customer Relationship Management
GCP	Google Cloud Platform
GDPR	General Data Protection Regulation
HDFS	Hadoop Distributed File System
I&AM	Identity & Access Management
IoT	Internet of Things
ML	Machine Learning
NIST	National Institute of Standards and Technologies
OCR	Optical Character Recognition
PAYD	Pay As You Drive
P&C	Property & Casualty
PII	Personally Identifiable Information
PHYD	Pay How You Drive
RQ	Research Question
TOGAF	The Open Group Architecture Framework

1. Introduction

To begin with, the importance of Big Data for the insurance sector and the corresponding challenges are described, thus building the motivation for this thesis. Based on this motivation, the three Research Questions (RQ) to be answered in this thesis are posed and explained. At the end of this section, the scientific approach chosen to answer the RQs is briefly outlined.

1.1 Motivation

Big Data is undoubtedly one of the major drivers of digitization. Data is generated in vast volumes and in many cases also in real-time. Big Data forms the technological foundation for many new innovations and technologies including Internet of Things (IoT), Artificial Intelligence (AI), autonomous driving and many more. By collecting, storing and analyzing all this data, companies can gain huge advantages in both traditional and new markets. The insurance sector is no exception: According to a joint study by Google and the consultancy Bain & Company, there is a potential of 4 billions of euros in growth and 14 billions of euros in cost reduction in the German Property and Casualty (P&C) insurance sector alone [1]. McKinsey, another consultancy, reckons that the cost of settling insurance claims can be reduced by 30% due to automation [2]. Apparently there are many great opportunities for insurers, if they can successfully bring Big Data into action. For those who do not manage to put Big Data to work for them, the ramifications can be quite severe [2].
However, in spite of Big Data and the technologies it is based on already being widespread, many Big Data projects still do fail. According to a prediction by Gartner, a research and advisory company, from 2015 to 2017 60% of all Big Data projects will not come further than to an exploration stage and fail [58]. Several companies have not even started working on Big Data projects that go beyond a Proof of Concept (PoC) stage. In a study conducted by the Technical University of Munich (TUM) the progress of Big Data projects at 25 selected DAX-companies[1] was analyzed. The result was that in 2015 only 1 out of 25 companies had Big Data projects at a stage of deployment and further four companies were at a stage of preparing a deployment [3].
One reason for the poor performance so far is the approach many companies choose when it comes to new Big Data projects. Hitherto they have started out with technology by collecting and analyzing data directly without having defined a business goal for using Big Data. Although this way might work for data-driven companies like Google, which have their data ready for analytics, for the great majority of businesses this is the wrong approach [4, 15]. The reason is that most Big Data projects going beyond a PoC require new technologies and experts for working with them, thus causing a need for large financial investments. However,

[1] DAX: Deutscher Aktienindex, the main German stock market index consisting of 30 German companies whose shares are traded at the Frankfurt Stock Exchange.

as long as one does not have a strategic goal and a clearly defined added value for such a project, these investments are not justified. This is also often the case in the insurance sector.
Furthermore there is a lack of architectural standards for developing Big Data solutions, which are not focused on a specific product landscape from a software vendor [9]. Such standard would be for instance a Big Data Reference architecture that can be used as a guideline for implementing Big Data Use Cases in insurance. The advantage of such a product-independent Reference Architecture would be that a company could first choose the components and afterwards the corresponding products needed to operationalize a Use Case flexibly without having the risk of an expensive vendor lock-in.
This thesis chooses a different approach by starting out with regarding first the strategic and business aspects of Big Data projects [4]. It envisages the identification of Use Cases, which can change the business model of an insurance company. The Use Cases cover most insurance products, including P&C, health, life and industrial insurance. Additionally internal processes and customer analytics are scrutinized for possible usage of Big Data there. For each of these Use Cases an analysis of its added value and feasibility is conducted. Afterwards the requirements for implementing these Use Cases are analyzed before designing a new product-independent Reference Architecture for the insurance sector. The final goal of this thesis is to provide a suggestion on how to implement a Big Data Use Case in insurance using the newly developed Reference Architecture.

1.2 Research Questions

Research Question 1. What are possible Big Data Use Cases in the insurance sector and which ones do have the highest potential?
This RQ focuses on identifying the Big Data Use Cases, which can be implemented for solving business problems in insurance companies. Their goal can be for instance to reduce costs through automation or to create growth by offering new products with services based on Big Data. According to an analysis by McKinsey, companies focusing on business aspects first when it comes to Big Data projects, perform better than those starting directly with technology issues. This can be achieved by concentrating on business strategy and business-driven Use Cases [15]. The Use Cases themselves are identified through a literature review, consisting particularly of whitepapers describing case studies, PoC-projects or already implemented Big Data applications in insurance. For the list of Use Cases and a detailed explanation, please refer to chapter 3.2.
Above all, the Use Cases are evaluated in structured interviews with senior managers or experts in the field of Big Data or Data Science at the large insurance company. The goal here is to identify the Use Cases with the highest potential, so that implementing them will deliver an added value to the company whilst keeping complexity and risks at a reasonable level. The evaluation is inspired by a framework from the consulting company PricewaterhouseCoopers (PwC) and includes categories such as added value and feasibility [5]. Risks arising when implementing a Use Case are analyzed as well [4]. For a detailed overview of the evaluation methodology, please refer to chapter 3.1.

Research Question 2. Which requirements have to be fulfilled in order to implement the Use Cases from RQ1?
In this RQ requirements for operationalizing the Use Cases identified in RQ1 are elicited and analyzed. For getting a better understanding of the requirements for a specific Use Case, further interviews were conducted with experts on this respective Use Case or insurance section. Furthermore, a brief literature analysis was used for answering this RQ. Eventually each Use Case does have an overview of requirements needed to be met for its implementation being made up of several categories (data source, analytics, business, etc.). Additionally, a collection of generic requirements for all Use Cases is derived from the overview of specific requirements for the single Use Cases.
The approach in RQ2 is based on the methodology of the National Institute of Standards and Technologies from the U.S. Department of Commerce for analyzing requirements for designing a Reference Architecture [6, 7]. For more details on the requirements analysis methodology please refer to chapter 2.2, for the requirements based on the Use Cases to chapter 4.1.

Research Question 3. What can a Big Data Reference Architecture look like in order to operationalize the Use Cases from RQ1?
The goal of this RQ is to design a new Big Data Reference Architecture for the insurance sector. This Reference Architecture would then work as a blueprint for implementing single Use Cases and developing a Use Cases specific Reference Architecture.
For designing the generic Reference Architecture, first, existing Big Data Architectures from companies like e.g. Google, Oracle or Microsoft are analyzed. The goal is to identify common components, which every Big Data Reference Architecture should contain. Afterwards the generic requirements from RQ2 are mapped to an architectural component. Together the requirements and the common components make up the foundation of the new Big Data Reference Architecture.
The approach in RQ3 is based on the methodology of the National Institute of Standards and Technologies from the U.S. Department of Commerce designing a Reference Architecture [7, 8]. For more details on the methodology please refer to chapter 2.3, for the new Reference Architecture to chapter 5.2.

1.3 Approach

This section describes the general scientific approach used for working on the entire thesis. It is based on a methodology named Design Science Research Methodology that provides a guideline for research in the field of Information Systems [10]. Figure 1.2 shows the generic process used in the Design Science Research Methodology. In order to simplify the process, this thesis dispenses with the loop and goes only through one iteration of the Design Science Research Methodology. This is visualized in figure 1.1. Although the loop is not carried out for the Use Cases, the results of the evaluation of the Reference architecture, the Big Data platform and the Solution Architecture are taken into account before designing the final version of these architectural artifacts.

In the beginning, the topic is motivated by identifying the problems with the current state of Big Data in insurance. Afterwards the objectives of the thesis are defined by formulating and explaining the Research Questions. Then the work on designing and developing the artifacts for the problem's solution starts. The artifacts created in this thesis are the following:

- A list of possible Big Data Use Cases in the insurance sector.
- A classification of the Use Cases to identify the ones having the highest potential. The development of this classification is also part of the Evaluation stage.
- An overview of both Use Case specific and generic requirements needed to operationalize the Use Cases found before.
- A Big Data Reference Architecture for the insurance sector that is based on the requirements derived in the previous step.

This final artifact is then used for the Demonstration stage: By designing a Use Case specific Solution Architecture, the Reference Architecture is tested in order to find out, whether it is suitable to be used as a blueprint to operationalize Big Data Use Cases in insurance.
In the following Evaluation stage, the Reference Architecture is scrutinized in interviews with experts on architecture. Additionally the Use Cases found are analyzed in structured interviews with experts and senior managers from an insurance company.
Eventually the results are communicated in a final presentation and by publishing this thesis. For intellectual property and non-disclosure reasons, the results of the Use Case evaluations will only be available within the insurance company where the interviews were conducted.
Besides using structured and unstructured expert interviews, this thesis relies on literature reviews. Relevant literature is found in scientific databases like for example Google Scholar, SpringerLink or EBSCOhost. For simplifying the research, search keywords were identified, including Big Data, Big Data in Insurance, Insurance Use Cases, Big Data Reference Architecture, Machine Learning, Requirements Engineering and Requirements analysis in a Big Data context. The literature includes both scientific publications and business publications from companies excelling in the area of Big Data and or insurance.
For more details on the approach used for finding the answers to a specific Research Question, please refer to chapter 1.2, where each one of them is described.

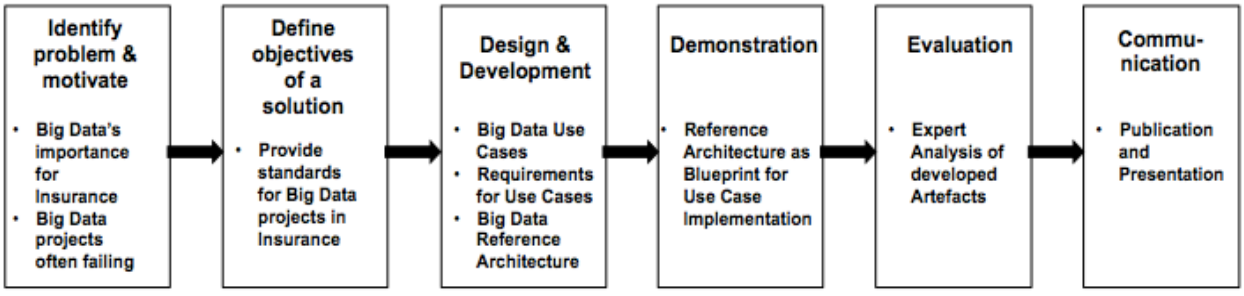


Figure 1.1 : Design Science Research Methodology applied in this thesis
Source: Own depiction based on:
http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.535.7773&rep=rep1&type=pdf

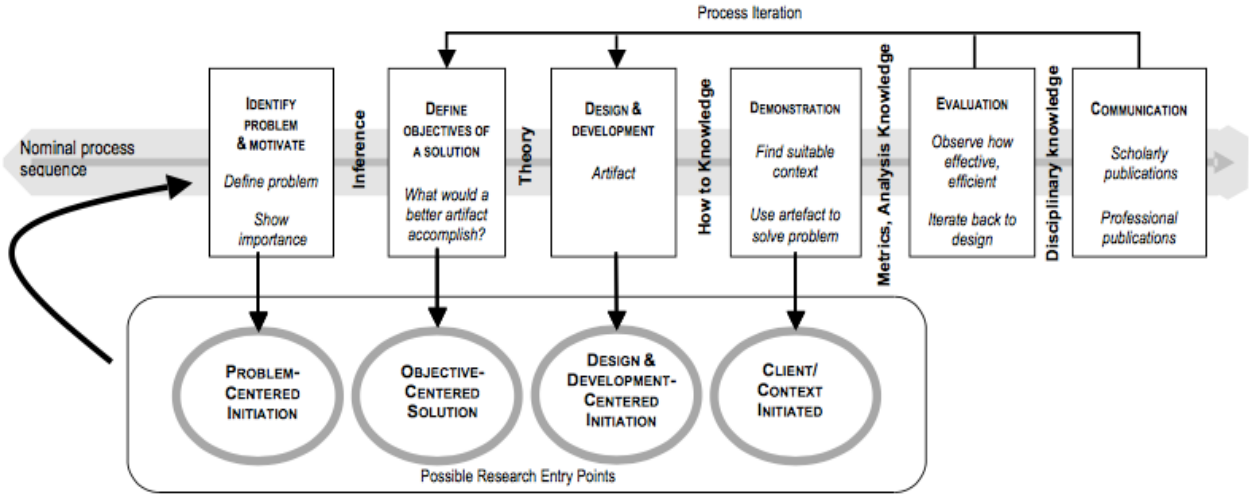


Figure 1.2: Design Science Research Methodology
Source: Figure 1 from:
http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.535.7773&rep=rep1&type=pdf

2. Foundations

This chapter describes the theoretical concepts, which are the foundations of this thesis. In the beginning the terms Big Data, Machine Learning (ML) and some other important technical terms are explained. After pointing out the relevance of Big Data for the insurance sector, Requirements Engineering and the process of operationalizing Big Data Use Cases using requirements analysis are presented. Then the term Reference Architecture is defined followed by an analysis of existing Big Data Reference Architectures. Afterwards common components in these Big Data Reference Architectures are established. Eventually, related work from several scientific sources is briefly outlined, which has been used as a reference for choosing an approach to writing this thesis.

2.1 Big Data

2.1.1 Definition

In an issue published in May 2017, the London-based magazine The Economist named data "the world's most valuable resource" dedicating the cover article to this topic. Unsurprisingly, the term Big Data is nowadays both widely known and used. Back in 2012 the Gartner Hype Cycle, an indicator for the maturity of technologies, had Big Data listed at the "Peak of Inflated Expectations" [11]. Three years later it was removed from the Hype Cycle entirely as other technologies like ML, IoT or Advanced Analytics relying upon Big Data have taken its place [12].
Since it is that widely spread, the term Big Data is often used to describe various technologies and paradigms that are related to processing and analyzing data. Basically, it refers to datasets so large they cannot be stored or processed in traditional relational databases [13]. The challenges conventional relational databases face are besides the vast amount of data also the lack of structure within the datasets and the speed of data generation. In a more broad sense Big Data covers not only the datasets themselves, but also the collection of technologies used for storing and analyzing them [14]. In spite of numerous definitions for Big Data available, most experts agree on using the so-called Three V's for characterizing it [16, 18, 19, 20, 21]:

- **Volume**: It is used to describe the vast amount of data where the "Big" in Big Data comes from. Without this property, Big Data and the corresponding Use Cases would not be possible at all. Having that much data enables applications relying on large datasets for training their models and predicting outcomes to deliver precise results, e.g. in ML. Peter Norvig, Google's Director of Research said, "We [Google] don't have better algorithms. We just have more data", thus pointing out the importance of the volume property [16]. Indeed the volume of data available today is huge. The US retail giant Walmart collects on average 2.5 Petabyte[2] of data per hour. Back in 2012 2.5

[2] One Petabyte is equivalent to one million Gigabytes.

Exabyte[3] were created per day making it about 900 Exabyte of data for the entire year [16]. In 2016 the annual amount of data created rose to 16.1 Zettabyte[4] and it is expected to rise to 163 Zettabyte by 2025 according to a report by IDC, a market research group [17].

- **Velocity**: This property refers to the speed at which new data is generated. Sensors in cars, manufacturing plants and wearable devices make it possible to collect new data at (near) real-time speed. That poses requirements for being able to collect and process the data at the speed it is created in order to be able to get the best results from it. Additionally the systems also need to be able to analyze the data with the same velocity, so that the insights from the data can be used instantly for business purposes. In some applications like for instance autonomously driving cars, this is of paramount importance for the passenger's security. The velocity of course helps boosting the volume of data thus forming another important characteristic for Big Data.

- **Variety**: The third V is used for characterizing the various data-sources and –types leading to the huge data volume. The variety of sources is particularly remarkable: Sensors, social networks, enterprise applications, customer interactions and many more generate data today. Depending on the source, data often needs cleansing processes for improving its quality before it can be pipelined to the analytical models. The data can be grouped in three categories for describing its type [4]:

    - **Structured**: Data from this category has a clear scheme and can be easily stored and processed. For instance customer data from enterprise applications that can be stored in a relational database is such structured data.

    - **Semi-structured:** This data has some sort of partial scheme, which can be used to structure the data at least to some extent. An example would be customer communication from an e-mail. It has a structured part, which is made up of a sender, a receiver, a subject and a date or a timestamp. However the content of the mail is unstructured since it consist of text written by the customer or the company's employee and needs specific technologies – in this case text analytics – to analyze it.

    - **Unstructured**: This type of data has no structure at all and is both difficult to store and analyze. Unstructured data often needs to be made machine-readable first before it can be analyzed. Examples here include texts, images, voice or video recordings. Some 80 to 85% of data is unstructured, however, it is not absolutely necessary to process and analyze it for most Big Data Use Cases [4, 18].

[3] One Exabyte is equivalent to one billion Gigabytes.
[4] One Zettabyte is equivalent to one trillion Gigabytes.

Besides the classification concerning the data structure, another categorization is possible concerning the data sources [4]:

- **Internal:** This refers to any data currently available inside the company and hence comparatively easy to access (sometimes data coming from legacy systems is difficult to get, however this is not a Big Data issue but a corporate IT one). Examples here would be customer, sales or transactional data. Internal data often tends to be structured.

- **External:** Generally, this data can be anything available outside of the company. This can include social media, weather or census data and much more. Here one of the main challenges is the integration of external data into the company's storage zone. Some of this external data is available open-source whilst other data has to be paid for if collected from a data provider.

Often applications do require various data sources for building their analytical models thus needing a central data integration component for these sources. Particularly dealing with these integration and transformation issues requires modern Big Data technologies.

Recently two more V's have appeared. IBM, a technology company, has introduced Veracity, which describes the completeness and quality of the datasets [18, 19]. This V is highly important, since the quality of the data has a huge impact for the precision of the analytical models. Hence the better the data quality, the better the outcome of the models is. However, often raw data has poor quality thus requiring structuring and cleansing processes before the datasets can be used by any models [57]. The last V refers to Variability [21]. This property refers to possible changes in data like for instance flow rate, the format or its meaning. One example where this characteristic is important is the use of Big Data for Natural Language Processing or sentiment analysis. Depending on the pronunciation or the usage of a word (e.g. "great" or another adjective) in a sentence it can have a completely different meaning [22].
Some critics point out that Big Data is nothing more than already known concepts from data processing and analysis like e.g. Business Intelligence (BI) [11]. Apparently the amount of data Big Data deals with already outperforms BI by far. BI is only able to process structured data, whilst Big Data offers technologies to work with unstructured data as well. Furthermore, back in the 1990s data could be neither processed nor analyzed at (near) real-time speed. The bottom line is that particularly the Three V's do differentiate Big Data from any previous technologies and concepts available in the field of data analysis.
In fact, there are even more V's for describing Big Data, e.g. value or visualization. However, since they are not that commonly agreed on or partially unfitting for a scientific definition of the term Big Data, they are left out here.

2.1.2 Machine Learning in Big Data

When it comes to Big Data, the term Machine Learning is never far away. And although used sometimes synonymously, the two terms describe different things. Whilst Big Data deals with the amount of data and some other properties concerning it, ML is about analyzing data algorithmically for recognizing patterns in it or predicting something using approximations. It is a subfield of AI with the key point being the ability of machines to learn themselves based on a dataset without having a programmed set of rules. In case the underlying data changes, the ML system adjusts the outcomes of its model(s) respectively without a human intervention. This significantly differentiates ML from a conventional enterprise application that is based on a set of human-defined, hard-coded and only human-changed business rules. Although ML has existed sin a basic form since the 1950s, it is Big Data that makes it so popular and useful for companies today. The more data a model is supplied with, the more precise its outcome gets, also because it can process many more examples than a human. When ML algorithms are applied to Big Data, i.e. a large dataset, the process is called data mining. Just like when mining gold or other precious metals, the valuable insights have to be extracted from raw data by using sophisticated techniques. Eventually, after training models with a huge dataset, ML systems become more precise, adaptive and powerful when compared to conventional applications [23, 25]. ML is also closely related to the field of Data Science, which relies on ML algorithms for analyzing data.
Generally speaking, ML can be used for solving the following two problems: Classification and regression. In classification the goal is to classify a variable into a category, e.g. whether an insurance claim is fraudulent or not. Also determining whether a component in a manufacturing plant will break down or not is a classification problem. Regression, on the other hand, aims to predict a concrete value, e.g. what is the price of a stock going to be [23, 24, 25]. Besides the usage in financial services, ML can be applied in various areas, such as customer analytics, medical research and diagnostics, spam filtering, image recognition or Natural Language Processing.
ML can be basically divided in two categories: Supervised and unsupervised ML. In supervised learning the data used for training the model is already labeled, i.e. each input has already a clear output or categorization. A supervisor provides the labeling – that is where the name supervised ML comes from. Both regression and classification problems can be solved through supervised ML. In unsupervised learning, there is no supervisor so the data used for training has no labels. Here the aim is to detect a structure in the underlying data so that the machine can come up with its own labeling. In order to do this, the machine clusters data into groups or segments by performing a so-called density estimation [25]. Thus unsupervised learning enables knowledge discovery so that companies can use these newfound patterns and connections. For example neural networks rely on unsupervised learning for e.g. clustering images in image recognition and detecting previously unknown patterns in them.
For providing a better understanding of the field of ML some of the most known and used algorithms for prediction purposes are explained here [24, 25].

- **Linear Models:** These are models used to predict an outcome using a simple formula based on set of data points. First, the variable to predict is identified and then the rele-

vant parameters are combined in a formula by weighting each of them. These parameters are attributes coming from the data-source the model is trained on. Since these models are very simple, and do not take complex relationships (i.e. non-linear) between parameters into account, they are unfitting for predicting complex outcomes. The problem with linear models is their "overfitting" to historical data. This means after a model has been trained and has found an approximation for predicting an outcome, it has difficulties adapting to new data when deployed to production.

    - **Linear Regression:** In this model, one tries to predict the value of a so-called dependent variable by analyzing the influence of so-called exploratory variables. In case there is only one exploratory variable, it is a simple linear regression. In case there are several of them, the model is called multivariate linear regression.

    - **Logistic Regression:** This algorithm is basically the same as linear regression with the only difference being the type of the variable tried to predict: It is a classification problem what means that it can only take values of true or false.

- **Tree-based:** These models are used for visualizing decision rules in form of tree branches and can be applied for predicting an outcome in a non-linear scenario or relationship. The so-called Divide-and-Conquer approach from computer science is applied here, i.e. a problem is split up in so many hierarchical steps sequentially until it can be solved. Tree-based models can be used for both classification and regression problems.

    - **Decision Tree:** Here the possible outcomes are visualized by branching them after each other in a tree. The model makes decisions sequentially as it runs through the tree. When training a decision tree, the underlying data is analyzed for finding the best branching points.

    - **Random Forest:** This is combination of several decision trees with the final outcome being the average of all decision trees involved in the model. The trees in the model are trained together at the same time with a randomly chosen dataset. A single tree in the forest is less precise than a fully trained decision tree. However the entire forest does outperform a full decision tree, hence it is more diversified and relies on more parameters and specifics.

    - **Gradient Boosting:** This model again combines several decision trees with the specialty that the single underlying trees are trained one after another. Due to this, the tree currently being trained can focus on data that has delivered inconsistent or false results in the previous trees. This approach makes it possible for gradient boosting to predict difficult outcomes as it also takes complex situations into account.

- **Neural networks:** A neural network is a concept from the sphere of AI. In fact, they are actually called artificial neural networks since they are trying to imitate the behavior of a human brain. In biology neurons sending messages to each other through a network form a so-called neural network, which are basically the foundations of a human brain. Neural networks require a vast amount of power and computing resources for training them, but are suitable for solving some of the most complex problems in

ML including the ones based on the analysis of large data volumes and unstructured data. This can be image recognizing or Natural Language Processing. Deep Learning, another popular term right now, uses neural networks for training its models. In fact, Deep Learning simply combines several layers of neural networks following the paradigm that the more layers do exist, the more complex problems can be solved.

For comparing the different algorithms, the availability of data required for training, the time it takes to train the model and processing speed before a prediction is made are relevant criteria. Of course, depending on the complexity of the problem to be solved, some of the algorithms are unsuited. Table 2.1 provides a brief overview of advantages and disadvantages when using the algorithms described before [24].

Algorithm	Advantages	Disadvantages
Linear Regression	Easy to build and use.	Too simple for complex prediction problems. Sometimes "overfits" to the data it is trained on.
Logistic Regression	Easy to build and use.	Too simple for complex prediction problems. Sometimes "overfits" to the data it is trained on.
Decision Trees	Easy to build and use.	Too simple for complex prediction problems.
Random Forest	Precise results based on average scores and fast to train.	Not suited for real-time or very fast predictions since it takes long time to perform a calculation. Predictions are difficult to understand.
Gradient Boosting	Precise results based on average scores and fast to train.	Small changes in the dataset can result in big changes to the model's outcome. Predictions are difficult to understand.
Neural Network	Suited for very complex prediction problems and ML tasks.	Take very long time for training. Require a lot of power and computational resources. Predictions are very difficult to understand.

Table 2.1: Overview of Machine Learning Algorithms with their advantages and disadvantages.
Source: Own depiction, based on http://dataconomy.com/2017/03/beginners-guide-machine-learning/

Another highly interesting aspect made possible through ML is reinforcement learning, which is an unsupervised learning model. Here the focus lies not on a single outcome of the model, but on the sequence of actions resulting from many outcomes. Depending on this sequence of actions the machine being trained can adapt its behavior in such a way, so that it optimizes the outcome. Above all, ML also enables outlier detection, i.e. after detecting a new rule, the model can focus on trying to find an explanation for the data points not fitting into this pattern. This can be helpful for detecting inconsistencies or anomalies that can for instance indicate fraudulent insurance claims [25].
Concluding, it can be said that ML is a cornerstone for Big Data since without it using Big Data would be pointless as no insights could be generated from the large amounts of data.

2.1.3 Other Important Technical Terms

After looking at Big Data and Machine Learning in the two previous sections, this one introduces and explains a collection of various technical terms that will be used throughout this thesis. They include database and data processing paradigms and also several concepts for data analysis.

NoSQL: Actually called Not-Only-SQL, this describes a paradigm in database technologies for storing datasets that are too big for conventional relational databases or do require sophisticated analytics possibilities. The need for these new paradigms comes furthermore from data types (semi- and unstructured data), volume and velocity of the data. According to a definition by the National Institute for Standards and Technologies, NoSQL describes data models that "do not follow relational algebra for the storage and manipulation of data". This means that data models in NoSQL databases are non-relational and often schema-less [21]. NoSQL databases are designed to deal with the following aspects that are very important when building Big Data systems: scalability, partition tolerance and high performance. The first one covers capacities to deal with the volume of data. Partition tolerance refers to the ability of recovering after an outage without data losses whilst high performance guarantees high availability for applications requiring low latency and high velocity. NoSQL database systems run distributed on a high number of machines so that all these requirements can be fulfilled [30]. Additionally, consistency is a very important issue when it comes to databases. Although some NoSQL databases offer ACID[5], they are not primarily designed for it. Most of them offer only weak consistency guarantees like for example eventually consistent[6].
NoSQL databases can be roughly classified in four categories [21, 30, 31]:

- **Column-Based:** Data is stored in columns what makes it possible to perform fast read and search operations on the data when compared to relational databases where data is

[5] ACID stands for Atomicity, Consistency, Isolation and Durability. It is the consistency model used in relational databases and guarantees high consistency
[6] Eventually consistent is a consistency guarantee offered by many NoSQL systems. It means that a data item will be finally consistent if no updates were made on it for a certain time period.

stored in rows. Examples include Google's BigTable, HBase and Apache Cassandra. One of the newest systems here is Apache Kudu that combines the advantages of HBase and HDFS by supporting both dynamic updates on data and fast analytics directly without the need for additional analytical frameworks [34, 35].

- **Key-Value-Store:** Here data is stored in key value pairs with the values being retrieved through the key. It is possible to store both structured and unstructured data using a key-value-store. An example would be Amazon's Dynamo or OracleNoSQLDatabase.

- **Graph Databases:** These are used for describing the relationships between data items using graphs. The items are modeled as nodes and the relationships as links between the nodes with each of them having certain properties. Such databases are very helpful for network analysis in social networks or fraud detection. Examples include Neo4J or OrientDB.

- **Document Databases:** Actually this is a form of a key-value-store where the data is made up of single documents. They are schema-less so that attributes can be added to any field needed thus enabling high flexibility compared to relational databases. Examples are MongoDB, CouchDB and IBMNotes.

In some cases instead of storing data in databases it is also possible to put it into a distributed file system. Often the Hadoop Distributed File System (HDFS) is being used for this purpose. This makes it possible to easily store large amounts of data in files as HDFS was designed for mass-storage. It follows the write-once read-often principle, which means that data cannot be updated later but enables analytics directly on the data in HDFS.

Data processing frameworks: In order to get an insight from the data collected, one has to process it so that analytical models can be applied to it. Processing means going through the data for detecting patterns or structures in it. This requires the usage of a processing framework that has a so-called core engine the processing algorithms can run on. There are many different frameworks but they can be roughly grouped into two categories: Depending on the way they handle the data there are batch-oriented or streaming frameworks whilst some offer both, e.g. Apache Spark [32].
Batch oriented frameworks are designed to process a huge amount of persistent data, e.g. historical records. Thus, averages, totals and other scores can be easily calculated even when handling a huge dataset. However, batch processing is quite slow and hence not suited for real-time applications requiring high velocity. The most famous example for a batch-processing framework is Apache Hadoop and is used by many companies working with Big Data. Its main components are HDFS, YARN and MapReduce. HDFS is the file system where the source data for processing is stored and takes care of data always being available for processing, even in case of a server outage. YARN (Yet Another Resource Negotiator) is a Resource Management component, which ensures that several processing tasks can run within Hadoop at the same time by coordinating the available resources. MapReduce that has

been developed by Google has been the algorithm at the core of Hadoop. It is responsible for executing the processing itself by extracting data from HDFS, dividing and distributing it between the machines, carrying out computations, and combining the results before writing them to HDFS [32]. However, MapReduce is increasingly viewed as outdated with new more performing algorithms available for data processing. In 2015 Cloudera, one of the largest Big Data services providers, decided to replace MapReduce as the core engine within Hadoop with Apache Spark. The reasons are that MapReduce is both difficult to implement and maintain and additionally not suited for velocity applications since it is too slow [33].

Streaming frameworks process data directly as soon as it comes into a Big Data application. This means models are applied not to an entire dataset but to single data items when they enter the system. In stream processing there is no complete dataset; there is only the data available so far. Unlike batch frameworks, streaming frameworks keep only very few records about the state of data items. The state includes information about interim computing results or previous values of a data item. Although these frameworks can process large amounts of data as well, they can only take care of a few data items at a time. Since stream processing frameworks do excel when it comes to low latency, they are perfectly suited for velocity applications requiring very quick data processing. An example for such a framework would be Apache Storm. Its processing capabilities are built on directed acyclic graphs that are called topologies. Each incoming data item will run through these topologies where processing steps will be applied to it. These processing steps are mostly simple operations that combined do form a topology. A topology is made up of streams, spouts and bolts. A spout is a data stream that enters Storm at the edge of a topology, e.g. through an API[7] or a queue. A bolt is a processing step that takes out data items and executes operations on them afterwards sending out the processing results as a stream. Apache Storm guarantees that each data item passing through it will be processed at least once what in cases of outages or failures can lead to multiple processing (duplicates). In order to bypass this, Trident, a high level abstraction of Apache Storm is available. Although Trident does solve the exactly-once problem and brings in some state to data items it also increases latency and uses micro-batching instead of direct stream processing. Besides Storm, Apache Samza is a framework that can be used for stream processing [32, 36].

One of the most famous data processing frameworks for streaming is Apache Spark. However, it actually is designed for micro-batching rather than single-item processing in streams like Apache Storm. Micro-batching is a mixture of batch processing and streaming where small batches are processed very fast with keeping states of data items. Spark's main goal is to improve the speed of batch processing by using in-memory computing. This means that the persistent storage layer is only accessed for loading data into Spark and sending computation results back there. Thus Spark is able to outperform Hadoop by 100 times concerning speed when run in-memory [37]. Just like Storm it uses directed acyclic graphs for defining the operations to be performed on data items as well as the data items themselves. As Spark was initially designed to improve batch processing, there is a component named Spark Streaming

[7] API: Application Programming Interface

that is responsible for the "real" stream processing. It is basically a streaming pipeline that uses the micro-batching approach. The general principle behind Spark Streaming is splitting the stream into small data items that are than processed as very small batches. Thanks to its integration with Spark SQL and MLlib[8], Spark Streaming can process stream data from various sources and apply ML algorithms to it. The bottom line about Spark is that it is able to support both streaming and batch-processing thus making it possible to use only one framework for all purposes. Spark jobs are easier to write than MapReduce jobs and it is also much faster than Hadoop, however as RAM[9] is more expensive than disk storage, Spark can be more expensive concerning operating costs. Another example of a framework that enables batch as well as stream processing is Apache Flink [32, 36, 37].

Data Lifecycle: Another important issue that comes up when dealing with data storage in a Big Data context is the data lifecycle. Depending on the purpose of the Big Data application the traditional data lifecycle where data is first pipelined through an Extract Transform Load process (ETL) and cleansed before being stored persistently in a data warehouse are unfitting. Instead when dealing with large data volumes, data has to be stored in its raw state with cleansing, transforming and aggregating being applied at the time when the data is actually extracted for analysis. This approach is called schema-on-read. In other systems, which require very high performance (velocity applications) data is directly cleansed, transformed and aggregated as soon as it comes in for direct pipelining into the analytics components. The data itself is persistently stored only afterwards, thus giving the performance needed [21, 57]. Since Identity & Access Management (I&AM) is a very important topic when it comes to data privacy, labeling data with respective access attributes as soon as it is collected can be required for compliance reasons. This ensures that no one unauthorized can access potentially sensible data like e.g. Personally Identifiable Information (PII) or health data [9].

Text Analytics: Text analytics describes a collection of algorithms and concepts for extracting insights from a large number of texts. In most cases these texts are text files. Text files belong to semi-structured data, with some so-called metadata about the file (author, date, subject, etc.) being available that make the categorization of texts easier. The content of the text file, i.e. the text itself is unstructured. Text files are often available in large amounts within companies stemming from reports, customer communication or similar sources. The goal is to gain new information from these sources by making the unstructured text file content structured in order to be able to analyze it[10] [4]. However, text analytics on its own provides no added value: This can only be achieved by integrating the newfound insights into a decision-

[8] Spark SQL and MLlib are two components within the Spark architecture. The first is responsible for querying and processing static data. MLlib is a library for Machine Learning containing models, data labeling and many more. Data for training the models can be extracted using Spark SQL and then pipelined to MLlib where algorithms can be applied to it.
[9] RAM: Random Access Memory, it is needed for performing operations in-memory.
[10] Additionally text content can be retrieved from image files using OCR (Optical Character Recognition). This is particularly useful for text analytics in claims automation (see the respective Use Case in chapter 3).

making process within an enterprise application or another Big Data system. Text analytics is strongly about segmenting texts: From letters and punctuation marks to sentences and paragraphs. Developing a structure within texts requires the usage of semantic resources like taxonomies, thesauri and dictionary, which provide the data used to train a text analytics system. Depending on the goal of the system it can also be trained with text files it will later be analyzing in future, for instance insurance claims or doctor's remarks on an attestation. Furthermore techniques for analyzing the texts are needed with the algorithms applied here often coming from the field of ML [26]. The terms text analytics and text mining are often used synonymously, but actually, in spite of some overlaps, text mining is rather about the algorithms used for analyzing the documents. Text analytics covers much more, e.g. extracting and preprocessing the text documents but also visualizing the results. So text mining can be regarded only as a part of the entire text analytics process [26].
Text analytics can be used in various scenarios. Besides categorizing and clustering text files it is also possible to extract the main concepts from a text and summarize it by analyzing its content. Thanks to this, entities like customers, products, activities or any other information that can be turned into data and that can be analyzed can be extracted from a text. Text analytics is also the basis for sentiment analysis, however, this requires complex algorithms for analyzing the context of words used in a text [4, 26].

Sentiment Analysis: Often named a subfield of text analytics, sentiment analysis has the aim to classify the sentiment or attitude within a text or spoken speech. The latter is made possible through converting voice to text data [26, 27]. Simple classifications can contain categories for describing polarities like e.g. positive, neutral or negative. More complex ones have more stages, e.g. a number scale from 1 to 10 or a set of types created for one's own purposes. The more stages in a classification there are, the more difficult it gets to train the models but the more precise the sentiment analysis becomes [28]. Just like text analytics, sentiment analysis applies ML algorithms and concepts from computational linguistics for analyzing the data.
It can be used for many different purposes within a company, for instance to track the attitude towards a product, a service or the entire company in social media and react respectively if needed. This can be particularly important for customer analytics since 90% of customers rely on customer reviews compared to only 14% relying on advertising when buying a new product [4, 27]. In a customer support department it can be used to prioritize tasks: If a customer is already outrageous then his request will be handled with highest priority in order not to lose the customer. In another example researches from the Penn University used sentiment analysis based on Twitter data for scrutinizing the attitude towards newly introduced vaccines [29].

2.1.4 Big Data in Insurance

Insurance companies have traditionally been data processors using historical data to price risks. However, the data amounts underwriters and actuaries were working with so far, are quite small compared to what has appeared now with the upcoming of Big Data. Big Data has the potential to transform the insurance sector entirely. Within a company almost all departments are affected by it: In the front office marketing and product development will change. In the back office pricing, underwriting and risk assessment will be conducted in a new way

using data from various new sources. And finally in claims management a much faster and almost fully automated claim settlement is not far away [2, 40]. The more data someone has, the better he can price risks associated with a potential customer since the risks become more individualized due to the data available about a customer. This opens lots of business opportunities for US internet giants like Google or Amazon that have large amounts of customer data and customer interactions. They now could offer insurances to low risk customers they could easily identify based on the analysis of their customer data. But it is not only internet giants or insurtechs[11] who could enter the insurance market: The food-retail company Tesco already offers house insurance to its customers in the UK [39].

Most insurance companies are highly interested when it comes to applying Big Data in their businesses, nevertheless most have not yet implemented projects that go beyond a PoC or development stage. In a survey conducted by BearingPoint, a consultancy, less than a quarter of all insurers questioned said they were beyond this stage and only 10 % had a company-wide Big Data strategy [39]. Since the advantages that can be reaped through Big Data are enormous, insurers should do more to roll out Big Data applications organization or business-line wide.

Besides the pricing advantages a better customer experience is possible using Big Data. AIG, Aviva and Prudential, all large-scale insurance companies, analyze credit card and sales data in order to detect a correlation between them and the results of health checks. The underlying models proved to work and besides the fact that the customer does not have to go to various health checks the insurers can save money: A data analysis costs five dollars compared to 125 for a health check involving blood and urine samples [38]. Furthermore the automation of internal processes, a better understanding of the customer and improved fraud detection can be achieved by using Big Data (for a detailed overview of possible Big Data Use Cases in insurance, please refer to chapter 3). In a joint study Google and Bain & Company reckon that the German P&C insurers can grow by 4 billion through additional premiums while cutting their operating costs by another 14 billion in total [1].

One of the main challenges when talking about Big Data in insurance is the lack of data. Although they have silos full of customer, health and property data, there are not enough customer interactions that could generate new data. The reason is the nature of an insurance: After completing an underwriting and risk assessment process where the customer passes data to the insurer, they have no intersections until a claim situation appears. And whilst in health and P&C insurance there are several interactions at least in claims settlement, in life insurance there are almost none. Therefore, insurance companies have to identify new data sources. These can reach from services offered – e.g. wearable devices in health insurance – or be external data bought from providers in order to invigorate risk assessment and pricing. A very important issue here is to act first in order to acquire good partners and negotiate reasonable contracts with them since the one who owns the data, also has the best opportunities in a data-

[11] Insurtech: A combination of the words insurance and technology, it is used for describing start-ups offering technology-based products in the insurance sector.

driven environment. The ones who act early outperform their responsive competitors by two percentage points on revenue and EBIT[12] growth [2].
Yet another reason for the poor performance of insurance companies in Big Data so far is the lack of knowhow in this field. In the BearingPoint survey mentioned before, 16% of the companies questioned said they don't know enough about Big Data and 53% said, they had experienced difficulties when hiring new people having the required knowledge. One consequence is that due to the lack of knowledge about Big Data, decision makers within insurance companies are not willing to try out something new using it: only 35% are ready to implement any innovative applications associated with Big Data [39]. One of the most important ramifications of the lack of knowledge is however the approach to Big Data projects. With only a small minority having a Big Data strategy, most insurers have passed the responsibility for Big Data projects entirely into the hands of their IT departments [39]. This causes an almost complete absence of business aspects in Big Data projects what inevitably leads to the outcome that eventually projects do not generate the expected added value and thus fail, in spite of the fact that they have a good technological foundation. Finally customers are often unwilling to share their data with an insurance company, particularly in a country like Germany, where data privacy is regarded as very important (see also figure 2.1). Especially when it comes to sensitive data like for instance health data (only a third are willing to share) or installing cameras at home (less than half are willing to do this) customers are very restrained when asked to share these data with their insurer. Nevertheless with time the readiness rises – the key point here is making a compelling offering to the customer, so that he can see the added value for himself when sharing his data [41]. An example for such an offering could be Google in the search engine sector: because it has one of the most precise search engines and offers many additional services for free, its market share in 2016 in Germany was 94.5% and 89% worldwide [42].
Although there are many different approaches on how to deal with Big Data in the insurance sector, a selection of recommendations from leading consultancies includes the following [1, 2, 39]:

- **Focus on the Customer:** Big Data offers various possibilities for analyzing what the customer really wants and insurers definitely should use them. Since customers can nowadays easily compare product offers from different insurance companies, customer retention, providing a good customer experience and excellent products will be of gargantuan importance for each insurer.

- **Partnerships:** In the digital economy scale and network effects can give a great boost to a business. By entering strategic partnerships with other companies (e.g. car manufacturers or health service providers), insurers can build up far-reaching networks for offering additional services to their customers and getting more data about them. This external data can help solving the problem of the lack of own data mentioned before.

[12] EBIT: Earnings before interests and taxes; a financial ratio used to calculate a company's revenue.

- **Technological Excellence:** Although it is heavily advised to involve business aspects when implementing Big Data projects, an excellent technological foundation remains very important. In order to remain ahead of competitors, insurers have to lead technological innovations. This can be achieved by launching PoC versions of new products built using newly upcoming technologies. This of course requires having experts in Data Science and Data Engineering in-house, for example in form a Research & Development unit.

- **Speed and Agility:** This envisages both moving quickly and being able to adapt to changes. As mentioned before, the insurers who will start implementing Big Data projects and building a partnership network first will be able to get the biggest share of the pie. Being agile is a guarantee for being able to react quickly to market changes, both in business and technology. However, in many companies this will require a change in the culture, as decision-making and development will have to become faster. The architectures will have to become more flexible so that technologies and products can be replaced quickly and easily.

If insurance companies do take these steps, they can give rise to great opportunities for their businesses but if not, they risk to become laggards and eventually be wiped out by their more innovative competitors. In the following chapters, possible insurance Use Cases in Big Data along with the requirements it takes to implement them are outlined.

2.1.5 Big Data and Privacy

Depending on the country and region, data protection is regarded as more or less important. Whilst in the US customers are ready to share their data with a company, in Germany on the contrary customers are rather unwilling to do this (see figure 2.1). This cultural difference can also be seen in regulation: metadata has to be secured in the EU but not in US [47]. Particularly in Germany data privacy is regarded as vital – with the BDSG[13] being the main German law on data privacy regulation. Big Data, which is based on collecting a large amount of (often customer) data, makes it possible to create very precise customer profiles. Although they are very helpful to companies, it is questionable whether using Big Data does comply with the principle of purpose-relation. The BDSG states that only as few PII as possible is allowed to be collected, stored and processed [48]. According to data privacy activists, collecting large amounts of data as needed for Big Data and applying prediction algorithms to them contradicts this principle [49].
So far a number of different laws have regulated data protection in Germany and the EU leading to a judicially unclear and complex situation. In May 2018 a new law for regulating data privacy in the entire EU will come into effect; the so-called General Data Protection Regulation (GDPR). GDPR imposes a number of new laws that have far-reaching consequences for applying Big Data in a company. Since violations can result in fines up to four percent of the

[13] BDSG – Bundesdatenschutzgesetz

previous year's turnover, GDPR has to be taken seriously. First of all it has to be said that GDPR is only relevant if PII (Personally Identifiable Information) is involved. As already in force in the BDSG, PII can only be stored and processed with the consent from the customer and also following the principle of purpose-relation. What is new however, is the right of the customer not to be the subject of a fully automated decision leading to a "legal implication" for him (e.g. in a credit loan process or a claim settlement). This means a human employee has to check the results provided by a Big Data system. In case a customer's application gets rejected, he has the right to get an explanation of the decision and also the right to challenge it [50]. This brings up the question what does explaining an algorithmic decision actually mean hence e.g. the way how a model based on neural networks has calculated its outcome are almost impossible to understand. Above all the customer has the right to get all PII data collected about him from a company in a "structured, commonly used and machine-readable format" so that he can transfer it to another company. This is called the data portability – a right that could lead to more competition as well as better products and services for the customers [50]. Finally, GDPR envisages the right to be forgotten: if a customer withdraws his consent to processing his PII or it is not needed in relation to the purposes for which it has been collected initially, the company has to delete all data on him [51]. In fact GDPR brings up limitations to Big Data and raises several questions, especially when it comes to using ML. If a customer switches a company and takes his data with him or asks for deleting the data saved about him the data nevertheless stays within the "decision rules" a ML algorithm uses and was trained on. A possible solution could be retraining the models without the customer's data, although this is very costly and rather infeasible. Nevertheless GDPR also opens opportunities for business and creates a mutual basis for data protection in the EU thus making it easier for companies to operate concerning data privacy regulation.

As data privacy is such an important issue and the cornerstone for building trust, it will be addressed both in the requirements analysis phase and afterwards in the design of the new Big Data Reference Architecture for the insurance sector.

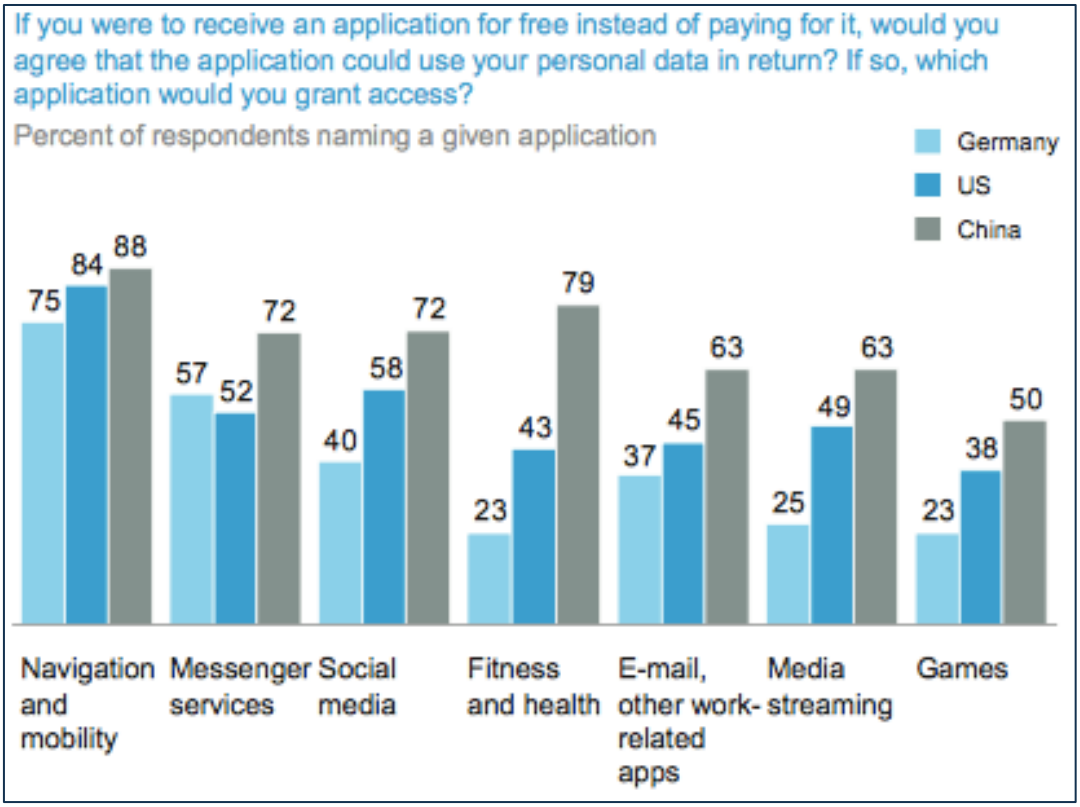


Figure 2.1: Customers' readiness to share their navigation data with companies.
Source: Car Data: Paving the way to value-creating mobility. Page 7. McKinsey & Company - Advanced Industries, Nr. 03, 2016.

2.2 Requirements Engineering

2.2.1 Common Process

Requirements Engineering is regarded as one of the most important steps in software engineering and takes about 30% of project time [43]. When done properly, it can provide a good foundation for the system design and development as the functionality and components needed for the system become clear during the Requirements Engineering process.
First it is important to understand what a requirement is. Generally spoken, it is the ability or property of the system requested by a stakeholder for solving a problem, achieving an aim or fulfilling a contract [43]. A stakeholder can be a user of the developed system or a customer who is buying a product that is supported by the system. Since requirements can be the basis of a contract between the software provider and customer, they have to be described precisely and unambiguously. On the other hand it should be possible to adjust them later during the project in case the circumstances do change, so the requirements should also be flexible [43, 45]. Requirements Engineering envisages defining the requirements that need to be covered in order to build a software system. It consists of two main activities [44]:

- **Requirements Elicitation:** Here the requirements for the system are identified using terms understood by customers and users. The elicitation takes place in close collaboration with the users and customers, often involving interviews and workshops in order to understand their needs. The requirements are then reconciled with the different stakeholders, analyzed for their feasibility and consolidated before being written down for documentation purposes. The final result of this process step is the Requirements Specification document that describes the purpose, functionality and environment of the system to be developed on a high-level using abstractions. Its main goal is to show what the system does and not how it does it.

- **Analysis (Modeling):** Here the first steps of the system design take place. The system to be developed is described using a semi-formal modeling language, e.g. UML or BPMN. The result is a so-called technical specification that is used as the foundation for the development by business analysts and programmers and already contains detailed descriptions. After it comes the system decomposition and the detailed design of the system.

Requirements can be classified in two categories [44, 45]:

- **Functional**: As the name already tells, a functional requirement describes the functionality of a system. The functionality is basically what the system does do if a user interacts with it through a user interface, for example the execution of a sequence of functions. It also covers the relationship between inputs and outputs in the system. A functional requirement can also state what the system shouldn't do in specific/abnormal situations and is formulated independently from the actual implementation.

- **Non-Functional:** In most cases these do refer to the system as a whole instead of one single feature in it and thus have a great impact for the system's architecture. Non-functional requirements deal with the three following topics: performance, quality requirements and constraints. In performance requirements the system's speed, recovery and response times and the availability are described. When it comes to a distributed Big Data system, they are very important hence they are relevant for the system's stability. Issues like the number of simultaneous users or the amount of data processed are clarified here. Quality requirements include aspects like adaptability, maintainability and security. Constraints – or pseudo-requirements – are additional limitations to the system design brought up either through compliance or through the customer. This can be for instance the requirement to implement the system in a specific programming language or the need to comply with regulatory standards. Furthermore, organizational requirements have to be considered when eliciting and analyzing non-functional requirements.

Before starting an implementation and particularly before rolling out the system, it makes sense to validate them with the customer, as the cost of solving a requirements error after a roll-out are up to 100 times more than fixing an implementation error.
Finally, it is vital to document the requirements after their elicitation and reconciliation. This is helpful for the customer and the provider of the system as well – especially managers, analysts, developers and testers can use it to get a better understanding of the desired system. Whilst there is a common understanding of what a Requirements Specification document should include (refer to Requirements Elicitation explained before), the description of single requirements can vary depending on the goal the respective project and system. Often templates are used for describing single requirements in order to have a comparable and structured visualization of them [44]. In the following section a template to describe requirements for a system implementing Big Data Use Cases will be introduced.

2.2.2 Operationalizing Use Cases Based on Requirements

Requirements Engineering lays the foundation for a proper system development. For having a profound Requirements Engineering process it is needed to first elicit the requirements and then document them using a template. But how is this all related to Big Data? When developing a Big Data system it is essential to understand both the business and the technical background of the Use Case to be implemented. Therefore it is needed to analyze requirements that have to be met before a Use Case can be operationalized.
In 2012 the White House called for a Big Data Research and Development project that led to the setup of a workgroup under the guidance of the US National Institute of Standards and Technologies (NIST) – the so-called NIST Big Data Public Working Group. It is made up of various participants from the industry, universities and governmental institutions. The goal of the workgroup was to develop a set of cross-industrial Big Data artifacts, including definitions, security standards and finally a Big Data Reference Architecture [7]. As the aim of this thesis is to design a Big Data Reference Architecture for the insurance sector, the approach chosen by the NIST workgroup is highly interesting. For developing their Reference Archi-

tecture, the NIST workgroup started out by collecting a number of cross-industrial Use Cases – in total 51 of them. The Use Cases covered many areas from government and defense over commercial and social media to life sciences and physics. Each Use Case was then analyzed concerning the requirements needed to implement it. The Use Case specific requirements were afterwards scrutinized for deriving generic, cross Use Case requirements from them and finally map these generic requirements to components in the Reference Architecture [6].
A similar approach is also applied in this thesis. In chapter three several Big Data Use Cases are presented for the insurance sector whilst chapter four outlines the requirements for these Use Cases and chapter five introduces a Reference Architecture for the insurance sector based on the requirements. In order to elicit the requirements for the Use Cases, expert interviews and a literature analysis were conducted. This ensures that the Reference Architecture developed addresses insurance Use Cases and is tailored to their needs. The requirements are described using a template that is based on a template the NIST workgroup used [6] and has been modified slightly according to the NIST Use Case description template [46]. Furthermore, a few categories were added that resulted from questions asked in the expert interviews for evaluating the Use Cases in this thesis (for the Use Case evaluation methodology please refer to section 3.3.). The template is made up as follows:

1. **Use Case Title**
2. **Use Case Description**
3. **Big Data Characteristics**
    1. Data Sources: The data sources used for delivering the data for operationalizing a Use Case are listed here. However, they are not limited to all data sources that could be used for a Use Case. For some Use Cases not all sources named here have to be necessarily used in order to implement the core Use Case.
    2. Volume: States how high the amount of data is for each Use Case. Since a detailed analysis of this was not possible within the scope of this thesis, the volume category is limited to a qualitative comparison of data volumes between the Use Cases.
    3. Velocity: States whether data is streamed or batch-loaded from the data sources.
    4. Variety: Depending on the amount of sources, it can be needed to integrate the data within a data lake. Several Use Cases can then use the data lake in order to access the data needed for the respective analytical models.
4. **Big Data Science**
    1. Veracity and Data Quality: Describes the quality of the data coming from the input sources. In case the data has low quality, cleansing mechanisms need to be applied here.
    2. Presentation & Operationalization: Explains how and to whom the results of the data processing need to be presented. It also covers possible scenarios for post-processing the results in other applications or systems.
    3. Data Types: States which types of data are used in the Use Case: structured, semi-structured or unstructured.

4. Data Analytics: The analytical models and frameworks required for analyzing the data are described here (e.g. ML algorithms). It is also noted whether real-time or batch-oriented data processing have to be used.

5. **Security and Privacy**
    1. Personally Identifiable Information (PII) used?: States whether PII is required for operationalizing the Use Case.
    2. Highly Sensitive Data Used?: States whether any sensitive data (besides PII, e.g. anonymized health data) is needed for the Use Case. The consequence is often the need for high IT-Security and I&AM standards within the system.
    3. Governance, Compliance & Audit: The NIST workgroup template refers to governance as a process for ensuring high data quality and assigning the responsibility for it to the respective department. In this thesis governance is used together with compliance and describes which processes have to be set up in order to ensure compliance with regulatory standards (e.g. GDPR or other data protection laws).
6. **Organizational Requirements**
    1. Knowhow: This part describes whether additional knowhow is needed in the insurance to operationalize the Use Case. For intellectual property and non-disclosure reasons, the respective results will be published only inside the insurance company where the interviews have been conducted.
    2. External Partners: If an external partner is needed for operationalizing the Use Case, he and his role is mentioned here. A cooperation partner can act as a provider of external data or be the manufacturer of a sensor device that is needed to collect data.
7. **Other Big Data Challenges:** Any specific challenges and requirements not covered in the sections before are written down here. Additionally this part is used for summarizing the most important issues from the sections before.

Figure 2.2 shows the requirements template as an Excel-spreadsheet for getting a better overview of it.

Use Case Title		
Description		
Big Data Characteristics	Data Sources	
	Volume	
	Velocity	
	Variety	
Big Data Science	Veracity and Data Quality	
	Presentation and Operationalization	
	Data Types	
	Data Analytics	

Security and Privacy	Personally Identifiable Information (PII) used?	
	Highly sensitive data used?	
	Governance & Compliance	
	Audit requirements	
Organizational/Business Requirements	Knowhow	
	External Partners	
Other Big Data Challenges		

Figure 2.2: Big Data Requirements Template
Source: Own depiction

2.3 Reference Architecture

2.3.1 Definition

Although defined differently, most people agree on the fact that IT-architecture plays a major role in software design and development. Before turning to the Reference Architecture it is needed to understand what IT-architecture actually is. According to the IEEE[14], architecture is "the fundamental organization of a system embodied in its components, their relationship to each other, and to the environment, and the principles guiding its design and evolution" [52]. TOGAF[15], an architecture framework, offers the following two definitions, depending on the usage context: "(1) [Architecture is a] formal description of a system, or a detailed plan of the system at component level, to guide its implementation. (2) [Architecture is] the structure of components, their inter-relationships, and the principles and guidelines governing their design and evolution over time" [53]. So generally spoken architecture is a collection of artifacts that describe the set-up of a system by defining the relationships between them. An artifact can for example be an architectural component (e.g. a data lake in a Big Data context) or a guideline that governances the design of a system. Eventually IT-architecture creates the foundation for a structured system design and development.
However, architecture is not always the same – several types of architectures do exist that vary depending on the usage context. TOGAF names the following three main architecture fields[16] [54]:

- **Business Architecture:** Here the business strategy, core business processes, business capabilities and the organizational structure are defined. The interrelation between all the artifacts named is described in this section – as are the high-level business requirements. Business Architecture is closely related to other business topics such as enterprise planning or business product development and has to be reconciled with them. It is also the precondition and baseline for designing the other architectures.

- **Information System Architecture:** In some cases also called Solution Architecture, it is made up of two parts:
    - Data Architecture: All data objects, entities and their relationships needed for implementing the business processes and functions that were defined in the Business Architecture are listed here. A deeper view provides also the attributes for each data entity. Aspects like data integration, transformation, quality, migration (in case an existing application will be replaced with a new one), storage and governance are addressed in the Data Architecture as well.

[14] IEEE: Institute of Electrical and Electronics Engineers – a professional association that publishes trade journals and develops standards in the area of Electronics and Information Technology.
[15] TOGAF: The Open Group Architecture Framework.
[16] The architecture subcategories listed here are part of the Architecture Development Method (ADM), TOGAF presented for managing IT-architecture in an enterprise. Besides them, ADM also includes aspects like e.g. implementation guidelines or architectural change management.

- Application Architecture: Here the applications and (sub-) systems required for enabling the business processes are described thus giving an overview over the application landscape. Furthermore, it includes the description of the application's relationships to each other. From a more detailed point of view, especially concerning (sub-) systems, the interfaces between them are covered by the Application Architecture, too. The Application Architecture relies upon the data objects identified in the Data Architecture for designing the entire software architecture.

- **Technology Architecture:** Sometimes also called technical architecture, this part of architecture deals with logical and physical infrastructure issues, including the infrastructure landscape, its setup and operation. Above all the server nodes are mapped to the applications that run on them and the network protocols for the communication between systems and nodes are defined.

Another term worth mentioning is Enterprise Architecture: it provides a holistic, high-level overview over all the other architectures explained previously by elaborating enterprise-level standards, designs and guidelines. According to Gartner Enterprise Architecture is an "enterprise planning process that translates an enterprise's business vision and strategy into effective enterprise change" [55]. Given this definition, one sees that Enterprise Architecture can be defined quite broadly, just as most other terms in IT architecture.
Finally the term Reference Architecture has also no unique definition. Seen from a more technical point of view, a Reference Architecture is a set of patterns designed for being used in a specific technical or business context and supported by a further set of artifacts. Other, more business-oriented definitions take also business goals and organizational aspects into account [56]. Based on this it can be said that a Reference Architecture should present a standardized solution consisting of functional architectural components for a specific domain or industry problem. The components should cover both technological issues on an abstract level (e.g. data integration or data processing) and business requirements (e.g. for complying with regulatory standards). In a Big Data context a Reference Architecture should provide a conceptual standard or model that can be used as baseline or blueprint for developing Big Data systems. This standard should contain all components required for building a Big Data system in the respective domain whilst ensuring that the components are vendor-, product- and technology-neutral. This means in case of a real-time data processing component the component will not be named Apache Spark but e.g. "real-time processing engine". Furthermore the Reference Architecture should introduce some common terms understandable for stakeholders both from business and technology [7]. Since the Reference Architecture should cover business needs, the components should be functionally motivated. Finally security and data privacy issues have to be addressed by including the respective components into the Reference Architecture.

2.3.2 Analysis of Existing Big Data Reference Architectures

When the NIST workgroup developed its new Big Data Reference Architecture, besides analyzing Big Data Use Cases, it asked companies and academic institutions to submit their own Big Data architectures. The goal was to compare the structure of existing Big Data architectures in order to identify components that exist in most Reference Architectures. The same approach is also used in this thesis. Since the NIST workgroup already provides a good overview over common components, the result of their research is used here as well; it can be found in section 2.3.3. The Big Data architectures analyzed by the NIST workgroup were the following [8]:

1. ET Strategies
2. Microsoft
3. University of Amsterdam
4. IBM
5. Oracle
6. EMC/Pivotal
7. SAP
8. 9sight Consulting
9. LexisNexis

In this section an analysis of three more Big Data Reference Architectures from NTT Data, Google and Microsoft Azure is conducted.

NTT Data: NTT Data, a consultancy, presents a Big Data Reference Architectures where all components are grouped into three main categories: Data Platform, Analytics Platform and Management Platform. The components in these platforms in turn consist of a number of subcomponents that are briefly described here. Additionally to the platforms NTT Data's Reference Architecture also lists possible data sources including media, sensors, databases, files and social media data thus covering most types of data [86]. Figure 2.3 shows an architecture diagram where NTT Data's Reference Architecture is depicted.

- **Data Platform:**
  - **Information Gathering:** Covers all mechanisms needed for ingesting data into the platform such as streaming, messaging and ETL. Furthermore basic validations and transformations are executed here. A task scheduler for managing the available resources is in place as well.
  - **Information Store:** This layer offers technologies for storing the available data. It ranges from conventional relational databases over document storage to technologies for storing massive amounts of data such as NoSQL. For enabling fast access to data In-memory databases exist.
  - **Data Processing:** Here are the technologies for processing large amounts of data and/or processing data at high speed, e.g. through distributed parallel processing. Conventional data processing, for instance by using Rules Engines, can be found here as well as data enrichment.

- **Analytics Platform:**
  - **Data Analytics:** This layer covers the technologies required for getting insights from the data such as ML, Text and Data Mining, and Natural Language Processing. Furthermore mechanisms for simulating and optimizing the ML models are available here.
  - **Decision Support/Utilization:** Here various tools and mechanisms for presenting and visualizing the results from the data analysis are covered. Furthermore possible consumers of these results such as Business Process Management or search functions can be found here. Interestingly Online Analytical Processing (OLAP) is mentioned as a single element in this layer, although the complete OLAP process covers many other components such as Information Gathering, Data Processing and Data Analytics.

- **Management Platform:**
  - **Governance:** All processes for data lifecycle management, ensuring high data quality and data audits can be found here. Furthermore the highly important aspect of IT Security is covered here.
  - **Infrastructure:** Any tools or technologies required for operating the components from the Big Data Reference Architecture are covered here. The goal of this layer is ensuring scalability, performance and availability.

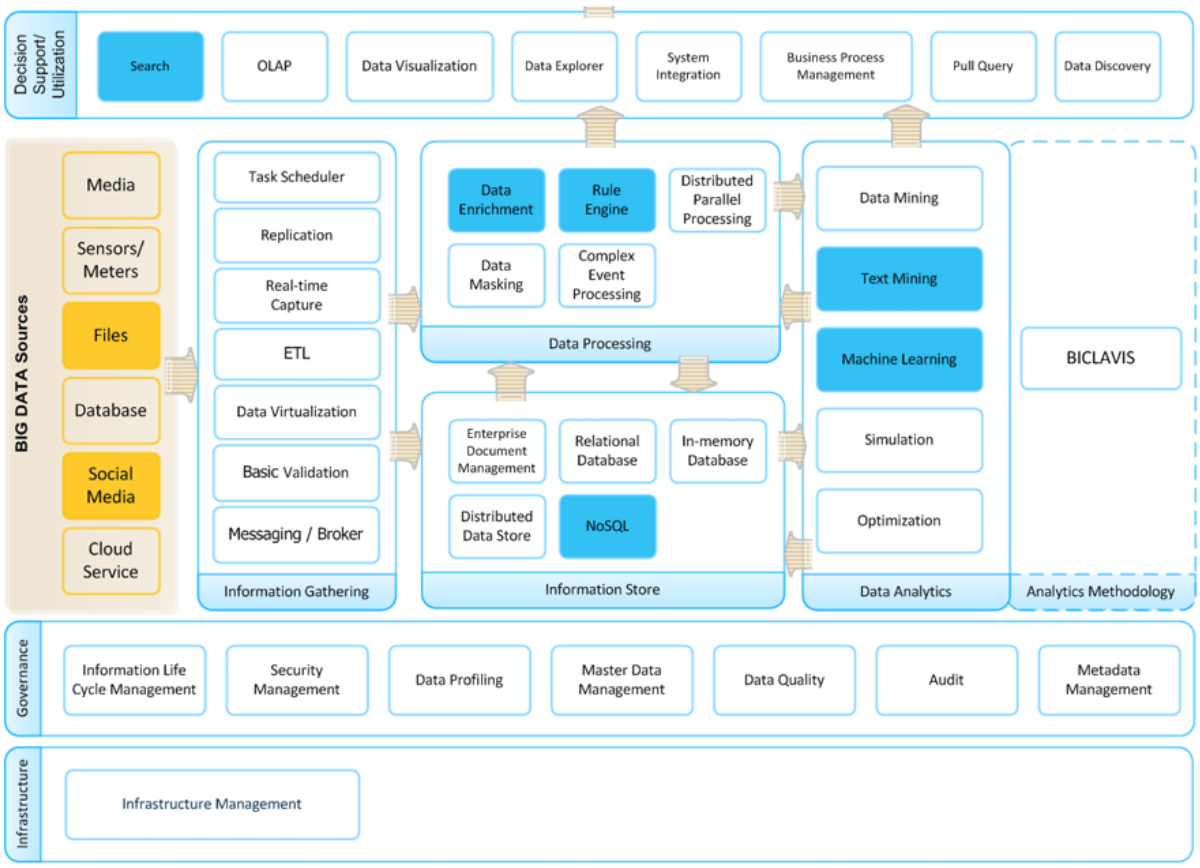


Figure 2.3: NTT Data's Big Data Reference Architecture
Source: Page 3, [86]

Google: Google's Big Data Reference Architecture (or Google Cloud Platform Architecture - GCP) consists of four main stages: Ingest, Store, Process & Analyze, and Explore & Visualize. The data passes these four stages from left to right. Although no infrastructure, governance or security components are shown in the provided architecture diagram (figure 2.4), GCP of course offers all these components [87]. Next, an overview over the components within GCP is given, although not each of them is described here in detail.

- **Ingest:** It provides tools for collecting any incoming data and pipelining it to the Storage layer. Specifically, GCP provides tools for three possible sources: application data, streaming data and batch data. Application data can come from click-streams or transactions as well as application monitoring systems and will pass Stackdriver Logging (a component designed for log transfer). All this data can either be streamed or batch-loaded to the target storage or processing system, depending on the latency required for the respective Use Case. Streaming can be needed e.g. for pipelining sensor or clickstream data by using Cloud Pub/Sub, a real-time messaging service offered by GCP. Bulk data eventually can be loaded as a batch from the respective source, be it a relational or NoSQL database, or another existing cloud storage e.g. at Amazon Web Services (AWS). Here GCP also provides a number of tools, e.g. Cloud Transfer Service or Transfer Appliance [87].

- **Storage:** In this layer GCP provides a number of storage technologies for keeping the data that came in during the Ingestion phase. With Cloud Storage GCP has got an object (file) storage for both structured and unstructured data. It can store data from ETL processes or media and is integrated with many other GCP components, like e.g. ML APIs. Cloud Storage guarantees fast access to data and can be configured for both frequent or seldom data access. When it comes to database storage, GCP covers both relational and NoSQL databases. Cloud SQL makes it possible to store any data intended for a conventional database such as Online Transactional Processing (OLTP) data or customer data. With Cloud Spanner there is now a relational database that guarantees not only ACID consistency but is also highly scalable for large loads of data. Cloud Datastore is a document database with a flexible, yet structured data schema ensuring high scalability. Cloud Bigtable is a column-oriented NoSQL database that was designed for guaranteeing high throughput and low latency for large datasets. It is comparable to other NoSQL databases HBase or Cassandra and can be used for storing IoT sensor data, real-time application and streaming data. Finally GCP offers also an analytical database – BigQuery. It can be seen as a Data Warehouse for Big Data that makes it possible to execute both real-time and conventional analytics on data stored there. If necessary – for instance in a velocity application – data can be pipelined directly to Big Query for being analyzed there. BigQuery is also a possible solution for OLAP tasks [87].

- **Process & Analyze:** Simply storing data is not enough – it has to be analyzed for deriving insights from it so that the company's business can benefit from it. Again GCP offers a number of tools that can be used for processing and analyzing data. When it

comes to analyzing large datasets, distributed processing clusters are required. By using Cloud Dataproc, companies can move their existing Hadoop or Spark clusters to a service that administers and monitors these clusters and above all enables integration with other GCP components, e.g. from the Storage layer. Cloud Dataflow is a service for optimizing both stream and batch-oriented processing tasks by offering on-demand resources instead of having a predefined cluster size. One can use Apache Beam (a programming model) for writing ones own batch and streaming data processing pipelines and then deploy them to Cloud Dataflow. It is also integrated through connectors with various products from the Storage layer like Bigtable or Cloud Storage [88]. Since BigQuery is an OLAP solution it has to ensure not only the respective storage functionalities but also the analytical and querying capabilities needed for Business Intelligence or real-time analytics. Finally ML is a very important aspect in GCP's Process & Analyze layer. Here GCP provides two things: the first is a legion of ML APIs for proven models that have already been trained by Google. These API's include text analytics (Cloud Natural Language API), image processing and recognition (Cloud Vision API), audio data analysis, video recordings analysis (Cloud Video Intelligence API) and language translation. The second thing is Google Cloud Machine Learning. It can be applied for executing and training models a company has developed on its own by using TensorFlow, Google's ML framework. TensorFlow offers a wide range of ML algorithms including Deep Learning and is highly scalable as well. After a preprocessing stage, TensorFlow models are converted into models that can run on Cloud Machine Learning (also called graph building) where they are afterwards trained on a large dataset. Finally when a model has been trained it is used for making predictions [88]. Above all it is also possible to deploy ML models on GCP that were developed with other ML tools such as MLlib from Apache Spark.

- **Explore & Visualize:** In this layer GCP provides several tools for visualizing the results of the calculations from the Process & Analyze layer. With Cloud Datalab data scientists can explore datasets by executing test models (that were developed for example with TensorFlow) and see their outcomes directly – this approach is also called sandboxing). Cloud Datalab is intended for visualizing the results and the data from data science calculations and analyzes. For visualizing data from any kinds of Business Intelligence analyses Cloud Data Studio is available, where it is possible to create drag-and-drop dashboards, reports or other visualizations. When integrated with BigQuery data has not to be imported to Data Studio but can be accessed directly thus enabling a visualization of real-time analytics. Above all BigQuery can be integrated with external providers of data visualization tools like e.g. Tableau [87].

An aspect that is not depicted in the architecture diagrams but nevertheless is highly important is coordinating tasks executed by GCP's various components through resource management – Google calls this orchestration. Orchestration also includes monitoring the entire platform for stopping tasks if needed or triggering new workflows [87]. Finally figure 2.5 presents a possible data pipeline (with a slightly altered data flow concerning the layers) built from some GCP components that are shown in figure 2.4.

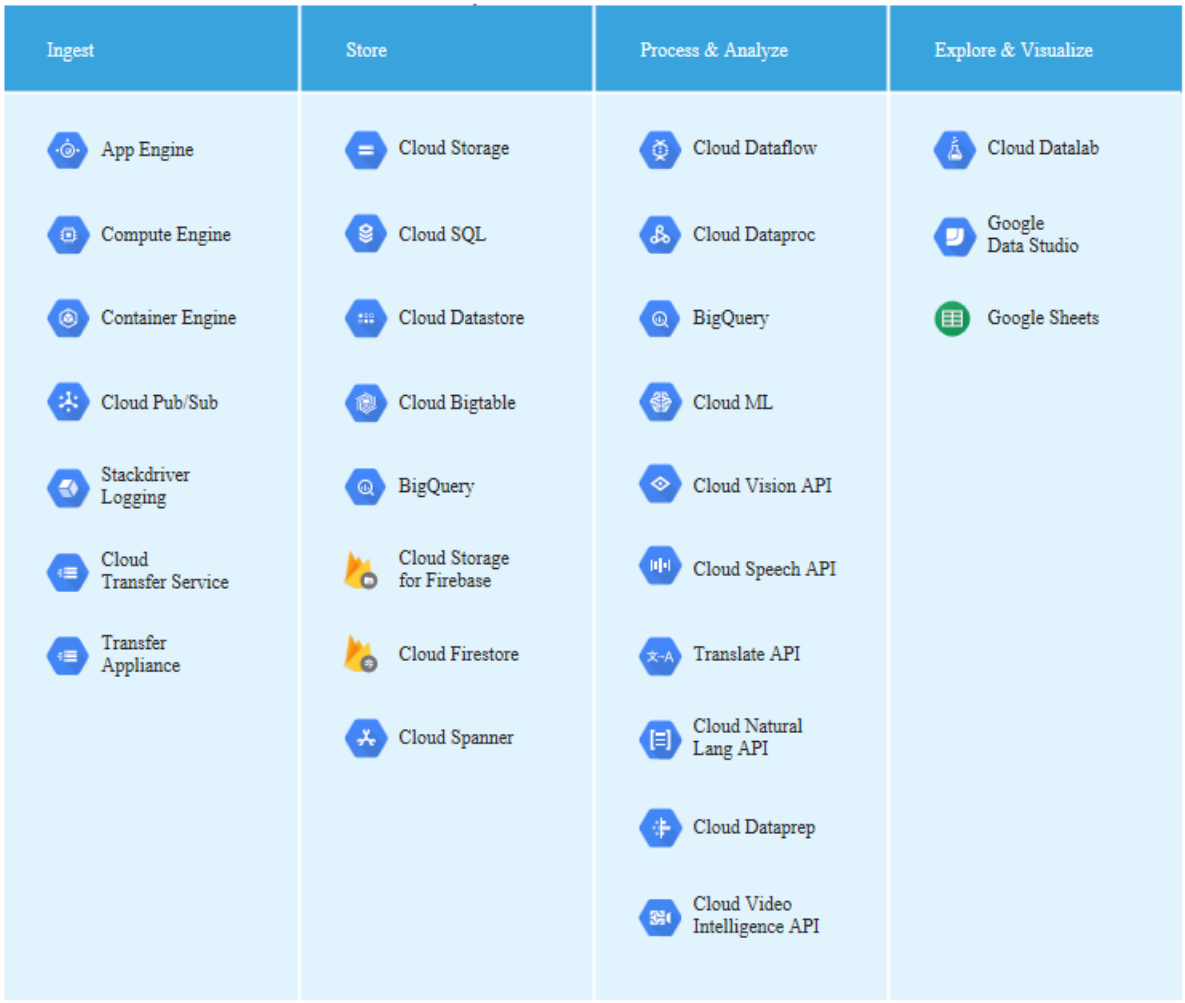


Figure 2.4: Google's Big Data Reference Architecture (Google Cloud Platform)
Source: [87]

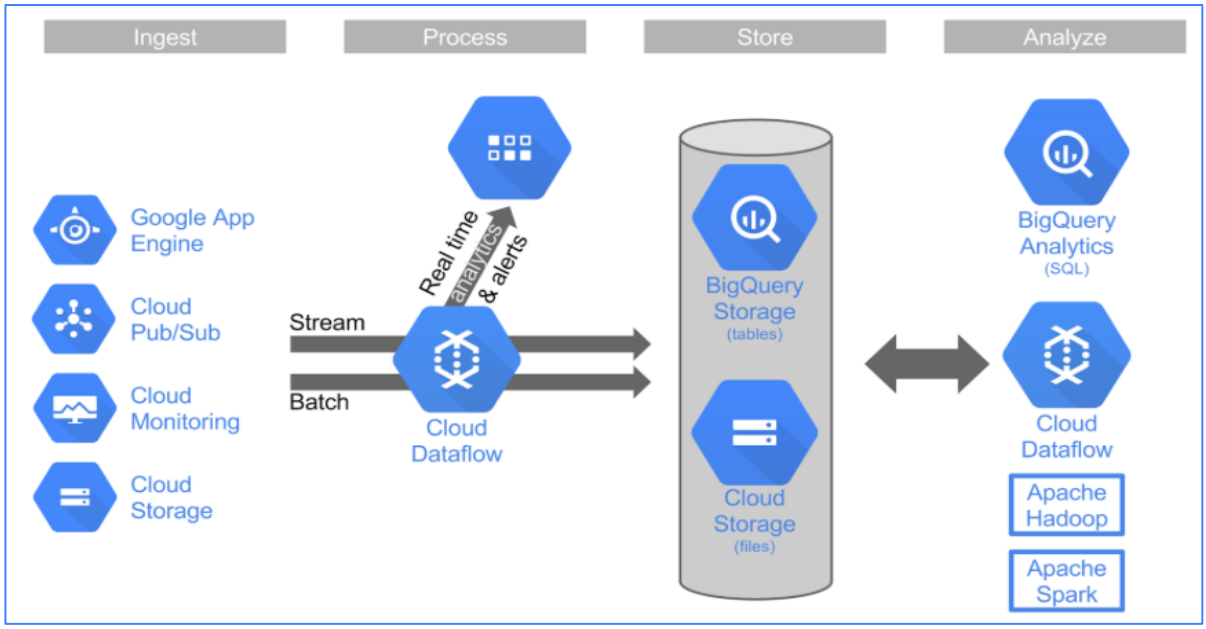


Figure 2.5: Google's Big Data Reference Architecture in action
Source: [89]

Microsoft Azure: The architecture described here is different from Microsoft's architecture analyzed by the NIST workgroup as it shows which technologies Microsoft Azure does offer for a Big Data architecture as of today (figure 2.6). Like the architecture diagram showing the components in GCP (figure 2.4) Microsoft's architecture presents a number of technologies that can be used in the different phases of a data pipeline. Depending on the Use Case or project that has to be implemented, the required components can be selected and put into a solution architecture. Microsoft Azure is able to take in data from many different sources, such as sensors, transactional or CRM systems, social media and so on. A cross-layer component is Data Factory, an orchestration tool that takes care of coordinating task execution by single components (when they process data workloads) and distributes existing resources amongst them. It is also worth noting that the Streaming and Batch layers introduce technologies for both ingesting the data and also processing it so that these two phases of a data lifecycle are somewhat intermixed here. After data comes in from one of the sources it passes the following layers (not necessarily all or in this order) [91]:

- **Streaming:** For any data that has to be loaded at (near) real-time speed into a Big Data application both Microsoft's own technologies and other open source products are available. The latter are for instance Spark Streaming or Apache Storm that can be used not only for pipelining but also for processing incoming data (please refer to section 2.1.3 for a deeper explanation of them). With Azure Stream Analytics it is possible to process streaming data by applying data transformations and manipulations, and integrate it with both storage technologies and PowerBI (see Presentation layer for more details). Although it offers a more limited range of functions than other streaming technologies, Stream Analytics is far easier to implement. Event Hub and IoT Hub are simple messaging (or event) queues for streaming data into an application in the conventional sense, i.e. without processing it. Both are able to persist data for a period of eight days in case the data pipeline should break down so that no data will be lost during the time of an outage. The IoT hub supports most common protocols in the IoT area and can be used for bidirectional communication, i.e. not only for passing data from an IoT device into the pipeline but also for sending event triggers from the Big Data application back to the device [92].

- **OLTP:** Interestingly, Microsoft Azure lists OLTP with the respective storage technologies as an own layer in its architecture. Azure SQL Database is a storage technology that easily scales for large amounts of data and offers fast access to data – however it can only store data that has a strict schema and therefore is not suitable for many Big Data datasets. DocumentDB (or since May 2017 Azure Cosmos DB) is a schema-less NoSQL database that is optimized for high throughput and can guarantee ACID for database transactions. Finally with Azure HDInsight there is a managed service available where one can set up clusters for HBase or other open source Big Data technologies (also for data processing such as Apache Spark or Apache Storm). HBase is a column-oriented NoSQL database that is highly scalable and can be used for fast OLTP operations.

- **Storage:** For storing data Microsoft Azure offers Blob Storage that stores objects such as text data or media files and can be used as a data repository. Since it has some limitations when it comes to storing large amounts of data or accessing them quickly, Azure Data Lake Storage exists for solving these problems. It is based on Hadoop Distributed File System (HDFS); can be used for analytical workloads and can be easily integrated with other Big Data technologies from the Hadoop ecosystem. Any large data sets from batch or streaming, such as clickstreams or sensor data can be stored here. It is possible to use analytical applications like Apache Hive or Apache Impala (for real-time analytics) for accessing data directly from Azure Data Lake Storage [92].

- **Batch:** When it comes to batch processing, Microsoft Azure supports a wide range of open source and Microsoft's own products. In HDInsight, a managed service from Microsoft, batches can be processed using a Hadoop implementation provided by Hortonworks, a software company. Furthermore it offers support for micro-batches with Apache Spark or conventional Hadoop batches executed using Hive (a data-warehouse solution) or Pig (a platform for analyzing and transforming large datasets). Additionally Azure Data Lake, a storage solution mentioned before, provides Azure Data Lake Analytics, a service that makes it possible to query large amounts of data that has no schema by using the language U-SQL (a combination of C# and SQL). Unlike HDInsight it does not need a pre-configuration of the amount of resources needed for executing a task, so it is much easier and flexible to use. Azure SQL Data Warehouse makes it possible to integrate data with a strong schema with data having only a weak schema.

- **Analytics:** In order to analyze data, Microsoft Azure provides support for R and Apache Spark MLlib. Using R, a statistical programming language, ML models can be created and trained. Spark MLlib is a ML framework from Apache Spark that contains pre-trained models and mechanisms for tuning these models. Due to its integration with Spark SQL it can access large data sets required for training the models directly. Furthermore Microsoft offers Azure Machine Learning (or Machine Learning Studio), a managed service where own ML models can be easily developed, e.g. by using Python or R, and afterwards deployed. Above all it contains several pre-trained models for solving common problems such as text analytics or image recognition [92].

- **Presentation:** Besides visualizing data with Excel, Microsoft offers PowerBI, a tool for visualizing data graphically with dashboards, scorecards and other presentation tools. PowerBI can be integrated into business applications thus also enabling direct access to data and ad-hoc analytics. RStudio and Azure Machine Learning also offer several mechanisms for visualizing results of analytical and predictive ML models.

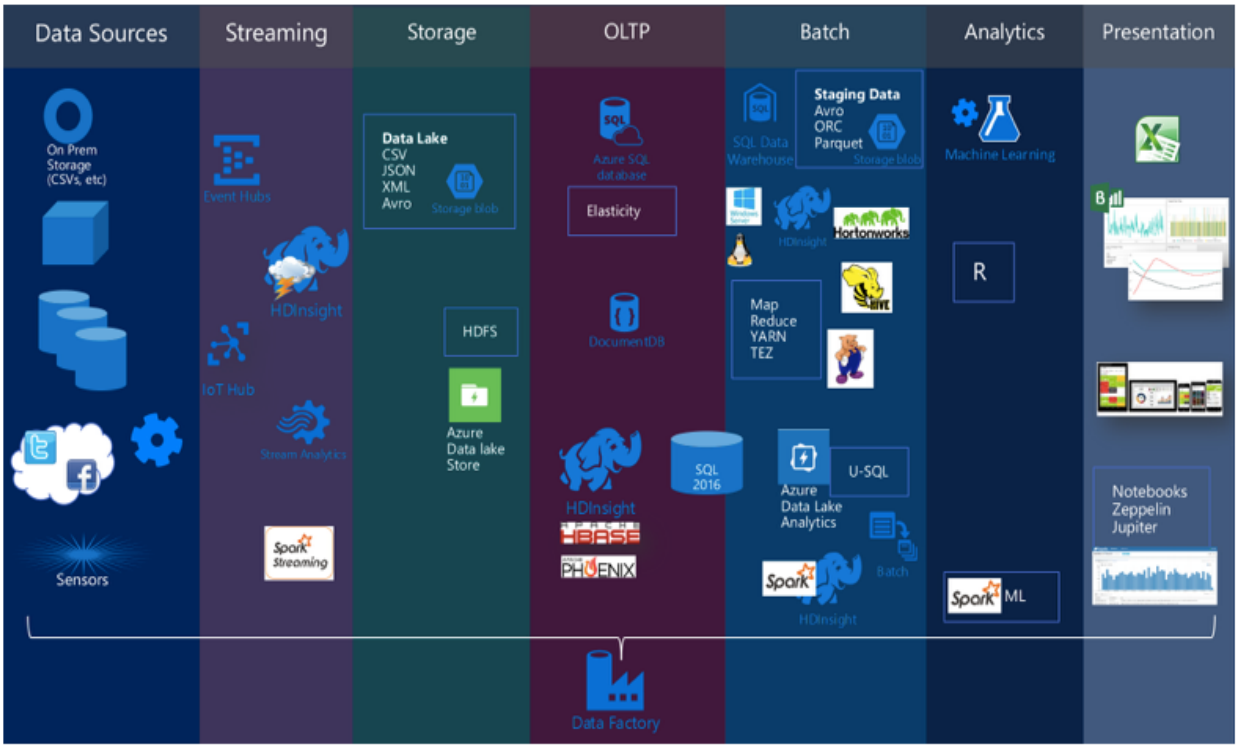


Figure 2.6: Big Data architecture in Microsoft Azure
Source: [91]

2.3.3 Deriving Common Components

After having compared nine Big Data architectures from companies and academic institutions, the NIST workgroup was able to identify three major parts a Big Data Reference architecture should have: Big Data Management and Storage, Big Data Analytics and Applications Interfaces, and Big Data Infrastructure. It is important to note that the Management and Storage layer offers technologies for dealing with both structured and unstructured data. The following table shows a selection of components that can be found in the respective layer [8].

Layer	Components
Management and Storage	• Data Sources (Legacy/ERP/CRM systems, sensor data, web logs, etc.) • Repositories (NoSQL, relational databases, in-memory databases, distributed file systems, data warehouses, etc.) • Integration (Data cleansing, transformations, extractions, etc.) • Data Processing & Discovery (Stream processing, information discovery) • Metadata Management • Auditing & Logging

	• Security & Access Control
Analytics and Applications Interfaces	• ETL & Data Mining • Analytics (Predictive, Text analytics, batch analytics, real-time analytics, ML, etc.) • Reporting & Visualization (Score-cards & Metrics, BI reporting, events and alerts, etc.) • Governance (Data quality, data profil-ing, etc.)
Infrastructure	• Infrastructure Services (Hardware, networks, infrastructure management and monitoring, etc.) • Security Infrastructure • Systems Management

Table 2.2: Common components discovered by the NIST workgroup
Source: Own depiction based on [8]

The NIST workgroup also pointed out that most Big Data architectures have components for managing and coordinating available resources required for performing computations and data processing. The analysis of additional architectures in section 2.3.2 came to the same result. Although many of the components derived by the NIST workgroup in their own comparison do appear in the additional architectures scrutinized in this thesis, several new could be identified.
NTT Data mentions data processing using a Rules Engine to cover existing business rules – a component, which is highly important for insurance companies that often have a large set of rules for various business processes. The architectures analyzed by the NIST workgroup barley cover the ingestion phase where data is loaded from sources into the Big Data platform. NTT Data, GCP and Microsoft Azure offer a wide range of tools and mechanisms for batch-loading or streaming data (e.g. GCP's Cloud Pub/Sub) from the data sources. NTT Data's architecture also includes a component for carrying out first validations on data already in the ingestion phase.
Furthermore NTT Data, Microsoft Azure and GCP (with Cloud Datalab) have components for sandboxing and exploration so that data scientists can try out and train new models in a separate area that is decoupled from production. It is also possible to analyze and optimize the performance of analytical and ML models in NTT Data's architecture so that models can be adjusted if necessary. GCP offers itself a number of ML APIs that are ready for usage in ones own models; the same can be said for Microsoft Azure. From this it can be derived that a Big Data Reference Architecture should also be able to integrate external APIs into its ML com-

ponents for enabling transfer learning and make it easy to solve standard problems like for instance image recognition.[17]
Most of the components derived by the NIST workgroup and identified in the analysis in section 2.3.2 will be used for designing the Big Data Reference Architecture in chapter five. When regarding the various architecture diagrams one can easily see that the single components are grouped into architectural layers quite differently. Therefore the newly designed Reference Architecture will not follow an exemplary architecture diagram when defining the layers – instead an own layering will be set up. In addition, the three architectures of NTT Data, GCP and Microsoft Azure will be evaluated for their capability to implement the generic requirements for the insurance sector derived in section 4.2 (please refer to section 4.3 for this).

[17] In transfer learning, Machine Learning problems are solved by using already existing models for similar problems. APIs offering access to pre-trained models can be highly helpful for this.

2.4 Related Work

In this section, papers and books are presented that address the most important issues dealt with in this thesis such as a general approach to Big Data, the design of a Big Data Reference Architecture and the description of requirements for Big Data applications.

2.4.1 Marr (2015)

In his book "Using SMART Big Data, Analytics and Metrics To Make Better Business Decisions and Improve Performance" Bernard Marr presents a holistic approach to using Big Data in enterprises. In the beginning he points out how important it is to start with strategic, business-oriented aspects instead of focusing on technology directly. One core artifact in his book is the so-called SMART strategy board that is used in this thesis as a foundation for describing the Big Data Use Cases and developing the questionnaire for evaluating them in expert interviews (please refer to sections 3.1 and 3.2 for more details on this). In the following chapters he provides a guidance on how to implement a Big Data application. In order to do this he outlines the single steps to be taken, ranging from data collection and defining metrics to applying analytical models, visualizing their outcomes and putting the insights from them into action. In each chapter he also briefly explains the most important technical terms, e.g. the different data categories available or the various types of analytical models [4].
Although he gives a good overview and an integrated approach to Big Data, the descriptions are often too superficial so that for a more profound insight other sources had to be used.

2.4.2 National Institute of Standards and Technologies (2015)

In 2013 the National Institute of Standards and Technologies (NIST) started a public Big Data workgroup made up of various contributors from industry, science and government. Their aim was to provide a set of standardized artifacts that can be used cross-industrially and product-independently. The most important artifact for this thesis was the Big Data Reference Architecture. Although the architecture itself is not very relevant for this thesis since its aim is to develop a Reference Architecture for the insurance sector, the approach the NIST workgroup used for designing their architecture is highly interesting. The workgroup started out by collecting Use Cases from several industries and then derived requirements for implementing them. These requirements were eventually mapped to architectural components in the NIST Reference Architecture [7]. Additionally the workgroup analyzed existing Big Data Reference Architectures from leading companies in the field of Big Data (e.g. Oracle, Pivotal or SAP) in order to identify components available in most Reference Architectures [8]. The same steps were taken in this thesis to design the Reference Architecture: chapters three and four focused on Use Case development and requirements elicitation and analysis from them, while in section 2.3.2 existing Reference Architectures were scrutinized.
Another noteworthy artifact developed by the NIST workgroup is a list of definitions of various terms related to Big Data (e.g. what is Big Data, NoSQL, data lifecycle and many more) that has been used for the foundations part in this thesis [21].

2.4.3 Fox et al. (2014)

Although this is actually a document developed by members of the NIST workgroup, it has to be mentioned here as a single related work item, hence it was used as the basis for answering RQ2. The paper outlines how the requirements elicited from the Use Cases were analyzed and suggests a template for describing these Use-Case specific requirements. This template has also been used in a slightly modified way to write down the requirements for the insurance Use Cases (see section 2.2.2 for the template used in this thesis). The paper then shows how the Use Case specific requirements were consolidated in order to provide a generic view on them, independently from the single Use Cases [6]. These generic requirements were afterwards mapped to components in the Reference Architecture. Since several of these generic requirements have to be met in every Big Data Reference Architecture, many of them were also found to be applicable for the design of the Big Data Reference Architecture in the insurance sector.

2.4.4 Lanquillon et al. (2015)

This chapter from the book "Practical Handbook for Big Data" presents a generic, cross-industry Big Data Reference Architecture, which is based on a number of generic, non-functional Big Data requirements. These requirements result from Big Data's V's (Volume, Velocity, Variety and Veracity) and additionally from the field of analytics. Each component of the Reference Architecture and several underlying paradigms are explained, also using exemplary products from the Hadoop ecosystem. The chapter also points out the importance of data privacy that has to be taken into account when designing Big Data architectures [57]. Both the requirements and the Reference Architecture introduced here were used when developing the own artifacts in this thesis.

3. Big Data Use Cases in Insurance

3.1 Use Case Description Methodology

One of the key points about this thesis is its approach to the design of Big Data artifacts, particularly the Reference Architecture. It is based on business-oriented aspects, a fact that is ensured through developing business-driven, not technology-driven Big Data Use Cases. As several experts have emphasized, the companies doing best are the ones focusing on business and strategy aspects first when it comes to Big Data projects [4, 15]. Starting with strategy means analyzing what data one needs for which purposes so that these purposes generate an added business value. It is also advised to rely on proven Use Cases that have already been tried out by other companies [15]. However, insurance companies have to lead innovations as well [2] – this is why a few Use Cases presented in this chapter are rather visionary and have either not been put into action at all or tried out only by a few companies.

To identify the Use Cases a literature study was conducted. The sources were primarily whitepapers and implementation case studies from companies excelling in the area of Big Data and/or insurance. It has to be pointed out that all Use Cases presented here have either the aim to transform the conventional business models in insurance or the execution of core processes, e.g. claims settlement. On the other side Use Cases that are merely designed to support other processes or Use Cases – e.g. sentiment analysis on its own – are not listed as a single Use Case in this thesis.

In order to have a uniform as well as holistic view on the Use Cases each one of them is described and evaluated following a methodology. This methodology is based on the so-called Smart Strategy Board from Bernard Marr (see figure 3.1) [4]. Basically, it covers all relevant strategic aspects by splitting them up in various so-called panels. Each panel represents an aspect that has to be dealt with when it comes to implementing a Use Case. The panels in the Smart Strategy Board are described in figure 3.1. The most important panels for describing and evaluating the Use Cases are the following:

- **Customer Panel:** If a Use Case envisages introducing a new product, then it is essential to understand what the target market for this product is. One has to know whether the product offered would be accepted by the customers and whether a market does exist at all. Above all the value proposition to the customer has to be absolutely clear, since most Big Data products require the customer to share his data with a company. For convincing a customer to do this, he has to be able to recognize the benefits of the product offered him easily, e.g. in form of a lower insurance premium (compare also with section 2.1.4).[18]

[18] The value proposition in the customer panel is different from the added business value category in the expert interviews. There the added value for the insurance company implementing a Big Data Use Case is scrutinized and it is relevant for each Use Case analyzed. The value proposition here refers to the value a customer gets by buying a Big Data product or service. It is only relevant to Use Cases offering a new product or service – that means some Use Cases presented in section 3.2 are not directly affected by this.

- **Operations Panel:** The internal processes affected by applying Big Data have to be identified for each Use Case. In some cases these processes enabling the Use Cases will have to be restructured as the needs for data processing, e.g. in risk assessment, will change significantly. Since in insurance basically every business process is about data processing, a number of core processes will change entirely and therefore will be regarded as single Use Cases. Another very important issue raised in the operations panel, are cooperation partners. There are many possible scenarios where an insurance company will need a cooperation partner for operationalizing a Use Case: examples include manufacturers of sensor devices or simple providers of external data.

- **Resource Panel:** The current state of play in insurance companies has to be analyzed as well to detect areas where the insurer has to take steps before being ready for implementing Big Data Use Cases. Especially human resources issues have to be pointed out, as talent in the area of Big Data or Data Science is scarce and difficult to acquire. Although the resource panel is not directly part of the Use Case description in section 3.2, it was taken into account when developing the Big Data requirements template (see section 2.2.2).

- **Competition and Risk Panel:** Just as every other IT-project, a Big Data project and Use Case has its own risks. Depending on the risks, appropriate measures have to be taken in order to be able to mitigate them. For instance data privacy is a serious issue in Big Data Use Cases, especially when they use PII. In Germany a number of laws exist to ensure data protection – non-compliance with them would be a serious risk for an insurance company, particularly for the company's reputation with the customers. In the Use Case descriptions such issues are already addressed so that critical aspects about a Use Case become clear as well. The competition and risk panel was also used to develop the questionnaire for the expert interviews and derive possible risk categories from it.

The purpose panel was not regarded in detail for each Use Case hence it takes rather a high-level, enterprise-wide view on the entire company. The finance panel aims to develop an investment strategy and predict profit margins for products – as this is rather a topic for the calculation of a business case the finance panel was not regarded either.

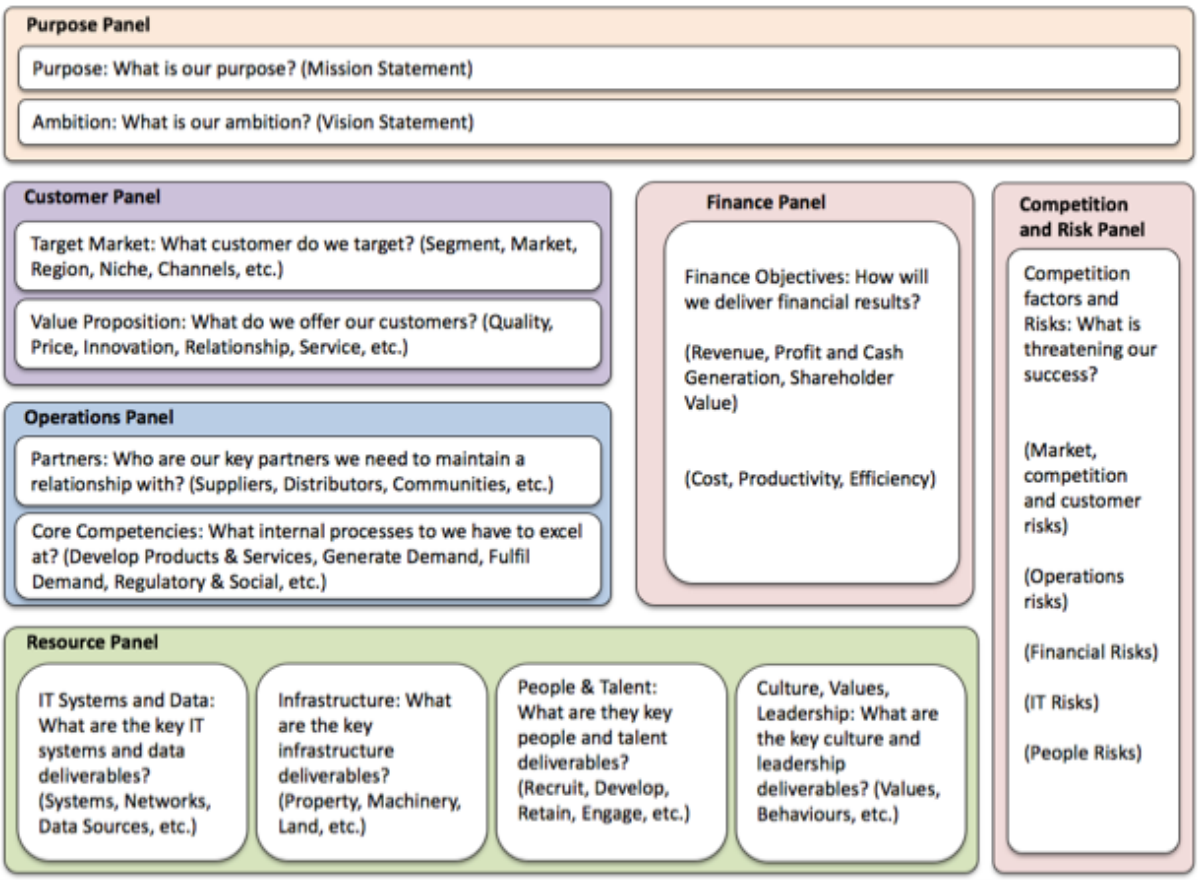


Figure 3.1: Smart Strategy Board
Source: Source [4], Page 30.

Infrastructure costs, scalability issues, lack of skilled data scientists and engineers, restructuring the existing application landscape and business processes – there are many factors driving the costs of Big Data projects. Above all, these projects sometimes do not deliver the promised results or business value. Therefore, it is important to find the most promising Use Cases for the insurance sector. This envisages evaluating a Use Case's potential, which is made up from the added business value it generates and its complexity or feasibility. Eventually such an evaluation of Use Cases can provide decision-makers with a good basis for selecting the Use Cases with which to start a Big Data initiative in their company. So after identifying the Use Cases, interviews were conducted with various experts from a large insurance company. The evaluation methodology and results can be found in section 3.3.

3.2 Use Case Overview

For a better overview, the Use Cases were grouped into four clusters that describe the various areas where Big Data can be applied in insurance. First Big Data's impact on sales and distribution processes in insurance are scrutinized in the cluster Customer Analytics. Afterwards internal processes that could entirely be transformed through applying Big Data are analyzed. These core processes are normally of such high importance to an insurer that the transformation of them could result in high cost reductions for the company. The last two clusters cover new insurance products from Property & Casualty, health and life insurance made pos-

sible through Big Data. Table 3.1 provides an overview on which Use Case belongs to which cluster.
Although the descriptions of the Use Cases follow the methodology introduced in section 3.1, for some few Use Cases – particularly the ones that are not widespread or implemented at all yet – not enough information was obtainable to cover all aspects discussed in section 3.1. In each Use Case description there are examples for possible data sources, however, they are not complete. Other examples of data sources can be found in the requirements description for the Use Cases in section 4.1.

Cluster	Use Case Title
Customer Analytics	1. Churn Management 2. Targeting
Internal Processes	1. Fraud Detection 2. Claims Automation 3. External Data for Optimized Pricing and Risk Assessment 4. Analysis of Enterprise Architecture and Business Processes Based on Monitoring Data
IoT in Property and Casualty	1. Telematics 2. Industrial Insurance 3. Smart Home
Smart Health and Smart Life	1. Health Insurance Based on Wearable Data (discounts) 2. Health Services Based on Wearable Data 3. Disease Management 4. Sensor-based Services in Life Insurance

Table 3.1: Use Case Overview
Source: Own depiction.

3.2.1 Customer Analytics

Churn Management: Keeping its existing customers is essential for every company, not only in the insurance sector. In fact customers are often ready to change their insurer when they get better offerings from other insurers or their current premiums are raised. The prevalence of comparison portals like e.g. Check24 in Germany make it possible for customers to easily compare many different product offerings from various insurance companies. This transparency strongly increases the number of customers changing their insurer. In the German car insurance market 2.18 million customers changed their car insurer in 2014 – an increase by 17% compared to the year before [59]. At the same time acquiring new customers is five to 25 times more expensive than keeping existing ones since these new customers have to be identified first. Therefore customer retention is highly important. However, conventional approaches are not enough for managing customer churn, e.g. by simply monitoring the churn

rate for the company an insurer does know how many customers have left. But the insurer will know this ex-post, i.e. when it is already too late for him to react. The reason is that the churn rate goes up six to eight months after the customer decided to leave [60]. At this point Big Data can play a significant role and help insurers with churn management. It can help to identify the customers who are likely to cancel their contracts and then make them an appealing offer so that they will consider to stay [1]. The identification process can rely on many parameters. The insurer could analyze his interactions with the customer (e.g. from mails, chats, letters or phone calls) and apply sentiment analytics. He can also find out whether the customer did have any complaints or was not satisfied with the service offered. Furthermore, he can analyze the impact of internal processes for the customer. For example if a claim settlement took longer than the average then it is more likely that the customer affected will leave. Above all the current amount of products and services purchased is an indicator for the likelihood to stay: the more insurance products a customer has, the less it is likely that he will cancel his contracts compared to a customer who has only one contract with the insurer. Additionally, existing CRM systems and customer databases can be used as a data source for predicting the churn. Finally social media analysis e.g. in form of Twitter feeds can help to recognize which customers have a negative opinion about their insurer and serve as an early-alert indicator. All this data can be integrated in a data lake that forms the basis for the analytical models that predict the churn probability [61]. If this probability rises beyond a certain threshold then the insurer has to act. He can offer premium discounts or additional services either for free or with a discount as well in order to convince the customer to stay. Of course these offers can also be made when the customer has already canceled his insurance contract. For convincing the customer, the insurer has to make an offer that really suits the customer's needs and wishes, so that the retention probability will be high. This requires a very precise targeting, which is the second cornerstone of customer analytics. The two Use Cases are closely related to and should be interconnected with each other.
If the customer has agreed to a processing of his data no legal issues do arise in this Use Case. It could be however questionable if the insurer does carry out social-media analytics, but since it is not necessarily required for implementing the core Use Case hence many other data sources are already available, this risk can be regarded as rather low. Of course regulatory standards concerning data protection have to be complied with, since PII is processed in this Use Case (see section 2.1.5 for details on data privacy). A possible risk is that customers try to "fool" the churn detection model by imitating the behavior of a customer who wants to churn. This risk can be mitigated by taking many parameters into account when developing the predictive models and continuously evaluate them.

Targeting: One of the greatest promises and threats of Big Data is that it makes possible to know almost everything about people in case they leave any traces from their digital activity. This is particularly interesting for advertisements and marketing. If a company knows what its existing or potential customers really want, it is able to make them respective offerings. The probability of the customers buying these products or services is higher than if the products would have been offered randomly. The basis is the creation of data-based customer segments and profiles. From conventional CRM-data, the customer's current products, his search- and

web history and the communication with him the insurance company can set up various customer profiles. The main principle in targeting is the more data one has, the more precise the profiles get thus increasing the probability of a contract conclusion. The insurer can analyze which products the customers in each segment do mostly have and then offer customers who don't have them but are in the same customer segment the respective products. A simple example is that most customers living in a certain area (determined for instance through the customers' ZIP code) have a burglary protection insurance. He now can offer all customers who have the same ZIP code such an insurance, too. Of course the real targeting and customer segmentation will be based on many more parameters. The basic principle can be compared to what online retailers like Amazon do when using a Recommendation Engine. When implementing a customer segmentation or clustering, unsupervised ML algorithms can be used. They can scrutinize large amounts of customer and external data in order to group customers together based on common values in various data points. Thus customer segments and profiles that have been previously unknown to the marketing and sales departments can be discovered. So eventually Big Data enables very precise cross- and upselling[19] in insurance [1]. When it comes to cross-selling, offering one-time-insurance products becomes possible through Big Data. Imagine the following scenario: a customer has already an insurance product, e.g. a casualty insurance. By analyzing his current location from his smartphone and knowing the current season, the insurer can guess that the customer went on skiing holidays. He now can offer the customer an insurance for sport incidents or ski damages the customer is likely to purchase for the duration of his holidays [62]. In the best case the insurer can offer his customers proactively before the customer knows he needs them and reaches out to the insurer for purchasing them. For example if the insurance company knows that his health insurance customer has now a newborn or is going to have one any time soon (e.g. from personal communication with a sales agent), then he can offer the customer a health insurance for the child.

Furthermore, Big Data makes it possible to move from conventional marketing campaigns towards a so-called real-time marketing. Here a customer triggers a certain event through his interaction with his insurer or a happening in his life. The targeting system then starts the (near) real-time processing of this event by taking into account the customer profile, the time and the place where the event happened. Based on these parameters the customer gets a tailored product or service offering almost in real-time [63].

A highly interesting special case in insurance is analyzing insurance contracts with several insured persons inside them for understanding which person can be put in the best fitting customer segment. A combination of business rules stored in a Rules Engine and Big Data analytics makes it possible to identify the needs of the single customers inside the common insurance contract and thus a more precise targeting [61]. This approach can also be applied in churn management.

---

[19] Cross-selling envisages selling additional products or services that can be related to the product the customer already has. Upselling means the customer gets offered products or services that are more expensive and valuable than the ones he already has, e.g. an insurance rate with more services.

During an online sales process the insurer could use a chat-bot who is able to answer basic questions from a customer who wants to buy a new insurance product. Only if the chat-bot is not able to answer a question a human agent takes over the conversation with the customer, thus helping the customer support department to be more productive. Additionally, customers in P&C could upload an image of the product they want to insure (e.g. a car) and by using image recognition the system could automatically detect the model, license plate and other parameters so that the customer does not have to type in these parameters himself. This would provide a highly improved customer experience to the customer [92].
Besides the obviously created growth through more products and higher turnovers the insurer can also reduce his costs when advertising and selling products. Advertisement campaigns get more precise so that sales agents do not waste time on visiting customers who are unlikely to buy specific products. Additionally letters with product offerings – still a popular communication medium in the personal insurance sector – do not have to be send to large groups of customers, but only to the ones where the insurer knows that they are likely to purchase the offered products or services.
The scenarios presented so far can be applied to selling products to existing customers, however acquiring new ones is highly important as well. And yet, as data is the foundation for targeting, it is far more difficult to implement than targeting for existing customers because the insurer simply has no data about these potential customers available directly. This is why setting up cooperations with providers of external data or services (e.g. a car manufacturer) is so important. Only through these partners the insurer can get the data he needs to be able to approach potential customers with the offers fitting them. Technically this raises issues of developing APIs and setting up standards for exchanging data between the insurer and his cooperation partners.
Finally, data privacy issues are very important for this Use Case. Because of PII processing common data protection laws (e.g. GDPR) have to be complied with. For example in Germany the customer has to agree to product advertisements through e-mail or phone calls (the so-called opt-in regulation) since otherwise this would be a violation of the BDSG. When it comes to advertisements via mail, such a customer consent is not required – however, if the customer actively says he does not want to receive advertisements via mail the insurer has to stop sending him such letters (so-called opt-out regulation) [63]. Furthermore, customer consent is needed if an insurance company covering various insurance products wants to advertise products across this business lines. An example would be an insurer offering both health and life insurance products with a customer who already has a health insurance at this company. If the company wants to offer this customer a customized life insurance, it needs his permission to use his data from health insurance for analyzing which life insurance product would be the best fitting for him.

Both the churn management and targeting Use Case can be used not only in insurance but practically in almost every company from any industry selling products or services to individuals. This is one reason why so many products from commercial software vendors such as Salesforce or IBM do exist for customer analytics. One of the most known concepts is called Customer-360-Degree view. As the name already says, its aim is to get an encompassing view

of the company's customer by collecting data from all interactions between customers and the company. The Customer-360-Degree view is not merely a data lake – it is used as a foundation for creating customer profiles and segments by applying analytical models and business-rules to the integrated data from the data lake [61]. So when implementing the Use Cases from the customer analytics cluster, insurer have to decide whether they want to build the respective systems entirely on their own or buy CRM-software from commercial vendors and then – if needed – adapt or extend them with own analytical models or other functionality.

3.2.2 Internal Processes

Fraud Detection: Each year German insurers lose 4 billion euros on insurance fraud [1]. It is estimated that some five to ten percent of all insurance claims are fraudulent, causing 40 billions of dollars damage to non-health insurers in the US [66]. According to a study by Accenture, a consultancy, insurance companies believe that they could decrease their claims costs by 5% if they could improve their fraud detection systems [65]. Reducing the number of fraudulent claims can also make it possible to lower premiums so customers would profit as well and the insurer might acquire new customers due to pricing advantages. So far insurers have mostly relied on manual analysis and a rules-based processing of claims for identifying fraudulent ones. Often the fraud detection processes do slow down the processing of all claims filed thus affecting all customers. Legacy IT systems are a large hurdle for simplifying and invigorating the anti-fraud processes and systems [65]. Big Data enables insurers to use other approaches for this task by applying ML and complex analytical models. These models have the aim to identify patterns within claims so that newly incoming claims can be automatically classified. The data sources for these models are quite extensive: historical claims data, customer communication data and other customer data are possible internal sources. Particularly text analytics of historical claims can offer great insights for anti-fraud departments. Criminal records or information on whether someone does have a difficult financial situation are possible external data sources an insurance company can come by easily. A very interesting parameter for fraud detection is the combination of numbers on a fraudulent invoice. It has been proven that certain combinations appear more often on fraudulent invoices than on average [64]. For training the analytical models two different approaches from ML are available: supervised and unsupervised learning (please refer to section 2.1.2 for more details on ML). In supervised ML the focus lies on analyzing historical claims data first and labeling the claims as fraudulent or white claims. Additionally it can be analyzed which connections do exist between the fraudulent historical claims. The models then detect patterns in the claims marked as fraudulent while in training and apply these patterns to newly incoming claims when in production. In unsupervised ML data is not labeled first. Instead the analytical algorithms scour through historical and new claims as well as customer data in order to cluster all claims for identifying anomalies in them. These anomalies are then analyzed by anti-fraud experts in order to check whether the anomalies are fraudulent claims or not. If yes the anti-fraud department can identify patterns it has not known before. This is the main difference to supervised learning: there historical claims and patterns in them are known as fraudulent whilst in unsupervised learning new patterns are discovered or claims identified that have not been known as fraudulent before. The unsupervised ML algorithms can then even detect the

impact of single parameters on the entire model and adjust their weighting respectively if required [64]. It has to be noted that especially unsupervised learning algorithms are rather complex and difficult to implement requiring highly skilled data scientists for this.
A noteworthy concept for fraud detection is network analysis. Here the relevant participants in the settlement of historical claims are identified and their interconnections are analyzed for detecting entire fraudulent networks [62, 65]. Possible parameters for a network analysis include [64]:

- Damages repaired at same place.
- Anomalies between a customer's income and the amount of money in an invoice.
- Same places where accidents have happened.
- Same appraiser surveying the damage.
- In car insurance: A cheap car causing an accident with an expensive car.

By combining all the approaches from business-rules, predictive analytics and network analysis, insurers can identify fraudsters quicker and easier than before. Additionally the number of false positives, i.e. claims that are falsely classified as fraudulent, goes down. A large insurance company introduced text analytics and some other modeling techniques for their fraud detection processes. As a result the duration for a claim analysis decreased from 72 hours to some five seconds and the new Big Data based models were twice as precise as the old ones when detecting fraudulent claims. Thus claims can get settled faster, providing a better customer experience [67]. A further possibility for fraud detection is sentiment analysis applied to customer communication, particularly to phone calls when customers file claims. If it can be found out that the caller is highly nervous when filing a claim then this can be used as a parameter in the analytical models as well. However, such analytics capabilities are rather difficult to implement and not needed for the core Use Case – yet when extending the analytical models in the future, such parameters could be taken into account, too.
There are also a few risks when operationalizing this Use Case. Yet legal risks can be regarded as almost inexistent: GDPR poses no restrictions on automated fraud detection hence this is a core purpose of an insurer's business. Because PII is processed, compliance with common data protection laws is required, but standard compliance processes – e.g. getting customer consent to the processing of his data – are already in place and no additional ones are needed for this Use Case only. The only risk that has to be really dealt with is that models can be imprecise at the beginning hence they need time for training. In case of a claim being labeled falsely as fraudulent and the customer being falsely accused of being a fraudster, this can lead to a very negative customer perception and a bad reputation for the insurance company. Therefore it would make sense to introduce a high number of manual checks on the results of the fraud detection system, particularly the ones marked as fraudulent. As time passes the models would get more precise and the manual checks could be reduced to a few samples for quality assurance. Finally the existing processes within the anti-fraud departments will have to be adapted to working with the new Big Data based fraud detection systems. Trainings for the employees and also organizational restructuring could be required as well [65]. Insurers

should nevertheless try to set up such new fraud-detection systems since the benefits from them can quite enormous.

Claims Automation: Another promise and threat from not only Big Data but from digitization as a whole is the possibility to fully automate standardized and routine processes. It is estimated that by 2055 half of all today's jobs could be automated. The ones affected strongest will be those having many routine processes, such as manufacturing or data processing [68]. Indeed insurance is where much automatizing is possible as it is mostly standardized data processing, especially in claim settlement. A study by Oxford University from 2013 predicts that the probability for automating the job of an insurance clerk settling claims within the next ten to twenty years is 98% [69]. McKinsey reckons that the cost of settling insurance claims can be cut by 30 % due to automation [2]. But how can automation be achieved? Already today insurers use automated data processing for claims settlement, however, it is mostly based on business rules that are applied to newly incoming claims. Furthermore, the data that has been analyzed so far is mainly structured. With Big Data it is possible to analyze all historical claims data and combine it with customer and external data. ML algorithms can uncover patterns in how historical claims were settled so far. These patterns can then be used for settling newly incoming claims automatically hence the system knows how to do this based on the discovered patterns. Text analytics can be applied to new claims for analyzing the unstructured text content in them, e.g. from doctors' or surveyors' remarks or Electronic Medical Records. These results can afterwards be compared to the content of a customer's insurance policy or contracts – which can be analyzed with text analytics too – using a set of business rules thus settling the claim. In fact the first insurers have already taken steps towards the development of such systems. In the beginning of 2017, the Japanese life insurer Fukoku Mutual Life announced the implementation of such a system for claim settlement that is based on IBM Watson, the company's AI-system. Watson's task is to analyze doctors' remarks, operation codes and other claims data for settling it automatically [70]. Other insurers like Japan Post Insurance or Versicherungskammer Bayern already use IBM Watson or similar systems for automating parts of their claims management processes as well.
Besides the more or less obvious cost reduction and productivity invigoration from automating the process of claim settlement, such systems could also improve the customer experience. If a white claim can be processed and paid out within a few minutes this would make a huge difference compared to today's claims settlement and make an insurer offering this, highly attractive to new customers and satisfy existing ones. However, this also requires a quick and reliable functioning of the fraud detection systems in order to pay out the justified claims only. Both Use Cases are strongly interrelated to each other: if the data is already cleansed and has high quality inside the claims management system, then the fraud detection system can directly use its data for its own analysis and processing purposes. A common landing zone for claims data would definitely make sense when implementing the two Use Cases.
Just as in the fraud detection Use Case, the ML models need time for training, so that in the beginning it would make sense to introduce a high number of manual checks for quality assurance. Additionally, another ML component can be set up that can take into account manual corrections in case the result of the automated system was wrong, so that in future a claim

under the same circumstances will be settled correctly. Thus the claim settlement system's quality would strongly increase with time and manual checks could be reduced to a few samples.
Compliance with standard data protection laws of course has to be ensured, since claims settlement uses PII and for instance in health insurance also sensitive data. However these issues can easily be solved by getting customer's consent for processing his data. Another much more serious risk is of rather political or organizational nature: the development of such automated claim settlement systems will undoubtedly lead to the reduction of workforce since far less clerks will be required in claims settlement departments. After introducing IBM Watson in its claims settlement process, Fukoku Mutual Life was able to cut the number of their staff in the respective departments by a third [70]. Insurance companies building similar systems therefore should also bear in mind that they have a social responsibility and set up programs for retraining the employees affected in order to keep job cuts at a minimum.
It is beyond any question that most insurers will implement automated claims settlement systems – the only question is when ML models will get precise enough to outperform insurance clerks when settling the claims.

External Data for Optimized Pricing and Risk Assessment: Basically insurance is more or less about assessing and pricing risks for each customer and product. Based on the results of a risk assessment, the insurance premium is calculated. The process of risk assessment in insurance is often called underwriting, although this term is used nowadays almost only in industrial insurance where risks are assessed individually. In personal insurance most companies rely today on common, standardized models for all customers based on internal data. Due to Big Data, insurers now can also integrate vast amounts of external data into their risk assessment processes thus making them more individualized and precise. External data can be used for pricing many different products – some of them like e.g. telematics are described as an own Use Case in this thesis (see section 3.2.3). In this Use Case here particularly the usage of external data for pricing and risk assessment in P&C insurance will be discussed. Since there are many possibilities for different sources of external data, two selected examples will be presented here: weather data and so-called milieu data.
When insuring house roofs against hail, weather data can be taken into account for analyzing which areas have been affected how strongly in the recent years. As weather data is nowadays highly precise, it would be possible to predict hail on a ZIP-code level making the risk assessment highly individualized. Weather data can also be used for insuring real estates against natural disasters such as floods or wildfires. Another useful external data source in house insurance is so-called milieu data. It reaches from the house price to the year when the house was built and when which repair and maintenance work was conducted for the property. All these parameters can be used for calculating the insurance premiums more precisely and on an individual basis for each customer [64]. Just as in claims automation, some insurers have already implemented this Use Case, at least partially. For instance the German reinsurance company Munich Re uses geospatial data for predicting the probability, course and possible damage (as well as the costs resulting from them) caused by wildfires [71].

For acquiring external data, insurance companies need external partner companies that can provide them with it. High quality and granular weather data is available from many commercial sources, for example IBM bought the digital part of the Weather Chanel in 2015. However, there are also various sources of open source data the insurers would not have to pay for, e.g. from governmental or administrative sources. Technically this Use Case requires insurers to build APIs for integrating external data and adapting the current risk assessment and pricing processes to the new parameters.
The benefit for the insurance company from this Use case would be a better and individualized pricing of risks for its products. Above all pricing advantages are possible as well. Imagine the following scenario: so far an entire region was classified as uninsurable due to high risks of natural disasters such as floods or wildfires. If an analysis based on more precise geospatial data delivers the result that some areas within this region are less likely to be affected by a natural disaster than the entire region on average, the insurance can insure property in these areas and make money there. Areas or market segments judged before as uninsurable could thus become new markets.
In the examples presented here no PII is involved so that no data privacy issues do arise. Yet it is also possible to use external data in life or health insurance, e.g. social media, credit scores or others. In such cases insurance companies have to comply with data protection laws. Generally spoken, this Use Case is able to provide a comprehensive view on risks associated with a specific customer, thus mitigating risks for the insurance company significantly.

Analysis of Enterprise Architecture and Business Processes Based on Monitoring Data: Huge sources of data inside companies are their own systems and applications. Each action performed in them and each click done on a company's website does generate monitoring and tracking data. Resource consumption by single applications or tasks within applications can be monitored as well. This is very important for all Big Data Use Cases since if a single application consumes the majority of all available computing resources, other Big Data applications will go down causing an enterprise-wide outage [9]. If a retailer's website is not accessible the company loses large amounts of money since it cannot sell its products. In 2013 a 40-minute outage did cost Amazon, the world's largest online retailer, 4.72 million dollars in lost sales [85]. Whilst insurance companies certainly are not affected this strongly by an outage of their retail websites, it is nevertheless helpful for them to prevent outages here. If monitoring data is available, then it is also possible to analyze causes that lead to an outage and set up mechanisms for preventing them in future. For example response times or hardware failures can be analyzed for detecting erroneous deployments and correcting them. In case of any incidents or outages monitoring data can be scrutinized for finding out which component or system is responsible for the incident – a process that is also called root cause analysis [90]. Based on historical and real-time monitoring data it could be even possible to predict how many people will use which application or (sub-) system in the near future and distribute resources and computing power respectively to prevent long response times and outages. In order to provide this service for websites an integration of web-tracking and monitoring data is required. So eventually this Use Case makes it possible to operate business processes more stable and reliable by using monitoring data.

By tracking the navigation history of a customer on a website it is possible to analyze what exactly a user does there, where exactly he does move away from the website and for how long he does stay on which subpage. Knowing these parameters, insurers can improve their websites concerning e.g. usability or a website's structure. If the dropout rate can be reduced, then more people eventually buy an insurance on the website thus driving growth. Furthermore it is possible to find out from where the users have accessed the website – for instance from a search engine or a direct call. This information is highly helpful for adjusting channels for marketing campaigns and saving costs there. Clickstream data can also be used as a source for the Targeting Use Case or for developing more individualized products based on the customer's preferences thus generating more growth (please refer to section 3.2.1 for more details on this). So the analysis of tracking data helps getting a better understanding of the customer. Yet web-tracking is not directly profiling as data is collected without PII, customer inputs and additionally partially anonymized. On the other side, by analyzing the content of user inputs on retail websites one can detect anomalies there by applying ML algorithms – something very helpful for detecting underwriting fraud.
From a technical point of view, integration issues drive the complexity for implementing a solid system for monitoring the Enterprise Architecture since there are many different applications existing in large companies. Often these include legacy systems that are difficult to integrate with a modern monitoring tool thus posing a great challenge to implementing a real-time data collection. Above all, an end-to-end monitoring system has to be able to cover all layers in the Enterprise Architecture: business layer, application layer and infrastructure layer. The data from these different layers has to be integrated for being able to correlate log-events on the hardware with business driven events executing service calls on the application layer. As of today especially the connection between business events and resource consumption on the hardware layer is technically difficult to implement. The monitoring system further has to be able to communicate with all the monitored applications through a central middleware component and store the dependencies and interrelations between these systems [90].
Whilst monitoring data poses no issues concerning data privacy, when dealing with tracking data one has to make sure no PII is collected there without customer consent. Otherwise one would violate data protection laws such as GDPR. For complying with these laws, PII can be anonymized as soon as it is collected, i.e. appears in a dataflow pipeline.
To draw a bottom line, this Use Case can either be used as a data source for other, more business driven Use Cases or function as an own Use Case that offers stability to enterprise applications and business process whilst also reducing maintenance costs.

3.2.3 IoT in Property and Casualty

One of the really "big" sources for Big Data are sensors installed on various devices such as cars, machines, turbines and so on. When connected with the Internet, these devices or rather the ecosystem they are forming, is called Internet of Things (IoT). Gartner reckons that by 2017 8.4 billions of devices will be connected and almost 2 trillions of US-dollars will be spend on IoT [78]. These figures point out the gargantuan business potential created through IoT – a potential insurance companies in Property and Casualty (P&C) could benefit from as well. The Use Cases described in this section cover only the usage of IoT for P&C products

although IoT devices can also be utilized in health and life insurance. The respective Use Cases are presented later in section 3.2.4.

Telematics: With cars being more and more packed with various different sensors, they become data generators making it possible to analyze the driving style of drivers. This also offers new possibilities for car insurers. They can offer products, which take the results of a driving style analysis as a basis for calculating premiums for car insurance. So far the risk assessment models for car insurance were based on a large set of common parameters for each customer such as car type, the driver's age, etc. Thanks to Big Data these assessment models can now become individualized by using telematics. Generally spoken two approaches are possible: PAYD and PHYD. In PAYD, or Pay As You Drive, the insurance premium, which has been initially calculated in a conventional way, is adjusted depending on the amount of kilometers driven. The basic idea behind this is that the fewer kilometers are driven, the lower the risk of an accident also is – with a driver getting a discount for his premium if he drives less than the initial amount of kilometers written in his insurance contract. One insurer offering PAYD products is the insurtech Metromile in the US. Their product uses an odometer for measuring the distance driven and offers each customer who drives less than 12000 miles per year a discount. On average, Metromile's customers can save up to 50 or 60% on their insurance premiums per year [74]. Since the only relevant parameter for PAYD is the distance driven, it is rather easy to implement. On the other side PHYD, or Pay How You Drive, is more complex to build. PHYD envisages the analysis of the driver's driving style. i.e. a large number of parameters have to be utilized including driving speed, acceleration, braking, the distances driven and various more (see section 4.1 for more parameters). Using this large amount of driving data the insurance company can calculate a risk score and offer a discount for the premiums depending on this score. The theory is that a rather careful driver (without hard braking, driving only at the allowed speed, etc.) has a lower risk of an accident. This would lead to fewer payouts for the insurer so that he could save costs. Therefore insurance companies want to motivate their customers to drive more carefully [61, 72]. Just as in PAYD, PHYD can use either a special device such as a telematics box or a smartphone for collecting the data. Particularly in PHYD the question arises whether incautious drivers should then have to pay higher premiums as a penalty for their riskier driving style. On the one hand insurers have to hedge themselves against such drivers, but on the other hand these drivers could then change their insurer and the company would lose customers [61]. When implementing the Use Case, it is needed to clarify whether an insurer wants to build the entire system on his own or partner with an expert company for the collection and analysis of sensor data.
Additionally telematics data can be used for a more precise calculation of new premiums both for existing and new customers. An analysis similar to the approach described in the targeting Use Case in section 3.2.1. that is based on the development of risk profiles by integrating telematics and customer data could be used for this [61]. Thus the calculation of premiums in car insurance would become more individualized as well, so that insurance companies would have an improved risk management approach for their car insurance customers.

And yet it is highly controversial whether the parameters from a telematics box are really sufficient for predicting a car accident since factors such as the driver's distraction in case he looks at his phone are not tracked and analyzed at all. Actually, in PHYD it is needed to collect and analyze not only a few data points for preventing accidents, but a large number of parameters. These parameters should include contextual data, e.g. the car's location and the routes driven. Depending on whether the customer drives only in a city or spends lots of time on a highway, the risk of an accident either decreases or rises respectively. The analytical models for calculating risk scores should not only use averages or total numbers of harsh brakes, but also try to analyze the reason that caused the harsh brake. This means that behavioral data of the customer should be taken into account as well. For example driver distraction can cause harsh brakes, which in turn are strongly correlated with accidents. For preventing accidents it is needed to analyze such types of driver behavior and motivate the customers to change them, e.g. with discounts on premiums [75].
Besides offering discounts, insurance companies could also provide services to their customers in car insurance. These services can reach from helping when looking for parking lots to providing information on fuel consumption, the state of the car and so on. Metromile, the insurtech mentioned before, offers its customers an app that by using another device analyzes the car's state data in order to detect any anomalies. The app is able to scrutinize the issue and if needed suggest the customer to visit a repair shop that has partnered with Metromile [74]. For such offerings it would make sense for insurers to enter partnerships with car manufacturers that are experts in these areas and already have enough data available for providing such services. Own developments by insurers would require high investments in technologies they are unfamiliar with and would make no sense since a large number of possible partners is available.
Finally some customers might feel tracked because the telematics system monitors where people are driving so that highly detailed movement profiles of customers could be set up. As this has to be regarded as PII, the telematics Use Case when based on a PHYD model, has to ensure compliance with data protection laws. If in future most car insurers offer only telematics products then customers who do not want to have such a product because of fearing surveillance this would raise the question of solidarity in insurance – however this is rather a political question. A possible solution could be that it is mandatory for all insurers to offer a standardized car insurance tariff without telematics that is based on conventional parameters.

Industrial Insurance: The disruptions caused by Big Data are also visible in manufacturing. Due to the rise of IoT, many manufacturing plants will be equipped with sensors for monitoring their state. Given all this sensor data, a manufacturing company could set up a so-called predictive maintenance system. Its goal is to analyze the sensor data from all components in the manufacturing plant (or a single machine in this manufacturing plant) in order to predict an outage of a component. Due to a far-reaching integration of the single components within the plant it is possible to monitor entire production processes within a plant. As soon as the values of the sensors do pass a certain threshold, the system does trigger maintenance activities conducted by a technician. Thanks to this, a manufacturing plant will not break down so that the factory has not to stop production. Several companies have already built such predic-

tive maintenance systems. One of them is the German car manufacturer Daimler, who uses a solution from IBM for monitoring their production of cylinder heads. More than 500 parameters are analyzed for this, including temperature, pressure, the size of the single component parts and many more. In case of any anomalies or deviations, checks and if needed, maintenance measures are conducted [73]. Of course the parameters for analyzing the risk of an outage do vary depending on the type of manufacturing plant and processes, but the basic idea is the same.

Such scenarios become interesting for industrial insurers who offer insurance against operations interruptions.[20] With such an insurance industrial companies can assure themselves against high losses in case of an outage in a manufacturing plant. If an industrial company has such a predictive maintenance system, the risk for an outage decreases. For the insurer this means that the risk of having to pay out the industrial company does go down as well. Since the sums the insurer would have to pay in case of an outage are very high (besides the costs for a standstill of the manufacturing plant it also takes long time to replace the damaged components in a plant since these are often very specific for each plant) they can also profit from predictive maintenance.

Industrial insurers now have two possibilities. One is simply offering discounts on premiums to each industrial company that has such a predictive maintenance system. First an underwriter would confirm the existence and correct or proper functioning of such a system and then calculate the respective discount. The other, more complex option would envisage using the monitoring data from the manufacturing plants as a basis for assessing the risk of an outage in this plant and calculating the insurance premium using this assessment. First of all, this would require the industrial companies to be ready and willing to share their monitoring data with their insurance company. This is why they have to be treated by the insurance company not only as simple customers but also as partners. Additionally whilst the first option only poses a need for some additional training for the underwriters, the second one would transform corporate underwriting in a significant way. A new technical infrastructure would be needed for enabling the data pipelining from the industrial companies to the insurer and completely new analytical models for the risk predictions. Of course the industrial insurer would then need to hire new experts in both data science and data engineering for building such an application. Yet the larger part of the Use Case implementation lies not with the insurer but with the industrial companies who have to build the predictive maintenance system.

Concerning partnerships, a big opportunity for industrial insurers would be to partner with manufacturers of production plant elements or large machines. For instance every gas turbine produced by Siemens or General Electric could be then insured by one insurance company, no matter to whom it is afterwards sold. Thus insurers could quickly get new customers and access to gas turbines' data.

Since the data used here is technical sensor data and no PII is involved, no compliance issues arise for this Use Case concerning data privacy. What has to be taken care of, however, is IT-Security: the monitoring data of production plants is highly sensitive for the industrial com-

[20] In German they are called Betriebsunterbrechungsversicherungen.

panies and must be protected from unauthorized access. This has to be done by the insurer when he has this data in his own systems. Only if an insurance company can ensure this, the industrial companies will be ready to share their plant monitoring data with it.
Although this Use Case has not been implemented by many insurers and is rather unknown so far, it is worth testing, especially because the benefits an insurer could reap through this are very far reaching. This Use Case has the potential to transform industrial insurance entirely – away from insuring single components and plants towards insuring complete manufacturing processes.

Smart Home: In the world of Internet of Things it is not only cars and factories that are packed with lots of sensors but also homes. Almost every device from a fridge to water pipes inside a house can be equipped with sensors. Smoke detectors can quickly detect a fire and automatically inform the fire brigade since they are integrated with a communication system. Thus the damage caused by a fire would be far smaller than if the fire brigade would be informed later or not at all. By analyzing data from sensors on water pipes, overpressure that could cause the burst of a water pipe can be detected before this could happen. The water supply is then cut off and a technician sent to fix the pipes as soon as possible. By installing cameras and other sensors, anomalies can be detected that are associated with a burglary thus increasing the security for the residents. All these scenarios are interesting for P&C insurers as well, because they help lowering the risk of damages in house insurance. This would help insurance companies to save money by having to pay out less because of fewer claims filed. In order to motivate their customers to install such sensors at home, insurers could offer discounts on their premiums for house insurance products. Furthermore, insurers could also offer services that supplement the sensors. This can be either partner repair shops offering repairs in case damages do occur or in the water pipe monitoring example a provider specialized on fixing water pipes. This would result in a better customer experience and the insurer could provide the customer a services ecosystem that covers all aspects from insurance to damage fixing [61, 76].
The risks and challenges for implementing the Use Case of course do vary depending on the insurance products. In the water pipes example the difficulty is equipping pipes with sensors when a house has already been built – an undertaking that besides being technically complex is also quite expensive. One possibility for the insurance companies would be partnering with large real estate owners such as Deutsche Wohnen or JLL. These own large housing or office complexes and have the budgets required for installing sensor devices for water pipe monitoring in these complexes. If the real estate owners set up the monitoring systems, then the P&C insurer could offer them discounts for their premiums. However, this is only an option for commercial P&C clients – the difficulties with retail customers remain. When it comes to intrusion detection systems, data privacy issues do arise if cameras or sound sensors are used since customers could then feel surveilled. Additionally the misuse of the data and unauthorized access to the data, e.g. camera recordings has to be prevented by setting up high IT-Security and I&AM standards.
Another issue insurance companies have to deal with, is that they definitely cannot produce, install and configure the sensors. Therefore a company excelling in these areas would have to

be acquired as a cooperation partner. In this case it would have to be clarified whether the cooperation partner does only provide the data or whether he also performs the data analysis so that the Use Case would be completely outsourced to the partner company. Furthermore the question arises who would then own the sensor data – in case the insurer would have to pay large sums for the data, it is questionable whether the entire Use Case would still be profitable for him. Finally there is another technical issue concerning connecting different sensor types with each other because the sensor landscape is highly heterogeneous and very few standards do exist so far, although this could change over time. For now it requires a high effort integrating the sensors – if a customer already has a sensor installed that does not fit to the standard used by the insurer's partner company providing sensors, does he have to change his sensor devices in order to be eligible for a sensor-based house insurance? Such questions would have to be answered when operationalizing the Use Case.
First insurers nevertheless have moved to offering such products. In France, Allianz sells a sensor-based fire insurance that uses data from smoke detectors to identify anomalies. In case of an incident the customer is informed first, who can then look up what is happening in his home using a camera. If he does not react, then an Allianz employee does notify the fire brigade, who then go to the customer's house. With the amount of connected devices growing, the attractiveness for the insurers of offering respective insurance products making use of them will only rise [78]. Given the fact that water damages have the biggest share of all damages in P&C and are very expensive to fix, the water pipe monitoring products are especially attractive for P&C insurers [77].

3.2.4 Smart Health and Smart Life

Health insurance based on wearable data (discounts): The world of IoT in insurance is not limited to P&C products only. Sensors installed in smartphones and fitness trackers make it possible for people to monitor their own health just as the state of manufacturing plant. These devices are called wearables, hence they are worn everywhere by their user. For simplification, all fitness tracking devices and smartphones, which can be used for health monitoring purposes too, are referred to as wearables here.
Using their sensors, wearable devices already today can monitor how many steps someone has walked, analyze the sleep phases, the pulse, the heart rhythm and the calorie consumption. Modern devices can even measure the glucose level, which is very helpful for people suffering from diabetes. The current health state of the user is then visualized on dashboards in a mobile app. The results of these analyses is highly interesting for health insurers since it is believed that the more a person is moving (or working out) the likelier it is that he will stay healthy. Now every health insurer is interested in keeping his customers healthy since he then will not have to pay money for curing their diseases. Health insurers could start offering their customers wearable devices in addition to their insurance products or request access to the data of the customers' own devices for analyzing it or simply receiving an average score value that indicates whether a person is moving a lot or not [76]. Based on the score the insurer either receives from a provider or calculates directly on his own, he can then offer discounts to his customers on their initial premiums. The data needed for calculating this score can be limited to only of a few data points such as the number of steps walked or calorie consump-

tion – this would be sufficient for the core idea of the Use Case. Of course the models could be extended by taking into account all the parameters mentioned before. Indeed, first insurers have already started offering such products. Even in Germany, where the public opinion concerning sharing own data is very reserved, the insurance company Generali offers a product called Generali Vitality that analyzes how much the customer has been moving. The wearable device is provided by Garmin, a manufacturer of such devices. Generali additionally offers its customers a 50% discount on buying the Garmin device. Based on the analysis of the health data, Generali offers four different stages of discounts for occupational disability insurance or term life insurance with the plan to expand into health insurance products. To ensure that customer data and the detailed fitness tracking information will not be integrated, Generali even did set up a new company that functions as an own legal entity. While it has the job of analyzing the fitness data, it only reports scores resulting from these analyses to the Generali parent company for calculating the discounts [79]. Nevertheless Generali already got criticized by data protection officers and activists for their new product. Besides pointing out that the customers have to agree to the processing of their health data because otherwise they are not eligible for the product, data protection officers criticize that people who are old or disabled cannot participate in the program [79]. Indeed the entire product model of offering discounts to people who are more active than others is doubtful when seen from the point of view of insurance solidarity. Generally spoken, customers who are less active have to pay for the customers who are more active. Eventually, it could be possible that insurers charge people who are not active enough more for their premiums if such products are not properly regulated. Additionally, since health data is to be regarded as the most sensitive data besides financial data, data privacy and IT-Security issues arise for this Use Case as well. Because of the importance of the data protection question for all Use Cases using health data, an entire passage at the end of this section discusses it.

Another issue is that people who work out too much have a higher risk of sports injuries, so that the basic idea behind the Use Case would collapse. The models calculating the scores should take this into account as well by reducing the discount from a certain maximum threshold again.

Besides the expected lower risk for future illnesses and the associated cost reduction, younger customers could be attracted by an established health insurer as the discount system can provide an interesting customer experience for them. Therefore the insurance company could also acquire new customers if it implements the Use Case.

As the Generali example shows, insurance companies have to partner with manufacturers of wearable devices for operationalizing this Use Case. The question as in all IoT Use Cases is who will own the data then: the insurer or the partner? The profitability of the whole Use Case depends on this answer. Furthermore, insurance companies have to decide whether they want to use the partner for data collection only and develop the analytical models on their own or whether they want to outsource the implementation of the Use Case entirely. They also have to clarify whether they want to offer insurance products based on wearables data through their normal health insurance company or by a newly founded legal entity – just as Generali did do. The latter could help persuade customers that their insurer has no access to

their fitness tracking data – an approach that is fitting in countries where data privacy is regarded as vital.

Health services based on wearable data: In the previous Use Case a new insurance product was introduced that analyzes fitness tracking data and offers customers discounts on their insurance premiums if they are physically active. Although this Use Case is technically quite similar to the previous one, its goal for value creation is completely different. Instead of offering discounts, the insurance company now offers health services based on the analysis of the fitness data. Depending on the current health state, they can include consultation on how to remain healthy, nutrition plans, workout plans and so on. Compared to the previous Use Case the insurer can help his customer staying healthy more proactively and offer him consultation on how to do this [76].
Again the insurance company needs a cooperation partner for the tracking devices and the questions of data ownership and whether to outsource the implementation entirely to a provider, arise here, too. Additionally, if the insurer wants to provide a highly precise consultation, he needs more than only the fitness tracking data. Customer data about previous illnesses and treatments has to be integrated with fitness tracking data so that the analytical models can generate precise outcomes and offer good individualized consultation. Apparently this raises additional data privacy issues when compared with the previous Use Case (see the end of this section for a deeper analysis of data privacy) and makes it also more complex to implement. Nevertheless this Use Case could generate more growth than the one relying on discounts only, since offering health-consulting services in addition to a health insurance provides a completely new customer experience. This is especially interesting for younger customers who are also rather willing to share their health data with their insurance company.
Another interesting aspect about wearables is that they can be used for a continuous monitoring of the health status of a customer. Whilst a doctor has little time to check a patient and healthy persons do visit doctors rather seldom, the wearable is always monitoring the person wearing it. This means it is also possible to use wearable data for predicting illnesses in case of strong deviations or anomalies in the values monitored. The predictions become also much more individualized, hence they are based on the customer's own health data. Above all, wearables can also be used for treatment purposes too: if a wearable is equipped with glucose sensors it can be used for helping diabetes patients. People suffering from chronic[21] diseases can be reminded to take their medicine if the wearable's monitoring system notices any anomalies [80]. In extreme situations wearable devices could even call an ambulance for the patient. As the treatment of chronic diseases makes up for 70 to 80 % of all health spending, Big Data apparently could provide a remedy here. The deterioration of a patient's state could be prevented and the duration of hospitalizations could be reduced, thus helping to cut costs in the health sector and for the insurers [81]. Eventually wearable devices and especially smartphones can also be used for telemedicine: here a patient does not have to visit a doctor; instead the doctor can check the patient's state remotely using the sensors and the camera in

[21] A chronic disease is a disease lasting longer than three months.

the smartphone. If he detects any issues, the patient comes for a more detailed check and treatment to the doctor. The entire approach helps saving costs and time and additionally provides a better customer experience for the patients [61]. As wearable devices move from being merely a gadget for people working out a lot towards medical devices ready for clinical use, insurers should keep up with this development and try to benefit from it.

Disease Management: In the whole area of health insurance this is the Use Case with the greatest opportunities but also the most complex one. The key point about it, is that the insurance company collects data on the customer from many different sources in order to predict whether a customer will get sick in the future and which disease he is going to have. With this information the insurer can then take steps to prevent the disease by offering the customer either consultation on how to remain healthy or offer a preventative treatment before the customer gets ill. Thus it has the potential to transform the health insurance entirely: from a reactive model where the insurers helps the customer when he already got sick, towards a proactive approach where the insurer helps the customer to prevent illnesses. Everyone in the healthcare sector would benefit from this: people are less likely to get sick and have fewer illnesses whilst insurance companies and hospitals can save money. Keen predictions even reckon that if disease management could be fully implemented, 80% of all doctors would not be needed anymore because the amount of sick people would decrease that strongly [81].
The required data for predicting a disease can come from various sources: Electronic Medical Records, customer data, ICD-codes[22], previous illnesses and many more. Data from wearables as in the previous Use Case can be used as well, since it offers a continuous monitoring of the customer's health. This data has now to be integrated and complex predictive ML models have to be applied to it in order to calculate the probabilities of diseases. Due to the large number of different illnesses, there are many possible outcomes for such predictive models, so they have to be highly sophisticated. Nevertheless, it is possible to set up these models for health analytics by using some of the proven predictive models from Google or other US Internet giants excelling in this area, since the underlying ML concepts are the same [80]. The basic idea of disease management is comparable to a Customer-360-Degree view, where all available data from customer interactions is collected and analyzed. The difference is that in this Use Case it is mainly health data that is collected and analyzed.
When regarding only the technical feasibility, disease management is no future dream for Big Data visionaries, but reality. Already in 2014 a clinic operator in Virginia partnered with IBM for a pilot project and was able to predict the risk for future heart failure diseases with an accuracy of 85%. The data used consisted of structured data from transactional systems in the hospitals and unstructured data such as doctor's remarks, notes or reports from medical checks. For analyzing unstructured data, text analytics was applied. Parameters used for the predictive models included blood pressure, the illnesses a patient has so far suffered from, the medicine the patient has taken so far and other environmental factors such as age, sex, occupation and several more. People at high risk of heart failure could now be offered an individu-

[22] An international classification system for almost all existing illnesses.

alized treatment to prevent this disease for them [82]. By applying text analytics to the content from medical journals, various US Internet companies are trying to develop cancer treatment therapies. Furthermore, through analyzing patients' DNA, cell structure, biological ecosystems and many more it is possible to learn more about diseases and their properties. Big Data is also capable of transforming biology and medical research.

There are two main challenges when operationalizing the Use Case: obviously the first is data privacy, hence the customer has to share all of his most sensitive data with his insurance company. The second is the lack of data, because currently no insurance company in the world has got enough data for making a sufficiently precise prediction on future illnesses for its customers. In fact, health data is spread between many different silos in insurance companies, clinics, pharmaceutical companies and so on. There is lots of it available: in 2013 there were 150 Exabyte of data in the entire health sector. However, only 10 to 15 % of all companies or hospitals in the health sector have so far exploited the full potential of their data. Besides the silos mentioned before, the lack of data integration within single companies or clinics is a reason for this poor performance. The data stored and processed in the health sector is rather complex when compared to data used in banks or retail companies. Additionally, IT budgets in the health sector are quite low, so that not enough money for data driven integration projects has been available so far [83]. The solution lies in breaking up the data silos and other organizational barriers by creating a common data platform for all actors in the health sector where they could pipeline their data to for integrating it. This "health data lake" could be the basis for all analytical models used in disease management for predicting illnesses or clinical trials of new drugs. Setting up this data lake makes it possible to uncover new patterns and causalities for studying and predicting illnesses. Yet building it up requires trust and partnerships with many actors such as clinics, pharmaceutical companies, academic research institutions and many more. Therefore the data integration is not only a technical issue, but should also include governance and building trust. Additionally, since health data is highly sensitive, the data lake should guarantee the highest IT-Security standards [81, 83]. This is where the chance for insurance companies to take the lead in setting up a "health data lake" lies. Since insurance companies are regarded as very trustworthy and have implemented various IT-Security mechanisms, they could use this reputation for convincing other actors in the health sector to put their data into a data lake managed by insurers.

If the creation of this "health data lake" should fail, then insurance companies could try to persuade their customers to share their data directly with the insurers, e.g. by granting the access to a patient's Electronic Medical Record. Insurers could also partner with single actors from the health sector: for example to get data from clinical trials or the course of diseases in order to have enough data for training their own predictive models. The predictive models could also be developed by partnering with research institutions or pharmaceutical companies. Eventually partnerships will be vital for successfully operationalizing the disease management Use Case.

To draw a bottom line, disease management has vast potential for insurance companies and all other actors in the health sector. Nevertheless it will take many years before it can be fully implemented, PII issues can be solved and trust can be established between the various actors involved.

Sensor-based services in life insurance: When it comes to Big Data, life insurance is a special case. Unlike in P&C or health insurance, very few claims are filed here so that the amount of customer interactions is very low. Actually, after the underwriting process there are almost no customer interactions. One possibility to increase customer interactions and provide a better customer experience is offering wearable-based assistance services to the customers. The basic idea is quite similar to the Use Case where health consultation services are offered using data from wearable devices. However, instead of providing consultation, the aim here is to support the customer in case of any incidents. Therefore his health state is continuously monitored using a wearable device – in case of any anomalies or deviations in the vital signs such as pulse or blood pressure the customer can be advised to visit a doctor [40]. If the deviations are critical the system can call an ambulance to check whether the customer has suffered a heart attack and help him if needed thus saving his life. As in all Use Cases using IoT devices, the life insurer needs to partner with a manufacturer of the monitoring device.
Besides the data privacy issues that arise in each Use Case processing health data, there is also a high risk for the insurer in case of false positives. Here the system either does give alarm if no incident has happened at all or it does not give alarm in case of an incident. Implementing a function where the customer can deactivate the alarm before anyone is informed if there is no incident, can solve the first scenario. However, the second scenario cannot be ruled out completely even if the system is properly tested and guarantees high quality. This means in case this scenario becomes reality, the life insurer would have to expect lawsuits from the customer or his relatives. Such options of course do pose a hurdle for life insurers to implement this Use Case.

As it has already been pointed out, data privacy and protection is of paramount importance for each Big Data Use Case that is processing health data. Health data is the most sensitive data of all data types and it takes a good deal of trust for customers to be ready to share it with insurers. Since health data often has to be regarded as PII or is used together with PII so that the analytical models can deliver precise results, the respective data protection laws have to be complied with. Customers also have to agree to share their data and its processing by an insurance company. Above all, IT-Security is a very important issue, as it has to be made sure that hackers do not steal health data. In 2014 one of the largest clinic operators in the US, Community Health Systems, was hacked. The stolen data included patients' names, policy numbers, diagnoses and information on invoices and billing. This data can then be used to file fraudulent insurance claims or buy medicine for reselling it later. Because such hacks and the theft of health data is often not noticed quickly enough, hackers and criminals can use the stolen data for several years. This leads to the fact that a single data set of health data is sold for ten dollars which is 10 to 20 times more than a credit card number. At the same time it is rather easy to hack the IT systems of a clinic or a healthcare provider, since they are often operating very old legacy systems that are not able to withstand modern hacker attacks. In fact the number of these attacks on the health sector is rising: in 2009 only 20% of all companies in the health sector were affected, by 2013 already 40% were [84]. Insurance companies are better off when it comes to IT-Security as they have invested large amounts of money into

renewing their systems and setting up new digital defense lines. This is also why they could be the ones leading data integration projects in the health care sector (like in the disease management Use Case).
Finally people's readiness to share their health data is decisive for the success of all health Use Cases. Whilst in the US 43% of people are ready to share their health data if they get a financial benefit from it, in Germany only 22% are. In China even 79% are ready to share their health data (see figure 2.1). In the US it is believed that customers will be ready to share their data if they get a financial incentive like the premium discount [80]. Therefore it is likelier that customers in China or the US will purchase data driven health insurance products than customers in Germany. Nevertheless insurers should focus on such markets too and offer customers a compelling added value in order to convince them to share their data.

3.3 Use Case Evaluation Methodology and Results

In order to find the most promising Use Cases for the insurance sector the ones identified in section 3.2 had to be evaluated concerning different criteria by conducting expert interviews in a leading insurance company. To develop the questionnaire, two main sources were used. One was an evaluation framework for Big Data Use Cases from PwC where the added value and complexity analysis were derived from [5]. The other was the Smart Strategy Board. Based on the completion and risk panel the risk analysis question and the respective risk categories were derived. Finally the operations panel brought the question for the need of a cooperation partner when operationalizing a Use Case [4]. At this point it is worth noting that a cooperation partner could be either someone helping to keep a Use Case running (e.g. by providing a sensor device for data collection) or someone who helps implementing the underlying system for the first time (e.g. the provider of a large AI-system). The questions asked in the expert interviews are the following:

1. Is the Use Case already implemented in your company? If not, do you plan an implementation?
2. On a scale from one to four with one being lowest and four being highest, how high do you think is the added business value created through this Use Case?
3. Which type of added value is created through this Use Case? Possible categories included cost reduction, growth creation, automation, risk mitigation, the introduction of a new product or providing an improved customer experience.
4. On a scale from one to four with one being lowest and four being highest, how high do you think is the implementation complexity for this Use Case?
5. What are possible risks that could arise when implementing this Use Case? Possible risks categories included IT-Security, market risks, financial risks, data privacy/legal, lack of data or negative customer perception.
6. Do you think there is a need for a cooperation partner when implementing this Use Case?

These six questions were asked for each Use Case presented in section 3.2. Additionally the interviewees were asked for their line of business (P&C, life insurance, health insurance or

another cross-functional unit), their company department (IT, Business Analysis & Architecture, Business Owners) and their work experience in years.
During the interviews, the interviewees were shown a presentation in order to give them an overview over each Use Case. For each Use Case there was a description of the main facts about it and an industry example or a possible scenario for the Use Case when it is implemented. Presenting an example did help getting a better understanding and perception of the Use Case, so that particularly the decision makers previously not familiar with a Use Case were able to give a profound assessment of the respective Use Case.
There were 14 interviewees with 18 years of work experience on average, working in different company departments and covering various roles. These roles included senior managers, Big Data experts and also many business owners so that one could get a proper, business-based assessment of the added value each Use Case did generate. The interviewees were chosen in a way that all possible products and service lines in an insurance company were covered, which are P&C, life, and health insurance as well as internal functions like data analytics, BI or architectural governance.
For non-disclosure reasons the detailed results of the Use Case evaluation are only be published inside the insurance company where the interviews were conducted and will not publically available. However, an overall summary can be given describing the main results of this evaluation. The claims automation Use Case was regarded as the one that could by far generate the highest added business value through reducing costs and providing a new, compelling customer experience when settling claims within a few minutes. At the same time, this Use Case is judged to be rather complex to implement, especially because of the organizational issues to be solved during the restructuring of a claims-settlement department, where the number of employees would be significantly reduced after implementing it. Fraud detection, a Use Case that is required for operationalizing claims automation, is judged to provide less added value than claims automation (though still at a reasonable level) but is just as difficult to implement. Here the difficulties are not caused by organizational issues, but through technical challenges, because sophisticated fraud detection requires complex unsupervised ML models. Because of this combination of a high added value and various challenges to implement them, these two closely related Use Case were selected for testing the Big Data Reference Architecture by designing a Solutions Architecture for each of them in section 5.3.3.
Each Use Case from the Smart Health and Smart Life category was judged to be associated with a number of risks when operationalizing them. The reason is the processing of highly sensitive health data, which poses challenges in data privacy and IT security for an insurance company. Additionally the insurer does have to convince the customer to share his health data (e.g. from wearables or electronic medical records) with him, what leads to the risk of a lack of data. This, and the need for highly complex predictive ML models is the reason why disease management, which is the Use Case with the highest added value in this category, is regarded as most difficult to implement of all Use Cases presented in section 3.2.
Finally, there are some Use Cases that provide a good added value and are easy to operationalize. One of them is the usage of external data for pricing, where external data could offer new parameters for the underwriting process thus significantly improving the insurer's risk management and assessment. Since technical difficulties are reduced to building APIs for

external data and slightly adjusting existing risk assessment processes, the Use Case is easy to implement. Another such Use Case is churn management, where the probability of a customer canceling his contract is predicted and if needed, an offer is made to him so that he is convinced to stay. This helps the insurer save a large amount of money, as keeping existing customers is far cheaper than reacquiring new ones. Since the required data for predicting a churn is available, the relevant parameters are clear and the predictive models are not very complicated, the Use Case can be implemented without major problems.

4. Requirements for Operationalizing Big Data Use Cases

4.1 Use Case specific Requirements

In this section for each Use Case that was outlined in chapter three the requirements needed to operationalize it are presented. The descriptions follow the template introduced in section 2.2.2. The data sources listed in the requirements are only examples for a set of parameters that can be used for the respective Use Case and can be extended if needed. The presentation and operationalization category also offers only a few examples or scenarios on what the presentation of the results, the user interface or a subsequent processing of the results can look like. As mentioned before, the knowhow category is not filled in for non disclosure reasons. Furthermore, some important aspects pointed out in chapter three in the Use Case descriptions are included here as well.

4.1.1 Customer Analytics

Use Case Title		**Churn Management**
Description		Through analyzing customer interactions it is predicted via a statistical model how likely it is that the customer is going to cancel his contract. When reaching a significant probability of churn, the customer gets an offer in order to persuade him to stay with the company
Big Data Characteristics	**Data Source**	Customer interactions: mails, phone calls, letters, click-history, existing CRM-data, data from transactional systems (Core-Insurance), internal process data (e.g. settling a claim takes too long or longer than average time), customer complaints, social media, business rules. PII-data on customer, data about his contracts (when last cancelled if applicable, how many contracts does he have, payment method).
	Volume	Average amount of data when compared to other use cases.
	Velocity	Depending on source, some inputs require **streaming** the data (e.g. social media monitoring) to get into the landing zone. Data from transactional systems or CRM-data can be **batch-loaded**. Data from customer interactions and communication should be **streamed** so that the insurer can react instantly to any changes that might seriously affect the probability of a churn.
	Variety	High, many different sources involved, so a **data lake** for their integration is needed.

Big Data Science	**Veracity and Data Quality**	Particularly external data needs **data cleansing,** as the quality is not sufficient for direct pipelining into a model. Internal structured data (e.g. from transactional systems or customer databases) has mostly a good quality. Of course this requires the existence of respective processes to ensure good data quality in transactional systems. Nevertheless it should pass a cleansing process as well, although when comparing it with the cleansing of external data some process stages can also be left out.
	Presentation and Operationalization	In case of a high churn probability the churn management application has to trigger a targeting process for generating a fitting offer so that the customer can be convinced to stay with the insurer (if the customer is profitable, otherwise his contact cancellation should not be prevented). The communication channel towards the customer varies depending on the event that triggered the churn process. It can be either an automatically sent mail or push notification (real-time event processing) or a sales agent who receives a daily report on who of his customers are likely to churn (batch-oriented processing).
	Data Types	Data from existing systems or logs is structured, textual communication is semi-structured, phone calls and external data (e.g. social media) is unstructured.
	Data Analytics	Predictive models for predicting the churn probability are needed and the relevant parameters have to be identified. However a part of these parameters can be derived from other common solutions for churn management. The data has to be processed at **real-time speed** so that insurers can react instantly on any changes that affect the risk of a churn. Semi-structured data requires the usage of text analytics in order to analyze the content of a customer mail. When it comes to analyzing the content of phone calls, speech to text algorithms and text analytics have to be applied. Furthermore, sentiment analysis can be used for identifying difficult cases and reacting respectively as frustrated customers are likelier to cancel their contracts.
Security and Privacy	**Personally Identifiable Information (PII) used?**	Yes

	Highly Sensitive data used?	Not directly, only in case of enrichment with social media or other external PII additional risks arise. However if the churn prediction model involves health data for customers in health or life insurance, the respective data protection standards have to be ensured here too.
	Governance, Compliance & Audit	Customer has to agree to a processing of his data since PII is used. Thus compliance with BDSG and GDPR (EU-DSVGO) is required as well.
Organizational & Business Requirements	**Knowhow**	n.a.
	External Partners	None needed
	Other Business Challenges	Although not directly part of the churn prediction, the following aspect is of high importance for the churn management Use Case: the maximum amount of discounts in an offer to the customer has to be calculated so that the whole business case remains profitable for the company. Thus a financial risk can be mitigated.
Other Big Data Challenges		Predictive models for predicting the churn require a long time for training. For getting a 360-degree view on the customer, data integration and bypassing system divisions is vital. **Special case:** Analysis of contracts that contain several insured persons by comparing addresses and family names (cleansing needed here, as this combination isn't always correct or unambiguous). This can be done either with a Rules Engine or by setting up an additional analytical model. Using standard enterprise software could be a possible solution here since churn management applications are already available from many providers. The technical difficulties would then be mostly reduced to data integration issues.

Table 4.1: Churn Management requirements

Source: Own considerations and [1], [61], [64], [93]

Use Case Title	**Targeting**
Description	By analyzing customer interactions and customer data, customer profiles and segments can be set up. Then the customers in the respective segments can be offered products that other customers in the same segment have. By making the targeting as precise as possible the customer can receive offers he is likely to buy as the company knows what the customer is interested in (products, campaigns).

Big Data Characteristics	**Data Source**	Customer interactions: mails, phone calls, letters, click-history (weblogs), existing CRM-data, data from transactional systems (Core-Insurance), social media. PII-data on customer, data about his contracts (when last cancelled, how many contracts does he have, payment method), data from sales agents, any other external data that helps to get a better view on the customer, business rules.
	Volume	For a basic targeting an average amount of data is enough, when the goal is to have a very precise targeting, then it is essential have a large and broad dataset.
	Velocity	Depending on the source, some inputs require **streaming** the data (e.g. social media monitoring or click-logs) before it gets into the landing zone. Data from transactional systems or CRM-data can be **batch-loaded.** Data from customer interactions and communication should be **streamed** so that the insurer can react instantly to any changes in the customer's behavior and enable real-time marketing and churn prevention.
	Variety	Very high, lots of different sources involved, a **data lake** for their integration is needed. Non-centralized data from sales agencies is highly various as well and has to be integrated in the data lake, too.
Big Data Science	**Veracity and Data Quality**	Particularly external data needs **data cleansing**, as the quality is not sufficient for direct pipelining into a model. Internal structured data (e.g. from transactional systems or customer databases) has mostly a good quality. Of course this requires the existence of respective processes to ensure good data quality in transactional systems. Nevertheless it should pass a cleansing process as well, although when comparing it with the cleansing of external data some process stages can also be left out.
	Presentation and Operationalization	Various possibilities for the customer communication after triggering a targeting event are possible. One is where sales agents do receive a notification with the currently fitting product offering. In case of full automation, the system has to trigger communication with the customer (including the new offering) either via mail or e-mail. The targeting Use Case also serves as the source for offers to prevent a customer churn.
	Data Types	Data from existing systems or logs is structured, textual communication is semi-structured, phone calls and external data (e.g. social media) is unstructured.

	Data Analytics	Predictive models are needed for advanced targeting and they have to include many different parameters as there are many possible recommendations and outcomes in targeting. A basic recommendation engine is possible without complex models. When it comes to customer clustering unsupervised ML algorithms can be used because they are likely to find new insights that were completely not known before. Semi-structured data requires the usage of text analytics in order to analyze the content of a customer mail. When it comes to analyzing the content of phone calls, speech to text algorithms and text analytics have to be applied. In case of an event driven application (real-time marketing), (near) **real time processing** of the data is required. If the goal is only to improve existing marketing campaigns then **batch-oriented** processing is sufficient.
Security and Privacy	**Personally Identifiable Information (PII) used?**	Yes
	Highly sensitive data used?	No, only in case of enrichment with social media or other external data additional risks arise.
	Governance, Compliance & Audit	The customer has to agree to a processing of his data since PII is involved. Above all there are opt-in issues (can data from a customer in health insurance be used for advertising life insurance?). Furthermore a consent from the customer is needed for receiving advertisements/offers (Werbeeinwilligungserklärung). Compliance with BDSG and GDPR is required, since PII is being used here.
Organizational & Business Requirements	**Knowhow**	n.a.
	External Partners	Any provider of external data for enriching the basis of the targeting models - APIs are needed to be able to connect with them. For acquiring new customers external data providers (Acxiom, Payback) are vital.

	Other Business Challenges	Models have to take into account that a customer shouldn't get too many offers (no spam) for avoiding negative customer experience. Furthermore clerks working in the back office should receive trainings on how to sell products based on automatically generated suggestions (e.g. a Next Best Offer).
Other Big Data Challenges		**The models have to prevent spurious correlations and require long time for training.** For getting a 360-degree view on the customer, data integration and bypassing system divisions is vital. **Special case:** Analysis of contracts that contain several insured persons by comparing addresses and family names (cleansing needed here, as this combination isn't always correct or unambiguous). This can be done either with a Rules Engine or by setting up an additional analytical model. Precise targeting requires a broad data basis, which currently isn't available in most insurance companies. The integration of existing data sources and enrichment with external data in a data lake are of paramount importance for achieving a segment-of-one view on the customer. Above all the lack of traffic on the company websites is a challenge for delivering enough data for a precise targeting based on click-streams. Using standard enterprise software could be a possible solution here since Next Best Action, Next Best Offer or Customer-360-Degree View applications are already available from many providers. The technical difficulties would then be mostly reduced to data integration issues.

Table 4.2: Targeting requirements

Source: Own considerations and [1], [61], [62], [63], [64], [93].

4.1.2 Internal Processes

Use Case Title		Fraud Detection
Description		Through analyzing claims and customer data, fraudulent claims can be identified. Therefore models have to identify patterns in settled claims (analytical models based on historical data) in order to check new incoming claims. Another possibility is to apply prescriptive models for identifying new patterns in fraudulent claims that are not known so far.
Big Data Characteristics	**Data Source**	PII Data on customers, historical claims data (who repaired damages, damage costs or price for fixing it, place where the accident/incident happened, invoices), customer's contract/policy information, customer interaction data (phone calls, mails, letters), information on who is a known submitter of fraudulent claims (e.g. from the Insurers association GDV), new submitted claims (also if available images of the accident/incident/damaged property), business rules, social media data (to detect anomalies when comparing claims data with a person's real life).
	Volume	Given the large amount of claims being processed and the data coming with it, the data volume is to be regarded as high.
	Velocity	The submitted claims have to be **streamed** in order to enable near real-time processing of the claims (please refer to the claims automation use case for more information on this).
	Variety	Because of many different data sources, an integration in a **data lake** is needed (landing zone from where data can be accessed by ML models). Especially the integration with legacy systems poses a great challenge here. Above all the real-time integration (since streaming is involved) in the landing zone is quite difficult to implement. The data lake should be the one used for the claims automation Use Case.

Big Data Science	**Veracity and Data Quality**	Semi- and unstructured data requires **data cleansing** (particularly data from customer communication and input for photo forensics - if not already cleansed in the claims automation Use Case). Structured data from claims and transactional systems should have high quality in case respective processes to ensure this do exist in transactional systems. If this data comes from the claims management data lake then no cleansing is needed here. Structured data from other internal systems should nevertheless go through a cleansing process although some steps can be omitted when compared to the cleansing of external data. Ensuring high data quality is of paramount importance here since otherwise the number of false positives will increase heavily.
	Presentation and Operationalization	In case a claim is identified as fraudulent, a clerk from the anti-fraud department has to be notified for checking the specific claim manually before contacting the customer. In case of a white claim the result has to be automatically passed to the Claims Automation application for continuing the settlement process.
	Data Types	Data on customers from customer databases is structured, as is a part of the incoming claims data and the historical claims data. The data from written communication and from claims description is semi-structured and data from phone calls and accident/damage images is unstructured.

	Data Analytics	Conventional fraud detection does rely on a set of business rules for detecting suspicious claims. ML models are used for identifying patterns in the claims data, which is enriched with customer and external data. The more parameters and data is used for detecting patterns in structured data, the more complex the respective models do get. Text analytics and photo forensics need very complex analytical models (see section Other Big Data Challenges for this). The analysis of networks consisting of who settled the claim, accident locations, who was involved in the accident, who repaired the damage, etc. provides a good basis for pattern recognition. For storing the networks a **graph database** would make sense. The data has to be **processed in (near) real-time** for being able to settle claims as fast as possible. Two approaches are possible: •Supervised learning: based on historical data use known patterns for analyzing newly incoming claims and detecting fraudulent claims among them. •Unsupervised learning: use historical and newly incoming claims for finding patterns in claims that have not been known so far.
Security and Privacy	**Personally Identifiable Information (PII) used?**	Yes
	Highly sensitive data used?	Depends on company department: in life and health insurance highly sensitive health data is being used. This leads to high requirements concerning IT-Security and Identity&Access Management.
	Governance, Compliance & Audit	Customer has to agree to a processing of his data since PII is involved, however this should be covered in the claims settlement process. No explicit customer consent for data usage for fraud detection is needed. GDPR poses no restrictions on automated fraud analytics, so no compliance issues concerning "legally binding decisions" arise here. Usage of social media is also covered by GDPR if it follows the purpose of detecting fraudulent claims. However since PII is used, compliance with BDSG and GDPR has to be ensured.
Organizational & Business Requirements	**Knowhow**	n.a.
	External Partners	Providers of data on known fraudsters, e.g. the GDV (association of insurance companies in Germany)

	Other Business Challenges	As long as the models are not sufficiently trained, claims that are declared as fraudulent must be analyzed manually in order to avoid false positives and thus giving the customer a negative experience without a reason. As soon as the models are trained properly, the number of false positives will decline. Another issue is that only a few claims are entirely fraudulent, i.e. they are completely made up. In most cases real claims are filed with slightly increased invoice sums. This raises the question whether the high investments needed for implementing the whole Use Case will eventually pay off.
Other Big Data Challenges		Text analytics (analyzing content of mails, letters, claim descriptions), natural language processing (sentiment analysis in calls) and photo forensics (analyzing images of accidents/damages) are all technically very difficult to implement. The latter even requires the usage of neural networks do deliver proper results in image recognition. The models for detecting patterns have to be slightly adjusted for the respective country when being used internationally in a large insurance company. Above all the models take time to get precise (requires time-consuming training) and deliver proper results when detecting fraudulent claims.

Table 4.3: Fraud Detection requirements

Source: Own considerations and [1], [61], [62], [64], [65], [67], [93].

Use Case Title		**Claims Automation**
Description		By extending the Rules-Engine based approach in claims settlement with text analytics and ML the claims settlement process can largely be automated. A combination with the fraud detection use cases makes sense.
Big Data Characteristics	**Data Source**	Customer data (PII) from customer databases, historical claims data (especially for training the ML models) or newly incoming claims (description of operation, doctor's or surveyor's notices and reports, invoices), business rules, customer's contract data (particularly content of insurance policy).
	Volume	Given the large amount of claims being processed and the data coming with it, the data volume is to be regarded as high.

	Velocity	Newly incoming claims have to be **streamed** to the landing zone for enabling near real-time claim settlement.
	Variety	Many different data sources require an integration in a **data lake** (landing zone from where data can be accessed by ML models). Especially the integration with legacy systems for accessing historical claims poses a great challenge here. Above all the real-time integration (as streaming is used) in the landing zone is very difficult to implement.
Big Data Science	**Veracity and Data Quality**	Semi- and unstructured data requires **data cleansing** (particularly data from customer communication, doctor's or surveyor's notices and images in the claims). Structured data from historical claims and transactional systems should have high quality in case respective processes ensure these do exist in transactional systems. Structured data from new claims definitely has to pass **data cleansing** processes too. Internal data has to pass a cleansing process as well, although when comparing it with the cleansing of external data some process steps can be omitted. High data quality is of paramount importance here since otherwise the claims will be settled in a wrong way causing financial losses for the company.
	Visualization	In case a claim is settled without any objections, the pay-out has to be triggered and the customer informed automatically. In case there are any objections, a clerk should analyze the claim before triggering a possible partial pay-out and eventually notifying the customer. This manual analysis is particularly important in the beginning when the models are not yet entirely trained. After a certain time these manual checks can be reduced to a few random samples.
	Data Types	Data on customers from customer databases is structured, as is a part of the incoming claims data and the historical claims data. The data from written communication and from claims description is semi-structured and data from phone calls and accident/damage images is unstructured.

	Data Analytics	First an incoming claim has to be properly recorded. Besides conventional data transformation processes this includes applying text analytics for analyzing the claim's content. Afterwards a set of business rules and an analytical model are used for comparing a claim with the content of the customer's insurance policy to determine whether the claim will be paid and to what extent. The comparison-component is trained with ML algorithms using a large amount of historical claims data. If the insurance company does not have a proper product model, the content of insurance policies has to be analyzed with text analytics too. Additionally image analytics (for images of accidents/damages – however difficult to implement as neural networks are required) and natural language processing (analyzing phone calls with the customer) can be applied. The entire application can be extended through a ML component that learns with each settled claim whether it was settled correctly and thus strongly improves the system's accuracy (reinforcement learning approach). The data has to be **processed in (near) real-time** for being able to settle claims as fast as possible. Since claims are processed in micro-batch style, Apache Spark could be a possible processing framework that still guarantees reasonable latency whilst being able to process high workloads.
Security and Privacy	**Personally Identifiable Information (PII) used?**	Yes
	Highly sensitive data used?	Depends on company department: in life and health insurance highly sensitive health data is being used. This leads to high requirements concerning IT-Security and Identity&Access Management.
	Governance, Compliance & Audit	The customer has to agree to a processing of his data since PII is involved (actually covered during under-writing and contracting). Since GDPR does not allow entirely automated deci-sions on claim settlement, in case of a result that the claim will not be settled, it has to be ensured that it is checked by an agent before contacting the customer. Compliance with BDSG and GDPR has to be ensured because of PII usage.
Organizational & Business	**Knowhow**	n.a.
	External Partners	None needed

Requirements	**Other Business Challenges**	A large automation program leads inevitably to job cuts in the claims settlement department. This results in a political/organizational risk, which has to be dealt with by setting up training programs and mitigating the ramifications for those affected by the job cuts.
Other Big Data Challenges		The main challenge here is the data integration; the models needed for text analytics are regarded as simple by data science experts. However for a full implementation of claim automation, the Fraud Detection Use Case has to be up and running, which requires very complex models. Additionally, system divisions have to be bypassed.

Table 4.4: Claims Automation requirements
Source: Own consideration and [1], [2], [64], [70], [93]

Use Case Title		**External Data for Optimized Pricing and Risk Assessment**
Description		The pricing processes are enriched with external data from different providers for an improved risk assessment. The data actually used depends on the respective product being priced.
Big Data Characteristics	**Data Source**	Depends on product priced: examples include: -Provider of weather data e.g. for flood or house insurance -Provider of geospatial data for insurance against wildfires or house insurance -Provider of customer segment data for enriching targeting in order to acquire new customers -Information on houses (milieu-data: construction structure, information on house age, etc.) for pricing in house insurance.
	Volume	Depends on product priced.
	Velocity	For real-time pricing engines, **streaming** the data is required, in other cases batch-loading the data to the landing zone would be enough (these will be the case in most scenarios).
	Variety	Given the high number of different data sources, storage and integration in a **data lake** is required. The data lake will serve as the central landing zone from where the risk assessment processes can extract the data for pricing the respective product.

Big Data Science	**Veracity and Data Quality**	Depends on the contract with the external data provider and his input: in the best case the data provided already has a very high quality and is ready to use without cleansing. For quality assurance purposes the data sets should nevertheless pass a cleansing and transformation process. However it does not have to be as extensive as when dealing with raw external data.
	Presentation and Operationalization	Depends on the product which is priced. However as this Use Case's goal is simply to supply the pricing and risk assessment process with additional parameters, no visualization is necessarily needed. A possible visualization could however include the impact the external parameters have for the whole risk model.
	Data Types	Depends heavily on the external data and the contract with the provider, in the best case the provider already takes cleansing, clustering and structuring the data upon him.
	Data Analytics	Depends on the product priced. Adaptions in existing pricing and risk assessment processes are needed, however **no complex ML models are required for the Use Case on its own.** The pricing processes can however be extended through a ML component which takes care of analyzing which parameters are truly relevant for correctly pricing a product.
Security and Privacy	**Personally Identifiable Information (PII) used?**	As long as it is used only in P&C or industrial insurance no, for life and health insurance maybe yes.
	Highly sensitive data used?	Only when used in life and health insurance.
	Governance, Compliance & Audit	As long as no PII is involved, no specific issues do arise here.
Organizational & Business Requirements	**Knowhow**	n.a.
	External Partners	The providers of the external data sets.
	Other Business Challenges	An adaption of the existing pricing and risk assessment processes is needed concerning the new parameters from the external data.

Other Big Data Challenges	No complex models involved, the only challenge is the data integration, which can be solved using a data lake. Another challenge would be defining an API for having a connector to the external data sources (especially when it comes to streaming the data the connector issue is difficult - a possible solution could be Kafka)

Table 4.5: External Data for Pricing requirements
Source: Own considerations and [71], [64], [93]

Use Case Title		**Enterprise Architecture and Business Process Analysis based on Monitoring Data**
Description		By collecting monitoring and tracking data, the insurance company can analyze its Enterprise Architecture and the applications for possible reasons leading to outages or too long response times.
Big Data Characteristics	**Data Source**	Every system and application that is part of the enterprise landscape and can provide monitoring data of it. Additionally web-tracking data from the company's retail websites can be collected. Data from users who visit and navigate through the insurer's retail website.
	Volume	Comparably high, if all enterprise applications are involved. Concerning tracking data, it depends on the amount of users of a website (today it is rather low for retail websites).
	Velocity	Monitoring and web tracking data will come in real-time, i.e. it has to be **streamed** in order to be able to derive optimal insights as fast as possible from it**.**
	Variety	Given the high amount of different applications existing, the number of data sources is high. As each application will generate different monitoring and tracking data, the variety is high as well. It makes sense to integrate the data in a **data lake** so that other Use Cases (e.g. Targeting) can use the tracking data for their purposes as well. Both monitoring and web-tracking data can be stored in NoSQL databases optimized for storing logs (e.g. Cassandra or HBase).
Big Data Science	**Veracity and Data Quality**	Since the data comes in in a raw state it has definitely to be **cleansed** before it can be pipelined into an analytical model.

	Presentation and Operationalization	The current state of the Enterprise Architecture and the application landscape has to be visualized using several dashboards. In case of an outage the responsible departments have to be notified about the application/subsystem/service/etc. the outage was caused. Tracking data can be visualized on dashboards showing user's navigation history and dropout rates and other KPIs.
	Data Types	Tracking and monitoring data is structured (comparable to sensor data in the IoT Use Cases).
	Data Analytics	Data has to be processed at **(near) real-time speed**, so that the users of the monitoring system are always up to date about the state of the application landscape. The algorithms for analyzing the monitoring and tracking data don't need to be complex since it is mostly a time series analysis that is required for the core Use Case. The algorithms required for detecting underwriting fraud, developing new individualized products and providing product recommendations are more complex but not directly part of this Use Case.
Security and Privacy	**Personally Identifiable Information (PII) used?**	Not for the core Use Case, however depending on the application tracked it can make sense to enrich the monitoring data with PII. This would however lead to compliance and data protection issues. Tracking data has to be anonymized.
	Highly sensitive data used?	No.
	Governance, Compliance & Audit	If no PII is involved, no specific issues arise here. If yes, compliance with the respective data protection laws (such as BDSG or GDPR) is required.
Organizational & Business Requirements	**Knowhow**	n.a.
	External Partners	None needed.
	Other Business Challenges	Currently the business value particularly lies in the reduction of maintenance costs. However other Use Cases can also use the analysis results or especially the tracking and monitoring data making this Use Case rather a foundation or data source for other ones.

Other Big Data Challenges	Particularly the integration poses high challenges since especially in large companies there is a high variety of applications with many legacy systems being available. Integrating them into a holistic tracking and monitoring environment is very difficult. The various possibilities for deploying the applications (either in the cloud or on-premise) add further complexity to implementing the monitoring. As both tracking and monitoring are quite common applications, many standard software solutions from commercial vendors do exist and could also be used for implementing parts of the Use Case.

Table 4.6: Enterprise Architecture Analysis Based on Monitoring Data Requirements.
Source: Own depiction

4.1.3 IoT in Property & Casualty

Use Case Title		**Telematics**
Description		By analyzing the driving behavior of a customer, the insurance company can offer discounts on the premium to careful drivers. Additionally the insurer can offer services to customers in car insurance (e.g. information on fuel consumption, search for parking lots, etc.)
Big Data Characteristics	**Data Source**	Telematics device or smartphone for the sensors: -speed -braking/accelerating -Roads driven (motorway vs. city) -What time (during day or night) For offering services integration with other (external) data it is required.
	Volume	Large amount of sensor data to be processed in order to analyze driving behavior (more than 15 billion kilometers driven).
	Velocity	Sensor data has to be **streamed** as it is generated at real-time speed.
	Variety	Low variety, only sensor data from one main source needed for the core Use Case.
Big Data Science	**Veracity and Data Quality**	The sensor data needs **cleansing** before the algorithms calculating the driving scores can use it.
	Presentation and Operationalization	The customer gets shown his driving score and the discount he can get for it.
	Data Types	Sensor data is entirely structured

	Data Analytics	The **models** needed for calculating the driving score are **not very complex**, require basic algorithms and are already known. The processing of the streamed sensor data can be done **batch-oriented** so that the customer gets his score a few times a day. Real-time processing can be used as well for providing an improved customer experience, however it is not really necessary. The entire application can be extended with an additional component that takes the risk scores into account when calculating new premiums for both existing and new customers.
Security and Privacy	**Personally Identifiable Information (PII) used?**	Not directly: only through analyzing the routes which are driven one can determine the places where a person lives/works/etc. However customers might feel themselves tracked what could lead to a negative customer perception.
	Highly sensitive data used?	No
	Governance, Compliance & Audit	In case of an analysis of the routes driven it has to be ensured that the customer agrees to such a way of processing his data.
Organizational & Business Requirements	**Knowhow**	n.a.
	External Partners	Car-sharing platforms and car manufacturers can be partners, however not for implementing the Use Case but for accessing a large group of customers. Above all, they can be partners for offering services like for instance searching for a parking lot or providing information on fuel consumption since they have a much better data basis for this than an insurance company.
	Other Business Challenges	**The correlation between driving style and the risk of an accident is highly controversial.** This raises the question whether in the longer term offering discounts to careful drivers will be profitable for the insurance companies.
Other Big Data Challenges		The only challenge is collecting the large amount of sensor data and setting up the sensors. However the latter can be implemented by a sensor provider company. Additionally the existing pricing processes in car insurance have to be adjusted to take into account the telematics parameters.

Table 4.7: Telematics requirements. Source: Own considerations and [61], [74], [75].

Use Case Title		Industrial Insurance
Description		An industrial customer deploys a monitoring system for his manufacturing plants/factories. This makes it possible to carry out predictive maintenance so that eventually the risk of a production plant outage is reduced. From the viewpoint of the insurer, this lowers the risk in an operations interruption insurance and is highly useful because individual sums in case of such an outage are very high. The insurer has two possibilities: -Give a discount for having such a monitoring system -Collect the monitoring system's data for making individual risk pricing possible. This second option is regarded here.
Big Data Characteristics	**Data Source**	Customer's production plant with parameters and sensor data from the respective monitoring system such as: -temperature -pressure -size of products manufactured -time needed for a single work station
	Volume	Very high: a large amount of data is collected in each production plant, when combined from different customers and plants, this adds up to a very high data volume. All the data from the different sources has to be stored in a **data lake**, which then works as the landing zone and source for the analytical models predicting the risk of an outage.
	Velocity	Streaming the data is not necessary needed, a **batch-load** from time to time would be enough. However this batch-load would then contain a very high amount of data.
	Variety	Given the high number of various factories and plants being insured and the vast amount of sensors in each plant, the variety of data is very high. The various sources have to be integrated in a data lake, at least from the same customer. Before integrating data from different customers, their agreement is needed.
Big Data Science	**Veracity and Data Quality**	Since the data is provided by the customer, it is to be expected that it already has a good quality and should be ready for being used in the predictive models. Nevertheless it should pass a data cleansing process for quality assurance. Additionally data transformations have to be applied depending on the solution of the API issue (see also the Other Big Data Challenges category).

	Presentation and Operationalization	The underwriter needs to receive a risk report for the manufacturing plant containing the explanations for each component's risk analysis. The risk score has to be included into the pricing process for the operations interruption insurance.
	Data Types	Depends on the sensors: the majority of sensor data is structured and the customer will try to bring his unstructured data into a form possible to work with. However it is possible that a customer also sends unstructured data sets.
	Data Analytics	Complex models are needed for predicting the future risk of an outage in a production plant. The models have to take into account many possible outcomes and scenarios which can lead to an outage of the plant since factories are very complex systems. **In the best case the models can rely on comparable data from other plants of the same type.** However this depends on the customer's readiness to make his data available for comparisons with other customers' data (see also the Governance & Compliance section for more details). The data can be processed batch-oriented.
Security and Privacy	**Personally Identifiable Information (PII) used?**	No
	Highly sensitive data used?	Depending on the manufacturing plant. In any case, the data used here gives a deep insight into the functioning of a production plant of the customer which poses high IT-Security and Identity&Access Management requirements for the system where the data is stored and analyzed.
	Governance, Compliance & Audit	It has to be made sure that the customer agrees to using his data for comparisons with the data from other industrial companies if the insurance company wants to carry out such analyses. It has to be made sure that the high requirements for Access Management to the data basis are ensured. A certification from an audit company can help establish trust, particularly for new customers.
Organizational	**Knowhow**	n.a.

& Business Requirements	**External Partners**	The customer should be regarded as a partner here, since he is the provider of the data basis. Above all the manufacturers of the production plants or large machines can be partners as well. For the same industrial insurer can insure instance every gas turbine made by Siemens or GE through a sensor based operations interruption insurance. These manufacturers can also be sold a liability insurance in case the plant or turbine breaks down in spite of predictive maintenance.
	Other Business Challenges	Existing pricing and risk assessment processes have to be adjusted as the new approach envisages individualized risk pricing. Above all there is the issue whether the whole Use Case is eventually profitable for the insurer since the required investments are very high. Some experts believe that the Use Case is too complex for an insurance company as it never will have as much knowhow about manufacturing plants as the industrial customers.
Other Big Data Challenges		Since the real Use Case is implemented by the customer, the only challenge is the data integration from the many different sources in a data lake and the conception of models predicting the risk of a future outage. The key point is the definition of an API for receiving the data from the different customers. Given the high variety of the data sources from the different customers, a custom-made API for each of them could be necessary.

Table 4.8: Industrial Insurance requirements.
Source: Own considerations and [64], [73], [93]

Use Case Title		Smart Home
Description		The insurer equips his customers in home insurance with different sensors (e.g. a smoke detector). Using a sensor a damage can be prevented or mitigated (e.g. the fire brigade is alarmed in time so that the whole house does not burn off). This lowers the risk for the insurance company and makes it possible to give discounts on the sensor based home insurance products.
Big Data Characteristics	**Data Source**	The respective sensor: -Smoke detectors (fire insurance) -Water pressure on water pipes (sprinkler leakage insurance) -Cameras and intrusion detection systems (anti-burglary insurance)
	Volume	A large amount of sensor data is collected in each home.
	Velocity	**Real-time streaming** to a landing zone is needed for being able to react instantly in case of fire or water incidents.
	Variety	For a single house insurance product (e.g. fire insurance) the variety of data is quite low.
Big Data Science	**Veracity and Data Quality**	The raw data coming from the sensors requires **data cleansing** before it is possible for the algorithms to analyze it. The cleansing has to be applied as soon as the data comes in hence the Use Case is a velocity application.
	Presentation and Operationalization	Automatic alarming of the fire brigade, the police or a service technician (for fixing the damaged water pipe).
	Data Types	Sensor data is structured.
	Data Analytics	The algorithms for calculating the critical threshold level are not complex. However as it is essential to detect anomalies as fast as possible, the incoming sensor data has to be processed at **(near) real-time** speed. Apache Storm could be a possible solution since the sensor data consists of single data items and the Use Case is a velocity application where latency is very important. The calculation of the correct discount on the premium for having such a sensor is an actuarial issue and has to be performed by the respective department.

Security and Privacy	**Personally Identifiable Information (PII) used?**	No, however people can feel surveilled in their own homes (particularly in Germany).
	Highly sensitive data used?	In case of camera usage yes, otherwise no.
	Governance, Compliance & Audit	Since people are afraid of being tracked and surveilled, Identity&Access Management as well as IT-Security standards have to be very high for the entire system. It also has to be ensured that employees don't access particularly cameras or sensors unauthorized (I&AM issue).
Organizational & Business Requirements	**Knowhow**	n.a.
	External Partners	The sensor manufacturer, additionally a partner company can carry out the data collection from the sensors. Furthermore companies for fixing damaged property can be partners.
	Other Business Challenges	Some experts believe that the added value is rather low in insurance since the damages can't be prevented entirely. The only exception is sprinkler leakage insurance: if the water supply is turned off immediately as soon as an anomaly is detected and a service technician fixes the pipe before the water damage can happen (water damages account for the majority of paid out sums). Finally the sensors are expensive and with the added value being unclear, it is not sure whether the Use Case will be eventually profitable for the insurer. There is more potential with industrial customers, e.g. owners of large apartment complexes.
Other Big Data Challenges		The main challenge is the collection and processing of a large amount of sensor data in real time. Furthermore, currently no standards for sensors do exist and the sensor environment is very heterogeneous. Above all many customers, particularly in Germany, will not be ready to install such sensors in their home because of privacy and tracking issues. When insuring for instance office buildings, these problems do not arise. Another point is the question whom does the sensor data belong, in case the customer already has sensors at home. If it is the sensor provider, then the use case is unlikely to be profitable for the insurance company.

Table 4.9: Smart Home requirements.

Source: Own considerations and [61], [76]

4.1.4 Smart Health and Smart Life

Use Case Title		Health Insurance Based on Wearables Data (discounts)
Description		The insurer distributes wearables for analyzing whether the customers do sports/are active. If a customer passes a certain threshold, he is awarded a discount on his premium for his health insurance.
Big Data Characteristics	**Data Source**	The wearable device is the data source and collects data such as: -number of steps -pulse -calorie consumption -other parameters Additionally health data such as previous diseases and general customer data should be taken into account. Generally the number of parameters for a basic implementation of this Use Case can be far lower than for the Health Services Use Case. For instance Generali does not take much more into account than the number of steps for calculating the discounts for the Vitality product.
	Volume	Large amount of sensor data coming from different sensors, which has to be integrated in **a data lake or central landing zone**. The development of the APIs for getting the data to the landing zone is complex as well.
	Velocity	Data should be **streamed** for real-time calculation of scores/discounts.
	Variety	Although several parameters are tracked, the number of sources isn't too high.
Big Data Science	**Veracity and Data Quality**	The raw data coming from the sensors requires **data cleansing** before it is possible for the algorithms to analyze it. If data from internal systems is used too, then it should go through a cleansing process although some steps can be omitted when compared to the cleansing of the sensor data.
	Presentation and Operationalization	The customer has to receive an overview (either in an app or on a webpage) with his activity and the discounts he will receive for it.
	Data Types	Sensor data is structured.

	Data Analytics	The algorithms required here are not very complex, since they only have to calculate scores for the customer's physical activity. The calculation of the correct discount on the premium for being sportive is an actuarial issue and has to be performed by the respective department. It is possible to have several discount stages depending on the level of physical activity. The processing can be done **batch-oriented.** A real-time processing approach could however provide a new customer experience so that the customer can get instantly his current health status and the associated discount after a workout. Further analytical models are possible to analyze the health of the customer and offer him consulting and services on how to improve it (refer to the next Use Case for this).
Security and Privacy	**Personally Identifiable Information (PII) used?**	Yes
	Highly sensitive data used?	Yes, since health data is very sensitive. This leads to high requirements for Identity&Access Management as well as IT-Security for the entire system.
	Governance, Compliance & Audit	The customer has to agree to the processing of his data, because health data and possibly PII are involved. It has to be assured that the standards for Identity&Access Management and IT-Security are granted. If using PII in the calculation of premium discounts, compliance with BDSG and GDPR is required. GDPR however does not directly pose restrictions on the processing of health data.
Organizational & Business Requirements	**Knowhow**	n.a.
	External Partners	The manufacturers of the device should certainly be acquired as partners. Furthermore cooperations with expert companies on real-time data analysis are possible here.

	Other Business Challenges	**The correlation between doing sports and the risk of an illness is highly controversial.** This raises the question whether in the longer term offering discounts to sportive customers will be profitable for the company. Nevertheless such a product is able to create growth since it is attractive to new customer segments. On the other side some customers (particularly in the German market) might feel surveilled and tracked so that such a product would lead to negative customer perception. Additionally customers that are very sporty have an increased risk of sport injuries. Above all the entire model is highly questionable when it comes to insurance solidarity: not sportive customers have to pay for the sportive ones.
Other Big Data Challenges		The main challenge is the collection and processing of a large amount of sensor data in real time. The analytical models used here are not to be regarded as complex.

Table 4.10: Wearables (discounts) requirements.
Source: Own considerations and [64], [76], [79]

Use Case Title		**Health Services Based on Wearables Data**
Description		The insurer distributes wearables for analyzing the customer's health. Depending on the results, the insurer offers health services such as consultation concerning nutrition, work-out plans and so on. Thus the insurer can proactively take care of the customer's health.
Big Data Characteristics	**Data Source**	The wearable device is the data source and collects data such as: -number of steps -pulse -calorie consumption -calorie intake (however this is difficult to analyze, e.g. through food images - customer needs to write down the specific product) -sleep records -blood pressure -other parameters Additionally health data such as previous diseases and general customer data should be taken into account.

	Volume	Large amount of sensor data coming from different sensors, which have to be integrated in a **data lake or landing zone**. The development of the APIs for getting the data to the landing zone is complex as well.
	Velocity	Data should be **streamed** for real-time analysis of the customer's health and being able to show the customer his current state.
	Variety	Although several parameters are tracked, the number of sources isn't too high. However depending on the device policy (which devices are supported) new integration and data processing issues do arise. The reason is the different input and format the various devices do have (e.g. Fitbit vs. Garmin).
Big Data Science	**Veracity and Data Quality**	The raw data coming from the wearable's sensors requires **data cleansing** before it is possible for the algorithms to analyze it. Internal data should go through a cleansing process as well although some steps can be omitted when compared to the cleansing of the sensor data.
	Presentation and Operationalization	The customer needs to receive an overview of his health status and the respective recommendations (services). This can be via a mobile application or a website.
	Data Types	Sensor data is structured.
	Data Analytics	Given the high number of possible outcomes (e.g. different nutrition programs or work-out plans) the models required here are difficult and several of them are needed. The models can be based on Python and R since common ML algorithms are sufficient here (e.g. with decision trees) with the data processing being done batch-oriented. For real-time processing a framework like for instance Spark would be required. For providing health consultation, stream processing is not needed, but for showing the customer his current state in real-time it is.
Security and Privacy	**Personally Identifiable Information (PII) used?**	Yes

	Highly sensitive data used?	Yes, since health data is very sensitive. This leads to high requirements for Identity&Access Management as well as IT-Security for the whole system.
	Governance, Compliance & Audit	The customer has to agree to the processing of his data, since both PII and health data are involved. It has to be assured that the standards for Identity&Access Management and IT-Security are granted. Additionally compliance with BDSG and GDPR is required because of using PII (not using it would make the recommendation models rather imprecise). GDPR however does not directly pose restrictions on the processing of health data.
Organizational & Business Requirements	**Knowhow**	n.a.
	External Partners	The manufacturers of the device should certainly be acquired as partners. Furthermore a cooperation with an expert on real-time data analysis is possible here. Above all an entire outsourcing of the Use Case to a cooperation partner who offers the consulting services (e.g. Fitbit) is possible.
	Other Business Challenges	Currently there might still be no market for such a product (particularly in Germany). The reason is that some customers might feel surveilled and tracked so that such a product would lead to negative customer perception. Nevertheless such a product is able to create growth since it is attractive to new customer segments and by offering services the marketing effect is higher than by simply offering discounts. However the product does have high potential for the future since it enables the insurer to transform health insurance towards a proactive healthcare approach. There is also the question whether the whole product is offered by the insurer or a new legal entity (this would lead to waiving the historical health data but would grant better customer perception of such a new product).

Other Big Data Challenges	The models required here are more complex and sophisticated than in the discounts Use Case what makes the implementation more difficult. Additionally, more parameters have to be taken into account in order to provide the customer with correct and precise recommendations on how to stay healthy. Nevertheless, the technology concerning data collection and storage behind the Use Case is basically similar to the discounts Use Case.

Table 4.11: Health Services Based on Wearables requirements.
Source: Own considerations and [61], [76], [79], [80]

Use Case Title		**Disease Management**
Description		The insurer collects and analyses various health data sets about the customer. Based on this analysis the insurer can derive the risk of future illnesses for the customer and offer him advice on how he can prevent this illness thus proactively taking care of the customer's health.
Big Data Characteristics	**Data Source**	Data from various sources such as: -Electronic Medical Record (EMR) -wearable data -blood pressure illnesses so far -customer data (age, sex, weight, profession, other PII) -previous claims -ICD-Codes from the WHO -open source health data -data from customer check-ups if paid for by the insurance company (yearly or every two to three years) -other parameters
	Volume	Very high as lots of health and customer data is being collected and processed.
	Velocity	Depends on the source, **batch-oriented** (new diagnosis in his EMR) and **streaming** (wearable data) will be needed.
	Variety	The data comes from many different sources (customer databases, EMR, wearables, etc.) and has to be integrated in a **data lake**. This data lake is the landing zone where the data can be accessed by predictive models.

Big Data Science	**Veracity and Data Quality**	Data from internal systems (customer and claims data) or the EMR does have high quality, the data from wearable devices will have to be **cleansed.** Depending on the quality of the Claims Management processes and the data there, cleansing needs to be applied as well hence the Use Case requires excellent data quality for the models. One possibility is to map medical diagnoses from free texts written by doctors to ICD codes through text analytics.
	Presentation and Operationalization	The customer needs to receive an overview of his health status. This can be via a mobile application or a website. In case of an increased risk disease the customer needs to receive an update on this including the recommendations on how he can prevent this particular illness.
	Data Types	Data from internal systems or the EMR is structured as is data from wearables. However when using external sources, semi- or unstructured data may appear as well. This would require **data cleansing**.
	Data Analytics	The required models for predicting the risk of a future illness and for finding the fitting consultation on how the customer can stay healthy are tremendously complex. They need to take into account many different outcomes (both the various diseases ICD-codes contains and the number of possible ways prevent the illness). It has also to be analyzed how different parameters do affect/influence each other when predicting future illnesses. Prescriptive analytics can be used for discovering patterns leading to illnesses that have not been known so far. The models can be based on Python and R since common ML algorithms are sufficient here (e.g. with decision trees). A **batch-oriented** analytical processing can be regarded as sufficient.
Security and Privacy	**Personally Identifiable Information (PII) used?**	Yes

	Highly sensitive data used?	Yes, since health data is very sensitive. This leads to high requirements for Identity&Access Management as well as IT-Security for the whole system.
	Governance, Compliance & Audit	The customer has to agree to the processing of his data, since both PII and health data are involved. It also has to be ensured that the predictions and recommendations are very precise - if not, this could lead to a highly negative customer experience and a bad reputation. It has to be assured that the standards for Identity&Access Management and IT-Security are granted. Additionally compliance with BDSG and GDPR is required because of using PII. GDPR however does not directly affect the processing of health data.
Organizational & Business Requirements	**Knowhow**	n.a.
	External Partners	Since there is a lack of data and it will be needed for the models being able to generate precise and correct predictions, any data provider in the health sector would be helpful. This includes: pharmaceutical companies, universities/research institutions, clinics, etc. Furthermore the development of the models for predicting the diseases could be outsourced to experts in this field as well or done in close cooperation with them (e.g. a research institution or companies like IBM for the AI components). Partnerships in the field of Clinical Research are possible as well.
	Other Business Challenges	Many customers (particularly in the German market) are not willing to share their health data with an insurance company. This leads to a lack of data for the data basis without which the Use Case can't be implemented. However the Use Case holds tremendous potential for the future as more and more people are ready to share their data. Eventually it can form the basis for enabling telemedicine through the insurer (however this is currently forbidden in Germany). Above all the entire Use Case envisages a completely new approach to health insurance that will inevitably lead to restructuring the existing organization and the respective processes in the insurance company.

Other Big Data Challenges	The Use Case is very difficult to implement because of the lack of a data basis and the gargantuan efforts required when developing the ML algorithms for predicting the risk of future diseases. Therefore it will take long time and also high financial resources until the Use Case is fully implemented. Still experts in the field of Data Science regard it as feasible.

Table 4.12: Disease Management requirements.
Source: Own considerations and [61], [80], [81], [82], [83], [84]

Use Case Title		**Sensor-based Services in Life Insurance**
Description		The insurer distributes wearables among customers in life insurance for checking the customer's health. In case of an incident a relative or the ambulance are informed immediately.
Big Data Characteristics	**Data Source**	The wearable device is the data source and collects data such as: -pulse -body temperature -blood pressure -other parameters Additionally health data such as previous diseases and general customer data should be taken into account.
	Volume	Not too many sensors involved, so the volume can be regarded as medium.
	Velocity	**Streaming** the data to a landing zone is required for (near) real-time analysis of the data and instant reaction in case of an incident.
	Variety	Low, only a few sensors are required.
Big Data Science	**Veracity and Data Quality**	**Data cleansing** has to be applied to the data as soon as it comes in because the raw sensor data isn't ready for direct processing and pipelining into an analytical model.
	Presentation and Operationalization	In case of an incident a relative or the ambulance have to be informed automatically.
	Data Types	Sensor data is structured.

	Data Analytics	The algorithms required here are not complex, since they only have to check actual values vs. a defined threshold. The trigger for an incident is clearly defined. In fact the data processing/analysis part of the Use Case is rather rules-based than a Big Data Use Case. However the data has to be processed at (near) real-time speed. As the data stream consists of a large amount of single data items from the sensors and low latency is very important, Apache Storm could be a solution here.
Security and Privacy	**Personally Identifiable Information (PII) used?**	Not for the analysis of the threshold values needed for implementing the core Use Case, however customer data (age, previous illnesses, etc.) can be taken into account here as well.
	Highly sensitive data used?	Yes, since health data is very sensitive. This leads to high requirements for Identity&Access Management as well as IT-Security for the whole system.
	Governance, Compliance & Audit	If a customer has an incident and the system does not send an alarm/alert, then this could lead to lawsuits and a high reputational damage. Thus the system has to be properly tested and guarantee very high precision. Quality checks are required to ensure this. If the Use Case uses PII, compliance with BDSG and GDPR is required. Concerning the processing of health data, customer consent is needed.
Organizational & Business Requirements	**Knowhow**	n.a.
	External Partners	The manufacturer of the device should be acquired as a partner. The data streaming part could be outsourced too.
	Other business challenges	Most experts regard the Use Case only as an assistance service and not an own product that can't create enough added value for being profitable. Additionally, they don't believe that the customers will pay money for such a service, because a comparable product has already failed and wasn't wanted by customers.
Other Big Data Challenges		Based on the opinion of the experts, this isn't really a complex Use Case since it does not need any analytical models for predicting an outcome.

Table 4.13: Sensor-based Services in Life Insurance requirements.
Source: Own considerations and [40], [61]

4.2 Generic Requirements

After analyzing the Use Case specific requirements from the previous section, a number of generic requirements is derived from them. Some of the requirements listed in the following overview are based on the ones the NIST workgroup has found in their research and are also applicable for the Use Cases in the insurance sector. In total 33 different generic requirements have been identified. Besides naming a single generic requirement, the following table also shows which Use Case does need it – if a Use Case is in brackets, this means that it can implement the requirement, however it is not needed for the "core" Use Case.
All requirements are clustered similarly to the structure used by the NIST workgroup with a few extensions for providing a better overview [6]. The clustering looks as follows:

A. Data Source Requirements
B. Transformation and Analysis Requirements
C. Data Consumer Requirements
D. Privacy and Security Requirements
E. Lifecycle Requirements
F. Business and Organizational Requirements
G. Infrastructure and Capability Requirements

ID	Generic Requirement	Use Cases needing the Requirement
A: Data Source Requirements		
1	Needs to support reliable real time, asynchronous, stream loading to collect data from centralized or distributed data sources, sensors, or instruments. This includes collecting data in-motion.	•Churn Management •(Targeting) •Fraud Detection •Claims Automation •(External data for pricing) •Telematics •Smart Home •Smart Life •Wearables Discounts •Wearables Health Services •(Disease Management) •Monitoring
2	Needs to support slow, and high-throughput (e.g. batch loads) data transmission between data sources and the Big Data platform (e.g. transactional systems).	•Churn Management •Targeting •Fraud Detection •Claims Automation •External data for pricing •Industrial Insurance •(Wearables Health Services) •Disease Management

ID	Generic Requirement	Use Cases needing the Requirement
3	Needs to support diversified data content (semi- and unstructured data) ranging from text, document, graph, web, geospatial, compressed, timed, spatial, multimedia, simulation, and instrumental data.	•Churn Management •Targeting •Fraud Detection •Claims Automation •(External data for pricing) •Disease Management
4	Needs to support structured data from sensors, transactional or CRM systems, etc.	•Churn Management •Targeting •Fraud Detection •Claims Automation •External data for pricing •Telematics •Industrial Insurance •Smart Home •Smart Life •Wearables Discounts •Wearables Health Services •Disease Management •Monitoring
5	Needs to make sure that the data sources for a Use Case can be extended easily and quickly by adding new data sources.	all
B: Processing & Analysis Requirements		
1	Needs to support diversified ML frameworks (e.g.H2O or TensorFlow) and APIs for developing own models.	•Churn Management •Targeting •Fraud Detection •Claims Automation •Industrial Insurance •Wearables Health Services •Disease Management •(Telematics) •(External Data for Pricing)
2	Needs to support real-time, stream processing.	•Churn Management •(Targeting) •Fraud Detection •Claims Automation •Smart Home •Smart Life •Monitoring •(Telematics) •(Wearables Discounts) •Wearables Health Services

ID	Generic Requirement	Use Cases needing the Requirement
3	Needs to support batch-oriented analytic processing.	•Churn Management •Targeting •Industrial Insurance •Telematics •Wearables Discounts •Disease Management
4	Needs to provide capabilities for measuring and analyzing the performance of analytical or predictive models. Depending on the result models are required to be adaptable and/or enabled for retraining.	all
C: Data Consumer Requirements		
1	Needs to support diversified output file formats for visualization and reporting.	•Churn Management •Targeting •Claims Management •(Wearables Health Services) •(Disease Management)
2	Needs to support visual layout for results presentation.	•Churn Management •Targeting •Fraud Detection •Claims Automation •Industrial Insurance •Telematics •Wearables Discounts •Wearables Health Services •Disease Management •Monitoring
3	Needs to support rich user interface for using browser visualization tools.	•Churn Management •Targeting •(Fraud Detection) •(Claims Automation) •(Industrial Insurance) •Wearables Discounts •Wearables Health Services •(Disease Management) •(Smart Life) •Monitoring

ID	Generic Requirement	Use Cases needing the Requirement
4	Needs to support streaming/loading results to clients or other applications for post-processing (e.g. payment systems or pricing processes).	•Fraud Detection •Claims Automation •External data for pricing •Industrial Insurance •Smart Home •Smart Life •Wearables Discounts •Disease Management
5	Needs to automatically trigger customer communication (through mail, e-mail or a push notification in a mobile application)	•Churn Management •Targeting •Claims Automation •Telematics •Smart Home •Smart Life •Wearables Discounts •Wearables Health Services •Disease Management
D: Privacy & Security Requirements		
1	Needs to protect and preserve security and privacy on highly sensitive data (for mitigating IT-Security risks).	•Fraud Detection •Claims Automation •Industrial Insurance •Smart Home •Smart Life •Wearables Discounts •Wearables Health Services •Disease Management
2	Needs to ensure compliance with central data privacy laws (e.g. BDSG, GDPR when PII is involved).	•Churn Management •Targeting •Claims Automation •Fraud Detection •Smart Life •Wearables Discounts •Wearables Health Services •Disease Management •(External Data for pricing) •(Monitoring)

ID	Generic Requirement	Use Cases needing the Requirement
3	Needs to support multi-level access control and authentication (I&AM-issues) on protected data.	•Fraud Detection •Claims Automation •Industrial Insurance •Smart Home •Smart Life •Wearables Discounts •Wearables Health Services •Disease Management •Churn Management •Targeting
E: Lifecycle Requirements		
1	Needs to support data quality curation including pre-processing, data cleansing, data clustering, classification, reduction, format transformation.	•Churn Management •Targeting •Fraud Detection •Claims Automation •Telematics •Smart Home •Smart Life •Wearables Discounts •Wearables Health Services •Disease Management •Monitoring •(Industrial Insurance) •(External Data for pricing)
2	Needs to support a data lifecycle, a long-term preservation policy and prevention of data loss or corruption.	•Churn Management •Targeting •Claims Automation •Fraud Detection •Wearables Discounts •Wearables Health Services •Disease Management
3	Needs to support data validation (e.g. for quality control or audits).	all

ID	Generic Requirement	Use Cases needing the Requirement
4	Needs to support human annotation for data validation and result correction/validation.	•Fraud Detection •Claims Automation •(Churn Management) •(Targeting)
5	Needs to support standardizing, aggregating, and normalizing data from various sources. This includes providing a central landing zone for data integration from different sources, i.e. a data lake.	•Churn Management •Targeting •Fraud Detection •Claims Automation •External data for pricing •Industrial Insurance •(Wearables Discounts) •Wearables Health Services •Disease Management •Monitoring
F: Business & Organizational Requirements		
1	Need for a cooperation partner for providing devices for data collection.	•Smart Home •Telematics •Smart Life •Wearables Discounts •Wearables Health Services •(Disease Management)
2	Need for a cooperation partner for providing external data.	•Targeting •External data for pricing •(Disease Management)
3	Need for a cooperation partner for implementing the Use Case or parts of it.	•Targeting •Telematics •Industrial Insurance •Smart Home •Smart Life •Wearables Discounts •Wearables Health Services •Disease Management
4	Need for additional knowhow in data engineering for data collection, integration, storage, etc.	•Claims Automation •Fraud Detection •Smart Home •Smart Life •Wearables Discounts •Wearables Health Services •Disease Management

ID	Generic Requirement	Use Cases needing the Requirement
5	Need for additional knowhow in data science for developing new models/algorithms.	•Fraud Detection •Claims Automation •Industrial Insurance •Disease Management
G: Infrastructure and Capability Requirements		
1	Need for a Resource Management component for assigning processing capacity to the respective Use Case.	all
2	Needs to support legacy software packages, storage (e.g. Oracle Relational DB or Business Rules Engine) and platforms. However their usage within the Big Data platform should be limited.	•Churn Management •Targeting •Fraud Detection •Claims Automation •(Smart Life) •(Wearables Discounts) •Wearables Health Services •Disease Management •Monitoring
3	Needs to support advanced software packages (e.g. Scala), storage (e.g. NoSQL) and platforms (e.g. Spark).	•Churn Management •Targeting •Fraud Detection •Claims Automation •Telematics •Smart Home •Smart Life •Industrial Insurance •Wearables Discounts •Wearables Health Services •Disease Management •Monitoring •External Data for pricing

Table 4.14: Generic Requirements for Big Data Use Cases in the Insurance Sector
Source: Own considerations and [6].

Finally a few remarks on some of the requirements have to be made. Generally speaking, the requirements are needed for operationalizing Use Cases, i.e. most but not all of them have a direct impact for the new Big Data Reference Architecture. This is particularly the case when looking at the Organizational & Business requirements (F1-F5). Another special requirement is G2. Here it has to be made sure that even though the Big Data Reference Architecture supports some legacy technologies, their usage should be limited in order to keep legacy technologies within existing legacy systems and not rebuild old functionalities again. These legacy systems can be integrated with the Big Data platform through APIs or connectors. Finally the requirements concerning the Data Consumer (C1-C5) have not entirely to be covered by the Big Data Reference Architecture – instead they can be implemented within the application

using the Big Data platform for separation of concerns reasons. This decision depends on the respective Use Case: for example in claims automation the Big Data application could simply provide a result containing the decision on settling the claim and the amount of money to be paid out. The further processing of this result (e.g. contacting the customer, paying out the money) is then covered by the claims management system. The Solution Architecture for the claims automation case study shown in section 5.3 provides more details on this.

4.3 Comparison of Big Data Reference Architectures

In this section the three Big Data Reference Architectures from NTT Data, GCP and Microsoft Azure that were described in section 2.3.2, are compared by assessing them for whether they are able to fulfill the generic requirements derived in section 4.2. Although these architectures are similarly built with all having phases for ingestion, storage, processing and visualization of data, the difference lies in the fact that NTT Data's architecture is product independent, whilst GCP and Microsoft Azure apparently are not.
Each generic requirement – with the exception of the business & organizational requirements (F1- F5) – from section 4.2 is scrutinized for whether it is covered by the respective architecture in table 4.16. Table 4.15 explains the notation for the classification of the architectures.

Classification Sign	Explanation
✔	This means that the Reference Architecture is able to entirely fulfill the generic requirement.
O	This means that the Reference Architecture is able to partially fulfill the generic requirement.
✖	This means that the Reference Architecture is not able to fulfill the generic requirement.

Table 4.15: Requirement Assessment Notation
Source: Own depiction

Requirement	NTT Data	GCP	Microsoft Azure	Notes
A1	✔	✔	O	GCP excels with Cloud Pub/Sub, which is even able to outperform Apache Kafka by processing up to a million of messages per second. Microsoft Azure offers a number of good own solutions when it comes to sensor data and IoT, however for loading e.g. claims data, Apache Kafka is a better solution here.
A2	✔	✔	✔	
A3	✔	✔	✔	All Reference Architectures provide technologies for storing and processing unstruc-

Requirement	NTT Data	GCP	Microsoft Azure	Notes
				tured data.
A4	✔	✔	✔	All Reference Architectures provide technologies for storing and processing structured data.
A5	○	✔	✔	In NTT Data's case it is not possible to properly analyze this requirement hence it depends on the respective technologies used for ingesting data.
B1	○	✔	✔	Whilst NTT Data has a component for ML, it is not clear whether it also does support ML frameworks and APIs with it. GCP excels with TensorFlow here and Microsoft Azure supports Spark MLlib. Azure ML Studio makes it possible to develop own models by using Python or R.
B2	✔	✔	✔	GCP offers Cloud Dataproc for running existing Spark or Hadoop clusters. With Apache Beam GCP provides an own stream-processing framework (runs on Cloud Dataflow). Microsoft Azure supports Spark Streaming and offers Azure ML Studio.
B3	✔	✔	✔	Since Apache Beam supports also batch processing, GCP fulfills this requirement as well. With HDInsight Microsoft Azure offers a managed services for batch processing and supports a number of open source tools.
B4	✔	○	○	Neither GCP nor Microsoft Azure do offer an entire component for only monitoring the ML model execution. However such functionality is available within the respective managed services.
C1	✔	✔	✔	
C2	✔	✔	✔	
C3	✔	✔	✔	GCP's and Microsoft Azure's visualization tools are web-based.
C4	✔	✖	✖	NTT Data definitely can do this through its system integration component. For GCP and Microsoft Azure this is rather out of the scope.
C5	○	✖	✖	NTT Data could perhaps do this through its

Requirement	NTT Data	GCP	Microsoft Azure	Notes
				system integration component. For GCP and Microsoft Azure this is rather out of the scope.
D1	✔	✔	✔	
D2	○	○	○	In NTT Data's case compliance with GDPR cannot be entirely ensured hence the Reference Architecture does not contain a data stewardship component. Although GCP and Microsoft Azure are GDPR compliant, in some cases it can be not allowed to store sensitive data (e.g. health data) in the cloud for legal reasons. Therefore using GCP or Microsoft Azure is not always possible and depends on the respective Use Case.
D3	✖	✔	✔	NTT Data offers no component for I&AM.
E1	✔	○	○	Neither GCP nor Microsoft Azure provide single components for transformation purposes only. Respective Spark or Beam jobs can be written on owns own and run in a managed service.
E2	✔	○	○	All three Reference Architectures ensure recovery and loss prevention. But neither GCP nor Microsoft Azure provide single components for data lifecycle purposes only.
E3	✔	✔	✔	
E4	✖	✖	✖	The presentation layers of all three Reference Architectures have no components for human result annotation.
E5	✖	✖	○	NTT Data and GCP offer a wide range of storage technologies but they do not have a central storage zone like a data lake. Microsoft Azure offers Azure Data Lake Storage that can be used as a central HDFS based data storage. However, it does not have a component that sets up a physical data lake that is made up of various storage technologies and ensures the required processes like access control or data audits.
G1	✖	✔	✔	NTT Data offers a task scheduler during the ingestion phase, however it has no central

Requirement	NTT Data	GCP	Microsoft Azure	Notes
				resource management component. GCP provides a tool called orchestration for resource and workflow management; Microsoft Azure has Data Factory that takes care of these issues.
G2	○	✖	✖	NTT Data supports a number of legacy software and tools such as relational databases or Rules Engine processing. For GCP and Microsoft Azure this is out of scope.
G3	○	○	✔	Microsoft Azure supports a large number of modern software & technologies that are open source available such as Apache Spark or a Hortonworks based HDFS. GCP rather focuses on offering own modern technologies such as Big Table, Cloud Pub/Sub or Big Query. NTT offers components for ML and complex event processing, however it cannot be judged to what extent.

Table 4.16: Reference Architecture comparison based on Requirement Assessment
Source: Own depiction and [92], [95].

Although it does not make sense to simply count which Reference Architecture fulfills how many requirements, hence they are all differently important, a common comparison can still be made. NTT Data covers almost all phases of a data pipeline and offers a wide range of capabilities for ingestion, presentation and application integration. Yet it lacks tools for ML such as support of ML APIs and frameworks as well as the concept of a central data lake, which is particularly important here – especially since NTT Data's architecture aims to provide data processing and storage concepts and not an overview of products. GCP boosts a large number of own technologies and excels in ingestion, storage and analytics phases. However, it too lacks the concept of a central data lake. Microsoft Azure offers several own tools for ingestion and storage – in the latter phase it also supports many open source technologies such as HBase or a Hortonworks implementation of HDFS. The integration with many open source technologies can also be seen in the analytics phase, where it supports Apache Hive, Apache Impala, Spark MLlib, R and Python for various analytics purposes. However, in the analytics layer Microsoft Azure offers only few own technologies like Azure ML Studio – this is where GCP outperforms Microsoft Azure by even providing an own processing framework with Apache Beam and more managed services. Whether this is a disadvantage or not, depends on the insurer's circumstances and his requirements for a Big Data provider. If an insurance company has already many own Big Data applications based on open source tools, GCP could be difficult to integrate with them. If the insurer has only a few own Big Data applications yet, using GCP could be a better alternative than Microsoft Azure.

5. Big Data Reference Architecture for the Insurance Sector

5.1 Mapping Requirements to Architecture Components

Before starting out with the design of the Big Data Reference Architecture for the insurance sector, some explanations have to be given on the design methodology itself. As it has already been pointed out, the Big Data Reference Architecture is based, inter alia, on an analysis of Big Data requirements. Generic requirements that are listed in section 4.2 are now mapped to architectural components in the Big Data Reference Architecture following the approach chosen by the NIST workgroup [7]. The architectural components can be found in the top-level Big Data Reference Architecture presented in section 5.2. Another reference used for developing the new Big Data Reference Architecture is a chapter from the "Practical Guidebook to Big Data", where a cross-industry Big Data Reference Architecture is presented that is based on a list of generic requirements [57]. These requirements are mostly derived from Big Data characteristics such as the V's (Volume, Velocity, Variety and Veracity) and from analytical computations. In the explanations for the architecture diagram, one sees that the requirements, which can largely be regarded as non-functional requirements, are mapped to single components or entire groups of components.
Applied to the generic requirements from section 4.2 the mapping looks as shown in table 5.1. The only exception are the requirements from the Business & Organizational category, i.e. F1 to F5, since it makes no sense to map them to architectural components.

Requirement	Architectural Component
A1	Processing Engines, Data Loading
A2	Processing Engines, Data Loading
A3	Processing Engines, Data Loading, Storage Technologies
A4	Processing Engines, Data Loading, Storage Technologies
A5	Data Sources, Data Loading
B1	Machine Learning
B2	Processing Engines
B3	Processing Engines
B4	Machine Learning
C1	Result Visualization
C2	Result Visualization
C3	Result Visualization
C4	Event Triggering
C5	Event Triggering
D1	Data Privacy & Security
D2	Data Privacy & Security (Data Stewardship, I&AM)
D3	Data Privacy & Security (I&AM)
E1	Data Transformations, Data Quality
E2	Recovery & Loss Prevention, Data Lifecycle Management

Requirement	Architectural Component
E3	Data Quality, Data Transformations
E4	Event Triggering
E5	Data Lake, Storage Technologies
G1	Workflow & Resource Management
G2	Storage Technologies, Analytical BI, Data Loading, Result Visualization
G3	Storage Technologies, Machine Learning, Data Loading, Result Visualization, Processing Engines, Sandboxing

Table 5.1: Mapping Requirements to Architectural Components
Source: Own depiction

5.2 Top-Level Big Data Reference Architecture for the Insurance Sector

Based on the analysis of requirements for operationalizing Big Data Use Cases and an analysis of existing Big Data architectures (see section 2.3.3) a number of components for setting up a Big Data Reference Architecture for the insurance sector has been identified. Particularly the structure of the new Reference Architecture has been derived from the comparison of existing Big Data architectures. The new Reference Architecture can be seen as a top-level modular system from where any components can be picked for designing a Solution Architecture for a Big Data Use Case in the insurance sector. Since the Reference Architecture is based on a number of requirements for operationalizing insurance Use Cases it is made sure that the Reference Architecture provides components needed for implementing each of them.
The Reference Architecture itself consist of two levels with the first level depicting all architectural layers (shown in figure 5.1) and the second level showing each of these layers in more detail. Level two also gives a deeper explanation the architectural components each layer is made up of. The following two sections present the two levels of the new Big Data Reference Architecture.

5.2.1 Big Data Reference Architecture - Level 1

Level one of the Big Data Reference Architecture shows a top-level view on all the architectural layers in the Reference Architectures. It covers data sources, core components and cross-functional components. As it has been pointed out in the comparison of existing Big Data Reference Architectures, a data pipeline is always built following the same structure with four main phases. Therefore the core components consist of ingestion, storage, analytics & processing, and presentation & operationalization. Ingestion and storage deal with loading, transforming and storing data as it comes from the various data sources. In analytics & processing the data that is now available after it has passed the two previous layers is analyzed by executing analytical or predictive algorithms on processing engines. The goal of this layer is to derive insights from the available data. These insights can then be visualized or sent for post-processing to the presentation & operationalization layer where a data consumer uses them.
In order to group the architectural components and improve the readability of the architectural diagram, several categories of the architectural components are introduced. The four phases within the core components and the cross-functional components can be regarded as function-

al Big Data processes. These processes, particularly within the core components are implemented using Big Data applications, systems and components. Data arrives to a Big Data application that is built using the Big Data Reference Architecture from a number of sources with some examples being pictured in figure 5.1 as well. Both the core components and the cross-functional processes are executed on infrastructure components, which form the last category.

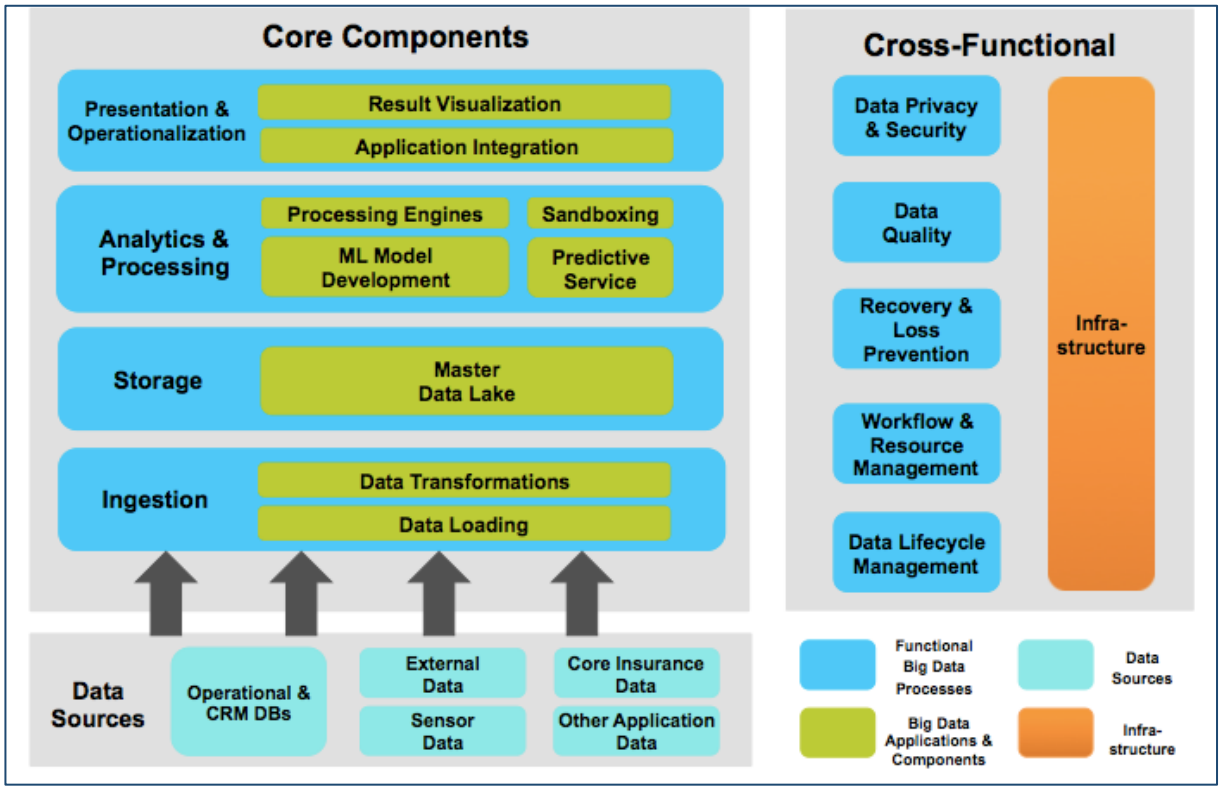


Figure 5.1: Big Data Reference Architecture – Level One
Source: Own depiction

5.2.2 Big Data Reference Architecture – Level 2

In this section the components within the single layers of the level one Big Data Reference Architecture are described.

Data Sources: The data sources provide the data required to gain insights in order to be able to create an added value from Big Data Use Cases. Hence the sources vary with each Use Case, the Big Data Reference Architecture shows only a few examples. They include external (e.g. social media) and sensor data or data from existing operational or CRM systems. A very important data source in insurance companies is core insurance: these are transactional systems that cover the most important functional processes like for instance underwriting, contract and claims management.

Ingestion: In the ingestion phase data is loaded from the data sources into a Big Data application. There are several ways how it is possible to bring data into the application: either

through batch-loading or stream-loading. In batch-loading data gets into a queue and at a certain configurable time the content of the queue gets loaded to the Big Data application. This approach can be chosen when ingesting data from operational systems' databases into the data lake. A possible tool that could be used for this is Apache Sqoop – an application that transfers data from relational databases into HDFS. In stream-loading data gets ingested into the Big Data application as soon as it is generated – therefore it is used in most IoT Use Cases and other velocity applications relying on getting insights from data as fast as possible. Streaming itself is a broadly defined term, here it refers only to ingesting the data without manipulating, i.e. processing it. There are many different applications existing for stream-loading data – the most known is Apache Kafka. It is a streaming platform that works as a publish/subscribe queue where data from data sources (called producers) is written to a distributed log. The log is split into topics, which in turn are split into partitions. Applications using the data (called consumers) can subscribe to different topics to get the data they need from the streaming platform. Finally data can also be loaded into the Big Data application through conventional ETL tools that already are in place in most insurance companies for transferring data into enterprise data warehouses.
After data enters the Big Data application transformations can be applied to it. However the placing of the data transformation components strongly depends on the Use Case and governance principles. For example data can be loaded in a raw state into the data lake and transformations can be applied to it only when it is extracted for processing in the analytics layer. Here all Use Case specific transformations can be covered. However in order to prevent the data lake from becoming a "data swamp" it would make sense to apply cross Use Case cleansing processes and validations to all data before it is put to the data lake in the storage zone. Data transformations can range from changing the format of the data over cleansing and validating the data to enriching and aggregating the data.

Storage: The central component in the storage layer is the master data lake. Its core is the physical data lake where all data required for analysis purposes in the insurance company is stored. The physical data lake is implemented through different storage technologies that depend on the data that has to be stored and the respective Use Case. Examples include HDFS, NoSQL and relational databases. The physical data lake is split into several zones for assuring that not everyone within the insurance company has access to all data. This also guarantees compliance with data protection laws, as e.g. data from health insurance is not allowed to be stored and combined together with data from life insurance contracts. One zone that is explicitly named in the Reference Architecture diagram is the data science data lake, which is used for storing data that is required for sandboxing. The reason for this zone is that data scientists exploring and developing new models are not always allowed to have direct access to production data that comes to the data lake from operational systems and other sources. Thus this zone contains data that is partially anonymized in order to meet compliance requirements but also make it possible for data scientists to develop proper models. In order to ensure that no one can get access to data within the respective storage zone without permission, an access control component is in place that is connected to the Identity & Access Management process from the cross-functional layer. Furthermore the data lake is strongly integrated with data

lifecycle management, data quality and data privacy & security processes that are implemented in the cross-functional layer (please refer to the explanations there for more details). Metadata can be stored in an own database, however as it has to be available for all storage zones in the data lake, it makes sense to decouple it from the physical data lake and show it as single component.
Data can be accessed from the analytics & processing layer of any Use Case through querying connectors. Depending on the state of the data, transformations can be applied after extracting data from the data lake (e.g. in case of using the schema-on-read principle). Additional data aggregations, validations and enrichments can be performed as well – therefore there is also a transformations connector in the data lake. The following figure 5.2 shows the components within the storage layer.

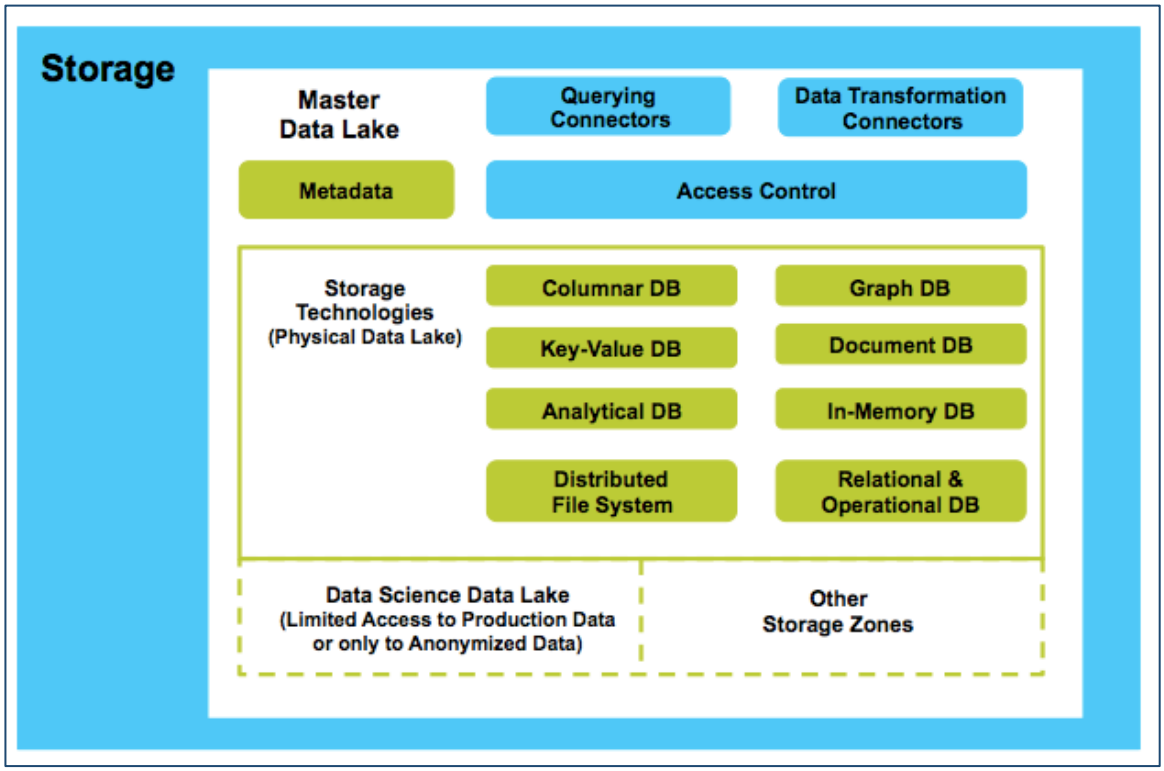


Figure 5.2: Big Data Reference Architecture – Level Two: Storage Layer
Source: Own depiction

Analytics & Processing: In this layer data is processed so that by using analytical and predictive algorithms insights can be derived from the data and later used by data consumers. New ML models can be developed by using existing ML frameworks such as TensorFlow or H2O that afterwards are trained with large datasets and finally executed as a predictive service. In model development, common languages for ML like Python, R or Scala are supported. For solving common ML problems like text analytics or image recognition existing ML APIs like Google Cloud Vision or Natural Language API can be used as a basis for transfer learning. Within the predictive service where the trained models lie, there is a component for monitoring and evaluating the precision and accuracy of the model during production performance. If needed, model parameters can be adjusted or the model can be retrained for optimizing its results. Additionally unsupervised ML algorithms can be executed for clustering data and

uncover new, so far unknown patterns. Data scientists can also explore and test new models within the sandboxing component that supports such exploratory analytics. Conventional analytics such as OLAP or batch analytics are available as well so that a Big Data application is able to support a BI-style report generation if required by a Use Case. Finally for keeping all business rules that will still be needed in insurance companies to keep business processes up and running a Rules Engine (e.g. Drools) exists. Existing rules can either be migrated to a new Rules Engine within the Big Data application or accessed from the application via an API in the existing operational or core insurance systems.
All analytical and predictive algorithms, no matter whether in a training or production phase require processing engines to run on. Depending on the processing speed the respective Use Case needs, different processing engines or frameworks exist. They can be grouped in three categories: batch-processing, micro-batching and stream-processing (more details on the different processing frameworks can be found in section 2.1.3). The following figure 5.3 shows the components within the analytics & processing layer.

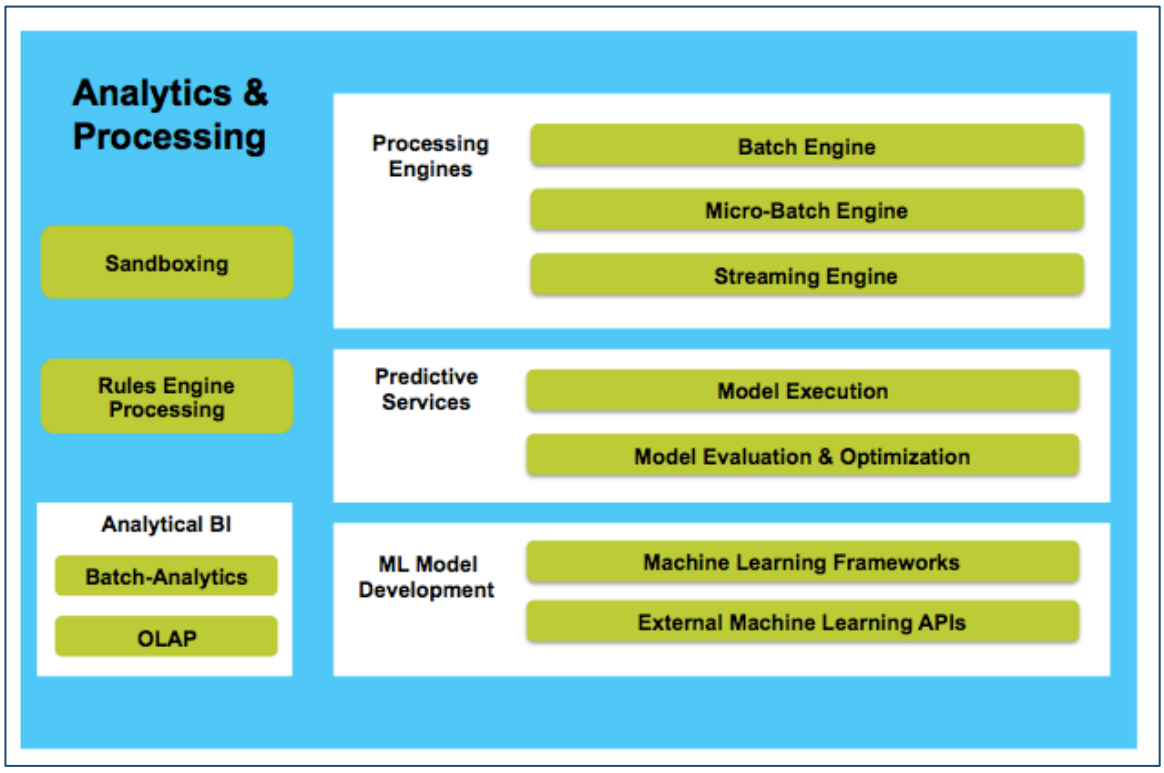


Figure 5.3: Big Data Reference Architecture – Level Two: Analytics & Processing Layer
Source: Own depiction

Presentation & Operationalization: After deriving insights from data by performing calculations in the analytics & processing layer, their results have to be visualized and passed to other applications for post-processing. Data visualization consists of common visualization tools like e.g. dashboards, diagrams and report creation where reports are automatically generated in natural language based on the results retrieved from the analytics & processing layer. Furthermore tools for visual data exploration, i.e. for sandboxing should be available – examples are GCP's Cloud Data Lab or Microsoft Azure Machine Learning Studio.

However for most Use Case in insurance, data analysis results have not only to be visualized but also to be processed further for being able to get an added value from them. This post-processing is enabled through application integration where a Big Data application can trigger workflows in other systems or pass its results there through APIs. Another very important component is responsible for making the results available for manual checks and validations before passing them over to the following system. This is particularly important for ensuring result quality and handling complex cases, e.g. in fraud detection or claims settlement. Apparently it has to be closely integrated with visualization tools so that clerks can see the data analysis results and get additional information on the respective cases.

Cross-Functional: This layer covers all functional processes that support the core components of the Big Data Reference Architecture, spanning the entire data pipeline and also including infrastructure. Data privacy and security issues are dealt with here, ensuring both compliance with data protection laws and IT-security for the system. A component here is Identity & Access Management where roles and rights for accessing data (particularly different storage zones within the master data lake) are defined. In Data Stewardship it is outlined why a role has access to certain data sets and how it uses this data for making sure that the Big Data platform follows the principles of purpose for data usage and thus complies with GDPR [94]. Anonymization apparently provides capabilities for anonymizing data, which is particularly required for Use Cases where PII is involved, in order to comply with GDPR and other data protection laws.
Another highly important process is to ensure data quality within the Big Data platform. This is achieved through applying data validations and quality checks to the data, particularly to what is stored in the master data lake. The results produced by the analytical and predictive models have to be checked for quality assurance as well – this is partially covered by the model evaluation and optimization component in the analytics and processing layer. Data audits ensures data quality and analyzes whether the audited data is fit for serving its intended purposes.
Additional processes make sure that data is not lost in case of outages through deploying replication mechanisms within the Big Data platform and also provide support for recovering core components after an outage for keeping operations up and running. Data lifecycle management takes care of issues such as for how long data can be stored within the platform and where to store which data sets depending on factors like for example access frequency, data type or data structure. Workflow and resource management deals with orchestrating workflows within the Big Data platform and managing and distributing available resources between processing tasks within these workflows. The latter is particularly important for a Big Data platform since resource consumption is often high and it has to be made sure that enough of them is available so that a pipeline does not break down in case a single task starts consuming all resources [9].
Finally infrastructure provides the capabilities required for running the Big Data platform in a distributed environment. This includes the hardware, i.e. the server nodes where Big Data applications can be deployed and the networks connecting them. Here it is especially important to take non-functional requirements concerning availability, latency and partition tol-

erance into account. The entire server landscape has to be closely monitored through a system management component to ensure an even load distribution between the single server nodes and a reliable and stable operations environment. The following figure 5.4 shows the components in the cross-functional layer.

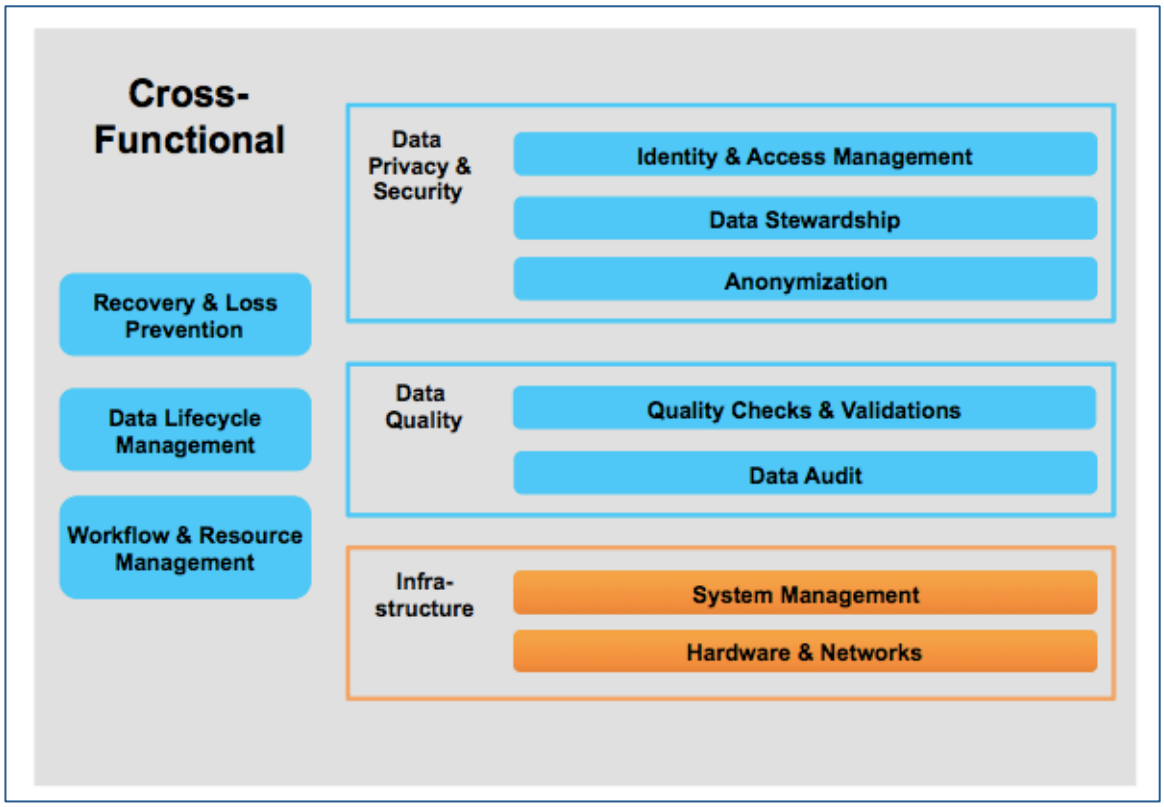


Figure 5.4: Big Data Reference Architecture – Level Two: Cross-Functional Layer
Source: Own depiction

5.3 Case Study: Big Data Solution Architecture for Selected Use Cases

In this section the new Big Data Reference Architecture is tested for its suitability by designing a Solution Architecture for two selected insurance Use Cases that were presented in chapter three, following a notation that is defined in section 5.3.1. The Solution Architectures mainly visualize the data- and workflows that are required to implement a Use Case. The two Use Cases selected here, fraud detection and claims automation, are particularly interesting because of the results of the Use Case evaluation (please refer to section 3.3 for more details) and because of an insurtech currently excelling in that area. This insurtech is US-based Lemonade, which offers customers P&C insurance for their homes. Lemonade promises customers to get insured within 90 seconds and be paid within at maximum three minutes when filing a claim. As a matter of fact, Lemonade even set up a world record by settling 25% of all claims within less than three seconds. Lemonade achieved this through the usage of AI for fraud detection and claims settlement. In order to provide established insurers with similar capabilities a Solution Architecture for the two Use Cases is designed [98].

5.3.1 Notation Definition

When visualizing an architecture in a diagram there is always the question of which notation to use for this. Many different notations like e.g. UML or BPMN as well as tools such as Sparx Enteprise Architect do exist with each of them having advantages and disadvantages. For depicting a Solution Architecture it is required to be able to show a dataflow and workflow steps where data is processed, what is basically a mixture between UML sequence, component and activity diagrams. As no standard notation exists for this purpose, it was decided to use a self-defined "boxes-and-lines" notation that is shown in figure 5.5.

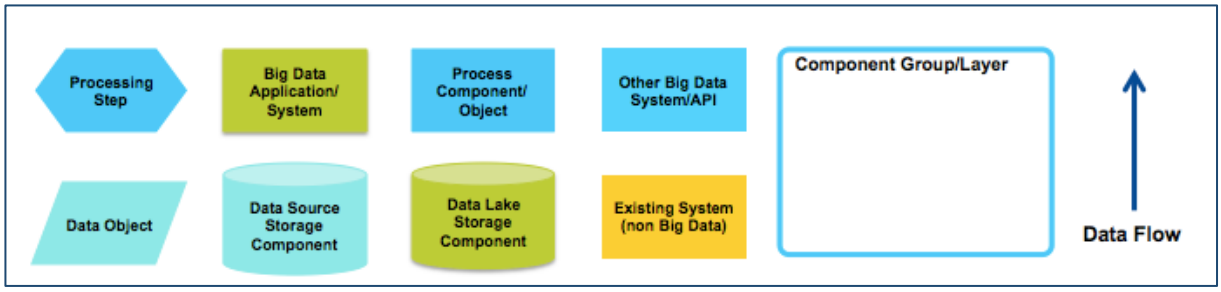


Figure 5.5: Big Data Solution Architecture notation
Source: Own depiction

5.3.2 Big Data Platform Approach

Before designing the Solution Architectures for the selected Use Cases, some architectural decisions have to taken in order to provide a common approach for implementing these Use Cases. The basic idea is to use a platform approach where several components can be used by all Use Cases so that operation costs go down and redundancies are reduced. These common components particularly cover the ingestion, storage and parts of the analytics & processing layers from the Big Data Reference Architecture. Figure 5.6 shows the Big Data platform and its interactions with the Big Data Use Cases.
Any data coming into the Big Data platform used by any Big Data Use Case has to enter the platform via a streaming platform that will be implemented through Apache Kafka, a common streaming application used by many companies in different industries. Although it is designed as a solution for streaming data, Kafka can also deal with other ways of data loading named in the Big Data Reference Architecture, e.g. batch-loading. In case of existing systems, this simply requires adding a new Kafka connector for the respective system, which functions as a data source so that it becomes a publishing producer (Kafka term for a data source). In case of new systems that are sources for data that has to be processed at near real-time speed it makes sense to develop an own Java connector to Kafka and ingest the streaming data through it (the Solution Architecture for claims settlement gives a better understanding of the two possible approaches). After the data has entered the streaming platform all of it has to pass a number of common data transformations that are executed using Apache Spark (a Spark job fetches the data from Kafka by subscribing to respective Kafka topics before applying transformations to it). These data transformations include data format transformations, data cleansing and common validations to ensure that data arriving in the master data lake has good quality.

The master data lake is where all data, which is used for any analytical purposes by any Big Data Use Case is stored, serving as a “Single source of truth” for analytics in an insurance company. The concept of the master data lake is described in section 5.2.2 with figure 5.6 showing only a few examples for storage technologies and data that can be stored in a master data lake used by an insurance company. The storage technologies within the master data lake support both structured and unstructured data as well as SQL-querying through tools like Apache Hive or Apache Impala. Hive enables querying for generating BI-style reports whilst Impala makes it possible to perform real-time analytics. Therefore the master data lake can not only be the basis for ML applications, but also the foundation for BI applications, thus being able to replace conventional data warehouses. Other components provided by the Big Data platform are ML frameworks or APIs and processing engines. The first are used for developing new, insurance specific ML models for the Big Data Use Cases whilst the latter are clusters of engines where the models can be trained and executed on. These processing engines can be for instance a Spark cluster used for micro-batching or a Hadoop cluster used for batch-processing large data loads. It is necessary to point out that neither all ML frameworks nor processing engines pictured in the diagram here have to be used at once by an insurance company. For instance if a company uses Spark there is no need for supporting Flink additionally.
Finally the Big Data platform also provides cross-functional components like data privacy and security or resource management as well as infrastructure components. However as these two are needed for the Use Case specific parts of a Big Data application as well, they are not limited to the Big Data platform.
Any Big Data Use Case can use the Big Data platform, but they still have to implement their own predictive or analytical models since they contain Use-Case specific logic within the algorithms or models. The visualization and post-processing of the models results has to be implemented Use-Case specific as well, since especially the integration with the applications for the post-processing depends on what is required in the respective Use Case.

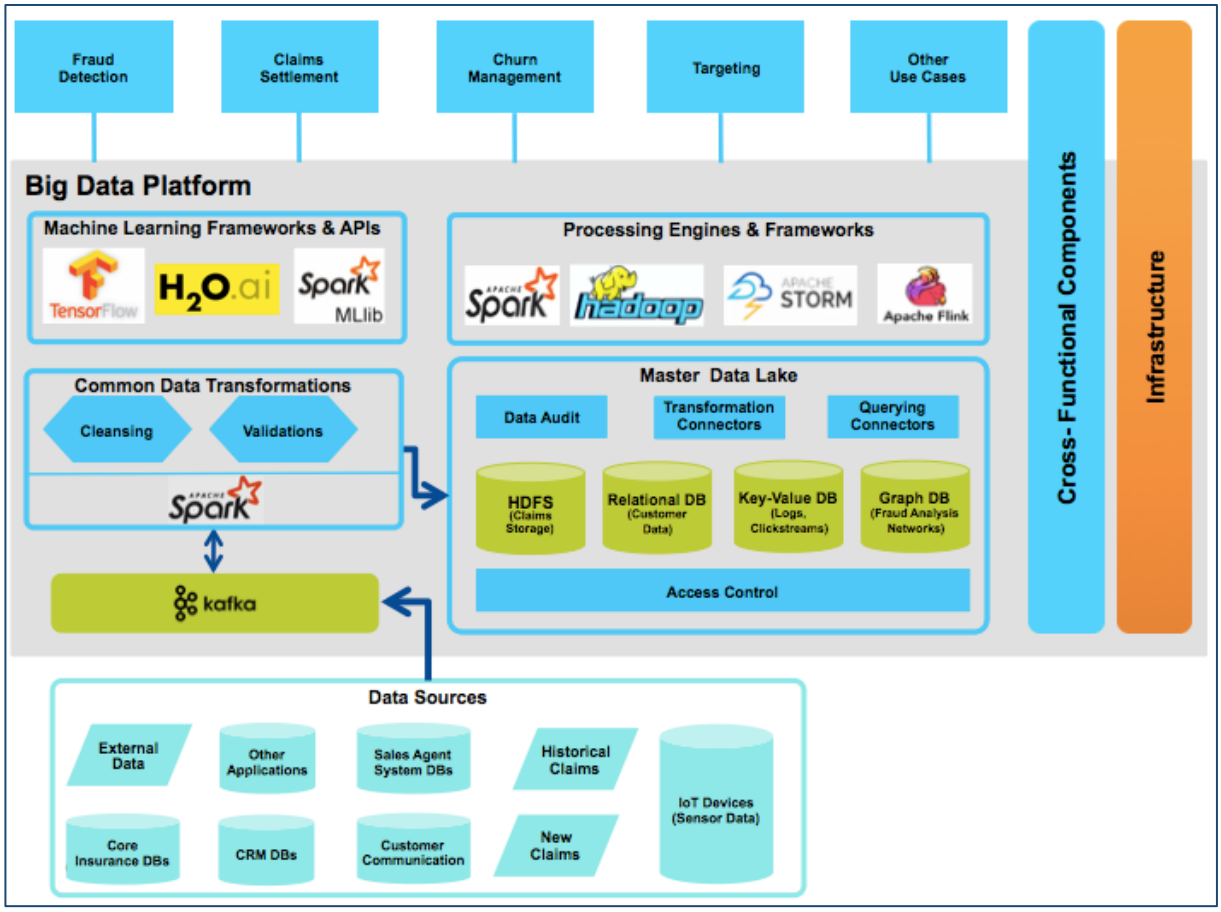


Figure 5.6: Big Data platform architecture
Source: Own depiction

5.3.3 Case Study – Claims Settlement

This case study deals with the claims automation (shown in figure 5.7) and fraud detection (shown in figure 5.8) Use Cases and provides the respective Solution Architecture for each of them. The main aim of these two Use Cases is to settle claims automatically and as fast as possible whilst ensuring a claim is not fraudulent. As well as all other Use Cases these two do use the common Big Data platform with the capabilities it provides for the ingestion, storage and processing phases. Although the architectural diagrams in figure 5.7 depict the Use Case in production stage with trained predictive models it is worth taking first a look at what happens in the model development stage.

Here in the beginning data from core insurance and other IT-systems, historical claims data and external data is ingested into the data lake and there into the respective storage system, e.g. HDFS for claims or a relational database for customer data. When it comes to training the ML models in the Analytics layer, the required data is retrieved from the data lake, transformed and aggregated with other required data through a Spark job. As soon as training data is available in the needed quality, the model development component begins training the models by applying the respective ML algorithms and using frameworks such as H2O or TensorFlow. The model training is executed by using a Spark cluster as processing engine. For claims automation particularly supervised ML techniques are used and applied to historical

claims. As soon as the models are ready, they are deployed to the predictive service where they can be executed when the Use Case enters production stage.
In production stage the data pipeline looks as follows: in the ingestion phase, new claims arrive at the claims registration system via different customer interaction channels, e.g. from a mobile app or the company's website. Since in most cases a claim has the form of an image file coming from an invoice or expertise scan, claim-specific OCR and data structuring has to be applied already there for retrieving the content of the claim. This retrieved content is then directly sent to the Kafka Streaming platform via a specific API that has to be developed for connecting the claims registration system with Kafka. Updates in data from other sources, e.g. in core insurance data coming from relational databases, can be sent to the Kafka Streaming platform via Kafka source connectors that are already available for most standard storage technologies and products. By using a Kafka Streaming platform it is possible to ingest claims at real-time speed thus fulfilling one core requirement of the claims automation Use Case.
Once inside Kafka, claims are fetched by a common Spark job that also applies first data transformations to it, including data cleansing and validations. After this first common validations claims data is send first of all to the next round of claim-specific validations and data enrichment processes that contain Use-Case specific business logic and are also based on Spark jobs (hot path). Data enrichment happens via retrieving data from the Master Data Lake via a querying connector. Additionally the data pipeline junction sends claims data to the Master Data Lake where it is stored e.g. in HDFS or any other fitting database (cold path). Furthermore, the new claim is sent to the fraud detection application.
After the claims-specific transformations comes the predictive service where the decision on how to settle the claim is made. At the heart of the predictive service lies the execution of the previously trained supervised ML models that are executed on a Spark cluster, which is used as processing engine for micro-batching the claims. This ensures a very fast processing of the claims. As soon as it is put to action, the ML models analyze the content of a claim by setting up a relationship between what has been previously identified using OCR and the claims settlement logic. It also sends a request to the product modeling system for getting information on the customer's current products. Afterwards it uses a business rules engine (e.g. Drools) to support its own calculation in order to check whether the filed claim is covered by the current product of the customer and look up how much to pay out for the claim. Above all it calls the fraud detection system to check whether the newly filed claim is fraudulent or not.
The result of the ML models is sent for further processing, which varies depending on the concrete result. In case the claim gets settled directly the result is sent to the payment system for paying the customer his money and customer communication is triggered as well, for informing him about the result. If it is not settled directly, an insurance clerk can check the result first, adjust it if necessary or contact the customer for further discussion.
The predictive service has another component for model monitoring that collects the results of the ML model execution and sends them for evaluation to an analytics component, which can improve ML models by adjusting them in order to optimize model accuracy and precision. Finally a component for sandboxing allows data scientists to explore new ML algorithms and models so that better ones can be found in order to continuously improve the predictive service and the entire Use Case.

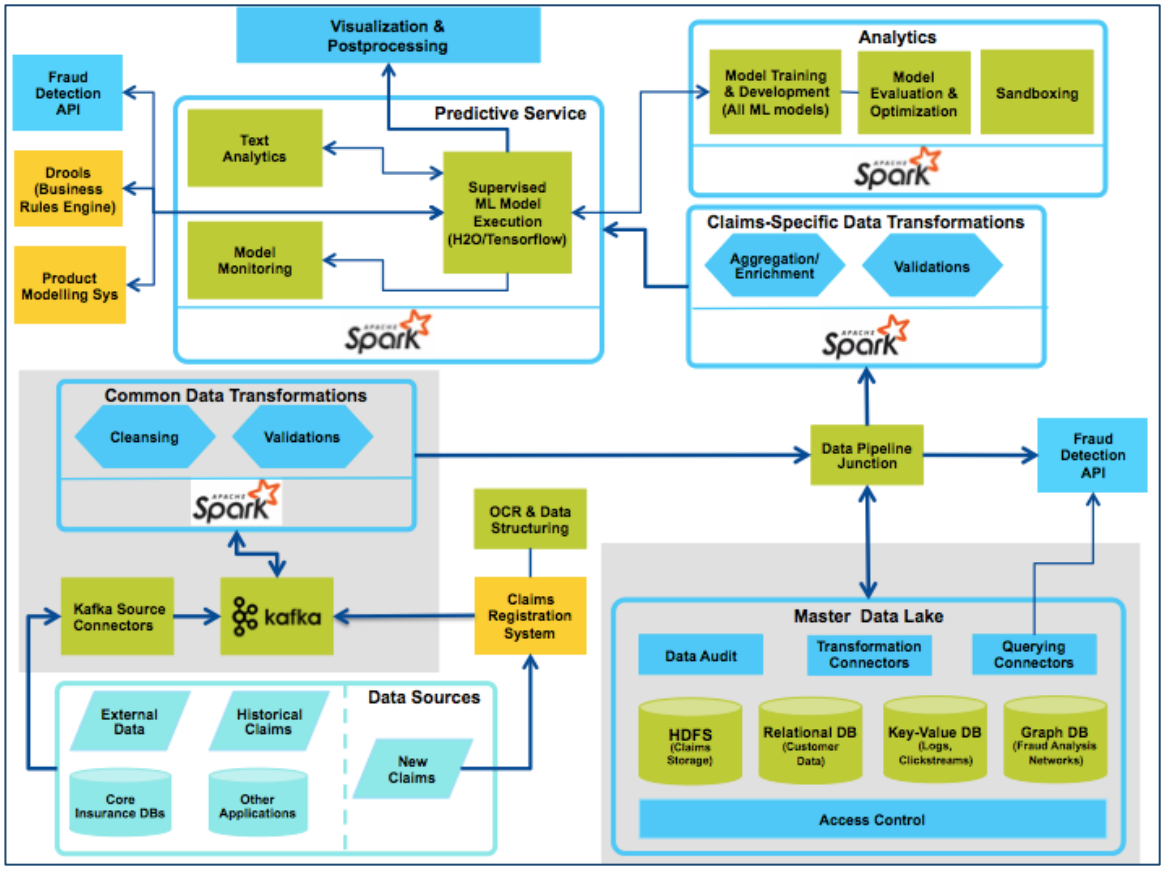


Figure 5.7: Solution Architecture for Claims Settlement
Source: Own depiction

As already said, claims automation requires checking newly filed claims for being fraudulent in a fraud detection application. The respective architecture is shown in figure 5.8. Just as in claims automation the ML models for determining whether a claim is fraudulent have to be trained first. The required training data comes from core insurance systems, historical claims and external data whose ingestion into the data lake is shown in figure 5.7 and has been described before. In order to train the ML models, data is retrieved from the data lake via a querying connector and then aggregated. As soon as the training data is available, the model training and development component trains the ML models by applying ML algorithms and frameworks. Just as in claims automation the main data source is data from historical claims, particularly for the supervised ML models. When the ML models are ready, they are deployed to the predictive service.

In production stage new claims follow the same path in the data pipeline built using the Big Data Platform as shown in figure 5.7 until they are sent by the data pipeline junction to the fraud detection API. Inside the fraud detection system the new claims first pass through fraud-specific data transformations that consist of validations and data enrichment. These processes are executed using Spark jobs. Afterwards the new claims enter the predictive service where at first supervised ML models are executed, which run on an Spark cluster. The supervised ML models detect fraudulent patterns by applying text analytics in order to analyze the content of a claim, sentiment analysis and photo forensics for identifying anomalies. Additionally

network analysis is used to scrutinize the relationship between the one who filed a previous claim, who fixed the damage, who settled a previous claim, possible surveyors and other actors involved in the settlement process. Storing the network data that shows the relationship between the actors requires a graph database, which is shown explicitly in the Master Data Lake in figure 5.7. When the supervised ML models have determined whether the claim is fraudulent, the result is passed for post-processing, especially to the claims automation system that needs this result for settling a claim. However, the predictive service is not limited to supervised ML models. Unsupervised ML models allow to cluster claims so that patterns can be detected that have so far been unknown to anti-fraud agents inside the insurance company or sector. This works as follows: The unsupervised ML models are executed on a Spark cluster and deliver a list of possible new patterns that is sent to an IT-system an anti-fraud agent works with. Here the agent can decide whether the newfound pattern makes sense, i.e. can be generalized and indeed applied for uncovering fraudulent claims in the future. If yes, he shows this new pattern to a data scientist who can analyze it further by using the sandboxing component and eventually improve the existing ML models by including this pattern into the models within the predictive service. Of course both supervised and unsupervised ML models are monitored when in production stage, evaluated and if necessary optimized using the respective component inside the analytics layer.
Certainly it will take time until the ML models both in claims automation and fraud detection are precise enough that no or almost no human intervention will be needed. Therefore in the beginning it is strongly recommended to have insurance clerks check the system's results before paying out money or contacting the customer. However in time through the integration of claims automation and fraud detection and a high accuracy of the ML models, claims can be settled automatically within a few minutes or even less by using a real-time processing architecture.

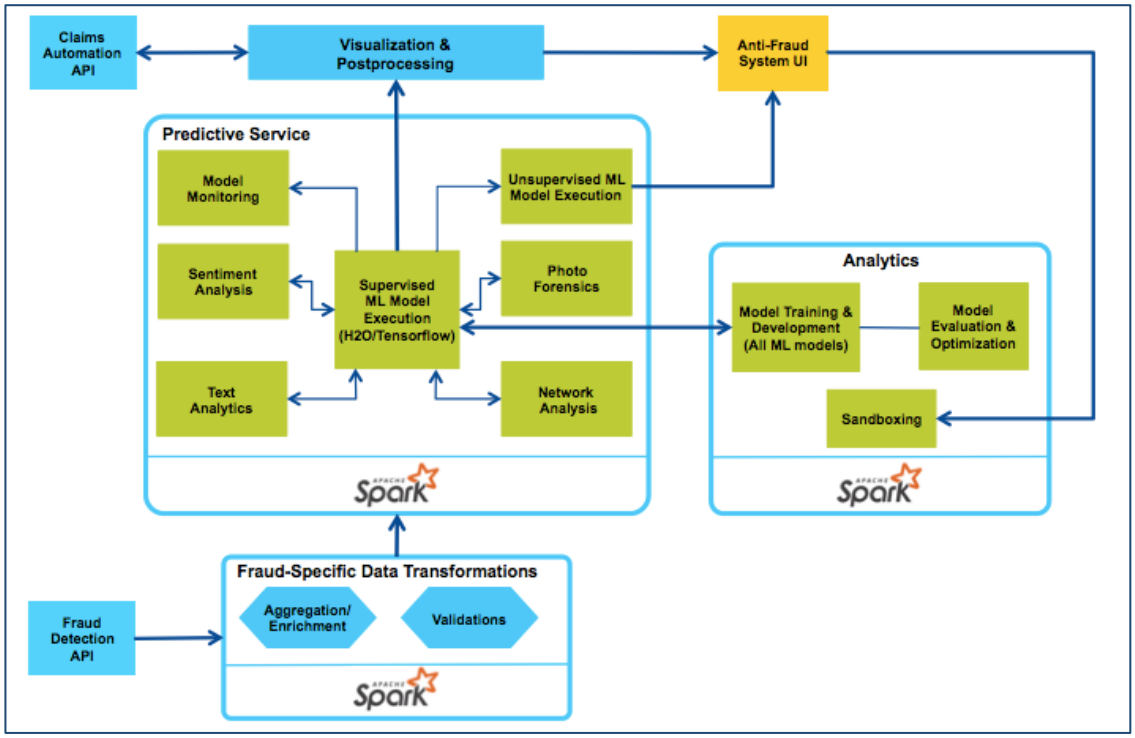


Figure 5.8: Solution Architecture for Fraud Detection *Source: Own depiction*

6. Evaluation

Following the Design Science Research Methodology, after developing artifacts comes their evaluation [10]. The Big Data Use Cases shown in chapter three were evaluated using interviews with senior managers, business owners and Big Data experts. The aim here was to find out, which Use Cases are most promising for the insurance sector regarding various factors such as added value, complexity/feasibility and possible risks. Given the high number of Use Cases, the evaluation offers additional help to decision makers in an insurance company. The results of this Use Case evaluation give an insurance company a possibility to decide which Use Cases it wants to focus on first and tackle arising risks respectively. However, the detailed results of these interviews are not publically available for non-disclosure reasons. The evaluation methodology and some overall results of the expert interviews are shown in section 3.3. It has to be pointed out that based on the results of the Use Case evaluation, the two Use Cases (claims automation and fraud detection) for testing the Big Data Reference Architecture were selected.
The other artifacts developed in this thesis are the requirements for operationalizing the Big Data Use Cases and based on them, the Big Data Reference Architecture together with the Big Data platform. The latter two were evaluated by designing a Solution Architecture for two Use Cases from chapter three in order to see, whether the components offered by the Big Data Reference Architecture and platform can satisfy the requirements from the two Use Cases (see section 5.3.3). Eventually, both the Big Data Reference Architecture and Solution Architecture were discussed in unstructured interviews with experts from leading Big Data and cloud providers and a leading insurance company [92, 93, 94, 95, 96, 97]. The evaluation of the architectures was conducted after designing their first versions. Depending on the feedback, new components were added or restructured. Unlike in the Use Case evaluation, the experts were not provided with a structured questionnaire, but asked generally what they thought about the Big Data Reference Architecture. The experts were shown both levels of the Big Data Reference Architecture and the Solution Architecture in detail and asked whether:

- The Big Data Reference Architecture is complete from their point of view.
- The Big Data Reference Architecture and the capabilities it offered were sufficient to cover the requirements the Big Data Use Cases in the insurance sector had.
- The Solution Architecture could be built using the capabilities offered by the Big Data Reference Architecture.
- The Solution Architecture covered all processing steps required by the claims automation or fraud detection Use Cases.
- The Big Data platform approach made sense and was suited to complement the Big Data Reference Architecture (this question was asked only when the platform approach was introduced after the first feedback).

In order to give a better overview of who the experts who evaluated the Big Data Reference Architecture were, table 6.1 lists their roles and work experience.

Role	Work Experience
Cloud Solution Architect	Less than 5 years
Former Chief Architect & Managing Director	More than 20 years
Cloud Solutions Architect	5 to 10 years
Senior Enterprise Architect	15 to 20 years
Program Lead Central Data Storage (Data Lake)	15 to 20 years
Senior Data Scientist	Less than 5 years

Table 6.1: Interview partners for the architecture evaluation

Source: Own depiction

The overall response to the Big Data Reference Architecture was positive with the experts pointing out that it covered all important capabilities required for a Big Data application. They agreed with the fact that the Reference Architecture is layered like other Big Data Reference Architectures (see section 2.3.2) with the core components containing nothing that would be insurance specific. Nevertheless, insurance specific aspects play an important role during the ingestion phase when integrating core insurance and other legacy systems with Big Data applications. They also are important when it comes to operationalizing the results calculated in a Big Data application because it has to be integrated with other applications in an insurance company, like for instance payment or core insurance systems, so that the results calculated in the Big Data application can be used for further processing.

Based on the experts' feedback, amendments to the Big Data Reference Architecture were made and presented in later expert interviews for another review. Thus, in the Big Data Reference Architecture the analytics & processing phase was split up into an area where ML models are developed and an area where they are executed (the predictive service) [92, 93]. In the cross-functional area data privacy and security were enhanced through a component for data stewardship, so that the Big Data Reference Architecture is compliant with GDPR. Data stewardship helps determining and explaining why a role has access to a certain data item or storage zone within the Master Data Lake [94, 95, 96]. Additionally, during the first expert interviews the idea for setting up a Big Data platform that supports the Big Data Reference Architecture and offers a common Big Data infrastructure as well as a tool-stack to all Use Cases was suggested [93, 94]. In interviews conducted later, this platform approach was judged to be highly helpful for setting up Big Data standards and a cross Use Case infrastructure, thus confirming the opinions of the other experts [95, 96]. Furthermore, the experts recommended to use Apache Kafka as a standard for ingesting data into the Big Data platform because of its scalability and good integration with other storage technologies through Kafka connectors [92, 93]. Above all, Kafka is able to cover all ingestion capabilities required from the Big Data Reference Architecture including both batch- and stream-loading data. Since Kafka is pull based, it is possible to easily extend or remove data sources required for a Use Case by simply adding new producers, without having to significantly change the inner set-up of the Kafka Streaming platform. Thus using Kafka helps fulfilling a central generic requirement.

The Solution Architecture was found to be a very good example for using the capabilities of the Big Data Reference Architecture and the infrastructure provided by the Big Data platform

across the Use Cases [94, 95, 96, 97]. The Solution Architecture particularly pointed out, how two Use Cases could use a common ingestion mechanism via the Kafka Streaming platform and store claims data in a common data lake. Thus redundancies between the two Use Cases concerning application components, infrastructure and data storage could be eliminated.

7. Conclusion

Finally, this last chapter provides a short summary of the artifacts developed in this thesis and offers an outlook for future work on Big Data and respective architectures in the insurance sector.

7.1 Summary

Here the answers to the Research Questions from section 1.2 are briefly summarized.

Research Question 1: What are possible Big Data Use Cases in the insurance sector and which ones do have the highest potential?

A number of Big Data Use Cases for the insurance sector (13 in total) has been identified within this thesis. The Use Cases are clustered in four categories: customer analytics, internal processes, Internet of Things for P&C insurance and finally Smart Health and Smart Life insurance. Thus the Use Cases identified cover practically all areas, products and service lines in an insurance company. 12 Use Cases are rather business-focused whilst one – Analysis of Enterprise Architecture and Business Processes Based on Monitoring Data – can be seen as technically oriented and supporting some of the other Use Cases, such as e.g. targeting. All 12 of the business oriented Use Cases are intended to bring strategic advantages to an insurance company, either through significant cost reduction or risk minimization, growth creation or offering entirely new products. However the evaluation showed that several Use Cases are not really promising when it comes to implementing them. Above all some Use Cases that are judged to deliver a high added business value to an insurance company are also regarded as very difficult to implement. The reasons are either technical (complex models or lack of data) but sometimes also organizational (e.g. if a Use Case causes restructuring measures in an insurance company). Yet, there are several Use Cases that do promise a reasonable added value whilst still being easy to implement and operationalize.

Research Question 2: Which requirements have to be fulfilled in order to implement the Use Cases from RQ1?

For each Use Case a complete requirements analysis was conducted. In order to bring in some structure and make the requirements comparable a template was used that covered all key aspects. These included data sources, data characteristics such as the 4 V's (Volume, Velocity, Variety and Veracity), data privacy and security issues, required analytical algorithms and business requirements. The latter consist of possible organizational impacts and whether enough technical know-how is available. It is not possible to generalize all the Use Case specific requirements, however the number of Use Cases needing data processing at near-real-time speed, at least for some scenarios in a Use Case, is quite significant. The Big Data Reference Architecture designed later and particularly the Big Data platform did take this strongly into account. All the Use Case specific requirements were further analyzed and finally grouped into generic requirements that were clustered into similar categories as described

previously. All of the generic requirements – with the exemption of the ones dealing with business and organizational requirements – were afterwards mapped to components in the Big Data Reference Architecture. Besides providing a good overview of all requirements that have to be fulfilled for operationalizing a Use Case, the requirements offer a good link between a Use Case and the required components – that come from the Big Data Reference Architecture – for implementing it.

Research Question 3: What can a Big Data Reference Architecture look like in order to operationalize the Use Cases from RQ1?

Based on the generic requirements from RQ2 and an analysis of existing Big Data Reference Architectures a new Big Data Reference Architecture for the insurance sector has been designed. This new Reference Architecture functions as a modular kit from where different components can be picked out depending on the requirements of a Use Case in order to implement it. The Reference Architecture consists of several core components and cross-functional components that support the core components. Like in most other Big Data Reference Architectures, the core components consist of four layers or phases data items do pass: ingestion, storage, analytics & processing and finally visualization & operationalization. Additionally to the Big Data Reference Architecture this thesis offers a Big Data platform that provides Big Data components and infrastructure across the Use Cases. Thus redundancies in components and data storage can be reduced what in turn leads to lower costs and an improved data quality within the Master Data Lake, the core storage component. When a Use Case is implemented by combining the required components from the Reference Architecture the Big Data platform is used for the parts of the data pipeline that are not Use Case specific. Furthermore the Big Data platform provides Use Cases with ML frameworks for developing Use Case specific ML models. Additionally processing frameworks for training and later executing ML models in a predictive service are available. In order to demonstrate how to use the Big Data Reference Architecture and the Big Data platform, a Solution Architecture has been designed for claims automation and fraud detection – two Use Cases that were judged to be both highly promising but still complex to operationalize. Based on the results of the architecture evaluation both the Reference Architecture and the Big Data platform are regarded as perfectly fitting for implementing the Use Cases identified in this thesis. Eventually it can be said that the Big Data Reference architecture is able to serve as an architectural blueprint for any insurance company.

7.2 Outlook

This thesis has shown that Big Data can be applied in various ways for many different Use Cases in the insurance sector. It also offered a technical basis for implementing these Use Cases by providing a Big Data Reference Architecture and a Big Data platform. Yet the products and frameworks within the Big Data platform have to be continuously evaluated for whether they are still state-of-the-art and sufficient for fulfilling the requirements insurance Use Cases have. Keeping the Big Data Reference Architecture and using a platform approach

for reducing redundancies and assuring flexibility should always be part of the IT strategy of an insurance company.
The next steps for most insurance companies will be to start both exploratory and large-scale projects or programs for operationalizing Big Data Use Cases and solutions. With the technical path being set, it is up to insurance executives to decide which Use Cases their companies should implement. The Use Case evaluation methodology can be used by any insurer to evaluate with which Use Cases to start. Depending on their own results they could determine, which Use Cases are suited for a large scale implementation and which ones should be restricted to an exploratory PoC or dismissed at all.
Big Data and AI will play a key role in the insurance sector in the future as well – this is why insurance companies should also continuously explore possible new Use Cases and how they could influence the respective Big Data architecture. When exploring new Use Cases and redesigning the architecture, the business-driven approach used in this thesis should be preserved since it helps aligning business needs and the technical solutions for them. Exploring new opportunities is also essential because insurtechs and internet companies like Google and Amazon will seek new business opportunities in the insurance market and offer new products with a new, compelling customer experience there. Established insurance companies must not treat insurtechs with contempt – instead they should embrace their Big Data based approaches and try to develop similar products of their own. Insurers face the choice whether to become leaders in the usage of Big Data and AI or to be replaced by more innovative competitors providing a better customer experience, products and services.

Bibliography

[1] Kotalakidis, N. & Naujoks, H. & Mueller, F. *Digitalisierung der Versicherungswirtschaft: die 18-Milliarden Chance.* Google and Bain & Company, 2016.

[2] Catlin, T. & Morrison, C. & Lorenz, J. & Wilms. H.: *Facing digital reality*. Pages 7-17. McKinsey & Company, 2017.

[3] Chen, H. & Matthes, F. & Kazman, R.: *Demystifying Big Data Adoption: Beyond IT Fashion and Relative Advantage*. Page 6. Proceedings of PreICIS (International Conference on Information System) DIGIT workshop, 2015.

[4] Marr, B. *Using SMART Big Data, Analytics and Metrics To Make Better Business Decisions and Improve Performance.* Pages 23- 44, 59-65 and 108-134. Wiley, 2015.

[5] Buschbacher, F. & Stüben, J.: *Big Data Use Case Radar: Use Cases identifizieren, bewerten und definieren.* Pages 1-3. PricewaterhouseCoopers, 2014.

[6] Fox, G. & Chang, W.: *Big Data Use Cases and Requirements*. Pages 1-6. National Institute of Standards and Technologies, 2014.

[7] NIST Big Data Public Working Group Reference Architecture Subgroup: *NIST Big Data Interoperability Framework: Volume 6, Reference Architecture.* National Institute of Standards and Technologies, 2015.

[8] NIST Big Data Public Working Group Reference Architecture Subgroup: *NIST Big Data Interoperability Framework: Volume 5, Architectures White Paper Survey.* National Institute of Standards and Technologies, 2015.

[9] Interview with Lars George, conducted on 21.09.2017

[10] Peffers, K. & Tuunanen, T. & Rothenberger, M. & Chaterjee, S.: *A Design Science Research Methodology for Information Systems Research.* Pages 45-78. Journal of Management Information Systems, Volume 24 Issue 3, 2007.

[11] Buhl, H.U. & Röglinger, M. & Moser, F. & Heidemann, J.: *Big Data - Ein (ir-)relevanter Modebegriff für Wissenschaft und Praxis?* Pages 63-68. Wirtschaftsinformatik, Vol.55, Nr. 2, 2013.

[12] Gartner Hype Cycle from 2015. Source: https://www.gartner.com/newsroom/id/3114217 Last accessed on 31.10.2017.

[13] Plattner, H.: http://www.enzyklopaedie-der-wirtschaftsinformatik.de/lexikon/daten-wissen/Datenmanagement/Datenmanagement--Konzepte-des/Big-Data, 2017. Last accessed on 30.10.2017.

[14] Horvath, S.: Aktueller Begriff - Big Data. Page 1. https://www.bundestag.de/blob/194790/c44371b1c740987a7f6fa74c06f518c8/big_data-data.pdf, 2013. Last accessed on 31.10.2017.

[15] Aggarwal, S. & Manuel, N.: Big Data and analytics should be driven by business needs, not technology. Pages 1-2. McKinsey & Company, 2015.

[16] McAffee, A. & Brynjolfsson, E.: *Big Data: The Management Revolution.* Pages 62-68. Harvard Business Review, Nr.10 , 2012.

[17] Reinsel, D. & Gantz, J. & Rydning, J.: *Data Age 2025: The Evolution of Data to Life-Critical.* Pages 2-3. IDC, 2017.

[18] Dapp, T. & Heine, V.: *Big Data – Die ungezähmte Macht.* Pages 6-10. Deutsche Bank Research, 2014.

[19] Klein, D. & Tran-Gia, P. & Hartmann, M.: *Big Data.* https://www.gi.de/service/informatiklexikon/detailansicht/article/big-data.html. Last accessed on 01.11.2017.

[20] Podesta, J. & Pritzker, P. & Moniz, E.J. & Holdren, J. & Zients, J.: *Big Data - Seizing opportunities, preserving values.* Pages 2-5. Executive Office of the President - The White House, 2014.

[21] NIST Big Data Public Working Group Definitions and Taxonomies Subgroup: *NIST Big Data Interoperability Framework: Volume 1, Definitions.* National Institute of Standards and Technologies, 2015.

[22] McNulty-Holmes, E.: *Understanding Big Data: The Seven V's.* http://dataconomy.com/2014/05/seven-vs-big-data/ Last accessed on 02.11.2017

[23] Schapire, R.: *Theoretical Machine Learning.* https://www.cs.princeton.edu/courses/archive/spr08/cos511/scribe_notes/0204.pdf, 2008. Last accessed on 03.11.2017.

[24] Douetteau, F.: *A Beginner's Guide to Machine Learning Algorithms.* http://dataconomy.com/2017/03/beginners-guide-machine-learning/, 2017. Last accessed on 03.11.2017.

[25] Alpaydin, E.: *Introduction to Machine Learning.* Pages 1-41 and 185-192. MIT Press, 2010.

[26] Reamy, T.: *Deep Text – Using Text Analytics to Conquer Information Overload, Get Real Value From Social Media, and Add Big(ger) Text to Big Data.* Pages 21-36. Information Today, 2016.

[27] Sanders, L. & Woolley, O. & Moize, I. & Antulov-Fantulin, N.: *Introduction to Sentiment Analysis.* Pages 3-12. Eidgenössiche Technische Hochschule Zürich, 2017.

[28] Jurafsky, D.: *Sentiment Analysis.* Pages 13-15. https://web.stanford.edu/class/cs124/lec/sentiment.pdf Last accessed on 05.11.2017.

[29] Salathé, M. & Khandelwal, S.: *Assessing Vaccination Sentiments with Online Media: Implications for Infectious Disease Dynamics and Control.* https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3192813/, 2011. Last accessed on 05.11.2017

[30] Wu, T.L.: *An Overview of Present NoSQL Solutions and Features.* Pages 1-4 and 6. Indiana University, 2013.

[31] https://de.wikipedia.org/wiki/NoSQL. Last accessed on 07.11.207

[32] Ellingwood, J.: *Hadoop, Storm, Samza, Spark and Flink: Big Data Frameworks Compared.* https://www.digitalocean.com/community/tutorials/hadoop-storm-samza-spark-and-flink-big-data-frameworks-compared, 2016. Last accessed on 09.11.2017.

[33] http://fortune.com/2015/09/09/cloudera-spark-mapreduce/ Last accessed on 08.11.2017.

[34] https://kudu.apache.org Last accessed on 09.11.2017.

[35] Hein, O.: *Fast Analytics on Fast Data*. Cloudera Sessions Munich, 2017.

[36] Mayo, M.: *Top Big Data Processing Framework.* https://www.kdnuggets.com/2016/03/top-big-data-processing-frameworks.html, 2016. Last accessed on 09.11.2017

[37] https://spark.apache.org. Last accessed on 09.11.2017.

[38] Mayer-Schönberger, V. & Cukier, K.: Big Data: *Die Revolution, die unser Leben verändern wird.* Pages 74-76. Redline, 2013.

[39] Mäder, P.: *Der smarte Versicherer: Verankerung von Big Data in der Unternehmensstrategie.* BearingPoint Institute, 2014.

[40] Verma, S. & van Deel, L. & Nadimpalli, R & Sahoo, D.: *Big Data Analytics in Life Insurance.* Pages 7-11. Capgemini, 2014.

[41] Maas, P. & Milanova, V.: *Zwischen Verheissung und Bedrohung – Big Data in der Versicherungswirtschaft.* Pages 23-25. Die Volkswirtschaft, Nr. 5, 2014.

[42] https://de.statista.com/statistik/daten/studie/225953/umfrage/die-weltweit-meistgenutzten-suchmaschinen/ and https://de.statista.com/statistik/daten/studie/167841/umfrage/marktanteile-ausgewaehlter-suchmaschinen-in-deutschland/ Last accessed on 11.11.2017.

[43] Matthes, F.: *Software Engineering betrieblicher Anwendungen: Requirements Engineering.* Pages 5-10. Technical Unversity Munich, 2016.

[44] Bruegge, B.: *Requirements Elicitation.* Pages 12-20 and 40-54. Technical Unversity Munich, 2016.

[45] Kurkovsky, S.: *Software Engineering – Requirements Engineering*. Central Connecticut State University, 2011.

[46] NIST Big Data Public Working Group: *Big Data Use Case Template 2.* Pages 1-7. National Institute of Standards and Technologies, 2017.

[47] Schaar, P.: Datenschutz in Zeiten von Big Data. Pages 842-852. HMD – Praxis der Wirtschaftsinformatik, Vol. 51, 2014.

[48] German Ministry of Justice: Paragraph 3a, Bundesdatenschutzgesetz. https://www.gesetze-im-internet.de/bdsg_1990/__3a.html Last accessed on 10.11.2017.

[49] Seibel, K.: Gegen Kreditech ist die Schufa ein Schuljunge. http://www.welt.de/finanzen/verbraucher/article139671014/Gegen-Kreditech-ist-die-Schufa-ein-Schuljunge.html, 2014. Last accessed on 10.11.2017.

[50] Jarmul, K.: *GDPR and You: Benefits of Secure, Privacy-Concerned Data Management.* Cloudera Sessions Munich, 2017.

[51] European Union: Article 17, General Data Protection Regulation. https://www.privacy-regulation.eu/en/17.htm. Last accessed on 10.11.2017.

[52] Hilliard, R.: *All About IEEE Std 1471.* Page 13. IEEE, 2007.

[53] http://pubs.opengroup.org/architecture/togaf9-doc/arch/chap03.html. Last accessed on 11.11.2017.

[54] http://pubs.opengroup.org/architecture/togaf9-doc/arch/ (Part 2 – Refer to the description of the respective ADM phase for each architecture). Last accessed on 11.11.2017.

[55] Buchanan, R.: *Enterprise Architecture Program*. Page 2. Gartner, 2010.

[56] Cloutier, R. & Muller, G. & Veram, D. & Nilchiani, R. & Hole, E. & Bone, M.: *The Concept of Reference Architectures.* Pages 14-17. Wiley InterScience, 2009.

[57] Lanquillon, C. & Mallow, H.: *Big Data Lösungen.* Pages 263-278. From "Praxishandbuch Big Data", Springer, 2015.

[58] https://www.gartner.com/newsroom/id/3130017. Last accessed on 12.11.2017.

[59] https://yougov.de/loesungen/ueber-yougov/presse/presse-2015/pressemitteilung-wechseltaetigkeit-in-der-kfz-versicherung-2014/ Last accessed on 12.11.2017.

[60] Gallo, A.: The Value of Keeping the Right Customers. Harvard Business Review, 2014. https://hbr.org/2014/10/the-value-of-keeping-the-right-customers Last accessed on 12.11.2017.

[61] Interview with Lars George from 04.07.2017

[62] Brat, E. & Heydorn, S. & Stover, M. & Ziegler, M.: Big Data: The Next Big Thing for Insurers. BCG Perspectives, 2013. https://www.bcgperspectives.com/content/articles/insurance_it_performance_big_data_next_big_thing_for_insurers/. Last accessed on 09.12.2017

[63] https://www.wirtschaftswissen.de/marketing-vertrieb/werbung/online-marketing/opt-in-oder-opt-out-so-sorgen-sie-fuer-die-richtige-werbeeinwilligung/. Last accessed on 25.11.2017

[64] Interview with Nikolaos Radouniklis from 16.07.2017.

[65] *How to Effectively Fight Insurance Fraud.* Pages 3-6. Accenture, 2013.

[66] Higginson, M. & Lorenz, J.T. & Olesen, P.B. & Münstermann, B.: The promise of blockchain. Page 69. McKinsey & Company, 2017.

[67] *Intel and Cloudera Reduce Insurance Fraud and Dramatically Improve Time to Access Claim Data.* Pages 1-3. Intel, 2015.

[68] Manyika J., Chui M., Miremadi M., Bughin J., George K., Wilmott P., Dewhurst M.: *A future that works: automation, employment and productivity. Executive summary.* Pages 4-8. McKinsey Global Institute, 2017.

[69] Frey C.B., Osborne, M.: *The future of employment: How susceptible are jobs to computerization?* Page 72. Oxford University, 2013.

[70] http://mainichi.jp/english/articles/20161230/p2a/00m/0na/005000c. Last accessed on 02.12.2017.

[71] https://news.sap.com/earth-observation-analysis-service-powered-by-sap-hana/

[72] Balasubramanian, J. & Beiker, S. & Chauhan, S. & Colombo, T. & Hansson, F. & Inampudi, S. & Jaarsma, R. & Kässer, M.: *Car Data: Paving the way to value-creating mobility.* Page 8. McKinsey&Company - Advanced Industries, Nr. 03, 2016.

[73] *Automobilindustrie 4.0: In Ettapen zu digital vernetzten Wertschöpfungsketten.* Pages 4-5. IBM Germany, 2015.

[74] Truong, A.: A New Take On Auto Insurance: Pay By The Mile. FastCompany, 2014.

[75] https://www.willistowerswatson.com/en/insights/2017/07/not-all-telematics-data-is-created-equally. Last accessed on 03.12.2017.

[76] Tiedemann, M.: *Die Top 9 Big-Data- und Data-Science-Trends in der Versicherungsbranche. Teil 2: Datengetriebene Produkte und Services*. Alexander Thamm, 2017. https://www.alexanderthamm.com/artikel/die-top-9-big-data-und-data-science-trends-in-der-versicherungsbranche-teil-2-datengetriebene-produkte-und-services/. Last accessed on 08.12.2017.

[77] http://www.gdv.de/zahlen-fakten/schaden-und-unfallversicherung/wohngebaeudeversicherung/. Last accessed on 30.11.2017.

[78] https://www.gartner.com/newsroom/id/3598917. Last accessed on 30.11.2017.

[79] http://www.procontra-online.de/artikel/date/2017/05/generalis-vitality-programm-droht-aerger/. Last accessed on 06.12.2017

[80] Schadt, E. & Chilukuri, S.: The role of big data in medicine. McKinsey & Company, 2015. https://www.mckinsey.com/industries/pharmaceuticals-and-medical-products/our-insights/the-role-of-big-data-in-medicine. Last accessed on 04.12.2017.

[81] Carter, R.: *Could technology be the cure?* Pages 14-17. Healthcare 3.0, KPMG, 2015.

[82] http://www-03.ibm.com/press/us/en/pressrelease/43231.wss. Last accessed on 05.12.2017.

[83] Bakalar, R.: *Making data count*. Pages 7-10. Healthcare 3.0, KPMG, 2015.

[84] http://www.reuters.com/article/us-cybersecurity-hospitals-idUSKCN0HJ21I20140924

[85] https://venturebeat.com/2013/08/19/amazon-website-down/. Last accessed on 01.12.2017.

[86] *NTT Data Big Data Architecture – Version 1.0.* Pages 3-5. NTT Data, 2015.

[87] https://cloud.google.com/solutions/data-lifecycle-cloud-platform#explore_and_visualize Accessed on 08.12.207.

[88] Osterloh, A.: *Processing data at scale with Google Cloud BigQuery, Dataprep, Dataflow and Dataproc.* Google Cloud Summit Munich, 2017.

[89] http://www.storagereview.com/google_tackles_big_data_through_updates_to_bigquery_dataflow, last accessed on 07.12.2017.

[90] Kleehaus, M. & Uludag, Ö & Matthes, F.: *Towards a Multi-Layer IT Infrastructure Monitoring Approach based on Enterprise Architecture Information*. Pages 12-14. 2nd Workshop on Continuous Software Engineering, Hannover, 2017.

[91] https://blogs.msdn.microsoft.com/robinlester/2016/03/09/architecting-a-big-data-project-in-azure/, last accessed on 08.12.2017.

[92] Interview with a Cloud Solution Architect, conducted on 15.12.2017.

[93] Interview with a Senior Enterprise Architect, conducted on 29.11.2017.

[94] Interviews with Lars George, conducted on 15.12.2017 and 09.01.2018.

[95] Interview with a Cloud Solution Architect, conducted on 25.01.2018.

[96] Interview with a Senior Data Scientist, conducted on 04.01.2018.

[97] Interview with Program Lead Central Data Storage, conducted on 11.01.2018.

[98] https://www.lemonade.com, last accessed on 10.03.2018.