DEPLOYING CLOUD TECHNOLOGY TO GENERATE BUSINESS INTELLIGENCE WITH CONSIDERATION TO MODERN SECURITY AND PRIVACY FRAMEWORKS

Author: Dale Stephenson

Student ID: 13502967

Bachelor of Information Technology

PRJ701

Year: 2021

Project Journal: https://d-stephenson.github.io/categories#PRJ701

ACKNOWLEDGMENTS

Throughout this research report and work placement, a great deal of support and assistance has been provided. I cannot begin to express my thanks to my partner Rebecca and two boys Tate and Brock, for whom I am ultimately doing this, without their encouragement, patience, and understanding this would not have been possible. I am deeply indebted to you all. I would like to extend my deepest gratitude to the company that offered me the opportunity to conduct this work placement, I am truly grateful for their trust and unwavering commitment to the project. I would also like to thank my mentor Sandra Dyke, for her consistently positive attitude, genuine interest in the project and valuable guidance. Although very different, the contributions from all these people cannot be underestimated.

ABSTRACT

Gaining meaningful business intelligence from disparate data sources requires a data warehousing solution that is capable of meeting business requirements and acting as a single source of truth. However, in light of recent global regulatory reforms surrounding the privacy of individuals personal data, and organisational demands for industry recognised security accreditation, the scope of these business requirements has become increasingly complex. As cloud providers have become the de-facto solutions to meeting these demands, the complexity has been exacerbated. These factors have a significant impact on the decision-making process and design of the data warehouse architecture deployed by the company commissioning this project, to ensure the company is best placed to meet its contractual obligations to clients and partners. The goals outlined encompass the research, analysis, and testing that has been conducted to support the data team, forming the content of this report and the physical implementation of the data architecture.

The research objectives have led to a set of results that can support data pipeline development, highlighting several methods that can be deployed to produce a successful outcome. Although conclusions have been drawn for this project development, the findings indicate that not one size fits all where data warehousing is concerned. This is particularly true with regards to the service providers deployed in the data pipeline and the methods chosen for the protection and security of individuals' data. Choices must be based on the identified requirements within a given organisation. The results of this report can be used to provide clarity over the processes involved in the implementation of a data warehouse solution that is capable of providing meaningful analytics in organisations that take their security and privacy commitments seriously. The report outlines the reasons why IT professionals must become familiar with these frameworks to ensure that their efforts do not fall foul of the encroaching regulatory burden placed on the IT sector.

TABLE OF CONTENTS

TABLE OF FIGURES

TABLE OF TABLES

1. INTRODUCTION

A quality single source of truth is essential for improved decision making by business stakeholders, a data warehouse is intended to act as this source of trust within businesses (Park, 2005). It is not uncommon for businesses to process data at levels that are insufficient to produce the consistency, and reliability necessary to make informed business decisions that are capable of achieving a competitive advantage (Park, 2005). Park (2005) states that information and systems quality standards reduce operational uncertainty and enhance the quality of decisions made, meaning the successful outcome of data warehouse implementation should not only concentrate on the success of the technical architecture, but also its ability to affect the performance of the decisions made across an organisation. This project aims to deploy a collection of IT tools that work in conjunction to transform transactional data to improve the performance of business units, measured through a process of validation, the generation of analytics, and the utilisation of tools made available to business stakeholders.

Ranjan (2008) states that business intelligence (BI) is an essential component of business success, to the extent it is capable of impacting profitability and long term viability. This project aims to solve the main area of uncertainty the company faces, a lack of quality data that allows stakeholders to identify the profitability of product lines and deals. A clear understanding of the business objectives and key metrics will be required if improved operational agility, customer retention, and decreased costs are to be realised (Hota, 2011). The company has experienced significant growth in recent years. To maintain this trajectory management stakeholders and high-level department users must have access to readily available information that can direct the business operation and support the wider business goals. The company deploys several cloud-based heterogonous data sources that simultaneously collect data capable of providing quality data insights. The analytical capabilities of these disparate systems are limited and siloed. Quality, reliable data analytics can be achieved through the implementation of a data warehouse (Ranjan, 2008).

The emergence of technology as an increasingly critical component of the modern enterprise is well understood (Ranjan, 2008), this is recognised by the company through the investment of time and financial resources allocated to this project. Park (2005) supports this decision, highlighting findings that show both historical and aggregated data improved the performance of the decision-making process.

Data warehousing as a process extends beyond the traditional requirements of the business stakeholders and users, and the technology and vendors deployed (Ranjan, 2008). The introduction of the European Unions (EU) General Data Protection Regulations (GDPR) expands the scope of individual's rights as they relate to the collection and processing of data by businesses (Tankard, 2016). Tankard (2016) expands on the directives, emphasising that they are not meant to be prescriptive in the use of technology capable of supporting compliance, instead stating that technological and operational controls must be leveraged to secure data and safeguard data subjects. The exception is the regulations specific mention of *encryption* and *pseudonymisation* as appropriate methods for the secure storing of data (Tankard, 2016). Compliance controls with the GDPR, in addition to the wider corporate responsibility regarding data privacy and security, is currently in progress at the company as evidenced by the undertaking of a SOC-2 Type-II audit. These controls have been expanded as part of this project.

The company still considers itself a start-up business, the management structure is such that it is not expected that individuals or teams have all the answers. Investigative research is expected across business departments to solve problems. As a result, this report aims to include (a) the qualitative and quantitative research undertaken to improve the knowledge of the author, to better contribute towards the efforts of the team, (b) the investigative research and analysis of tools and applications to support the successful project outcome, (c) identify the benefits that quality data has on BI, and (d) gain clarity over the impact that data regulation has on the ability to process data generate valuable insights.

2. BACKGROUND

2.1 The Company

The company brings Software as a Service (SaaS) to the world of creative content, partnering with the world's leading media channels to empower global brands with handcrafted video creative, at scale. The company operates globally with offices based across the Asia Pacific region, Europe, and North America. The company is experiencing a period of sustained growth that is projected to gain momentum, with aims to become the market leader in the sector. These goals will be achieved through the deployment of technology across the organisation, built by a growing, experienced, and highly skilled technical department. To protect company confidentiality and maintain data privacy, figures and tables produced for this report use generic headings and terms.

2.2 Problem Domain

The company is encountering challenges generating consistent business insights and reporting from the data currently available across the various SaaS applications utilised across business units. Three main problems have been identified:

1 There is sufficient data available but a lack of communication between technology platforms

2 The available data will require mapping to form relationships

3 There are insufficient constraints in data collection that would allow the company to answer the business questions

Data analytics is the greatest challenge stemming from the multiple heterogeneous data sources across the company (Bologa & Bologa, 2011). Bologa & Bologa (2011) reference anecdotal evidence that data integration consumes the most significant effort and resources, including locating, identifying, and profiling the source data to be ingested into the BI tool. As the company grows the challenges it encounters will become more pronounced, particularly as the number of data sources grows. These challenges will exacerbate the complexity and increase the level of skill required to maintain the BI solution. Data warehousing, on the surface, can appear to be a simple solution from a technical perspective, furthermore, the various extract, transform and load (ETL) integration tools can make the process far easier. Since the

introduction of the GDPR and the California Consumer Privacy Act (CCPA), the scope of storing, transforming and processing data has become more complex and difficult to navigate.

2.3 Requirements Gathering

To realise the objectives of the project the first crucial step is requirements gathering, which can impact significantly on the success of the project outcome. There are three defined areas of the requirements gathering process. These areas are (1) business process requirements, (2) technical requirements, and (3) regulatory and contractual requirements.

2.3.1 Business Process Requirements

Selecting the appropriate business process is essential to form the basis of the requirements gathering (IBM Docs, 2021). A business process does not need to relate to one department or organisational unit, more likely it will be defined by a set of related activities (IBM Docs, 2021). According to IBM Docs (2021), it can be beneficial to create a candidate list of business processes that have been identified as high priority and likely to have far-reaching operational value. If requirements are met independently, it is likely to result in data duplication, data redundancy, and create problems with quality and consistency (IBM Docs, 2021). IBM Docs (2021) do not recommend this as a solution, indicating it would be an inefficient and ineffective use of time and resources, realised through unnecessary storage and compute fees.

Identifying a single business process can be challenging given the multitude of processes within many organisations (IBM Docs, 2021). The project team have prioritised business processes based on the significance of the impact, the availability and readiness of quality data, the needs of stakeholders, and the feasibility of the business process considering its relative complexity. IBM Docs (2021) identifies the following information and metadata that should be collated:

- Business requirements
- Business processes
- Data owners and controllers
- Data source applications
- Issues of data quality
- Common terms and definitions

- Business-related metadata

An analysis of the problem domain and requirements outlined has resulted in the formulation of a set of objectives and key results (OKRs) that must be achieved for the project to be considered a success. The project's primary focus in the early development stages is the creation of a data warehouse storing high-level sales and revenue data. Several reasons have led to this decision, (1) the data is of key importance to management stakeholders, (2) it is of sufficiently high quality in its existing state to gain meaningful business insights early in the sprint cycle, and (3) the current process to deliver this information is labour intensive and time-consuming, requiring the manipulation of multiple spreadsheets and manual calculations.

The roadmap defines the scope of the project, with the objective being to gain insights across multiple heterogeneous data sources to provide stakeholders with a detailed understanding of the operation, which is not currently possible. Stakeholders require intelligence on individual deals, product lines, the performance of sales representatives and regions to improve the business model and produce greater efficiency and tailored metrics. The complexity becomes more apparent when production data is considered. Company management, finance, and sales teams must understand how much time is being spent on producing creative content, and the costs associated with those product lines. Time spent on deals must be categorised by employee type and the associated salary rate, in addition to the direct and fixed costs, this greatly increases the data sources, which will require the identification or formation of relationships.

Stakeholders from all company departments were contacted to clarify the types of questions they are currently unable to answer, the detailed overview provided has been reviewed in conjunction with the creation of a ticketing portal in Atlassian Jira Service Management (JSM), allowing data requests to be collated into one central location and used to inform and direct the data team. JSM was chosen as a result of a recently completed project that distributed the platform globally to meet the requirements of a SOC 2 Type-II audit. Form design requirements were gathered from the team members responsible for answering these questions, and fields were created to meet this use case. As the backlog of requests grows it is hoped the team can start to identify trends in the type of data most requested, and the level of detail stakeholders expect to gain.

2.3.2 Technical Requirements

Several technologies have emerged that allow cloud computing to be adopted by businesses, these technologies shape how organisations structure themselves, in particular the mission-critical area of information management (Stanoevska-Slabeva et al., 2010). There is a need to position the company in a way that allows it to adapt to constantly changing and turbulent markets. Stanoevska-Slabeva et al. (2010) emphasise the speed of change experienced by both startup companies and established corporations, in addition to the opportunities available with the adoption of infrastructure in the cloud ranging from reduced capital expenditure to increase scalability. To maintain a dynamic business environment, the company deploys cloud services for its global IT infrastructure, providing the flexibility to adapt to change quickly. This has a beneficial impact on the company's ability to maintain a highly agile operational environment that is ready to take advantage of global opportunities and improve the mobility of the workforce.

2.3.3 Regulatory and Contractual Requirements

Data processing raises regulatory concerns for the company resulting from a commitment to security and privacy of the data stored relating to clients, partners, employees, and contractors. Regulatory concerns will be considered by the data team with equal importance to the technical aspects of the solution. The preferred service providers will not simply offer the best solution at the price point. The importance of the geolocation of the stored data has not to be underestimated, for instance, the company would not meet compliance standards with the GDPR if a provider stored data in the United States resulting from the EU Court of Justice ruling that struck down the EU-US Privacy Shield. The ruling concluded that due to the US government national security laws, the country does not adequately protect EU data subjects ("EU-US Privacy Shield for Data Struck down by Court," 2020). The principles inherent to the GDPR are (*Understanding the 7 Principles of the GDPR - Blog - OneTrust*, n.d.):

1. Lawfulness, fairness, and transparency
2. Purpose limitation
3. Data minimization
4. Accuracy

5. Storage limitation
6. Integrity and confidentiality
7. Accountability

Hinely (2018) describes the broad impact the GDPR has on the technology sector, and the platforms and emerging technologies that businesses and organisations deploy. Li et al. (2019) describe the risk of non-compliance and regulatory action from authorities, whose reach extends from oversight to significant fines, both of which have the potential to cause significant reputation damage. Large firms such as Google, Facebook, and Amazon have updated their policies and procedures to meet the requirements of the regulations (Li et al., 2019). As a consequence, IT professionals should be aware of the GDPR (Tankard, 2016). The company recognises that these matters can no longer be confined to those employees and stakeholders charged with corporate governance. The GDPR further recognises this by explicitly defining organisations that would need to designate a Data Protection Officer (DPO) ("Art. 37 GDPR – Designation of the Data Protection Officer," n.d.). Currently, the scope of this role is limited to public authorities and organisations whose core data processes meet the definitions outlined in the article ("Art. 37 GDPR – Designation of the Data Protection Officer," n.d.). A DPO is not a requirement for all businesses, however, they may decide it is in their best interests ("Art. 37 GDPR – Designation of the Data Protection Officer," n.d.). There are key activities inherent to the formation of this role that shows a merging of technological and regulatory expertise. For instance, DPOs should have a strong background in privacy and security as well as IT programming, infrastructure, and information systems auditing (*Skills a Data Protection Officer Must Have*, 2021). Consideration must be made to the GDPR when large scale projects instigated by the company have the potential to pose a high risk to data subjects personally identifiable information (PII) or personal data.

Compliance with the GDPR in respect of large scale projects includes a requirement for a Data Protection Impact Assessment (DPIA), outlined under Article 35 of the GDPR and is part of the overarching principle of 'protection by design' ("Art. 35 GDPR – Data Protection Impact Assessment," n.d.). The purpose of the assessment is to account for the nature, scope, context, and purpose of data processing where the rights and freedoms of data subjects are at significant risk resulting from an organisation deploying new technologies ("Art. 35 GDPR – Data Protection Impact Assessment," n.d.). The project team must understand the legal bases for processing personal data to ensure compliance with

the principles of lawful, fair, and transparent processing. “Art. 6 GDPR – Lawfulness of Processing” (n.d.) outlines the six legal bases for the processing of PII and personal data, organisations must map their data so they are aware, and have a good understanding, of the legal justification of its collection and processing. Several conditions would trigger the requirement for a DPIA, a description of the legal basis for the collection and the triggers and compliance requirements are outlined in Appendix A. A DPIA has been conducted by the company for this project, the act prefers a DPIA to be conducted during the planning stages of the project (“Art. 35 GDPR – Data Protection Impact Assessment,” n.d.). JSM was deployed to create an audit log capable of meeting the requirements outlined in the Article.

2.4 Scrum Methodology

To achieve the project OKRs an interdisciplinary team was formed that included technical team members and employees from the finance and product teams. As the project develops it is anticipated that the team will grow to include production employees. This cross-functional team operates within a scrum development framework outlined in Figure 1.

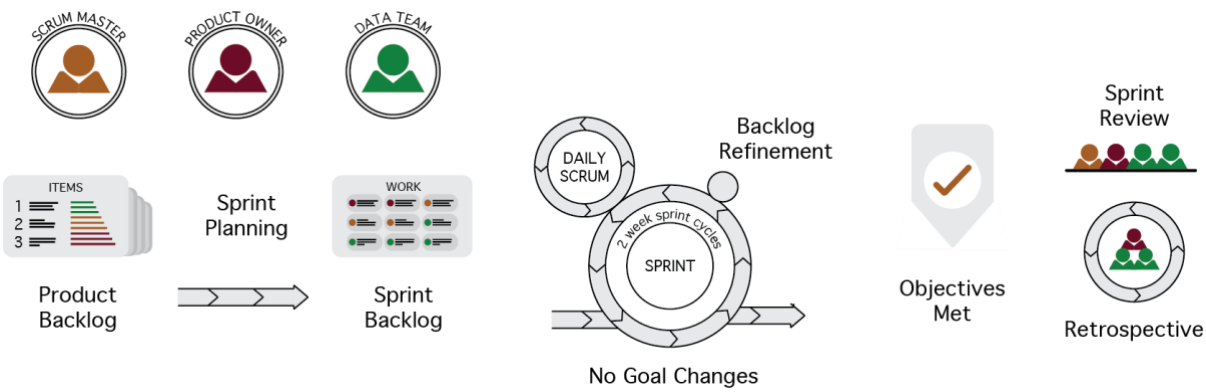


Figure 1 *The Scrum Methodology*

2.4.1 An Agile Framework

Scrum is an agile framework devised to answer questions and produce solutions to the growing complexity of challenges faced by system development teams (Deemer et al., 2012). Deemer et al. (2012) describe scrum as a methodology that allows teams to inspect and adapt through a process of learning, innovation, and surprises. Scrum is deployed across the organisation’s technical teams to develop projects

in an iterative, incremental manner through structured development work cycles known as sprints. Sprints occur consecutively in two-week cycles. The length of scrum cycles is flexible; however, they do not normally exceed four weeks (Deemer et al., 2012). The time-based nature of sprints means they will end regardless of whether the work has been completed, the project sprints are concluded with a demonstration meeting that includes the key project stakeholders (Deemer et al., 2012). Deemer et al. (2012) outline the purpose of demonstration meetings as being to inspect the progress made, discuss the successful outcomes, communicate issues and challenges, offer feedback, and provide direction for the next sprint including the priority requirements.

Decisions for forthcoming sprints are agreed upon collectively amongst the team, the targets and deliverables are intended to be achievable and produce tangible benefits to the organisation. During the sprint period, no new items can be added, focus remains on the clear and relatively stable goals outlined in the demonstration meeting. Despite this, the opportunity for change and flexibility to adapt to emerging business goals is embraced in the scrum, however, this is implemented during the preparation for the next sprint (Deemer et al., 2012). Communication and collaboration are emphasised throughout the deployment of the scrum methodology, which is achieved in daily stand-ups, the team gather to discuss updates on progress and the next steps (Deemer et al., 2012). Garzaniti et al. (2019) set the maximum duration of the daily scrum to fifteen minutes, however, in practice and due to the interdisciplinary nature of the project team and the global spread of the workforce, these tend to overrun. The focus of the stand-ups remains on the end product and a system that is analysed, designed, integrated, tested, and documented.

2.4.2 Product Backlog

The data team manages sprints with supportive tools designed to meet the requirements of a systems development environment. JSM has been selected to track progress during each sprint, allowing the team to view what other members are working on, and estimate the work remaining to plot a trajectory to reaching 'zero effort remaining' or ‘Minimum Viable Product’ (MVP) (Garzaniti et al., 2019). A product backlog documents and lists the items necessary for the project’s success and lives in the life cycle of the project, it is a single and definitive view of all the items that could be completed by the team (Deemer et al., 2012). The project backlog is continually updated, refined, and modified to reflect the emerging needs of stakeholders.

2.4.3 Sprint Backlog

The goals identified in the demonstration meeting inform the next sprint, backlog items are created and added to the sprint. Any items from the previous sprint are moved forward to the current sprint to create one planned sprint list. Items are assigned to team members and allocated a story point estimate that is based on the Fibonacci sequence; this allows for a retrospective analysis to be conducted when preparing for the next sprint (Garzaniti et al., 2019). Project items identified as a higher priority are more fine-grained and detailed, lower priority items are often broader in scope (Deemer et al., 2012).

2.4.4 Defining Roles

Deemer et al. (2012) outline the three scrum roles that form the project team. The roles are (1) Product Owner, (2) Development Team Member, and (3) Scrum Master. A detailed description of these roles is outlined in Appendix B.

3. OBJECTIVES AND RESEARCH METHODS

3.1 Team Objectives

The team objectives have been set by the Chief Technology Officer (CTO) and Senior Vice President (SVP) Data as part of the technology wide deployment of high-level OKRs. These objectives are recorded at the start of each business quarter and set the tone for the work. The data team was tasked with three distinct objectives, [TO1] develop a data pipeline capable of providing near real-time sales and revenue data for managing stakeholders and [TO2] produce sufficiently granular data from the product and production databases that allow financial stakeholders to view gross profit at the individual deal level. [TO3] The data should be available with minimal downtime and capable of manipulation on desktops and mobile devices. No additional detail has been provided to guide the decisions and choices made by the data team. Trust is placed in the skills and knowledge of individual members to organise and allocate sprint items that best meet the objectives. Although the data pipeline development is a team effort, establishing connections with the data sources and developing the analytics is an individual key result of the author [IKR1].

3.2 Research Objectives

Research objectives have been considered to support the project development and aid the decision-making process. An investigation will take place into the solutions available that are capable of meeting the project requirements and maintaining the company's technology infrastructure. The supportive research has been split into sections as follows:

RO1: Investigate the processes involved and techniques available in data mapping.

RO2: Research dimensional schema design, structural components, and the effect granularity have on the level of detail achievable.

RO3: Investigate alternative data pipeline processes, the advantages and disadvantages of an ETL process when compared with an Extract, Load, Transform (ELT) process, and how this might impact data privacy.

RO4: Research the process of data cleansing, the methods deployed for the identification of anomalies, and the criteria for ensuring data quality.

RO5: Research the methods and tools available to maintain the security and privacy of data, recognise corporate confidentiality, mitigate the risk of cyberattacks, and maintain compliance with the GDPR.

RO6: Investigate the implementation of a data warehouse, the benefits of deploying cloud solutions, and how this can be leveraged to gain valuable business intelligence.

RO7: Explore options and produce a comparative analysis of the tools available for data visualisation, research the methods that can be deployed to automate the analytics, and make them available for manipulation for non-technical users, particularly those with no Structured Query Language (SQL) knowledge.

3.3 Research Methods

Research methods encompass secondary sources that include published articles, industry-relevant textbooks, research journals, and technical documentation. The research will be gathered from online sources including Google Scholar, ProQuest, library resources, and internet search results. Primary research will be conducted through the practical deployment of technology to provide direct experience, the creation of a data request ticketing process, and interviews with important stakeholders. When combined, these methods will improve the information gathered and any conclusions drawn by providing both quantitative and qualitative research, balancing the separate strengths. The research and analysis presented in this report are structured around the processes and requirements of the data pipeline, starting with data mapping and concluding with the data visualisations as the final result output.

4. RESEARCH AND ANALYSIS

4.1 Data Mapping

Data mapping is a technique used to link values and attributes across data sources to provide meaningful insights and accurate analytics, enabling access to homogenised data for valuable insights where organisations previously had no visibility (What Is Data Mapping?, n.d.). Data analysts must map the data pipeline to reduce redundancies and replication errors, this is easier to achieve when it is possible to identify where data sources are redundant (The Essential Guide To Data Mapping, n.d.). If there is a lack of systems oversight, data analytics could become compromised leading to an over-emphasis on anomalies resulting in misrepresented results, inaccurate analysis, and reduced confidence (The Essential Guide To Data Mapping, n.d.).

The company deploys multiple tools and applications to ensure operational control and deliver a better service, these systems store data differently making consolidation of the information a challenge (What Is Data Mapping?, n.d.). Consolidating data sources will support the team's effort to create quality data capable of becoming a reliable source of truth and offering tangible value to the business.

4.1.1 Data Mapping Process

Understanding the business requirements and use cases allows tables, fields, and the data format to be accurately defined. A process of data migration can be performed that extracts the data from the source system and loads it to the destination system, completed as a one-time event (What Is Data Mapping?, n.d.). This new destination store of the migrated data becomes the source of truth (What Is Data Mapping?, n.d.). The extraction and loading of data can be performed through the deployment of a data integration process that regularly loads data to the destination system, which can be scheduled at pre-defined intervals (The Essential Guide To Data Mapping, n.d.). Regardless of the chosen method, a process of transformation is often performed that includes data cleansing (What Is Data Mapping?, n.d.). It is vital that data mapping is transparent and documented. There are many tools available that can perform this task with fewer errors and less maintenance, tracking changes as they are performed. The data mapping process should be tested to identify problems and uncover opportunities to improve the quality, and precision of the data, in the source destination.

4.1.2 Data Mapping Techniques

Data mapping techniques can vary but are generally grouped into three categories, (1) Manual, (2) Automated, and (3) Semi-Automated (The Essential Guide To Data Mapping, n.d.). A detailed description of these techniques is outlined in Appendix C. Manual data mapping has been performed for the project. There are several reasons for this decision, firstly, there is sufficient technical expertise in the data team to map the data required to answer the business questions, secondly, the data sources are well understood, and finally, the data set is comparatively small given the business-to-business product model. Automating the data mapping process may be considered as the complexity increases, a business case would need to be made given the cost implications, integration time, and training that would be necessary (The Essential Guide To Data Mapping, n.d.).

The sales and revenue system deployed at the company utilises the *representational state transfer* (REST) conventions and is designed to have a predictable URL Structure. It deploys standard `HTTP` features to return standard JavaScript Object Notation (JSON) to compare and relate data sets, these are `POST`, `GET`, `PUT` and `DELETE` (What Is REST, n.d.). REST is an architectural style for distributed hypertext systems with six guiding principles or constraints. For an interface to be referred to as *RESTful* the constraints must be satisfied (What Is REST, n.d.). To perform the data mapping a collaborative Application Programming Interface (API) client, Insomnia, was deployed that integrated with the sales software. Insomnia is an open-source API client that allows for easy and fast REST, SOAP, GraphQL, and GRPC requests, communicating directly with the source (*The API Design Platform and API Client*, n.d.). Insomnia was used to locate data relationships and identifiers as part of the data process.

4.2 Data Modelling

Data modelling is used to visualise the scope of the information systems and the connections between data points and structures, they can be used to display an entire system or parts of a system, and can include both online transactional processing (OLTP) databases and online analytical processing (OLAP) data warehouses (IBM Docs, 2021). As with database models, data warehouse models are built around the business needs and a set of business rules (IBM Docs, 2021). IBM Docs (2021) notes that the data representing the system can be modelled at various levels of abstraction through formal techniques and

standardised schemas. In the case of OLAP data warehouses, dimensional data models are used (IBM Docs, 2021). IBM Docs (2021) describes the purpose of dimensional models as being designed to optimize the speed and efficiency of SQL queries for analytics. This is achieved through the increased redundancy making data easier to locate (IBM Docs, 2021). In dimensional models, data is organised into *facts*, which are measurable data points, and *dimensions* that include information for referencing (Dimensions and Facts, n.d.).

4.2.1 Data Warehouse Schema Design

Dimensional schemas are maintained in the same way an OLTP database utilises a relational schema (Data Warehousing - Schemas, n.d.). Data Warehousing - Schemas (n.d.) outlines three commonly utilised schemas, the star schema, snowflake schema, and fact constellation schema. A detailed description of these schemas is outlined in Appendix D. The high-level business requirements for the project data requires a fact constellation schema, represented in Figure 2.

4.2.2 Data Warehouse Structure

The schema *fact* tables record one or many single measurements of real-world observations, these are almost always numeric (Dimensions and Facts, n.d.). In an organisation, they typically involve financial transactions as an atomic or single measurement, or an aggregated snapshot, or a summarised measurement (Eder et al., 2001). In the project data set, atomic values include specific order amounts represented as dollar figures and aggregated values that include the total sales for the month within a particular region. These different levels of granularity will be defined in separate *fact* tables in the data warehouse, despite containing similar measurements at different aggregation levels (Dimensions and Facts, n.d.). The benefit of multiple *fact* tables is increased flexibility when analytics are being performed, eliminating the need to recompute the different aggregation levels for each report (Kimball & Ross, 2002). In addition, *fact* tables will also contain the *foreign key* relationships with the descriptive information in the *dimension* tables, to describe the context of the fact (Kimball & Ross, 2002). For instance, a sales *fact* table would contain foreign key relationships with the customer or salesperson *dimension* tables (Dimensions and Facts, n.d.). This allows for analysis of the detail of a transaction using a filter, group, and sort, and will often store variations in the format of the data (Dimensions and Facts, n.d.). *Dimension* tables are generally wide but

relatively short, containing many columns but few rows (Dimensions and Facts, n.d.). In comparison, *fact* tables are very narrow with a limited number of columns but can grow very tall containing millions of rows. A description of the two naming standards commonly deployed in dimensional modelling is outlined in Appendix E.

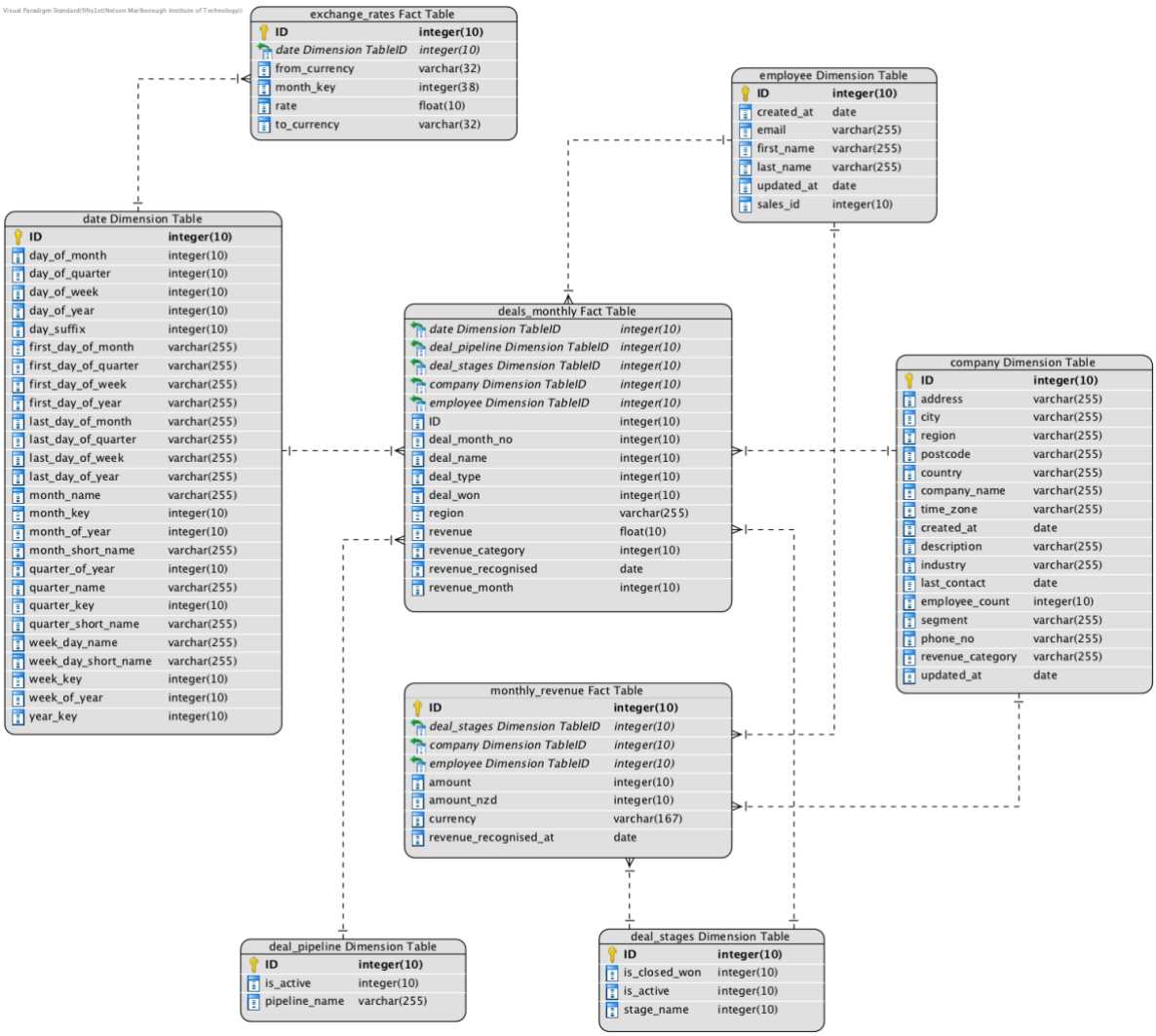


Figure 2 *Representation of the Project Dimensional Model Fact Constellation Schema*

It is common for staging tables to be created before ingestion into the dimensional schema (Kimball & Ross, (2002). The purpose of staging tables is outlined in Appendix F. The company's sales and revenue data has been loaded into a data warehouse before the transformation has occurred, acting as a staging table before being transformed. This is necessary to allow for transformation from its raw form as a data string in JSON, to comma separated format for SQL. Production data has been utilised exclusively for the business intelligence system.

4.3 Granulation

The dimensional schema design should consider the level of detail associated with each table; the granularity of the data is essential to gain the necessary insights to meet the business questions (IBM Docs, 2021). Granularity identifies the exact contents of a table record, and the level of detail that will be derived from the measurements (IBM Docs, 2021). The greater the level of detail the lower the granularity, conversely, less detail produces a higher granularity (IBM Docs, 2021). Separate *fact* and *dimension* tables can contain different levels of granularity, for instance, a date dimension table that contains *Year* and *Month*, has a granularity to the month level, but no information on weeks or days (IBM Docs, 2021). However, a date dimension table with *Year*, *Month* and *Week,* would have the granularity to the week level, but not days (IBM Docs, 2021). Therefore, the grain of a dimensional model is the finest level that can be derived as a result of joining the *fact* and *dimension* tables (IBM Docs, 2021). This is best illustrated with an example dimensional model containing Date, Sales Representative and Product, the granularity is 'the date that the products sold by which sales representative'. The findings of the requirements gathering process determine the level of granularity and the measures that must be present in the *fact* tables (IBM Docs, 2021). If the data available from the requirements gathering phase reveals there is insufficient information to produce an adequate granularity level, it might not be possible to fully answer the business questions (IBM Docs, 2021). The dimensional model for the sales and revenue data requires all financial information pertaining to the sales generated, including that of recurring revenue where the sales amount is split over several months evenly, or split over several months, in varying amounts, or with some months generating no revenue. This data has not been historically recorded in the sales application, which has led to a situation whereby the granularity is too high to answer the business questions.

4.3.1 Classification of Granulation

The schema granularity must be classified with consideration to the end-user, specifically the level necessary to provide the required service and meet many of the requirements (La Rosa et al., 2011). The approach recommended by La Rosa et al. (2011) is represented in Figure 3 and is a process of dividing the granularity into types, (a) quantity, being the amount of data available to satisfy the queries, (b) functionality, which refers to the amount of functionality provided by the data pipeline, and (c) business

value, being the tangible value added to the organisation. Further information on the granularity types is discussed in Appendix G.

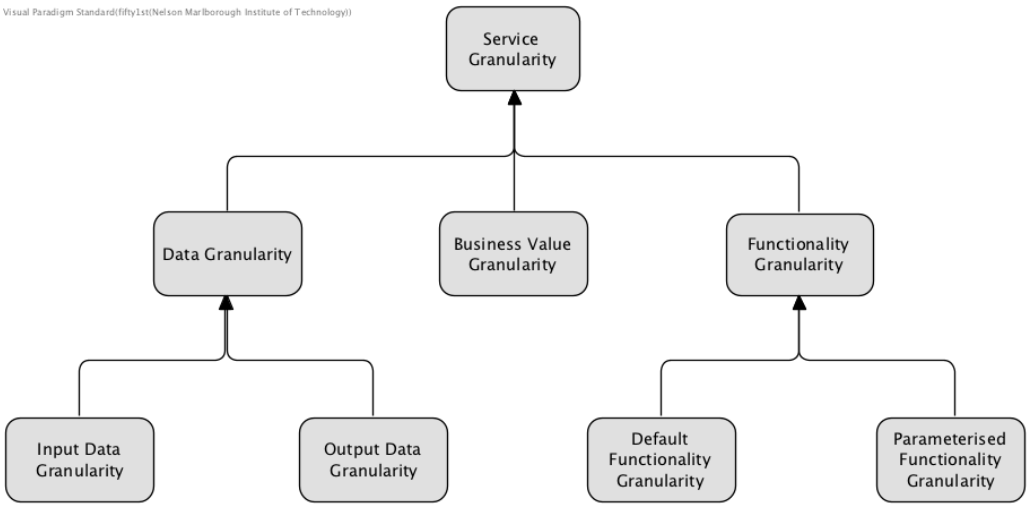


Figure 3 *Types of Granularity*

La Rosa et al. (2011) further sub-divides these granularity types. Data quantity is sub-divided into the data quantity *input* into the data warehouse, and data quantity is returned as an *output* from the query. Functionality is sub-divided into the *default* functionality granularity offered by the data warehouse, and the *parameterised* functionality granularity that can be optionality offered to satisfy the business needs (La Rosa et al., 2011). La Rosa et al. (2011) defines the business requirements as organisations seeking to gain the greatest operational and financial value possible from the services they deploy, this makes the choice of service provider a task that requires serious consideration, research, and testing to gain the highest reuse efficiency. As a result, *parameterised* granularity is considered a primary concern (La Rosa et al., 2011). However, La Rosa et al. (2011) indicates there is a limit to the benefits of moving to a coarser *parameterised* granularity. La Rosa et al. (2011) imagine a scenario where a service has been identified as capable of meeting all the business requirements, but as a consequence, it adds complexity, this complexity must be understood by the end-users to realise the benefits. In contrast, a finer grain may not offer the reuse efficiency, however, the trade-off might be more acceptable depending on the specific use case (La Rosa et al., 2011).

The company has the technical skills and expertise to implement a system with coarser *parameterised* granularity, however, given the high-level data available, the benefits of such a system may not be realised

and may lead to higher project costs. Due to the iterative nature of the project, it is recommended that a finer *parameterised* granularity is implemented, offering sufficient flexibility for change if, and when, the complexity increases. The importance of the business value granularity level should not be underestimated as it can be used as an indicator for stakeholders in the decision-making process (La Rosa et al., 2011). The ability this has to influence the decision-makers is important given the overall complexity of the system architecture being deployed, and the shared resources involved, making both implementation and modification of equal concern (La Rosa et al., 2011). For this reason, organisations will often prefer services be available in one package with the technical capability to meet the needs of end-users, rather than multiple finer-grained services that will increase overhead, specifically regarding implementation and maintenance (La Rosa et al., 2011).

4.3.2 Conflicting Granularity

Business requirements can lead to multiple grains deployed in the dimensional model; in these instances, La Rosa et al. (2011) recommends creating multiple *fact* tables, rather than attempting to make all the measures work in a single *fact* table. La Rosa et al. (2011) highlights the importance of considering the impact the schema design has on data storage capacity and performance requirements. Separating *fact* tables should also be considered in cases where OLTP databases are designed with a specific purpose, making it difficult or impossible to consolidate them into a single *fact* table (La Rosa et al., 2011). Considering the array of disparate data sources deployed across the company, this is likely to become an important design consideration. La Rosa et al. (2011) suggests that should an attempt be made to store data with different grains into one *fact* table, a column should be added that communicates the level of the grain.

4.3.3 Granular Atomicity

When determining the atomicity, or level of detail of the grain, business requirements should be considered to support future-proofing (La Rosa et al., 2011). Anticipating business needs and incorporating them into the model when choosing the atomicity level, can mitigate the risk of a model redesign as requirements change (La Rosa et al., 2011). Taking a date dimension table as an example, a business may

only require product sales to the month level to meet current business needs, however, the data should also be available by *week* and *day* so this can be made available to the business in the future.

La Rosa et al. (2011) outline the impact of atomicity for a data warehouse, it should have the ability to access data quickly and efficiently when querying large data volumes that occur as a result of denormalisation. There are opportunities to find the appropriate trade-off between the performance of querying the data and the volume of the data stored, versus the ability to access detailed data against the performance of accessing that data (La Rosa et al., 2011). A higher granularity offers more efficient data storing, the trade-off is that the ability to answer varying queries becomes diminished as a consequence (La Rosa et al., 2011). Conversely, lower granularity can support a greater range of queries at the expense of storage space and diminished performance (La Rosa et al., 2011).

The architectural design of the project data pipeline is such that it meets the broad spectrum of granularity levels required to answer the business questions. Granularity is less of a concern at this point in the project lifecycle and will require further consideration as the data increases. The impact of the architectural design has been considered concerning reusability, efficiency, scalability, stability, and performance, to offer the greatest level of business value (La Rosa et al., 2011).

4.4 Data Pipeline

The project dimensional schema design was considered in conjunction with the practical data pipeline architecture. In data warehousing, a data pipeline is a mechanism for transferring data from the point of the data producers to a state for use by the data consumers (IT k Funde, 2020). IT k Funde (2020) outlines several steps that the data must go through to get from one end of the pipeline, where the data is ingested, to the other end, which is the output. The middle processing steps make up the pipeline, however, it is possible to have data outputs along the cycle of the data pipeline (IT k Funde, 2020). Data cleansing, data governance, data enrichment, and data processing all occur along the data pipeline (IT k Funde, 2020). The scope of this project is focused on business analytics and reports as an output, machine learning has been discussed and ultimately road mapped due to the volume of work and time scales involved. *IT k Funde* (2020) describes data pipelines as being necessary to inform the processes that make up the ETL activities within data warehousing. Figure 4 is a conceptual representation of a data pipeline.

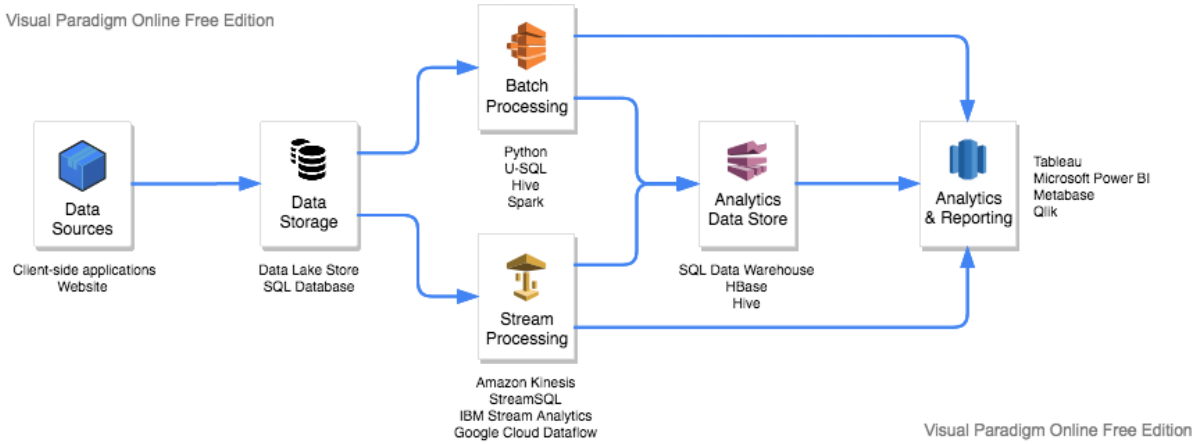


Figure 4 *Conceptual Data Pipeline Design*

4.4.1 Extract, Transform, Load

ETL is a triad of functions that retrieve data from numerous data sources and write it to a destination data warehouse (IBM Docs, 2021). The need for ETL arose as a result of the increasing amount of data sources across various applications deployed within organisations, often these applications will have incompatible formats, which makes data analytics impossible to achieve (IBM Docs, 2021). IBM Docs (2021) outlines the function of ETL as transforming data from OLTP relational databases such as Postgres, MariaDB, or SQL Server, or flat files such as spreadsheets, plain text files, and documents to a staging data store. The data is prepared for ingestion into an OLAP central dimensional data warehouse, and where necessary pushed to data marts (Rifaie et al., 2008). The project has revealed the challenges of effective data management, particularly where the ETL process involves different code bases such as SQL and JSON. The functions that make up the ETL process are described in Appendix H.

IBM Docs (2021) recommends that the ETL process be performed on a regular, scheduled basis. To achieve the best performance this is often conducted nightly so the data warehouse can provide consistent analysis for users during working hours. To enable an OLAP data warehouse to ingest data, the source must be in a relational format, which may require mapping, particularly when combining multiple data sources (IBM Docs, 2021). Smallcombe (n.d.) states that for an ETL process to be effective, it must be deployed around a clearly defined workflow that is continuous, extracting data from either a homogeneous system, a single database source, or a heterogeneous system, where several different OLTP databases operate collectively but as distinct units. The process can be time-intensive, requiring careful and detailed planning that requires input from data engineers and developers (Smallcombe, n.d.). It is possible to

simplify this process and accelerate deployment by utilising a cloud-based SaaS solution capable of performing this function across numerous sources instantly, the project currently deploys Stitch for this purpose. As with any technical process, there are advantages and disadvantages of an ETL process that should be considered, these are outlined in Table 1 (Smallcombe, n.d.).

Advantages	Disadvantages
The preservation of metadata that supports the correct understanding of the data by analytical tools An established process with many available tools on the market designed with user-friendly interfaces Built-in task management that eliminates the need for coding There is a lower level of skill required as a result of the tools available, including supplier support and documentation Supports regulatory compliance because data transformation occurs before loading to the data warehouse	Tools can be expensive, with costs associated with storage and compute time The complexity associated with transformation logic, or data staging, can be difficult to achieve SQL processing performance can be impacted resulting from the generic nature of these interpretive tools

Table 1 *Advantages & Disadvantages of the ETL Process*

4.4.2 Extract, Load, Transform

ELT takes a different approach to ETL, with this function the data is extracted and loaded into the data warehouse, no transformation or staging occurs (Marín-Ortega et al., 2014). Transformation occurs instead after it is loaded and made ready for the business intelligence tools (Marín-Ortega et al., 2014). “Why Shift from ETL to ELT?” (2016) lists the data types capable of being loaded in an ELT process:

- Structured relational data
- Unstructured data
- Semi-structured data
- Raw data

Marín-Ortega et al. (2014) highlight the flexibility of this process as being capable of loading data regardless of the format or type, even if no formal format exists. However, before the data can be made available for business intelligence, it must be cleansed, enriched, and transformed (Marín-Ortega et al., 2014). A consequence of transforming the data at the point of analyses is that it can be slower to generate insights, conversely, it does allow the data to be transformed in different ways, in real-time, to produce metrics, forecasts, and reports (Marín-Ortega et al., 2014).

Consideration should be made to the relatively new state of the technology available for ELT, which has been made possible due to the increased processing capabilities and scalability of cloud service providers such as Amazon Redshift and Google BigQuery. Additionally, the ability to pair ELT with a data lake allows organisations to ingest growing sets of raw data as it becomes available (Marín-Ortega et al., 2014). Table 2 outlines the advantages and disadvantages of the ELT process ("Why Shift from ETL to ELT?," 2016).

4.4.3 Comparative Analysis of ETL and ELT

Smallcombe (n.d.) highlights the architectural and implementation differences in the approach to data transformation when deploying ELT, differences that will require a paradigm shift, primarily with the leveraging of the data warehouse to carry out the transformation and negating the need for data. As a consequence, ELT pipelines require less planning and effort to set up (Smallcombe, n.d.). The second critical difference is concerned with privacy as it relates to PII (Smallcombe, n.d.). An argument can be made for an ETL pipeline as being better able to support the company's commitment to ensuring compliance with the GDPR and CCPA. Data can be cleansed of any sensitive information and secured before loading into the data warehouse. This could include encrypting sensitive data fields, transforming personal data such as emails to only include the domain, or removing the final section of an IP address. Deploying an ELT process means that the data is loaded before being transformed, making any sensitive or private data available to *SYSADMINs* in the logs, potentially violating compliance standards. This could be particularly problematic if, as a result of the transformation, the data leaves the EU or recognised *third-country* safe harbour ("Third Countries," n.d.). Furthermore, ETL is capable of performing sophisticated data transformations more cost-effectively than can be achieved with ELT ("Why Shift from ETL to ELT?," 2016). As a data integration process, ETL has been around for more than two decades, naturally,

there are a greater array of tools available on the market that can assist with the process ("Why Shift from ETL to ELT?," 2016).

Advantages	Disadvantages
In some instances, there may be no requirement to deploy a transformation engine as the work can be performed in the target system Loading and transformation can happen in parallel reducing the time, effort, and resources necessary to deploy the process Reduces the time data spends in transit which can be more cost-effective	Careful planning and design are required to ensure the data warehouse is capable of performing the transformations needed to meet the business requirements ELT is considered to be an emerging solution, the availability of tools is not as readily available, which could make the process more difficult to plan and establish If the preferred supplier is not thoroughly investigated, it may be discovered later that its deployment does not meet the regulatory obligations of the organisation

Table 2 *Advantages & Disadvantages of the ELT Process*

Much of the decision making between these two processes will come down to the company and how it structures its data across the various business units. If a data lake is integrated into the infrastructure and contains a vast array of varying data types, an ELT might be preferred to take advantage of its increased flexibility and capability to provide immediate access to data as it is stored. ELT eradicates the need for a complex ETL process, Smallcombe (n.d.) suggests this will save time when analysing new information as a result of the reduced load time as there will be no need to wait for the data to be cleaned or modified. ELT is a processing method created to recognise that BI is an evolving and changing solution, which makes choosing the right process essential (Bologa & Bologa, 2011). Bologa & Bologa (2011) emphasise the importance this choice can have, particularly on the long-lasting impacts it can have on the organisation and the detrimental budgetary effects on the wider IT infrastructure. Furthermore, performance optimisation might be affected which could result in the inability to respond to business-critical needs across departments (Bologa & Bologa, 2011).

The project deploys an ELT pipeline, the tools have been deemed as readily available, reliable, cost-effective, and the benefits are clearly understood by the decision-makers. Several reasons have supported this decision. Firstly, the initial data set specifically excludes any PII or personal data that would fall outside the scope of processing as defined by the GDPR, therefore, privacy compliance has been satisfied. Secondly, the data is required in near real-time to be most effective for the stakeholders. Figure 5 is a representation of the deployed ELT pipeline process.

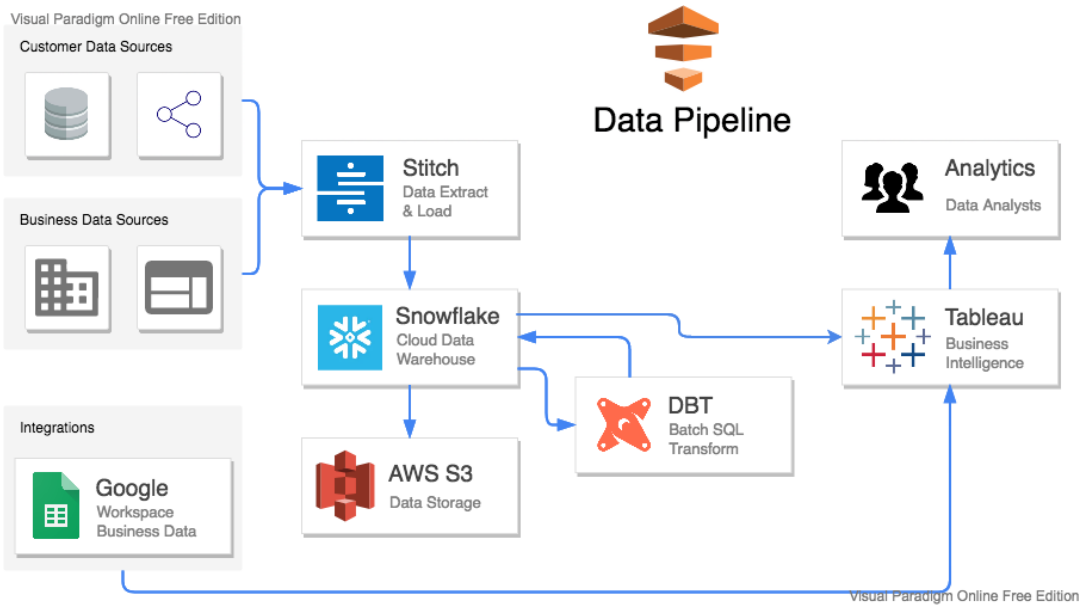


Figure 5 *Project Data Pipeline*

DBT (Data Building Tool) has been selected to perform the transformation in the ELT pipeline. DBT allows data analysts and engineers to transform data through command-line `select` statements within the data warehouse or data lake environment, which can be used to produce data visualisations to business users (*What Is Dbt?*, n.d.). DBT does not offer *extract* or *load* functionality, instead, it allows transformation to occur more efficiently by taking code, in this case, JSON, and compiling it into SQL (*What Is Dbt?*, n.d.). The use of SQL means there is no requirement for analysts to learn new tools or coding languages, as such aids collaboration and understanding of the transformed data (*Hooks & Operations | Dbt Docs*, n.d.). SQL queries can be run to perform database management tasks, these include:

- Creation of user-defined functions
- Granting table privileges

Hooks and operations are used in DBT to execute these tasks. Hooks are snippets of SQL that can be executed at different times, these include:

- `pre-hook`: executed before a model, seed or snapshot is built
- `post-hook`: executed after a model, seed or snapshot is built
- `on-run-start`: executed at the start of DBT run, DBT seed, or DBT snapshot
- `on-run-end`: executed at the end of DBT run, DBT seed or DBT snapshot

By way of example, on-end-run hooks can be used to grant usage on a target schema to a specific role, likewise, post-hook can grant `SELECT` on models to a specific role (*Hooks & Operations | Dbt Docs*, n.d.). Operations allow for macros to be invoked without the need to run a model and are triggered in the CLI using the DBT run operation command (*Hooks & Operations | Dbt Docs*, n.d.). Where operations differ from hooks there is a need to explicitly execute the query within a macro. Macros can perform similar actions to webhooks, for example granting schema usage to a role or granting `SELECT` to a table (*Hooks & Operations | Dbt Docs*, n.d.). DBT can be deployed using dev environments with version control in GitHub, further enabling collaboration and allowing data engineers to return to previous states ("Dbt for Data Transformation – Hands-on Tutorial," n.d.). The environment architecture allows models to be tested before production. In addition to the command-line interface (CLI), DBT is also available as a more user-friendly user interface (UI), however, the CLI must be used to run tests, for compiling data, and for generating documents. The UI is mainly used for documentation (*What Is Dbt?*, n.d.). DBT is capable of supporting most databases as follows, and meets the integration requirements of the company:

- Snowflake
- Postgres
- Redshift
- BigQuery
- Presto

Installation of DBT is through Python package installers (pip), both the CLI and UI are part of the package (*What Is Dbt?*, n.d.). A further benefit of DBT is that it is an open-source solution that allows for customisation to meet the company's technical needs (*What Is Dbt?*, n.d.).

4.4.4 Data Streaming

The project data warehouse architecture is leveraged to deliver operational data through a data stream of events, instead of batch updates, this is accomplished with a connected stream analysis engine (De Rougemont & Cao, 2012). Batch processing is explained in Appendix I. Before integration, the data is subject to analysis to ensure the crucial information is ingested (De Rougemont & Cao, 2012). De Rougemont & Cao (2012) highlights the benefits of data streaming, allowing real-time information to be deployed tactically to provide feedback to operational data sources or strategically for management level decision making. Data streaming is necessary for the project to produce real-time business intelligence, necessitating the need for the ETL process to shift away from periodic batch updates. Despite the benefits of data streaming, it is not without its problems (De Rougemont & Cao, 2012). Firstly, inconsistency with views can arise as a result of the continuous and discriminate updating of data, and lastly, concurrent updates must be resolved to maintain the effectiveness of analysis queries that produce the long-term data views (De Rougemont & Cao, 2012). Further challenges surrounding schema updates are outlined by De Rougemont & Cao (2012), namely that sufficient time is not available for the implementation of changes. Additionally, data transformation must be considered in an ETL process because it occurs before load, impacting load time (De Rougemont & Cao, 2012).

The real-time benefit of data streaming is likely to result in a higher level of complexity and greater maintenance of the system, however, both batch processing and data streaming techniques can be deployed to meet disparate needs (De Rougemont & Cao, 2012). The project data pipeline uses the extract and load solution Stitch, which updates the tables in the data warehouse on a near real-time basis. Stitch Import API is a REST API that enables data engineers to send data from a source, this means the import API does not extract data, instead it acts as a receiving point for the data, enabling it to be processed and pushed to a data warehouse (*Data Transformation and Data Quality*, n.d.). The import API accepts `JSON` or `Transit` and returns `JSON` for all methods using `HTTP` verb for example `GET` and `POST`. Data is processed using `UPSERT` to ensure that updates are captured, and any new data inserted, this prevents duplication from occurring in the data set (*Data Transformation and Data Quality*, n.d.). The benefit of Stitch's extensibility is it allows organisations to connect to disparate SaaS data sources deployed across departments, sources can include (*Stitch Platform*, n.d.):

- Salesforce
- Trello
- Facebook
- MailChimp
- SQL Server

The Stitch platform enables data engineers to maintain visibility and control over the data pipeline as data is transmitted from data sources to the destination (*Stitch Platform*, n.d.). Data replication scheduling can be performed as necessary to meet specific needs, this can range from every minute to once a day (*Stitch Platform*, n.d.). Further advanced scheduling can be configured, including specific granular extraction start times, or the whitelisting of set hours to perform data extract (*Stitch Platform*, n.d.). Extraction logs and loading reports are produced that allow data engineers to monitor the replication process and quickly identify results that are recognised as either unexpected or incorrect (*Stitch Platform*, n.d.). This provides a fast route to remedy any problems that arise (*Stitch Platform*, n.d.). Error handling is an additional feature integrated into the service, Stitch detects any errors that arise in the pipeline and where possible will make any necessary corrections automatically (*Stitch Platform*, n.d.). If user input is needed by a data engineer, a notification is sent that details and describes the issue. These notifications can be integrated with external applications such as Slack, Datadog, and PagerDuty (*Stitch Platform*, n.d.).

The platform is a highly available infrastructure that allows businesses to process billions of records every day, whilst being flexible enough to scale up or down the data volume as business requirements change (*Stitch Platform*, n.d.). This is achieved through a combination of the service level agreements that guarantee system uptime and the stated support response times, meaning there is no need for a business to concern itself with hardware provisioning or workload management (*Stitch Platform*, n.d.). The transformation and data quality solution, in conjunction with the performance attributes, allow data teams to deliver analytics in near real-time (*Stitch Platform*, n.d.). Transformation can be performed through a vast array of connectors and components such as *sort*, *aggregate*, *join*, *map*, and more (*Stitch Platform*, n.d.). Data quality can be improved on any size or format, including unstructured data, through the parsing technology integrated into the service (*Stitch Platform*, n.d.).

Consideration is made to the privacy and security of the data transmitted during the ETL process (*Stitch Platform*, n.d.). Sensitive data can be masked to protect the privacy of individuals, supporting organisational efforts to meet regulatory frameworks (*Stitch Platform*, n.d.). This is further supported by

Stitch's compliance with the EU's GDPR, and the company's independent audit accreditation against SOC-2 security, availability and confidentiality principles (*Stitch Platform*, n.d.). The security features provided by Stitch include:

- HSTS encrypted communication for web browsers
- Configurable minimum permissions that allow read-only access to necessary data, or subsets of data through replication
- SSH tunnelling, SSL/TLS, and IP whitelisting for secure connections to data sources
- Direct access to logs from data source integrations for auditing purposes
- Data retention only as long as necessary to ensure successful load to destination sources

Process automation is used to perform updates that produce mixed results. After extensive testing of the solution, the data team was able to confirm that the data stream does identify new records added to the sales application for extraction and load and performs updates to existing records where values have changed, this technique is referred to as `UPSERT`. Should this not have been the case, recurring revenue, which is subject to change every month, would be inaccurate.

4.5 Data Cleansing and Validation

Gaining meaningful insights from the data stream is made possible through the removal of impurities to enhance quality, the greater the quality the more accurate the insights (Müller & Freytag, 2003). Müller & Freytag (2003) define the process of data cleansing as the methods used to enhance data accuracy. The closer the data can get to being error-free, the greater the confidence that can be gained to make informed business decisions (Müller & Freytag, 2003). In its simplest form, data cleansing is used to eliminate duplicates, a problem that becomes compounded where data is ingested from multiple sources (Müller & Freytag, 2003). Müller & Freytag (2003) further explain that data cleansing extends beyond the scope of this simplistic view, indicating that no prescribed level of data cleansing must be achieved to satisfy any arbitrary standards, instead, it must meet the demands required of the processing. Therefore, comprehensive data cleansing is achieved through the definition of standards and criteria, and the classification of anomalies that must be eliminated to reduce frustration and avert ineffective utilisation for data analysts (Müller & Freytag, 2003). Effective data cleansing can also increase query performance (Müller & Freytag, 2003). Data cleansing is most effective when the process is as automated as possible, which normally requires the involvement of a domain expert to maximise the knowledge and information

available to detect and correct anomalies (Müller & Freytag, 2003). Müller & Freytag (2003) conclude that manual cleansing of data is time-consuming and difficult, recognizing that it is not always possible to automate the process entirely, therefore, data cleansing is considered a semi-autonomous process.

4.5.1 Classifying Data Anomalies

Müller & Freytag (2003) classify data anomalies broadly into three category types:

1. Syntactical Anomalies
2. Semantics Anomalies
3. Coverage Anomalies

Syntactical anomalies are identified by the format and values of entities and their characteristics (Müller & Freytag, 2003). Müller & Freytag (2003) further define these into three sub-set anomalies. The first is lexical discrepancies, which occur when the actual data structure does not conform to the defined format, the second is domain format errors, being values that do not conform to the expected format, and the third are irregularities that relate to non-uniform values (Müller & Freytag, 2003).

Semantic anomalies prevent the collected data from existing in a non-redundant and comprehensive state (Müller & Freytag, 2003). Müller & Freytag (2003) further define these into four sub-set anomalies. The first of these anomalies are integrity constraint violations, which are tuples that fail to meet one or more defined integrity constraints, the second are contradictions, which occur as a result of a dependency breach, the third are duplications of tuples representing the same entity, and finally, there are invalid tuples that do not fall into any category, making them difficult to detect (Müller & Freytag, 2003).

Coverage anomalies decrease the number of entities and their properties that are represented in the collected data (Müller & Freytag, 2003). Müller & Freytag (2003) further define these into two sub-set anomalies. The first is missing values resulting from errors or omissions, and the second is missing tuples from the data collection or the amount of data accounting occurring for a constraint violation (Müller & Freytag, 2003). Appendix J provides a detailed description of the three data anomaly classifications and their sub-sets.

4.5.2 Data Quality Criteria

To ensure effective processing and high-quality interpretability, data ingested to the warehouse must satisfy a set of established quality criteria described as a set of criteria that provide a comprehensive process of data cleansing (Müller & Freytag, 2003). The criteria include a hierarchy of nine specific criteria that are represented in Figure 6, the more the criteria is sub-divided the finer the grain (Müller & Freytag, 2003). Müller & Freytag (2003) set the criteria for data quality to include accuracy, integrity, and completeness of the data, additionally, the data should be validated and maintained in a consistent state and conformance with the schema. Lastly, Müller & Freytag (2003) state that the data should be uniform across the data set and the density assessed to ensure the quality level is not downgraded, whilst being unique through the removal of duplicates.

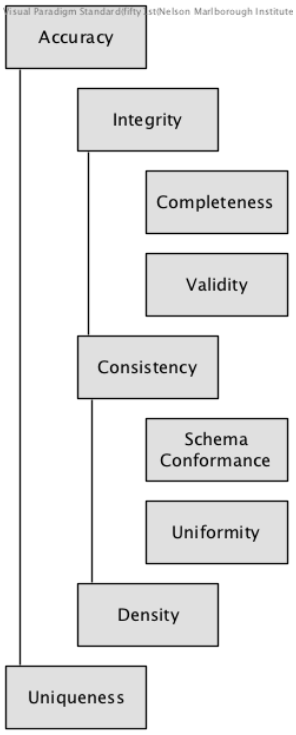


Figure 6 *Quality Data Criteria Hierarchy*

A description of these nine quality criteria is available in Appendix K. Table 3 lists the quality criteria that are not aggregated, in addition to the anomalies that can impact the quality criteria being achieved (Müller & Freytag, 2003). The circles represent where quality criteria are downgraded as a result of the anomaly, the dashes represent anomalies that can prevent or make it difficult to detect other anomalies that might downgrade the quality criteria (Müller & Freytag, 2003).

4.5.3 Data Cleansing Methods

Several methods can be deployed to perform the data cleansing process and eliminate anomalies (kexugit, n.d.). kexugit (n.d.) separates these five methods, categorised as (1) parsing, (2) correction, (3) standardisation, (4) matching, and (5) consolidating. These methods are used to detect if syntactical errors exist in attribute values or tuples, which can also include the correction of spelling errors, collectively this encompasses the parsing method (kexugit, n.d.). Parsed data is then subjected to the statistical correction that acts as an auditing process for correcting and eliminating complex errors (kexugit, n.d.). kexugit (n.d.) states the purpose of this method is to uncover relationships that often do not conform to any expected characteristics of the data set. Data in a data warehouse must be standardised to provide meaningful analytics formatted to a common schema design, this can include formatting changes such as dates or addresses, the conversion of value types such as `int` to `float`, and to remove irregularity at the instance level (kexugit, n.d.). kexugit (n.d.) suggest that once the data is standardised, duplicates can more easily be identified in the data set and eliminated. kexugit (n.d.) further indicates that integrity constraints can be enforced against the cleansed data set and any identified violations rejected to guarantee the results, this final method is the act of data consolidation. A detailed description of the data cleaning methods is available in Appendix L.

	Completeness	Validity	Schema Conformity	Uniformity	Density	Uniqueness
Lexical Error		-	●	-	-	-
Domain Format Error		-	●	-		-
Irregularities		●		●		-
Constraint Violation						
Missing Value					●	-
Missing Tuple	●					
Duplicates						●
Invalid Tuple		●				

Table 3 *Anomalies capable of impacting non-aggregated quality criteria*

4.5.4 Cleansing and Anomaly Correction

The iterative and incremental nature of the scrum methodology has resulted in data being ingested into the data warehouse and visualisation tools tested against the data set early in the development lifecycle and before a complete cleansing and validation process. Should the team have deployed a waterfall methodology this might have been viewed as a failure of the process. However, the scrum methodology has allowed for the identification of unexpected issues with missing data, resulting in a beneficial contribution to the validation process.

Missing values were identified in the project data set, an omission that had historical not been considered problematic. The missing data values had to be defined and split into two distinct categories, simple revenue recurrence split over a set monthly period, and complex revenue recurrence where the amounts vary month to month or skip months entirely. This was resolved through the addition of an object field in the sales application that either defines the number of months, allowing for the recognised revenue to be divided evenly across the months beginning with the revenue recognised date or the monthly amounts to be inserted in a comma-separated format so they can be allocated to the relevant months in order. Months with no allocated revenue would include `zero` or `null` values before the comma. Performing this anomaly correction led to duplications in the data set as a consequence of a lack of data input standardisation. In this case, data fields had been used incorrect by sales representatives, meaning deals were identified as both *one-off sales* and *sales with recurring payments*. Two methods have been deployed to resolve the issue. Firstly, the extraction process prioritises the sales type and ignores the recurring revenue amount to prevent duplicates, and secondly, the data team members responsible for the sales operation have eliminated duplications in the sales system, and have provided additional instructions to the sales representatives on the correct data input procedure. A further issue was identified from the data stream. Due to the organisational structure, one region operates independently, meaning their data must be bulk ingested into the sales application. This data set does not contain values for all fields, in particular the fields that the data stream identifies for the extraction process. Once identified this was corrected. The resolution of these anomalies resulted in a reliable source of truth for the revenue generated across all regions with granularity to the day level.

4.6 Protecting Data Subjects Privacy

The company takes security and privacy seriously, evidenced by the SOC 2 Type-II security audit that was carried out concurrently with this project. Meeting the audit standards necessities two organisational controls that have been completed for each new application chosen for this project, firstly a *Vendor Risk Assessment* was conducted, and secondly, entries were made into the *SaaS Risk Register*. The requirements for these security tasks are outlined in Appendix M. The audit also incorporated a significant GDPR component that has been partly developed for this project as a direct consequence of the increased data processing activities, which, in addition, had precipitated the need to research the principles and methods of *anonymisation* and *pseudonymisation*.

4.6.1 Anonymisation

Data anonymisation is a process of transforming data to remove both direct and indirect personal identifiers (K-Anonymity, 2021). Data anonymisation can be performed on any unit of information that may lead to a data subject being identified, and if necessary, can be performed on information clusters (K-Anonymity, 2021). K-Anonymity (2021) states that the benefit of this process is that the confidentiality of the data subject is safeguarded, meaning it cannot be inferred from the anonymised data. Once the data has been transformed it is impossible to link it to a data subject, meaning it is no longer defined as PII and falls out of the scope of the GDPR (K-Anonymity, 2021).

The anonymisation process is not without its challenges (K-Anonymity, 2021). A thorough assessment of the adequacy and scope of the technique must be undertaken to ensure that the data isn't devalued to such a degree that it is difficult to gain meaningful insights (K-Anonymity, 2021). The granularity should be sufficient to meet the use case the data was originally intended, which can often lead to trade-offs between the quality of the output, and the level of de-identification (K-Anonymity, 2021).

4.6.2 k-Anonymity as a Privacy Protection Model

Data anonymity can help businesses and organisations manage their growth in data collection as computer systems and network connectivity become more capable and affordable (Sweeney, 2002). The relative ease of data collection that is now possible must be counterbalanced with the need to protect data

that is corporate sensitive, particularly when it is liable to be shared outside the organisation with third parties or governmental bodies (Sweeney, 2002). An alternative approach to meeting these requirements is through a process of data minimisation and data deletion as outlined in the GDPR (*Understanding the 7 Principles of the GDPR - Blog - OneTrust*, n.d.). However, this has the potential to conflict with the company's need to meet its contractual obligations and service level agreements with its customers. Sweeney (2002) offers k-anonymity as an approach that is capable of removing explicit identifiers that can result in data sets that become anonymous. Sweeney (2002) acknowledges that anonymisation can become challenging where private data in one form, also appears in publicly available sources. If an assumption on this quasi-identifier is made by the data controller, the sensitivity has the potential to be misjudged, adversely affecting any organisational risk mitigation (Sweeney, 2002). Weighing the attributes of quasi-identifiers can provide more granularity and flexibility (Sweeney, 2002). Consequently, Sweeney (2002) suggests that anonymisation be used in collaboration with regulatory frameworks, internal policies, and contractual obligations. Appendix N outlines the challenges encountered with the process of data anonymity.

Sweeney (2002) defines *k-anonymity* as a protection model and set of deployment policies that have been developed to offer scientific guarantees to organisations, ensuring that PII or private data can be deployed in a way that would make it impossible to identify the data subjects. The data anonymisation process means that the data subjects are indistinguishable from other individuals who appear in the data set, this is referred to as *hiding in the crowd* (Sweeney, 2002). Since its proposal over two decades ago, *k-anonymity* has evolved to become an effective tool to protect data privacy in an age of increasing governmental regulation (Sweeney, 2002). The driving force behind the concept was the idea of combining sets of data that have similar attributes, therefore obscuring the identifying information by grouping it within a larger group (Sweeney, 2002). Sweeney (2002) defines the *k* in *k-anonymity* as a variable in the same way *x* and *y* are used in algebra, being used as an identifier of the number of combinations of a value. If a data set is anonymised to *2-anonymous* or $k = 2$, then it signifies that the data has been generalised to allow for a minimum of two sets of every combination in the data set (Sweeney, 2002). By way of example, imagine a data set that contains a group of data subjects' location and ages, $k = 2$ would need to be generalised so that each location and age pair appears in the data set twice, at a minimum. Sweeney (2002) states that if this process of generalisation results in the removal of an attributes value entirely, it

will become suppressed, becoming irrelevant or mostly irrelevant. A detailed description of generalisation and suppression is available in Appendix O. Deploying the *k-anonymisation* technique cannot completely mitigate the risk of re-identification, however, due to the impracticality of the effort involved, it would be unlikely (Sweeney, 2002). It is important to note that zero per cent risk of re-identification is not industry standard, and this is acknowledged within the regulatory framework of the GDPR, making reasonably impossible re-identification the goal (K-Anonymity, 2021).

4.6.3 Pseudonymisation

Pseudonymisation as an alternative solution is referenced by the EU's GDPR to protect data subjects in such a way that it can no longer be attributed to the specific individual (What Is Pseudonymisation? | Thales, n.d.). Pseudonymisation can play an important role as both a security measure as outlined in Article 32 of the GDPR and in the context of data protection by design as described in Article 25 (What Is Pseudonymisation? | Thales, n.d.). What Is Pseudonymisation? | Thales (n.d.) outlines the pseudonymisation procedure, which includes the mapping of identifying fields of a set of data records, followed by a technical process that transforms the identifiable data so it requires the inclusion of one or more artificial identifiers that are stored separately. The artificial identifiers, or pseudonyms, can include one pseudonym that is applied to a cluster of fields as a replacement data point, or a pseudonym for each field to be replaced (What Is Pseudonymisation? | Thales, n.d.). This information is subjected to rigorous organisational and technical measures to preserve the identity of the data subject (What Is Pseudonymisation? | Thales, n.d.).

What Is Pseudonymisation? | Thales (n.d.) outline the benefits to businesses as a means to meet their commitments to the principle of data minimisation as outlined in Article 5(1c) and 5(1e) of the GDPR. Once an organisation has analysed the data set and clarified what constitutes PII, it is important to recognise that the pseudonymised data still falls with the scope of the GDPR, as it allows for indirect re-identification (What Is Pseudonymisation? | Thales, n.d.). Esayas (2015) discusses the ability of encryption to conceal data that has been identified as PII, where the plain text is transformed into unintelligible code. Encryption has become an increasingly valuable privacy enhancing measure as more data services are being deployed in cloud environments (Esayas, 2015). Esayas (2015) argues the case for an industry recognised strong encryption algorithm utilising a string encryption key that is secured, meaning that

should the data fall into the hands of a legitimate third party or a threat actor, it would not be considered as PII or personal data. The counter-position is that the security and privacy of the data are reliant on several supporting factors, which include organisational controls and the technical skills of those charged with the implementation and robustness of the encryption (Esayas, 2015). Ultimately, the data is still capable of identifying data subjects, and the original data is still deducible and potentially deducible in the event of a security attack resulting in a data breach (Esayas, 2015). Encryption through the use of two-way cryptology algorithms is, therefore, still subject to data privacy regulations (Esayas, 2015). Esayas (2015) recommends an alternative solution utilising a one-way cryptography identifier that renders the data irreversibly encrypted, the wider consensus is that this would offer a safe harbour solution. One-way encryption, or keyed-hash function, where the key has been deleted, may allow processing to take place independently of the obligations outlined under the GDPR (Esayas, 2015). Esayas (2015) outlines a lack of consensus, with opinions not clear as the method only ensures the data becomes computationally difficult for an attacker to decrypt through a process of testing every possible key.

The European Union Agency for Cybersecurity has published a framework outlining the techniques and best practices for considering pseudonymisation (Bourka et al., 2019). Bourka et al. (2019) outline the deployment of these techniques by organisations to obscure the identity of data subjects from third parties, reducing the risk of cross-application linking. These practical scenarios are detailed in Appendix P.

4.6.4 Comparing Anonymisation and Pseudonymisation

The decision to anonymise or pseudonymise the data will depend on the use case and techniques available to achieve the goals (“Data Anonymization Techniques and Best Practices,” 2020). Esayas (2015) suggests that under the GDPR, data pseudonymisation techniques can reduce the regulatory restrictions when handling PII. Furthermore, it is relatively easy to perform when compared with anonymisation (Esayas, 2015). However, Esayas (2015) points out that there is a benefit to the increased complexity of performing data anonymisation through the addition of layers of security to the data set. Fundamentally, both methods are capable of providing a safe harbour from certain obligations under the regulations. Data breaches are one such example, providing the method irreversibly prevents identification (Esayas, 2015). Regardless of the method deployed, it will constitute the processing of personal data under the regulations and must therefore satisfy the legal grounds for processing (Esayas, 2015). Esayas (2015) is

unequivocal, data violation would occur should the process to anonymise be in breach of the original purpose for processing, unless certain criteria are met, particularly surrounding the reliability of the anonymisation process. Anonymisation is, however, capable of satisfying the requirements under Article 6(1)e that requires data not to be retained for longer than necessary for its intended purpose when originally collected (Esayas, 2015). Anonymisation can also be used as a solution to satisfy the requirements of data deletion when the original legal basis has been exhausted (Esayas, 2015).

Unlike anonymisation, pseudonymisation does not prevent data subjects' information from being linked across data sources (Esayas, 2015). This approach allows individuals to be singled out and re-identified if a threat actor obtained the keys through a brute force attack or data breach (Esayas, 2015). To ensure the effectiveness of any data protection effort, Esayas (2015) suggests deploying a combination of anonymisation techniques and data deletion processes that ensures the required level for safe harbour status. Esayas (2015) caveats this approach by recognising this view is not universally agreed, with authorities concluding that despite the higher risk of re-identification with pseudonymisation, it is still capable of offering adequate protection.

Determining whether the project use case includes in its scope PII as identified in Article 4(1) of the GDPR can be challenging, this extends from directly identifiable information to indirectly identifiable information (Esayas, 2015). If the data is deemed to fall under the terms as defined as PII, then as a general rule deploying either technique will support risk mitigation and limit the exposure to fines that can result from non-compliance (Esayas, 2015). Esayas (2015) indicates that the starting point in the decision-making process should be the clarification of the legal distinction. For instance, anonymised data cannot be re-identified without a disproportionately large effort. If pseudonymisation is the preferred method then there may be a direct, indirect, or remote form of re-identification as the process does not remove all the PII, instead, it reduces the number of relations that can be used to identify the data subject, whereas anonymised data eradicates any trace of a data subjects' identity (Esayas, 2015). This eradication of data means that all of the obligations under the GDPR are no longer applicable, such as the right to be erased, the right to be forgotten, and the right to make user requests (Esayas, 2015). Both techniques render the data transformed to a sufficient level it can be transferred to other countries, breaking down the barriers with regards to the data storage location. Depending on the size of the data set there may be additional organisational implications such as cost benefits and accessibility (Esayas, 2015).

Despite the company operating a business-to-business model and collecting minimal quantities of PII, risks remain. For instance, there are business questions that seek to understand the gross profit for each deal, the sales representative generating the deal, and the role type of the employees creating the final product. Additionally, the company accesses data from a variety of social media platforms that collect a large amount of data. Considering the global nature of the business and its significant operations inside the EU, anonymisation was tested for this project as the method negates the need to directly consider the regulations.

4.6.5 The Anonymisation Process

An investigation of several tools to test the data anonymisation process has been conducted. These tools include Anonimatron, Amnesia, and ARX Data Anonymization Tool (ARX). The implementation of these tools was met with varying degrees of success. Anonimatron was deployed first and deployed in a sandbox environment to protect the wider IT infrastructure. The set-up failed party due to a lack of sufficient documentation to make any further effort worthwhile. Amnesia was then tested having been recognised as the preferred solution being a cloud-based service capable of meeting the company's wider IT infrastructure requirements. For the scope of this exercise a full vendor risk assessment, including a determination of the geolocation of the cloud infrastructure to ensure it met the regulatory compliance required with regards to transferring information to data centres outside the EU or safe harbour countries, would be an undesirable process that would require authorisation from management level stakeholders. If data centres were discovered to be outside the EU or safe harbour countries this would negate the purpose of the data anonymisation process. As a result, this solution was abandoned. Lastly, ARX was installed and a suitable CSV file was chosen that included specific fields that had the potential to identify a data subject as defined by the GDPR.

The process of data transformation involved some decision making. Firstly, the level of transformation to ensure that re-identification cannot be performed without significant effort must be determined, and secondly, it must remain in a state that allows a business to obtain valuable analytics. The process can also be used as part of the wider extract, transform and load (ETL). Figure 7 displays the generalised data output that has been transformed from specific identifiable ages to an age range. Information relating to marital status, education, and work class have been suppressed as part of this process. Suppression

removes the data that might be considered irrelevant for producing valuable data analytics. In this specific example, spousal information has been transformed to identify only those individuals that have spouses present, and those that do not.

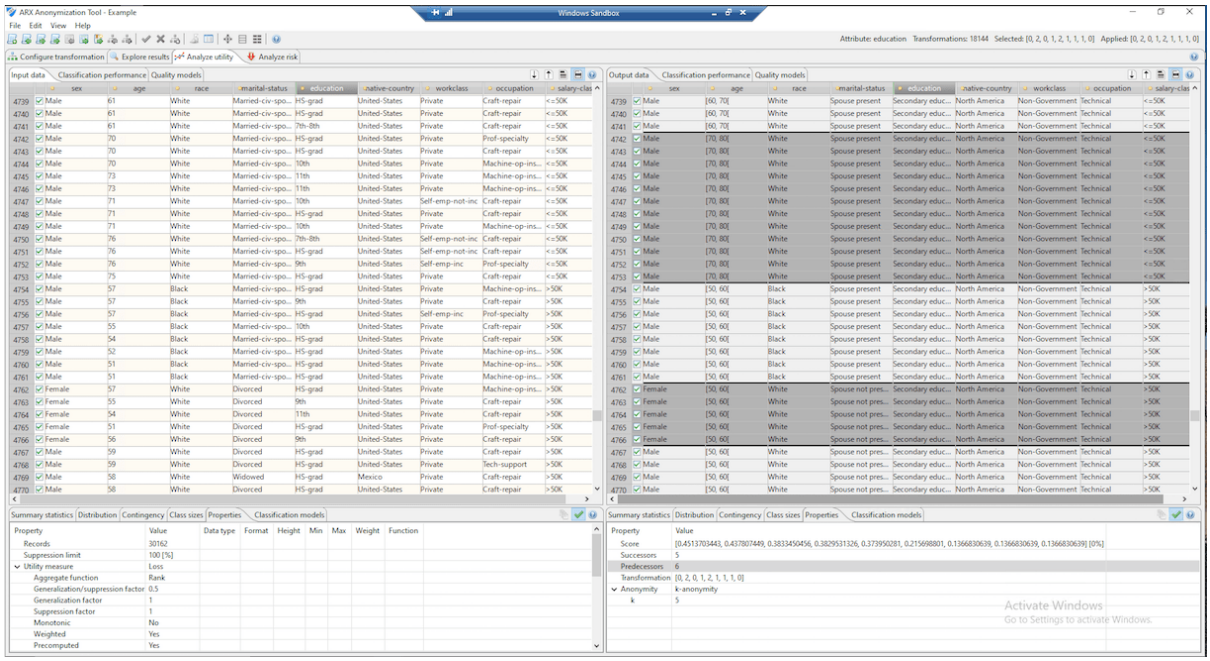


Figure 7 *Generalised data subject ages into age range utilising ARX*

The same application of the process has been applied to education, work class, and occupation. Figure 8 shows one of several metrics and analytics generated from the results. These results can be used to evidence compliance with the internal organisational processes and external regulatory frameworks.

Figure 8 *Distribution of affected records relating to data subject's industry segment employment*

Figure 9 and Figure 10 display the graphical interpretation of the anonymity score, this can be adjusted to meet the anonymity requirements of the use case. This process requires serious consideration. If the score is not specifically tailored, then the granularity of the data will be insufficient to either meet the business needs or the obligations of the regulations. The project stakeholders should manage this trade-off between the risks to data subjects, against the impact on the level of business intelligence that can be gained. The completed anonymisation process produced a data set that was exported for analysis in Microsoft's Power BI, the output provided useful insights even at this level of granulation, whilst no longer being capable of identifying data subjects directly. Additional metric results are provided in Appendix Q.

Two concerns arose from the testing. Firstly, the main purpose of the project data is to display the sales representative's names, meaning anonymisation would result in less detail and a higher granularity, protecting the identity of the individual but making any business intelligence effectively worthless. Countering this with the fact that processing the data in this way has been assessed as legitimate under the GDPR, makes anonymisation unnecessary for the project's use case. Equally, pseudonymisation would also be ineffective for this purpose, again disguising the sales representative's name. Secondly, further problems arise with the ELT data pipeline architecture. Anonymisation is intended to limit the exposure of PII in the event of a data breach, this can only be achieved by deploying an ETL pipeline architecture that transforms data before it is loaded. Anonymisation intends to limit the attack surface area and mitigate the risk.

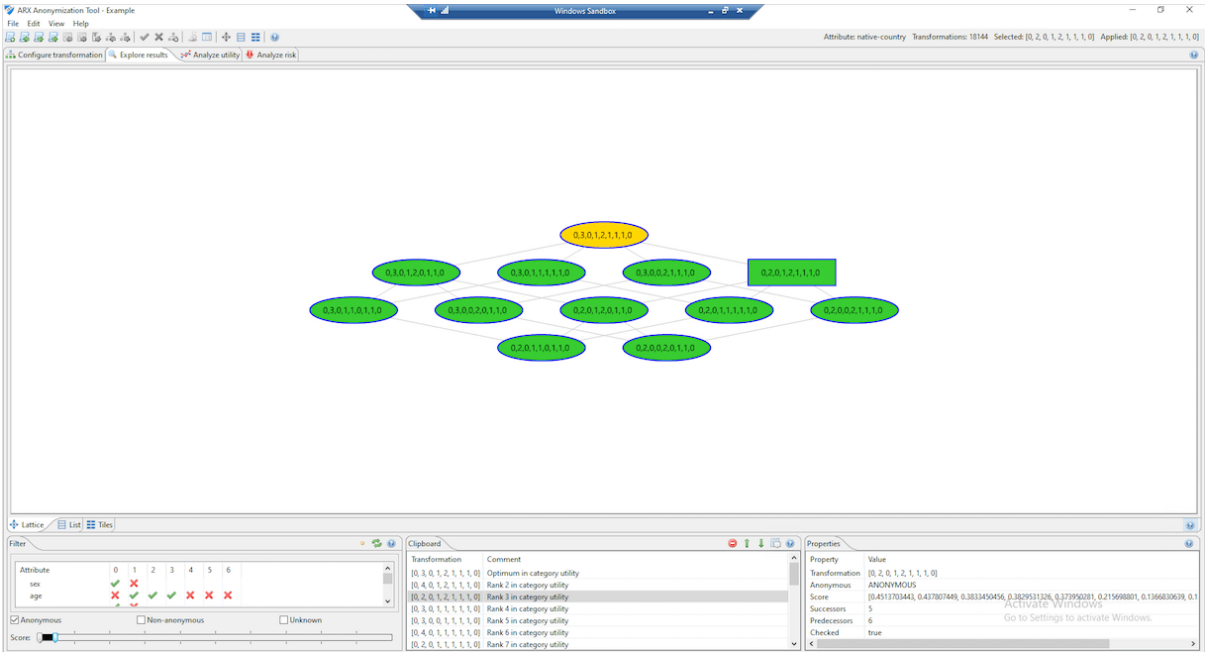


Figure 9 *Low anonymised rank score*

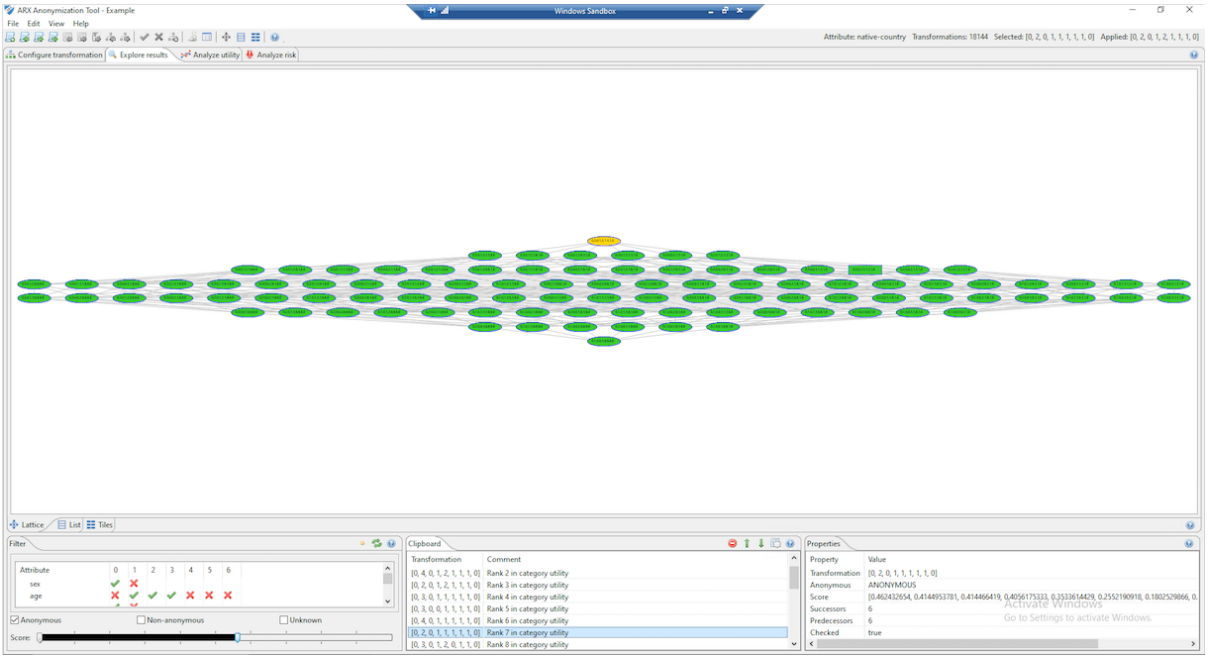


Figure 10 *High anonymised rank score*

4.7 Data Warehouse

Data warehouses are deployed to solve the problem of not having a single source of clean, accurate, and integrated data that can act as a source of truth for an organisation (Ariyachandra & Watson, 2010). A

data warehouse is a large analytical database that gathers and controls the consolidated data from heterogeneous systems for the broadest deployment, structured for easy querying, reporting, and analysis (Ariyachandra & Watson, 2010). It is the final data repository for loaded data in an ETL or ELT process ("Data Engineering and Its Main Concepts," n.d.).

Users accessing the data should be able to clearly understand what they're are accessing and why it is structured the way that it is (Rifaie et al., 2008). Data warehouses are not intended to store data to handle niche requirements, the purpose is to support almost everyone within an organisation (Rifaie et al., 2008). Ariyachandra & Watson (2010) affirm that data warehousing cannot be achieved with a single product or service that a company is capable of acquiring, it must be designed around the company's requirements. The benefits and importance of data warehousing for business management are clearly understood, despite this, there remains considerable discussion over the appropriate architecture that offers the best solution (Ariyachandra & Watson, 2010). The heavily structured environment and wide use of data warehouses mean they can be slow to change, their structure must be maintained as data is added or changes are made (Ariyachandra & Watson, 2010). In instances where changes are necessary to the structure of the data, Ariyachandra & Watson (2010) recommend a communication plan be created before it is rolled out across an organisation, preventing or mitigating the risk of problems for users.

4.7.1 Architecture Solutions

A data warehouse architecture can be described as a storage repository designed for data to be deployed in a manner that can be accessed and interpreted by non-technical business users through analytics tools (Ariyachandra & Watson, 2010). There are several architectural solutions available that are provided by a multitude of vendors. For this report, the strengths and weaknesses of four alternatives, (1) independent data marts, (2) data mart bus architecture, (3) enterprise data warehouse architecture, and (4) federated architecture, have been analysed (Rifaie et al., 2008). This analysis can be reviewed in Appendix R.

4.7.2 Enterprise Warehouse Environment

The data warehouse architecture analysis indicated that the appropriate solution for the project is an enterprise warehouse environment that required the creation of an adequate data model, adopting new

techniques to transform the operational data from the OLTP environments to an OLAP environment. OLAP models require a software solution capable of performing multidimensional analysis at high speeds on large volumes of data (Rifaie et al., 2008). In contrast, operational data in a relational OLTP database is designed to be updated continuously, this is not normally the case with a data warehouse, which focuses on historical business data (Rifaie et al., 2008). Rifaie et al. (2008) recommend that batch updates occur daily and overnight so as not to affect querying, or updated weekly or even monthly depending on the business needs. The advantage of a data warehouse architecture is an ability to study trends and answer strategic business questions from the related data, this cannot be achieved with the separate data sources deployed across most organisations (Ariyachandra & Watson, 2010).

The data warehouse architecture will typically ingest data from an entity-relationship database model (Ariyachandra & Watson, 2010). Data often requires significant transformation to be effectively represented in the dimensional data warehouse schema (Moody & Kortink, 2000). It is important to understand the distinction between the two models. Appendix S provides an analysis of the technical and operational difference between an OLTP relation database and OLAP analytical data warehouse (*Data Warehouse Tutorial For Beginners | Data Warehouse Concepts | Data Warehousing | Edureka - YouTube*, n.d.). The properties inherent to a data warehouse are subject-orientated, integrated, time-variant, and non-volatile collection of company data, which make it a faster and more accurate method for processing queries and retrieving information (Rifaie et al., 2008). These properties are described in Appendix T.

4.7.3 Principles of Data Warehouse Architecture

Deploying a data warehouse provides organisations with a holistic, consistent view of the enterprise through the integration of data sources, becoming a consolidated target for reporting that is made available to business stakeholders in a timely and cost-effective manner (Rifaie et al., 2008). Rifaie et al. (2008) outline the characteristics and technical capabilities to achieve these goals, *extensibility*, *scalability*, and *availability*, designed to improve business knowledge and support the management of compliance and regulatory obligations. It is recognised that as the data warehouse is deployed and grows in line with stakeholder requirements, so too will the quantity and complexity of the queries and the number of user's generating analytics from the data.

Rifaie et al. (2008) outline a set of guiding principles that should be considered for an architecture to be implemented and maintained:

- Data captured at the point of contact in a form that is both accurate and complete
- Integration of metadata across the organisation to improve communication and increase efficiency
- Data that is consistent and has enterprise-wide integrity irrespective of the storage location
- Development of strategies to support the management of data and information and knowledge assets
- Development of quality controls to ensure business needs can be met
- Creation of an enterprise-wide data model that clearly defines the data to be supported
- Standards that are specifically designed to eliminate data redundancy, whilst enhancing the integrity of the data available
- The appointment of a data steward and data custodian who can take ownership and responsibility
- Determine the information accessibility, security, and privacy, and articulate the classification of data and the definitions and rules

The principles outlined should be incorporated into the data pipeline to ensure the development of a robust architecture that eliminates redundancy and avoids inconsistency, resulting in higher quality reports across the business (Rifaie et al., 2008).

4.8 Deploying Cloud Infrastructure

The IT sector has witnessed significant shifts in the deployment of services to cloud-based providers, data warehousing is no different in this regard (Ly, 2019). Significant increases in the volume of data collected by organisations have opened opportunities that are being realised, particularly amongst businesses that are comfortable outsourcing business-critical processes to cloud-based service providers (Ly, 2019). Ly (2019) identifies the trend of traditional services being replaced with platforms such as Amazon Redshift and Snowflake.

4.8.1 Comparing Tradition and Cloud-Based Data Warehouse

A cloud approach allows companies to benefit from efficient solutions and the latest innovative technologies without high capital expenditure, allowing the cost can be moved to operational expenditure (V & erweide, 2019). Ly (2019) draws a comparison between on-premises data warehousing solutions and

cloud-based solutions highlighting the benefits of cloud adoption, these are outlined in Table 4.

On-Premises Solution	Cloud Solution
Longer to set up and deploy	Significantly faster deployment time
High cost of storage and computing	More affordable, as much as one-tenth of the cost
Limited flexibility, balancing, and tuning	Elastic, flexible, and more scalable
Possibility of delays and downtime	Almost no delays or downtime
Higher costs relating to security and discovery	Comparatively, far lower costs

Table 4 *Comparative Analysis between On-Premises and Cloud Data Warehouse Solutions*

In any organisation time is money, reducing the implementation time through the deployment of a cloud solution makes it an attractive option, whilst limiting exposure to market downturns without access to the data necessary to adapt (Ly, 2019). The reduced planning and implementation time of a cloud data warehouse can lead to it quickly becoming a crucial supporting technology (Ly, 2019). Ly (2019) highlights the speed of markets and the necessity for companies to keep up, this drives them to limit their exposure to capital expenditure that is supported through the use of cloud solutions. In contrast, on-premises solutions require servers, storage devices, high-speed network implementations, licensing fees, and increased staffing levels to be budgeted, in addition to forecasted events such as upgrades, ongoing maintenance, security protocols, firewall protection, and the monitoring and identification of vulnerabilities and threats (Ly, 2019). The advantages of cloud data warehouses should also be considered from a technical perspective (Ly, 2019). For instance, cloud providers are better positioned to handle peak usage, can offer seemingly unlimited storage and compute, and provide scalability to meet workload demands without affecting system performance (Ly, 2019). Ly (2019) concludes that cloud data warehouse architectures are capable of producing more effective data pipelines and efficient query run times.

4.8.2 Data Warehouse Solution

Snowflake has been chosen as the preferred service provider for this project to support the company to continue to realise the benefits of zero management of SaaS applications. Snowflake allows the organisation to bring together users, data and workloads into a single solution deployed completely on cloud infrastructure, breaking down the technological barriers experienced with other solutions and supports both Amazon Web Services (AWS) and Microsoft Azure, taking advantage of both platforms. In addition to the infrastructure advantages, users benefit from Snowflake's use of SQL. Furthermore, Snowflake supports common data formats including JSON, Parquet, and XML, creating a single source data warehouse from petabytes of both structured and unstructured data. A thorough analysis of the practical, technical, and security features included in the Snowflake application was conducted and factored into the decision-making process. Snowflake's architecture separates data processing, data storage, and data consumption. The separation of computing and storage allows for more flexible scalability, which is necessary to quickly scale up or down. System management is performed by the data engineers in a separate application side user experience to that of the data analysts extracting business insights. This process separation supports the management of concurrent bottlenecks during periods of high demand.

The benefit of Snowflake’s Data Platform is its use of micro-partitioning to automatically divide tables into small contiguous units of uncompressed data storage. Snowflake automatically determines a compression algorithm that offers the most efficient method to reduce the actual data size. Rows are mapped into individual micro-partitions that allows for extremely granular pruning of large tables. In contrast, traditional data warehouse's rely on static partitioning that increases maintenance overhead and data skew, which results in disproportionately sized partitions. Micro-partitioning has a beneficial impact on performing SQL Data Manipulation Language (DML) operations such as deleting rows from a table. The underlying micro-partition metadata facilities and simplifies the maintenance of tables, making them metadata only operations. The metadata maintained by Snowflake enables precise column pruning in micro-partitions, including columns that store semi-structured data (Micro-Partitions & Data Clustering — Snowflake Documentation, n.d.). For example, a table containing historical and uniformed data would allow a query to target and scan only those micro-partitions that meet the filter predicate on the range of defined values, then scan the micro-partitions that contain the data. Snowflake uses columnar scanning that

significantly improves efficiency and performance, if the query filter is by one column, then only the filtered column is scanned (Micro-Partitions & Data Clustering — Snowflake Documentation, n.d.).

Clustering is another important factor in query performance. To improve efficiency, Snowflake orders data when it is stored in tables. Snowflake collects clustering metadata and records it for each micro-partition when data is loaded (Micro-Partitions & Data Clustering — Snowflake Documentation, n.d.). The metadata is leveraged during query runs to avoid scanning irrelevant micro-partitions, offering accelerated performance (Micro-Partitions & Data Clustering — Snowflake Documentation, n.d.). Clustering metadata is maintained by Snowflake and includes the total number of micro-partitions for a table, the total number that contain overlapping values with one another, and the depth of the overlapping (Micro-Partitions & Data Clustering — Snowflake Documentation, n.d.). Clustering depth is an indicator of clustering health, which can diminish over time as a result of DML being performed on the table (Micro-Partitions & Data Clustering — Snowflake Documentation, n.d.). Figure 11 illustrates the data clustering of a table consisting of 24 rows, stored by columns, equally across 4 micro-partitions. Snowflake can prune the micro-partitions that are not necessary for the query, and then prune the remaining partitions by column. If tables become very large and the data is not suitably ordered, query performance can become noticeably diminished, clustering keys can also be defined to offer overall query improvement.

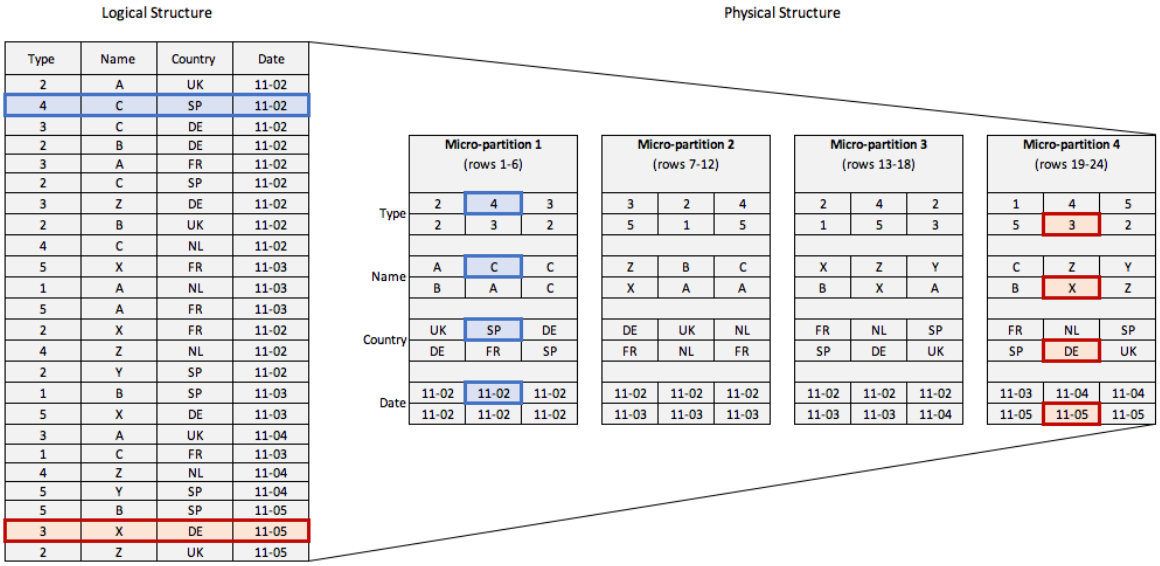


Figure 11 *Logical and physical representation of micro-clustering in Snowflake data warehousing*

Clustering keys are explicitly designated table columns that can be leveraged to co-locate data in the table within the same micro-partitions, this can also be useful if the clustering depth is large (Clustering

Keys & Clustered Tables — Snowflake Documentation, n.d.). Clustering keys offer efficiency improvements when scanning by avoiding data that does not match filtering predicates (Clustering Keys & Clustered Tables — Snowflake Documentation, n.d.). Furthermore, clustering keys provide improved table compression and require no additional maintenance by users once deployed, instead Snowflake automatically ensure optimal clustering performance (Clustering Keys & Clustered Tables — Snowflake Documentation, n.d.). It is important to consider the use of clustering keys carefully as they will consume higher compute resources meaning they should only be created where there is a substantial benefit. Common SQL `where` queries sort on the table clustering key with an `order by`, `group by`, and some `join` operations (Clustering Keys & Clustered Tables — Snowflake Documentation, n.d.).

4.8.3 Implementing Data Security and Privacy

Specific security and privacy features range from Snowflake's commitment to the principle of security by default, which includes encryption at rest and Single Sign-On (SSO) integration with Multi-Factor Authentication (MFA), to the ability to restrict access to tables or views by IP address and role-based administration access through the deployment of two independent roles:

1. `SYSADMIN` with permissions for granting privileges against databases, schemas, data queries, views, and procedures
2. `SECURITY ADMIN` used for granting privileges and permissions for user profiles

The default setting in Snowflake allows for user connection with any device's IP address (*Network Policies — Snowflake Documentation*, n.d.). To provide extra security for the company's production data, a network policy has been enabled that *whitelists* the IP addresses of the cloud version of the BI solution deployed, in addition to the company's head office router to enable connections from the desktop version of the BI solution. In addition to allowed IP address ranges, Snowflake allows administrators to block ranges of IP addresses, for instance, all IP addresses can be blocked except for the current IP address where necessary (*Network Policies — Snowflake Documentation*, n.d.).

The Snowflake solution considers the governance requirements of businesses and incorporates supporting tools. Dynamic Data Masking is one such tool that allows security and privacy officers to implement masking policies that protect column data, prohibiting users from unnecessarily viewing sensitive information from databases (*Using Dynamic Data Masking — Snowflake Documentation, n.d.*).

Data masking uses encryption and data shuffling techniques to create a randomised, inauthentic version of PII, effectively obfuscating an individual's identity (*Using Dynamic Data Masking — Snowflake Documentation, n.d.*). Analysis of the masked data will yield the same results as the original data set. Masking can be used to add another layer of security to data anonymisation, showing only the most relevant column information to authorised data handlers (*What Is Data Anonymization? Definition and FAQs | OmniSci, n.d.*). The process is implemented through a custom role authorised to create and define a masking policy that can be applied to columns across the data warehouse (*Understanding Dynamic Data Masking — Snowflake Documentation*, n.d.).

4.8.4 API Integration

The data pipeline deploys several API tools for ingesting data from the sales application and other solutions such as currency conversion rates. Snowflake allows API objects to be created in the company's account that stores information about the `HTTPS` proxy service, such as the cloud provider, the type of service, and identifier and access credentials (*CREATE API INTEGRATION — Snowflake Documentation*, n.d.). The API object includes several security features that include resources on the proxy service that can be accessed only by those users that have been granted the appropriate privileges. Specific resources and endpoints can also be allowed, or blocked, on the proxy service to prevent unnecessary user fees from being incurred (*CREATE API INTEGRATION — Snowflake Documentation*, n.d.).

4.9 Data Analytics for Business Insights

The data warehouse deployment will be considered a success if it is capable of disseminating information from various organisational or business units for business intelligence, whilst eliminating operational impact on OLTP databases (Castellanos et al., 2009). Castellanos et al. (2009) referenced the proliferation of relational OLTP databases, deployed across disparate systems within organisations, as the reason for the increased demand for online reporting and data analytics for high-level decision making. Castellanos et al. (2009) further describe the increasing demand placed on these systems as unsustainable, with the extraction of data from multiple sources as being a labour-intensive task often prone to errors. The more these systems are relied upon, the more detrimental the impact on the day-to-day business operations

resulting from the large table scans and data aggregation that consumes I/O resources (Castellanos et al., 2009).

The ability for data warehousing to carry out large scale analytic tasks for data-driven strategic decision making has expanded the role of data analysts, who are now expected to extract useful information from the raw data (Castellanos et al., 2009). Castellanos et al. (2009) highlight data mining as an emerging technology to leverage statistical techniques and machine learning algorithms, to assist with the sophisticated analysis of historic patterns across organisations. The success of the technologies deployed for business intelligence inevitable increases the demand, it is not simply the case that they are to achieve meaningful insights, they must be delivered in real-time (Castellanos et al., 2009). Castellanos et al. (2009) conclude that real-time business intelligence is playing an increasingly important role in routine business operations, the degree of timeliness of this real-time information must be justified to prevent prohibitive costs and unnecessary resources allocated to the system.

4.9.1 Comparative Analysis

A variety of data analytic tools has been tested by the project team including Power BI, Tableau, Metabase, and Qlik. A preliminary comparative analysis was conducted of these four industry-leading tools followed by extensive testing, the comparative analysis is provided in Appendix U. Research of the services included an investigation that included the following metrics:

- Ease of integration with company deployed SQL databases and applications
- Set-up time and time to deployment
- The ability to create meaningful data using SQL and replication using the GUI environment
- Automation of charts and figures that reference monthly, quarterly, and annual results
- Alerts and notifications for triggers identified as important to business stakeholders
- Dashboard and result visualisation, both within a secure environment and for publication online through a web interface, is compatible with mobile devices, large office monitors, and for insertion into presentation documents
- Quantity and quality of the features available, particularly surrounding the ability to visualise data for non-technical business users
- Consideration to cost that includes user profiling for license requirements
- Security and privacy features with specific consideration to GDPR and CCPA
- Company branding, including colour schemes and fonts for organisational consistency

The research conducted provided a sufficient overview of the offerings that narrowed the list for testing, which began with the removal of Power BI. This was a technical decision aimed at limiting the exposure to Microsoft products as a result of the corporation's perceived reputation for forcing users to stay within their technology ecosystem. Testing was conducted on Tableau, Metabase, and Qlik.

4.9.2 Technical and Functional Requirements Analysis

Several factors became obvious as a result of the systems testing. Firstly, the two cloud-based solutions, Metabase and Qlik, were simpler to set up and integrate with the wider technology infrastructure. Tableau, in comparison, required the installation of a desktop application followed by an ODBC driver. The solution will require installation support for non-technical users, which, when combined with a system that is stored locally, reduces the desirability of the product that effectively increases the attack surface area from a security perspective. SOC 2 Type-II requires that the company's physical assets be encrypted, have strong password access, and be secured in premises that are covered by CCTV, this will mitigate the security risk. Despite these drawbacks, Tableau does allow for the creation of data visualisations in the cloud. However, many key features are only available in the desktop instance, with the dashboard being uploaded to the cloud. Secondly, Tableau benefits from a GUI environment that is similar to Power BI, being user friendly and simple to operate for non-technical users, it is also familiar by members of the data team. Thirdly, with Tableau and Metabase, visualisations can be created with SQL queries, meaning charts to be developed more quickly by those skilled in the language. Metabase and Qlik take very different approaches to Tableau and Power BI, requiring more testing to determine if there were additional benefits that could be realised.

Metabase met several business and technical requirements. Firstly, it is more affordable, and secondly, it has an interface with reduced complexity and feature clutter. Metabase deploys SQL structure as part of the user interface in the form of *tick boxes* that run queries against the data, despite this integrated GUI feature, knowledge of SQL would still be required, making it redundant. Ultimately, the simple design proved to be a result of missing features that were desired by the team, limiting the ability to future proof the data pipeline. Qlik proved to be a different approach to data analytics, deploying an *associative engine* that seeks to do the heavy lifting through the deployment of machine learning to understand the data and automatically generate data visuals, requiring little or no input from the user. Qlik is an impressive

application with a unique pricing structure that is based on the compute time of the users actively engaged with the product, should analytics be rolled out across the company, it had the potential to significantly reduce operating costs. However, following the testing period and a demonstration meeting displaying the features, it was decided that the complexity would require a greater degree of hands-on training for non-technical users to get the best out of the solution. The conclusion of the testing period led to Tableau being deployed in the data pipeline, despite the drawbacks.

4.9.3 Generating Meaningful Business Intelligence

Two functional requirements were specified for the data visualisations. Firstly, automation, data must be displayed for the month, quarter, year, and region with no input for business users. Secondly, data manipulation, business users should be able to filter the data to view previous months, quarters, or compare figures against last year's results. Technical users' priority was on automation, with an expectation that minimal or no further input would be necessary, except where changes are requested. However, business users need to manipulate the data to gain the insights they desire. Initial systems development indicated that these were conflicting requirements, requiring further development work to satisfy both. The sales and revenue dashboard went through numerous design and functional iterations to meet these needs, ultimately dashboard insights were split for visualisations that required automation, and those that required manipulation.

After completion of the sales and revenue dashboards, requests were made for sales representative figures to be made available and be displayed by the current month compared against targets, this resulted in further challenges. Sales targets are calculated against base employee salary meaning they are private to the individual. A dashboard had to be designed that allowed total sales representative data to be displayed for management stakeholders, whilst allowing individual representatives to access only their data. To achieve this Row Level Security (RLS) was deployed that utilised user filtering, allowing for rows to be made visible only to specific users and removing the need to create multiple dashboards for each user, which would have been time-consuming and require significant management when onboarding and offboarding new sales representatives. Figure 12 is an example of one of the dashboards created, using readily available open-source data.

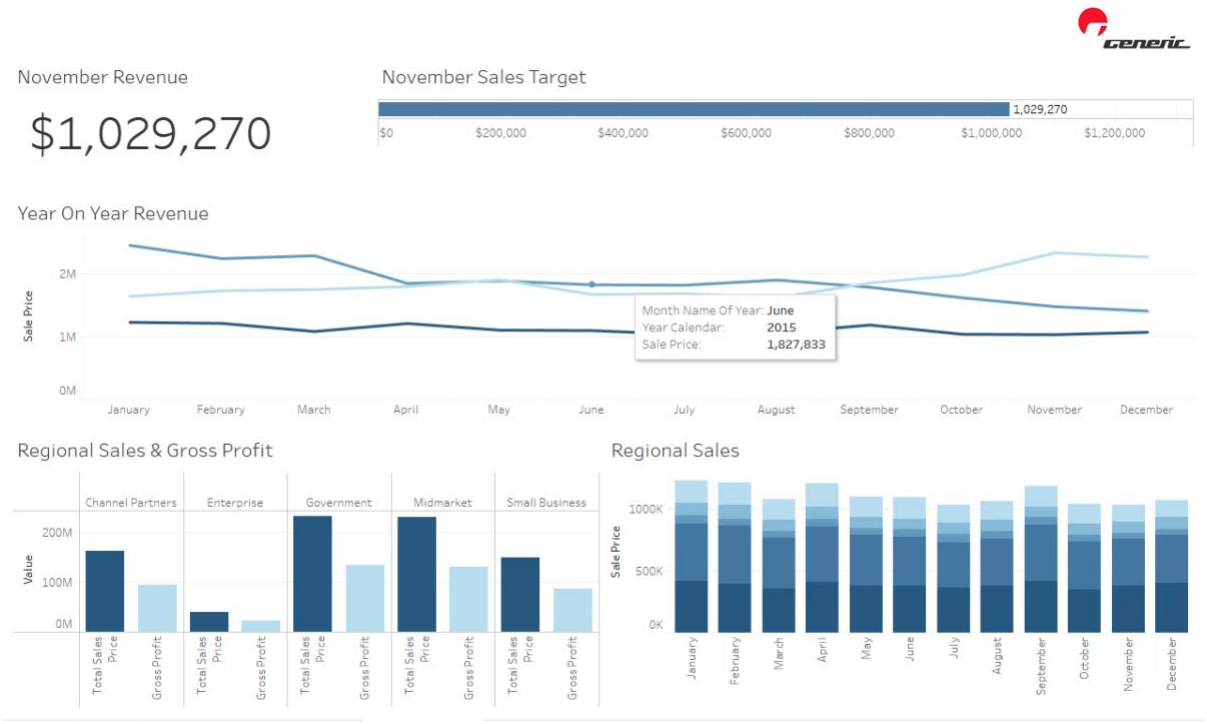


Figure 12 *Visual representation of project dashboard relating to sales and revenue data*

5. CONCLUSION

Conclusions are drawn based on the objectives set out in section 3 of this report. Three key results were identified, [TO1] a data pipeline providing near real-time sales and revenue data, [TO2] data at a granularity level capable of producing gross profit at the individual deal level, and [TO3] visualisations that could be accessed on desktop and mobile devices and manipulated without knowledge of SQL. *TO1* and *TO3* were successful, however, the team recognised the overly ambitious nature of objective *TO2*, and the lack of time and resources to complete this objective. Several factors contributed to this outcome, firstly, the project was ambitious in the scope given the regular work commitments of each team member. Secondly, the development of the data pipeline was a complex undertaking that had not been previously conducted, and finally, the process of data validation and ensuring data quality consumed more time than anticipated. It has been agreed that *TO2* will be the primary objective for the next quarter, with work already proceeding on data mapping. Much of this development work will fall outside the scope of this project report.

Creating data visualisations was a key result of the author [IKR1]. Being the end of the pipeline, this was the area that would be viewed and interacted with by the widest set of users across the company, it was therefore essential that requirements were met. It is recognised that business intelligence is only as successful as the availability and quality of the data. To maintain the success of the project, the standards realised to date must be maintained as the project advances into the second and more complex set of objectives.

The research objectives set for this project involved gaining clarity and understanding of the methods, functions, and theories that make up the processes of the data pipeline, within the framework of the regulatory obligations as they apply to the company.

RO1: Manual data mapping was deployed successfully to extract the relevant data needed for the analytics. Comparatively, this was an uncomplex data set that offered challenges when attempting to identify data fields and transforming the data from JSON to SQL. The research conducted in this area will offer more value in the forthcoming quarter through the integration of a larger set of data sources.

RO2: The structure of the data warehouse is based on a fact constellation schema that includes several aggregated values that support the data analytics. The iterative nature of the Scrum methodology means

that it is unclear how this schema will develop over the coming quarter, as more data sources are integrated into the pipeline. The added complexity and demands placed on the data warehouse may lead to the current schema being deemed insufficient to meet the challenges, and the data may be structured into data marts designed to serve the varying and separate needs of the business units.

RO3: The data pipeline successfully deploys an ELT process to meet the technical and business requirements. The research indicates that this may not offer sufficient privacy protection surrounding PII, or security resulting from the increased attack surface area. Currently, limited PII is being ingested into the data warehouse, and the PII that is ingested meets the legal basis for processing. As more personal data is included as part of the next quarterly objectives, a greater emphasis will be placed on whether to retain the ELT pipeline, move to an ETL pipeline, or implement a combination of the two.

RO4: The data set includes several anomalies identified in the research. The closer the team reached the point of verified data, the more challenging the anomalies became, an increasing amount of effort was expended comparatively to the degree of accuracy gained. This was partly due to the type of anomalies such as missing values and unintentional misuse of the system that resulted in errors. The method deployed, prioritising work that offered the most value in the shortest time, amplified this, the team dealt with simpler fixes first that gained the highest result output to meet sprint goals. Again, it is anticipated that the degree of anomalies will be significantly higher and more complex to solve in the next quarterly iteration.

RO5: The investigation into anonymisation and pseudonymisation has produced mixed results. The current data set did not ultimately deploy either technique, instead choosing to deploy the principle of data minimisation. This was possible as *TO2* was unachievable and this is where the greatest ingestion of PII exists. A preliminary reflection indicates that anonymisation would be the preferred option, provided the level of granularity does not impact the ability of the analytics to answer the business questions.

RO6: The investigation into cloud data warehouse solutions supports the company's decision to operate the data pipeline in a cloud environment. This allowed the team to produce two out of the three key results within the time frame, and meet the security and privacy obligations. The benefits of a cloud infrastructure outlined in the literature have been supported through the practical application of the tools and processes deployed. Any future planned changes to service providers will not as a result of their cloud-based status.

RO7: The data analytics tool deployed by the company had the potential to consume a significant portion of the project cost, particularly if licences are extended to each employee. Despite the comparative analysis indicating Tableau as being the more expensive solution, the cost has been mitigated in two ways. Firstly, a single year licence has been applied to only those employees that currently require access, allowing further users to be incrementally added throughout the year, and secondly, exploring the features of the application may allow for analytics to be displayed on large monitors that can be shared and viewed by multiple users, reducing the number of accounts that are ultimately required. Conducting a thorough analysis of the features and integrations, in conjunction with the testing of each solution was extremely valuable for determining which would be the preferred application.

6. RECOMMENDATIONS

The project roadmap extends beyond the structured data warehouse and data analytics, referencing two areas that are likely to be beneficial to the organisation. The first is the creation of a data lake that can be deployed for analysis. The nature of the business model means the company collects a great deal of data relating to the creative products, such as video files, images, and other creative content. The file sizes of this content, and its unorganised structure, make a data lake an attractive proposition. A description of data lakes has been provided in Appendix V. The second recommendation from the roadmap surrounds the best deployment of machine learning to improve insights and data intelligence. The team is looking to grow over the next quarter and the implementation of machine learning is being used to attract talent to the company. Machine learning has the potential to introduce as many problems as it solves, this is particularly true regarding data privacy and security. A preliminary investigation into machine learning is available in Appendix W.

REFERENCES

Ariyachandra, T., & Watson, H. (2010). Decision Support Systems: Key organizational factors in data warehouse architecture selection. *Elsevier B.V.*, *49*. https://doi.org/10.1016/j.dss.2010.02.006

Art. 6 GDPR – Lawfulness of processing. (n.d.). *General Data Protection Regulation (GDPR)*. Retrieved October 3, 2021, from https://gdpr-info.eu/art-6-gdpr/

Art. 35 GDPR – Data protection impact assessment. (n.d.). *General Data Protection Regulation (GDPR)*. Retrieved October 3, 2021, from https://gdpr-info.eu/art-35-gdpr/

Art. 37 GDPR – Designation of the data protection officer. (n.d.). *General Data Protection Regulation (GDPR)*. Retrieved October 2, 2021, from https://gdpr-info.eu/art-37-gdpr/

Batch Processing—A Beginner's Guide | Talend. (n.d.). Retrieved September 26, 2021, from https://www.talend.com/resources/batch-processing/

Bologa, A.-R., & Bologa, R. (2011). Integrating Data Sources from Different Development Environments: An E-LT Approach. *Quality - Access to Success*, *11*(123).

Bourka, A., Prokopios, D., & Agrafiotis, I. (2019). Pseudonymisation techniques and best practices: Recommendations on shaping technology according to data protection and privacy provisions. *The European Union Agency for Cybersecurity (ENISA)*. https://doi.org/10.2824/247711

Castellanos, M., Dayal, U., Pedersen, T. B., & Tatbul, N. (2015). *Enabling Real-Time Business Intelligence: International Workshops, BIRTE 2013, Riva del Garda, Italy, August 26, 2013, and BIRTE 2014, Hangzhou, China, September 1, 2014, Revised Selected Papers* (1st ed., Vol. 206). Springer-Verlag Berlin Heidelberg.

Castellanos, M., Dayal, U., & Sellis, T. (2009). *Business Intelligence for the Real-Time Enterprise: Second International Workshop, BIRTE 2008, Auckland, New Zealand, August 24, 2008, Revised Selected Papers* (1st ed.). Springer-Verlag Berlin Heidelberg.

Clustering Keys & Clustered Tables—Snowflake Documentation. (n.d.). Retrieved September 26, 2021, from https://docs.snowflake.com/en/user-guide/tables-clustering-keys.html

CREATE API INTEGRATION — Snowflake Documentation. (n.d.). Retrieved October 25, 2021, from https://docs.snowflake.com/en/sql-reference/sql/create-api-integration.html

Data Anonymization Techniques and Best Practices: A Quick Guide. (2020, July 29). *Record Evolution*. https://www.record-evolution.de/en/data-anonymization-techniques-and-best-practices-a-quick-guide/

Data Transformation and Data Quality. (n.d.). Stitch. Retrieved November 10, 2021, from https://www.stitchdata.com/platform/datatransformation/

Data Warehouse Tutorial For Beginners | Data Warehouse Concepts | Data Warehousing | Edureka—YouTube. (n.d.). Retrieved October 8, 2021, from https://www.youtube.com/watch?v=J326LIUrZM8

Data Warehousing—Schemas. (n.d.). Retrieved September 26, 2021, from https://www.tutorialspoint.com/dwh/dwh_schemas.htm

Dbt for Data Transformation – Hands-on Tutorial. (n.d.). *KDnuggets*. Retrieved November 10, 2021, from https://www.kdnuggets.com/dbt-for-data-transformation-hands-on-tutorial.html/

De Rougemont, M., & Cao, P. T. (2012). Approximate answers to OLAP queries on streaming data warehouses. *Proceedings of the Fifteenth International Workshop on Data Warehousing and OLAP*, 121–128. https://doi.org/10.1145/2390045.2390065

Deemer, P., Benefield, G., Larman, C., & Vodde, B. (2012). *The Scrum Primer: A Lightweight Guide to the Theory and Practice of Scrum* (2.0). InfoQ Enterprise Software Development Series. http://www.infoq.com/minibooks/Scrum_Primer

Dimensions and facts. (n.d.). Retrieved September 26, 2021, from https://www.linkedin.com/learning/implementing-a-data-warehouse-sql-server-2019/dimensions-and-facts

Eder, J., Koncilia, C., & Morzy, T. (2001). A Model for a Temporal Data Warehouse. *Proceedings of OES-SEO 2001 Workshop*, pp 48-54.

Esayas, S. Y. (2015). The role of anonymisation and pseudonymisation under the EU data privacy rules: Beyond the 'all or nothing' approach. *European Journal of Law and Technology*, *6*(2).

EU-US Privacy Shield for data struck down by court. (2020, July 16). *BBC News*. https://www.bbc.com/news/technology-53418898

Garzaniti, N., Briatore, S., Fortin, C., & Golkar, A. (2019). Effectiveness of the Scrum Methodology for Agile Development of Space Hardware. *IEEE Aerospace Conference*, 1–8. https://doi.org/10.1109/AERO.2019.8741892

Hinely, M. (2018, August 23). *6 Legal Bases for Processing Personal Data: GDPR Fundamentals | Video*. KirkpatrickPrice Home. https://kirkpatrickprice.com/video/gdpr-fundamentals-legal-basis-for-processing/

Hooks & Operations | dbt Docs. (n.d.). Retrieved November 10, 2021, from https://docs.getdbt.com/docs/building-a-dbt-project/hooks-operations

Hota, J. (2011). Business Analytics: A tool for Organizational Transformation. *CSI Communications*. http://ssrn.com/abstract=2162509

IBM Docs. (2021a, March 8). https://prod.ibmdocs-production-dal-6099123ce774e592a519d7c33db8265e-0000.us-south.containers.appdomain.cloud/docs/en/ida/9.1?topic=phase-step-identify-business-process-requirements

IBM Docs. (2021b, March 8). https://prod.ibmdocs-production-dal-6099123ce774e592a519d7c33db8265e-0000.us-south.containers.appdomain.cloud/docs/en/ida/9.1?topic=phase-step-identify-grain

IBM Docs. (2021c, July 29). https://prod.ibmdocs-production-dal-6099123ce774e592a519d7c33db8265e-0000.us-south.containers.appdomain.cloud/docs/en/spm/7.0.11?topic=explained-etl

IT k Funde. (2020, August 31). *What is Data Pipeline | How to design Data Pipeline ? - ETL vs Data pipeline*. https://www.youtube.com/watch?v=VtzvF17ysbc

K-Anonymity: Everything You Need to Know (2021 Guide). (2021, April 14). Immuta. https://www.immuta.com/articles/k-anonymity-everything-you-need-to-know-2021-guide/

kexugit. (n.d.). *SQL Server—Data Quality Testing Using SQL Server 2012 Data Quality Services*. Retrieved September 26, 2021, from https://docs.microsoft.com/en-us/archive/msdn-magazine/2012/december/sql-server-data-quality-testing-using-sql-server-2012-data-quality-services

Kimball, R., & Ross, M. (2002). *The Data Warehouse Toolkit—Second Edition: The Complete Guide to Dimensional Modeling*. John Wiley & Sons, Inc.

La Rosa, M., Sadiq, S., & Teniente, E. (2011). *Advanced Information Systems Engineering: 33rd International Conference, CAiSE 2021, Melbourne, VIC, Australia, June 28 – July 2, 2021, Proceedings* (Vol. 12751). Springer International Publishing.

Li, H., Yu, L., & Wu, H. (2019). *The Impact of GDPR on Global Technology Development, Journal of Global Information Technology Management*. *22*(1), 1–6. https://doi.org/10.1080/1097198X.2019.1569186

Ly, D. H. (2019). *Data analytics in cloud data warehousing: Case company*. Metropolia University of Applied Sciences.

Marín-Ortega, P. M., Dmitriyev, V., Abilov, M., & Gómez, J. M. (2014). ELTA: New Approach in Designing Business Intelligence Solutions in Era of Big Data. *Elsevier Lt*, *16*, 667–674. https://doi.org/10.1016/j.protcy.2014.10.015

Metabase vs Microsoft Power BI 2021—Feature and Pricing Comparison on Capterra. (n.d.). Retrieved October 2, 2021, from https://www.capterra.com/business-intelligence-software/compare/176651-176586/Metabase-vs-Power-BI

Micro-partitions & Data Clustering—Snowflake Documentation. (n.d.). Retrieved September 26, 2021, from https://docs.snowflake.com/en/user-guide/tables-clustering-micropartitions.html

Moody, D. L., & Kortink, M. A. R. (2000). *From Enterprise Models to Dimensional Models: A Methodology for Data Warehouse and Data Mart Design*. *Proceedings of the International Workshop on Design and Management of Data Warehouses (DMDW'2000)*.

Müller, H., & Freytag, J.-C. (2003). Problems, methods, and challenges in comprehensive data cleansing. *Humboldt-Universität Zu Berlin Zu Berlin, 10099 Berlin, Germany*.

Network Policies—Snowflake Documentation. (n.d.). Retrieved October 25, 2021, from https://docs.snowflake.com/en/user-guide/network-policies.html

Park, Y.-T. (2005). An empirical investigation of the effects of data warehousing on decision performance. *Elsevier B.V.*, *Information&Management 4*, 51–61. https://doi.org/10.1016/j.im.2005.03.001

Ranjan, J. (2008). Business justification with business intelligence. *Emerald Group Publishing Limited*, *38*(4), 461–475. https://doi.org/0.1108/03055720810917714

Rifaie, M., Kianmehr, K., Alhajj, R., & Ridley, M. J. (2008). Data Warehouse Architecture and Design. *IEEE IRI 2008*.

Skills a Data Protection Officer Must Have. (2021, February 4). Arkansas State University Online. https://degree.astate.edu/articles/media-management/data-protection-officer-skills.aspx

Smallcombe, M. (n.d.). *ETL vs ELT: 5 Critical Differences*. Xplenty. Retrieved September 26, 2021, from https://www.xplenty.com/blog/etl-vs-elt/

Stanoevska-Slabeva, K., Wozniak, T., & Ristol, S. (2010). *Grid and Cloud Computing: A Business Perspective on Technology and Applications*. Springer.

Stitch Platform: Enterprise-grade security and compliance. (n.d.-a). Stitch. Retrieved November 10, 2021, from https://www.stitchdata.com/platform/security/

Stitch Platform: Extensibility with Singer and the Import API. (n.d.-b). Stitch. Retrieved November 10, 2021, from https://www.stitchdata.com/platform/extensibility/

Stitch Platform: Orchestration. (n.d.-c). Stitch. Retrieved November 10, 2021, from https://www.stitchdata.com/platform/orchestration/

Stitch Platform: Performance and Reliability. (n.d.-d). Stitch. Retrieved November 10, 2021, from https://www.stitchdata.com/platform/performance/

Sweeney, L. (2002a). ACHIEVING k-ANONYMITY PRIVACY PROTECTION USING GENERALIZATION AND SUPPRESSION. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, *10*(05), 571–588. https://doi.org/10.1142/S021848850200165X

Sweeney, L. (2002b). k-ANONYMITY: A MODEL FOR PROTECTING PRIVACY. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, *10*(05), 557–570. https://doi.org/10.1142/S0218488502001648

Tankard, C. (2016). What the GDPR means for businesses. *Digital Pathways*.

The API Design Platform and API Client. (n.d.). Retrieved October 5, 2021, from https://insomnia.rest/

The Career Force. (2019, May 31). *What is a Data Lake*. https://www.youtube.com/watch?v=1H0pXzfnH28

The Essential Guide To Data Mapping. (n.d.). Tableau. Retrieved September 26, 2021, from https://www.tableau.com/learn/articles/guide-to-data-mapping

Third Countries. (n.d.). *General Data Protection Regulation (GDPR)*. Retrieved September 26, 2021, from https://gdpr-info.eu/issues/third-countries/

Trappenberg, T. P. (2019). Fundamentals of Machine Learning. *Oxford University Press (2020)*, 272. https://doi.org/10.1093/oso/9780198828044.001.0001

Understanding Dynamic Data Masking—Snowflake Documentation. (n.d.). Retrieved September 26, 2021, from https://docs.snowflake.com/en/user-guide/security-column-ddm-intro.html

Understanding the 7 Principles of the GDPR - Blog—OneTrust. (n.d.). Retrieved October 2, 2021, from https://www.onetrust.com/blog/gdpr-principles/

Using Dynamic Data Masking—Snowflake Documentation. (n.d.). Retrieved September 26, 2021, from https://docs.snowflake.com/en/user-guide/security-column-ddm-use.html

V, D. & erweide. (2019, April 15). OpEx vs. CapEx: The Real Cloud Computing Cost Advantage. *10th Magnitude*. https://www.10thmagnitude.com/opex-vs-capex-the-real-cloud-computing-cost-advantage/

What is a Data Lake? - YouTube. (n.d.). Retrieved September 26, 2021, from https://www.youtube.com/watch?v=LxcH6z8TFpI

What is Data Anonymization? Definition and FAQs | OmniSci. (n.d.). Retrieved September 26, 2021, from https://www.omnisci.com/technical-glossary/data-anonymization

What is Data Mapping? Definition and Examples | Talend. (n.d.). Talend - A Leader in Data Integration & Data Integrity. Retrieved September 26, 2021, from https://www.talend.com/resources/data-mapping/

What is Data Modeling? | IBM. (n.d.). Retrieved September 26, 2021, from https://www.ibm.com/cloud/learn/data-modeling

What is dbt? | dbt Docs. (n.d.). Retrieved October 9, 2021, from https://docs.getdbt.com/docs/introduction

What Is Pseudonymisation? | Thales. (n.d.). Retrieved September 26, 2021, from https://cpl.thalesgroup.com/faq/data-protection-security-regulations/what-pseudonymisation

What is REST. (n.d.). REST API Tutorial. Retrieved October 3, 2021, from https://restfulapi.net/

Why shift from ETL to ELT? (2016, March 18). *Hexanika*. https://hexanika.com/why-shift-from-etl-to-elt/

BIBLIOGRAPHY

Art. 9 GDPR – Processing of special categories of personal data. (n.d.). *General Data Protection Regulation (GDPR)*. Retrieved October 2, 2021, from https://gdpr-info.eu/art-9-gdpr/

Atlassian. (n.d.). *Jira Service Management | A new take on ITSM software*. Atlassian. Retrieved October 2, 2021, from https://www.atlassian.com/software/jira/service-management

Create a User Filter and Secure it for Publishing. (n.d.). Retrieved October 25, 2021, from https://help.tableau.com/current/pro/desktop/en-us/publish_userfilters_create.htm

Data Engineering and Its Main Concepts: Explaining the Data Pipeline, Data Warehouse, and Data Engineer Role. (n.d.). *AltexSoft*. Retrieved September 26, 2021, from https://www.altexsoft.com/blog/datascience/what-is-data-engineering-explaining-data-pipeline-data-warehouse-and-data-engineer-role/

Data Protection Impact Assessment (DPIA). (2018, August 9). GDPR.Eu. https://gdpr.eu/data-protection-impact-assessment-template/

Data Tracking: How to Create a Successful Data Tracking Plan. (n.d.). Segment. Retrieved September 26, 2021, from https://segment.com/academy/collecting-data/how-to-create-a-tracking-plan/

Dbt Explained | Blog. (2021, July 12). Fivetran. https://fivetran.com/blog/dbt-explained

Soliman, M. A., Antova, L., Sugiyama, M., Duller, M., Aleyasen, A., Mitra, G., Abdelhamid, E., Morcos, M., Gage, M., Korablev, D., & Waas, F. M. (2020). A Framework for Emulating Database Operations in Cloud Data Warehouses. *Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data*, 1447–1461. https://doi.org/10.1145/3318464.3386128

Welcome to Stitch Documentation | Stitch Documentation. (n.d.). Stitch Docs. Retrieved November 10, 2021, from https://www.stitchdata.com/docs

What's the difference between homogeneous and heterogeneous data sets? (n.d.). Quora. Retrieved September 26, 2021, from https://www.quora.com/Whats-the-difference-between-homogeneous-and-heterogeneous-data-sets

APPENDICES

Appendix A – Description of the legal basis for the collection, the triggers, and compliance requirements under the GDPR

The legal basis for the collection of data:

1. Consent from the data subject
2. To perform contractual obligations
3. There is a legitimate interest
4. It is of vital interest
5. There is a legal requirement
6. It is of public interest

Conditions that would trigger the requirement for a DPIA:

- When deploying new technologies
- If the location or behaviour of data subjects is being tracked
- If a publicly accessible location is being monitored systematically on a large scale
- If special category data as defined under Article 9 is being processed
- If data relating to children is being processed
- If the processing of data could lead to physical harm is leaked following a data breach
- If the processing of data is used for automated decisions about data subjects that could lead to legal consequences

The GDPR does not specify the use of a DPIA for these processes alone. It is reasonable for organisations to consider conducting a DPIA to mitigate risk and limit any liability in the event a data breach triggers regulatory intervention. Assessments can be performed as part of wider data security and privacy standards and best practices. To comply with the requirements of the Article, a DPIA should contain the following elements:

- The data controllers legitimate interest in the collection and processing of the data
- A systematic description of the planned processing operations and the purpose of processing
- An assessment regarding the purposes of the processing, specifically assessing the necessity and proportionality of the project
- An assessment concerning the data subjects and the impact on their rights and freedoms

It is important to understand these elements so that measures, safeguards, and mechanisms can be formulated and implemented to mitigate the risk. Documenting these steps is essential to ensure organisations can demonstrate compliance if necessary. At all times, the DPIA should be completed in

consultation with the organisation's DPO if one exists, and the stakeholders that have an interest in the project outcome ("Art. 35 GDPR – Data Protection Impact Assessment," n.d.).

Appendix B – Description of the defined roles outlined by the Scrum Methodology

The *Product Owner* is responsible for maximising the return on investment (ROI). This is not the same definition as ROI for a commercial product as the project is internal to the organisation. The ROI, instead, focuses on choosing the highest value requirements and features through a process of continual re-prioritising and refining for the forthcoming sprint.

The *Development Team* is involved in building the product as directed by the *Product Owner*. The scope of the project requires the team to be cross-functional and interdisciplinary to ensure it contains the necessary skills, and expertise, to achieve the OKRs. The sprints afford the team a high degree of autonomy and accountability, requiring members to be self-organising in the responsibility of their tasks. The structure of the team does not allow for fixed or specialists titles, members work together appropriately to achieve the targets set.

The *ScrumMaster* supports the team through the learning and application of Scrum, to achieve maximum business value from its implementation. The *ScrumMaster* serves the team by assisting them in the adoption of new technologies, removing impediments, and protecting the team from interference from outside influences. They are often viewed as educators or guides through the difficult process of systems development ("Art. 35 GDPR – Data Protection Impact Assessment," n.d.).

Appendix C – Description of the Data Mapping Techniques

Manual data mapping requires the identification and connection of data sources, this process should be mapped and documented. There are several benefits to a manual data mapping process, firstly, it can be customised to meet the use case requirements, and secondly, it is flexible and places control in the hands of the data professionals. There are, however, several drawbacks. The work must be carried out by professionals, which can be time-consuming and can make it difficult to justify to business stakeholders looking for a quick solution, it is also resource-intensive and code-dependent.

Automated data mapping deploys platforms that are designed to perform the heavy lifting that would otherwise be handled by data professionals. Automated tools can be deployed by both professionals and

business analysts with limited technical knowledge, requiring zero coding experience. Depending on the use case and skill level of the data team, data mapping platforms are often quicker to implement, offer greater scalability, are easier to schedule, and can be more flexible in their deployment. Some platforms now utilise natural language to match data fields and attributes to connect data sources, which can reduce the number of assumptions made that could lead to inaccurate results. Despite these platforms affording a lower barrier to entry, they often require training due to being tool specific, the costs of these services must also be considered when budgeting a project.

Semi-automated data mapping includes the use of graphical interfaces for data links, visual interface functionality allows data professionals to create schemas that produce output scripts in coding language, as would be produced in a manual process. This mapping process can offer a better balance of flexibility and effectiveness. However, it still requires a level of coding knowledge that requires a specific skill set. This makes it a resource-intensive option, requiring data professionals to navigate the process between manual elements and automation, having a clear understanding of where one process ends and the other begins to gain the greatest efficiency benefit (The Essential Guide To Data Mapping, n.d.).

Appendix D – Description of Dimensional Data Model Schemas

A star schema is made up *of dimension* tables surrounding a *fact* table that connects the *dimension* tables. The *fact* table contains the keys for each of the dimensions that make up the schema, *fact* tables also contain attributes. Dimensions in the star schema are represented with only one-dimension tables that contain the set of attributes associated with the dimension. This constraint can lead to data redundancy. For example, a location dimension may cause data redundancy with cities, states, or countries along with the corresponding attributes. The benefit of a star schema is that they only require one join to transverse the data to the context of the fact, which is less demanding on the query engine. In addition, a star schema is the simplest data warehouse schema to create. Figure 13 is a representation of a star schema.

A snowflake schema, unlike a star schema, contains some normalised *dimension* tables. For example, in the above-mentioned star schema, the location dimension suffered redundancy, normalisation of this table through the creation of a city dimension would reduce the redundancy, making it easier to maintain and save storage space. However, Snowflake schemas are more difficult for end-users to navigate and the additional joins are more taxing for the query engine. Figure 14 is a representation of a snowflake schema.

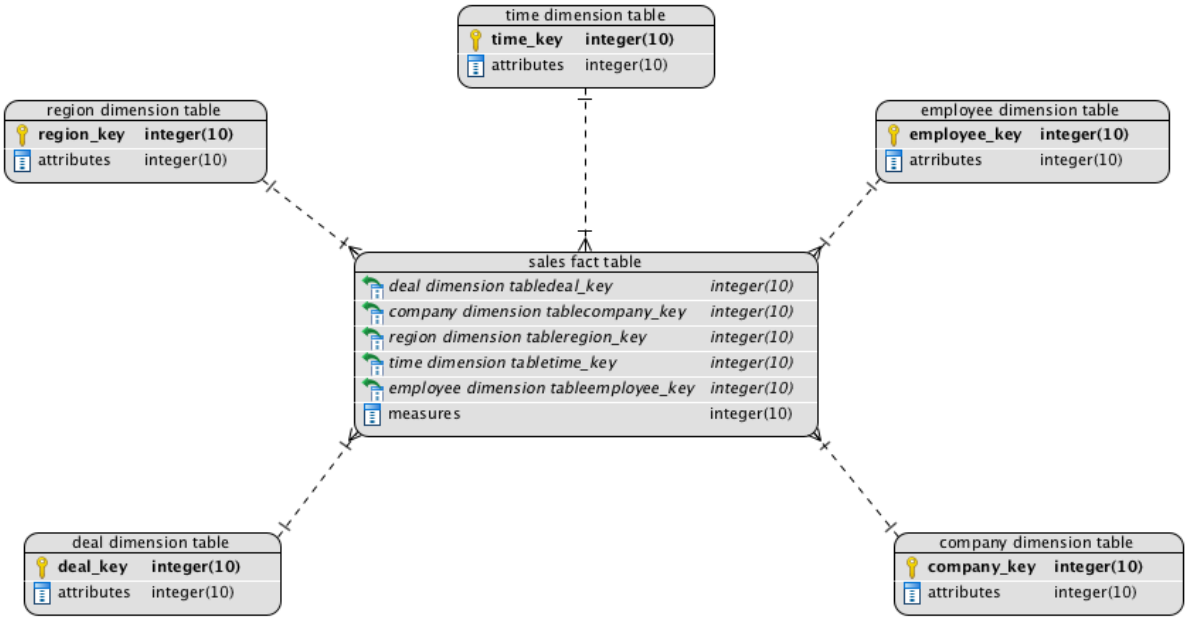


Figure 13 *Representation of a Star Schema*

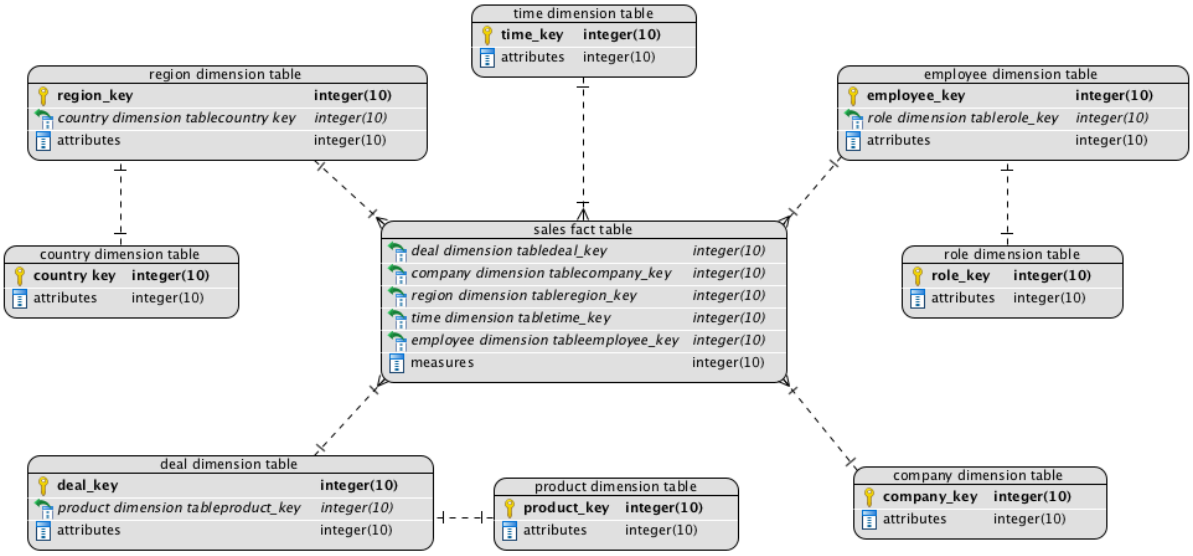


Figure 14 *Representation of a Snowflake Schema*

A fact constellation schema (also known as a galaxy schema) contains multiple *fact* tables. It is possible in these schemas to share *dimension* tables between the *fact* tables (Data Warehousing - Schemas, n.d.). Figure 15 is a representation of a fact constellation schema.

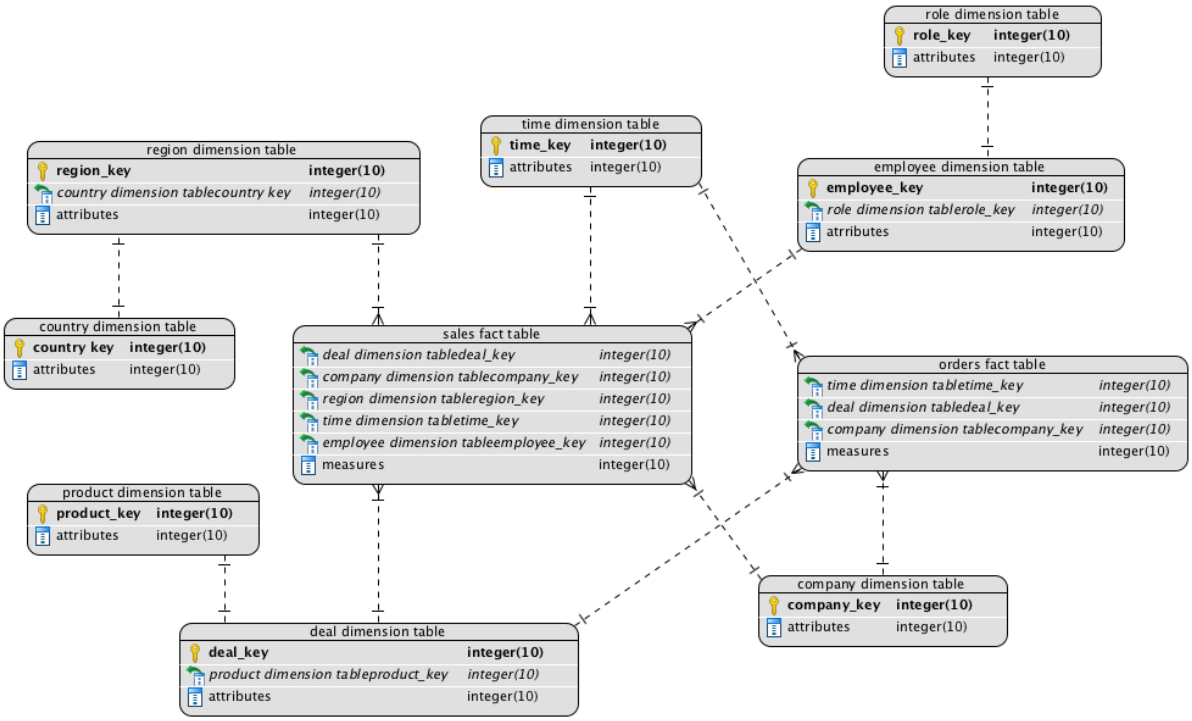


Figure 15 *Representation of a Fact Constellation Schema*

Appendix E –Naming structures in dimensional modelling

In a data warehouse, *fact* and *dimension* tables can deploy two types of naming structures. The tables can be defined by the naming structure of the tables, for example, *dimension* tables could be named as DimDate, DimCustomer, and DimEmployee, with *fact* tables named as FactSales, FactOrders, and FactCurrency. The alternative method is similar but uses the schemas in the database to make the distinction, creating a *fact* and a *dimension* schema. Taking the above example, these would be distinguished as Dimension.Date, Dimension.Customer, Dimension.Employee, Fact.Sales, Fact.Orders, and Fact.Currency. Despite the structural difference, both versions are performing the same function. Variations of these approaches are commonplace (Dimensions and Facts, n.d.).

Appendix F – Description of staging tables in an ETL pipeline

Staging tables are used for holding relational data temporarily for the ETL process, with data copied from the source location to the staging table during the extract process. The steps involved in transforming the data are performed on this copy of the data, once the transformation process is complete, the staging table is emptied as it is loaded to the data warehouse. A staging table can be beneficial when loading data

that is not sourced from a relation database, such as CSV file, text file, or JSON. Staging tables in the dimensional model schema are referred to as integration tables, examples might include Integration.City_Staging, Integration.Stock_Staging, and so on (Kimball & Ross, (2002).

Appendix G – Description of granularity types - input data, output data, default functionality, parameterised functionality, and business value

1. Input Data Granularity

The input data granularity reflects how much data is passed to the data warehouse. The more references passed into the data warehouse the coarser the grain, conversely, if no or fewer details are passed, then the finer the grain becomes. Coarser granularity can prove to be more beneficial if fewer transactions are required, reducing the overhead on the system. For instance, an *invoice* table that contains all the data relating to that invoice would have a coarser grain than an *address* table. In an OLAP data warehouse where query performance is considered of greater importance than storage costs, this coarser grain may be a satisfactory trade-off, despite the inherent redundancy.

2. Output Data Granularity

The output data granularity indicates the level of data that can be returned from varying queries. The more references returned by the query the coarser the grain, conversely, if no or fewer details are returned then the finer the grain becomes. Therefore, if a query was returning information on business clients from the *client* table, the grain would be coarser than if the query was returning birth dates of those clients from the *date* table. The use case will ultimately determine the level of granularity. However, a courser grain output can be beneficial for data reuse across multiple queries. It is still possible to keep queries small by discarding the data that is not required to answer the business question.

3. Default Functionality Granularity

Default functionality granularity is applied to the functionality offered by the deployed data warehouse, describing the level of functionality offered that cannot be adjusted through the inclusion of a set of parameters defined by the business needs or use case. For instance, a data warehouse that is capable of supporting a business process has a coarser grain than one that is only capable of performing part of that process.

4. Parameterised Functionality Granularity

The value of the data warehouse provider will also be determined by the level of parameterised functionality granularity. If the service is coarse-grained, meaning many services can be optioned to better meet a broader range of business needs, then the value of the service will increase. This classification does not simply relate to the number of optional functions, but also the types of functions to define the coarseness of the service.

5. Business Value Granularity

Business value granularity seeks to attach a business value to a service and describe the extent of that value across business units. Understanding where, in the business, the most value is being generated, and what business goals are being met, can be used as the metrics for determining the business value granularity. For instance, if the service can produce the gross profit against a sales representative for each sale, the grain is coarser and offers more value than if it matches sales representative against their sales (La Rosa et al., 2011).

Appendix H – Description of the functions that make up the ETL process

1. Extract

Extraction is the start of the ETL process and involves the reading and extracting of the data from its source locations. Data warehousing is a means to consolidate data from various data source systems from separate or siloed business departments. The extract process often requires the data to be interpreted and verified to ensure it is capable of meeting the business requirements and technical structure of the data warehouse. If the data is not capable of meeting the minimum standard, it may be partially or completely discarded.

2. Transform

The transformation stage takes the extracted data from its source form, into the form required for its destination database. Transformation of data is performed through the implementation of defined business rules, functions, lookup tables, or a combination of data sets. Not all data sources will require the same level of manipulation, and some data sets may require no transformation to meet the technical and business needs necessary to produce valuable data analytics.

Data transformation can take many forms, such as the selection of only certain columns from a table, splitting columns, the translation of code values, encoding free form values, using calculations or aggregating data to make data sources meaningful to the business stakeholders, sorting and filtering data, and combining data from numerous sources. Once transformed the data will require validation, the complexity of the transformed data will determine the difficulty of this process, however, should it fail to meet requirements, the data can be fully or partially rejected.

3. Load

Load is the final process and involves the loading and writing of the transformed data into the destination data warehouse. Consideration must be made to how the data is loaded, it may be a full refresh that overwrites exiting data, or updated data, which means it will be added to the existing data. To ensure the quality, accessibility and stability of the data in the warehouse, this stage must be performed within the constraints defined against the database schema to prevent data loss, data duplication that could lead to result errors (IBM Docs, 2021).

Appendix I – Description of Batch Processing

The choice of the deployed pipeline is influenced by other concerns surrounding the availability of historical and near real-time data. Batch processing is one solution for processing high-volume and repetitive or frequent data loads with limited user involvement. The benefit of batch processing is the ability to allow data to be merged seamlessly, and integrated, into the data warehouse from the applications and data stores utilised across an organisation. The data is collected and stored before processing, which is scheduled to occur during a batch window designed to prioritise those data processing jobs that are considered more essential to the business operation. Scheduling batch processing events in a timely manner improves the efficiency of data processing when computing and other resources demands are low. The automation of batch processing results in improved data quality as a result of the lack of user involvement, reducing the risk of errors. Batch processing allows large volumes of data across several data load jobs to be processed simultaneously, speeding up delivery time for faster data analytics, however, where real-time data analytics are required, data streaming should be considered (Batch Processing - A Beginner’s Guide | Talend, n.d.).

Appendix J – Description of the classes of data anomalies

1. Syntactical Anomalies

- Lexical

These are discrepancies that occur between the defined format and the structure of the data items. This will occur if the number of records, or tuples in a table, are significantly lower or higher than anticipated. For example, if a table was expected to have six columns to reflect the six attributes relating to a tuple, then the actual data structure does not conform to the defined format.

- Domain Format

These are errors where the attribute value does not conform to the expected format. For example, if the expected address format is '50 Main Street', but a concrete value of '50, Main St.' is ingested, it will represent a violation of the expected format, albeit the value is functionally correct.

- Irregularities

These are non-uniform values, they can include units and abbreviations such as currencies or country codes. For example, a global organisation will use different currencies for their payroll. If the correct currency is not explicitly stated, it could have profound consequences on the results if it is assumed to be uniform across the business.

2. Semantic Anomalies

- Integrity Constraint Violations

These are violations of one or more tuples that do not meet one or more of the integrity constraints used to describe the understanding of the data. Integrity constraints restrict the set of valid instances of data, for example, a fact relating to age cannot be less than 0, or birth date cannot be greater than today's date.

- Contradictions

These are violations either within a tuple or between sets of tuples that breach a dependency. An example would be *age* and *birth date*, if the birth date is '1st January 1970' then the age cannot be '20' given the current year '2021'. This would represent a contradiction in a functional dependency through the lack of an integrity constraint or a duplicate containing inexact values.

- Duplicates

These are two or more tuples representing the same entity in the database. The duplicates do not need to be identical to represent duplication, inexact duplicates are instances of contradictions between the tuples.

- Invalid Tuples

These are data records that do not violate any rule or constraint and therefore do not fall into any anomaly category as described. This makes them difficult to detect and correct despite the fact they do not represent valid entities. Entries defined as invalid will result from an inability to describe the entity within the formal model, through the use of integrity constraints.

3. Coverage Anomalies

- Missing Values

Data collection is an imperfect process, often data is not collected through omission or error. An organisation may have historically seen no need to collect the data, or there could a missing constraint that has allowed `null` values to be input against an attribute. Missing values that should exist to meet business requirements are regarded as anomalies.

- Missing Tuples

These are records that are missing in their entirety from the data set.

- Data Accounting

This is a classification relating to the amount of data accounting for a constraint violation. This can range from a single value, values from a tuple, values from one or more columns, or tuples and sets of tuples from different tables connected through their relationships (Müller & Freytag, 2003).

Appendix K – Description of the criteria for data quality

1. Accuracy

The accuracy of the data can be determined by the number of correct values in the data set, in addition to the overall number of values that makes up that data set. Accuracy is an aggregated value containing integrity, consistency, and density. Meeting this criterion means that the data set will not contain any defined anomaly except for duplicates.

2. Integrity

Data integrity is achieved when a data set contains a representation of all expected entities, the data will not contain any missing or invalid tuples, and there will be no violations of the integrity constraints. Integrity is an aggregated value of the quality criteria of completeness and validity.

3. Completeness

Completeness can be considered a data integrity issue, which in isolation is not a primary concern of data cleansing. Completeness is achieved through the correcting of tuples that contain anomalies and not through their deletion.

4. Validity

Validity occurs when invalid tuples have been identified and removed from the data set. Verifying the correctness of a value can be made impossible given the high cost or inability for repeating measures for verification. Validity can often be approximated through the use of integrity constraints on the data set, these constraints represent an understanding of the mini world, meaning any violation is regarded as invalid. Constraint violations may arise where integrity has not been enforced, either due to the demands on users, performance issues, or a lack of future-proofing.

5. Consistency

Consistency concerns itself with ensuring that syntactically uniformed data is free of contradictions. Consistency is an aggregated value containing schema conformance and uniformity.

6. Schema Conformance

Schema conformance is the number of tuples that conform to the syntactical structure that has been defined by the schema. If a system does not enforce the syntactical structure, then tuples in a data set will not conform to the domain format, this is common in relational databases where adherence to domain format is reliant on the user.

7. Uniformity

Any irregularities in the data set will lead to data that is not uniform. This arises through the improper use of values or the improper semantic use of values.

8. Density

Density is the number of missing values within the tuples of the data set, and the number of values that ought to be known because they exist concerning the entities described by the tuples. The inclusion of

`null` values can be used where values are non-existent but must be represented. `null` has the same value as *not known* in these instances, ensuring there is no downgrade in quality. Any attempt to estimate these non-existent values would constitute a downgrade in quality.

9. Uniqueness

Duplicates are the concern of uniqueness and the total tuples that are representative of a mini-world entity. Duplicates do not exist in a data set that meets the criteria of uniqueness (Müller & Freytag, 2003).

Appendix L – Description of the methods deployed for data cleansing

1. Parsing

Parsing is a method used to detect syntactical errors in a data set. For instance, a parser for grammar would be a program utilised to determine if a given string is an element of language that can be defined by grammar. The strings can be either individual attribute values or complete tuples of the relational instance. An example of parsing would be the correction of spelling errors or the inclusion of postal or ZIP codes. Microsoft's SQL Server's Data Quality Services (DQS) would be a solution for supporting this process. Data managed by a database management system (DBMS) is unlikely to contain lexical or domain errors, which would be anticipated in a flat file, however, both DBMS and flat files would be expected to contain formatting errors for individual attributes. Discrepancy detectors can reveal these types of anomalies through the use of pattern learning techniques that use a sample value set to deduce the domain formats. Enforcement mechanisms deployed against the schema are a useful determinate for the number of syntactical errors anticipated from a data set.

2. Statistical Correction

Statistical correction methods are used for auditing, correcting anomalies, and eliminating complex errors detected in the parsed data components. This process is achieved using sophisticated data or clustering algorithms, values of mean, standard deviation, range, and secondary data sources that go beyond the simple checking of integrity constraints. The aim is to uncover relationships between attributes that are difficult to detect as they often will not conform to the expected characteristics of a data set. Analysis of the data set may uncover unexpected values that would be identified as invalid tuples that can be studied in greater detail. If it is determined that values are unknown, a correction might be impossible, resulting in deletion as the only viable solution. Consideration might be made to correction through a

process of applying a statistical value. Missing values can also be treated this way, imputing one or more values into the tuples that are considered plausible. The domain expert would have to determine the best solution based on the data set and the reasonable outcome necessary to meet the business case.

3. Standardising

Standardisation is performed during the mapping of data from its current format, to the format of the data destination schema. Standardisation follows a custom set of identified business rules and affects the schema of the tuples and domains of their values. In data warehousing the data is mapped from various sources, to make this valuable for meaningful analytics, it must be formatted to the common schema design. Any data that does not conform to the schema must be standardised to ensure consistency in the data warehouse. Formatting transformations that conform to standardisation, are performed on the instance level to remove irregularities in the data set. Transformations can include the conversion of values or data types or the assigning of numeric values to sit in a fixed minimum and maximum interval. Examples of standardisation include replacing nicknames or using preferred street names to enforce the defined data consistency.

4. Duplication Elimination

Matching data to eliminate duplications is performed after the data set has been parsed, corrected, and standardised. The domain expert must determine whether potential duplicated tuples represent a matching entity. The process requires each tuple to be compared with all other tuples in the data set to reduce the number of values requiring comparison, the tuples are sorted by a key generated from an attribute. The intention is to narrow the window of tuples for comparison by bringing them closer together, these can be matched both within and across the data set based on predefined business rules created from the specific domain knowledge. This can be performed multiple times across the data set and combined to improve the accuracy of the results. Any formatting errors that remain in the data set may lead to tuples not being close together, which will impact the quality of the data. An example of duplication elimination is through the identification of names or addresses that are similar, this might be genuine differences or errors, further study would be needed to decide.

5. Data Consolidation

Data consolidation relies on the enforcement of integrity constraints to identify relationships between matched records, for the possible merging of tuples into a single representation. Any violation of integrity

constraints should be eliminated for the satisfactory performance of transactional modification of tuples. Integrity constraints should be maintained and checked for rejected transactions to guarantee the resulting data. This is a supportive role due to the process being in the domain of the user, being under their control at all times. If a constraint violation is detected it must be repaired to support overall data quality. These methods should be deployed collectively where necessary across the data cleansing process to enhance the overall quality of the data, and to create a single source of truth that can be relied upon by business stakeholders. The extent of the methods necessary will be highly dependent on the use case and the business requirements, the methods deployed should be considered in the context of the problems and challenges faced by the team, particularly the management of multiple and alternative values, and the appropriateness of the supporting framework (kexugit, n.d.).

Appendix M – Outline of the organisation controls required for a SOC 2 Type-II audit

1. Vendor Risk Assessment

The company's *Vendor Risk Assessment Policy* requires an investigation of all vendors before the integration of the service or product into the wider IT infrastructure. This process includes the creation of an audit log that includes:

- Any security accreditation held by the vendor
- The geolocation of the vendor, considering support time, native languages, and GDPR for the storage of data
- Identify any regulatory risks the vendor may create for the company
- Identify any reputation risk that might occur by deploying the vendor services
- Review data protection policies and the values held by the vendor
- The scope of the integrations with other applications
- Whether the vendor meets the primary purpose of deployment
- Whether the vendor has a public status page
- The level and professionalism of the customer service offered
- Testimonials of existing companies
- The pricing structure to mitigate any lock-in periods, exit penalties or unused credits
- Any trial period offered and to what extent it is available

2. SaaS Risk Register

Once the Vendor Risk Assessment is complete and a vendor is chosen, risks can be identified and logged in the *SaaS Risk Register*. The risk register records:

- The company account holders
- A description of the application and its scope within the organisation
- A description of the risk or risks identified
- An assessment of the probability of the risk occurring
- An assessment of the impact of the risk should it occur
- A detailed description of any mitigation tools or controls that can minimise the chance of the risk occurring
- Develop a response plan that meets the companies incident response program
- Copies of any documents, including but not limited to security audit reports, privacy compliance statements, and white papers

Appendix N – Case study (Data Anonymity)

Experimentation into data anonymity using U.S. Census summary data discovered that a small number of characteristics when combined across a population set, could uniquely or nearly uniquely identify individuals. Specific attribute data such as name or address was excluded from the study, instead of utilising quasi-identifier data to form its conclusions. The study found that out of a population of 248 million, 87% or 216 million had characteristics that had the potential to make them uniquely identifiable based solely on their *5-digit ZIP, gender,* and *date of birth*.

The potential to uniquely identify an individual based on this quasi-identifier data alone is sufficient to render it not compatible with the intended purpose. The study showed how it is possible to link the Census data with publicly available health records based on just these three data characteristics. An example was provided that proved the point. Cambridge, Massachusetts Voter data was cross-referenced with Group Insurance Commission (GIC) data for the state of Massachusetts. According to the data only six people matched the tested date of birth, of which only three were men, of them, only one lived in the tested zip code, this was for then State Governor William Weld. Governor Weld's voter data was used to identify his anonymised medical data through these shared attributes (Sweeney, 2002).

Appendix O – A description of generalisation and suppression in the k-Anonymity model

1. Generalisation

The practice of substituting a specific data value for a more generally applied one is known as generalisation. Essentially, a value is replaced in the data set by a less specific value that stays true to the

original, decreasing the number of distinct tuples. This can be seen with age value, a data subjects age of '35' can be grouped into a decade range, '30-39' for example, alternatively, employment type could be generalised from specific business segments to just 'government' and 'non-government'. We can see in these two examples that the specificity has been reduced, therefore removing the identifying information. This can also be achieved with ZIP codes, for example removing the rightmost digit in the data set '90210', '90211', and '90213', and generalising to '9021*' references a larger geographical area. Generalisation can be performed hierarchically through incrementations of one. In the above examples, age can be generalised into hierarchies based on five-year increments. Likewise, the same hierarchal structure can be deployed for ZIP codes. A business decision must be made on the level of hierarchal generalisation to ensure the data is sufficiently anonymised, whilst offering a level of granulation capable of serving the intended requirements of the business. The maximum hierarchical generalisation level results in the suppression of the data, the attribute is rendered as a suppressed value at this.

2. Suppression

Suppression is the process of removing an attributes value entirely from the dataset. Suppression is applied to data points that are either irrelevant or mostly irrelevant, this will be determined by the business case and the output requirements. If, for example, age is being collected to determine the likelihood of a product being purchased based on age groups, suppressing the age data would result in a data set that does not answer the business case. Suppression is achieved through a process of reaching maximum generalisation. Suppression should be considered in combination with generalisation to realise the advantages of the technique (Sweeney, 2002).

Appendix P – The European Union Agency for Cybersecurity best practices scenarios when considering pseudonymisation to protect the data subject's PII

Scenario 1: Internal Use

The data controller is responsible for data pseudonymisation, performing the selection and assignment of pseudonyms to identifiers. Data subjects may never learn of their pseudonym as the secret is retained by the organisation. In this scenario, the purpose is to enhance the security of the PII stored by the organisation.

Scenario 2: Processor Involvement

This scenario identifies cases where a cloud hosting provider, for example, acts as the data processor, collecting the data from the data subject then forwarding it to the data controller who performs the pseudonymisation. The data controller is still responsible for implementing the pseudonymisation before any subsequent processing of the data.

Scenario 3: Forwarding pseudonymised data

As with the previous two scenarios, the data controller performs the pseudonymisation after receiving the PII, the processor is not yet involved. The processor receives the data from the controller in its pseudonymised form. This process might be used for data analysis or persistent data storage but means that the data processor is not able to access the identifiers of data subjects thus providing the intended protection goal. This scenario might also be deployed where data is sent from one data controller to another where the organisational structure identifies multiple legal entities as controllers.

Scenario 4: Pseudonymised performed by Processor

If there is insufficient technical expertise in the organisation, or the organisation has deemed it appropriate to only store pseudonymised data in the interest of enhanced security and risk mitigation, it might be beneficial to assign the process to a data processor before it is received by the data controller. This scenario would still allow for the controller to re-identify data subjects through the data processor, it would also require a vendor risk assessment to be performed to ensure the controller is satisfied with the security integrity of the data processor. This could involve the use of a sequence of pseudonymisation processors acting as a chain to perform the task.

Scenario 5: Third-Party Pseudonymisation

This scenario is similar to scenario 4 in that the pseudonymisation is performed by a trusted third-party entity before it is sent to the controller. Contrary to scenario 4, the third party is not a processor under the control of the data controller, meaning the controller cannot access the data subjects' identifiers. This enhances the security and data protection at the controller level and supports the principle of data minimisation. An example of this scenario might be organisations where controllership is shared across legal entities, one controller would perform the pseudonymisation, leaving the other to conduct any further processing against the pseudonymised data.

Scenario 6: Data Subject Pseudonymisation

This is a very different approach to data pseudonymisation. In this scenario, the data subject plays a role in the process by retaining a self-generated pseudonym. This could be performed through a system that uses the public key of a pair in blockchain systems, which would prevent the controller from learning the data subjects' identifiers. The responsibility of the implemented structure would still rest with the data controller; however, it would support efforts toward data minimisation (Bourka et al., 2019).

Appendix Q – Results of data anonymisation utilising ARX

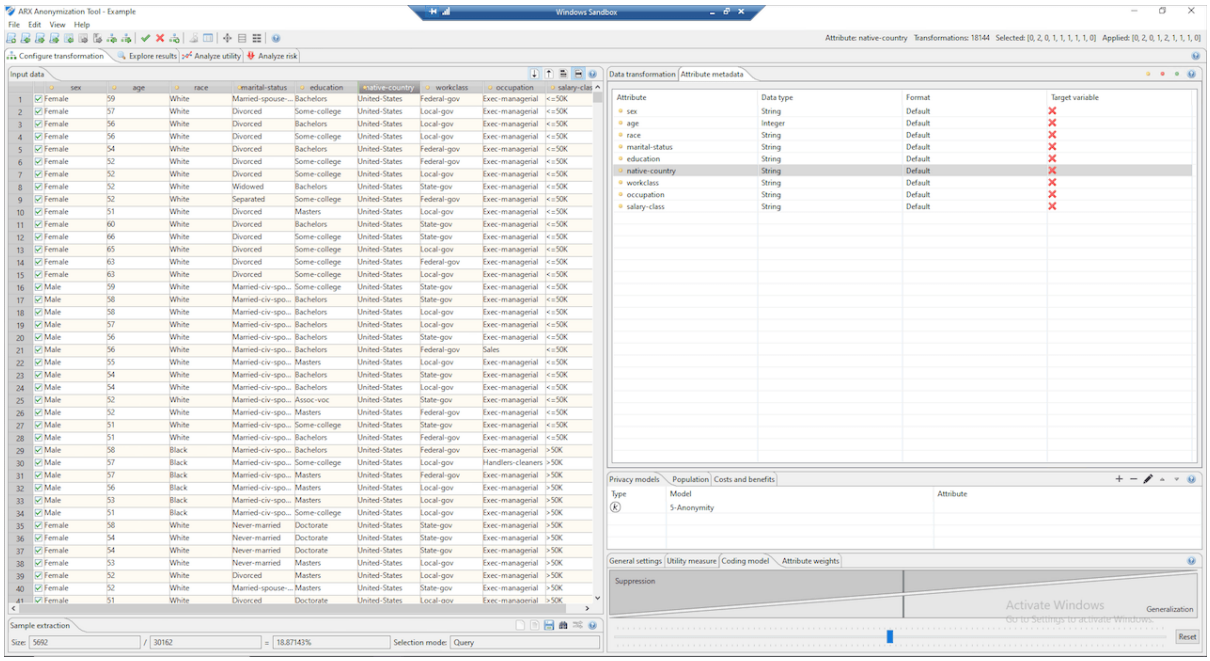


Figure 16 *Anonymised data set results produced using ARX*

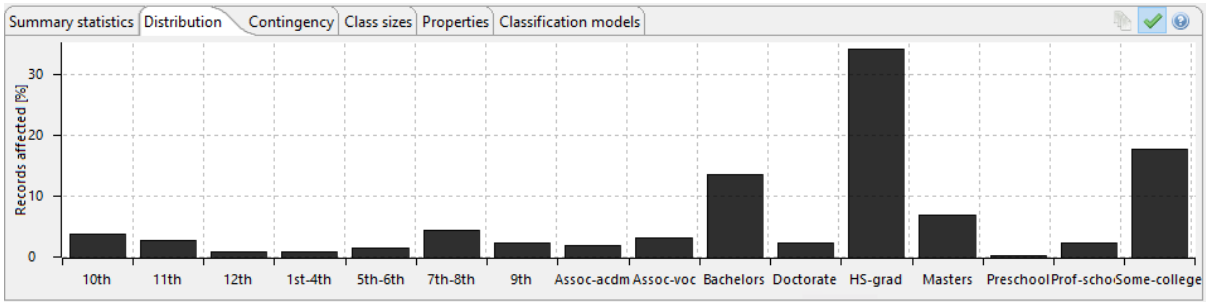


Figure 17 *Distribution of affected records relating to the achieved education level of data subjects*

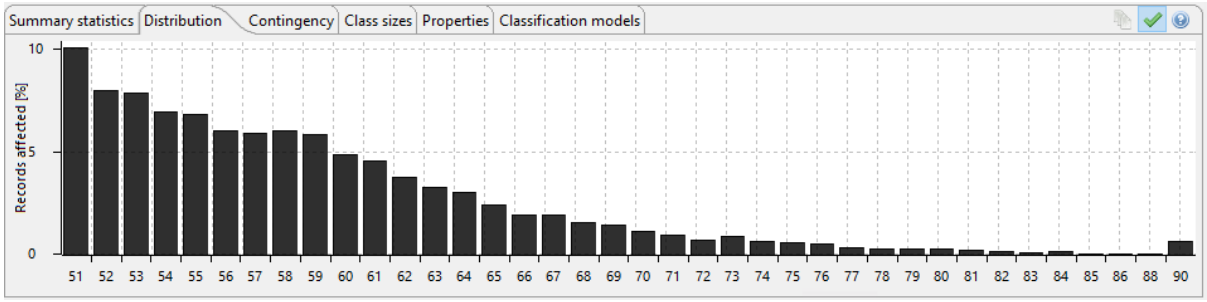


Figure 18 *Distribution of affected anonymised records relating to data subject's age*

Figure 19 *Distribution of affected anonymised records relating to data subject's marital status*

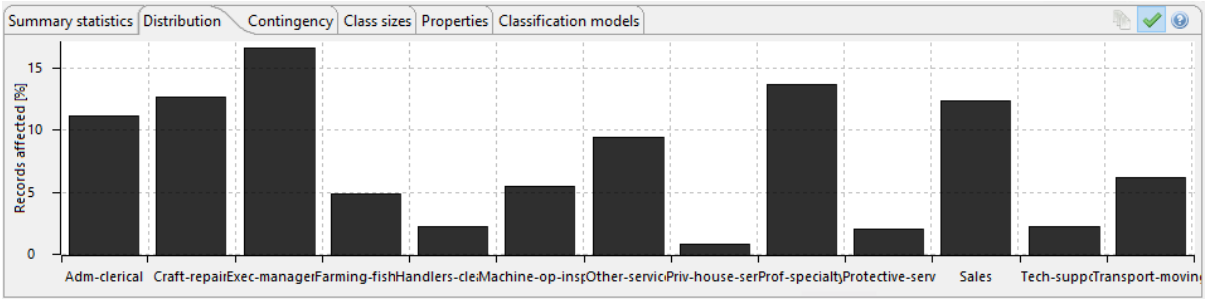


Figure 20 *Distribution of affected anonymised records relating to data subject's job category*

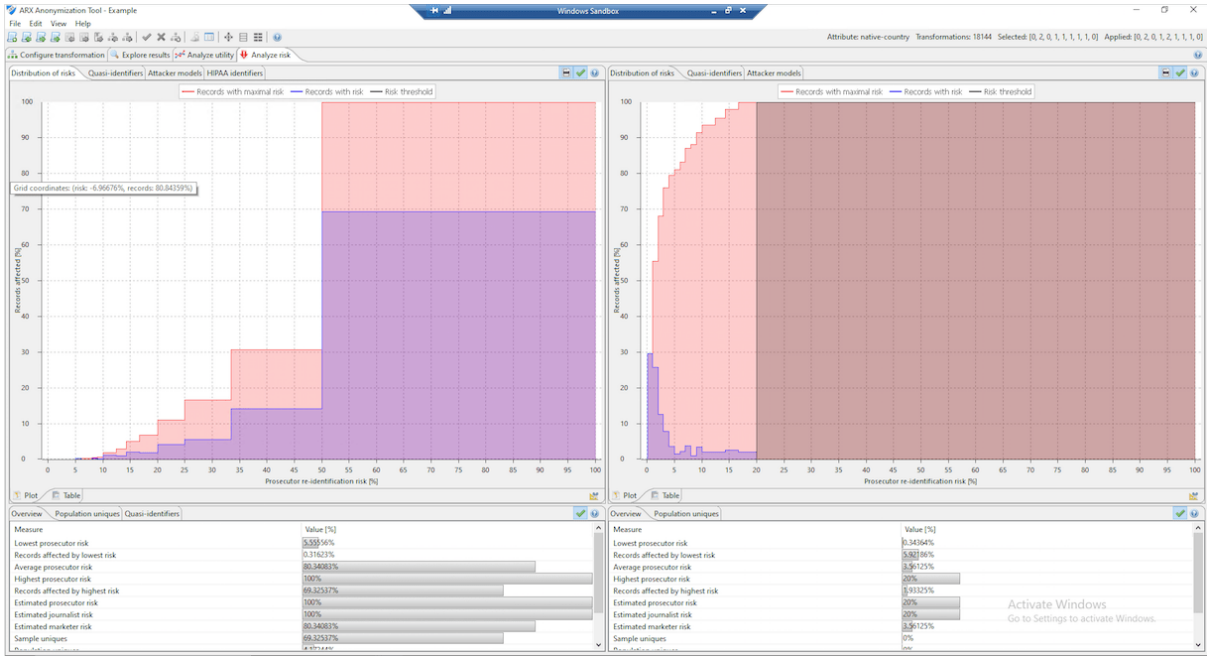


Figure 21 *Anonymised data set results produced using ARX*

Appendix R – Analysis of data warehouse architecture solutions

1. Independent Data Marts (IDM)

Data marts are often chosen to meet localized needs within organisational units or departments. However, the problem with this siloed approach is they become incapable of producing a centralised and single source of truth across a business or organisation. Further drawbacks often found with data marts is a lack of consistency and conformity, utilising different dimension tables making it technically difficult, or operationally challenging to analyse the data across the various data marts.

2. Data Mart Bus Architecture (DBA)

One solution to the problems encountered with an IDM architecture is a DBA with dimensional data marts that link across the existing data marts. Reviewing business requirements allows for a data mart to be created for a specific business process, deploying conformed dimensions and measures that can be used to link with additional marts when created. The net result is a series of marts that are integrated and organised into a star schema, to produce an enterprise view of the data.

3. Enterprise Data Warehouse Architecture (EDW)

The development of an EDW requires an extensive analysis of the business requirements to ensure the scalability of the architecture. The development of the architecture is iterative and prioritised on the needs of the business stakeholders. EDW typically maintains data in the 3rd normal form as defined by E.F.

Codd. EDW should be designed and integrated to eliminate the inconsistencies that evolve across application data sources and data stores that exist across organisations. EDW as a solution allows for dependent data marts that can perform specialised purposes within a department or functional area. This is generally considered an enterprise-wide data warehouse solution due to the data marts being planned and developed over time conforming to the established dimensions, in contrast to starting with a single data mart and growing from that point.

4. Federated Architecture (FED)

A federated architecture retains the existing data structure with no attempt to integrate the existing environment into a single solution. This is often preferred due to challenges faced as a result of an acquisition or merger, often the structure is too complex making integration impractical. The data is integrated through the application of shared keys, global metadata, distributed queries and enterprise information integration (EII) amongst others. This is considered to be a realistic solution to meet the demands of this type of complex organisational structure (Rifaie et al., 2008).

Appendix S – Comparison analysis between OLTP relation database and OLAP dimensional data warehouse (*Data Warehouse Tutorial For Beginners | Data Warehouse Concepts | Data Warehousing | Edureka - YouTube*, n.d.).

OLTP Relational Database	OLAP Dimensional Warehouse
Designed for Online Transaction Processing	Designed for Online Analytical Processing
Schema definition: Data Definition Language (DDL), Data Manipulation Language (DML), Data Control Language (DCL)	Schema definition: Data Mining Query Language (DMQL), Data Control Language (DCL)
Purpose: Running the business operations	Purpose: Analysing the business operation
Based on the Entity-Relationship Model	Based on Dimensional Star, Snowflake or Fact Constellation Schema
Primitive and highly detailed data that may not produce results that exactly meets end-user requirements	Summarised and consolidated data providing processed and accurate data that meets end-user requirements
Write data into the database	Read data from the database
Database size range from 100MB to 1GB	DWH size range from 100GB to 1TB

The database is fast and provides high performance	Not as fast but is highly flexible
No. of records accessed is in tens	No. of records accessed is in thousands
Example use case: • All customer bank transactions for a particular account • Delivery service to record and track every parcel • Airport flight reservation server to record passenger transactions	Example use case: • All customer bank transactions at particular cash machines covering a particular period • Discover the number of adverts sold by a particular sales team or sales representative to support performance management

Table 5 *OLTP & OLAP Comparison Analysis*

Appendix T – Definition of the properties inherent to data warehouses

1. Subject-orientated

Subject-orientated means the data is no longer stored by the application, but rather it is categorised by business subject as defined by the business requirements. This might include *orders*, *leads*, *customers*, and *marketing*, which when combined allows for metrics to be created and used for analysis.

2. Integrated

Integrated is the collection and standardisation of data from disparate sources being ingested and stored in a single location. The data must be flexible to the extent it allows new questions to be asked of it to improve decision making and problem-solving capabilities, address changing requirements, and identify trends in customer needs.

3. Time-variant

Time-variant is represented by each separate time data is stored in the data warehouse, collectively acting as a series of snapshots that can be used by the analyst to understand the company status or progress from a time-variant approach.

4. Non-volatile

Non-volatile refers to the typically stable nature of the data. To ensure stability, it is recommended that it is not updated or deleted to help prevent corruption (Rifaie et al., 2008).

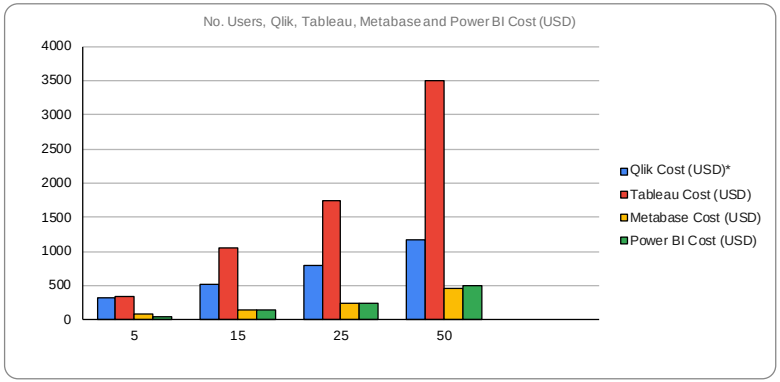
Appendix U – Comparative analysis of BI tools

(*Metabase vs Microsoft Power BI 2021 - Feature and Pricing Comparison on Capterra*, n.d.)

	Company	Qlik	Tableau	Metabase	PowerBI
Business	Subsidiary of:		Salesforce		Microsoft Azure
	HQ Location	PA, USA	Seattle, WA USA	USA	WA, USA
	Regions	Worldwide	TX, CA, MA, Canada, UK, EU, JP, AUS		Worldwide
	Employee Size	1001-5000 (LinkedIn)	1001-5000 (LinkedIn)	11-50 (LinkedIn)	
	Established	1993	2003	2014	1975
	Key Roles	Listed	Listed		
Security	SOC 2	SOC 2, SOC 3	SOC 2 Type II	SOC 2 anticipated Q4 2021	SOC 1, SOC 2, SOC 3
	ISO	ISO 27001			ISO/IEC 27001 and ISO/IEC 27018
Privacy	Storage Geolocation	Three regions: United States, Ireland or Australia	Live connections or extract		Most Azure services enable you to specify the region where
	GDPR Compliant	Anonymises data: GDPR no longer applies. Configuration	Not explicitly stated	Not explicitly stated	Yes
	CCPA Compliant	Not explicitly stated	Yes	Not explicitly stated	Yes
Competitive Analysis	Pros	Quick analytics and efficient reports, Analytics produced using	Live visual analytics, Vast amount of supported data sources	Cloud-based and on-premise platform for businesses of all	Connect with Microsoft and third party applications including
	Cons	Less user friendly, steeper learning curve, no desktop app	Expensive	Not as feature rich as competitors, no support	Requires desktop application
	Features	Ad hoc Reporting	Ad hoc Reporting	Dashboard	Ad hoc Reporting
	Deployment	Cloud, SaaS, Web-Based	Cloud, SaaS, Web-Based	Cloud, SaaS, Web-Based	Cloud, SaaS, Web-Based
	Support	Email/Help Desk	Email/Help Desk	Documentation	Email/Help Desk
Credibility	Customers	Lloyds, NHS, PayPal, Ford, BP, Deloitte	Red Hat, Henkel, Hello Fresh, Chipotle, Lenovo, Verison,	Usabilla, treebo, N26, gojek, AngelList Venture	Adobe, PharmID, WorldSmart, Gartner, Forrester
	Recognition	Top Performer Integration Software, Data Analysis Software,	Top Performer Data Analysis Software & Data Visualization		Data Analysis Software, Data Visualization Software,
Integrations		Amazon Redshift, Postgres, Snowflake, Google BigQuery,	Salesforce, PostgreSQL, Amazon Redshift, Snowflake and	Snowflake and more	Postgres, Amazon Redshift, Snowflake and more
Price	Billing Period		Annual	Monthly	Monthly
	$USD Full License	70	70	5 - 85	10
	Edit existing dashboard	Individual Analyser - 40 (No time limit, using a lot & often)	42		
	Viewer Only	Capacity Analyser -100 (1000 minutes per month can be	15		
	Period		User/Monthly	First 5 users then $5 per additional user	User/Monthly
	Trial Period		14-day	14-day with credit card	Free version

	No. Users	Qlik Cost (USD)*	Tableau Cost (USD)	Metabase Cost (USD)	Power BI Cost (USD)
Per User Cost - Full Licence	5	320	350	85	50
	15	520	1050	135	150
	25	800	1750	235	250
	50	1180	3500	460	500

*Unlimited Capacity Users



	No. Users	Qlik Cost (USD)*	Tableau Cost (USD)	Metabase Cost (USD)	Power BI Cost (USD)*
Per User Cost - Tableau Mixed Licence (5% Full / 15% Edit / 80% View)	50	320	1090	310	500
	100	520	2180	560	1000
	150	800	3270	810	1500
	250	1180	5450	1310	2500

*Unlimited Capacity Users

*Enterprise plan may reduce cost

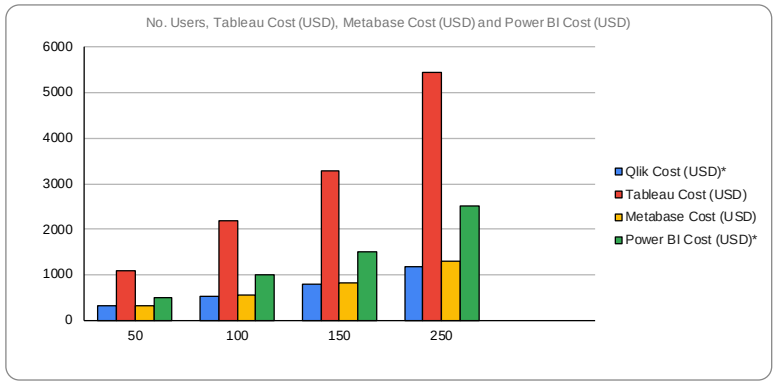


Table 6 *Comparative analysis of BI tools*

Appendix V – A description of a data lake

Data lakes are deployed by organisations to collect and store data. Data lakes are the largest data storage locations and have the potential to store all the data an organisation collects, resulting in an unorganised storage environment with no filtering. An organisation can make a decision later as to what to do with all this information. The data and information contained can include:

- Structured data
- Chat logs
- Emails
- Images and videos
- Online reviews
- Forum discussion about the product or company

There are two reasons an organisation would choose a data lake. Firstly, to store large quantities of data cost-effectively, and secondly, to store information an organisation has determined as useful to retain, but do not yet know what it will be useful for, deciding it will be used at some point, but are unsure on the how, why, or when. A data lake is an ideal solution as it does not place constraints or apply structure to the data. Data lakes are primarily accessed by specialists, this is due to the work involved when interpreting the data resulting from its unstructured nature. Generally, access will be by those within the organisation that has a high level of access and technical coding expertise (The Career Force, 2019).

Data lakes offer further benefits when it comes to changes as they can be performed quickly. In some instances, specialists can use a data lake for quick analysis precisely because all the information is readily available and in a flexible state to be transformed as required. However, data lakes are not recommended for broad use across an organisation as they lack the structure needed for a large implementation. Data lakes can also store data that is contained in a data warehouse; however, it is not cleansed, structured or organised in the way it is in a data warehouse (*What Is a Data Lake? - YouTube*, n.d.).

Appendix W – A description of machine learning

The importance and rapid development of machine learning to perform specific tasks is not underestimated, nor is its contribution to the technologies now associated with artificial intelligence (AI), however, these are beyond the scope of this report. Machine learning is concerned with the modelling of data and what can be learnt from the recording of measurements and objects across business sectors.

Machine learning is achieved through the process of optimising an objective function with specific examples, which can be formulated to result in desired answers to questions, or novel solutions to problems. To accomplish these goals, computers require explicit programming to be utilised for advanced object recognition, data mining, and business intelligence systems. Machine learning is the engine deployed for data analytics, big data, and data science. Advances in machine learning, in particular deep learning methods, are realised in natural language and image processing or more generally, data analytics.

As this report outlines, data analytics requires extreme care with regards to security and privacy in the age of encroaching regulation resulting from the increased ability to collect and store personal data. Problems in this area can arise from machine learning methods, namely the ability of machine learning to solve problems that can lead to new concerns, specifically the aggregation of information that has the potential to compromise the privacy of individuals. This report highlights the anonymisation methods used that remove personal identifiers from data sets, allowing organisations and businesses to meet their regulatory obligations. Machine learning is capable of analysing this anonymised data and linking it back to the individuals, a process that would not ordinarily be possible, or at the least be difficult to the extent it would not be attempted. Regardless of any privacy concerns surrounding the right of individuals, machine learning is continuing to advance. Examples are identified in this report with the testing of Qlik, a business intelligence tool that deploys natural language machine learning intelligence to rapidly-produce insights from data sets. The benefits of deploying machine learning in a data warehouse pipeline must, like any other technology, be considered alongside the responsibilities placed on an organisation (Trappenberg, 2019).