The present work was submitted to the

Chair of Computer Science 5 - Information Systems
RWTH Aachen University

Developing an Architecture for Data Quality Measurement to Achieve Utility-driven Data Suppression

authored by:

Sanchit Alekh

Matriculation Nr. : 359831

Supervisors:

Prof. Dr. Stefan Decker, RWTH Aachen University

Prof. Dr. Christoph Quix, RWTH Aachen University

Advisors:

Christoph Kreibich, AUDI AG

Dr. Sandra Geisler, Fraunhofer Institut für Angewandte Informationstechnik FIT

Aachen, 28. January 2019

Zentrales Prüfungsamt/Central Examination Office

Eidesstattliche Versicherung
Statutory Declaration in Lieu of an Oath

Name, Vorname/Last Name, First Name

Matrikelnummer (freiwillige Angabe)
Matriculation No. (optional)

Ich versichere hiermit an Eides Statt, dass ich die vorliegende Arbeit/Bachelorarbeit/ Masterarbeit* mit dem Titel
I hereby declare in lieu of an oath that I have completed the present paper/Bachelor thesis/Master thesis* entitled

selbstständig und ohne unzulässige fremde Hilfe (insbes. akademisches Ghostwriting) erbracht habe. Ich habe keine anderen als die angegebenen Quellen und Hilfsmittel benutzt. Für den Fall, dass die Arbeit zusätzlich auf einem Datenträger eingereicht wird, erkläre ich, dass die schriftliche und die elektronische Form vollständig übereinstimmen. Die Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.
independently and without illegitimate assistance from third parties (such as academic ghostwriters). I have used no other than the specified sources and aids. In case that the thesis is additionally submitted in an electronic format, I declare that the written and electronic versions are fully identical. The thesis has not been submitted to any examination body in this, or similar, form.

Ort, Datum/City, Date

Unterschrift/Signature

*Nichtzutreffendes bitte streichen

*Please delete as appropriate

Belehrung:
Official Notification:

§ 156 StGB: Falsche Versicherung an Eides Statt
Wer vor einer zur Abnahme einer Versicherung an Eides Statt zuständigen Behörde eine solche Versicherung falsch abgibt oder unter Berufung auf eine solche Versicherung falsch aussagt, wird mit Freiheitsstrafe bis zu drei Jahren oder mit Geldstrafe bestraft.
Para. 156 StGB (German Criminal Code): False Statutory Declarations
Whoever before a public authority competent to administer statutory declarations falsely makes such a declaration or falsely testifies while referring to such a declaration shall be liable to imprisonment not exceeding three years or a fine.
§ 161 StGB: Fahrlässiger Falscheid; fahrlässige falsche Versicherung an Eides Statt
(1) Wenn eine der in den §§ 154 bis 156 bezeichneten Handlungen aus Fahrlässigkeit begangen worden ist, so tritt Freiheitsstrafe bis zu einem Jahr oder Geldstrafe ein.
(2) Straflosigkeit tritt ein, wenn der Täter die falsche Angabe rechtzeitig berichtigt. Die Vorschriften des § 158 Abs. 2 und 3 gelten entsprechend.
Para. 161 StGB (German Criminal Code): False Statutory Declarations Due to Negligence
(1) If a person commits one of the offences listed in sections 154 through 156 negligently the penalty shall be imprisonment not exceeding one year or a fine.
(2) The offender shall be exempt from liability if he or she corrects their false testimony in time. The provisions of section 158 (2) and (3) shall apply accordingly.

Die vorstehende Belehrung habe ich zur Kenntnis genommen:
I have read and understood the above official notification:

Ort, Datum/City, Date

Unterschrift/Signature

Sperrvermerk

Die vorliegende Masterarbeit basiert auf internen, vertraulichen Daten und Informationen des Unternehmens AUDI AG. Diese Arbeit darf daher nur den Erst- und Zweitgutachter sowie den befugten Mitgliedern der Prüfungsorgane der Hochschule zugänglich gemacht werden. Eine Veröffentlichung und Vervielfältigung der Arbeit ist - auch in Auszügen - nicht gestattet. Eine Einsichtnahme der Arbeit durch Unbefugte bedarf einer ausdrücklichen Genehmigung durch den Verfasser und das Unternehmen.

Confidential Clause

This Master thesis is based on internal, confidential data and information of the AUDI AG. This work may only be available to the first and second reviewers and authorized members of the board of examiners. Any publication and duplication of this master thesis - even in part - is prohibited. An inspection of this work by third parties requires the expressed admission of the author and the company.

Acknowledgements

This Master Thesis work was written in co-operation with the Production IT department at the AUDI AG from July to December 2018. It would not have been possible without the unflinching faith and support of my professors, advisors and colleagues. To these people, I owe a heartfelt gratitude and appreciation.

In particular, I would like to thank PD Dr. Christoph Quix, who gave me the opportunity to work on an interesting and relevant research problem. I would also like to thank Dr. Sandra Geisler, who reinforced me with her thoughts, ideas and valuable feedback throughout the course of the thesis. Thank you for your exceptional patience, and for believing in me. I would also like to express my gratitude to Prof. Dr. Stefan Decker for kindly consenting to be my first advisor.

A big thank you to Christoph Kreibich and Christian Wolf of AUDI AG, who steered me through all technical and bureaucratic difficulties during my stay in Neckarsulm. They were always ready to answer my questions and arrange meetings with important stakeholders, which was extremely important for this work. I would also like to thank my colleagues at the TechHub Data Driven Production division at Audi, who were always very supportive. It was truly enriching and fruitful to have engaging discussions with you.

Last but not least, I would like to thank my beloved parents, who always showed tremendous faith in me, provided much-needed mental support and never let me doubt my ability, especially when the going got tough. Thank you Mummy and Papa for your unconditional love, and for always being there.

Abstract

Automotive companies collect a plethora of process and product data, ranging from data originating from sensors in the production line, to sales and marketing data. This presents a great opportunity, as they can fine-tune complex processes based on historical data. However, it also comes with two main challenges: lack of guarantees about the reliability of the collected data, and a dearth of comprehensive knowledge about the use-cases that can leverage the power of this data.

In cases where there is no indicative information about the trustworthiness of the collected data, ascertaining the data quality is a promising and multi-faceted solution. On the other hand, to make sure that companies get the most out of their data, mutually-beneficial data sharing is important. For data sharing scenarios, data quality guarantees are also essential in order to establish an environment of trust and amity. A transparent definition and use of privacy-enhancing technologies, followed by a comprehensive evaluation of their subsequent impact on data quality is also desirable.

In this thesis, we provide a solution to the aforementioned challenges by the following contributions: we create an effective measure for the quality of data collected within the organization. Furthermore, we use privacy-enhancing technologies such as anonymization and attribute suppression as a means to gain granular control over data quality and privacy. In this way, we create a mechanism to evaluate data quality, and to augment data privacy in a controlled manner, for use-cases such as data sharing with third parties. Thereby, we ensure that organizations can choose a suitable trade-off between data quality and privacy, instead of guaranteeing one or the other, so that they can create differentiated business value from their collected data.

Contents

List of Figures

List of Tables

1 Introduction

Data Science has risen to become one of the most prominent tools for organizations to plan, organize, monitor and improve their products, processes and services. The need and demand for efficient ways to store and analyse large amounts of data led to the development of new tools and techniques for the same. This ushered in a paradigm shift in the way we interacted with our data, and consequently, larger and more traditional companies such as General Electric, Daimler and Audi joined the bandwagon[DD13]. Nowadays, the increased efficiency of learning algorithms, a steady increase in processing power and the flourishing of open-source technologies in the field of data science has ensured that big data analytics capabilities are not only available to a select few bigwigs in the industry, but to companies of all sizes and trades. Especially in the automotive industry, where processes are increasingly complex and even small optimizations lead to huge reductions in the cost of production, data is widely regarded as the new oil[1].

1.1 Big Data in Automotive Production

Automotive companies are extremely competitive on a global scale, and are always trying to cut down their production costs, while making absolutely no compromise with quality and safety [CHT09]. Especially in high-wage economies such as Germany, this is only possible by increasing the efficiency and reducing the costs of these production processes by making efficient utilization of the collected data.

For example, in its body shop, AUDI AG uses classification and clustering techniques to monitor processes such as welding, screwing and riveting. By scoring the real-time data from these processes on pre-computed models trained on historical data, quality assurance is further enhanced. Being able to detect faults in the processes as they occur is extremely cost-effective, as it ensures that remedial actions can immediately be performed. This also saves the expensive cost and effort of random inspections, where the entire chassis has to be deconstructed to its original parts. In the paint shop, the company uses a complex machine learning model to predict the fitness of the painting

[1]Phrase first coined by Clive Humby, British mathemetician and architect of Tesco's Clubcard, 2006

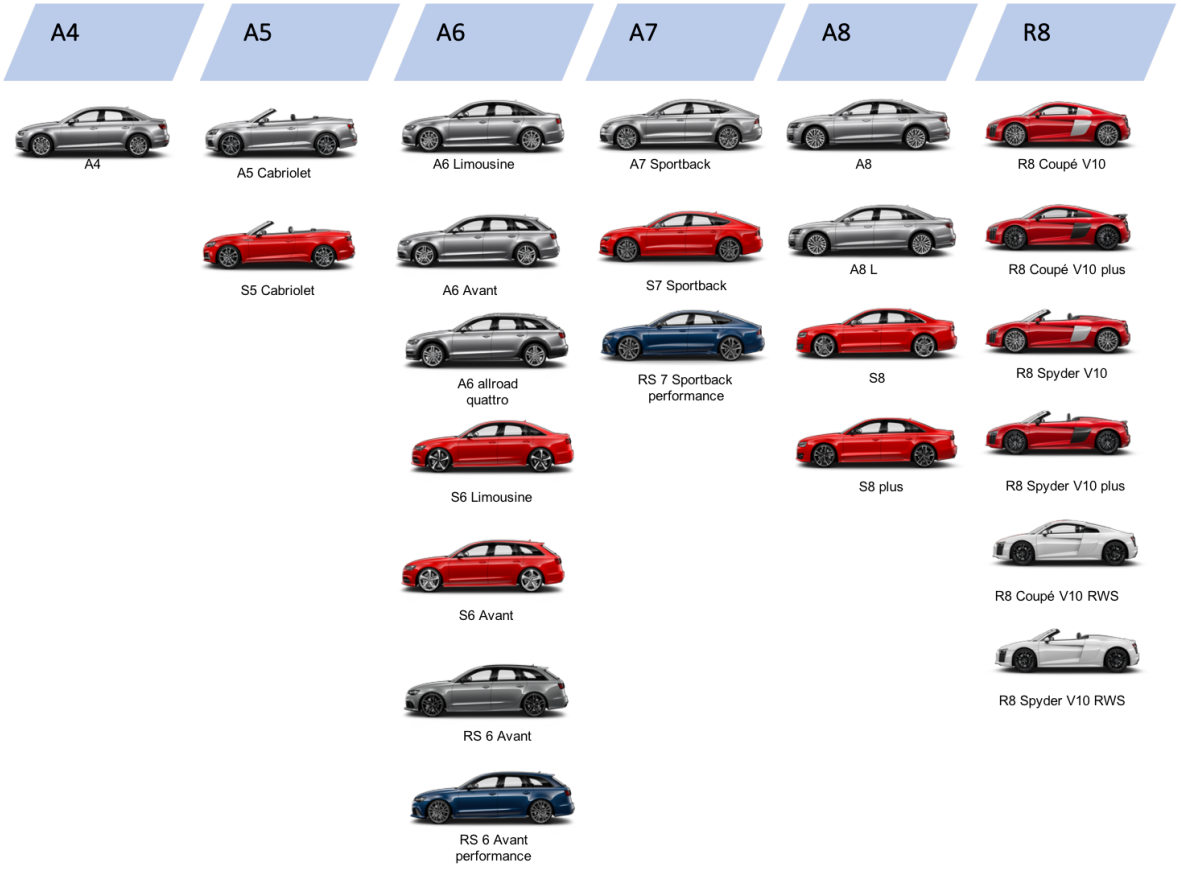

Figure 1.1: Models Produced at Audi's Neckarsulm Factory

process, based on a collection of 1500-3000 variables [AG17]. This model is currently implemented at Audi's Gyor factory in Hungary. In the assembly line, drawing analyses from data is very useful, especially considering the fact that unlike old days, cars are no longer series-produced. Indeed, each and every car is built for a specific customer, and this increases the complexity of the assembly processes. As illustrated in Figure 1.1, Audi's Neckarsulm factory is one of Europe's most complex automotive manufacturing facilities based on the number of car models that it produces. This requires a well-coordinated effort between the different departments, most importantly production and logistics. To aid the logistics of *just-in-time* and *just-in-sequence* production, data science can be used very effectively. For example, the time of completion of production of a particular automobile can be predicted using a regression model, and the cars can be dispatched in batches to their destinations without further delays. This saves a lot of storage space inside the factory premises, where space is always a luxury.

Property	Data Warehouses	Data Lake
Data Structuring	Structured, processed	Structured/Semi-Structured/Unstructured/Raw
Data Integration	Schema-on-write (schema first)	Schema-on-read (schema last)
Data Storage Costs	Expensive for large volumes of data	Low-cost storage due to use of commodity hardware
Querying	Limited to pre-defined views	Highly flexible
Agility	Low agility, schema is mostly fixed, reconfiguration is difficult	Highly agile, configuration is possible whenever required
Maturity	Highly mature, in-use by several businesses	Maturing and rapidly developing

Table 1.1: Key Differences Between Data Warehouse and Data Lakes

1.2 Production Big Data: A case study of Audi

As described in the previous section, automotive manufacturing benefits massively from the ability to generate and collect precise information about production processes. Therefore, automotive companies collect large amounts of data, ranging from data from robots in production, to data about sales, marketing and business management. To handle such a varied collection of heterogeneous sources of data, especially Internet of Things (IoT) data, traditional database solutions such as relational databases and data warehouses were seen as infeasible. Using RDBMS for data storage leads to creation of silos with little or no interconnectedness, which makes analysis of combined data next to impossible [Ter+15]. In this context, a *Data Lake* presents an interesting medium for data storage and analysis. It serves as an infinite, schema-less repository for raw data with a common access interface [HGQ16]. This capability makes it possible to perform an assimilated knowledge extraction from wide-ranging data sources, irrespective of their schema. Table 1.1 lists the key differences between data warehouses and data lakes.

To support this knowledge-extraction endeavour, an organization-wide project called the *Data Lake P (DLP)* was introduced at AUDI AG.

The main aim of this project was to lay the basis for the development of a big data architecture for production from the ground-up, to support the company's endeavours towards building factories of the next generation, where production would indeed be fully data-driven and self-orchestrated. Benefiting from the architecture described in Figure 1.2, several production processes such as welding, screwing and riveting started publishing real-time data into the DLP. This led to exciting prospects for applications such as predictive maintenance, condition monitoring and process optimization. Consequently, this has also raised the stakes and necessitated that analytics tasks are performed with

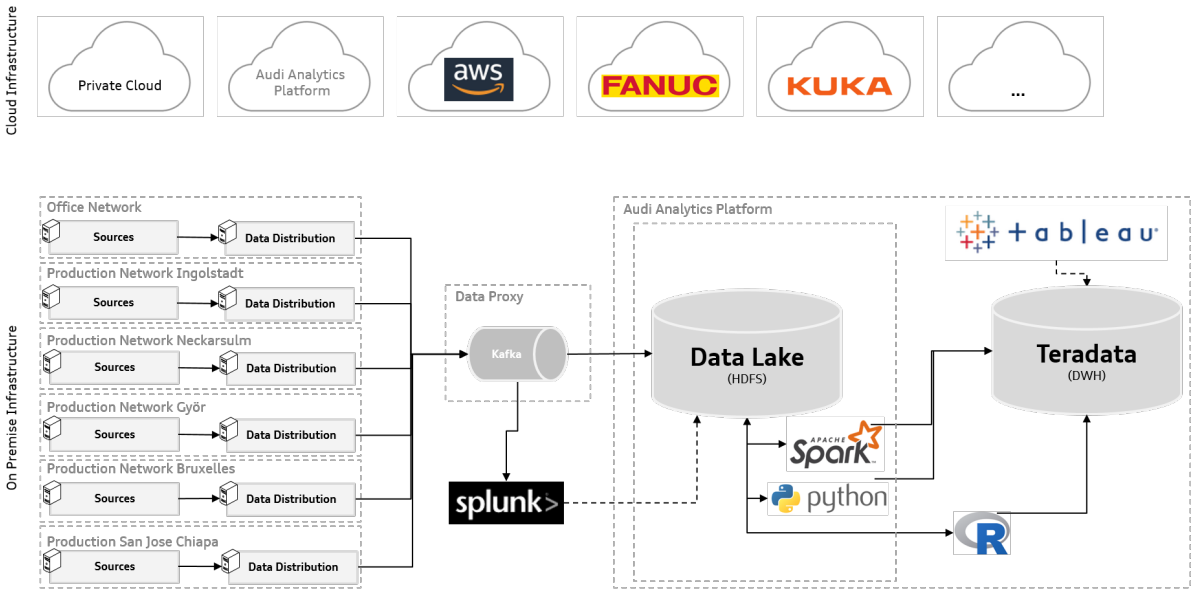

Figure 1.2: Big Data Architecture Within the *Data Lake P* Project *Source: AUDI AG*

high-quality data to generate trustworthy insights which create business value. Indeed, the notion of *Data Quality* has become extremely important not just within the scope of the DLP, but organization-wide, to better understand the qualitative aspects of the collected data.

1.3 Poor Data Quality: A Missed Opportunity

By this time, it has already been established that poor data quality can have substantial social and economic impacts on organizations [WS96; Red98]. This is especially true in the automotive industry, where data is used in a multitude of ways: from optimizing the supply chain and procurement process, to production, quality control, sales, after-sales and R&D, as shown in Figure 1.3. Such a massive inter-departmental use of data necessitates that the data be consistent, measurable and trustworthy. Consider the following use-case: the *after-sales* department would like to use the data collected by the *sales* department to selectively advertise winter tyres to customers of sedans without all-wheel drive. In a separate use-case, the *R&D* department would like to use the data collected by *after-sales* to determine the parts which needed to be serviced most. These are two hypothetical scenarios where inter-departmental data sharing can bring huge benefits for making data-driven decisions. Now, if there is no trust in the data provided by other departments, these data-driven use-cases are bound to fail. Furthermore, this

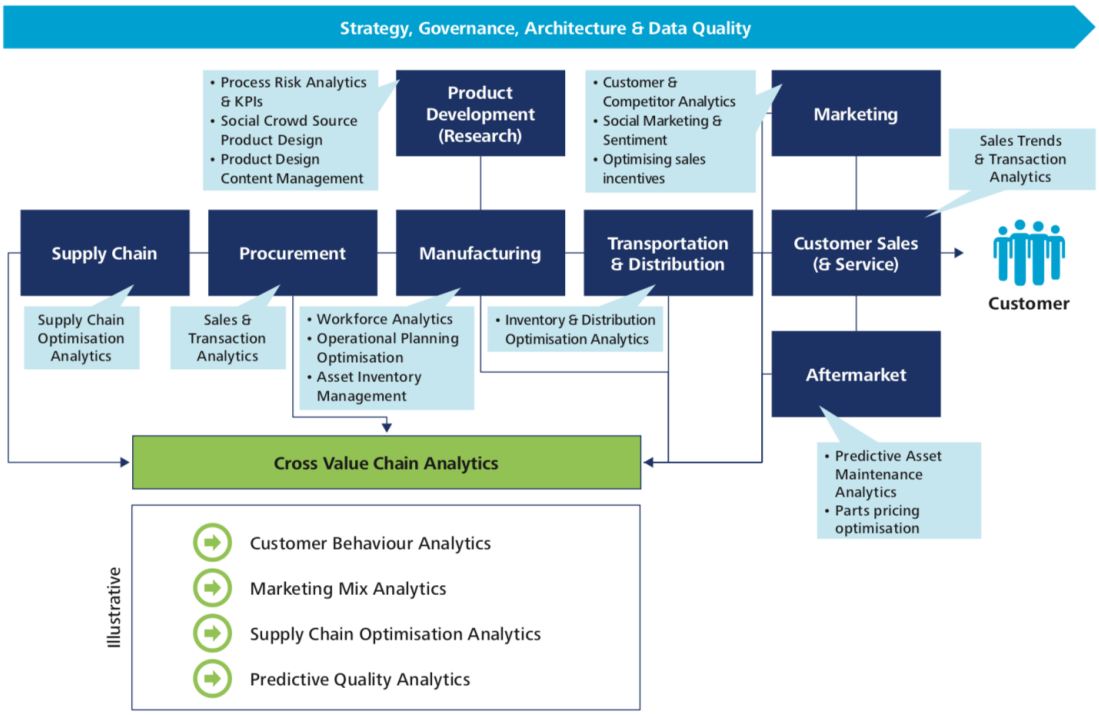

Figure 1.3: Various Ways in Which Automotive Companies Use Data. Taken from *Deloitte-Big Data and Analytics in the Automotive Industry* [Del]

will lead to each department trying to create its own (often inconsistent with others) database, even when similar data exists at other departments, leading to siloing and duplication of data.

This is a missed opportunity, especially for companies using big data platforms and technologies, where one of the foundational objectives is to integrate and open up data that has traditionally existed in silos within departments. This means that data quality concerns and intransparent practices can defeat the whole purpose of big data architectures, such as data lakes. In the forthcoming subsections, we see in more detail how data quality concerns can affect the automotive industry, and what we can really achieve by developing an approach for data quality calculation in data lakes.

1.3.1 Impact of Poor Data Quality on the Automotive Industry

In his comprehensive article[Red98] on how data quality affects enterprises, Redman argues that poor data quality can lead not only to customer dissatisfaction, increased operational costs, less effective decision-making, and a reduced ability to make and execute a strategy, but also hurts employee morale, breeds organizational mistrust, and

makes it more difficult to align the enterprise. Redman estimates the cost of bad data quality to the tune of 8-12% of the revenue of an enterprise, as found by proprietary studies.

Then, there is also the added scenario where decision-makers are unaware of the data quality, leading to decisions made on flawed or incorrect data. All of these aforementioned points remain relevant to the automotive industry. The impact of poor quality can be broken into 3 factions:

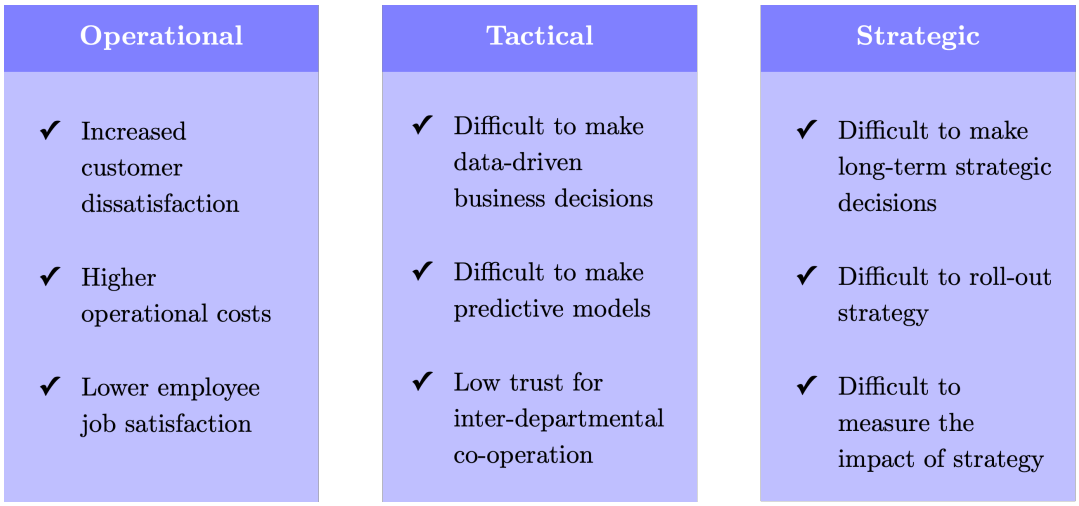

Figure 1.4: The Impacts of Poor Data Quality

1. **Operational Impact:** Poor data quality leads to *customer dissatisfaction*, *increased cost*, and *lowered employee job satisfaction*[Red98]. If imprecise or inadequate data is collected about a customer, e.g., in the sales department, the customer profiling will also be incorrect, leading to the customer being targeted with false advertisements or offers. In the domain area of *production IT*, bad data leads to hugely increased costs, if process data is incorrectly collected and used for process monitoring. This will require a much higher human interference for quality control, slowing down processes and bringing the costs up. More time and effort is spent on rectifying the errors lurking inside the data. In a more subtle way, it may also lead to employee dissatisfaction, e.g., in the case of *data scientists*, whose efficacy directly depends on the quality and insightfulness of the data they operate upon. Poor quality data will lead to a situation where data scientists can not perform their job properly.

2. **Tactical Impact:** More and more companies today base their tactical decisions on data. Therefore, it is surely the case that consistent, accurate, timely and

trustworthy data is likely to play a major role in advancing the organizational goals. On the other hand, poor quality data compromises decision making. Business decisions are no better than the data they are based on. In the automotive sector e.g., bad data makes it very difficult to create predictive models for use-cases such as predictive maintenance, which depends on high-quality labelled data. It also makes it very difficult to establish trust for inter-departmental data exchange.

3. **Strategic Impact:** The strategic impact of poor quality data is somewhat related to the tactical impact, because developing the organizational strategy itself is a form of decision-making. However, the effects of strategic decisions are much more far-reaching and can have long-term consequences. In this regard, poor data quality hinders not just the setting of strategy, but also its roll-out and execution. It also makes it difficult to measure the results of the strategic decisions, which could lead to misguided beliefs or hurt employee morale.

1.3.2 Data Quality in Data Lakes

Although data lakes are able to very efficiently tackle heterogeneity in the data sources, they still don't guarantee the effectiveness of the data analytics processes, and whether they are able to generate insights which add business value. Indeed, the effectiveness of a data mining exercise depends critically on the quality of the data [HMS01], and the garbage-in, garbage-out phenomenon can not be overlooked here. The worst possible case is that the most interesting patterns we discover during a data mining exercise will have resulted from measurement inaccuracies, distorted samples or some other unknown false values [HMS01]. Indeed, one of the major pitfalls is the belief that 'raw' data can simply be dumped into a data lake without any data quality analysis or metadata management. This leads to a 'data swamp' rather than a 'data lake'[2].

Terrizano et al.[Ter+15] put forward a very interesting opinion in this regard. According to the authors, "a raw data lake does not enhance the agility and accessibility of data, since much of the necessary data massaging is simply postponed, potentially to a time far removed from the moment that the data was acquired". Their concerns are definitely not misplaced, and this highlights the importance of data quality management in data lake systems. For a large automotive company, computing the data quality of their datasets is an exercise that has multifarious utilities, some of which we describe below:

[2]https://cacm.acm.org/blogs/blog-cacm/181547-why-the-data-lake-is-really-a-data-swamp/fulltext

1. **Data Governance and Cataloguing:** Companies set up enterprise big data architectures at huge costs with the assumption that the value obtained will exceed its cost in the long run[Tal13]. Since this belief is hardly ever put to test, companies like to justify their investment by ensuring that high-quality data is collected and maintained throughout its life-cycle. Data quality measures also help in data cataloguing efforts, such that enterprise-wide data is available for research and development purposes. Moreover, data cataloguing aims to solve an extremely widespread problem that plagues large organizations. On one hand, organizations are collecting increasingly large amounts of data about their products and processes, but on the other hand, business departments often complain about the lack of meaningful metadata and attributes that would help them select particular datasets for a particular analytics use-case.

2. **Data Selection:** Data quality measures help in selecting data for building machine learning models. This ensures that important analyses are made only with most relevant datasets, containing data of the finest quality, to guarantee a higher degree of correctness of the generated insights.

3. **Determining the Effort for Data Cleaning:** Although percentage estimates vary, most scientists argue that *data cleaning* is often the most involving of all the data science and engineering operations[ZZY03]. Getting a fine-grained measure of data quality is a huge step forward in calculating the effort involved in data cleaning. This is especially helpful to concentrate data cleaning efforts towards attributes or parts of the dataset which have the worst quality metrics.

4. **Data Migration:** Several enterprises maintain an in-house data warehouse to store critical and sensitive data important for the company. The data lake serves as the staging area for the data warehouse. Data quality measurement aids the decision regarding which data is more valuable to the organization, and therefore, should be migrated to the data warehouse. It can also be used to off-loading priorities when the data becomes too large to be stored on-premises.

5. **Selling Data to Third-parties:** This is another interesting use-case of data quality measurement. Companies may choose to create additional value out of their data by selling it to third parties. Quality metrics can be used as guarantees to the customer. In addition, the data owner will be able to sell the data at a more fair price if it itself knows how valuable the data is. Section 1.5 contains a more detailed treatment of this subject.

1.4 Data Quality Suppression with Privacy-Enhancing Technologies

A critical problem that arises hand-in-hand with collecting large amounts of data within an organization, is that of *privacy* and *confidentiality*. When we say 'privacy', we generally mean the protection of individual privacy of employees of the organization, whereas 'confidentiality' in a broader sense, refers to the protection of the company's business secrets which may be jeopardized by revealing data to third-parties. Ensuring both of these is not only a legal requirement in most cases, but is also motivated by business interests. Therefore, although applying privacy-enhancing technologies to a dataset would result in a measurable loss in data quality, this is important and desirable in many use-cases, most importantly in cases where sensitive data needs to be made available to other organizations, or even to other departments.

Privacy is now widely recognized as a fundamental human right. Article 8 of the European Union's charter of fundamental rights[3] specifies the following:

> *Everyone has the right to the protection of personal data concerning him or her. Such data must be processed fairly for specified purposes and on the basis of the consent of the person concerned or some other legitimate basis laid down by law. Everyone has the right of access to data which has been collected concerning him or her, and the right to have it rectified.*

Based on the recognition of this fundamental right, it is good corporate practice to self-regulate the data being collected within the organization, even if it does not fall within the purview of these regulations. This ensures fairness and good faith. However, as research has shown time and again [SRP13; VZ19; WBI17], there is a very clear trade-off between providing privacy-guarantees and ensuring that the data still remains relevant for generating new knowledge and insights. This is shown in Figure 1.5. Therefore, naive privacy-enhancing techniques which simply involve stripping key attributes and quasi-identifiers altogether are irrelevant for the industry, as they lead to a complete loss of data utility. Indeed, providing this fine balance between data utility and privacy for data collection and publishing is a complex, non-trivial task.

Privacy-preserving data mining (PPDM) techniques, which allow for knowledge discovery from large datasets without disclosing sensitive information, have been developed by researchers. Most PPDP algorithms rely on modifying key attributes or quasi-identifiers to preserve privacy, and the degradation in data quality resulting from this

[3] http://fra.europa.eu/en/charterpedia/article/8-protection-personal-data

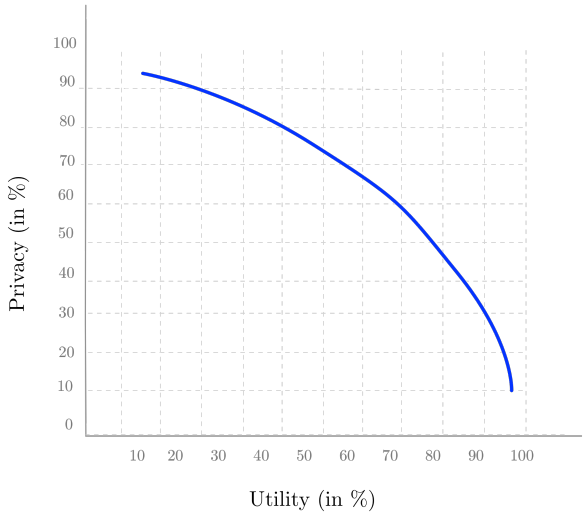

Figure 1.5: Trade-off Between Privacy and Data Utility. Adapted from [SRP13]

sanitization process is seen as a natural trade-off between data privacy and utility[MV17]. Choosing PPDM techniques has to be done in an extremely careful manner too. As Brickell and Shmatikov note in [BS08], even modest privacy gains require almost complete destruction of data-mining quality.

Therefore, there is a pressing need for a granular step-by-step privacy-enhancing infrastructure within the framework the data collected by an organization. This is particularly important, because a *one-size-fits-all* style of privacy assurance is almost worthless for a large organization that collects several different kinds of data on an everyday basis. At the same time, this must correlate intuitively to the concept of data quality. This is because the effects of applying privacy algorithms should be measurable, so that they can be tailored for different use-cases and datasets. To summarize this, data privacy is required as a complementary measure alongside data quality measurement, because of the following reasons:

1. **Serving a Legal or Policy Requirement:** Many a times, it is simply a legal requirement that companies must not store certain qualifying privacy-intrusive attributes, especially when concerned with data regarding people. In other cases, it is standard industry practice or an intentional self-regulation to restrict the data that can/should be stored. Therefore, privacy algorithms need to be tailored to the interests of the organizations, and the kinds of data collected

2. **Data Disclosure for Non-business Purposes:** For instances that require data sharing for scenarios where critical business decisions are not involved, for ex-

ample for hackathons or competitions, it is absolutely necessary that no unintended company information or secrets are shared with the data consumers. An often cited example related to this use-case, is the data shared during the *Netflix Prize*[NS08], which sought to improve the accuracy of movie recommendations. Scientists Narayanan and Shamatikov were able to show that it is possible to identify individuals from an anonymized dataset, if records from one database can somehow be linked to another. An important feature of such data disclosure scenarios, is that there can be a considerably higher degree of allowed tolerance for perturbations in exact numerical values in the data, as the data mining task does not affect critical business decisions.

3. **Data Disclosure for Business Purposes:** These disclosure scenarios can be further divided into two different use cases: (a) Use cases requiring data sharing within the organization, or with the parent organization and (b) Use cases requiring data sharing with third-parties, e.g., equipment supplier companies. In both of these use-cases, the data mining tasks have higher stakes associated with them, as results of the mining task are likely to impact critical business decisions. In these situations, it is absolutely imperative to ensure that the utility of the data mining task is not diminished to an extent that false or improper results are inferred from the data.

1.5 Data Economy and the Data Marketplace

We are inevitably moving towards a world where *data* is becoming an important asset, of great importance in the global economy. Although the phrase *Data is the new oil* was famously coined by the British mathematician Clive Humby in 2006, data has not yet reached its pinnacle as a resource, and in most cases, effective exchange and monetization has not been possible. Some of the main reasons for this include the inability to establish trust and transparency between parties, and a lack of standardized protocols for the transfer and use of big data.

With the rise in the collection of IoT data, data sharing is gaining prominence once again [ZPG13]. Companies collect, report and analyse large amounts of data, made possible by the sterling improvements in big data technologies, especially in in-memory and distributed computing. This exponential rise has made data sharing of huge interest to big corporations [Oph+16]. This development is of huge relevance to the automotive sector, where large amounts of IoT data are collected in each stage of production.

Therefore, a platform where *Data-as-a-Service (DaaS)* or *Analytics-as-a-Service (AaaS)* are offered is essential. Figure 1.6 illustrates a high-level conceptual framework of an enterprise data marketplace.

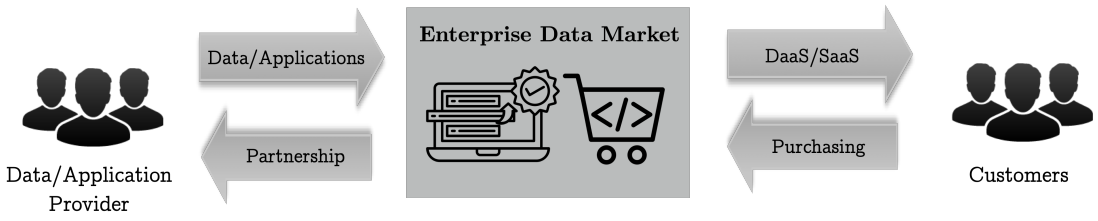

Figure 1.6: IoT Enterprise Data Marketplace Concept[a]

[a]Adapted from https://www.iais.fraunhofer.de/en

The Industrial Data Space (IDS), an initiative by the Fraunhofer Gesellschaft in Germany is a huge step forward in this direction. The IDS proposes an architecture for secure data exchange and trusted data sharing, and tries to bring together stakeholders from academia, industry, politics, and standards organizations [Ott+16]. However, crucial questions remain unanswered. There is still no well-recognized measure of data quality, resulting in an inability for the consumer to reliably determine what value the data brings for her business case. There are still no comprehensive studies on the relationship between data quality and privacy, and how applying privacy-enhancing technologies (PETs) affects the suitability of the data to be used for the business case.

Nevertheless, a plethora of concrete use-cases for data sharing in the automotive industry, such as the one illustrated in Figure 1.7, dictate that companies need to establish transparent quality metrics for data, as well as a suite of privacy-enhancing technologies, which can ensure that data can be confidently and securely shared for research and development purposes, and also as a medium for generating revenue.

1.6 Objectives and Contributions of this thesis

This thesis aims at tackling the complex problem of metadata management, data quality measurement and data suppression in a multi-faceted manner. We have proposed a flexible yet comprehensive approach that brings together metadata from data owners and use-case owners. This approach is designed for general-purpose data, i.e. it is not developed for a particular kind of dataset, and the design of the reference software application is done in a way which keeps it open for usage and deployment on a large variety of data sources.

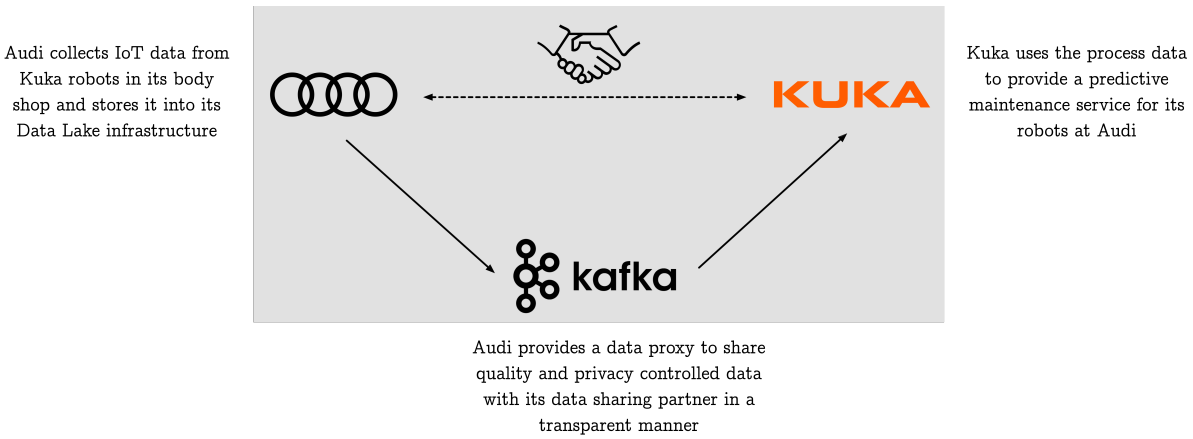

Figure 1.7: A Data Sharing Scenario in the Automotive Industry

The contributions of this thesis are:

- A well-researched *selection of the most important dimensions* for data quality and privacy measurement. *(Sections 3.4 and 3.5)*
- An approach for *bringing together crucial metadata* in a centralized metadata repository, which is available for use to various stakeholders. *(Section 3.6)*
- An approach for *measuring the fitness, usability and privacy* of the dataset by using the definitions of the various dimensions within each category. *(Sections 4.3 and 4.4)*
- An approach for *applying privacy-enhancing technologies* by leveraging recent advances such as distributed cluster-computing for big data and frameworks such as Apache Spark. *(Section 4.4.2)*
- An approach for *measuring the strictness* of pre-packaged privacy levels by studying its effects on the usability of the dataset. *(Section 4.4.4)*
- A detailed study into the *effects of applying PETs on the overall data quality and privacy* metrics. *(Section 5.2.2)*
- Development of a pilot project to *apply and test the approach* proposed in this thesis on two real datasets from the domain of production big data at AUDI AG. *(Sections 5.2.1 and 5.2.2)*

1.7 Structure of this Thesis

This thesis is divided into *six* chapters. The content of each of the upcoming chapters is as follows:

- **Chapter 2** describes the *Related Work*, and focuses on state-of-the-art contributions in *Data Quality* and *Privacy*, and the research works which have had the most impact on our approach.
- **Chapter 3** provides a bird's eye view of our methodology, and describes its fundamental concepts, e.g., the data quality and privacy dimensions.
- **Chapter 4** provides extensive details about the implementation of the quality and privacy modules, and the technologies we've used for the software application.
- **Chapter 5** illustrates the results of the data quality and privacy survey that we conducted as a part of the thesis. Furthermore, it also evaluates the results of our approach on two datasets from automotive production.
- **Chapter 6** sums up this thesis by listing our main contributions, and discusses the future of data quality and privacy, especially in upcoming use-cases such as *data sharing*.

2 Related Work

Both topics addressed in this thesis, i.e., *data quality measurement* and *privacy-preserving data mining* have received considerable attention from the research community over the past few years, owing to the growing importance of data for decision-making and the use of data-driven insights in several scientific fields. The emergence of big data technologies has led to a renewed interest in these fields. In the following sections, we will discuss methodologies which have been proposed for data quality measurement, as well as for privacy-preserving data mining.

2.1 Approaches for Data Quality Management

Strategies for measuring and improving the quality of data started primarily in the 1990s, when data quality started being recognized as a relevant performance issue of operating processes [Eck02]. Richard Y. Wang of the MIT was one of the first researchers in this field, who also put forward the most widely accepted definition of data quality as '*fitness for use*' [WS96]. Wang et al. also argue that data quality cannot be measured using a single attribute, but rather a multitude of parameters which vary depending on the perception of the data consumer. Based on this assertion, they define the notion of '*data quality dimension*' as a set of data quality attributes that represent a single aspect or construct of data quality. Moreover, Wang[Wan98] proposes the Total Data Quality Management (TDQM) methodology, in which he suggests a general 4-step process to support end-to-end quality improvement, from requirements definition until implementation. This is illustrated in Figure 2.1.

Post these initial developments, research in data quality became extremely important, especially as the amount of data started exploding. Several approaches have since then, been suggested to measure and improve data quality, which have been summarized in Table 2.1.

In the following sections, we will discuss the approaches which are most relevant for our own implementation, mention their individual merits, and reason why a new data quality measurement methodology is required for various domains that use big data.

Methodology Shorthand	Full Name	Authors
TDQM	Total Data Quality Management	Wang [Wan98]
TIQM	Total Information Quality Management	English [Eng99]
AIMQ	A Methodology for Information Quality Assessment	Lee et al. [Lee+02]
DQA	Data Quality Assessment	Pipino et al. [PLW02]
IQM	Information Quality Measurement	Eppler & Muenzenmaier [EM02]
DaQuinCIS	Data Quality in Cooperative Information Systems	Scannapieco et al. [Sca+04]
QAFD	Methodology for the Quality Assessment of Financial Data	De Amicis & Batini [Ami04]
CDQ	Comprehensive Methodology for Data Quality Management	Batini & Scannapieco [BS16]
ODQF	Ontology-based Data Quality Framework	Geisler [Gei17; GWQ11]
DMDL	Data Quality Management for Data Lake Systems	Dalevskaya [Dal17]

Table 2.1: Approaches for Data Quality Management.

2.1.1 Total Data Quality Management (TDQM)

As mentioned earlier, TDQM [Wan98] is important because it was the first general methodology for quality measurement, and helped to lay the foundations on which many other approaches are based. Wang et al. argue that a comprehensive list of quality dimensions and the perspectives of the data consumers can not be dissociated from the data quality analysis procedure. They were also the first ones to propose the 4-step process consisting of *Definition*, *Measurement*, *Analysis* and *Improvement* phases in a cycle, as illustrated in Figure 2.1.

The main objectives of each phase are as follows:

- **Definition:** The data is analysed at a *high-level*, i.e., describing the functionalities for consumers, and at a *low-level*, i.e., mapping the basic units of the data and its relationships. A DQ Requirements analysis is performed taking into consideration the perspectives of the stakeholders. Finally, the DQ dimensions are chosen and a process modelling is performed.

- **Measurement:** Metadata regarding the DQ dimensions is collected from the data, and this is verified against the requirements.

- **Analysis:** Based on the problems identified in the previous step, root causes of the discrepancies are traced and steps for improving data quality are suggested.

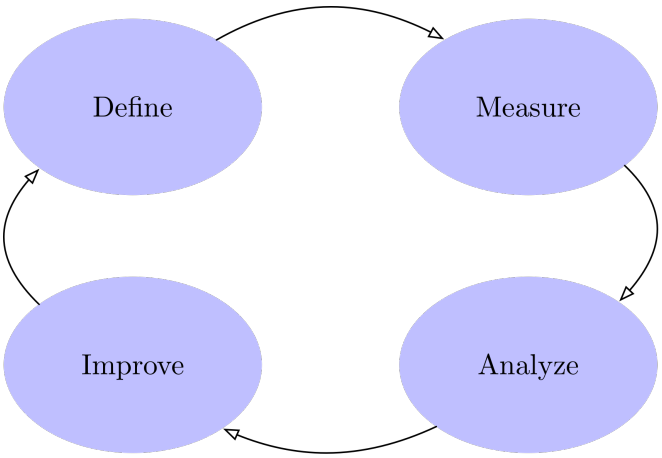

Figure 2.1: 4-step General Process Proposed in the TDQM

- **Improvement:** Based on the DQ metrics, key areas for improvement, as well as the strategy to be undertaken, are identified

In spite of its significance, in recent literature, no industry-specific techniques are referred and no guidelines for specialization of the general quality improvement techniques are offered.

2.1.2 Data Quality Assessment (DQA)

The DQA methodology [PLW02] provides the general guiding principles for the definition of data quality metrics [Bat+09]. Previously, the dimensions to measure data quality were defined on case-by-case, ad-hoc basis, and the authors tried to organize the common measurement principles in the previous research. This was the first approach to partition the metrics as *objective* and *subjective* metrics. Subjective metrics measure perceptions, needs and experiences of the stakeholders. Objective metrics are further divided into task-independent metrics, i.e., without any contextual knowledge of the application, and task-dependent metrics, i.e., including knowledge specific to the application such as business rules, company regulations etc. [Bat+09].

2.1.3 Comprehensive Methodology for Data Quality Management (CDQ)

The CDQ methodology [BS16] aims to provide a complete, flexible and simple approach by combining existing tools and techniques and integrating them into a common framework that can be applied in various contexts in on various kinds of data. The main goal of CDQ is to obtain a quantitative assessment of the extent to which business processes

are affected by bad information [Bat+09].

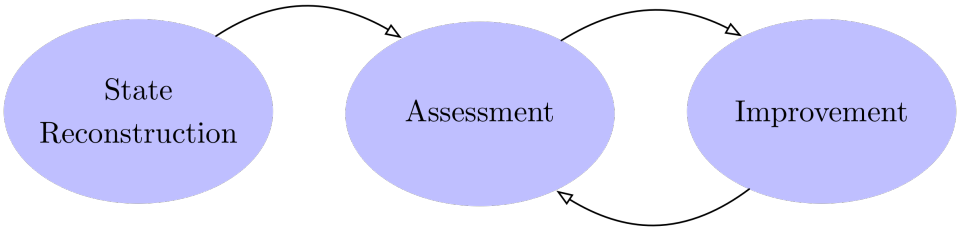

Figure 2.2: Phases of the CDQ Methodology

CDQ follows a three phase model, as illustrated in Figure 2.2. These phases are:

- **State Reconstruction:** The relationships between organizational units, processes, services and data are constructed and modelled. In this phase, the entire process frameworks along with their roles in the final production of goods and services, as well as the legal and organizational rules are described.
- **Assessment:** In the assessment phase, the target data quality standards are determined and the corresponding costs and benefits of the improvements are estimated. The critical variables, which are most severely affected by quality issues, are identified and focussed on.
- **Improvement:** The improvement phase is a 5-step process which identifies the most optimal improvement process, i.e. the sequence of activities with the best cost-to-effectiveness ratio. The methodology recommends that both *data-driven* and *process-driven* improvement techniques should be considered, and a set of 'mutually-consistent' improvement steps should be chosen to form an *improvement process*.

2.1.4 Data Quality Management for Data Lake Systems

In [Dal17], Dalevskaya proposes a data quality management approach for the *Constance* data lake [HGQ16] developed at RWTH Aachen. The aim of this approach to have a continuous and extensible data quality monitoring approach within the data lake, whereby a time-based quality analysis can be visualized within it. This approach benefits from the comprehensive metadata management technologies employed within Constance, such as *schema mapping and discovery*.

The approach is divided into 3 main steps: *DQ Definition*, *DQ Evaluation* and *DQ Monitoring*. In the definition stage, the quality requirements are identified and the

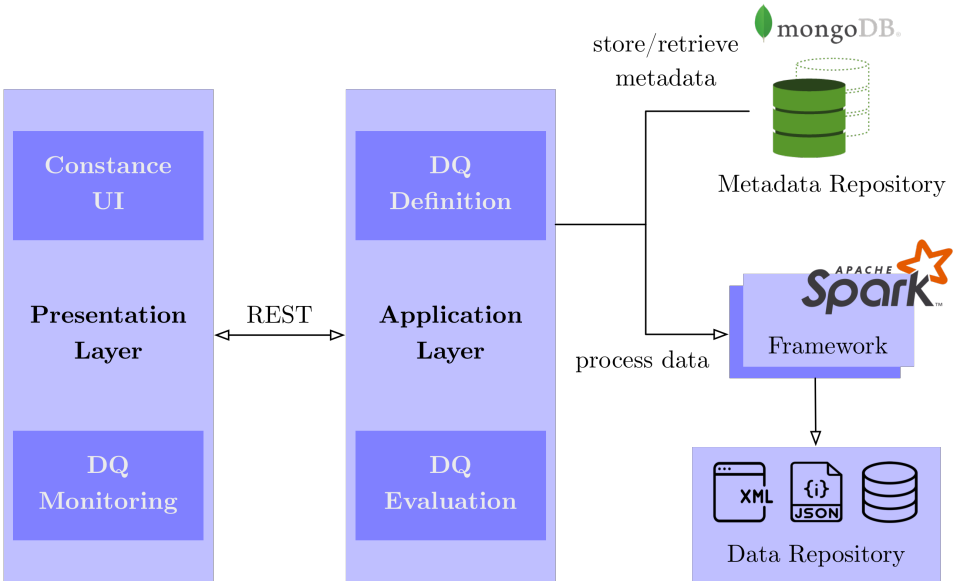

Figure 2.3: Architecture of Data Quality Management for Constance

dimensions for the quality assessment, as well as the formulas for DQ calculation are determined. The author uses completeness, uniqueness and validity as the default measures, but also specifies that the dimensions could be extended. The calculations take place in the evaluation step, whereas in the monitoring step, the quality measurements are visualized with the help of charts within the Constance UI.

2.1.5 Discussion

In spite of the availability of good data quality measurement techniques, we did not find a good match for the needs and use-cases of the automotive industry. The main issue is that most of these methods do not scale well enough with the size of the data. Some of the methods require continuous explicit feedback mechanisms from people through meetings, questionnaires etc., which becomes extremely tedious and unmanageable once the data volume grows. Others are either very strict about their quality dimensions, or do not make a clear-cut distinction between objective and subjective quality metrics, which is of fundamental importance to us. Therefore, we felt the need for a fresh approach, which is not only suited for large volumes of data, but one that is also able to handle subjective and objective DQ dimensions in a conjunctive as well as disjunctive manner, and in conformance with our business needs and objectives.

2.2 Approaches for Privacy-Preserving Data Publishing

As discussed in Section 1.4, Privacy-Preserving Data Mining (PPDM) approaches have allowed us to generate insights from data without revealing sensitive private information about individuals or organizations. These privacy-enhancing technologies (PETs) can be employed in various stages of the data life-cycle, namely at *data collection*, *data publishing*, *data distribution* or *during the output of data mining*. For most use-cases in the automotive industry, we deal with the second use-case. This is because data collection is performed by secure entities and is stored securely within the organizational cloud. Therefore, approaches such as adding *additive noise*[AS00] or *multiplicative noise*[KW03] become irrelevant for us. At the same time, we avoid collective distributed computation on company data, as it is extremely resource-intensive, too expensive for large amounts of data, and scenarios that really require shared computations such as *oblivious transfer*[EGL85; NP01] or *secure multi-party computations*[Cli+02] are rare. Applying PETs to hide the output of data mining is also a scenario that we seldom encounter, as most results of data mining processes are highly confidential and are not shared publicly. Therefore, this thesis focuses on the application of PETs during the *data publishing stage*. Data publishing is a highly-valuable endeavour for us, as it enables us to benefit from the expertise of our equipment manufacturers, who very often have a more comprehensive knowledge about their machine, its properties and behaviour. As a result, it becomes possible for us to perform operations such as predictive maintenance in a collaborative, efficient and well-informed manner.

Privacy is essential in several data publishing scenarios, because entities wish to release their data for research or other use-cases, without disclosing sensitive parameters or business secrets. This is generally done by anonymizing the records before publishing them. However, as mentioned in Section 1.4, naive anonymization techniques such as simply stripping off the key identifiers is inadequate, as individuals can nevertheless be identified based on their quasi-identifiers through linkage attacks [Wan+10]. Most anonymization approaches adopt a certain *privacy model* or a mixture of models to apply *sanitizing* operations on the raw dataset. These measures are as follows:

1. **Generalization:** In generalization, attribute values are replaced by a more general one (e.g. their parents in a tree-based classification). Numerical values can be replaced by an interval, whereas categorical attributes need the definition of a hierarchy. An example from our production scenario is that the processes *welding* and *screwing* can be generalized to the attribute *body shop process*.

2. **Suppression:** Suppression refers to the full or partial removal of an attribute value to hinder its complete disclosure. Full suppression leads to a removal of the sensitive column from the entire dataset, whereas partial suppression adds wild-card characters to the attribute value. E.g. the temperature value of $208K$ can be suppressed to $2 * 8K$.

3. **Anatomization:** The concept of anatomization is to separate quasi-identifiers and the sensitive attributes into separate tables [XT06a], i.e. dissociating them so that re-identification becomes difficult.

4. **Perturbation:** In perturbation, the actual attribute values are changed to different value. This can be done in different ways. In *data swapping*, the sensitive attributes are simply swapped in order to prevent linkage attacks. This might not change some statistical indicators, but it causes a huge barrier in training learning models, and makes the dataset almost useless for time-series modelling. On the other hand, *synthetic data generation* aims to imitate the real data by obtaining values from a statistical model of the real data.

These basic privacy models have enabled several popular privacy-enabling mechanisms. The most widely used among these is *k-anonymity*, proposed by Samarati and Sweeney [SS98]. According to the concept of k-anonymity, a dataset is said to be k-anonymous if the identifiable attributes of any database record are undistinguishable from at least other $k - 1$ records. In other words, k-anonymity, using generalization and suppression, aims to create a set of equivalence classes which hinders precise re-identification of records. It became extremely popular because of its easy understandability and measurable privacy levels, i.e., higher the value of k, higher the level of privacy. However, since k-anonymity does not change any sensitive attributes, it can not protect against background knowledge attacks. To overcome the limitations of k-anonymity, further extensions such as *l-diversity* [Mac+06] and *t-closeness* [LLV07] were proposed. Another fundamentally different concept of *personalized privacy* was presented by Xiao and Tao [XT06b]. The authors suggest that to achieve personalized privacy, a taxonomy tree should be created for generalization, and a *guarding node* should be set to define the level of disclosure. In the ϵ-differential privacy model [Dwo08], Dwork presents the concept of *differential privacy*, which hinges itself on the differentiation between the presence or absence of an individual's record in a statistical database. It is assumed that such privacy guarantees will motivate people to participate in surveys without ever revealing their personal identity or choices.

2.2.1 Discussion

Growing concerns over misuse of personal data, and the availability of tools and techniques to process large amounts of data, have led to a reinvigoration of privacy research in recent times. This has led to the development and advancement of many techniques such as *Homomorphic Encryption (HE)*, *Differential Privacy*, *Secure Multi-Party Computation (SMPC)* and *Verifiable Computation (VC)*. Some of these techniques, e.g. homomorphic encryption, are still in a nascent stage and have not been adopted widely in the industry due to their high computational demand and high complexity. Others such as SMPC and Differential Privacy have found niche uses, such as the use of 'local differential privacy' by Apple for collecting statistical user information from Safari[1].

For large amounts of industrial data however, more traditional techniques such as data minimisation, encryption, anonymization, and pseudonymization remain the most popular choices[2]. A combined study of data quality and privacy-enhancing technologies presents an interesting opportunity, as it gives us a perspective about the measurable effects of privacy on data quality. More effort towards privacy algorithms that guarantee high data usability for analytics will lead to a much more widespread adoption of PETs in the industry.

In the upcoming sections, we will describe the methodology and process which we propose as a part of our solution. We have tried to address some of the issues with the current approaches, and provide a flexible, extensible and robust framework for measuring and adjusting data quality and privacy which is well-suited for data with a high volume and variety.

[1] https://www.apple.com/privacy/docs/Differential_Privacy_Overview.pdf
[2] https://www.enisa.europa.eu/topics/data-protection/privacy-enhancing-technologies

3 Solution

In Section 1.3, we have argued about the importance of data quality measurement in data lakes, and how it applies specifically to various use-cases in the automotive industry. In Section 1.4, we have posited that privacy-enhancing technologies provide a convenient middle-ground to ensure the protection of business secrets and as well as to preserve the utility of the data for analytics purposes. On the basis of these arguments, this thesis proposes a unique solution, which we will discuss in this chapter. The data quality and privacy approach introduced in this thesis is named *Deus*, which means 'god' in Latin. Henceforth, *Deus* will refer to the data quality approach that we introduce and elucidate in the following sections, as well as the reference software application accompanying it.

3.1 Design Considerations

In this section, we discuss the design considerations that *Deus* had to be mindful of. These choices make sure that it is possible to use it not only for a wide-ranging variety of data across the organization, but also in other organizations and domains that face the same problems and challenges.

In spite of a widespread recognition of the importance of data quality, current approaches proposed in literature have not been overwhelmingly adopted by the industry. There are several reasons for the same. The main reason is the 'subjectivity' of data quality. It is very difficult to agree on a common definition for *good quality data*, and this is what has led to organizations abandoning attempts to employ approaches proposed in literature for their own use. Another important consideration is that officials as well as other employees within an organization feel extremely uncomfortable with black-box data quality approaches. Therefore, data quality approaches need to be as *fine-grained* and *transparent* as possible.

Extensibility is another criterion that shackles the scope of the current data quality and privacy approaches. More often than not, data quality approaches provide a fixed set of dimensions against which the measure is estimated. However, in a dynamic environment where companies are themselves learning perpetually about the data they

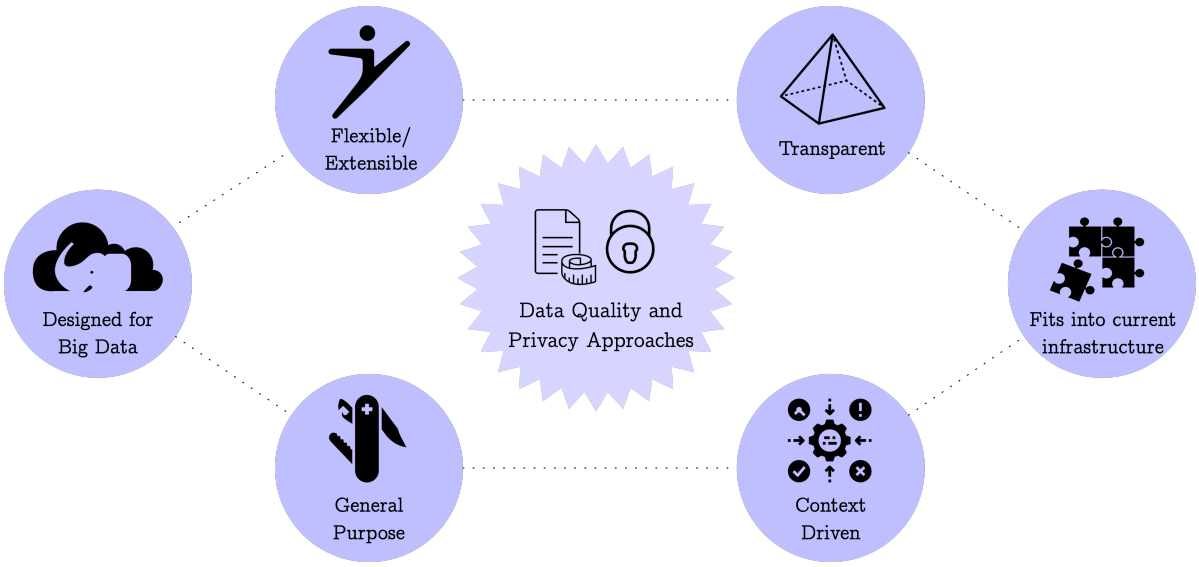

Figure 3.1: Desirable Qualities of a Data Quality and Privacy Approach

collect and the value it brings to the organization, the definition of data quality is bound to change perennially. Therefore, data quality approaches of today need to be *flexible* and *extensible*.

A data quality approach within a large organization also has to be *uniform* and *general-purpose*. Automotive companies collect data for wide-ranging use-cases and at various stages of the supply-chain, as illustrated in Figure 1.3, and therefore, the approach must be able to reliably compute the quality metrics for different types of data, and not be domain-specific. But at the same time, it should be *context-driven*, i.e., the quality metrics should not be independent of the way the data will be put to use. After several face-to-face interviews with officials in the *production, production planning* and *IT* departments at Audi, one of the conclusions was that data quality and privacy metrics need to be *metadata-driven*, as well as *dependent on the use-case* they will be employed for. This serves a dual purpose. It is directly beneficial for the relevance of the quality metrics, as it ensures that contextual information is not lost in an otherwise general-purpose approach. But at the same time, it provides a well-defined method for collecting and storing metadata within the organization. This acts as a first step towards the final goal of an organization-wide data catalogue, which would make it much easier to query and request metadata for the design of analytics use-cases within the organization. This will be discussed in more detail in Section 3.6.

Modern, future-ready approaches also need to be *compliant with big data technologies*. An academically superior data quality or privacy approach would not be effective for

real-world deployment if it wasn't usable for the sheer volume of data that organizations collect. This is also a major drawback of current approaches, many of whom require significant human intervention or fine-tuning. Last but not least is the design consideration that a new data quality and privacy approach must fit-in with the existing and possibly also the future data flow and infrastructure of the organization. These desirable qualities, which are also summarized in Figure 3.1, make sure that a data quality and privacy approach remains usable and relevant for a long time.

3.2 Current Practices for End-to-End Data Flow

Figure 3.2 illustrates a typical end-to-end data flow of process data at the AUDI AG. *Robots, Sensors, Drivers, and PLC Systems* generate data which is transmitted via communication protocols *(such as OPC-UA, MQTT)* and centralized in an *aggregating server/middleware*. *Middleware*, which typically uses a *RabbitMQ or Kafka* broker, applies required transformations to the incoming messages, and routes it to its destined *topic* in the data proxy. *Data Proxy* acts as a short-term persistent data store, from where data can be ingested into permanent storage, i.e., Data Lake P. *Data Lake P* contains historical process data stored in the *avro* format in the HDFS (Hadoop Distributed File System). It supports applications on top of it such as *Hive* or *Spark*, which can be used to query this data using SQL, or perform complex machine learning tasks respectively. All data analytics use-cases use the data stored in HDFS as source. *Data Sharing Broker* is the intermediate between the *Data Proxy* and the *external partners* with whom data needs to be shared.

There are several limitations of using this data flow architecture. The most significant factor is that *Data Lake P* becomes the target destination for a significant chunk of process data originating from different source systems. This makes it a huge cesspool of raw data without any major metadata association, apart from some basic schema information contained in the *avro* files. It is a nightmare for managers and data scientists, with a limited idea of the trustworthiness and credibility of the data, to formulate use-cases around it and employ it for productive decision-making tasks. Such a scenario would benefit hugely from additional metadata from a metadata repository, which contains information about the data quality and other essential cataloguing information about the dataset.

Moreover, this kind of architecture is also severely handicapped for data sharing. Providing third-party organizations/individuals access to potentially sensitive information without determining its privacy characteristics or applying any suppression, is a recipe

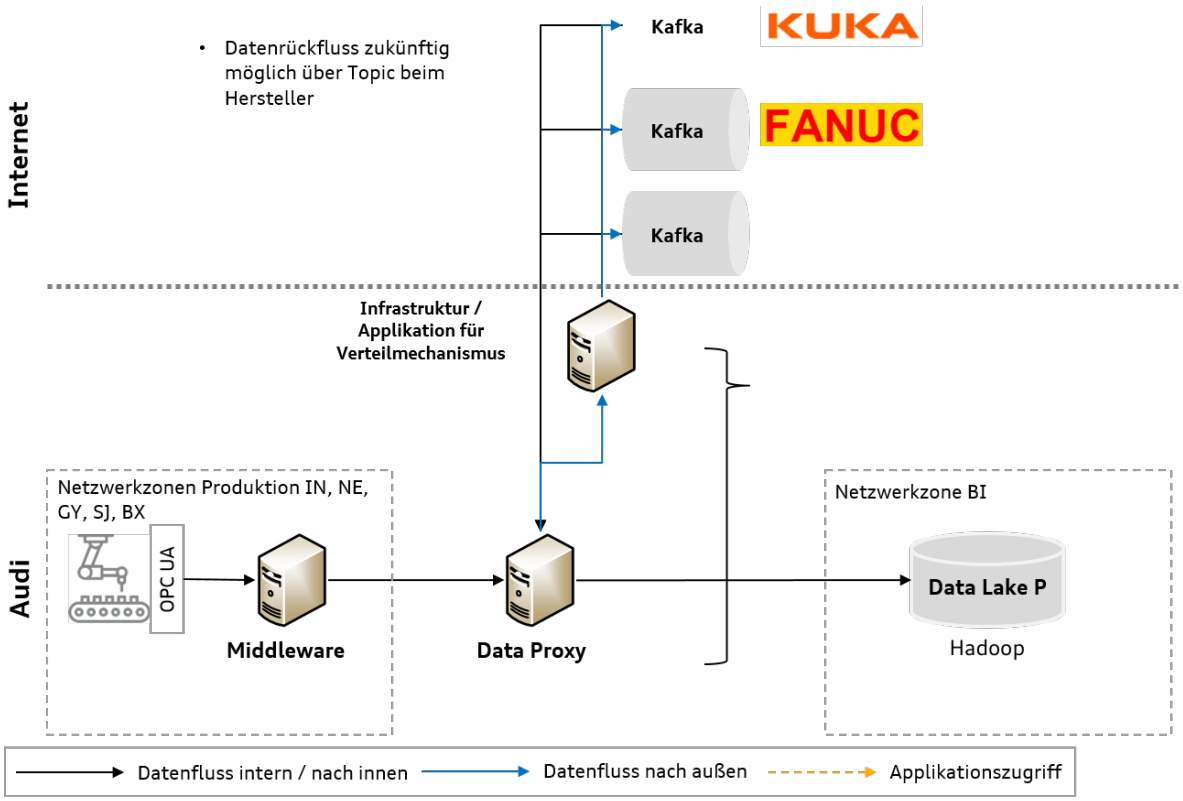

Figure 3.2: Data Flow within Audi's Data Lake P Project. *(Source: AUDI AG)*

for failure. More often than not, this situation discourages data sharing, and blocks any kind of data-based transactions leading to mutual business benefits. A transparent method for evaluating and managing data privacy, establishing an environment of trust and amity for data sharing, would be a major enabler in this case. As we will see in the upcoming sections and chapters, our proposed solution tries to address the deficiencies of this architecture by augmenting it with an alternative approach.

3.3 Solution at a Glance

Deus is an effort towards a comprehensive solution for data quality evaluation suited to fit within the data strategy of companies that leverage big data technologies. Although the approach itself is designed to be general-purpose, we evaluate it with real-world datasets from two different stages of the production process. Production IoT data, especially given its volume and veracity, presents a daunting challenge in terms of data quality. Our aim is to empirically quantify the amount of value contained in the data, and to exhibit how the insights generated by the results can be useful in various stages

of the data mining process.

Furthermore, we use privacy-enhancing technologies (PETs) packaged into privacy levels to suppress private data, in turn adding additional progressive layers of privacy guarantees. By combining data quality measures and privacy guarantees, we not only want to study and evaluate how the fitness and usability of datasets change with progressive levels of privacy, but also to empower the user of this solution to choose a suitable space within the privacy-utility trade-off for data sharing scenarios. This provides a real alternative for data monetization as well as in gaining mutual benefit out of the data, rather than the current stand-off, whereby most of the data is inaccessible to prospective data users because of privacy concerns.

Figure 3.3 illustrates a high-level architecture of the proposed solution. For architectural and implementation purposes, the thesis is segregated into four modules, each having its own pre-determined function. These modules are:

1. **Metadata Extraction and Management:** In Section 3.6, we have argued why metadata management is more relevant today than ever. For quality assessment, it is even more essential. The role of this module is to evaluate the dataset to compute the fundamental data quality and privacy dimensions, which are described in Sections 3.4 and 3.5. This metadata is enriched and enabled by using the pre-specified knowledge about the dataset and its individual attributes, and knowledge about the usability of the data for a specific use-case. Within the scope of this thesis, these files are henceforth called as *knowledge files*. The two knowledge files are called *general knowledge* and *use-case knowledge* respectively.

2. **Data Quality and Privacy Assessment:** This module is responsible for computing the data quality and privacy metrics based on the metadata and knowledge collected in the previous step. It includes *weighting*, *normalization* and *scoring* based on a user-defined formula. Based on our definition of data quality being measured as 'fitness to use', we provide individual metrics for *fitness*, *usability* and *privacy* in addition to a cumulative quality score. All of these aforementioned metrics are normalized and lie within the range of 0-1.

3. **Privacy-Enhancing Microservices:** In this module, privacy-enhancing technologies such as *attribute suppression, k-anonymity* and *tokenization*, which are packaged into privacy levels, are applied to the dataset to provide the requisite amount of privacy guarantee desired by the use-case. Data suppression leads to an inevitable decline in data quality, which is desirable in order to gain a granular

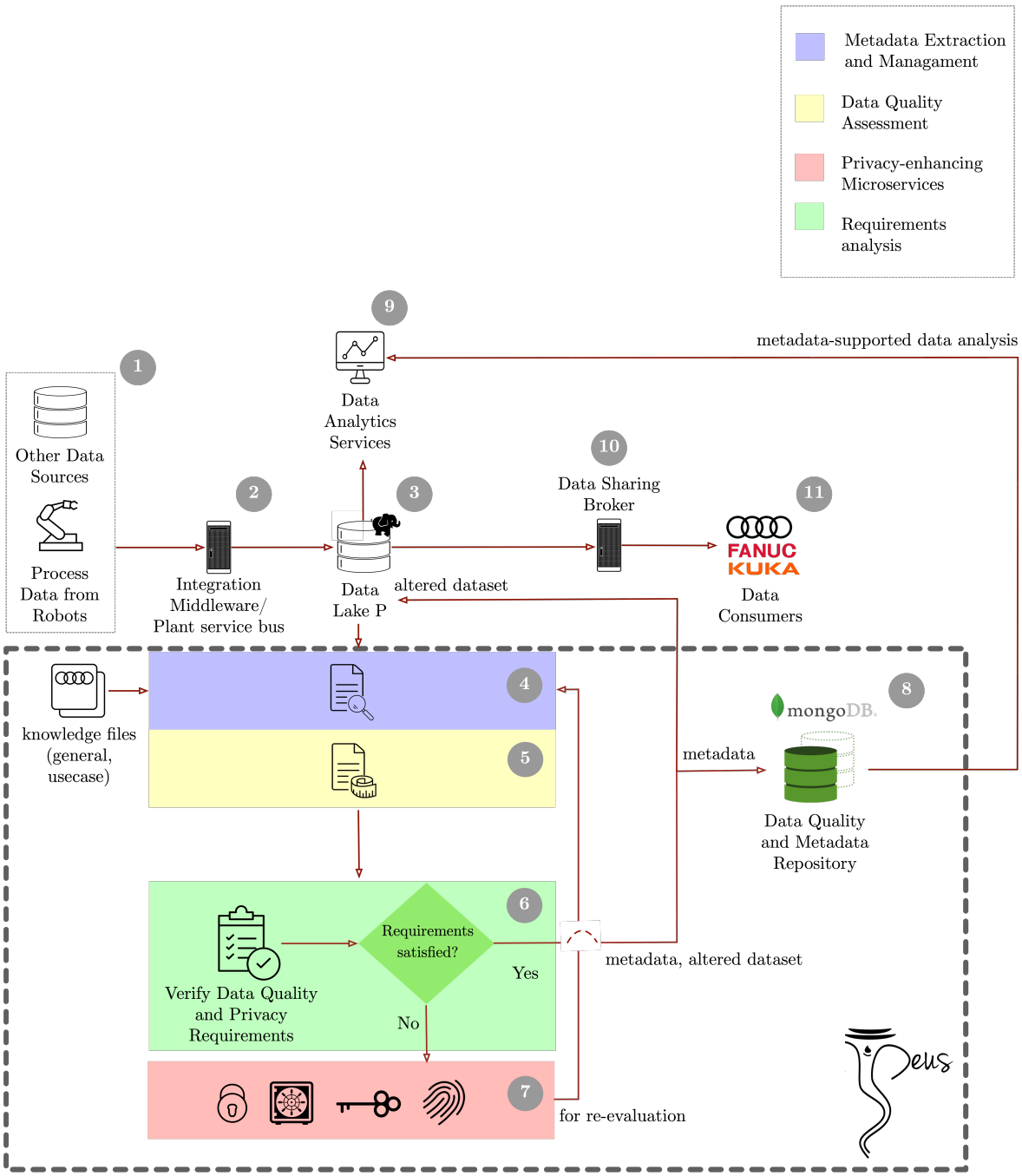

Figure 3.3: High-level Architecture of the Proposed Solution

control over it for use-cases, such as data sharing. The packaging of these PETs into levels is done to ensure that instead of an *all-or-nothing* based approach, i.e., privacy or no privacy, we are able to decide minutely on level of privacy and the corresponding degradation in utility that we would like to have.

4. **Requirements Analysis:** This is a specialized module whose job is to determine whether the current data quality and privacy levels suffice the quality and privacy requirements required for the use-case. According to the *priority* set for a particular use-case, requirements analysis leads to the decision whether PETs need to be employed on the dataset, and if so, the exact privacy level that needs to be employed.

In the upcoming sections, we will discuss the data quality and privacy dimensions that we have used in this thesis, as well as the scientific process of choosing them. We also provide their definitions within the scope of our implementation.

3.4 Data Quality Dimensions

The definition of data quality dimensions is a critical activity in data quality computation. The exact meaning of these metrics are specific to the scenario and the organizational use-case. There are ample classifications of data quality dimensions in literature, however there are several discrepancies in their definition, owing to the contextual nature of data quality [Bat+09]. We provide a set of reference definitions for our dimensions for data quality and privacy measurement. We have taken inspiration from the research work of Wang and Strong [WS96], Jarke et al.[Jar+13],Redman [RB97] and Catarci and Scannapieco [SC02], where several data quality dimensions have been meticulously recorded and systematically studied. These dimensions and definitions are also carefully designed to suit the needs of the automotive industry, and in accordance with the priorities and requirements for the data collected within the organization.

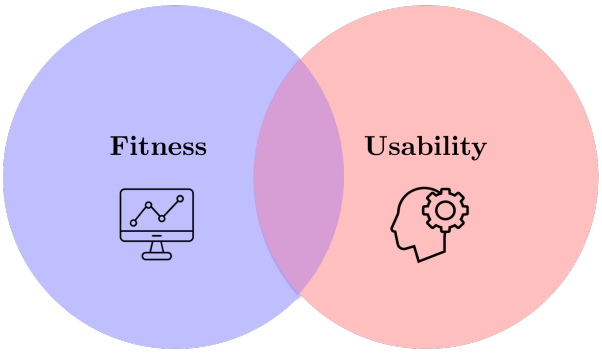

Figure 3.4: Quality Characteristics must include Dimensions from both Fitness and Usability Spheres

For purposes of unambiguity, we define data quality as 'Fitness for Use'. The emphasis is on two words: *'fitness'* and *'usability'*, which have entirely different connotations with

reference to data quality. This is also illustrated in Figure 3.4. While fitness can be objectively measured by estimating, e.g., the completeness or timeliness of individual attributes in the dataset, usability often has a very subjective connotation, which may differ from person to person. In addition, usability may also differ between different use-cases within the same organization, as different departments use data in a dissimilar fashion. Therefore, it is very difficult to objectively gauge usability. Therefore, we have designed our approach in a way that the fitness characteristic of a particular dataset always remains constant, however the usability metrics can change from use-case to use-case. The final cumulative score is a weighted sum of the fitness and usability measures, according to a weighing criteria that can be adjusted by the user.

The data quality dimensions which will be used corresponding to fitness and usability are listed separately in Tables 3.1 and 3.2. As mentioned before, these dimensions as well as their definitions were selected by taking guidance from the research studies on the topic, as well as keeping in mind the current requirements of the automotive industry. However, one of the strong points of *Deus*, whereby new dimensions for data quality and privacy can be added, or the current dimensions be modified or removed, in order to reflect the requirements for data quality and governance within the organization.

However, an important pre-requisite for useful comparison of data quality and privacy scores is that they should be evaluated against uniform criteria. Comparing scores for two or more datasets using disparate dimensions or definitions would provide an incorrect assessment.

3.5 Data Privacy Dimensions

Apart from data quality dimensions, we also engineered dimensions corresponding to the privacy characteristics of the dataset, to quantify the privacy level of the dataset. These dimensions are mentioned in Table 3.3. The privacy metrics are computed in addition to the data quality metrics, to get a better idea about how privacy impacts data quality, and to find a suitable balance between data quality and privacy for data sharing scenarios.

We make a comprehensive effort to study and quantify the relationship between data quality and privacy. These two interlinked concepts are closely intertwined with each other, but this degree of influence was always difficult to estimate because of the challenges in adequately quantifying them. By developing a unified approach to quantify quality as well as privacy, *Deus* tries to solve this problem.

Data Quality Dimension	Weight	Metric Definition
Accuracy	between 0 and 1	accuracy of the measuring device at the source or probability of defects in measurement
Completeness	between 0 and 1	number of values that are not null/total number of values of that attribute
Consistency	between 0 and 1	measured by the consistency criterion, e.g., a range of allowed values. For attributes with no consistency criterion, a check of the data type. Number of consistent values/total number of values
Timeliness	between 0 and 1	latency between the time of data collection and reporting (arrival in the Data Lake P or middleware etc.)
Uniqueness	between 0 and 1	proportion of duplicates in the dataset
Volume	between 0 and 1	size of the dataset
Interpretability	between 0 and 1	number of non-suppressed measurable attributes with units/total number of measurable attributes
Credibility	between 0 and 1	number of elements with default values/total number of elements

Table 3.1: *Fitness* Dimensions for Data Quality Measurement

Data Quality Dimension	Weight	Metric Definition
Volatility	between 0 and 1	time length for which the data remains valid.
Reputation	between 0 and 1	what is the trustworthiness of the source that generated the data? (discrete values)
Relevance (also known as Utility)	between 0 and 1	proportion of the total attributes which will actually be used for the analytics task
Desirability	between 0 and 1	does the analytics task desire any extra attributes which are not yet present in the dataset? What proportion of the desirable attributes are not present?

Table 3.2: *Usability* Dimensions for Data Quality Measurement

3.6 Metadata Management: A Desirable Aftermath of Data Quality Evaluation

As introduced in Section 1.3, *Data Cataloguing* is one of the important capabilities enabled by calculation and monitoring of data quality. Data cataloguing is essential for automotive companies, as it engages a fair deal of oversight on the ever-increasing volume of data collected by the organization, and ensures that the right data is used for the right analytics task. At the same time, it is also an important step in the management

Privacy Dimension	Weight	Metric Definition
Non- Sensitivity	between 0 and 1	proportion of attributes which are not key attributes or quasi-identifiers
Distinguishability	between 0 and 1	proportion of distinct pairs of key attributes and quasi-identifiers
Non- Linkability	between 0 and 1	proportion of the key attributes or quasi-identifiers which are tokenized

Table 3.3: *Privacy* Dimensions for Data Privacy Measurement

of organizational metadata.

Cataloguing Attribute	Attribute Definition
Accessibility	A list of departments which have access to the information
Responsibility	Contact person who is responsible for maintaining and updating the metrics and knowledge files for the dataset
Historical Use	A list of use-cases for which the dataset been used in past
Date of Deletion	The date by which the data is required to be deleted

Table 3.4: *Cataloguing* Attributes for Metadata Enrichment

Since *Deus* already collects, uses and provides valuable metadata as a part of its *Metadata Extraction and Management* module (explained in detail in Section 4.2), managing and augmenting this information, and making it available for easy streamlining of data analytics processes within the organization is a crucial responsibility. To enable this, we add extra cataloguing dimensions to specifically aid towards the maintenance of a data catalogue. These attributes, which are specified in Table 3.4, do not contribute functionally to the data quality approach itself. But they perform the pivotal task of organizing and collecting the metadata in a single repository, for the perusal of data scientists, business analysts and data governance experts. It is expected that the addition of these parameters will lead to a more streamlined data governance process, which would ultimately help business departments in a significant manner.

3.7 Discussion

In this chapter, we have made ourselves familiar with a bird's-eye view of the approach that we use to solve the problem discussed in Section 1.3. We have also discussed the

dimensions that we will ultimately use, for the quality and privacy assessment, as well as the reasoning for incorporating cataloguing information as a part of this approach.

In the next chapter, we will dive deeper into the actual implementation of the aforementioned solution, and look into the concrete steps that are taken to compute each of the quality and privacy dimensions. We will also describe how privacy-enhancing technologies are applied to the dataset, and how the relationship between the privacy and usability is ultimately determined. Ultimately, we will dissect the details behind the technical implementation of our solution by providing an introduction to the technologies we chose to use, and the organization of the application software.

4 Implementation

In this chapter, we explain the approach we took for the implementation of the approach described in Chapter 3. We try to fill the complete picture about how the individual modules fit together for data quality and privacy assessment. We provide a detailed explanation for some of the most important parts of the whole machinery, e.g. the *knowledge files*. We also provide a brief introduction to the technologies we used for implementing this approach, the rationale behind using them, followed by a description of the modular organization of the application.

4.1 Implementation Process

Our implementation process follows the approach proposed by Wang et al. in TDQM [Wan98]. As mentioned in Section 3.4, our data quality measurement is based on calculating dimensions classified into the *fitness* and *usability* criteria. Additionally, we also provide a privacy score as well as additional metadata for data cataloguing, which is essential within the organization. For implementing our approach, we have adopted a four-step strategy as illustrated in Figure 4.1.

1. **Define:** In the *Define* phase, requirements analysis for the data quality and privacy assessment is performed. This is the most important and lengthy phase in terms of time as well as effort. The data quality and privacy dimensions are agreed upon in a collaborative manner with the business department. The aim of this solution is to be of the greatest utility to the data consumers, and therefore deliberation in the early phases is fruitful and worthwhile. In addition, the parameters for weighting and normalization of the various dimensions are decided upon, and the composition formula for the final score is determined.

2. **Measure:** In this phase, the actual measurements of the defined data quality and privacy dimensions are performed, by gathering information from the metadata extractor and the knowledge files. In addition, the cataloguing information is also compiled. Thereafter, the composition formula is used to calculate the quality

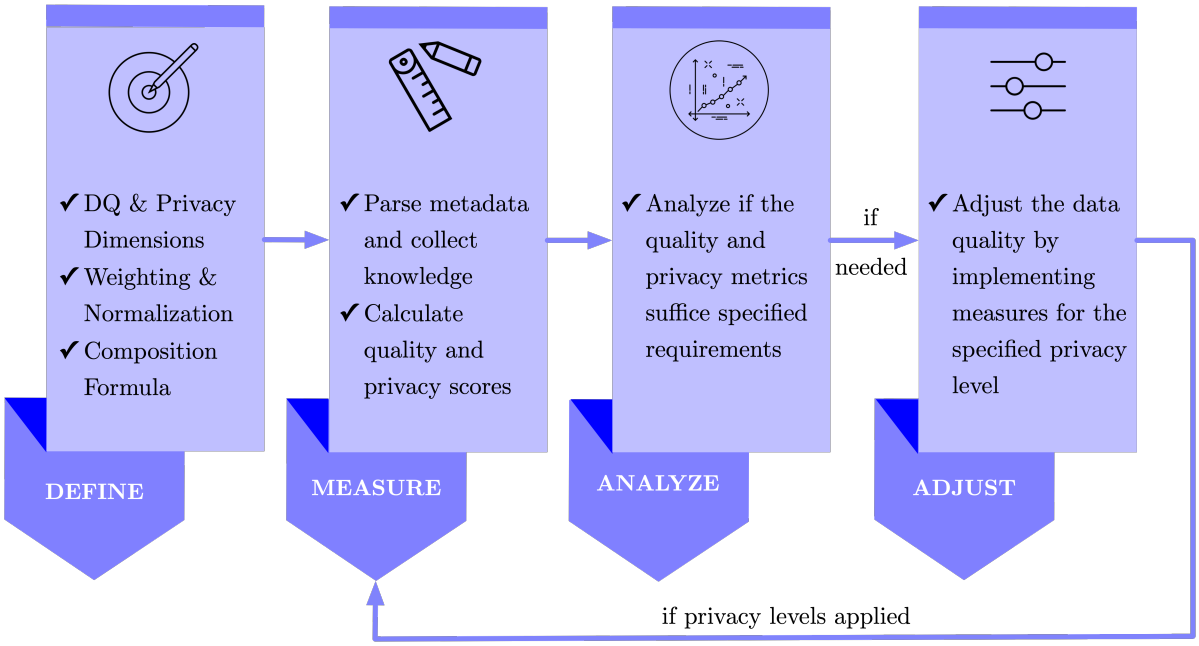

Figure 4.1: 4-step Plan for Implementation

and privacy scores. The data quality scores are of three different kinds: there are individual scores on the *fitness* and *usability* criteria, as well as a final combined score.

3. **Analyse:** In the *Analyse* phase, scores from the previous phase are evaluated against the requirements set for the use-case. This is performed using a specialized *requirements analysis module*, which collects requirements metadata from the *usecase* knowledge to determine whether the quality and privacy requirements are satisfied. If the requirements are found to be fulfilled, then the quality and privacy scores as well as the cataloguing metadata are stored inside the metadata repository. If not, then the data is passed on to the next step where quality and privacy adjustments are applied.

4. **Adjust:** In this phase, privacy-enhancing technologies are applied to the data in order to strengthen the privacy characteristics of the dataset. The privacy-enhancing microservices are bundled as privacy levels, and the respective privacy levels specified in the knowledge file is applied to the dataset. The level of privacy applied depends on the use-case requirements, but in general, the privacy levels are applied one-by-one from Level 1 onwards, until the quality and privacy requirements are satisfied. Thereafter, the modified dataset travels back to the *Measure*

phase for a subsequent round of quality and privacy measurement.

4.2 Knowledge Files

Knowledge Files form the backbone of our approach, and a very important factor which enables metadata-driven data analysis. As explained in Section 3.6, metadata management is an important step which ensures that data storage instruments such as data lakes retain contextual information and remain relevant for analytics use-cases. The rapid growth of digital data means that we are inundated with raw datasets which may or may not have any pre-defined use-cases when they are collected. Indeed, inexpensive storage combined with the ability to process large amounts of data in a short amount of time has meant that data collection often precedes thoughtful use-case creation.

Timestamp	**Temperature**	**Wear**
20170121T082010	32	41.5
20181218T164510	34	90
20180911T135638	40	78.9

Table 4.1: Hypothetical Dataset for Predictive Maintenance

However, this means that when datasets are scoured at a later stage to look for potential use-cases that may be derived from them, important metadata associated with it should be available to aid the creation as well as successful implementation of these use-cases. As an example, let us consider a hypothetical dataset in Table 4.1. Say that we want to use this dataset to perform *predictive maintenance* for a given production robot. Now, let us consider that this seemingly innocuous dataset contains the requisite attributes with which predictive maintenance can be performed in theory.

But on digging in closer, we notice that there is missing contextual information. As a starting point, the units associated with the physical quantities are missing. Therefore, it is impossible to ascertain whether the *wear* values are a percentage, or an entirely different, possibly proprietary measure. The same is true for temperature. At the same time, there is no information about the acceptable values for these attributes, i.e., there is no way to find out whether the recorded data is *consistent* with the expected values or not. This makes it extremely difficult for data scientists to perform their job effectively, as they are often not domain experts.

In most typical data collection use-cases in automotive companies, this information is readily available with *data owners*, possibly in the form of an excel sheet. However, it is lost when the data is stored inside the data centre. We are still able to retain some

of the metadata which is available from the schemata, but it doesn't help our analytics tasks considerably.

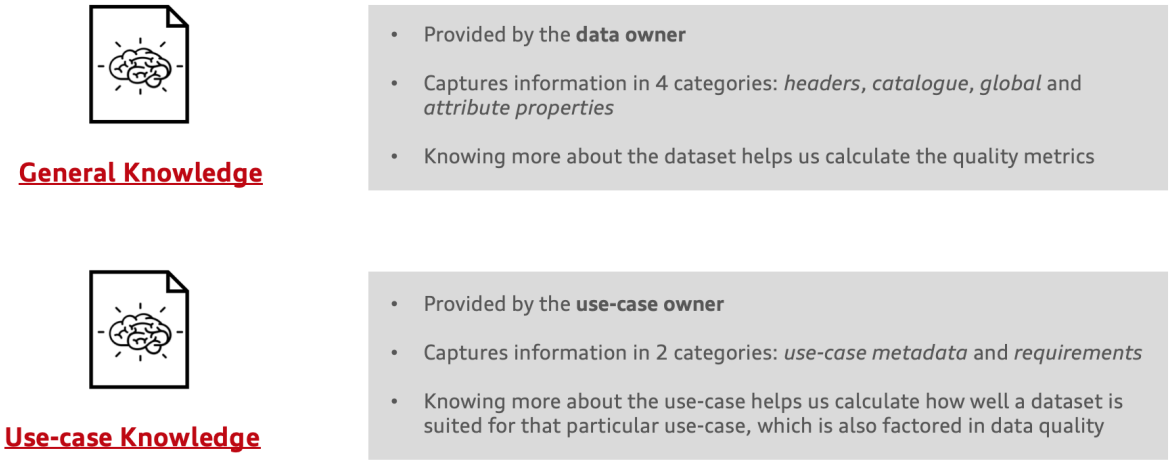

Figure 4.2: Brief Overview of the Two Types of *Knowledge Files*

We have used *knowledge files* in our approach to overcome this challenge. Knowledge files aim to collect metadata or contextual information about datasets. They are of two types: a *general* knowledge file and a *usecase* knowledge file. They are stored using the JSON syntax and have the extension `.know`. The reason for collecting knowledge files in a common format such as JSON is to ensure their human-readability, as well as the fact that modern JSON parsing libraries are quite powerful and require minimal effort on the programmer's side. The schema is easy to manipulate, and integrates very well with the MongoDB metadata repository, which is also a big advantage.

4.2.1 General Knowledge

The *general knowledge* of a dataset aims to collect information about the origin, destination, access and current analytics use-cases of a dataset, among others. Essential information is also collected individually about all the attributes present in the dataset. This is organized into four categories: a) headers b) catalogue c) global, and d) attribute properties. *Global* and *attribute properties* contain information that is used directly in the data quality and privacy evaluation. For example, the data type and range of the attributes provided in the general knowledge files is used to determine whether the incoming data for that particular attribute is *consistent* or not. Similarly, the measurement units help us decide whether a numeric attribute is *interpretable* or not. This kind of background information is quite essential, because it cannot trivially be inferred from

the dataset.

Headers and *catalogue* are collected because they were determined as being very helpful for data cataloguing and governance purposes. They do not contribute directly to the data quality and privacy evaluation. Figure 4.3 is a snapshot of a sample *general.know* which was provided for the *Bosch Weldlog* dataset. As explained above, it contains information organized within four categories. Figures 4.4 and 4.5 provide an illustration of the exact information is collected within these categories.


```
{
  "headers": {
    "datetime": "2018-09-20T07:48:05.724",
    "dataset-ID": "boschwps@datalakep1",
    "type": "general.know"
  },
  "catalogue": {
    "accessibility": "[FP/45,PN/62]",
    "responsibility": "Hr. Max Mustermann (N/FP-45)",
    "data-source": "Bosch WPS",
    "attributes": "[dateTime, timestamp, machine, current, energy, thickness, wear]",
    "data-description": "Data collected from the welding process in body shop",
    "historical-use": "[Data Lake P, Predictive Maintenance, Reporting]",
    "date-of-deletion": "2028-12-31T12:00:00.000",
    "data-storage": "Data Lake P"
  },
  "global": {
    "measurement-accuracy": "0.8"
  },
  "attributeProperties": {
    "dateTime": {
      "default": "0000-00-00T00:00:00.000",
      "unit": "n.a.",
      "privacy-sensitivity": "non-sensitive",
      "time-type": "collection",
      "maximum": "n.a.",
      "unit-shorthand": "n.a.",
      "data-type": "timestamp",
      "minimum": "n.a."
    },
    "current": {
      "default": "0.0",
      "unit": "ampere",
      "privacy-sensitivity": "sensitive",
      "time-type": "n.a.",
      "maximum": "10.0",
      "unit-shorthand": "A",
      "data-type": "numeric",
      "minimum": "0.0"
    },
```


Figure 4.3: A Snapshot of the Partial Contents of the *general.know* from the *Bosch Weldlog* Dataset

One of the most challenging facets of designing such an approach is to make it user-friendly and intuitive enough for managers and data owners, so that they would actually

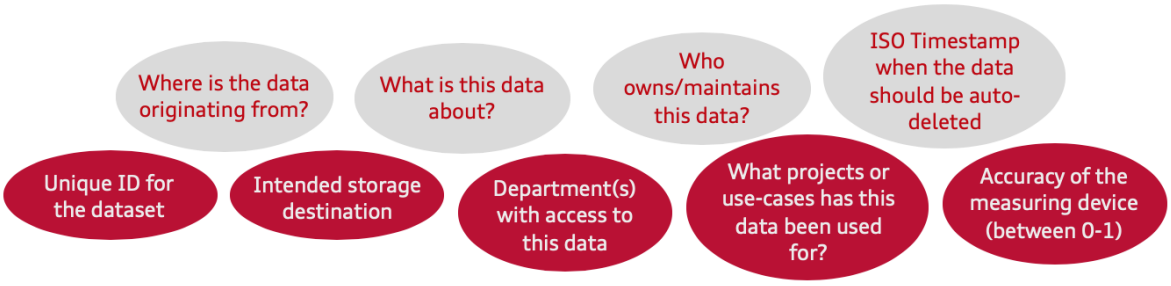

Figure 4.4: Metadata Collected within the *headers*,*catalogue* and *global* Categories

be able to use it properly, and adapt to it quickly. Therefore, for the implementation of our approach, we provided two options for managers to enter this metadata for both kinds of knowledge files. The first approach was a form-based interface using *Sonadier*[1] forms. Using this option, users can easily enter information in a minimalistic, form-based interface. It was also acknowledged that many data owners feel comfortable with Microsoft Excel. Therefore, we also provide an excel template where information can be filled in. The output in both these approaches is an *.xlsx* file, which is parsed by *excel parsers* provided in the *DeusHelper* module. These excel parsers convert the excel files into *.know* files, which are then used directly by the metadata extractor module.

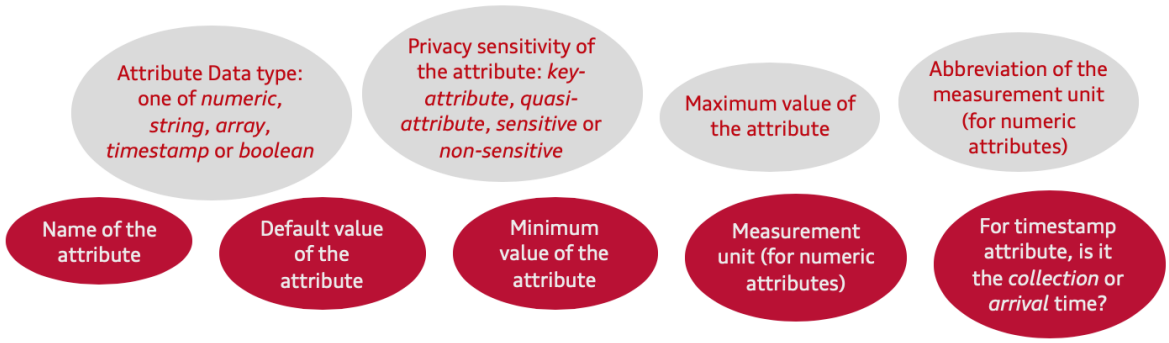

Figure 4.5: Metadata Collected within the *attribute properties* Category

4.2.2 Usecase Knowledge

Usecase knowledge is provided by *use-case* owner, i.e., the person/department responsible for conducting and implementing the analytics use-case. Using usecase knowledge,

[1]https://exeiscggxl.sonadier.io/forms

Deus evaluates the suitability of the dataset for the use-case in question, i.e., the *usability* part of the data quality metric. Information is collected within two categories: *a*) use-case metadata and *b*) requirements.


```
{
  "usecase": {
    "datetime": "2018-09-20T06:55:05.418",
    "usecase-owner": "[N/PN-63]",
    "dataset-ID": "boschwps@datalakep1",
    "usecase-ID": "ne_bodyshop_welding1",
    "analytics-attributes": "[current,energy,power]",
    "reputation": 1.0,
    "volatility": 1.0,
    "usecase-description": "Predicting welding faults as a result of high machine temperature",
    "type": "usecase.know",
    "desirable-attributes": "[voltage]"
  },
  "requirements": {
    "priority": "quality",
    "maximum-privacy-level": 1.0,
    "minimum-privacy-level": 0.0,
    "privacy-preference": "minimum",
    "minimum-data-quality": 0.8,
    "maximum-data-quality": 1.0
  }
}
```


Figure 4.6: A Snapshot of the Contents of the *use-case.know* from the *Bosch Weldlog* Dataset

Figure 4.6 is a snapshot of a sample *usecase.know* which was provided for the *Bosch Weldlog* dataset. As we can see, use-case metadata contains a description of the use-case, and information about the applicability of the dataset for it, such as *reputation*, *volatility* etc. This information is then used to calculate the values for the *usability* dimensions in the data quality evaluation. Information collected within the *requirements* category is used by the *Requirements Analysis* module (explained in detail in Section 4.4.3) to determine whether or not PETs are applicable on the dataset, and if they are, it decides which privacy level needs to be applied.

A comprehensive list of information that is collected within the scope of *use-case metadata* and *requirements* is shown in Figures 4.7 and 4.8

Usecase knowledge is collected once for every use-case and its corresponding dataset. This is intentional, and in order to make sure that data quality can be calculated for a particular dataset with respect to the use-case it will be employed for. As a part of our implementation, we also provided a detailed documentation with an explanation of the various fields for the knowledge files. This provides immense help to the concerned managers to provide the requisite information in its intended form.

The following sections will demonstrate in a more detailed way, the manner and tech-

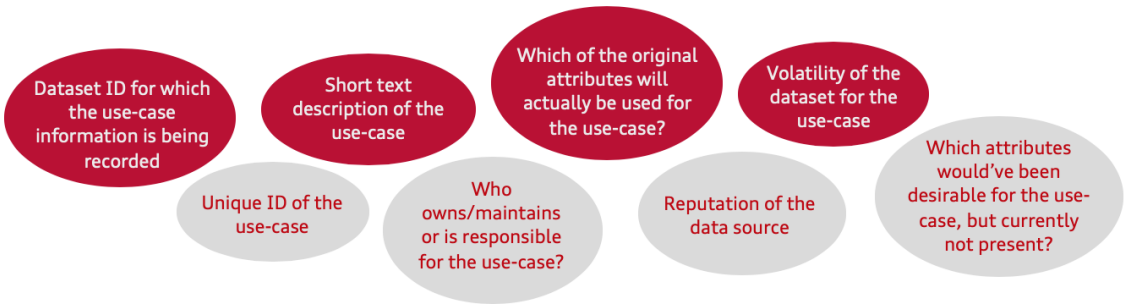

Figure 4.7: Metadata Collected within the *usecase metadata* Category

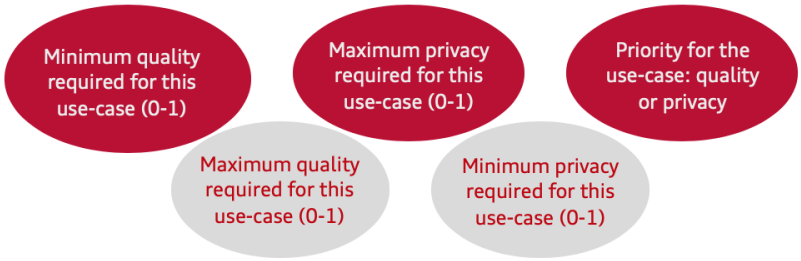

Figure 4.8: Metadata Collected within the *requirements* Category

nique using which we combine the given knowledge files and features extracted from the dataset. These are then subsequently used to retrieve the dimensional values for our data quality and privacy calculation in Apache Spark.

4.3 Data Quality

Data Quality is measured on the basis of *eight* fitness dimensions and *four* usability dimensions. These dimensions are calculated based on background knowledge available in the form of *knowledge files*, as well as a thorough mining of the dataset in question using the structured APIs provided by Apache Spark. In many cases, we agree on a scheme for converting the measured values to the dimensional measure for data quality based on an intensive feedback about how managers at Audi perceive the individual dimensions, and their effect on data quality as a whole. These conversion schemes, wherever applicable, are also illustrated. Another important consideration for interpreting the quality scores is: the scores always lie between *0* and *1*, with *0* being the worst possible quality and *1* being the best possible quality.

Once the dimensions have been calculated, the final score as well as the individual

fitness and usability scores can be evaluated using a user-defined formula. If no formula is specified, the final score is calculated as the average of the fitness and usability scores by default. The aim behind retaining fine-grained quality scores up to the attribute level, as this makes it very convenient to track down and resolve quality issues in data, as well as in designing use-cases that only use the most reliable and high-quality attributes. Let us see how the fitness and usability attributes are calculated.

4.3.1 Fitness Dimensions

Fitness dimensions are independent of data use, and are calculated solely on the merits of the data. The fitness dimensions, however, are calculated on a granular attribute level, i.e., every attribute in the dataset has their own set of fitness measures. These measures are then averaged to retrieve the fitness measures for the whole dataset. *Deus* provides eight fitness attributes by default, which are a careful selection of the techniques used in literature, especially the work of Wang and Strong[WS96], as well as the more recent paper by Firmani et al.[Fir+16] The definitions and recommendations provided in these papers were adjusted slightly to meet the requirements of automotive companies. For the formal definitions of these dimensions, please refer to Table 3.1

Accuracy

Accuracy of data refers to the closeness of the measured value to the correct value. E.g. in industrial robots, this is generally provided under the standard ISO 9283:1998[2], which covers 'Manipulating industrial robots - Performance criteria and related test methods'. Most industrial robots and IoT devices have more than one sensors embedded on them, e.g., for measuring temperature, position, rotation etc. Each of these physical quantities can have their own definition. As an example, position accuracy can be defined as 'the distance between the desired position and the centroid position which is actually achieved after repetitive movements of the end-effector toward the original desired position'[3]. In this case, the final accuracy measure is the average of the accuracy values of the individual embedded sensors.

The accuracy values for our approach are provided by data owners in the *general knowledge* file under the category *global*. It lies between 0-1 and is used as a global measure for each attribute.

[2]https://www.iso.org/standard/22244.html
[3]https://blog.robotiq.com/bid/72766/What-are-Accuracy-and-Repeatability-in-Industrial-Robots

Completeness

In the *completeness* dimension, we check the proportion of the dataset which are composed of *null* or *NaN* (not a number) values. These values are irrelevant for data analysis, and as a result, a dataset containing a high proportion of null or NaN values subsequently displays poor quality.

Numeric attributes are checked for null and NaN values, whereas *non-numeric attributes* are checked for null and null strings, i.e. where the input is the string 'null'.

Consistency

A measure of *consistency* helps us determine and track unexpected anomalies in the data. These anomalies could be wide-ranging: unexpected values ranging from sensor errors or error in data collection, to more serious errors such as data type errors, leading to higher data cleaning effort. If these values are not handled in a proper manner, they can also lead to incorrect results or in some cases, a malfunction of the learning algorithm.

In our approach, consistency is calculated as the proportion of the total records in the dataset follow the expected value or data type. For *numeric attributes*, a check is made whether the corresponding value lies within the expected range, i.e. in between its minimum and maximum value specified in *general.know*. On the other hand, *non-numeric attributes* are checked for their data type to determine whether the inferred *Apache Spark* data type conforms to the data type specified in *general.know*.

Timeliness

Managers swear by *timely* data, i.e. data which is timely collected and reported. This is indeed for good reason, as Cai et al.[CZ15] point out that data processing and analysis based on untimely or expired data will likely produce useless or misleading conclusions, leading to decision-making mistakes. Therefore, timeliness is another extremely important measure of data quality.

One of the important pre-requisites to measure timeliness is the availability of the timestamps for data collection as well as arrival. This holds true for most data collection scenarios which involve IoT devices. Therefore, we calculate timeliness as the 'average time difference between the collection of the data and its subsequent arrival in the data store'. The time difference calculated is then converted into the equivalent dimensional quality score using the illustration in Figure 4.9.

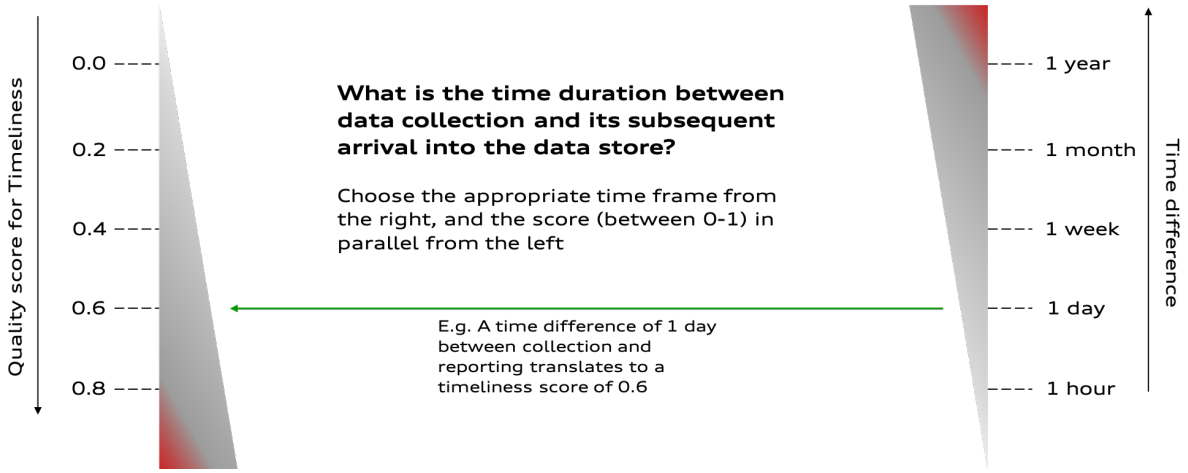

Figure 4.9: Converting Timeliness value to the Dimensional Measure of Fitness (not to scale)

Uniqueness

Uniqueness is another important dimension for data quality. According to Information Theory, a higher uniqueness signifies a higher *Shannon Entropy*, which is a measure of the amount of information contained in data. In our approach, uniqueness is measured as the proportion of rows of the DataFrame which are distinct.

Volume

In one of the most important papers ever written in the field of Data Science, Halevy et al.[HNP09] argue that *more data* often beats *better algorithms*. In fact, this is one of the major reasons for the inevitable rise of big data technologies. The power of data at scale can just not be underestimated. Therefore, volume becomes a very important indicator of data quality.

In our approach, the number of data rows is converted into the dimensional measure of data quality by using the illustration in Figure 4.10

Interpretability

In order to make crucial decisions based on the results of data mining algorithms, they must be interpretable [4]. This is possible only when the underlying data itself is also interpretable. The example in Section 4.2 proves the importance of interpretability towards data quality.

[4]https://www.oreilly.com/ideas/why-interpretability-matters-in-data-analytics

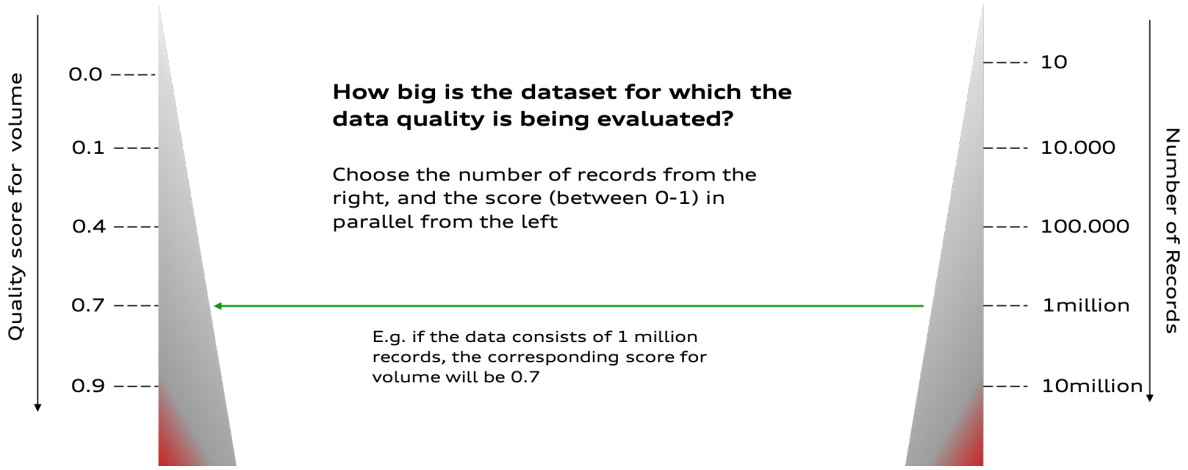

Figure 4.10: Converting Number of Rows in the Dataset to Dimensional Measure of Fitness (not to scale)

In our approach, we measure interpretability for the *numeric attributes* as the proportion of attributes which have well-defined units for unambiguous interpretation. This is calculated by determining whether their units and shorthand are defined in the *general.know*.

Credibility

The general knowledge file defines the default value for every attribute. This helps us in calculating the *credibility* dimension of data quality, which is the proportion of rows in the DataFrame which are not the default values of the particular attribute. Each row of the DataFrame is individually compared to the default value using a *user-defined function* in the Scala API for Spark.

4.3.2 Usability Dimensions

Unlike fitness dimensions, usability dimensions are dependent on the data use, and are calculated on the basis of how well the dataset is suited for the use-case in question. This means that the usability score of the same dataset varies from use-case to use-case, which provides a fairer interpretation of data quality. This is because a particular dataset must have a varying notion of quality based on the estimate of how useful would be for a particular scenario. *Deus* provides 4 usability dimensions by default, which are a reflection of the results of a survey conducted at the company. For the formal definitions of these dimensions, please refer to Table 3.2.

Volatility

It has been shown that *volatile* data, i.e. data which has a short expiry, and that becomes irrelevant for a use-case in a very short time, has a low utility for long-term decision making[AlD11]. At the same, volatile data should neither be used to access risk or for use-cases involving risk, nor to confirm a hypothesis. Therefore, the volatility of a dataset for a use-case is an essential dimension for estimating its utility.

The volatility of a dataset for a particular use-case is provided by the use-case owner in the *usecase.know* file, and is converted to the dimensional measure of utility according to the illustration in Figure 4.11

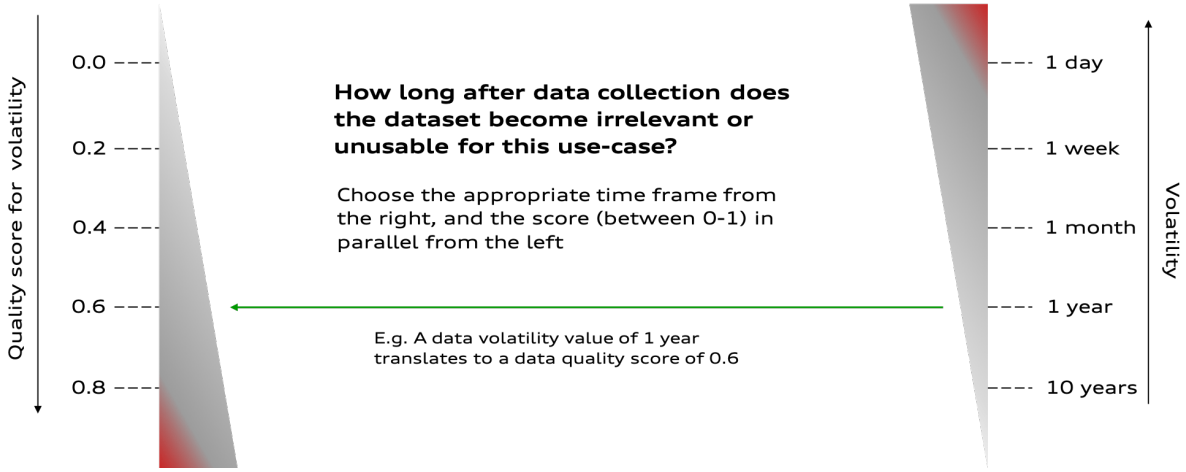

Figure 4.11: Converting the Volatility of the Dataset to Dimensional Measure of Usability (not to scale)

Reputation

Reputation, or the trust factor of the origin of data is an important factor that determines whether important business and strategic decisions are taken on the basis of data. This is extremely relevant for automotive companies, which are generally large organizations with a well-defined hierarchy. This hierarchical structure gives rise to the concept of *trust boundaries*, which must be taken into account. Based on how close the trust boundaries are, to the department that facilitates the use-case, the reputation score of a particular dataset can be calculated for that use-case.

In *Deus*, we divide the trust boundaries into 6 levels, as also illustrated in Figure 4.12. Here, we take the example of the AUDI AG, but the general solution holds true for any organization of comparable size and structure. For the sake of simplicity, let us assume that the department which drives the use-case is called *Department D*.

1. **Complete Trust**: This is the highest level of trust, which is assigned a reputation score of *1.0*. E.g. data originating from *Department D*.

2. **Reasonable Trust**: This is the second highest level of trust, which is assigned a score of *0.9*. E.g. data that originates from a department that works in close co-operation with *Department D*, or has a partnership with it.

3. **Considerable Trust**: This level is assigned a score of *0.7*. E.g. data that originates from within the AUDI AG.

4. **Plausible Trust**: This is the fourth highest trust level, and is assigned a score of *0.5*. E.g. data that originates from within the companies of the Volkswagen AG, such as Skoda Auto, Automobili Lamborghini S.p.A. etc.

5. **Sufficient Trust**: This trust level is assigned a score of *0.15*. E.g. data originating from partner companies or third-party suppliers which have a relationship with the company. E.g. Kuka, Böllhoff.

6. **Insufficient Trust**: This is the lowest trust level, which is assigned a score of *0.0*. E.g. data originating from public sources on the internet, or from other companies that have no relationship with Audi.

Relevance/Utility

Relevance or *Utility*, as the name suggests, measures the fraction of the attributes present in the dataset, which are actually used for the analytics task. In the *use-case knowledge*, we have the information about attributes which will be used for the impending data analytics use-case. The relevance/utility score of the dataset is then measured as this number divided by the total number of attributes in the dataset, as shown in Equation 4.1

$$\boxed{relevance = \frac{num(attributes_{usecase})}{num(attributes_{total})}} \tag{4.1}$$

Desirability

Often times, use-case owners are hassled by the fact that some of the important features required for a learning or analytics task is not available at hand. However, there is no measure of data quality that takes this into account, nor a proper channel or mechanism

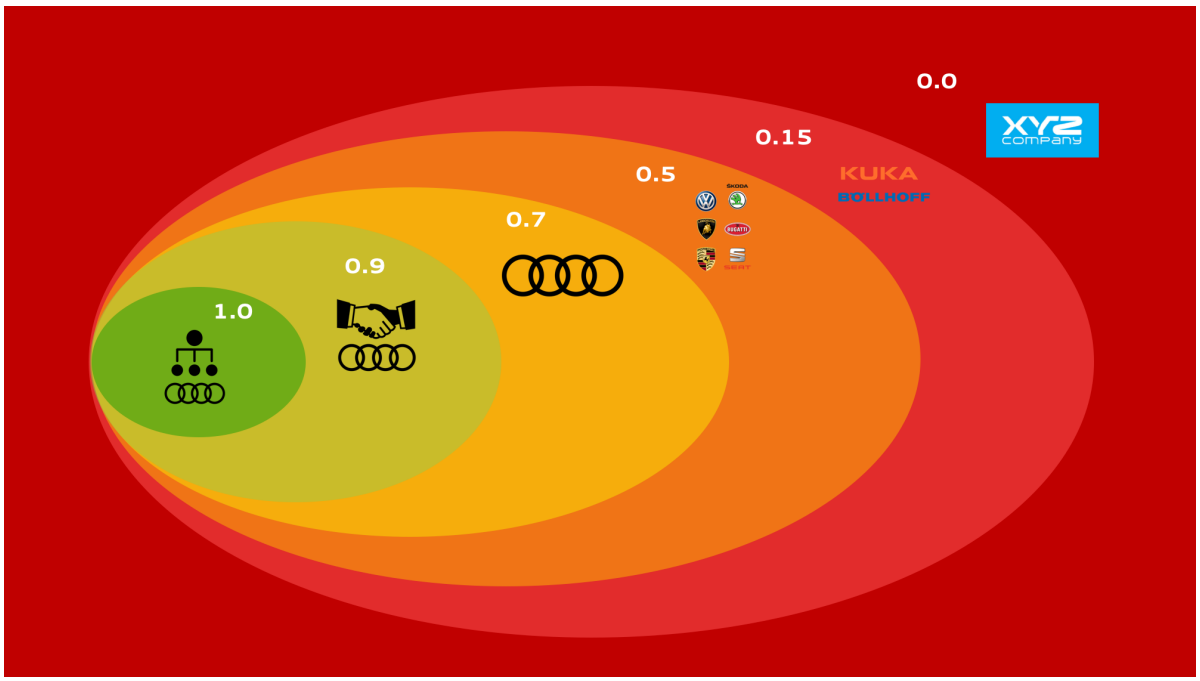

Figure 4.12: Trust Boundaries affect Reputation Scores

where this grievance can be recorded and analysed. The measure of *desirability* in our approach tries to address this problem.

The *usecase.know* consists of the list of attributes, which according to the use-case owner's evaluation, are desirable for the use-case, but are not available in the dataset. This is then used to evaluate desirable with the help of Equations 4.2 and 4.3

$$\boxed{frac_{desirable} = \frac{num(attributes_{desirable})}{num(attributes_{total})}} \tag{4.2}$$

$$desirability = \begin{cases} 1 - frac_{desirable} & , frac_{desirable} \leq 1, \\ 0 & , frac_{desirable} > 1. \end{cases} \tag{4.3}$$

4.4 Data Privacy

Alongside data quality, utility-driven *Data Privacy* is a cornerstone of this thesis. For this, we must use both the concepts side-by-side, in order to estimate their influence on one-another. Utility-driven is a win-win scenario for all the parties, and therefore such an approach which develops trust and enables data sharing is truly worthwhile. This requires an approach to measure the level of privacy along with data quality, and the

application of PETs to determine their effect on privacy, and consequently also on data quality metrics, especially usability.

Deus provides the following solution in this record: we offer an extendible set of dimensions to measure privacy, similar to the dimensions for data quality. Using these dimensions, we can not only measure privacy, but also evaluate the change in these metrics when PETs are applied. Regarding the privacy-enhancing algorithms themselves, we have packaged them into various privacy levels. The default implementation provides *privacy levels*, *1* to *3*, which are arranged in increasing order of strictness. This means that privacy level 1 contains the least strict PETs, whereas level 3 contains the strictest algorithms. We also illustrate the functioning of the *requirements module*, that determines which privacy level needs to be applied to the dataset based on the requirements set by the use-case owner.

Furthermore, to adjust the usability measure of the dataset according to the PETs applied, we provide a mechanism called the *Cumulative Penalty Factor*. This will be explained in 4.4.4. Finally, we describe the modular organization of the application software in 4.6.

4.4.1 Privacy Dimensions

Deus provides three privacy dimensions, namely *Non-Sensitivity, Distinguishability* and *Non-Linkability*. The dimensions are dependent on the classification of the attributes in the following categories:

1. **Key Attribute:** Also known as *personally identifiable information*, these attributes can uniquely identify a certain individual or entity (including machines, factories, parts etc.) within the organization. E.g. a unique ID or part number.

2. **Quasi-Identifier:** Quasi identifiers can't identify an individual or entity by themselves, but in conjunction with other key attributes, quasi-identifiers or background knowledge, they can possibly contribute to their re-identification. E.g. zip codes.

3. **Sensitive Attribute:** These are the main attributes on which the data analytics tasks are based. Sensitive attributes are those which, if externalized, can reveal business secrets of the organization. E.g. current, voltage readings from an IoT-enabled production robot.

4. **Non-Sensitive Attribute:** These attributes are also important in analytics tasks, but externalizing them does not reveal any sensitive information or business secrets.

E.g. weather.

The privacy metrics range from *0* to *1*, where *0* signifies the lowest level of privacy, whereas *1* signifies the highest level of privacy. Following are the privacy dimensions used for evaluating the privacy score:

Non-Sensitivity:

Non-sensitivity measures the extent to which a dataset is vulnerable, if it is externalized in an improper manner, i.e. without applying any privacy-enhancing technologies. It is measured as the proportion of attributes in the dataset which are *not* key attributes or quasi-identifiers, as shown in Equation 4.5.

$$\boxed{nonsensitivity = 1 - \frac{num(attributes_{key}) + num(attributes_{quasi})}{num(attributes_{total})}} \tag{4.4}$$

Distinguishability:

The measure of *distinguishability* is based on the premise that if there is diversity in the combinations of key attributes and quasi identifiers, one can be more certain that not a lot of information about one identifiable entity can be collected. However, on the other hand, if there are lots of redundancies in the key attributes and quasi-identifiers, it means that there is a lot of information in the dataset about a single entity, therefore plethora of information can be collected about it. Distinguishability is calculated by measuring the proportion of distinct combinations of key attributes and quasi-identifiers in the dataset.

Non-Linkability:

Linkability is a major privacy-threat, and once identifiable information has been released, it is extremely difficult to control its occurrence. Therefore, in order to get rid of this problem, we have used *tokenization* as one of the privacy-enhancing technologies, which hinders the linkability of data. *Non-Linkability* is measured as the proportion of attributes in the dataset which have been tokenized.

$$\boxed{nonlinkability = \frac{num(attributes_{tokenized})}{num(attributes_{total})}} \tag{4.5}$$

4.4.2 Privacy Levels and Privacy-Enhancing Technologies

To enable privacy guarantees upon release of datasets, we use different *privacy-enhancing technologies*. These PETs are packaged as privacy levels for various reasons: privacy levels provide an intuitive, ordered understanding to the data owner as well as the data user. Pre-packaged levels mean that there are lesser parameters to be set by hand, which is often a difficult task. Most importantly, privacy levels help manage and counteract the unreliable relationship between individual PETs and privacy metrics. Instead, pre-packaged privacy levels provide a reliable, more visible and gradual transformation in the privacy metrics as the PETs become stricter. This also helps us calculate their effects on the usability of the data, which we will also see in 4.4.4.

Our approach provides three privacy levels, ranging from *Level 1* to *Level 3*, and a higher numeric value signifies stricter privacy enforcement. However, more privacy levels can be added as and when required for data transfer or sharing scenarios. Figure 4.13 provides an illustration of the various PETs that are used for enabling the privacy levels, as well as the configuration in which they are used. The three major technologies that we use in our approach are: *k-Anonymity*, *tokenization* and *attribute suppression*, which we will also explain in the forthcoming sections. We have used these PETs because of their omnipresence as privacy-enforcing tools, as well as their ease of understanding.

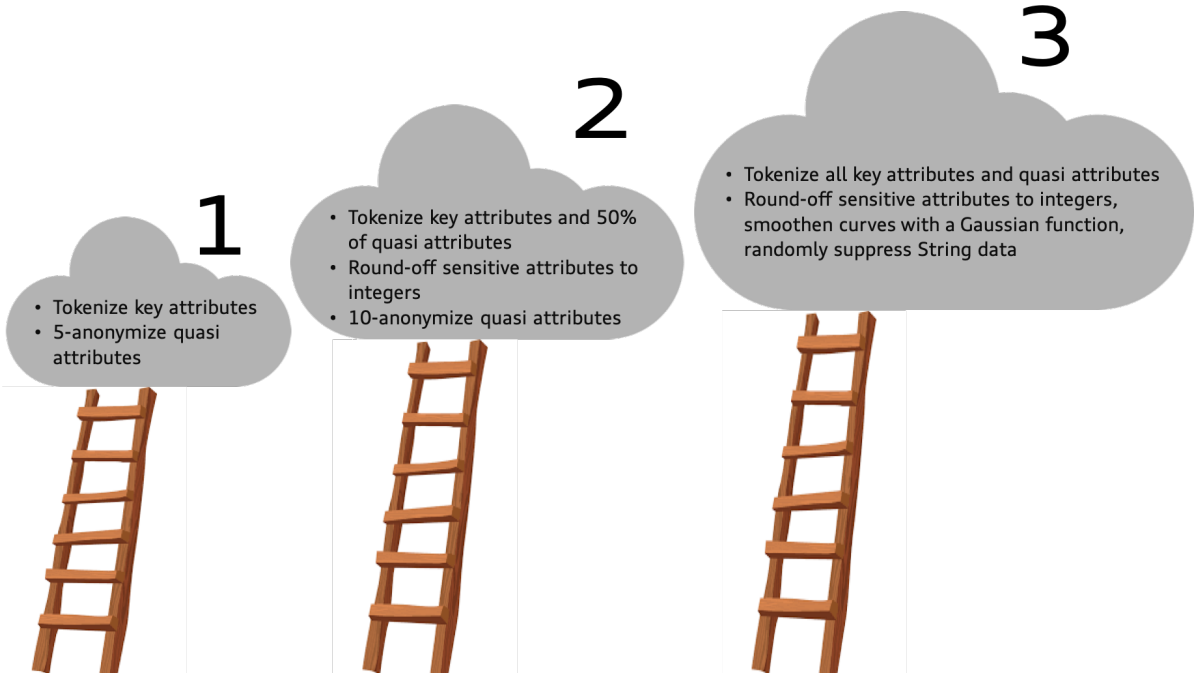

Figure 4.13: Privacy-Enhancing Technologies packaged into Privacy Levels

Mondrian k-Anonymity

k-Anonymity is a well-known and widely used PET for data publishing, which was introduced by Sweeney[Swe02], and which immediately caught the attention of privacy researchers as well as data scientists alike. Although the basic principle of k-Anonymity is quite simple and easy to understand, it provides a powerful safeguard towards possible re-identification of subjects contained in a released dataset.

Definition 1. *K-Anonymity Let $RT(A_1, \ldots, A_n)$ be a table and QI_{RT} be the quasi-identifier associated with it. RT is said to satisfy k-anonymity if and only if each sequence of values in $RT[QI_{RT}]$ appears with at least k occurrences in $RT[QI_{RT}]$.*

In essence, k-anonymity ensures that information for each person contained in the released table cannot be distinguished from at least $k-1$ individuals whose information also appears it. These records with the same quasi-identifier are said to form an *equivalence class*. Although k-anonymity provides considerable privacy guarantees, k-anonymized records are not immune to attacks, as revealed in by Narayanan and Shmatikov in [NS08]. In particular, attacks can take advantage of pre-existing knowledge about the subjects *(background knowledge attack)* or the homogeneity in the sensitive attributes *(homogeneity attack)*.

In terms of implementation, k-anonymity also presents some challenges, as *generalization* of records fundamentally relies on spatial locality, which means that each record must have k close neighbours. This is usually not the case in sparse datasets, sometimes leading to dissimilar records being merged into a single equivalence class.

In *Deus*, we use a version of k-anonymity known as *Mondrian k-anonymity* for implementation because it frequently leads to more desirable anonymizations[LDR06]. Mondrian k-anonymity was introduced by LeFevre et al., and is a greedy approximation algorithm. A *strict* multidimensional partitioning defines a set of non-overlapping multidimensional regions as shown in Figure 4.14, mapping each tuple to a summary statistic for the region in which it is contained. This approach is named after *Piet Mondrian*, a well known Dutch painter, because the resulting vector space resembles his artwork.

When k-anonymity is applied to a table, the tuple set in each non-empty region forms an equivalence class with respect to the quasi-identifiers. Thereafter, each record within a group is generalized such that each group has the same quasi-identifiers.

A pseudocode of the Mondrian k-anonymity algorithm is provided in Algorithm 1. In our implementation, we have tried to leverage the distributed processing capabilities of *Apache Spark*, and therefore, the k-anonymization is also written in Scala and meant to

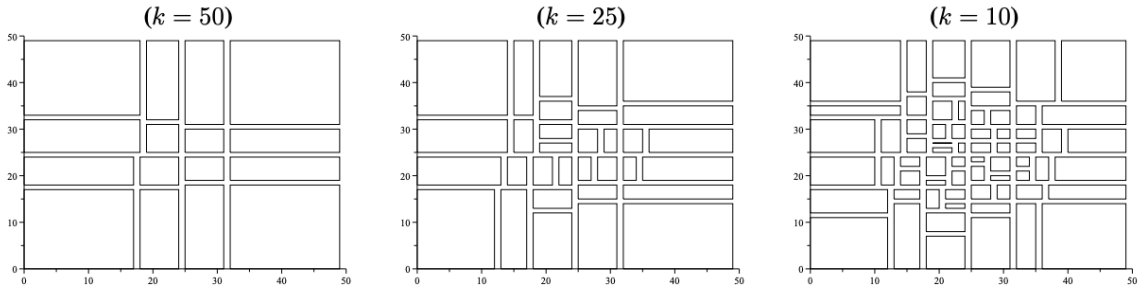

Figure 4.14: Partitions created by *strict* Mondrian k-Anonymity. Taken from [LDR06]

be executed on the Spark compute framework.

Algorithm 1 Mondrian k-anonymity

```
1: procedure ANONYMIZE(partition)
2:     if no allowable multidimensional cut for partition then
3:         return φ : partition → summary
4:     else
5:         dim ← choose_dimension()
6:         fs ← frequency_set(partition, dim)
7:         splitVal ← find_median(fs)
8:         lhs ← t ∈ partition : t.dim ≤ splitVal
9:         rhs ← t ∈ partition : t.dim ≥ splitVal
10:        return Anonymize(lhs) ∪ Anonymize(rhs)
11:    end if
12: end procedure
```

Tokenization

Tokenization is the process of randomly generating a token value for plain text, and storing the mapping inside a database. It is one of the more effective ways of information hiding, apart from *hashing* and *encryption*. In contrast to hashing, tokenization ensures that the same data values do not receive the same identifiers, thereby making it extremely robust and immune to linkage of datasets using these identifiers. This is our main motivation for using it as a part of our privacy approach. Another reason of not getting rid of these quasi-identifiers altogether, is that tokenization provides a unique reference, that serves as an identifier for the data user to later request the owner(s) for the actual information. This capability would be totally lost if we were to eliminate the quasi-identifiers completely.

Tokenization was once frowned upon as an expensive solution, unsuitable for large

amounts of data. However, with the advent of cheap cloud storage and an increasing preference for cloud-based infrastructure as opposed to on-premise, tokenization is picking up as an important tool for information privacy. Tokenization with secure cloud data vaulting is much more secure and ultimately less expensive than on-premise solutions. With cloud tokenization, sensitive data can be completely removed from an organization's IT environment [5].

For this thesis, we developed a unique tokenization approach called as *Secure Tokens*. Secure tokens is a method for assigning pseudo-random tokens in a large dataset. It takes advantage of the *md5* hash function, and adds a 4-character random string *(salt)* to it. The positioning of the salt is also done via pseudo-random selection, which provides a stochastic nature to the final token, making the probability of a collision extremely small. However, the real 'security' of the secure tokens originates in the manner that the token vault is stored on disk. The token vault uses symmetric encryption via *AES*, and always exists in an encrypted form on disk. The helper functions for retrieving the data value for a particular token always destroys intermediate data structures that store unencrypted data. This dispenses an additional layer of security to the token vault, while making it suitable for access using the helper functions, provided that the user is in possession of the secret key. Figure 4.15 provides a high-level illustration of the *Secure Tokens* approach.

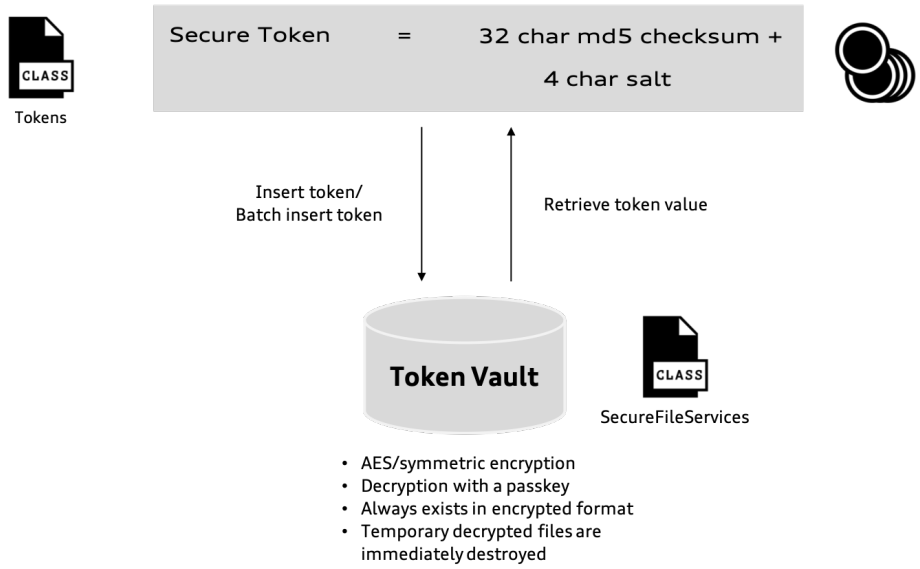

Figure 4.15: Tokenization Approach Developed as a Part of *Deus*

[5]https://tokenex.com/resource-center/tokenization-vs-encryption/

Pseudo-random Data Suppression

Data Suppression is the technique of withholding or removing certain pieces of information from a dataset. Within *Deus*, we approach data suppression using a parameter-based pseudo-random method. The *suppressor* function takes a positive integer n as input, and generates a pseudo-random number between 1 and n. If the row count of the Dataframe in question is fully divisible by the generated pseudo-random number, then the contents of the particular attribute for the given row number is replaced by 'suppressed'. Note that since the data suppression is always done column-by-column, the possibility of all entries in a row being suppressed is very low.

Let us consider an example: We want to apply *suppressor(20)* to column 3 and row 121 of a particular Dataframe. We choose a random number between 1 and 20, say 12. In this case, the data will not be suppressed as $121(mod 12) \neq 0$. However, if the chosen random number was 11, the data would be replaced by the string 'suppressed', since $121(mod 11) = 0$

The benefit of using such a parameter-based approach is that the suppression can be made more/less strict based on the supplied parameter, which makes it very convenient to be re-used for different privacy levels.

4.4.3 Requirements Analysis

The *Requirements Analysis* module, as described in Section 3.3, is responsible for ascertaining whether or not privacy levels are required to be applied on a particular dataset, based on the requirements of the use-case, which are contained in the *usecase knowledge*. If privacy levels need to be applied, the module's job is to determine which privacy level is suitable.

Let us consider an example. A hypothetical use-case places its priority on *privacy*, and mandates a minimum privacy level of *0.8*. Upon data quality and privacy evaluation, it is found that the privacy score is *0.6*. The requirements module then applies *Privacy Level 1* and re-evaluates data quality and privacy. This time, the privacy score is found to be *0.76*. Since this is still less than the mandated privacy level of *0.8*, the subsequently higher privacy level *(Privacy Level 2)* is applied. Upon re-evaluation of data quality and privacy, the privacy score comes out to be *0.85*. Since falls within the accepted boundary, it is accepted, and no more privacy levels are applied.

The complete decision-making flowchart of the requirements analysis module is shown in Figure 4.16

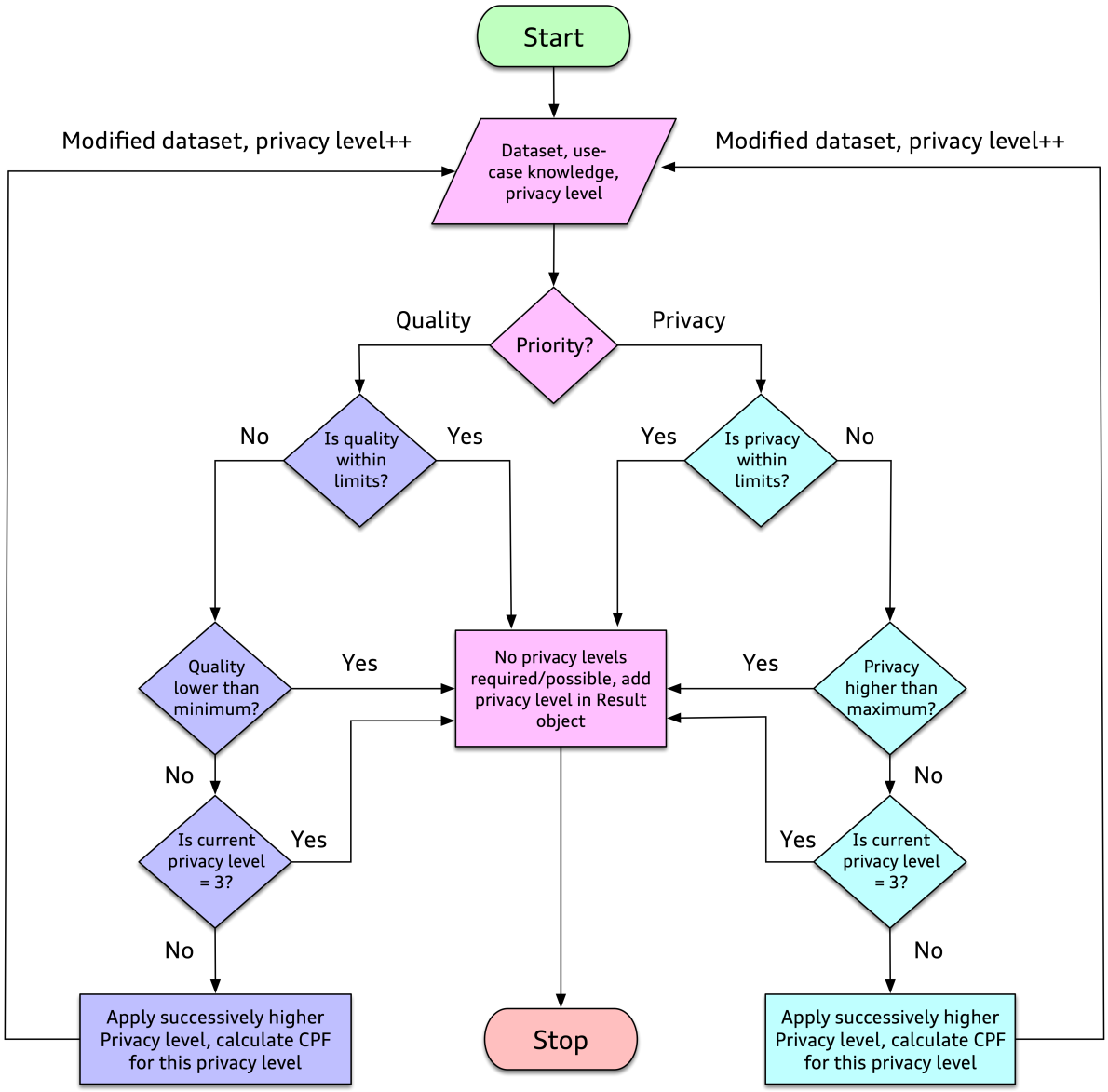

Figure 4.16: Flowchart of the *Requirements Analysis* Module

The final *Result* file generated by data quality and privacy evaluation entails information about the final privacy level applied on the dataset, along with a remark about the reason why this privacy level was chosen. If no privacy levels are deemed to be sufficient for the particular use-case, then the original dataset is returned. However, if one of the privacy levels are applied to the dataset, then the new dataset is written back to the original data source.

4.4.4 Cumulative Penalty Factor

Although data quality evaluation and application of PETs work very well isolation, computing the relationship between data quality and privacy metrics has been one of the biggest missing pieces of the puzzle so far. There is a lack of robust yet flexible techniques, which measure this relationship in accordance with the use-case requirements and the strictness of the applied PETs. *Cumulative Penalty Factor* is an effort to bridge this gap. CPF tries to calculate the usability loss that is affected by applying the privacy levels on the dataset. On multiplying the CPF with the usability score for the dataset, we introduce the *adjusted usability score*, which takes into account the aftermath of the PETs on the dataset, keeping into consideration the attributes of the dataset which are actually used for the use-case.

To achieve this, we first use a reference table that lists the *perceived usability loss* for each of the applied PETs individually, or if the PETs operate in tandem, their combined usability loss. The reference table used for calculation of the CPF in the default approach is shown in Table 4.2.

PET Used	**Usability Loss**
Tokenization	100
5-anonymization	20
10-anonymization	30
Rounding-off	5
Partial Suppression(*100*)	10
Rounding-off + Partial Suppression(*100*)	20

Table 4.2: Usability Losses for the various PETs used in *Deus*

These values are then used in the following equations, which ultimately yields the CPF value for the given privacy level.

$$Combined\,Usability\,Loss = \frac{\sum_{n=1}^{num(PETs)} num(attributes_{PET_i}) \times usabilityLoss_{PET_i}}{num(privacyAttributes) \times 100} \tag{4.6}$$

where,
$num(attributes_{PET_i})$ = the number of attributes affected by PET_i
$num(privacyAttributes)$ = the number of attributes affected by the given privacy level

$$\boxed{Privacy\,Ratio = \frac{num(privacyAttributes)}{num(totalAttributes)}} \tag{4.7}$$

where,
$num(totalAttributes)$ = the total number of attributes in the dataset

$$\boxed{Normalized\,Usability\,Loss = Usability\,Loss \times Privacy\,Ratio} \tag{4.8}$$

$$\boxed{Cumulative\,Penalty\,Factor = 1 - Normalized\,Usability\,Loss} \tag{4.9}$$

$$\boxed{Usability_{adjusted} = Usability_{raw} \times Cumulative\,Penalty\,Factor} \tag{4.10}$$

CPF is also a very useful indicator that helps in determining the strictness of the privacy levels, as a lower CPF value corresponds to a stricter privacy level. This means that they are also very effective in ranking arbitrary privacy levels. For example, for one of the datasets we used for evaluation, level 1 yielded a CPF of *0.80*, whereas level 2 and level 3 yielded scores of *0.642* and *0.457* respectively, confirming our strictness ranking for the privacy levels. When more privacy levels and PETs are added to the approach, the reference table can be updated accordingly, and the CPF can be calculated for the new privacy levels, thereby making it truly flexible and use-case driven.

4.5 Technologies

In Section 3.1, we have explained in detail, the important factors that we needed to keep into account while designing our approach for data quality and privacy. In a similar manner, the choice of technologies must take into account all those considerations, i.e. it must be *general-purpose, flexible, transparent, context-driven,* must *fit into the current infrastructure*, and must be *designed for big data.*

Big Data radically changes an organization's data strategy, and it implies that henceforth any application that operates on data must be built for scale. While making our choice of technology, we have to keep in mind the needs of today as well as tomorrow. An application for a data quality and privacy approach has to be especially scalable, because it will likely by employed on huge volumes of data. The resulting metadata, on the other hand, must be highly available and easily searchable. We see how the following technologies shown in Figure 4.17, which form the backbone of our approach, fulfil the requirements of an automotive company.

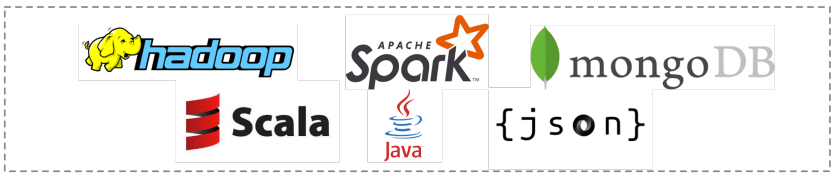

Figure 4.17: The Primary Technologies used for *Deus*

The Hadoop Ecosystem

Hadoop, as a tool, is now omnipresent in the big data space. The Hadoop software library is a framework that allows for the distributed processing of large data sets across clusters of computers using simple programming models. Hadoop forms an integral part of our design choice because of the fact that it is designed to scale up from single servers to thousands of machines, each offering local computation and storage. Since it provides a highly-available service on top of a cluster of computers, each of which may be prone to failures, Hadoop is likely to remain a technology of choice for some years to come.

Hadoop, along with its plethora of core and associated projects and technologies such as HDFS, YARN, MapReduce, Pig, Hive, Spark, Tez, HBase, Zookeeper, Oozie, Mesos, Storm etc. make Hadoop an absolute behemoth for transforming and analysing massive datasets on a cluster of computers. For this reason, Hadoop has become a central component of the data strategy at any company that handles and maintains big data. Many of these companies have also built their own data centres, as well as a well-defined data integration and ingestion pipeline. The fact that Hadoop is the technology of choice for today's computing needs, and that it is already an integral part of the technology stack of many automotive companies, makes it the platform of choice for *Deus*.

Apache Spark

Apache Spark is a general-purpose compute engine designed for distributed data processing at scale. As a truly unified platform, Spark can tackle a wide range of data analytics tasks, ranging from data loading and SQL queries, to complex machine learning and streaming analytics over the same compute framework and a consistent set of APIs. Spark's APIs are designed to deliver in very high-performance workloads by optimizing its tasks over the entire application, across the different libraries and functions.

Spark traces its humble beginnings from UC Berkeley's *Spark Research Project* which started in 2009. It sought to overcome the deficiencies of MapReduce, which was the dominant compute framework at that time. MapReduce made it extremely inefficient to

run large scale iterative machine learning algorithms, which required multiple passes over the data. The Spark API, on the other hand, performs efficient, in-memory data sharing across the computational steps, making it ideal for machine learning applications.

Spark Applications consist of a driver process and a set of executor processes. *Driver* is the center of the Spark Application; it maintains relevant information during the lifetime of an application, whereas *executors* are responsible for carrying out the work that the driver assigns them. In its first phase of execution, Spark takes user code and converts it into a logical plan. After successfully creating an optimized logical plan, Spark then begins the physical planning process. The physical plan, often called a *Spark plan*, specifies how the logical plan will execute on the cluster by generating different physical execution strategies and comparing them through a cost model.

At the core of Spark's data abstraction are the *Resilient Distributed Datasets (RDDs)*, its low-level programming interface, which are immutable, partitioned collection of data which can be operated upon parallel. On top of the RDDs, Spark provides extremely user-friendly *Structured APIs* such as *DataFrames* and *Spark SQL*. *Deus* makes heavy use of both these structured APIs for its metadata extraction module. A DataFrame can be thought of as a spreadsheet with named columns, however there's one fundamental difference: a Spark DataFrame can be partitioned and located across thousands of computers. The DataFrames API provides several functions to efficiently process large volumes of data.

For us, the most important functionality offered by the DataFrames API is the uniform representation that it provides for numerous data sources. This means that no matter what the data source is, be it JSON, XML, CSV, a relational table, MongoDB document or a Hive table, it can be effectively converted and processed as a DataFrame. This inherent integration step provided by the DataFrames API makes sure that *Deus* doesn't have to provide adapters to connect to various data sources. Indeed, the heterogeneity of data sources is elegantly handled by the consistent representation via DataFrames, and the powerful conversion functions within Spark's impressive API.

Spark SQL makes it possible to register any DataFrame as a table or view (a temporary table) and query it using pure SQL, which is convenient, but at the same time, efficient and powerful. It essentially means that Spark can represent and process a multitude of data formats in a single, unified manner. These capabilities make it very suitable and relevant for data science and engineering.

MongoDB

MongoDB is an *open-source*, *non-relational*, *document* database developed by MongoDB, Inc [6]. It stores data as documents in a binary representation called BSON (Binary JSON). Related information is stored together for fast query access through the MongoDB query language. Fields can vary from document to document; there is no need to declare the structure of documents to the system - documents are *self-describing*. If a new field needs to be added to a document, then the field can be created without affecting all other documents in the collection, without updating a central system catalogue, and without taking the system offline. Optionally, schema validation can be used to enforce data governance controls over each collection.

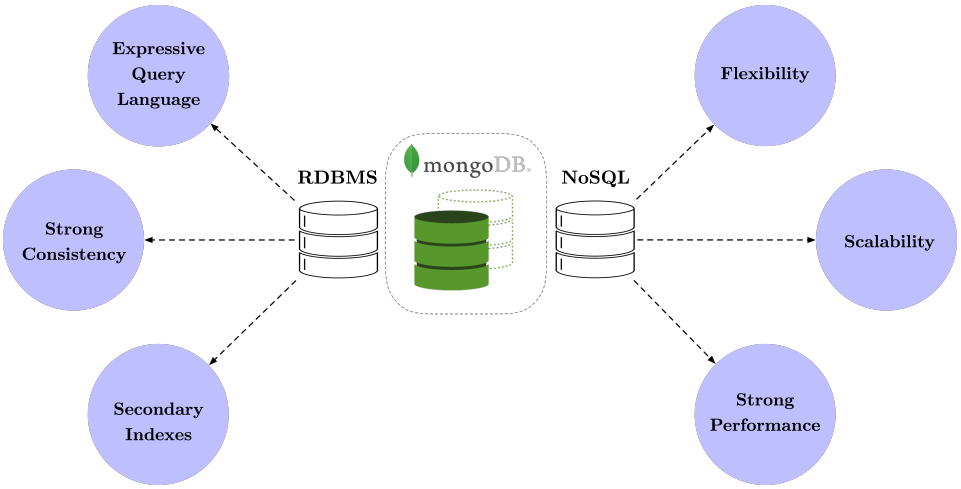

Figure 4.18: MongoDB Offers the Best of Both Worlds

These features, and the fact that MongoDB carries ahead many desirable features from a relational databases, make it an optimal choice for an organization-wide metadata repository. It is fit for the big data age: distributed, highly available because of replication, and highly scalable due to sharding. At the same time, it also provides the schema-flexibility that is desired in the metadata store of scale. Its query language is as expressive as SQL, and also almost equally as powerful. However, the strongest argument in its favour is the fact that MongoDB is able to efficiently address the issue of representing hierarchies in data. This is the biggest problem with RDBMS as a metadata store. Since metadata will often always be hierarchical, documents can be stored and optimally queried with reference to the hierarchical nature of data.

[6] https://www.mongodb.com/what-is-mongodb

Scala

Scala (short for 'SCAlable LAnguage'), is a highly scalable general purpose programming language which combines aspects of object-oriented and functional programming. It is becoming increasingly important in the world of data science, alongside more traditional options such like Java and Python. Much of this impetus is due to the growth in popularity of Apache Spark, which has been written in Scala. Spark has sprung forward to become the largest open-source project in data processing[7], giving Scala the well-earned reputation of being a reliable programming language for general-purpose data processing, machine learning and streaming analytics.

Scala is a JVM language, and brings along Java's strong reliability, safety and portability, as Scala source code is compiled to Java bytecode and run on the JVM. It improves on the verbosity of Java, and gives the programmer lots of options to write code in a concise and more expressive manner. Its hybrid design - combining object-oriented and functional methodologies, gives a programmer the flexibility to use, or not-to-use, the features of either of these paradigms. This is reinforced by its highly sophisticated type system, which provides the security of compile-time type checking, without the hassle of explicitly specifying them all the time.

The biggest attraction of Scala for this project, however, is its close relationship with Apache Spark. While Spark also provides libraries for Python, Java and R, the advantages of working in its original language are manifold, including the ability to use latest features which haven't yet been ported to other languages.

4.6 Implemented Modules

The main module is called as *DeusMain*, which is written in Scala using the Spark API. It performs a bulk of the processing for calculation of quality and privacy scores, as well as requirements analysis and application of privacy levels to the dataset. DeusMain is supported by three modules, out of which two *(util and privacy)* are internal, whereas one *(DeusHelper)* is external. The modular organization is also illustrated in 4.19. The roles and functions of these modules are explained below:

1. **DeusMain:** *DeusMain* is the main driver module. Among its main tasks are setting the spark context and establishing the Spark Session, setting the configuration file, which is provided as an argument while running the jar, connecting to

[7] https://medium.com/@HireDevOps/apache-spark-the-largest-open-source-project-in-data-processing-403c35028208

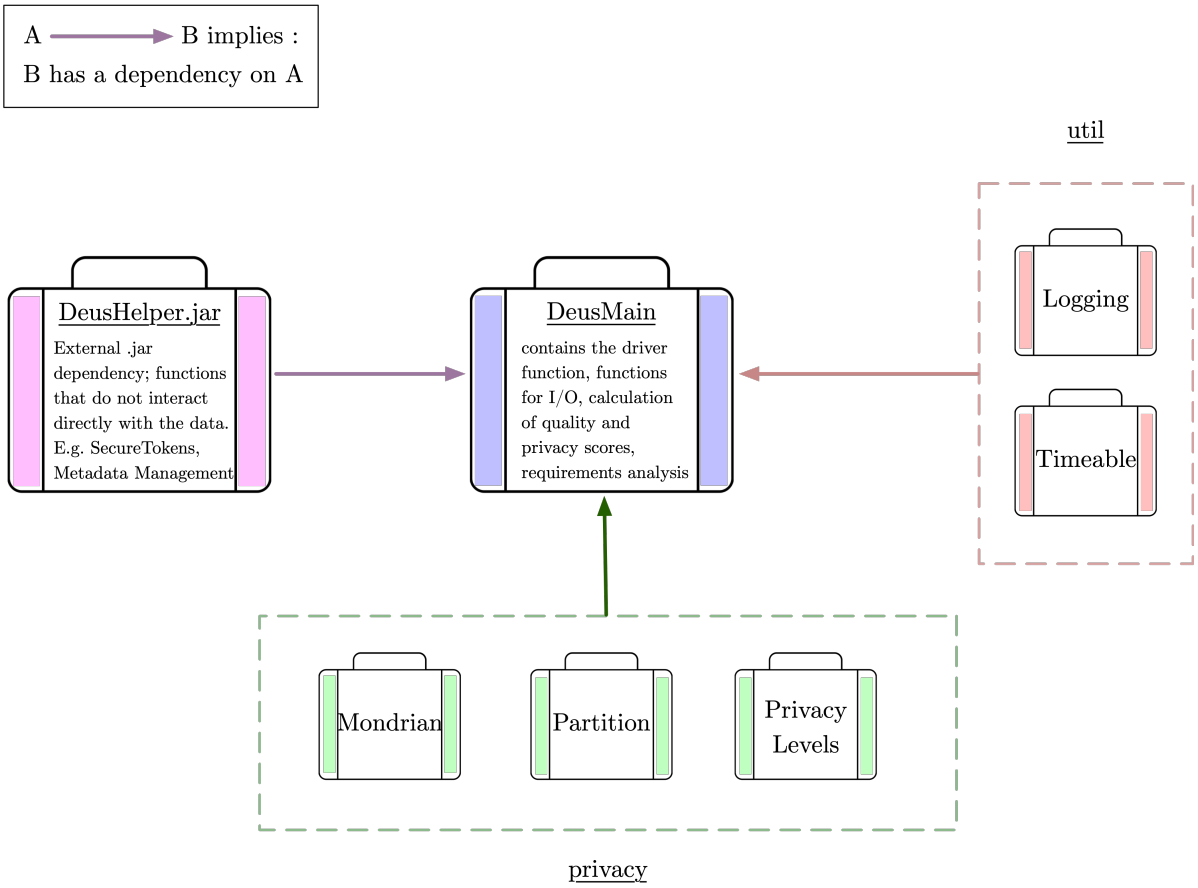

Figure 4.19: Modular Organization of the Project

the data source and converting the raw data into a DataFrame. Apart from this, it is also responsible for calculating the data quality and privacy, performing the requirements analysis and applying the privacy levels, if required. The functions for adjusting the usability using the CPF, after privacy levels have been employed, and writing the final result to a JSON file are also contained here.

2. **privacy:** The *Privacy* module contains classes for algorithms within the scope of the privacy levels, as well as the definition and application of the privacy levels themselves. Following classes/traits are present within the privacy package:
   - **Mondrian:** Implements the Mondrian k-anonymity algorithm.
   - **Partition:** Partitions the dataset greedily and recursively to yield smaller subsets which still satisfy the conditions for k-anonymity. This partition is required as a step under the Mondrian k-anonymity algorithm
   - **PrivacyLevels:** The various Privacy levels as well as the actions for each of

the levels is defined here.

3. **util:** The *util* module contains small utilities. Currently, it contains the *Logging* classes which manages logging, and the *Timeable* class which contains functions to time the execution of the code.

4. **DeusHelper:** *DeusHelper* is the part of the project which is written in *Java*. In the current deployment model, DeusHelper is typically packaged as a jar, and provided to the main application. DeusHelper mostly contains all the parts of the code which do not directly interact with the dataset. These include the definitions for *General* and *Usecase* knowledge, the result file, and the classes for parsing them from various sources *(e.g. .xlsx and .know files)*. Consequently, it also aggregates all the generated metadata from the quality and privacy evaluation, and publishes it to MongoDB. Apart from that, the tokenization algorithm as well as well the logic for encryption-decryption of the token vault is contained here.

4.7 Discussion

In this chapter, we have discussed the implementation of our approach. We have described, in a step-by-step manner, the measures that need to be followed in order to evaluate the data quality and privacy metrics of a dataset, as well as to apply suitable privacy levels in order to ensure higher privacy guarantees. Furthermore, we have also seen an approach whereby it is possible to empirically measure the effect of the applied PETs on the usability metrics of a dataset. We have also seen how the different quality and privacy dimensions are calculated, and how technologies such as *k-anonymity, tokenization, selective attribute suppression* are applied as a part of the approach. We also discussed our technologies of choice, and why we think they are best suited for a solution that is flexible, versatile, easy-to-use and is likely to be long-lasting.

In the next chapter, we will look at two real world use-case scenarios where the approach was used, and will closely analyse the results. In addition, we will also follow the results of a data quality and privacy survey conducted at Audi and RWTH to understand different perspectives on an extremely opinionated topic such as this.

5 Results and Evaluation

In Chapters 3 and 4, we have introduced *Deus*, and have seen its implementation details and design choices in a detailed manner. Since *Deus* is developed primarily for use in the industry, *effectiveness* and *ease-of-use* its must-haves. In addition, thorough considerations need to be made about the definitions of terms such as *data quality* and its dimensions, because of their inherent subjectivity. Therefore, we designed a survey on the topic, and would like to present its important findings in this chapter.

Last but not least, it is imperative that the approach should work well in practice. In this chapter, our aim, in particular, is to look at the usefulness of the quality and privacy measures, by determining how well the final scores are able to point out strengths and deficiencies of a dataset. To this end, we define and study datasets originating from two different soft use-cases to analyse the data which passes through the privacy algorithms. With this, we intend to evaluate the difference between the results of the raw and privacy-enhanced dataset in a typical data mining scenario.

5.1 Survey

We designed a *Data Quality and Privacy Survey* consisting of 10 questions. This survey was meant to be exclusive, and geared towards managers and data scientists at the AUDI AG *(data owners and users)*, as well as professors and researchers at the Databases and Information Systems group of the RWTH Aachen University *(data quality and privacy experts)*. The survey had a two-fold aim: to establish whether the definitions and methods used in this thesis conform and match-up with the respondents' expectations when it comes to data quality and privacy. The second aim was to ascertain whether the respondents believed that data quality and privacy measurement is an effective component of metadata management, and whether an effort towards such a transparent metric would be helpful and well-received in industrial data-sharing scenarios, especially in the automotive sector.

The survey was able to attract *42* respondents, and received an overall positive response. It was also able to provide us invaluable and conclusive insights on the subject,

and formed an ineludible part for the evaluation of our approach. This was crucial, especially because the importance of the opinion of the respondents its deployment and eventual use. Table 5.1 lists down the questions we asked in the survey, as well as the ultimate aim for asking them.

SNo.	**Question**	**Aim**
1	*Which is your role within in the organization?*	Determine the role of the respondent *(data owner, user or data quality and privacy expert)*
2	*What does 'good quality data' mean to you?*	Determine the most well-understood and favoured understanding of *good data quality*
3	*For the data quality dimension 'consistency', which of the following aspects do you consider as important?*	Determine whether the definition used in our approach closely matches the commonly understood meaning of the term
4	*In your opinion, what qualities should a dataset have, to be labelled as 'useful' for an internal use-case?*	Determine the most well-understood and favoured understanding of the term *usefulness*
5	*Do you agree that a dataset with high uniqueness, i.e. less number of repetitive or redundant values, has a better utility for analytics?*	Determine whether high uniqueness of values within a dataset is understood as an indicator of high data quality
6	*What are the preferable qualities of PETs?*	Understand the expectations that the respondents have, from a privacy-enhanced dataset, and whether our approach is able to ensure this
7	*How can adequate trust be established between parties in data sharing scenarios in the automotive sector?*	Understand critical motivating factors that drive trust in data-sharing
8	*According to you, what are the most important features of a highly 'interpretable' dataset?*	Determine the most commonly understood meaning of *interpretability*
9	*What would be an acceptable way of tokenizing a dataset?*	Evaluate the respondents' opinion about whether tokenization is acceptable in data-sharing, and what kinds of attributes should be tokenized
10	*What steps should be taken towards achieving 'metadata management' effectively?*	Understand whether the respondents opine that metadata management is important, and the most important steps that should be taken to ensure it

Table 5.1: Questions included in the Data Quality and Privacy Survey

Which is your role within in the organization?

Out of all the respondents, *74%* were *data users/use-case owners*, *10%* were *data owners/managers* and *17%* were *data quality or privacy experts*.

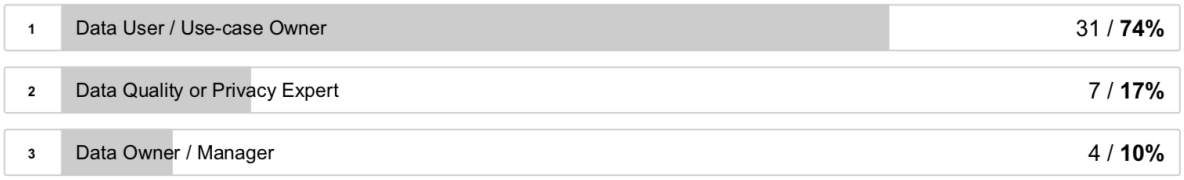

Figure 5.1: Responses of Question 1

What does 'good quality data' mean to you?

The most common answer for this question was '*data which follows the expected schema*' with *86%* of the respondents choosing this option. This was followed by '*data which doesn't have many null or incomplete values*', and '*data which is reported on time*' with *71%* and *50%* respectively. This confirms that the inclusion of the dimensions *consistency, completeness* and *timeliness* in our data quality approach matches very well with the user demographics' expectations. An interesting revelation from the answers was that only *12%* respondents chose the option '*data which other colleagues have found useful*', signifying that a departmental use-case driven data quality metric is relevant.

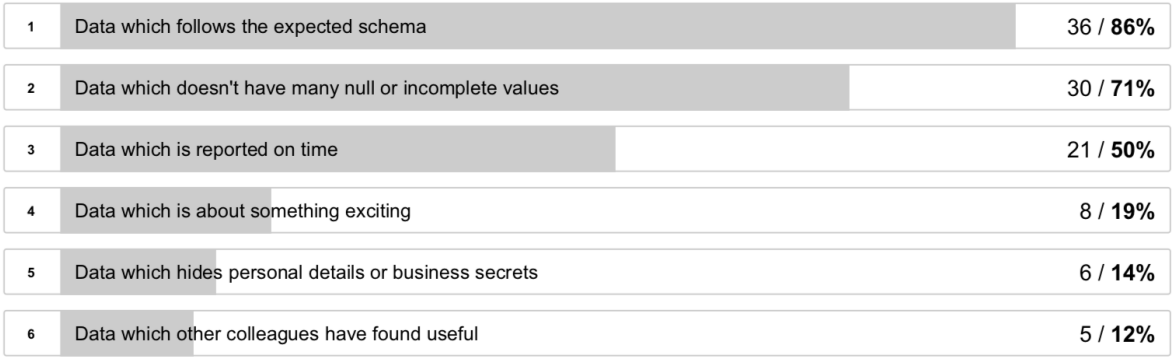

Figure 5.2: Responses of Question 2

For the data quality dimension 'consistency', which of the following aspects do you consider as important?

93% of the respondents selected '*values have the correct data type*', followed by the option '*values are within the expected range*' with *64%* votes. The importance of correct data type and range are also captured in our estimation of the *consistency* dimension. This is in line with the expectations of our respondents. The other options received less than *50%* votes.

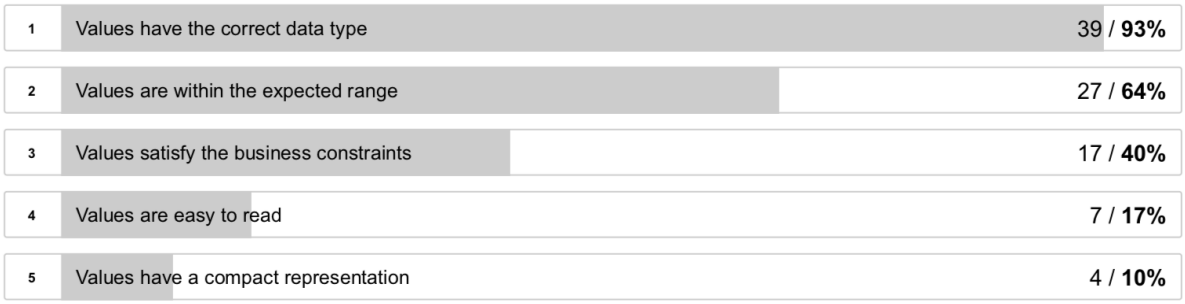

Figure 5.3: Responses of Question 3

In your opinion, what qualities should a dataset have, to be labelled as 'useful' for an internal use-case?

98% of the responded chose '*it contains all the attributes required for the use-case*', which is included in the *relevance/utility* dimension of our usability metrics. The next most popular option was '*it is time-relevant for the use-case, i.e. it has not expired*' with *76%* votes. This is taken care of by our usability dimension *volatility*.

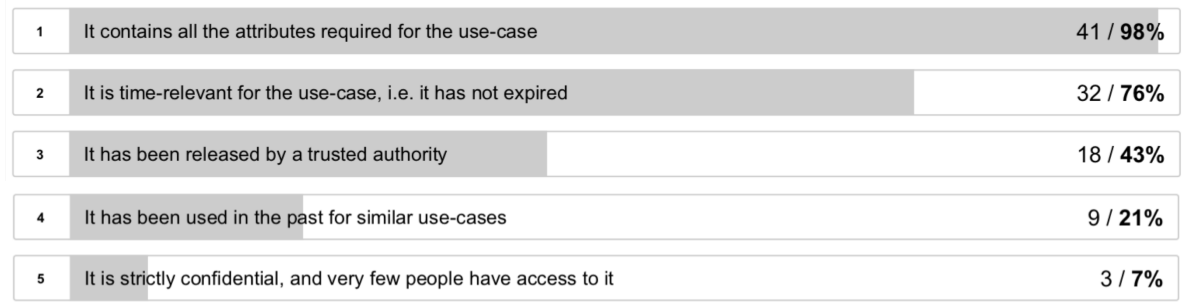

Figure 5.4: Responses of Question 4

Do you agree that a dataset with high uniqueness, i.e. less number of repetitive or redundant values, has a better utility for analytics?

The survey respondents' answers were divided on this question. The average score for all the answers was *5.6/10*. This is very interesting, because while there were a significant number of respondents who opine that high uniqueness (or high information content in information-theoretic terms) is indeed more useful for analytics, but many also went against this belief. A plausible explanation for this observation can be that the importance of high-uniqueness really comes down to the concrete use-case. E.g. in anomaly detection use-cases, which is very common in production big data, is likely to have few unique values, but is still not necessarily an indicator of poor quality. We still retained *uniqueness* as a data quality dimension, because reporting the metrics in a transparent manner gives the option to the data/use-case owner to take or not to take action.

What are the preferable qualities of PETs?

95% of the respondents said that the applied PETs should anonymize information about individuals and company secrets. *86%* believed that the resulting dataset should still be usable for data analytics. This explains the importance of techniques such as *privacy-preserving data mining*, and that of providing a suitable compromise between data quality and privacy rather than a *0 or 1* approach. *74%* respondents said that PETs should protect the dataset from linkage attacks and *67%* said that the effects of applying these PETs should be quantifiable and transparent. We try to shield the dataset from linkage attacks by tokenizing selected attributes, and also provide a way of factoring-in the effects of the PETs on usability via CPF.

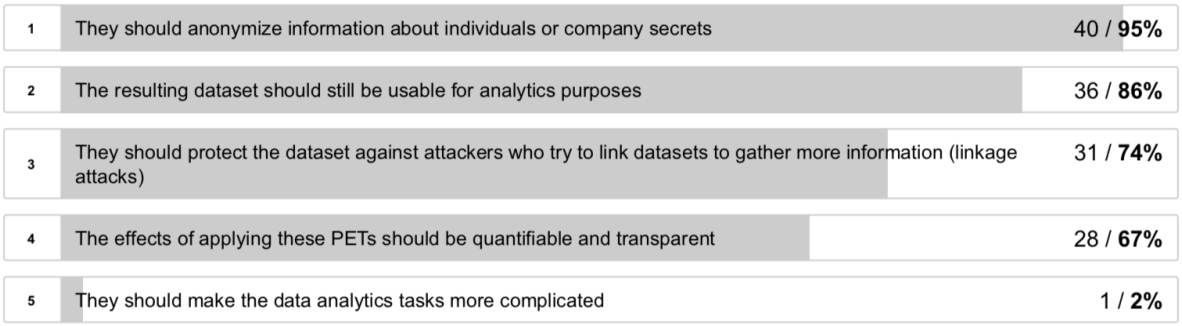

Figure 5.5: Responses of Question 6

How can adequate trust be established between parties in data sharing scenarios in the automotive sector?

The three options with the highest support were '*By maintaining a delicate balance between data privacy and usability*', '*By providing measurable quality guarantees for the released data*', each with *71%* votes and '*By being transparent about the pre-processing applied to the data before releasing*' with *67%* votes. All the three options establish the importance of measuring data quality in industry-wide data sharing or as an essential component of the data marketplace. It also demonstrates that transparency is also a very important factor that is essential for harbouring trust between parties in data sharing.

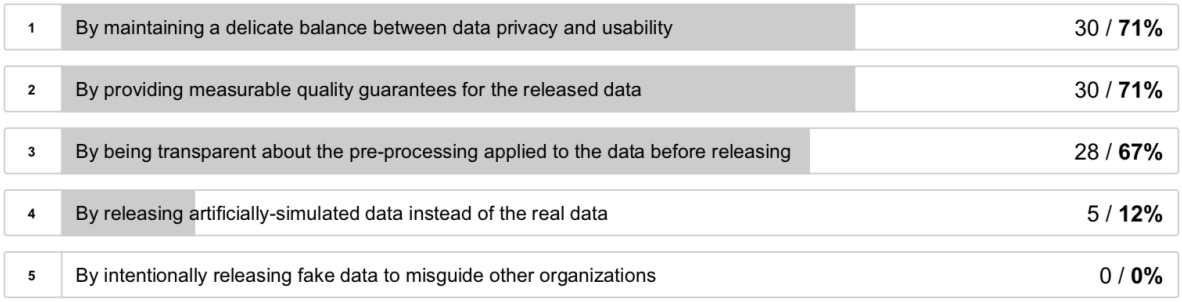

Figure 5.6: Responses of Question 7

According to you, what are the most important features of a highly 'interpretable' dataset?

79% of the respondents said that an interpretable dataset means that '*numerical values (e.g. current, voltage) should be associated with units*'. *67%* of the answers supported that '*the values should not be rendered useless by the PETs*', while *62%* agreed that values should be human-readable. Our definition of interpretability takes ensures that numerical data are associated with units, as well as the fact that the data is not suppressed by the PETs.

What would be an acceptable way of tokenizing a dataset?

79% of the respondents opine that '*only the most privacy-critical attributes should be tokenized*', while *36%* of them chose the option that '*only those attributes should be tokenized, which are not critical for the analytics task*'. The common point the answers raise is that all demographics of users in our survey agree that tokenization should be used as a tool for ensuring that linkage attacks can be prevented.

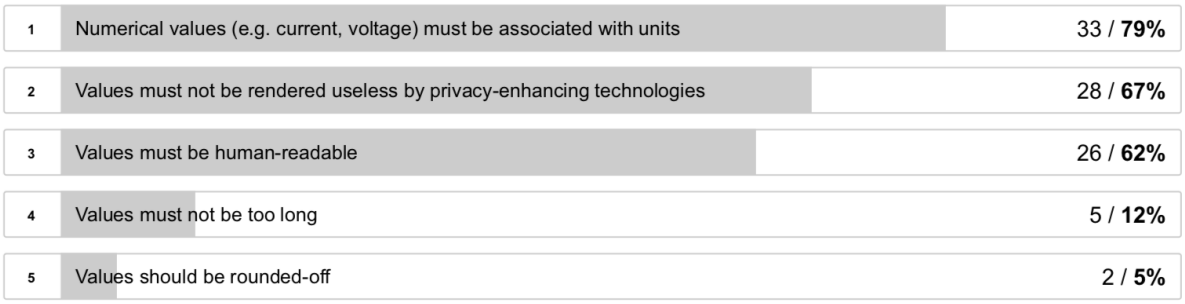

Figure 5.7: Responses of Question 8

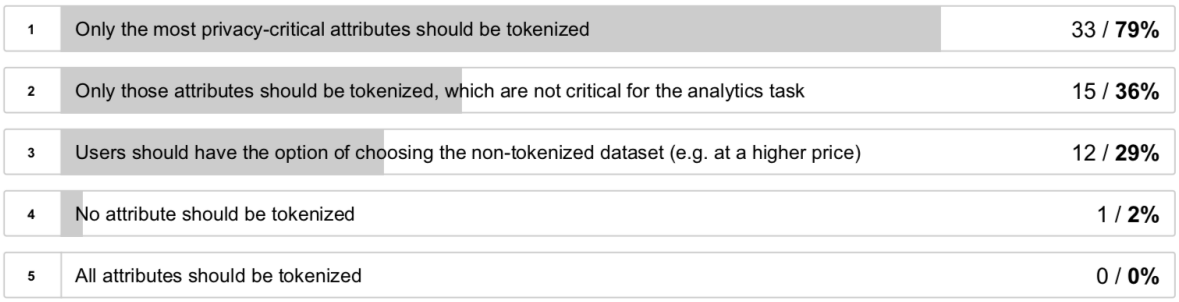

Figure 5.8: Responses of Question 9

What steps should be taken towards achieving 'metadata management' effectively?

A very encouraging *83%* acknowledge that '*metadata should be continually collected and updated*'. Out of the various ways of achieving this, opinion was split on the '*creation of an organization-wide data catalog or centralized metadata repository*', which got *69%* votes, and '*setting up a data marketplace for data bartering/sharing*', which garnered *52%* of the responses. It can be argued that *Deus* is one small step towards this big goal. With an effective approach for transparent quantification and management of data quality and privacy, we could be extremely close to fruitful industrial data exchanges in an environment of trust, confidence and mutual-benefit.

In conclusion, it would be fair to say that the most common opinions and beliefs of our demographics about the essential questions pertaining to data quality and privacy were in close consonance with the spirit of our approach. Indeed, the overwhelming support for data sharing and metadata management provides a strong motivation and raison d'être for our solution.

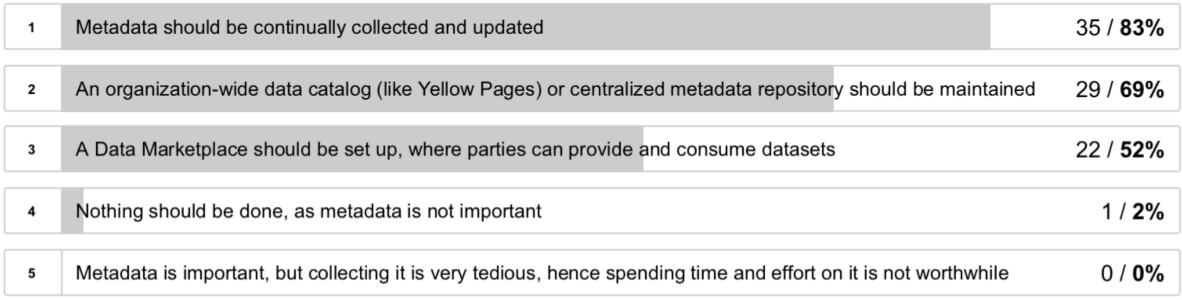

Figure 5.9: Responses of Question 10

5.2 Use-cases

To evaluate the results produced by our data quality and privacy approach, as well as the utility of those results in data mining, sharing and metadata management, we present two use-cases, which are illustrated in Figure 5.10. For the first use-case, we put *data quality* into perspective without applying any PETs on the dataset. Thereafter, we try to analyse the results produced by our solution, and how these results aid in various stages of the data mining process such as data pre-processing, data transformation etc.

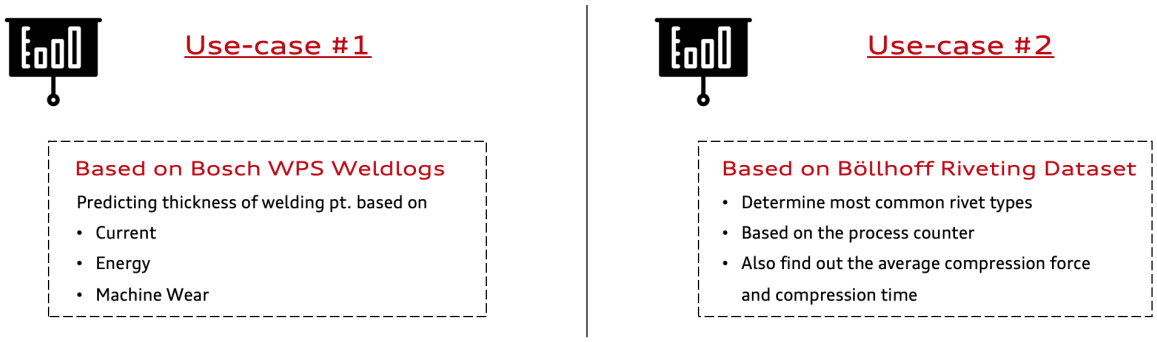

Figure 5.10: Use-cases for Evaluation

For the second use-case, we try to apply the three privacy levels which are described in 4.4.2, and compare the data quality metrics for all the cases. Our aim is to establish whether a trend or relationship exists for the data quality and privacy scores for the different privacy levels. We will also simulate a typical data mining task by designing a 'bag of queries', which will be evaluated against the raw and privacy-enhanced datasets to evaluate the deviation in the results. We evaluate whether the difference in data quality scores between the privacy levels is a good indicator of the deviation in real-world data mining results.

5.2.1 Bosch Weldlog Dataset

The *Bosch Weldlog Dataset* originates from the Bosch welding machines in the body shops of Audi's Neckarsulm site. The dataset consists of *15,000* records, and the attributes present in it are *(datetime, timestamp, machine, current, energy, thickness, wear)*, as shown in Table 5.2. Our data analytics use-case for this dataset is 'predicting the thickness of a welding point' based on the current, energy and machine wear.

Attribute	**Data Type**	**Privacy Sensitivity**	**Minimum Value**	**Maximum Value**	**Unit**
datetime	timestamp	non-sensitive	N.A.	N.A.	N.A.
timestamp	timestamp	non-sensitive	N.A.	N.A.	N.A.
machine	string	key-attribute	N.A.	N.A.	N.A.
current	numeric	sensitive	0.0	10.0	ampere
energy	numeric	sensitive	0.0	10000.0	joule
thickness	numeric	sensitive	0.0	10.0	centimetre
wear	numeric	sensitive	0.0	100.0	percentage

Table 5.2: Important Attribute Properties from the Knowledge File

Our specific aim for studying this dataset is to determine whether the insights generated from data quality assessment are crucial for our aforementioned data mining task. For this, we proceed step-wise with our methodology as illustrated in Section 3.3. First, we collect the knowledge files from the data owner and use-case owner, and run our application to calculate the data quality and privacy metrics. The final scores which are retrieved from the application are shown in Figure 5.11.

On a first glance, a moderate total data quality score of *0.748* conveys that this dataset is moderately well-suited for the use-case. On dissecting further, we find that *fitness* and *usability* scores are *0.775* and *0.721* respectively. The privacy score, however, is an abysmal *0.286*, conveying that the dataset, in its current form, is not well suited for sharing with external partners.

One of the strong points of *Deus* is its capability to provide a very fine-grained analysis of scores corresponding to each dimension within fitness, usability and privacy. In case of fitness, attribute-wise analysis is also possible, as shown in 5.12.

This gives us the potential to single out and eliminate errors in data generation/collection, or estimate the effort required in data cleaning. In this case, the data quality measures pointed out some glaring errors in the dataset which would otherwise not have been noticed. A few of these observations that were made in process of analysing the data quality scores are:

- The *current* attribute surprisingly displayed a completeness score of *0.906*. This

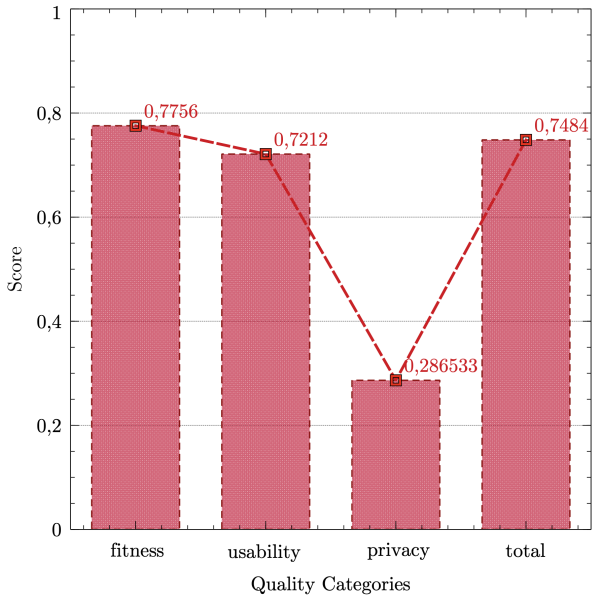

Figure 5.11: Final Data Quality and Privacy Scores of the *Bosch Weldlog Dataset*

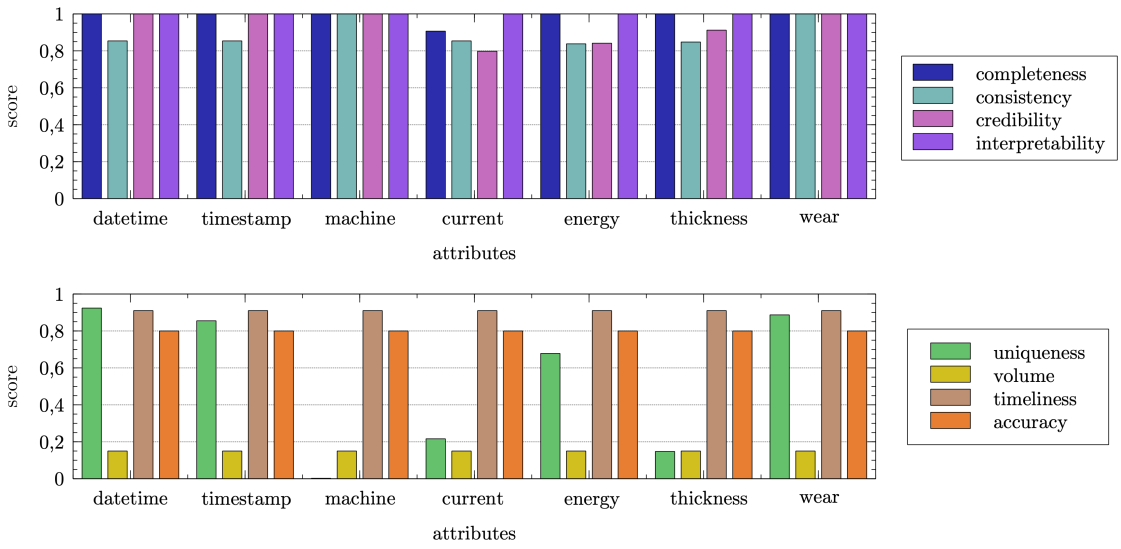

Figure 5.12: Fitness scores of the *Bosch Weldlog Dataset*

means that almost *10%* of the records were either *null* or *missing*. The reason was later identified as a parsing problem in the middleware software, due to which each of the attributes were shifted to the right, and the first attribute, which happened to be *current*, was assigned *null*. Let us call this the *attribute-shift problem*.

- The attribute-shift problem also leads to other attributes like *datetime* and *times-*

tamp getting average consistency scores. This is because they end up being populated by *float* values from other attributes, which is inconsistent with their expected data type *(timestamp)*.

- The low uniqueness and credibility score of *current* was also interesting, and we were able to pinpoint that many records were simply reported as the default value *(0.0)*. This was later attributed to a sensor malfunction, and was rectified thereafter.

- The low uniqueness score *(0.0024)* of *machine* was due to the fact that only one machine was regularly publishing data to the MQTT broker.

These observations are significant, because if undiscovered, they would have led to faulty modelling and prediction. Transparent reporting of the strengths and weaknesses of the dataset is essential for weeding out problems before the dataset can be used for critical tasks. Among the dataset's strengths are its perfect interpretability score of *1.0*, which is a testimony to the fact that all of its numeric attributes are associated with units of measurement, and no attributes are tokenized or suppressed.

When we look at the dataset's usability and privacy scores in Figure 5.13, we can immediately point out that the main culprits driving the usability score downwards are *utility(0.428)* and *volatility(0.6)*. The utility score can be attributed to the fact that only a fraction of the attributes will actually be used for the data mining task, while the volatility score is representative of the fact that the dataset remains valid for the use-case only for *one month*.

With the exception of *sensitivity*, the other two privacy dimensions are close to zero. Since none of the key or quasi attributes are tokenized, it leaves the dataset vulnerable to linkage attacks, and there is a high risk of re-identification and disclosure of sensitive information, due to the low uniqueness of the key-attribute*(machine)*. The overall privacy analysis would suggest that if this dataset is to be shared, PETs must be applied to it so that the privacy characteristics can be improved, and more privacy guarantees can be provided.

Therefore, in the examples and justifications that we saw above, we can agree that a well-categorized and granular assessment of data quality and privacy can lead to an 'opening-up' of the dataset, and characteristics that would otherwise be invisible in cleaning and pre-processing steps suddenly become visible. This has several advantages: it helps us in deciding whether a dataset is well-suited for the analytics task, aids the

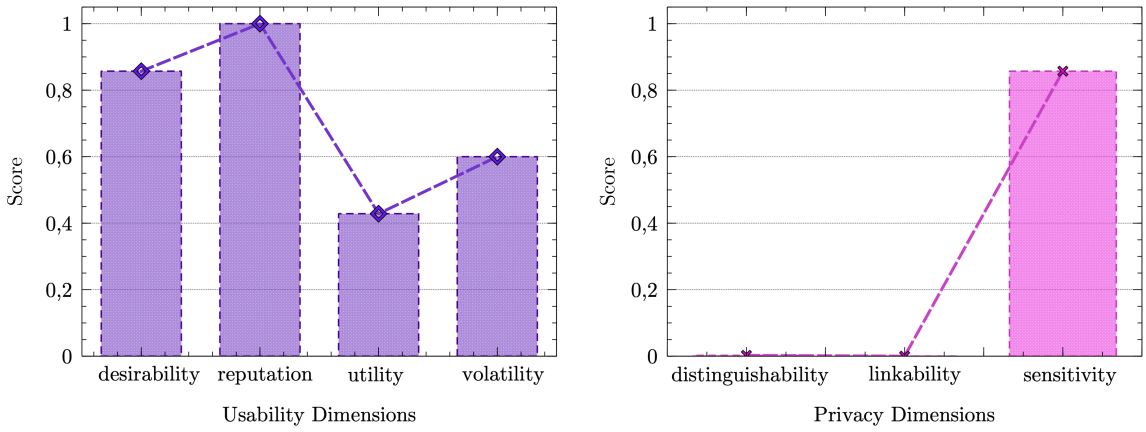

Figure 5.13: Usability and Privacy scores of the *Bosch Weldlog Dataset*

pre-processing steps in a major way and helps us understand the results of the machine learning algorithms in a more insightful manner.

5.2.2 Böllhoff Riveting Dataset

The *Böllhoff Riveting Dataset* is a process dataset originating from Böllhoff riveting robots in the body shops. It contains logs of the riveting joints, and consists of a total of *104* attributes and *34,118* records. For simplicity in understanding and illustrating the resulting metrics, we pick *15* essential features from the dataset for our use-case, which are shown in 5.3. We would like to answer a few queries about the most common *rivet types* occurring in the dataset, as well as the average *compression force* and *compression time*.

Through this use-case, we would like to show how the privacy levels interact with the data quality metrics, to ensure a granular control over data quality and privacy. We also illustrate that with carefully crafted privacy-enhancing technologies, it is possible to ensure a high usability of the data without compromising with its privacy. To achieve this, we have formulated *8* queries for the *bag of queries* model. *5* of these queries *(1-5)* are general *data analytics* queries, which are genuine queries that form a part of the use-case. On the other hand, *3* queries *(6-8)* are *private* queries which simulate a data user who wants to gather private information about the organization. We show how the various privacy levels offer a varying degree of protection against these queries, while still remaining usable for the queries from the data analytics task.

The queries for the use-case are as follows:

Attribute	Data Type	Privacy Sensitivity	Minimum Value	Maximum Value	Unit
sourceTimestamp	timestamp	key-attribute	N.A.	N.A.	N.A.
serverTimestamp	timestamp	key-attribute	N.A.	N.A.	N.A.
processCounter	string	quasi-identifier	N.A.	N.A.	N.A.
joiningPoint Description	string	quasi-identifier	N.A.	N.A.	N.A.
joiningPointName	string	quasi-identifier	N.A.	N.A.	N.A.
rivetType	numeric	quasi-identifier	0	10	unitless
rivetLength	numeric	sensitive	0.0	10.0	millimetre
contactForce	numeric	sensitive	0.0	10.0	newton
contactTime	numeric	sensitive	0.0	10.0	nanoseconds
preClampingForce	numeric	sensitive	0.0	10.0	newton
preClampingTime	numeric	sensitive	0.0	10.0	nanoseconds
compressionForce	numeric	sensitive	0.0	10.0	newton
compressionTime	numeric	sensitive	0.0	10.0	nanoseconds
processControl Relative	boolean	non-sensitive	N.A.	N.A.	N.A.
returnDistance	numeric	sensitive	0.0	10.0	millimetre

Table 5.3: Important Attribute Properties from the Knowledge File

1. How many distinct *joining points* are present in the dataset?
2. What is the average *compression force*?
3. What is the average *compression time*?
4. What is the most common *rivet type* in the dataset?
5. What is the most common *rivet type* for the most frequent *process counter*?

6. What percentage of riveting operations were performed after noon?
7. How many distinct counts of *process counters* are present in the dataset?
8. What is the difference between *source time* and *server time*?

We formulated these queries as a script in *Spark SQL*, and used them on the raw dataset, as well as the privacy-enhanced dataset from Level 1 up to 3. We recorded the responses in each case, as well as the deviation of the results from the actual answer, i.e., the answer provided by the raw dataset. These observations, which are listed in Table 5.4, reveal some very interesting occurrences. We are clearly able to observe a difference in the responses obtained in the different privacy levels.

For example, let us consider *Query 1*, which returns an answer of *13* from the raw dataset, whereas Levels 1-3 return *12, 10* and *(-)* respectively. The deviation from the

QNo.	Query Type	Response (Raw Dataset)	Response *(Deviation)* Level 1	Response *(Deviation)* Level 2	Response *(Deviation)* Level 3
1	Data Analytics	13	12 *(8%)*	10 *(26.08 %)*	- *(-)*
2	Data Analytics	4.71	4.71 *(0%)*	4.59 *(2.58%)*	4.16 *(12.4%)*
3	Data Analytics	2.50	2.50 *(0%)*	2.42 *(3.25%)*	2.28 *(9.21%)*
4	Data Analytics	2	2 *(0%)*	2 *(0%)*	- *(-)*
5	Data Analytics	2	2 *(0%)*	2 *(0%)*	- *(-)*
6	Private	35%	- *(-)*	- *(-)*	- *(-)*
7	Private	21,586	4214 *(134.6%)*	- *(-)*	- *(-)*
8	Private	0	- *(-)*	- *(-)*	- *(-)*

Table 5.4: Responses for the *Bag of Queries* for the Different Privacy Levels

raw dataset in Levels 1 and 2 is the result of k-anonymization of the quasi-attributes. This results in some of the joining points *(ones with low number of observations)* to be grouped together and generalized, leaving behind a lower number of distinct joining points. However, in spite of privacy-enforcement, Levels 1 and 2 are able to provide an answer to the query, albeit with a deviation. However, Level 3 is not able to provide a meaningful answer to the query at all, because both *joiningPointName* and *joiningPointDescription*, being quasi-attributes, are tokenized, rendering them unusable. The inevitable trade-off between utility and privacy is clearly on display in Table 5.4, as we observe a gradual rise in the deviation of the responses from the actual values, when the privacy-enforcement is increased.

Upon close inspection, another notable difference that can be observed, is the manner in which *data analytics* queries and *private* queries are handled. Our PETs are designed to be much more severe on key-attributes and quasi-identifiers, while retaining as much information as possible from the sensitive and non-sensitive attributes. The latter attributes are the ones which are most important for the data analytics tasks. For the aforementioned reason, the privacy levels perform much better on data analytics queries than private queries, which is fully intended. Table 5.5 summarizes the cumulative deviation for each of the query types and the various privacy levels. It can clearly be seen that only Level 1 is able to return an answer to the private queries, and that too, with a large deviation from the original answer. The other privacy levels protect our privacy-sensitive attributes to a sufficient degree, so as to avoid the disclosure of any kind of private or confidential information.

These results also hint towards a wider trend for the relationship between the *fitness*, *usability* and *privacy* characteristics of data. Figure 5.14 provides an illustration of this relationship. We can observe that with each jump in the privacy level, the privacy

Query Type	Cumulative Deviation (Level 1)	Cumulative Deviation (Level 2)	Cumulative Deviation (Level 3)
Data Analytics	1.6%	6.382%	10.805%
Private	134.6%	-	-

Table 5.5: Cumulative Deviation for the Bag of Queries Model

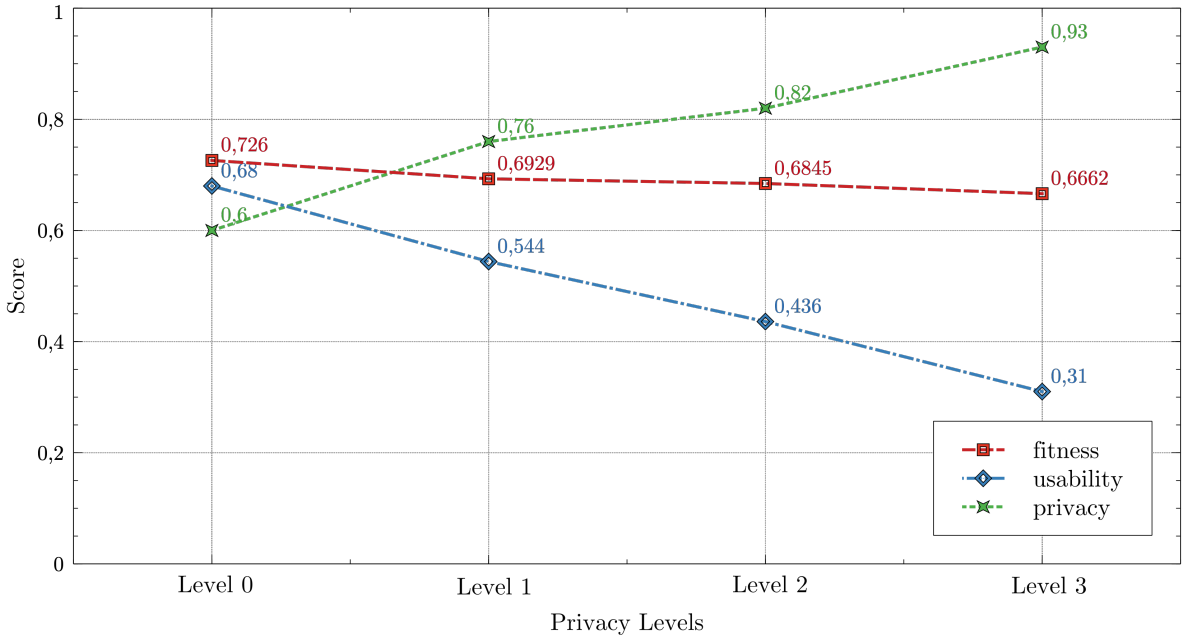

Figure 5.14: Variation of the *Fitness*, *Usability* and *Privacy* Scores for the Different Privacy Levels

score gets improved. There is a small yet noticeable difference in the fitness scores, and a more significant bump in the usability scores, as a by-product of stricter privacy guarantees. An interpretation of this trend could already be noticed in Table 5.5, where higher privacy levels yielded a higher deviation from actual results, clearly indicating diminished usability.

Table 5.6 lists down the attribute-wise fitness scores of the dataset with and without each of the privacy levels. Here, it must be pointed out that applying PETs on the dataset also leads to some unintended side-effects, which can be difficult to justify. Upon dissecting the fitness scores, one of the interesting observations, is the increase in *uniqueness* scores in the consecutive privacy levels. This can be explained as the effect of the tokenization algorithm, which assigns unique tokens to all data points, thereby increasing uniqueness. However, this doesn't necessarily contribute to an improvement of the attribute fitness in any meaningful way. Moreover, certain other dimensions such as *volume* and *accuracy* do not depend on the contents of the records themselves, and

Fitness Dimension	**Raw Dataset**	**Privacy Level 1**	**Privacy Level 2**	**Privacy Level 3**
Completeness	0.9998	0.9998	0.9998	0.9998
Consistency	0.866	0.732	0.732	0.615
Credibility	0.4018	0.3972	0.3972	0.4015
Interpretability	1.0	0.866	0.733	0.5672
Uniqueness	0.1996	0.2	0.266	0.3992
Volume	0.6388	0.6388	0.6388	0.6388
Timeliness	0.91	0.91	0.91	0.91
Accuracy	0.8	0.8	0.8	0.8

Table 5.6: Fitness Scores for the Various Privacy Levels

therefore, are unaffected by the privacy levels.

5.3 Discussion

The results of the survey as well as the two use-cases centred on production data, demonstrate that it is possible to empirically evaluate data quality to provide an estimate of the value contained a dataset. At the same time, the formulation of data privacy to go hand-in-hand with data quality enables a utilitarian study of the relationship between the two. We have also shown that if the PETs are carefully designed and implemented, it is possible to gain a granular control over data quality and privacy, which can be very useful in data sharing use-cases. Most importantly, we have been able to show from the results of the *Böllhoff dataset*, that a middle ground between privacy and utility can be achieved, and there need not be a situation where we are only able to guarantee one or the other.

6 Conclusion and Future Work

We started off on our path, with the bold aim of developing and proposing a solution for data quality and privacy which could work with diverse kinds of data originating from various domains. This wasn't necessarily straightforward because of two main reasons:

- *Data Quality is inherently subjective*, and it is difficult to come to a consensus on a set of factors or dimensions that is the best representative of 'good quality data'.
- *Data Quality and Privacy were seen as disparate concepts*, and although it is well-established that there exists a trade-off between privacy and utility, data quality and privacy have mostly been studied in isolation.

It is because of these aforementioned reasons that this work required much more than just a deep technical understanding of the subject, or a detailed study into the previously used approaches. An essential factor was to soak-in the experience and grievances of people in large organizations, who deal with large quantities of data on a day-to-day basis. This requires a huge deal of patience, time and persistence, and was indeed the most time-intensive part of this thesis.

However, once the important dimensions for the evaluation of data quality and privacy could be established, we could engage in the next steps, which were to calculate these dimensions and deliver a sturdy, reliable and robust measure of data quality and privacy. As a part of our approach, we were able to successfully accumulate essential metadata about the datasets in the form of *knowledge files*, and use this crucial piece of information to draw inferences about data, which would otherwise not have been possible. We were also able to employ and evaluate our approach on two different process datasets. This resulted in the identification of several errors and deficiencies in the dataset, which could be promptly acted upon. It is our firm belief, that if used in its intended manner, *Deus* can not only be used to evaluate the suitability of data mining use-cases, but also serve as a lighthouse for new projects.

We are also especially motivated by the sterling progress on the *Industrial Data Space*[Ott+16] in Germany, which made us coalesce the two seemingly disparate concepts of *quality* and *privacy*, to analyse how they affect one another. In the wake of

the GDPR [1] and growing privacy concerns, we realized that privacy would be of great interest in the topic of data sharing. Therefore, we wanted to illustrate that if our privacy-enhancing technologies are suitably designed and evaluated, it is indeed possible to achieve both utility and privacy.

Our efforts to create a centralized metadata repository to augment the already existing big data sinks in large organizations and support data analytics use-cases, takes us one step closer to the vision of a broad-based organizational data catalogue. These efforts are certainly in the right direction, as the results of our survey questions have also shown (*see Figure 5.9*). For a wider adoption of big data quality measures, more and more people need to be sensitized towards the importance of having 'good data', not just 'more data'. Our data quality measure could serve as a mirror in this scenario, providing encouragement for more use-cases to be built around good quality data, and the awareness to improve data which is found to be lacking. If used effectively, this could potential lead to a general improvement in the quality of data that is collected and used within an organization, ultimately resulting in much better data-driven decision making.

It is estimated that data marketplaces will unlock more than $3.6 trillion in value by the year 2030[2]. As we speak, new forums for data exchange and trading such as Iota[3] are coming up, which have the potential to revolutionize the way in which data is organized and shared. These kinds of data marketplaces can also benefit significantly from a transparent approach for data quality and privacy evaluation. This would make sure that the empirical value of the resource *(i.e. data)* is known to all parties, before a transaction is made.

In spite of its strong points, *Deus* still comes with its set of open issues, which can be suitably addressed in future work. To start off, the process of collecting knowledge files manually is a tedious task, and can be replaced by a more sustainable option. It can be observed that organizations mostly collect similar kinds of data on a day-to-day basis, and this doesn't change dramatically over a short period of time. With the help of a semantic knowledge base about the different kinds of data that are collected, we can do away with the manual entry of general and use-case knowledge for each dataset. Moreover, *Deus* currently provides no inbuilt method for visualizing the results in the form of dashboards. This can also be a very useful addition to the approach, which will ensure that it is more user-friendly. Last but not least, the full potential of the application software will be unleashed when different organizations tailor the approach

[1] https://eugdpr.org/
[2] https://www.accenture.com/us-en/insights/high-tech/dawn-of-data-marketplace
[3] https://data.iota.org/

to meet their own requirements. In principle, everything from the dimensions to the privacy levels is fully configurable. In future, more studies can be performed from different domains to determine how privacy guarantees affect data quality in each case.

In conclusion, we believe that we have achieved the goals that we initially set out to fulfill. *Deus* is a modern, flexible and general-purpose approach for data quality and privacy management, which can be very influential for automatic data governance measures, especially within infrastructures that handle and control large amounts of data. We are confident that *Deus* will be found suitable not only in the field of automotive big data, but across various domains that leverage big data technologies.

Bibliography

[AG17] Audi AG. *Perfektes Lackfinish*. 2017. URL: https://www.audi.com/de/unternehmen/menschen/perfect-paint-finish.html (visited on 12/18/2018).

[AlD11] Mutaz M Al-Debei. "Data warehouse as a backbone for business intelligence: Issues and challenges". In: *European Journal of Economics, Finance and Administrative Sciences* 33.1 (2011), pp. 153–166.

[Ami04] FD Amicis. "A methodology for data quality assessment on financial data". In: *Studies in Communication Sciences* 4.2 (2004), pp. 115–137.

[AS00] Rakesh Agrawal and Ramakrishnan Srikant. "Privacy-preserving data mining". In: *ACM Sigmod Record*. Vol. 29. 2. ACM. 2000, pp. 439–450.

[Bat+09] Carlo Batini et al. "Methodologies for data quality assessment and improvement". In: *ACM computing surveys (CSUR)* 41.3 (2009), p. 16.

[BS08] Justin Brickell and Vitaly Shmatikov. "The cost of privacy: destruction of data-mining utility in anonymized data publishing". In: *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM. 2008, pp. 70–78.

[BS16] Carlo Batini and Monica Scannapieco. *Data and information quality: dimensions, principles and techniques*. Springer, 2016.

[CHT09] Alok Kumar Choudhary, Jenny A Harding, and Manoj Kumar Tiwari. "Data mining in manufacturing: a review based on the kind of knowledge". In: *Journal of Intelligent Manufacturing* 20.5 (2009), p. 501.

[Cli+02] Chris Clifton et al. "Tools for privacy preserving distributed data mining". In: *ACM Sigkdd Explorations Newsletter* 4.2 (2002), pp. 28–34.

[CZ15] Li Cai and Yangyong Zhu. "The challenges of data quality and data quality assessment in the big data era". In: *Data Science Journal* 14 (2015).

[Dal17] Alexandra Dalevskaya. "Data Quality Management for Data Lake Systems". MA thesis. Germany: RWTH Aachen University, 2017.

[DD13] Thomas H Davenport and Jill Dyché. "Big data in big companies". In: *International Institute for Analytics* 3 (2013).

[Del] Deloitte. *Big Data Analytics in the Automotive Industry*. URL: `https://www2.deloitte.com/content/dam/Deloitte/uk/Documents/manufacturing/deloitte-uk-automotive-analytics.pdf` (visited on 12/18/2018).

[Dwo08] Cynthia Dwork. "Differential privacy: A survey of results". In: *International Conference on Theory and Applications of Models of Computation*. Springer. 2008, pp. 1–19.

[Eck02] Wayne W Eckerson. "Data quality and the bottom line: Achieving business success through a commitment to high quality data". In: *The Data Warehousing Institute* (2002), pp. 1–36.

[EGL85] Shimon Even, Oded Goldreich, and Abraham Lempel. "A randomized protocol for signing contracts". In: *Communications of the ACM* 28.6 (1985), pp. 637–647.

[EM02] Martin J Eppler and Peter Muenzenmayer. "Measuring Information Quality in the Web Context: A Survey of State-of-the-Art Instruments and an Application Methodology." In: *IQ*. Citeseer. 2002, pp. 187–196.

[Eng99] Larry P English. *Improving data warehouse and business information quality: methods for reducing costs and increasing profits*. Vol. 1. Wiley New York, 1999.

[Fir+16] Donatella Firmani et al. "On the meaningfulness of big data quality". In: *Data Science and Engineering* 1.1 (2016), pp. 6–20.

[Gei17] Sandra Geisler. "A Systematic Evaluation Approach for Data Stream-based Applications". PhD thesis. Germany: RWTH Aachen University, 2017.

[GWQ11] Sandra Geisler, Sven Weber, and Christoph Quix. "An ontology-based data quality framework for data stream applications". In: *16th International Conference on Information Quality*. 2011, pp. 145–159.

[HGQ16] Rihan Hai, Sandra Geisler, and Christoph Quix. "Constance: An intelligent data lake system". In: *Proceedings of the 2016 International Conference on Management of Data*. ACM. 2016, pp. 2097–2100.

[HMS01] David J Hand, Heikki Mannila, and Padhraic Smyth. *Principles of data mining (adaptive computation and machine learning)*. MIT press Cambridge, MA, 2001.

[HNP09] Alon Halevy, Peter Norvig, and Fernando Pereira. "The unreasonable effectiveness of data". In: *IEEE Intelligent Systems* 24.2 (2009), pp. 8–12.

[Jar+13] Matthias Jarke et al. *Fundamentals of data warehouses*. Springer Science & Business Media, 2013.

[KW03] J Kim and W Winkler. "Multiplicative noise for masking continuous data". In: *Statistics* (2003), p. 01.

[LDR06] Kristen LeFevre, David J DeWitt, and Raghu Ramakrishnan. "Mondrian multidimensional k-anonymity". In: *Data Engineering, 2006. ICDE'06. Proceedings of the 22nd International Conference on*. IEEE. 2006, pp. 25–25.

[Lee+02] Yang W Lee et al. "AIMQ: a methodology for information quality assessment". In: *Information & management* 40.2 (2002), pp. 133–146.

[LLV07] Ninghui Li, Tiancheng Li, and Suresh Venkatasubramanian. "t-closeness: Privacy beyond k-anonymity and l-diversity". In: *Data Engineering, 2007. ICDE 2007. IEEE 23rd International Conference on*. IEEE. 2007, pp. 106–115.

[Mac+06] Ashwin Machanavajjhala et al. "l-diversity: Privacy beyond k-anonymity". In: *Data Engineering, 2006. ICDE'06. Proceedings of the 22nd International Conference on*. IEEE. 2006, pp. 24–24.

[MV17] Ricardo Mendes and João P Vilela. "Privacy-Preserving Data Mining: Methods, Metrics and Applications". In: *IEEE Access* 5 (2017), pp. 10562–10582.

[NP01] Moni Naor and Benny Pinkas. "Efficient oblivious transfer protocols". In: *Proceedings of the twelfth annual ACM-SIAM symposium on Discrete algorithms*. Society for Industrial and Applied Mathematics. 2001, pp. 448–457.

[NS08] Arvind Narayanan and Vitaly Shmatikov. "Robust de-anonymization of large sparse datasets". In: *Security and Privacy, 2008. SP 2008. IEEE Symposium on*. IEEE. 2008, pp. 111–125.

[Oph+16] Albert Opher et al. "The Rise of the Data Economy: Driving Value through Internet of Things Data Monetization". In: *IBM Corporation: Somers, NY, USA* (2016).

[Ott+16] B Otto et al. "Industrial data space: digital souvereignity over data". In: *Fraunhofer White Paper* (2016).

[PLW02] Leo L Pipino, Yang W Lee, and Richard Y Wang. "Data quality assessment". In: *Communications of the ACM* 45.4 (2002), pp. 211–218.

[RB97] Thomas C Redman and A Blanton. *Data quality for the information age*. Artech House, Inc., 1997.

[Red98] Thomas C Redman. "The impact of poor data quality on the typical enterprise". In: *Communications of the ACM* 41.2 (1998), pp. 79–82.

[SC02] Monica Scannapieco and Tiziana Catarci. "Data quality under a computer science perspective". In: *Archivi & Computer* 2 (2002), pp. 1–15.

[Sca+04] Monica Scannapieco et al. "The DaQuinCIS architecture: a platform for exchanging and improving data quality in cooperative information systems". In: *Information systems* 29.7 (2004), pp. 551–582.

[SRP13] Lalitha Sankar, S Raj Rajagopalan, and H Vincent Poor. "Utility-privacy tradeoffs in databases: An information-theoretic approach". In: *IEEE Transactions on Information Forensics and Security* 8.6 (2013), pp. 838–852.

[SS98] Pierangela Samarati and Latanya Sweeney. *Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression*. Tech. rep. Technical report, SRI International, 1998.

[Swe02] Latanya Sweeney. "k-anonymity: A model for protecting privacy". In: *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 10.05 (2002), pp. 557–570.

[Tal13] Paul P Tallon. "Corporate governance of big data: Perspectives on value, risk, and cost". In: *Computer* 46.6 (2013), pp. 32–38.

[Ter+15] Ignacio G Terrizzano et al. "Data Wrangling: The Challenging Journey from the Wild to the Lake." In: *CIDR*. 2015.

[VZ19] Andre Calero Valdez and Martina Ziefle. "The users' perspective on the privacy-utility trade-offs in health recommender systems". In: *International Journal of Human-Computer Studies* 121 (2019), pp. 108–121.

[Wan+10] K Wang et al. "Privacy-preserving data publishing: A survey on recent developments". In: *ACM Computing Surveys* (2010).

[Wan98] Richard Y Wang. "A product perspective on total data quality management". In: *Communications of the ACM* 41.2 (1998), pp. 58–65.

[WBI17] Ye Wang, Yuksel Ozan Basciftci, and Prakash Ishwar. "Privacy-utility tradeoffs under constrained data release mechanisms". In: *arXiv preprint arXiv:1710.09295* (2017).

[WS96] Richard Y Wang and Diane M Strong. "Beyond accuracy: What data quality means to data consumers". In: *Journal of management information systems* 12.4 (1996), pp. 5–33.

[XT06a] Xiaokui Xiao and Yufei Tao. "Anatomy: Simple and effective privacy preservation". In: *Proceedings of the 32nd international conference on Very large data bases*. VLDB Endowment. 2006, pp. 139–150.

[XT06b] Xiaokui Xiao and Yufei Tao. "Personalized privacy preservation". In: *Proceedings of the 2006 ACM SIGMOD international conference on Management of data*. ACM. 2006, pp. 229–240.

[ZPG13] Arkady Zaslavsky, Charith Perera, and Dimitrios Georgakopoulos. "Sensing as a service and big data". In: *arXiv preprint arXiv:1301.0159* (2013).

[ZZY03] Shichao Zhang, Chengqi Zhang, and Qiang Yang. *Data preparation for data mining*. 2003.