NOVA IMS Information Management School

MDSAA

Master Degree Program in
Data Science and Advanced Analytics

USING DATA VAULT 2.0 IN THE BANKING INDUSTRY

Diogo Filipe Farinha Hipólito

Dissertation

presented as partial requirement for obtaining the Master Degree Program in
Data Science and Advanced Analytics

NOVA Information Management School
Instituto Superior de Estatística e Gestão de Informação

Universidade Nova de Lisboa

NOVA Information Management School
Instituto Superior de Estatística e Gestão de Informação

Universidade Nova de Lisboa

USING DATA VAULT 2.0 IN THE BANKING INDUSTRY

by

Diogo Filipe Farinha Hipólito

Master Thesis presented as partial requirement for obtaining the Master's degree in Data Science and Advanced Analytics, with a specialization in Data Science.

Supervisor: José Henrique Pereira São Mamede

Co-Supervisors:

July 2023

STATEMENT OF INTEGRITY

I hereby declare having conducted this academic work with integrity. I confirm that I have not used plagiarism or any form of undue use of information or falsification of results along the process leading to its elaboration. I further declare that I have fully acknowledge the Rules of Conduct and Code of Honor from the NOVA Information Management School.

Diogo Hipólito

Lisbon, July 8, 2023

ACKNOWLEDGEMENTS

I would like to express gratitude to my supervisor, José Henrique Pereira São Mamede, for guiding me throughout this project, never making me feel lost, and always encouraging me when I thought I couldn’t do it.

To my loving girlfriend for comforting me in the often stressful times I faced while writing this Dissertation, and for making me feel like everything I set my mind to is within reach.

To my family for providing me with a life that allowed me to have this opportunity. The constant effort and sacrifice by my Mother, raising me and my three siblings through the hardest of times. I will forever be grateful.

ABSTRACT

Organizations increasingly recognize data as a critical resource, demanding effective storage and processing methods to handle exponentially growing volumes of data. This is particularly pertinent in the banking industry, characterized by rapidly changing business requirements and heavy regulatory measures. This thesis investigates the application of the Data Vault 2.0 Enterprise Data Warehouse (EDW) methodology within the banking sector, an alternative to traditional Kimball and Inmon data warehouses, characterized by its flexibility, scalability, and its ability to adapt to new business requirements. This study particularly focuses on the potential of integrating data sourced from a data lake, a centralized repository capable of storing massive volumes of structurally diverse data, to amplify the potential of this solution. This research, conducted in collaboration with a leading Portuguese bank servicing three million customers, involved the creation of a Data Vault model using the bank's customer and current account data. The model's ability to accurately reflect the business logic and adapt to real-world requirements was demonstrated, and subsequently evaluated by experienced professionals within the organization. The results reveal significant potential for the implementation of a Data Vault 2.0 EDW in conjunction with a data lake in the banking industry, as a scalable, efficient system that can realistically be adopted and excel in an enterprise setting.

KEYWORDS

Data Vault 2.0; Data Lake; Data Warehouse; Data management; Data architecture

INDEX

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS AND ACRONYMS

DV	Data Vault
DL	Data Lake
DW	Data Warehouse
EDW	Enterprise Data Warehouse
DSR	Design Science Research
SLR	Systematic Literature Review
GL	Grey Literature
PK	Primary Key
FK	Foreign Key
ETL	Extract, Transform, Load
HK	Hash Key
ESG	Environmental, Social, and Governance

1. INTRODUCTION

As we delve deeper into the digital age, data has emerged as an invaluable resource, above all others. Data is everywhere, and its growth is exponential, not only in volume but also in variety and velocity (Sivarajah et al., 2017). Organizations are aware of this evolving landscape and are channelling significant resources to capitalize on the power of data, integrating its use across all their operational aspects. Though this shift brings forth exciting prospects, it does come with its own set of challenges. (Ribeiro et al., 2015)

The sheer volume of data companies store and process on a daily basis is astoundingly vast, and it continues to expand at an unprecedented rate (Sivarajah et al., 2017). To meet these demands, companies must institute robust, scalable systems (Hu et al., 2014). These systems must be designed not only to meet existing data demands but also to preemptively accommodate for projected future data intake. A lack of strategic foresight can lead an organization towards frequent overhauls of its data infrastructure, a scenario that is decidedly inefficient both in terms of time and resources (Linstedt and Olschimke, 2015).

The concept of a Data Warehouse, as articulated by Inmon in 1992 (W. H. Inmon, 1992), was proposed as a solution enabling organizations to manage large quantities of data efficiently. It centralizes data from various operational systems within an organization into a single data repository, emphasizing the integration of historical and time-variant data. This setup facilitates the maintenance of a comprehensive data history, allowing for in-depth trend analysis, the interpretation of historical patterns, and the facilitation of data-driven decision-making. (W. H. Inmon, 1992)

However, prevalent data warehouse designs, primarily based on the widely-used Kimball (2003) and Inmon (1992) methodologies, often encounter difficulties in adapting to the rapidly evolving data landscape. These designs are predominantly tailored for structured data, and have rigid structures, making them less flexible for new business requirements. Consequently, organizations are compelled to invest significant resources for continuous maintenance and adaptation.

The Data Vault was originally presented by Linstedt (2002) as an innovative data warehousing modelling approach. However, the advent of Data Vault 2.0 transcends this original scope by encompassing not just the data modelling component, but broadening it considerably. Data Vault 2.0 represents a comprehensive business intelligence system, incorporating elements of modelling, methodology, architecture, and implementation of an Enterprise Data Warehouse (EDW). (Linstedt and Olschimke, 2015)

The Data Vault distinguishes itself from other methodologies by its ability to adapt to rapidly changing business requirements, while preserving historical data, and its enhanced scalability — applicable not only to the sheer volume of data but also to the complexity of data it can accommodate. It integrates data from diverse sources while maintaining stringent auditability standards. This is particularly significant for highly regulated industries. (Linstedt and Olschimke, 2015)

Additionally, the data lake has emerged as a promising solution for organizations to efficiently store high volumes of structured, semi-structured, and structured data, allowing the ingestion of data from various sources in their native format, allowing for greater adaptability and scalability (Singh et al., 2022).

1.1. MOTIVATION

This study aims to investigate the application of a Data Vault 2.0 Enterprise Data Warehouse (EDW) sourced by a data lake within the banking industry. The research was conducted in collaboration with a leading Portuguese bank, which serves a customer base of three million.

The bank represents a large-scale banking institution grappling with the challenge of managing an ever-expanding volume of data, and an increasingly diverse array of data sources. This has resulted in performance and scalability issues in its existing traditional data warehouse, leading to operational inefficiencies, escalating costs, and an inability to fully capitalize on the potential value of its data assets. These issues are not unique to this organization; they represent broader trends within the banking industry, which is undergoing rapid digitization and is frequently subject to regulatory changes. (Diener and Špaček, 2021)

Current methods to address these challenges within the organization involve scaling up existing data warehouse infrastructure or implementing new data stores for specific needs. These solutions often provide temporary relief and don't resolve inherent scalability and performance issues. Moreover, these approaches often lead to isolated data silos, large amounts of separated data, connected only by loose inter-connections, or completely disconnected. (Hai et al., 2016)

In this context, integrating a Data Vault 2.0 EDW with a data lake emerges as a promising solution. The Data Vault 2.0 EDW provides the requisite flexibility, scalability, and agility, potentially addressing many of the challenges that the bank faces with its current EDW; while the data lake is capable of storing not only structured data, but also semi-structured and unstructured data. Furthermore, it stores raw data directly from the sources, which is crucial for auditability purposes, a requisite in highly regulated industries such as banking.

1.2. RESEARCH QUESTION AND OBJECTIVES

This study aims to explore the use of the Data Vault 2.0 EDW methodology within the banking industry. It also aims to explore the potential value and advantages of using a Data Lake as a source for a Data Vault 2.0 EDW. With these goals in mind, the central research question guiding this Dissertation is: ”How can a Data Vault 2.0 EDW address the primary data management challenges of an organization within the banking industry, being served by a Data Lake with a proper architecture?”

The research objectives of this study are the following:

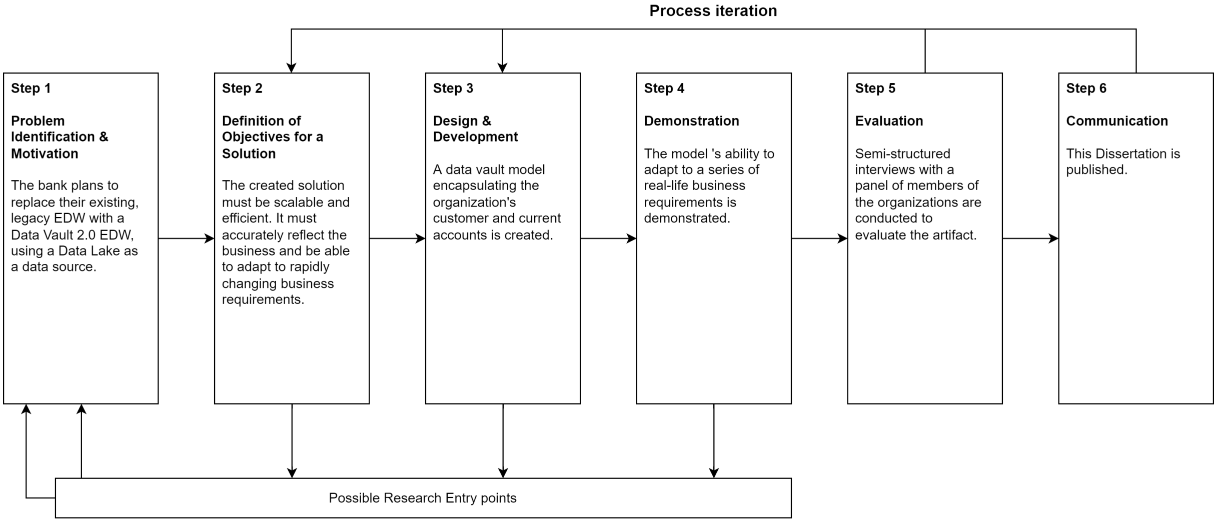
- To conceptualize and develop a data lake architecture that facilitates efficient integration with a Data Vault 2.0 EDW.
- To design a Data Vault 2.0 model that accurately reflects the business logic of a banking organization, while validating its robustness and flexibility in adapting to rapidly changing business requirements.
- To measure the efficacy and accuracy of a Data Vault 2.0 model in representing the business logic, and its fit within the organization.

1.3. DOCUMENT STRUCTURE

This document is divided into 7 sections, including the present section, where we present the motivation and research question driving this study. Section 2. delineates the specific methodology adopted in conducting this research. Section 3. provides a theoretical background that serves as the foundation for our research. Section 4. describes the development of the artifact that serves as the key output of this research. In Section 5. the artifact's ability to solve real-world problems is demonstrated. Section 6. describes the evaluation of the artifact by a set of experts. Finally, Section 7. outlines the conclusions drawn from the research and indicates the limitations of the research and potential avenues for future study.

2. METHODOLOGY

This research employs Design Science Research (DSR) methodology, a research paradigm developed by Hevner et al. (2004). This methodology endorses the concept that knowledge can be derived from the creation of innovative artefacts designed to address real-world problems. Given the practical implications of constructing a Data Vault model, the DSR methodology is particularly well-suited for our study. A visual representation of the multiple DSR steps, as applied in this study, is depicted in Figure 1.



The bank plans to replace their existing, legacy EDW with a Data Vault 2.0 EDW, using a Data Lake as a data source.

The created solution must be scalable and efficient. It must accurately reflect the business and be able to adapt to rapidly changing business requirements.

A data vault model encapsulating the organization's customer and current accounts is created.

The model 's ability to adapt to a series of real-life business requirements is demonstrated.

Semi-structured interviews with a panel of members of the organizations are conducted to evaluate the artifact.

This Dissertation is published.

Figure 1: Steps of DSR (adapted from (Peffers et al., 2007))

The DSR methodology was executed following the guidelines established by Peffers et al. (2007), which are comprised of the following key steps:

Initially, in the problem identification and motivation step, the research problems and the value of their solutions are identified and defined (Peffers et al., 2007). These are detailed in Section 1., which elaborates on the collaborating organization's challenges with its existing data infrastructure. Additionally, as part of this step, to gather an understanding on what's feasible and possible for a solution, a review of the existing literature relevant to the solution is conducted, as detailed in Section 3.

Once the problem has been identified, we define the objectives for a solution. These objectives are inferred rationally from the problem's specifications (Peffers et al., 2007), and are outlined in Section 1.2.

Following the definition of the research objectives, we progress to the Design and Development step commences, in which the creation of an artifact, is detailed. Before the artifact's creation can proceed, it's crucial to establish an understanding of the relevant theoretical knowledge that could inform the solution Peffers et al., 2007. In this research, this understanding is obtained via a Systematic Literature Review (SLR). The objective of the SLR is to grasp the state-of-the-art in the existing body of knowledge pertinent to the solution (Kitchenham, 2004). The SLR is presented in detail in Section 3.

In the Design and Development step, upon the recognition of the existing relevant body of literature, the artefact, the primary output of this research, and its creation process are outlined (Peffers et al., 2007). In the case of our research, the artefact is a Data Vault model representing the organization's data. Given the breadth of our collaboration with a well-established organization that boasts an operational history spanning over a century, fully modelling a Data Vault model that represents the entirety of the bank's operations would surpass the project's scope due to its extensive nature. Therefore, this study selectively focuses on a particular segment of the Data Vault model. The resultant artefact from our study is a data vault model that encapsulates the bank's customer-related data. This model represents the customers' interrelationships and relationships with current accounts. The design and development step is comprehensively described in Section 4.

In the Demonstration step, the artefact must exhibit its capability to address real-world problems (Peffers et al., 2007). Here, the model demonstrates its ability to efficiently adapt to rapidly evolving business requirements, a common occurrence in the banking industry. The demonstration step is thoroughly described in Section 5.

In the Evaluation phase, the artefact's capability to support a solution to the problem it attempts to solve is observed and measured (Peffers et al., 2007). Given that the model draws on the organization's data and mirrors its business operations, evaluating the model essentially entails assessing its capacity to solve problems within the organization's unique context. To comprehend the challenges the organization currently faces and to identify how the model might address them, we sought feedback from the individuals most familiar with these issues, the model's potential users. To this effect, we conducted semi-structured one-on-one interviews with various members of the organization occupying significant roles within the organization's data department. The results of the interviews allowed us to gain valuable insights on the model, particularly on its accuracy in reflecting the underlying business logic, the model's overall efficacy, and its importance in the context of the organization. Section 6. provides a detailed account of the evaluation step.

Finally, as part of the communication step, the results and possible effects of this research are shared in this Dissertation. This final step makes sure that the knowledge and practical lessons from this research are made available to both the wider academic world and those who may find it relevant in the industry.

3. THEORETICAL BACKGROUND

The second step of DSR encompasses the formulation of solution objectives, derived from an understanding of the problem and knowledge regarding what is feasible and possible, as outlined by Peffers et al. (2007). Thus, before delineating the objectives for a solution to the identified problem, it is critical to undertake a comprehensive review of the existing literature pertinent to the problem.

This section details a Systematic Literature Review (SLR), planned and conducted to explore, evaluate, and synthesize the state-of-the-art on the use of a data lake as a source for a Data Vault 2.0 EDW. This review has two primary goals: Firstly, it aims to gather insights into the existing knowledge base of this research topic, providing a substantial starting point for our research and contextualizing the research within the broader academic discourse. Secondly, it aims to identify potential gaps in the current research. Recognizing these gaps enable us to formulate the research questions that drive our study, highlighting areas that are either unknown or insufficiently explored. This process underscores the relevance of our research by demonstrating how it can add value to the existing body of literature. The structure of this section is as follows:

Section 3.1. introduces the methodology used in this SLR, describing the necessary steps to examine both formal and grey literature effectively. Section 3.2. focuses on the formal literature review, outlining the strategies used for data collection and its subsequent execution. Section 3.3. performs a similar task for grey literature. Section 3.4. delves into the main findings from both the formal and grey literature, analyzing, synthesizing, and identifying the most pertinent sources for this research. Section 3.5. concludes by summarizing the primary insights derived from the selected sources and presenting identified research gaps.

3.1. METHODOLOGY

This multivocal literature review is divided into two main parts. The first part consists of a formal literature review conducted following the guidelines for performing a systematic literature review proposed by Kitchenham (2004). The second part consists of a grey literature review conducted following the guidelines for including grey literature in a literature review by Garousi et al. (2019). The aim of the grey literature review was to supplement the findings gathered in the formal literature review and provide a more comprehensive overview of the research topics.

The methodology applied to the formal and grey literature reviews is organized into three phases: Planning, Conducting, and Reporting.

The Planning phase unfolds in four stages. Initially, the research questions were formulated. Subsequently, a strategy for information collection was devised. Next, the criteria for the inclusion and exclusion of papers were established. Lastly, a quality assessment stage was planned for the papers meeting the inclusion criteria, ensuring the selection of high-quality references for this review.

In the Conducting phase, the steps laid out in the Planning phase are executed and the results are documented. The results of the paper collection process, the application of the selection criteria, as well as the findings of the quality assessment phase, are accordingly presented.

During the Reporting phase, key insights derived from the literature review are synthesized and organized based on the respective research questions they address. This phase covers the findings derived from both the formal and grey literature sources.

3.2. FORMAL LITERATURE REVIEW

In our study, we conduct a systematic review of formal literature, wherein we thoroughly analyze and synthesize scientific papers to gain an understanding of the current state-of-the-art in data lake architectures and their integration with data warehouses.

The structure of this section is as follows: Section 3.2.1. details the planning phase of the SLR protocol that we employed. Following that, Section 3.2.2. elaborates on the execution of the protocol conceived during the planning stage.

3.2.1. Planning

Prior to conducting a literature review, it is imperative to establish a robust review protocol to facilitate a comprehensive, bias-free systematic literature review, as per Kitchenham's (2004) guidelines.

Each subsequent subsection delves into an essential component of the protocol, beginning with a justification of its necessity followed by an exploration of the actions undertaken and decisions made.

3.2.1.1. Research Questions

To conduct a systematic literature review, the review's research questions must first be laid out. These are the questions that the literature review attempts to answer. To aid with phrasing the research questions, as well as to help create a search string for searching the data sources, a set of aspects need to be considered. This process is known as "PICOC" (Wohlin et al., 2012). The aspects considered, their descriptions and the values given for this review are figured in Table 1.

Table 1: PICOC

Aspect name	Description	Value
Population	Which groups of people/programs/businesses are of interest for this review	The organization's source systems
Intervention	Which tools/technologies are under study?	Data Lake, Data Vault, Data Warehouse
Comparison	The comparison to which the intervention is compared	This aspect is excluded as it does not apply to the research
Outcomes	The outcomes of the experiment	A data lake architecture proposal more efficient than current alternatives
Context	The context of the study, the grander subject of the review.	Data engineering

As per Kitchenham's (Kitchenham, 2004) guidelines, research questions must be relevant and significant to both practitioners and researchers. They should aim to bring about changes in software engineering practices or reinforce the value of current practices and uncover any disparities between common beliefs and actualities. The literature review's research questions are presented in Table 2.

Table 2: Research questions of the literature review

Name	Description
RQ1	What architecture models are used in current data lake implementations?
RQ2	How are current data lake implementations integrated with data warehouses?
RQ3	How do current Data Lake implementations store and process batch and real-time data?

3.2.1.2. Search Process

The formal literature search process involved searching and gathering papers from various relevant bibliographic databases. To ensure a systematic and consistent search process across different databases, a common search string was defined and used to query all databases. For this search, journal articles, conference papers, and peer-reviewed books published from 2017 onward were considered, with exceptions made for works by notable authors in the field that are often cited in other researchers' papers. The databases, and search string used can be observed in Table 3.

Table 3: Search process details

Element	Research Details
Search string	”Data Lake” AND ”Architecture” AND (”Data Vault” OR ”Data warehouse”)
Databases	Scopus, Science@Direct, ISI Web of Science, ACM Digital Library

3.2.1.3. Inclusion and Exclusion criteria

In reviewing the relevant literature, each paper's title and abstract are carefully read. Based on predefined inclusion and exclusion criteria, papers are either incorporated into our study or dismissed. These criteria, formulated with an emphasis on consistent interpretation and precise classification, enable the selection of studies most pertinent to the research questions (Kitchenham, 2004). A list of these criteria is provided in Table 4.

Table 4: Inclusion and exclusion criteria

Inclusion criteria	Exclusion criteria
Discusses the design of a data lake architecture	The paper is not related to the research questions
Discusses the integration between a data lake and a data warehouse	The paper is unavailable in English or Portuguese
Discusses the integration between a data source and a data vault	Summary or mapping
Discuss the management of batch and streaming data in a data lake	Incomplete or unavailable paper
	Paper is not peer-reviewed
	Paper is not from conference proceeding, scientific journal, or peer-reviewed book
	Paper is published before 2017

3.2.1.4. Quality assessment

Following the first filtering of sources using the selection criteria, the sources are then fully read and go through a quality assessment phase. Quality assessment involves evaluating individual studies against predefined quality criteria questions, with three answer options: ”Yes” (1 point), ”No” (0 points), and ”Maybe” (0.5 points). Papers scoring over four points are considered high-quality, while those scoring between four and three are still usable. Papers scoring lower than three are unusable. The process aims to include only relevant, high-quality papers in the literature review. The quality assessment questions the studies from the formal literature review are evaluated against are listed in Table 5.

Table 5: Quality assessment questions

Criteria	**Description**	**Weight**
Methodology	Q1: Is the publishing organisation reputable?	1
	Q2: Are the limitations of the study clear?	1
Objectivity	Q3: Is the paper supported by a literature review?	1
	Q4: Are the results clearly stated?	1
	Q5: Are the results compared with previous studies?	1
Novelty	Q6: Do the results add to the literature?	1
	Q7: Does the study propose a data lake architecture?	1
	Quality Score threshold	**4**

3.2.2. Conducting

During the conducting phase, the steps outlined in the planning phase were carried out, resulting in the collection of 356 papers from four different databases. After removing duplicates, 313 papers were retained due to overlapping search results. The titles and abstracts of these papers were screened, and 42 met the inclusion criteria. The selected papers were then fully read and evaluated using a set of quality assessment questions, and out of 42 papers, 17 scored above the threshold for high quality, and 16 were still considered usable, totalling 33 papers. A visualization of the paper selection process is depicted in Figure 2.

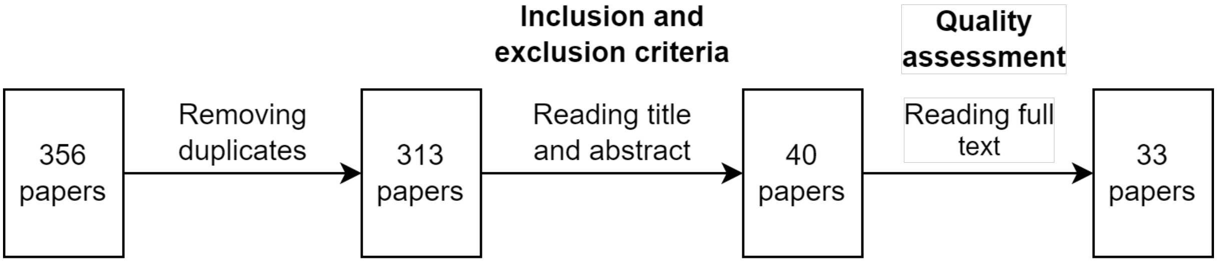


Figure 2: Visualization of the paper selection process

Table 6 presents the results of our inclusion criteria for selecting relevant papers in our study. Our analysis indicates that the majority of accepted papers (19 out of 40) centre on the design of a data lake architecture. However, there is a significant gap in the literature regarding the integration between data vaults and data sources, with only one paper meeting this inclusion criterion. Additionally, only eight papers discuss the management of batch and streaming data in a data lake, highlighting a potential area for further research.

Table 6: Inclusion criteria results

Inclusion criteria	**Number of included papers**
Discusses the design of a data lake architecture	19
Discusses the integration between a data lake and a data warehouse	12
Discusses the integration between a data source and a data vault	1
Discusses the management of batch and streaming data in a data lake	8
Total included papers	**40**

Table 7 presents the results of the exclusion process. 255 papers were eliminated from consideration due to their lack of relevance to our research questions. Additionally, five papers were excluded due to being incomplete or unavailable. It is important to note that our online database search was filtered to include only peer-reviewed papers published from 2017 onwards, and therefore, earlier, or non-peer-reviewed papers are not accounted for in this analysis.

Table 7: Exclusion criteria results

Exclusion criteria	**Number of excluded papers**
The paper is not related to the research questions	255
The paper is unavailable in English or Portuguese	2
Summary or mapping	11
Incomplete or unavailable paper	5
Total excluded papers	**273**

Of the papers that met the inclusion criteria, academic journals were the most common type of source, accounting for seventeen unique publications. The different publications of the academic journals selected in our study are presented in Table 8.

Table 8: Academic journals information

Publication	Number of Publications
Procedia Computer Science	2
Journal of Cleaner Production	1
ISPRS International Journal of Geo-Information	1
SN Computer Science	1
Information Sciences	1
ACM Transactions on Internet Technology	1
Future Generation Computer Systems	1
IEEE Access	1
Data Intelligence	1
Applied System Innovation	1
International Journal of Advanced Computer Science and Applications	1
Sensors	1
Journal of Intelligent Information Systems	1
Procedia CIRP	2
Baltic Journal of Modern Computing	1

In total, fifteen conference papers were collected and remained following the filtering process, of these, only two belonged to the same conference. The number of conference papers selected and their publications are presented in Table 9.

Table 9: Conference papers information

Conference	Number of publications
13th International Conference on Knowledge Management and Information Systems	1
2020 1st International Conference on Big Data Analytics and Practices (IBDAP)	1
2020 39th International Conference of the Chilean Computer Science Society (SCCC)	1
The 2nd International Conference	1
iiWAS2021: The 23rd International Conference on Information Integration and Web Intelligence	1
2020 3rd International Conference on Information and Computer Technologies (ICICT)	1
2022 45th Jubilee International Convention on Information, Communication and Electronic Technology (MIPRO)	1
The 2018 Artificial Intelligence and Cloud Computing Conference	1
iiWAS2019: The 21st International Conference on Information Integration and Web-based Applications & Services	1
17th International Conference on e-Business	1
2019 45th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)	1
2021 44th International Convention on Information, Communication and Electronic Technology (MIPRO)	1
2022 8th International Conference on Advanced Computing and Communication Systems (ICACCS)	1
IDEAS 2021: 25th International Database Engineering & Applications Symposium	2

Finally, eight peer-reviewed books were included in the research, whose number of publications by publisher are presented in Table 10

Table 10: Peer-reviewed books information

Publisher	Number of publications
Springer International Publishing	7
Elsevier	1

Papers meeting the inclusion criteria were sourced from four distinct bibliographic databases, with Scopus emerging as the main contributor, providing 18 out of the 40 total accepted papers. A breakdown of the databases that contributed to the selected papers is presented in Table 11.

Table 11: Included papers, by database

Bibliographical database	Number of included papers
Scopus	18
Science@Direct	6
ISI Web of Science	9
ACM Digital Library	7
Total included papers	40

3.3. GREY LITERATURE REVIEW

As the formal literature review gathered few sources and information regarding the topic, the research is complemented with a grey literature review.....

3.3.1. Planning

The grey literature review's planning phase is similar to the one seen in the formal literature review, with a few key differences, which will be presented in detail throughout the following subsections.

3.3.1.1. Necessity for a grey literature review

Before conducting a review of grey literature, it is crucial to first determine the necessity for such a review. Garousi et al. (Garousi et al., 2019) propose a checklist with the intent of helping researchers make this determination. The presence of one or more affirmative responses to the questions on this checklist indicates that the inclusion of grey literature in the review may be beneficial. The checklist aids in evaluating whether a grey literature review is advised and is presented in Table 12.

Table 12: Grey literature checklist

#	Question	Answer
1	Is the subject “complex” and not solvable by considering only the formal literature	Yes
2	Is there a lack of volume or quality of evidence or a lack of consensus of outcome measurement in the formal literature?	Yes
3	Is contextual information important to the subject under study?	Yes
4	Is it the goal to validate or corroborate scientific outcomes with practical experiences?	Partly
5	Is it the goal to challenge assumptions or falsify results from practice using academic research or vice versa?	Yes
6	Would a synthesis of insights and evidence from the industrial and academic community be useful to one or even both communities?	Yes
7	Is there a large volume of practitioner sources indicating high practitioner interest in a topic?	No

3.3.1.2. Search process

The grey literature search process followed the same search string employed for the formal literature search. However, instead of bibliographical databases, a Google search was conducted to identify relevant web articles. The search process was systematic and consistent, adhering to the predefined search string. The grey literature sources considered included reports, working papers, and non-peer-reviewed articles published from 2017 onwards. The search process was stopped when either 15 pages of search results were reached, or when data exhaustion occurred, which refers to the point when no new relevant sources were found. In addition, supplementary sources were obtained by collecting grey literature references cited within the formal literature papers that were included. This technique is often referred to as ”snowballing” (Garousi et al., 2019) and was utilized to enrich the depth of the literature review. The search process details are displayed in Table 14.

Table 13: Search process details

Search query	‘Data Lake’ AND ”Architecture” AND (”Data Vault” OR ‘data warehouse’ OR ”lambda architecture”)
Stopping criteria	15 pages or data exhaustion
Databases	Google search

3.3.1.3. Inclusion and Exclusion criteria

As the sources from grey literature do not contain titles or abstracts, short works are fully read, and longer works are briefly read in order to gather a sense of what the source discusses. The inclusion criteria remain the same, and the exclusion criteria remain mostly the same as in formal literature, excluding “Paper is not peer-reviewed” and “Paper is not from conference proceeding, scientific journal, or peer-reviewed book” as they are not applicable to the grey literature. The criteria are listed in Table 14.

Table 14: Inclusion and exclusion criteria for grey literature

Inclusion criteria	**Exclusion criteria**
Discusses the design of a data lake architecture	The source is not related to the research questions
Discusses the integration between a data lake and a data warehouse	The source is unavailable in English or Portuguese
Discusses the integration between a data source and a data vault	Summary or mapping
Discuss the management of batch and streaming data in a data lake	Incomplete or unavailable source
	Duplicate from formal literature
	Source was published before 2017

3.3.1.4. Quality assessment

Quality assessment in grey literature differs from the quality assessment for formal literature, as the publication process for grey literature is not as strict and controlled as it is for scientific papers Garousi et al., 2019. The quality assessment questions in grey literature look to evaluate the sources on a wide variety of factors such as authority of the producer and objectivity. Each question has 3 possible answers: ”Yes” (1 point), ”No” (0 points), and ”Maybe” (0.5 points). Sources with a quality score of 7 of higher are considered usable, while the rest are discarded. The quality assessment questions used are presented in Table 15.

Table 15: Quality assessment questions for grey literature

Criteria	Description	Weight
Authority	Q1: Is the publishing organisation reputable?	1
	Q2: Has the author published other work in the field?	1
	Q3: Does the author have expertise in the area?	1
Methodology	Q4: Is the aim of the source clear?	1
	Q5: Does the source follow a stated methodology?	1
	Q6: Are claims made in the source strongly supported by authoritative references?	1
	Q7: Does the source cover a specific question?	1
Objectivity	Q8: Is there a vested interest?	1
	Q9: Does the work seem to be balanced in presentation?	1
	Q10: Are the conclusions supported by the data?	1
Date	Q11: Is the date of the source clearly stated?	1
Related sources	Q12: Does the source link to key related GL or formal literature?	1
Novelty	Q13: Does the source add something unique to the research?	1
	Q14: Does the source strengthen or refute a current position?	1
	Quality Score threshold	**7**

3.3.2. Conducting

In total, 152 sources were gathered from the grey literature for review. Each source was either skimmed or thoroughly read, depending on its length. Applying the predefined inclusion criteria, we selected 36 sources, while excluding 116 based on the exclusion criteria. Following this, we evaluated the included papers using a pre-determined set of quality assessment questions. Out of the 36, only 12 sources scored above the minimum threshold for usability. Figure 3 offers a visual representation of this selection and filtering process.

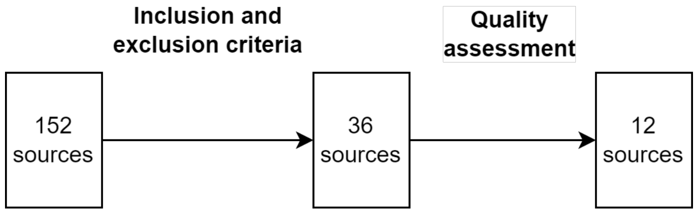


Figure 3: Visualization of the source selection process for grey literature

Table 16 presents the results of the application of inclusion and exclusion criteria for the identification of relevant grey literature sources. Notably, more than half of the accepted sources (22 out of 36) focused on the data lake architecture. As with the formal literature, the grey literature also exhibits a significant research gap in the integration of data vaults and data sources, with only two sources fulfilling the inclusion criteria.

Table 16: Inclusion criteria results for grey literature

Inclusion criteria	**Number of included sources**
Discusses the design of a data lake architecture	22
Discusses the integration between a data lake and a data warehouse	6
Discusses the integration between a data source and a data vault	2
Discusses the management of batch and streaming data in a data lake	6
Total included sources	**36**

Regarding the exclusion criteria, 108 out of the 116 excluded sources were disqualified due to their lack of relevance to the research questions. The remaining seven sources were excluded based on the remaining, less frequent criteria. The results of the exclusion process are presented in Table 17.

Table 17: Exclusion criteria results for grey literature

Exclusion criteria	Number of excluded sources
The source is not related to the research questions	108
The source is unavailable in English or Portuguese	2
Duplicate from formal literature	1
Incomplete or unavailable source	4
Source was published before 2017	1
Total excluded sources	**116**

3.4. REPORTING

In this section, we analyze and synthesize key findings from the review of both formal and grey literature. Each of the following subsections is dedicated to a research question posed in the literature review. Within these, we present the primary findings pertinent to each question and acknowledge the corresponding references.

3.4.1. RQ1: What architecture models are used in current data lake implementations?

In this subsection, the primary data lake architecture models identified in the literature are presented. An overview of each model's structure and functionality is provided, accompanied by a visual representation through a diagram. While the displayed diagrams do not represent the totality of the different currently implemented architectures, they represent the main archetypes found in the literature, with other architectures often building upon these foundational models.

A common approach among the selected papers is the ”zone” or layer-based architecture. In this approach, data in a data lake moves through a sequence of layers, each with a distinct functionality. The configuration of layers within a zone-based data lake architecture is not fixed, and researchers have put forth various proposals with different numbers of layers and different functionalities for each of the layers.

The three-zone architecture is the most utilized zone-based approach in the literature (Ravat and Zhao, 2019; Saddad et al., 2020; Sakr and Zomaya, 2019; Sarramia et al., 2022; Zhao et al., 2021). In this architecture, data from the data lake's sources are initially loaded into a raw data zone, where it is ingested in its native format and stored persistently with minimal to no processing. The process zone succeeds the raw data zone, it is used to store data during its various stages of processing before it is ready for consumption. Once data is formatted and consumption-ready, it is loaded into the access zone, which serves as a point of consumption for data lake users and applications.

In addition to these three zones, the architecture also contains a governing zone. Unlike the other zones, the govern zone is not part of the data pipeline through which data goes through. Instead, it ensures data quality, data life cycle, data access, and metadata management. Its role is to apply data governance policies to the other zones and ensure the reliability of the data lake (Ravat and Zhao, 2019). A model of a data lake architecture with three zones is displayed in Figure 4.

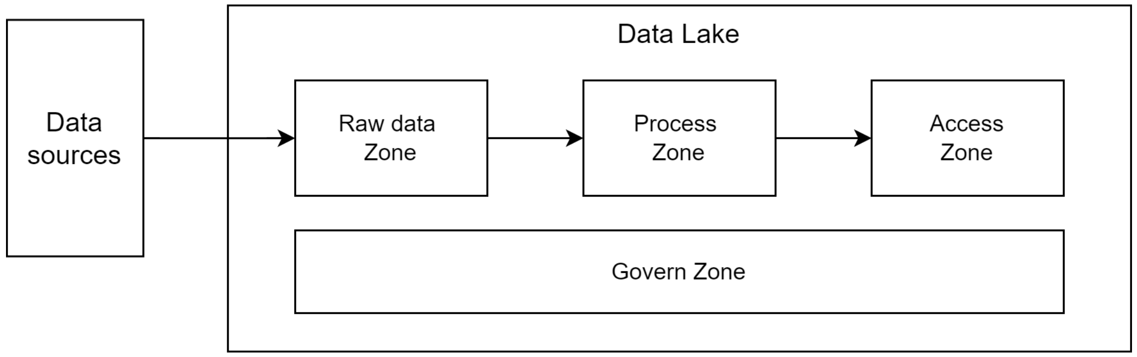


Figure 4: Data lake architecture with three sequential zones and a govern zone (adapted from (Ravat and Zhao, 2019))

Researchers have expanded upon this architecture by incorporating small features that are specific to their use cases and requirements. For instance, Oukhouya et al. (2021) have integrated a trusted data zone, which follows the process zone. The trusted data zone contains clean and transformed data that is meant specifically for loading into a Data Warehouse. The inclusion of other zones that are not part of the data lake's data pipeline, similar to the govern zone, are also suggested (Liu et al., 2021). The 'data source' zone precedes the raw data zone and is used to gather basic properties of data in the data sources, such as volume, velocity, and connectivity. Additionally, a 'data ingestion' zone provides tools for data engineers to ingest data into the data lake, either in batches or in real time, based on the information gathered in the 'data source' zone.

Another common approach to data lake architectures is the five-zone architecture (Li et al., 2018; Mitruś, 2021; Pisoni et al., 2021; Sharma, 2018). The five-zone architecture expands upon the standard three-zone architecture by incorporating a raw zone, a trusted zone, and a refined zone that mirrors the raw, process, and access zones of the three-zone architecture (Sharma, 2018). Additionally, two new zones are introduced to the data lake: the transient landing zone and the Sandbox Zone. The transient landing zone is located before the raw data zone and provides a temporary storage location for source data, particularly beneficial in highly regulated industries where data must undergo rigorous security measures before entering the data lake.

The Sandbox Zone serves as a dedicated workspace for data lake users to conduct data experiments within a secure and controlled environment. Data from any of the other zones can be imported into the Sandbox Zone. The knowledge and insights gained in the Sandbox Zone can then be transferred back to the Raw Zone, allowing the derived data to function as a new source for further analysis and exploration. An adaption of this model is depicted in Figure 5.

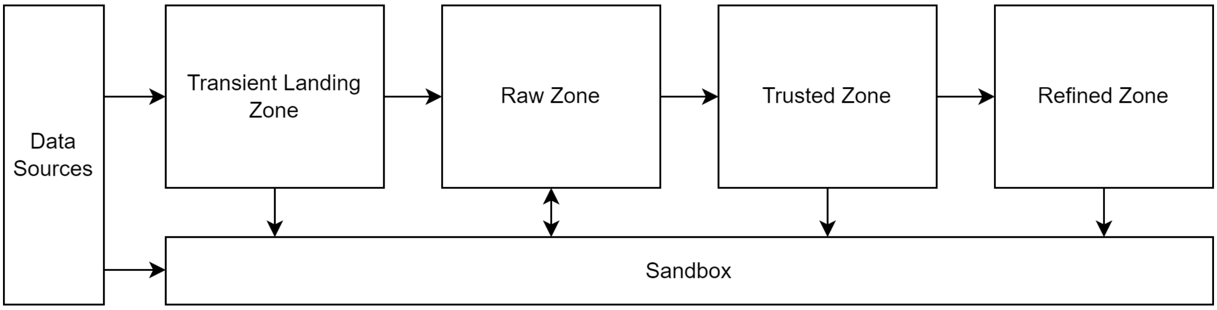


Figure 5: Data lake architecture with five zones (adapted from (Sharma, 2018))

Additionally, an application zone (Mitruś, 2021) can be added to the data lake, which follows the trusted/process zone and is used to store data meant for application consumption, this being automated, non-human consumption. Examples of applications for this zone are data loaded automatically into a data warehouse, or machine learning models that are calculated based on data from the data lake.

Another approach to data lake architecture is The Data Pond architecture (B. Inmon, 2016). In the Data Pond, data is not moved sequentially from one zone to another. Instead, data is first loaded into a raw data pond before being moved into one of three other ponds: The application data pond, which contains structured data generated from applications and transactions, such as sales and shipment data. This is the most business-relevant data. The textual data pond contains unstructured textual data such as emails and call centre conversations. This data requires disambiguation for analysis as it holds no inherent business value. The analog data pond contains mechanically generated data that is repetitive and often has little to no business value.

Data not currently needed for analysis is offloaded into an archival data pond, where it can be accessed and analyzed later. This approach allows for more efficient analysis within each pond. Data not currently needed for analysis is offloaded into an archival data pond, where it can be accessed and analyzed later. This approach allows for more efficient analysis within each pond. A visual representation of the Data Pond is depicted in Figure 6.

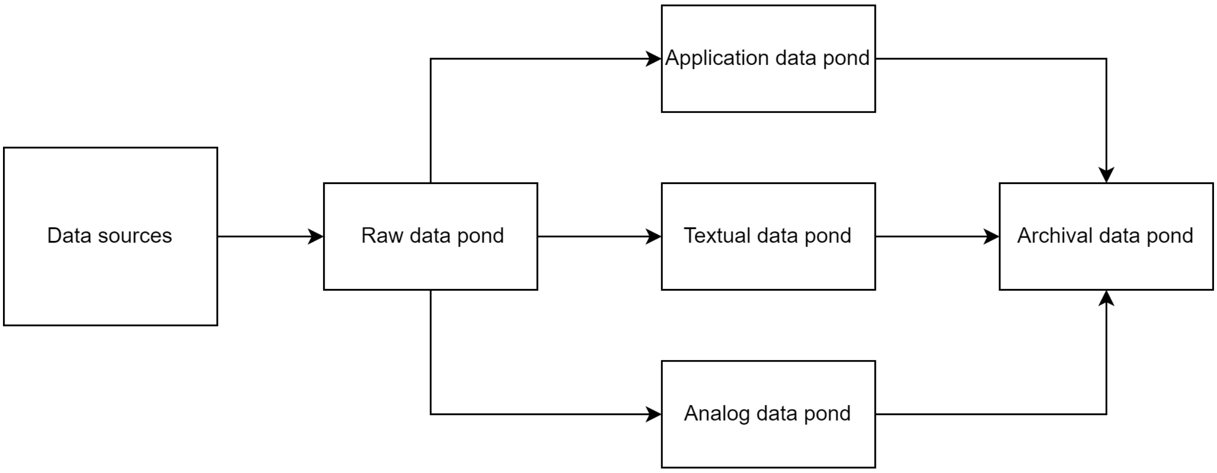


Figure 6: Data Pond architecture (adapted from (B. Inmon, 2016))

Table 18 summarizes the primary sources referenced in this subsection, categorized by the specific data lake architecture models they address. Three main architecture models have been identified and described in this review, these being: the three-zone architecture, the five-zone architecture, and the data pond architecture.

Table 18: RQ1, sources referenced

Data Lake architecture models	**Sources**
Three-zone architecture	(Liu et al., 2021; Oukhouya et al., 2021; Ravat and Zhao, 2019; Saddad et al., 2020; Sakr and Zomaya, 2019; Sarramia et al., 2022)
Five-zone architecture	(Li et al., 2018; Mitruś, 2021; Sharma, 2018)
Data Pond	(B. Inmon, 2016)

3.4.2. RQ2: How are current data lake implementations integrated with data warehouses?

The simplest and most common approach is utilizing a data lake and a data warehouse in a sequential manner (Jemmali et al., 2022; Oukhouya et al., 2023; Saddad et al., 2020), where data is stored in a data lake and is then loaded into a data warehouse. Data can also go from the data warehouse to the data lake, by offloading data from the data warehouse to the data lake, in order to relieve performance and storage in the data warehouse. Additionally, the data lake can include a 'trusted data zone' (Oukhouya et al., 2023) specifically meant for loading data into a data warehouse. In a sequential approach, the data lake replaces the integration layer of the data warehouse and acts as the single source of truth.

A less common approach to data architecture is the parallel approach, which involves using both a data lake and a data warehouse without integrating them. Data is loaded into either the data lake or data warehouse depending on the type of data source. Unstructured and semi-structured data is loaded into the data lake, while structured data is loaded directly into the data warehouse, though it may also be stored in the data lake. This approach can lead to data silos, as it is difficult to maintain consistency across data sources (Herden, 2020).

The *data lakehouse* is an increasingly popular alternative to the traditional data lake plus data warehouse architecture. Its purpose is to address common issues found in both data lakes and data warehouses by combining the best qualities of both. It is unique in the sense that it supports both structured queries, which are usually performed in the data warehouse, and also supports unstructured analytics, which are typically performed in the data lake (Orescanin and Hlupic, 2021).

The most commonly used data lakehouse architecture is the Medallion architecture, popularized by Databricks (2022). This architecture is composed of three layers: bronze, silver, and gold. Data flows through each layer sequentially, based on its processing stage. The bronze layer serves as the raw data layer, where data is ingested from various sources without any processing. Its goal is to provide historical archives, cold storage, auditability, and data lineage. After the data is merged, conformed, and cleansed, it is moved to the silver layer.

During this process, only minimal transformations and data cleansing rules are applied, in accordance with the Extract, Load, and Transform (ELT) data engineering paradigm, which prioritizes the speed at which data can be ingested and delivered. Finally, the gold layer contains highly refined and aggregated data, organized into project-specific, consumption-ready datasets. Typically, star-schema based data models or data marts fit into the gold layer. An adaptation of the lakehouse medallion architecture is depicted in Figure 7.

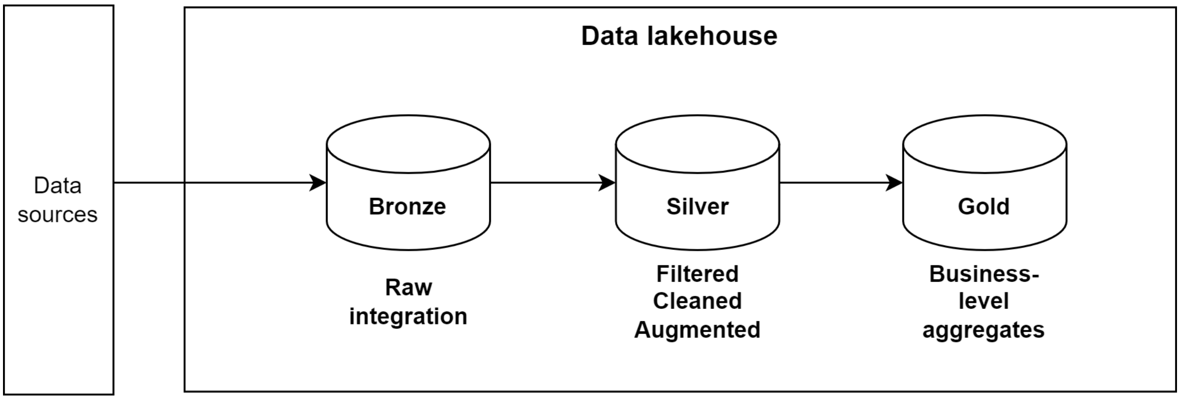


Figure 7: Data lakehouse medallion architecture (adapted from (Databricks, 2022))

Bhatt et al. (2022) propose an implementation of the Data Vault within the medallion architecture. In this implementation, the bronze layer of the data lakehouse acts as the data vault's staging zone. Data is converted from formats such as CSV, Parquet, and JSON into Delta-formatted tables, to optimize the efficiency of the following process.

The silver layer's focus is on the speed and agility with which it can ingest and deliver data, adapting the ELT data engineering paradigm, these features coincide with the Data Vault modelling methodology's key features, as such, data vault modelling is particularly well-suited for this layer. The gold layer contains Inmon-style data marts. The authors also highlight how the implementation of a Data Vault in the silver layer allows for the seamless loading of a dimensional model Data Warehouse in the Gold layer. Specifically, the use of hubs in the Data Vault facilitates key management by enabling the conversion of natural keys to surrogate keys in the data warehouse through the use of identity columns. Additionally, satellites facilitate loading dimensions as they contain all relevant attributes, and links simplify the process of loading fact tables by containing all necessary relationships. A representation of a Data Vault within the medallion architecture is depicted in Figure 8.

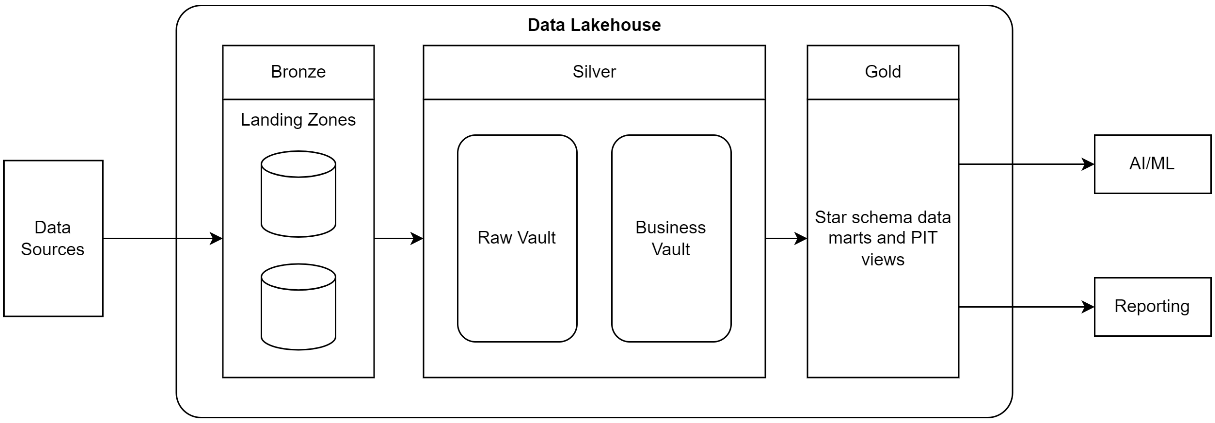


Figure 8: Data Vault within the medallion architecture (adapted from (Bhatt et al., 2022))

Dehghani (2019) introduces the concept of a data mesh, which challenges the traditional approach of centralizing data in a data lake and instead advocates for a distributed, domain-driven data architecture governed by centralised standards. In a data mesh, each domain has its own data team and datasets, which are made available to other domains through APIs. Data lakes and warehouses serve as nodes on the mesh, storing data from a single domain rather than from multiple domains. Implementing a data mesh at the enterprise level can be a significant challenge, as it requires a shift in thinking about data and a departure from the existing paradigm of the data lake.

The sources for the different approaches found in the literature for integrating data lakes and data warehouses are presented in Table 19.

Table 19: RQ2, sources referenced

DL and DW integration approaches	Sources
Sequential approach	(Jemmali et al., 2022; Oukhouya et al., 2023; Saddad et al., 2020)
Parallel approach	(Herden, 2020)
Data Lakehouse	(Bhatt et al., 2022; Databricks, 2022; Orescanin and Hlupic, 2021)
Data mesh	(Dehghani, 2019)

3.4.3. RQ3: How do current Data Lake implementations store and process batch and real-time data?

Most data lake implementations proposed support both ingestion and processing of both batch and real-time data. A transient landing zone can be added to a data lake, as a non-persistent storage area, for when the specifications of the raw data zone diverge from those of the ingested data, and data needs to be ingested at a faster rate than what the raw data zone can provide (Giebler et al., 2020). Data is first ingested into the Transient Loading Zone at a high rate, and then moved into the Raw Zone in batches. The landing zone then forwards streaming data to both a real-time raw zone and a batch raw zone, based on hybrid processing architectures, namely the Lambda Architecture (Kiran et al., 2015).

Lee (2020) presents the Delta architecture, which addresses the limitations of the lambda and kappa architectures (Kreps, 2014) by continuously and incrementally processing new data in a cost-effective manner. Conventional data lakes are based on the principle of immutability, leading to inefficiencies in batch processing due to the requirement to recreate the entire data structure whenever new transformations are performed on existing data (van 't Westeinde, 2022).

The Delta architecture departs from this approach, viewing incoming data as a ”delta,” or a difference from an existing record, which can be an insert, delete, or update. The delta architecture eliminates the need for having different APIs, engines and codebases for batch and streaming, instead using a single codebase. This means data does not have to be treated differently according to its speed of ingestion or processing method (Jia, 2020). The delta architecture is achieved using the “delta lake”, an open source optimized storage layer, which brings capabilities similar to those of a Data Warehouse, such as ACID transactions to a data lake, increasing its performance and reliability in data processing pipelines (Databricks, 2023).

The sources for the different approaches found in the literature for managing batch and real-time data are presented in Table 20.

Table 20: RQ3, sources referenced

Batch and real-time management approaches	**Sources**
Introduction of a transient landing zone	(Giebler et al., 2020)
Delta architecture	(Databricks, 2023; Jia, 2020; Lee, 2020)

3.5. DISCUSSION

In this section, a multivocal systematic literature review was conducted, following the guidelines of Kitchenham (2004) for performing a systematic literature review, and Garousi et al.'s (2019) guidelines for including grey literature in a multivocal literature review. This allowed us to gather an understanding on data lake architectures, data lake integration with data warehouses, and batch and real-time storing and processing in data lakes.

The main conclusions obtained from an analysis of 51 sources, from formal and grey literature are the following:

- The most widely adopted data lake architecture in formal literature is a zone-based architecture with three zones, with data going sequentially from a zone to another according to its degree of processing, a zone used to store raw data, a process zone to store intermediate data and an access zone that stores highly processed, curated data.
- The five-zone architecture is the second most adopted architecture in the literature, particularly popular in grey literature. It builds on the three-zone architecture by adding a landing zone to temporarily store data before it is loaded into the raw data zone and adds a sandbox zone that acts as a playground for data lake users to experiment with data.
- The Data Lakehouse architecture, although widely adopted in grey literature, is still not widely adopted in scientific papers, researchers mostly choose to go for a Data Lake + Data Warehouse architecture.
- Although it is not a proprietary architecture, most Data Lakehouse examples found in the literature are adaptations of the Medallion Architecture, popularized by Databricks.
- There are very few occurrences in the literature of a data lake being used alongside an EDW based on Data Vault 2.0.

This review also revealed that many topics of interest lack research and as such there are still questions that remain unanswered, these being:

- There is a clear lack of research on the best practices for designing a data lake architecture, optimized for loading data into a Data Warehouse, particularly a Data Vault 2.0 based EDW. Researchers rarely noted the integration with a data warehouse as a factor.

- Mentions of Data Vault modelling in a data lake context are scarce, with one article discussing the implementation of a Data Vault within a Data Lake Medallion architecture. Some benefits are listed, but the article does not go into sufficient detail. Additionally, as the article was created by an organization, conclusions must be taken sceptically.

- Although both formal and grey literature revealed that a majority of data lake implementations have the capability of both batch and real-time ingestion and processing, the best practices and steps that need to be taken in order to accommodate this capability are scarcely documented.

4. DESIGN AND DEVELOPMENT

The third step of DSR is Design and Development, which entails the creation of an artefact aimed at addressing critical organizational problems (Hevner et al., 2004).

In this section, a comprehensive description of the artefact will be provided. We begin with the introduction of the data lake serving as the source for the Data Vault 2.0 EDW. This is followed by an introduction of the primary Data Vault modelling concepts foundational to the artefact.

Subsequently, we present the artifact - a unique Data Vault model, developed and explained modularly. We delve into each of the model's concepts it portrays, such as customers, current accounts, and their interrelations, in detail, treating each as a discrete unit within the larger framework. This thorough and structured design and development process not only allowed us to create a model that effectively addresses the complex realities of our organizational context, but it also aligns with the key philosophy of Data Vault modelling: the model should be constructed around the core business concepts Hultgren, 2012.

4.1. FINAL DATA LAKE ARCHITECTURE

Our review of current literature on data lake architectures and their integration with data warehouses has facilitated a comprehensive understanding of best practices in designing data lake architectures. We identified not only the most prevalent and essential features of data lake architecture models but also specific features that are particularly relevant to our research.

We developed the data lake architecture in a collaborative process with various members of the organization. The design integrates key concepts from relevant literature and is tailored to meet the organization's specific needs.

The chosen architecture is meant to provide a flexible, organized, scalable, and future-proof solution with the ability to incorporate various amounts of different sources, clearly separate raw data from processed data, and serve both as an access zone for users and as a source for applications such as data warehouses. The data lake consists of five distinct zones, each playing a specific role:

- Landing Zone: This zone is the initial point of data ingestion from the organization's various data sources. Data, in their native formats, are stored here temporarily before being transferred to the Raw Zone. Once the data have been moved, they are erased from the Landing Zone. This approach was first presented in the literature by Sharma (Sharma, 2018).
- Raw Zone: All data are persistently stored in this zone in their raw formats. The Raw Zone allows the organization to maintain a repository of raw data, from which all data in subsequent zones and applications can be traced Giebler et al., 2020. This capability is of significant importance, especially considering the crucial role of traceability and auditabil-

ity in heavily regulated industries, such as banking. This feature is consistent across most data lake architecture models presented in the literature

- Structured Zone: In this zone, data are transformed from their original format into an adequate format for optimal data cleaning. It acts as a counterpart to the process zone, a zone prevalent across most data lake architecture models found in the literature.
- Enriched Zone: This zone acts as the primary consumption point within the data lake. Here, data are completely transformed according to business requirements, and may even be consolidated into project-ready packages to optimize consumption. This zone is pivotal as it aids in managing user permissions, allowing only final users to access this zone, while privileged users can access the rest of the zones within the data lake.
- Processed Zone: This zone serves as a source for application consumption, like supplying data warehouses and data marts. This mirrors the application zone proposed by Mitruś (2021). Having a distinct zone for automatic loading into applications simplifies and declutters the enriched zone, thereby enhancing user experience by making data access faster and easier.

A depiction of the finalized data lake architecture is shown in Figure 9.

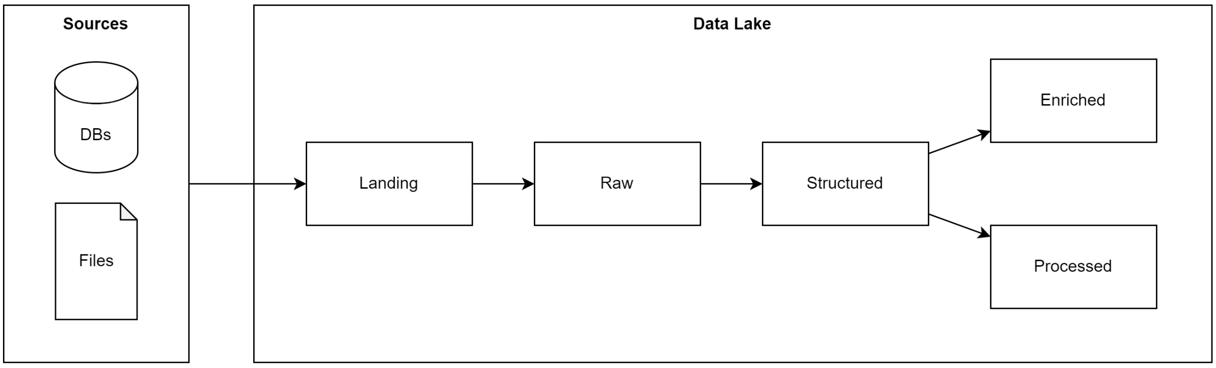


Figure 9: Final data lake architecture

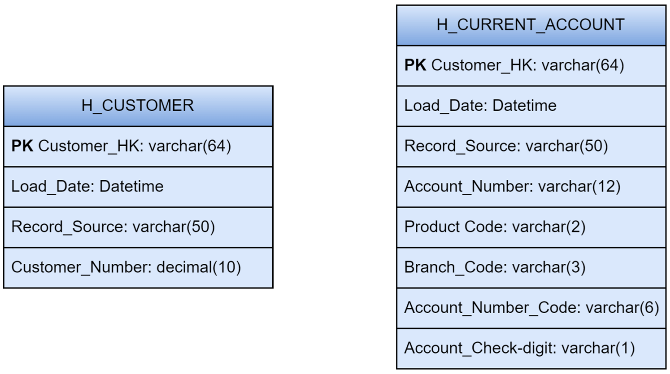
In a fully-developed Data Vault 2.0 EDW, data from the data lake's processed zone would be utilized as its source. For the purpose of this research, a subset of the Data Vault model is implemented within the data lake's enriched zone. This serves as a proof of concept for a full-scale implementation.

4.2. DATA VAULT 2.0 MODELLING CONCEPTS

This subsection aims to outline the primary concepts of Data Vault modelling, setting the stage for the subsequent introduction of the newly created artefact. This strategy ensures the following section can emphasize the adopted approaches for accurately capturing the organization's business logic, allowing for a deeper exploration of intricate decisions and less common features associated with Data Vault modelling.

4.2.1. Hubs

The hub tables present in our model are illustrated in figure 10. In this subsection, the hub concept is explained, as well as the meaning of the fields they contain.



H_CUSTOMER
PK Customer_HK: varchar(64)
Load_Date: Datetime
Record_Source: varchar(50)
Customer_Number: decimal(10)

H_CURRENT_ACCOUNT
PK Customer_HK: varchar(64)
Load_Date: Datetime
Record_Source: varchar(50)
Account_Number: varchar(12)
Product Code: varchar(2)
Branch_Code: varchar(3)
Account_Number_Code: varchar(6)
Account_Check-digit: varchar(1)

Figure 10: Hubs present in the Data Vault 2.0 model

In Vault 2.0 modelling, business objects are represented as hubs, which are inherently independent entities that hold their own intrinsic meaning without the need for additional context. Each hub is identified by a unique business key that corresponds to the object it represents. For instance, when modelling a ”Customer,” the business key for the ”Customer Hub” would be a unique identifier designated by the organization, in our particular case, a customer number.

In addition to the business key, hubs include additional fields, which further enhance their capability. These additional attributes, which are described below, serve various purposes in Data Vault modelling.

4.2.1.1. Hash Key

During the execution of an Extract, Transform, Load (ETL) job, a comparison is conducted to determine whether the business keys from the data source already exist in the target Data Vault hub. This process is optimized by the application of hash keys, typically calculated using MD5 hashing on the hub’s business key. These hash keys are fixed-length strings, as lookups on such strings tend to be faster than those conducted on variable-length strings. It’s worth noting, however, that these hash keys hold no inherent meaning; they exist solely for the purpose of improving operational simplicity and performance. (Linstedt and Olschimke, 2015, p. 118)

4.2.1.2. Load Date

The ”load date” field denotes the moment an entry is first introduced to a hub. Data is usually uploaded to a Data Vault in batches, necessitating that all data within a batch share the same load date. This uniformity facilitates tracing errors and identifying issues within a given batch.

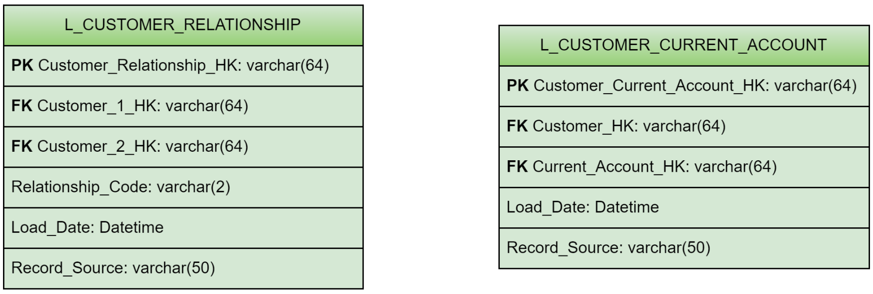
If a problem arises during a data load cycle, the process can be conveniently traced back to before the most recent batch was loaded, allowing for the correction of errors before resuming the loading process. Notably, once defined, load dates should remain unaltered. (Linstedt and Olschimke, 2015, p. 119)

4.2.1.3. Record Source

The ”record source” field denotes the originating data source of entries in a hub, serving as a crucial attribute for maintaining auditability within a Data Vault. When combined with the ”load date” attribute, every record in the Data Vault effectively documents its source and the time of entry (Linstedt and Olschimke, 2015, p. 119). This functionality is particularly important for highly regulated industries like banking, which are subject to rigorous regulations and regular audits.

4.2.2. Links

The link tables incorporated in our model are depicted in Figure 11. This subsection provides a detailed discussion of the link concept.



L_CUSTOMER_RELATIONSHIP
PK Customer_Relationship_HK: varchar(64)
FK Customer_1_HK: varchar(64)
FK Customer_2_HK: varchar(64)
Relationship_Code: varchar(2)
Load_Date: Datetime
Record_Source: varchar(50)

L_CUSTOMER_CURRENT_ACCOUNT
PK Customer_Current_Account_HK: varchar(64)
FK Customer_HK: varchar(64)
FK Current_Account_HK: varchar(64)
Load_Date: Datetime
Record_Source: varchar(50)

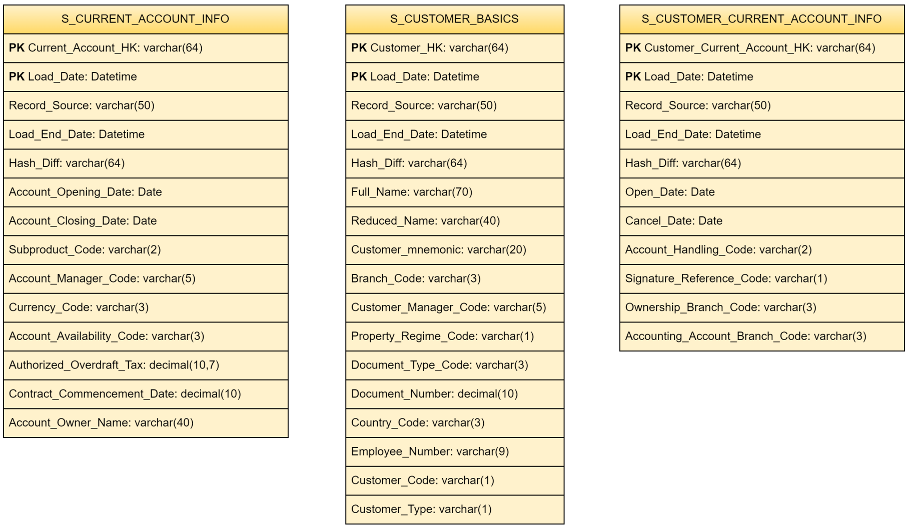
Figure 11: Modelled link tables

Links within the Data Vault modelling paradigm symbolize relationships between business objects, which are represented as hubs. These links facilitate connections through their business keys, and represent many-to-many relationships (Linstedt and Olschimke, 2015, p. 122). Every link comprises the hash key of each connected hub, in addition to the ”Load Date” and ”Record Source” fields (Linstedt and Olschimke, 2015, p. 122). A more comprehensive explanation of these fields can be found in the preceding subsection.

Links should not contain any descriptive data. If for example, a link represents a contract between a customer and a product, the specific details of the contract such as the contract amount and its commencement date are stored in a satellite table connected to the link, not the link itself. (Linstedt and Olschimke, 2015, p. 122)

4.2.3. Satellites

Our model encompasses a multitude of satellites, each populated with a wide range of attributes. Due to the extensive number, only a select subset of these satellites is illustrated in Figure 12. This subsection provides a detailed discussion of the satellite concept.



S_CURRENT_ACCOUNT_INFO
PK Current_Account_HK: varchar(64)
PK Load_Date: Datetime
Record_Source: varchar(50)
Load_End_Date: Datetime
Hash_Diff: varchar(64)
Account_Opening_Date: Date
Account_Closing_Date: Date
Subproduct_Code: varchar(2)
Account_Manager_Code: varchar(5)
Currency_Code: varchar(3)
Account_Availability_Code: varchar(3)
Authorized_Overdraft_Tax: decimal(10,7)
Contract_Commencement_Date: decimal(10)
Account_Owner_Name: varchar(40)

S_CUSTOMER_BASICS
PK Customer_HK: varchar(64)
PK Load_Date: Datetime
Record_Source: varchar(50)
Load_End_Date: Datetime
Hash_Diff: varchar(64)
Full_Name: varchar(70)
Reduced_Name: varchar(40)
Customer_mnemonic: varchar(20)
Branch_Code: varchar(3)
Customer_Manager_Code: varchar(5)
Property_Regime_Code: varchar(1)
Document_Type_Code: varchar(3)
Document_Number: decimal(10)
Country_Code: varchar(3)
Employee_Number: varchar(9)
Customer_Code: varchar(1)
Customer_Type: varchar(1)

S_CUSTOMER_CURRENT_ACCOUNT_INFO
PK Customer_Current_Account_HK: varchar(64)
PK Load_Date: Datetime
Record_Source: varchar(50)
Load_End_Date: Datetime
Hash_Diff: varchar(64)
Open_Date: Date
Cancel_Date: Date
Account_Handling_Code: varchar(2)
Signature_Reference_Code: varchar(1)
Ownership_Branch_Code: varchar(3)
Accounting_Account_Branch_Code: varchar(3)

Figure 12: Subset of the modelled satellite tables

In essence, a satellite is designed to store descriptive data pertaining to either a business object, represented as hubs, or a relationship, represented as links. Given the dynamic nature of business, the data contained within a satellite is subject to changes over time. However, it's critical to understand that the data within a satellite itself is not modified. This is due to the principles of the Data Vault modelling methodology, which maintains a stringent 'insert-only' rule, excluding updates or deletions.

Therefore, any updated information necessitates the creation of a new satellite instance. The identification of this instance relies on the parent's hash key, either from the hub or link it is connected to, along with a timestamp marking the change. This approach enables effective change tracking within a Data Vault 2.0 EDW. (Linstedt and Olschimke, 2015, p. 133)

4.2.3.1. Load End Date

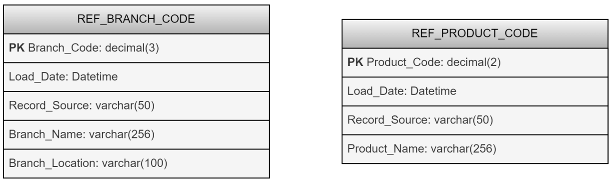
Similar to hubs and links, satellites incorporate a "Record Source" field that signifies the originating source of the data, along with the parent's hash key and the load date of the change. Unique to satellites is an additional required field: "Load End Date". Notably, this is the only attribute that is ever updated in a satellite. This particular field denotes the date and time when the satellite entry is deemed invalid. Consequently, when a new entry is loaded and assigned a 'Load Date', the 'Load End Date' of the prior valid entry is adjusted to equal the 'Load Date' of the new valid entry. This adjustment is instrumental in enhancing performance during data retrieval from a Data Vault. (Linstedt and Olschimke, 2015, p. 138)

4.2.3.2. Hash Diff

Moreover, satellites may contain an optional ”Hash Difference” (hash diff) field. This field serves as a hash value for the descriptive data in a satellite entry. Its design ensures that if any value in the satellite table, which is used to compute the hash diff, changes, the hash diff itself also changes. This process ensures that new satellite entry is added only when changes occur within the satellite, as directly comparing every attribute would be inefficient. The comparison is thereby conducted solely between the preceding hash diff and the newly computed hash diff value. A change in this value prompts the creation of a new satellite entry featuring the updated values. In the absence of a change, no new satellite entry is created. (Linstedt and Olschimke, 2015, p. 139)

4.2.4. Reference Tables

Unlike hubs, links, and satellites, reference tables are not part of the core architecture. However, they are used often in Data Vault 2.0 modelling (Linstedt and Olschimke, 2015, p. 184). The reference tables utilized in our model are depicted in Figure 13.



REF_BRANCH_CODE
PK Branch_Code: decimal(3)
Load_Date: Datetime
Record_Source: varchar(50)
Branch_Name: varchar(256)
Branch_Location: varchar(100)

REF_PRODUCT_CODE
PK Product_Code: decimal(2)
Load_Date: Datetime
Record_Source: varchar(50)
Product_Name: varchar(256)

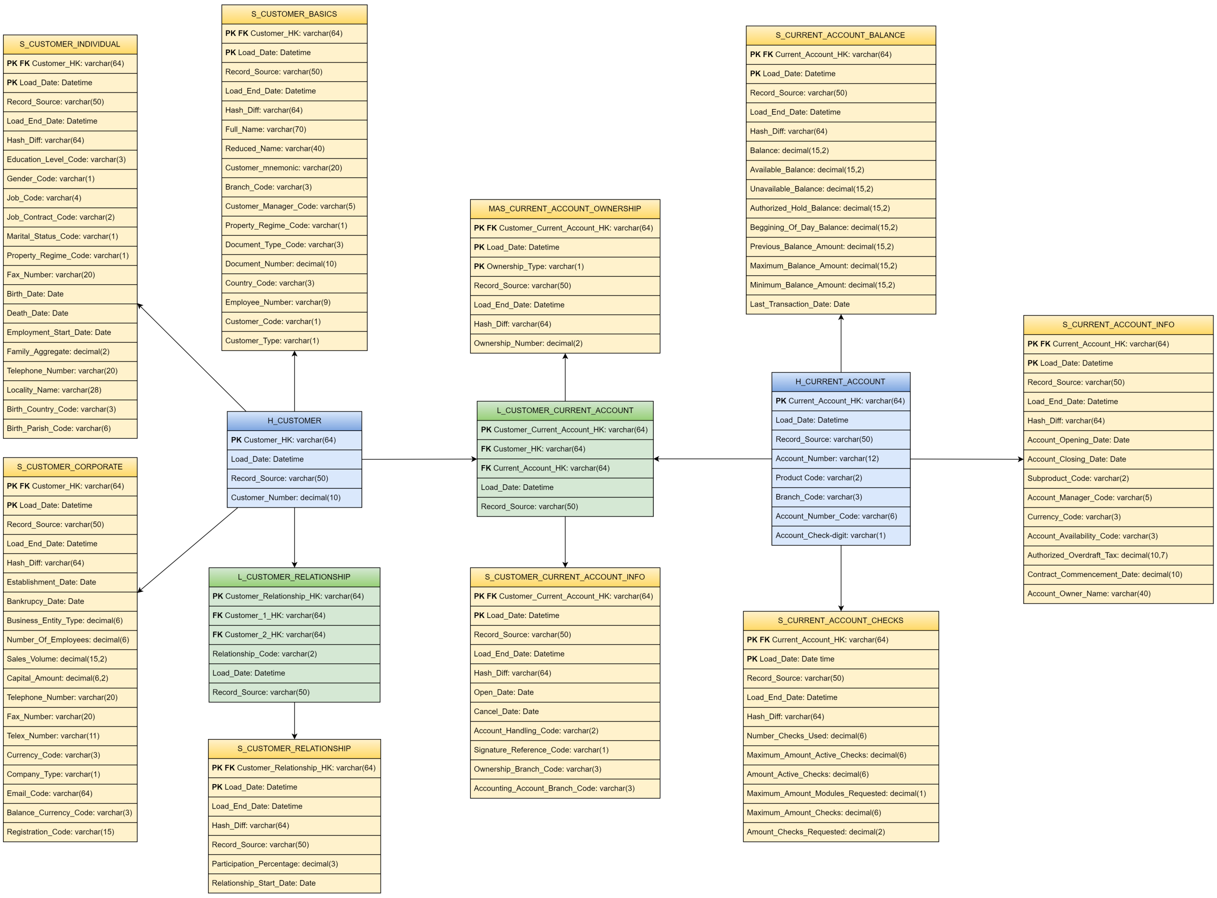
Figure 13: Modelled reference tables

Companies regularly employ data codes that don’t fit the definition of business keys. Such codes do not represent distinct business objects and thus are not modelled as hubs. Often these codes refer to information not strictly under the organization’s control, such as postal codes, which, while serving as an attribute, do not necessarily represent a business object. However, for organizations whose operations revolve around mail and deliveries, postal codes may very well be classified as business keys due to their direct relevance to the business operation. (Linstedt and Olschimke, 2015, p. 185)

Reference tables are characterized by a private key structured as a code, supplemented by ”Load Date” and ”Record Source” fields, along with one or more descriptive attributes. As an example in our model, the ”Branch Code” reference table includes attributes such as the name and location of the branch. The ”Branch Code” is found in multiple satellite tables throughout the model, where these attributes in turn link to the ”Branch Code” reference table, providing detailed information about the branches. In this context, branches are not modelled as hubs because they are not classified as business objects within the organization. Similarly, ”Product Code” follows the same modelling approach, with its reference table holding the necessary descriptive details about the products, although the products themselves are not considered business objects.

4.3. MODEL PRESENTATION

The Data Vault concepts outlined in Section 4.2. were meticulously examined and subsequently leveraged to develop a Data Vault model of the bank's customers and their relationships with current accounts, as depicted in Figure 14.



S_CUSTOMER_INDIVIDUAL
PK FK Customer_HK: varchar(64)
PK Load_Date: Datetime
Record_Source: varchar(50)
Load_End_Date: Datetime
Hash_Diff: varchar(64)
Education_Level_Code: varchar(3)
Gender_Code: varchar(1)
Job_Code: varchar(4)
Job_Contract_Code: varchar(2)
Marital_Status_Code: varchar(1)
Property_Regime_Code: varchar(1)
Fax_Number: varchar(20)
Birth_Date: Date
Death_Date: Date
Employment_Start_Date: Date
Family_Aggregate: decimal(2)
Telephone_Number: varchar(20)
Locality_Name: varchar(28)
Birth_Country_Code: varchar(3)
Birth_Parish_Code: varchar(6)
S_CUSTOMER_CORPORATE
PK FK Customer_HK: varchar(64)
PK Load_Date: Datetime
Record_Source: varchar(50)
Load_End_Date: Datetime
Hash_Diff: varchar(64)
Establishment_Date: Date
Bankrupcy_Date: Date
Business_Entity_Type: decimal(6)
Number_Of_Employees: decimal(6)
Sales_Volume: decimal(15,2)
Capital_Amount: decimal(6,2)
Telephone_Number: varchar(20)
Fax_Number: varchar(20)
Telex_Number: varchar(11)
Currency_Code: varchar(3)
Company_Type: varchar(1)
Email_Code: varchar(64)
Balance_Currency_Code: varchar(3)
Registration_Code: varchar(15)
S_CUSTOMER_BASICS
PK FK Customer_HK: varchar(64)
PK Load_Date: Datetime
Record_Source: varchar(50)
Load_End_Date: Datetime
Hash_Diff: varchar(64)
Full_Name: varchar(70)
Reduced_Name: varchar(40)
Customer_mnemonic: varchar(20)
Branch_Code: varchar(3)
Customer_Manager_Code: varchar(5)
Property_Regime_Code: varchar(1)
Document_Type_Code: varchar(3)
Document_Number: decimal(10)
Country_Code: varchar(3)
Employee_Number: varchar(9)
Customer_Code: varchar(1)
Customer_Type: varchar(1)
H_CUSTOMER
PK Customer_HK: varchar(64)
Load_Date: Datetime
Record_Source: varchar(50)
Customer_Number: decimal(10)
L_CUSTOMER_RELATIONSHIP
PK Customer_Relationship_HK: varchar(64)
FK Customer_1_HK: varchar(64)
FK Customer_2_HK: varchar(64)
Relationship_Code: varchar(2)
Load_Date: Datetime
Record_Source: varchar(50)
S_CUSTOMER_RELATIONSHIP
PK FK Customer_Relationship_HK: varchar(64)
PK Load_Date: Datetime
Load_End_Date: Datetime
Hash_Diff: varchar(64)
Record_Source: varchar(50)
Participation_Percentage: decimal(3)
Relationship_Start_Date: Date
MAS_CURRENT_ACCOUNT_OWNERSHIP
PK FK Customer_Current_Account_HK: varchar(64)
PK Load_Date: Datetime
PK Ownership_Type: varchar(1)
Record_Source: varchar(50)
Load_End_Date: Datetime
Hash_Diff: varchar(64)
Ownership_Number: decimal(2)
L_CUSTOMER_CURRENT_ACCOUNT
PK Customer_Current_Account_HK: varchar(64)
FK Customer_HK: varchar(64)
FK Current_Account_HK: varchar(64)
Load_Date: Datetime
Record_Source: varchar(50)
S_CUSTOMER_CURRENT_ACCOUNT_INFO
PK FK Customer_Current_Account_HK: varchar(64)
PK Load_Date: Datetime
Record_Source: varchar(50)
Load_End_Date: Datetime
Hash_Diff: varchar(64)
Open_Date: Date
Cancel_Date: Date
Account_Handling_Code: varchar(2)
Signature_Reference_Code: varchar(1)
Ownership_Branch_Code: varchar(3)
Accounting_Account_Branch_Code: varchar(3)
S_CURRENT_ACCOUNT_BALANCE
PK FK Current_Account_HK: varchar(64)
PK Load_Date: Datetime
Record_Source: varchar(50)
Load_End_Date: Datetime
Hash_Diff: varchar(64)
Balance: decimal(15,2)
Available_Balance: decimal(15,2)
Unavailable_Balance: decimal(15,2)
Authorized_Hold_Balance: decimal(15,2)
Beggining_Of_Day_Balance: decimal(15,2)
Previous_Balance_Amount: decimal(15,2)
Maximum_Balance_Amount: decimal(15,2)
Minimum_Balance_Amount: decimal(15,2)
Last_Transaction_Date: Date
H_CURRENT_ACCOUNT
PK Current_Account_HK: varchar(64)
Load_Date: Datetime
Record_Source: varchar(50)
Account_Number: varchar(12)
Product Code: varchar(2)
Branch_Code: varchar(3)
Account_Number_Code: varchar(6)
Account_Check-digit: varchar(1)
S_CURRENT_ACCOUNT_CHECKS
PK FK Current_Account_HK: varchar(64)
PK Load_Date: Date time
Record_Source: varchar(50)
Load_End_Date: Datetime
Hash_Diff: varchar(64)
Number_Checks_Used: decimal(6)
Maximum_Amount_Active_Checks: decimal(6)
Amount_Active_Checks: decimal(6)
Maximum_Amount_Modules_Requested: decimal(1)
Maximum_Amount_Checks: decimal(6)
Amount_Checks_Requested: decimal(2)
S_CURRENT_ACCOUNT_INFO
PK FK Current_Account_HK: varchar(64)
PK Load_Date: Datetime
Record_Source: varchar(50)
Load_End_Date: Datetime
Hash_Diff: varchar(64)
Account_Opening_Date: Date
Account_Closing_Date: Date
Subproduct_Code: varchar(2)
Account_Manager_Code: varchar(5)
Currency_Code: varchar(3)
Account_Availability_Code: varchar(3)
Authorized_Overdraft_Tax: decimal(10,7)
Contract_Commencement_Date: decimal(10)
Account_Owner_Name: varchar(40)

Figure 14: Full Data Vault model of the bank's customers and current accounts

The following subsections each describe a segment of the presented model. They begin with an overview of the segment's specific business logic, progressing into an exploration of the distinct modelling decisions taken to ensure an effective representation. This includes detailing the challenges encountered, evaluating alternative approaches considered for modelling, and correlating the selected modelling decision with the hubs, links, and satellite tables of the proposed model.

It is worth noting that the satellite tables in the model do not encapsulate all descriptive information of a respective entity held by the bank. The model does not aim to represent every individual attribute, as the main goal of this research is to accurately depict the organization's data and business logic concerning the types of entities modelled and their interrelations.

4.3.1. Customers

In the context of the organization, customers can be categorized into two types: individual customers and corporations. While there is a subset of descriptive information shared by both types of customers, certain attributes are exclusively linked to either individuals or corporations. To illustrate, the attribute ”Sales Volume” is specifically associated with corporate customers, whereas ”Family aggregate” is a feature unique to individual customers. An attribute like ”Country Code” is a shared feature, common to both customer types.

In terms of interactions with the bank’s products, the behaviour exhibited by both customer types is essentially the same. For instance, any form of association with a Current Account is facilitated through a unique customer number. Irrespective of whether the customer is a corporation or an individual, upon registering an account with the bank, a unique customer number is assigned.

The challenge then was to adequately model this business logic. Two alternatives were evaluated. The first involved modelling individual and corporate customers as separate hubs. However, this approach would result in duplications of links with other Hubs. Although this redundancy might be negligible when considering a single-product relationship, it becomes considerably complex and could lead to performance issues given the bank’s extensive product portfolio.

The adopted solution, therefore, was to incorporate both corporate and individual customers within the same hub. This model design includes two separate satellites for storing exclusive descriptive information: the S_CUSTOMER_CORPORATE for corporations and S_CUSTOMER_INDIVIDUAL for individuals. A third satellite, S_CUSTOMER_BASICS, is utilized to store descriptive information that pertains to both customer types. This approach is visually represented in Figure 15.

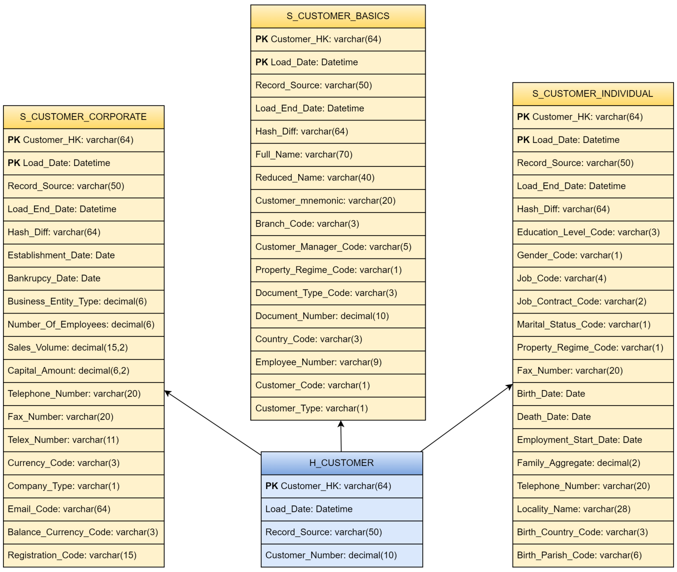

S_CUSTOMER_BASICS
PK Customer_HK: varchar(64)
PK Load_Date: Datetime
Record_Source: varchar(50)
Load_End_Date: Datetime
Hash_Diff: varchar(64)
Full_Name: varchar(70)
Reduced_Name: varchar(40)
Customer_mnemonic: varchar(20)
Branch_Code: varchar(3)
Customer_Manager_Code: varchar(5)
Property_Regime_Code: varchar(1)
Document_Type_Code: varchar(3)
Document_Number: decimal(10)
Country_Code: varchar(3)
Employee_Number: varchar(9)
Customer_Code: varchar(1)
Customer_Type: varchar(1)
S_CUSTOMER_CORPORATE
PK Customer_HK: varchar(64)
PK Load_Date: Datetime
Record_Source: varchar(50)
Load_End_Date: Datetime
Hash_Diff: varchar(64)
Establishment_Date: Date
Bankrupcy_Date: Date
Business_Entity_Type: decimal(6)
Number_Of_Employees: decimal(6)
Sales_Volume: decimal(15,2)
Capital_Amount: decimal(6,2)
Telephone_Number: varchar(20)
Fax_Number: varchar(20)
Telex_Number: varchar(11)
Currency_Code: varchar(3)
Company_Type: varchar(1)
Email_Code: varchar(64)
Balance_Currency_Code: varchar(3)
Registration_Code: varchar(15)
S_CUSTOMER_INDIVIDUAL
PK Customer_HK: varchar(64)
PK Load_Date: Datetime
Record_Source: varchar(50)
Load_End_Date: Datetime
Hash_Diff: varchar(64)
Education_Level_Code: varchar(3)
Gender_Code: varchar(1)
Job_Code: varchar(4)
Job_Contract_Code: varchar(2)
Marital_Status_Code: varchar(1)
Property_Regime_Code: varchar(1)
Fax_Number: varchar(20)
Birth_Date: Date
Death_Date: Date
Employment_Start_Date: Date
Family_Aggregate: decimal(2)
Telephone_Number: varchar(20)
Locality_Name: varchar(28)
Birth_Country_Code: varchar(3)
Birth_Parish_Code: varchar(6)
H_CUSTOMER
PK Customer_HK: varchar(64)
Load_Date: Datetime
Record_Source: varchar(50)
Customer_Number: decimal(10)

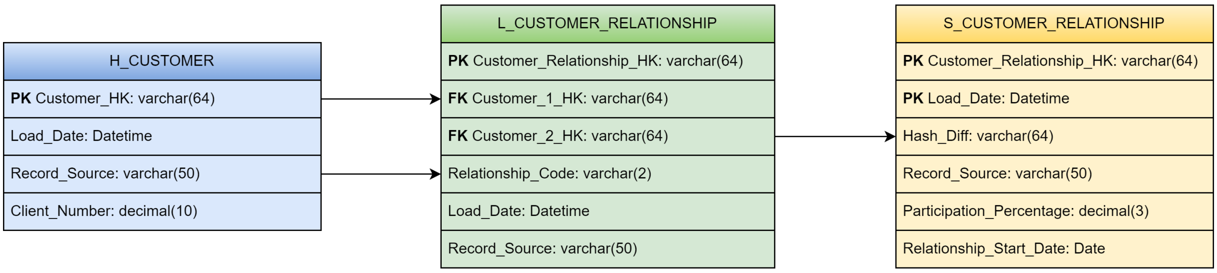
Figure 15: Modelling of the Customer business concept

4.3.2. Relationships between customers

The section of the Data Vault model related to relationships between customers is unique in the sense that it does not represent a relationship between two different business concepts. Rather, it depicts the interrelationships among customers. According to the bank's business logic, each customer can have multiple relationships with other customers. Furthermore, two customers can maintain various types of relationships simultaneously. For instance, Customer 1 might be both the tutor and the manager of Customer 2.

To accurately reflect this business logic in the model, a link table is used, which connects two customer hubs. If we merely used the hash keys from the customer hubs as foreign keys in the link table, it would prevent two customers from having multiple relationships, as the hash keys generated wouldn't be unique for each relationship type. To circumvent this issue, we introduce a variable named ”Relationship_Code”, representing the type of relationship, as a dependent child key.

A dependent child key is a field that, unlike hubs, cannot stand alone, as it possesses no inherent business meaning and requires a specific context to be valid. The purpose of this key is to determine the granularity and uniqueness of the data set in the link table. Consequently, this dependent child key becomes a defining element of the link structure, as the hash key is derived from not just the business keys of the referenced hubs, but their combination with the dependent child key (Linstedt and Olschimke, 2015, p. 132). The approach used to reflect the relationships between customers is visually represented in Figure 16.


H_CUSTOMER
PK Customer_HK: varchar(64)
Load_Date: Datetime
Record_Source: varchar(50)
Client_Number: decimal(10)

L_CUSTOMER_RELATIONSHIP
PK Customer_Relationship_HK: varchar(64)
FK Customer_1_HK: varchar(64)
FK Customer_2_HK: varchar(64)
Relationship_Code: varchar(2)
Load_Date: Datetime
Record_Source: varchar(50)

S_CUSTOMER_RELATIONSHIP
PK Customer_Relationship_HK: varchar(64)
PK Load_Date: Datetime
Hash_Diff: varchar(64)
Record_Source: varchar(50)
Participation_Percentage: decimal(3)
Relationship_Start_Date: Date

Figure 16: Modelling of the relationships between the bank's customers

4.3.3. Current Accounts

According to the business logic, the identifier for current accounts isn't represented by a singular value, but rather by an amalgamation of multiple characteristics. It adheres to the following structure: PP.BBB.AAAAAA.C, where each component stands for the Product code, Branch code, Account Number code, and account check digit, respectively.

This representation is realized in Data Vault modelling through the use of a Composite Key. The Current Account Hub table encapsulates multiple business keys, the combination of which is hashed. The order of the business keys in the table is significant as it represents the order of the account number. (Linstedt and Olschimke, 2015, p. 115). Furthermore, the account number has its standalone representation as a field in the hub. This is recommended because the composite business key, as a singular field, should also be incorporated within the hub if it carries business significance to the organization. (Linstedt and Olschimke, 2015, p. 119)

Concerning the descriptive information, it is advised to distribute data among multiple satellites according to the rate at which the data within each satellite changes (Linstedt and Olschimke, 2015, p. 137). Data pertaining to the account balance, such as Available Balance and Last Transaction Date, can be updated several times per day. On the other hand, attributes like Account Opening Date rarely, if ever, change. Data related to checks occupies a middle ground; it is subject to frequent changes, albeit not as often as the balance information. Figure 17 illustrates the Data Vault modelling of the bank's current accounts.

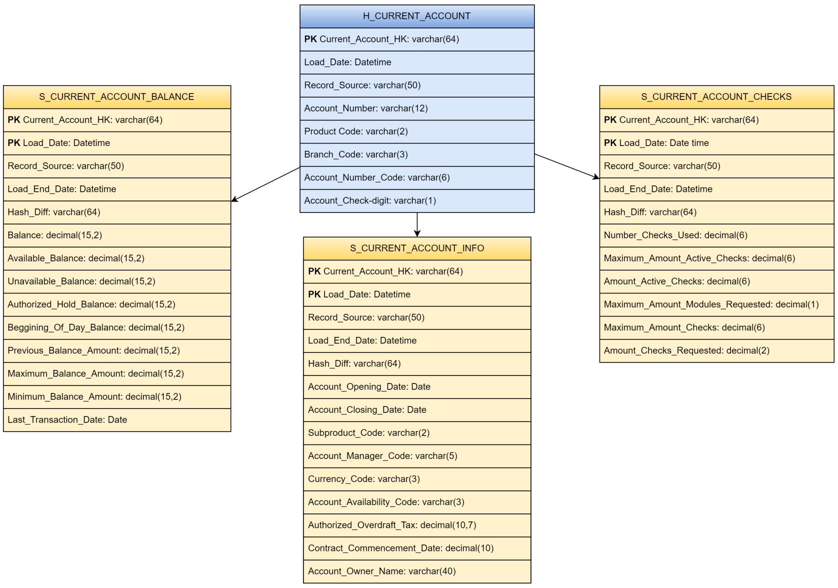

H_CURRENT_ACCOUNT
PK Current_Account_HK: varchar(64)
Load_Date: Datetime
Record_Source: varchar(50)
Account_Number: varchar(12)
Product Code: varchar(2)
Branch_Code: varchar(3)
Account_Number_Code: varchar(6)
Account_Check-digit: varchar(1)
S_CURRENT_ACCOUNT_BALANCE
PK Current_Account_HK: varchar(64)
PK Load_Date: Datetime
Record_Source: varchar(50)
Load_End_Date: Datetime
Hash_Diff: varchar(64)
Balance: decimal(15,2)
Available_Balance: decimal(15,2)
Unavailable_Balance: decimal(15,2)
Authorized_Hold_Balance: decimal(15,2)
Beggining_Of_Day_Balance: decimal(15,2)
Previous_Balance_Amount: decimal(15,2)
Maximum_Balance_Amount: decimal(15,2)
Minimum_Balance_Amount: decimal(15,2)
Last_Transaction_Date: Date
S_CURRENT_ACCOUNT_CHECKS
PK Current_Account_HK: varchar(64)
PK Load_Date: Date time
Record_Source: varchar(50)
Load_End_Date: Datetime
Hash_Diff: varchar(64)
Number_Checks_Used: decimal(6)
Maximum_Amount_Active_Checks: decimal(6)
Amount_Active_Checks: decimal(6)
Maximum_Amount_Modules_Requested: decimal(1)
Maximum_Amount_Checks: decimal(6)
Amount_Checks_Requested: decimal(2)
S_CURRENT_ACCOUNT_INFO
PK Current_Account_HK: varchar(64)
PK Load_Date: Datetime
Record_Source: varchar(50)
Load_End_Date: Datetime
Hash_Diff: varchar(64)
Account_Opening_Date: Date
Account_Closing_Date: Date
Subproduct_Code: varchar(2)
Account_Manager_Code: varchar(5)
Currency_Code: varchar(3)
Account_Availability_Code: varchar(3)
Authorized_Overdraft_Tax: decimal(10,7)
Contract_Commencement_Date: decimal(10)
Account_Owner_Name: varchar(40)

Figure 17: Modelling of the Current Account business concept

4.3.4. Customers and Current Accounts relationships

In the organization's business logic, one customer might have multiple current accounts, and vice versa. This structure implies a diverse set of ownership relationships between a customer and a current account. For instance, a customer could be an ”Account Holder” with complete control over the account or an ”Authorized User” who has been granted access by the Account Holder. Interestingly, a single customer might play different roles simultaneously, such as being both an ”Authorized User” and a ”Trustee”, who holds the account in trust for someone else.

To address this complex behaviour, we initially considered adding an ”Ownership Type” dependent child key in the link table ”L_CUSTOMER_CURRENT_ACCOUNT”. However, this approach would change the granularity of the relationship, linking the customer hub based on the number of ownership types it holds (Linstedt and Olschimke, 2015, p. 132). This strategy risks duplicating data as the customer-current account relationship carries descriptive data that exists irrespective of the ownership type.

The adopted solution was to use a standard link between the ”Customer” and ”Current Account” hubs and introduce a ”Multi-Active Satellite” that contains information about a customer's account ownership. This new satellite, titled ”MAS_CURRENT_ACCOUNT_OWNERSHIP”, allows multiple concurrent active instances, accommodating a customer's potential numerous ownership types. (Linstedt and Olschimke, 2015, p. 163)

This solution preserves the desired granularity of the relationship while also enabling the creation of a separate standard satellite. This satellite contains descriptive information about the relationship between a customer and a current account, regardless of ownership type. As a result, the descriptive information of a customer with two different ownership types would be represented by two active ”MAS_CURRENT_ACCOUNT_OWNERSHIP” satellites and one ”S_CUSTOMER_CURRENT_ACCOUNT_INFO” satellite. The adopted approach is visually represented in Figure **??**.

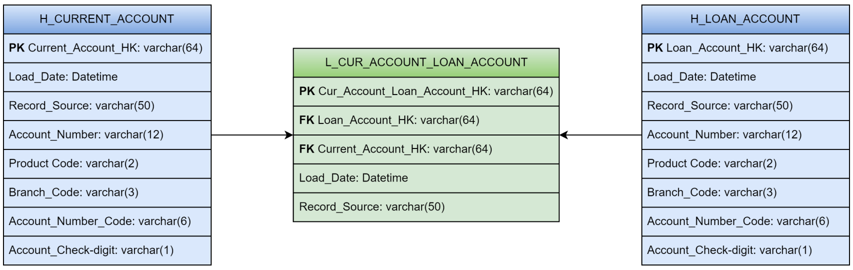

H_CURRENT_ACCOUNT
PK Current_Account_HK: varchar(64)
Load_Date: Datetime
Record_Source: varchar(50)
Account_Number: varchar(12)
Product Code: varchar(2)
Branch_Code: varchar(3)
Account_Number_Code: varchar(6)
Account_Check-digit: varchar(1)
L_CUR_ACCOUNT_LOAN_ACCOUNT
PK Cur_Account_Loan_Account_HK: varchar(64)
FK Loan_Account_HK: varchar(64)
FK Current_Account_HK: varchar(64)
Load_Date: Datetime
Record_Source: varchar(50)
H_LOAN_ACCOUNT
PK Loan_Account_HK: varchar(64)
Load_Date: Datetime
Record_Source: varchar(50)
Account_Number: varchar(12)
Product Code: varchar(2)
Branch_Code: varchar(3)
Account_Number_Code: varchar(6)
Account_Check-digit: varchar(1)

Figure 18: Modelling of the relationship between current and loan accounts, with updated business logic

5. DEMONSTRATION

In the demonstration step of DSR, the ability of the created artefact to solve relevant problems is demonstrated (Peffers et al., 2007). In this case, our artefact is a data vault model that reflects a specific aspect of the bank's data and operations: customers and current accounts. The fundamental real-world problem it aims to solve is effectively characterizing this dimension, and also to efficiently adapt to the bank's rapidly evolving business requirements.

In this section, we will first introduce a series of new business requirements that our model will be subjected to. Following this, we will present the fully adapted model and summarize the findings from the individual changes.

5.1. ADAPTING THE MODEL TO NEW BUSINESS REQUIREMENTS

In each of the following subsections, we will illustrate how our model can efficiently adapt to new business requirements. First, we will provide an overview of the new business requirements; secondly, we will detail the modelling decisions made to adapt to these changes. The business requirements depicted here not only represent real-world scenarios encountered within the organization but also represent common methods of modifying an existing Data Vault model.

5.1.1. Customers loaded from a different source

The organization has decided to incorporate customer data from an additional internal data source. Customers in this new system are already registered in the principal source system. However, a key distinction exists in the unique identification of customers in these two sources. The current model identifies customers using a ten-digit numeric customer number. In contrast, the new data source uses a distinct nine-digit customer number.

Given that the business key of the Customer hub is the leading system's customer number, a challenging question arises: how can we integrate data from a new source into the Data Vault if its business key differs from the hub's business key? The answer lies in Data Vault 2.0's ”same-as link” concept. A same-as link connects to the hub and houses two separate hash keys—one for the original source system, and another for the new source system. To correlate each customer number across both systems, a mapping table is utilized. (Linstedt and Olschimke, 2015, pp. 147, 151)

By leveraging the same-as link table, one can look up a customer number from the new source system by selecting the record with a matching master hash key. Specifically, the record with the chosen ”Customer_HK” value in the same-as link table is selected. This record will have a ”H_Master_Customer_HK” value matching the search value, while the new system's customer number will correspond to the ”H_Duplicate_Customer_HK” field. The updated model, in comparison with its previous version, is illustrated in Figure 19.

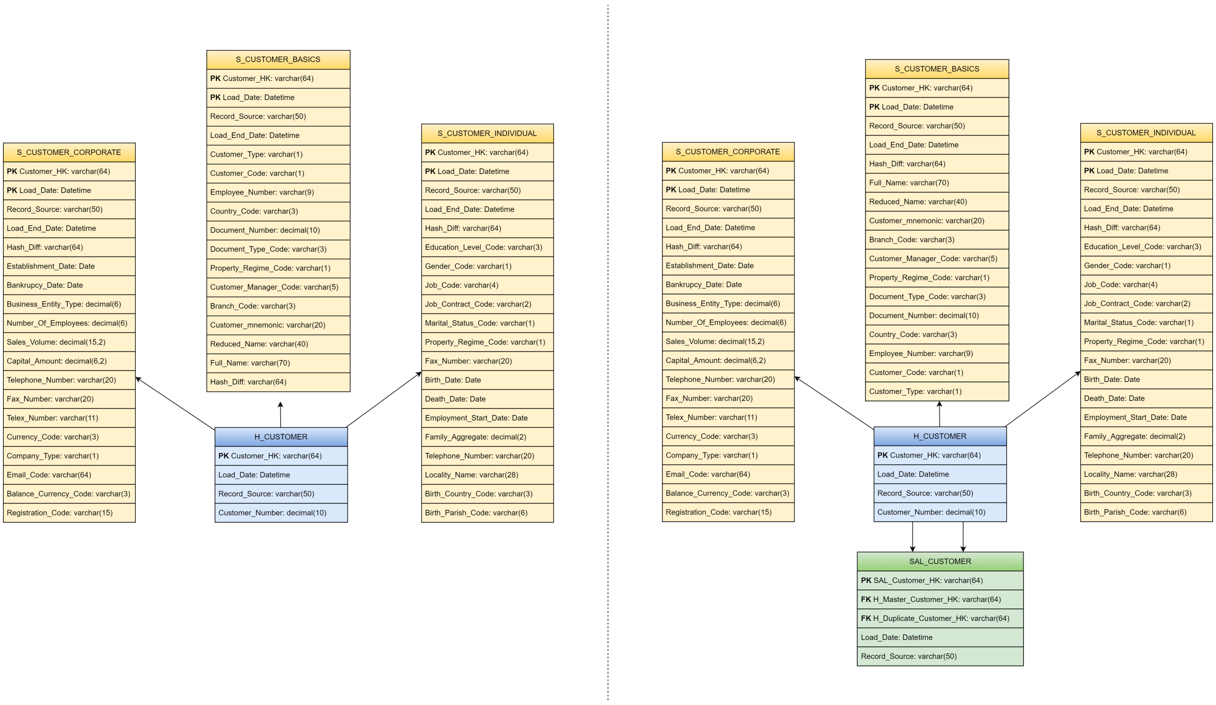

S_CUSTOMER_CORPORATE
PK Customer_HK: varchar(64)
PK Load_Date: Datetime
Record_Source: varchar(50)
Load_End_Date: Datetime
Hash_Diff: varchar(64)
Establishment_Date: Date
Bankrupcy_Date: Date
Business_Entity_Type: decimal(6)
Number_Of_Employees: decimal(6)
Sales_Volume: decimal(15,2)
Capital_Amount: decimal(6,2)
Telephone_Number: varchar(20)
Fax_Number: varchar(20)
Telex_Number: varchar(11)
Currency_Code: varchar(3)
Company_Type: varchar(1)
Email_Code: varchar(64)
Balance_Currency_Code: varchar(3)
Registration_Code: varchar(15)
S_CUSTOMER_BASICS
PK Customer_HK: varchar(64)
PK Load_Date: Datetime
Record_Source: varchar(50)
Load_End_Date: Datetime
Customer_Type: varchar(1)
Customer_Code: varchar(1)
Employee_Number: varchar(9)
Country_Code: varchar(3)
Document_Number: decimal(10)
Document_Type_Code: varchar(3)
Property_Regime_Code: varchar(1)
Customer_Manager_Code: varchar(5)
Branch_Code: varchar(3)
Customer_mnemonic: varchar(20)
Reduced_Name: varchar(40)
Full_Name: varchar(70)
Hash_Diff: varchar(64)
H_CUSTOMER
PK Customer_HK: varchar(64)
Load_Date: Datetime
Record_Source: varchar(50)
Customer_Number: decimal(10)
S_CUSTOMER_INDIVIDUAL
PK Customer_HK: varchar(64)
PK Load_Date: Datetime
Record_Source: varchar(50)
Load_End_Date: Datetime
Hash_Diff: varchar(64)
Education_Level_Code: varchar(3)
Gender_Code: varchar(1)
Job_Code: varchar(4)
Job_Contract_Code: varchar(2)
Marital_Status_Code: varchar(1)
Property_Regime_Code: varchar(1)
Fax_Number: varchar(20)
Birth_Date: Date
Death_Date: Date
Employment_Start_Date: Date
Family_Aggregate: decimal(2)
Telephone_Number: varchar(20)
Locality_Name: varchar(28)
Birth_Country_Code: varchar(3)
Birth_Parish_Code: varchar(6)
S_CUSTOMER_CORPORATE
PK Customer_HK: varchar(64)
PK Load_Date: Datetime
Record_Source: varchar(50)
Load_End_Date: Datetime
Hash_Diff: varchar(64)
Establishment_Date: Date
Bankrupcy_Date: Date
Business_Entity_Type: decimal(6)
Number_Of_Employees: decimal(6)
Sales_Volume: decimal(15,2)
Capital_Amount: decimal(6,2)
Telephone_Number: varchar(20)
Fax_Number: varchar(20)
Telex_Number: varchar(11)
Currency_Code: varchar(3)
Company_Type: varchar(1)
Email_Code: varchar(64)
Balance_Currency_Code: varchar(3)
Registration_Code: varchar(15)
S_CUSTOMER_BASICS
PK Customer_HK: varchar(64)
PK Load_Date: Datetime
Record_Source: varchar(50)
Load_End_Date: Datetime
Hash_Diff: varchar(64)
Full_Name: varchar(70)
Reduced_Name: varchar(40)
Customer_mnemonic: varchar(20)
Branch_Code: varchar(3)
Customer_Manager_Code: varchar(5)
Property_Regime_Code: varchar(1)
Document_Type_Code: varchar(3)
Document_Number: decimal(10)
Country_Code: varchar(3)
Employee_Number: varchar(9)
Customer_Code: varchar(1)
Customer_Type: varchar(1)
H_CUSTOMER
PK Customer_HK: varchar(64)
Load_Date: Datetime
Record_Source: varchar(50)
Customer_Number: decimal(10)
S_CUSTOMER_INDIVIDUAL
PK Customer_HK: varchar(64)
PK Load_Date: Datetime
Record_Source: varchar(50)
Load_End_Date: Datetime
Hash_Diff: varchar(64)
Education_Level_Code: varchar(3)
Gender_Code: varchar(1)
Job_Code: varchar(4)
Job_Contract_Code: varchar(2)
Marital_Status_Code: varchar(1)
Property_Regime_Code: varchar(1)
Fax_Number: varchar(20)
Birth_Date: Date
Death_Date: Date
Employment_Start_Date: Date
Family_Aggregate: decimal(2)
Telephone_Number: varchar(20)
Locality_Name: varchar(28)
Birth_Country_Code: varchar(3)
Birth_Parish_Code: varchar(6)
SAL_CUSTOMER
PK SAL_Customer_HK: varchar(64)
FK H_Master_Customer_HK: varchar(64)
FK H_Duplicate_Customer_HK: varchar(64)
Load_Date: Datetime
Record_Source: varchar(50)

Figure 19: Modification of the customer section to accommodate an additional data source. Pre-adaptation model (left) vs. post-adaptation model (right)

5.1.2. Addition of a loan account

The bank intends to introduce a new account type to its data vault model, namely, a loan account. A given current account can be associated with multiple loan accounts; however, each loan account links back to just a single current account. Funds the user uses to pay off the loan account are withdrawn from its linked current account.

The relationship between loan accounts and customers mirrors that of customers and current accounts. Each loan account can be associated with multiple customers, who can, in turn, hold various types of ownership concurrently. The business logic asserts that the customer-account relationship, irrespective of the account type, retains the same descriptive data.

Being a business object, the loan account is modelled as a hub, accompanied by a satellite to store its descriptive data. The link table that connects the customer hub and the loan account hub parallels the one connecting current accounts and customers, the only difference being the substitution of the current account hash key with the loan account hash key. The satellites contain the same descriptive data.

As for the connection between current and loan accounts, an additional link table, with no satellites, is introduced, also known as a ”Nondescriptive Link” (Linstedt and Olschimke, 2015, p. 158). This is due to the absence of descriptive data related to the linking of the accounts. The loan account and its relationships with existing business objects are displayed in Figure 20.

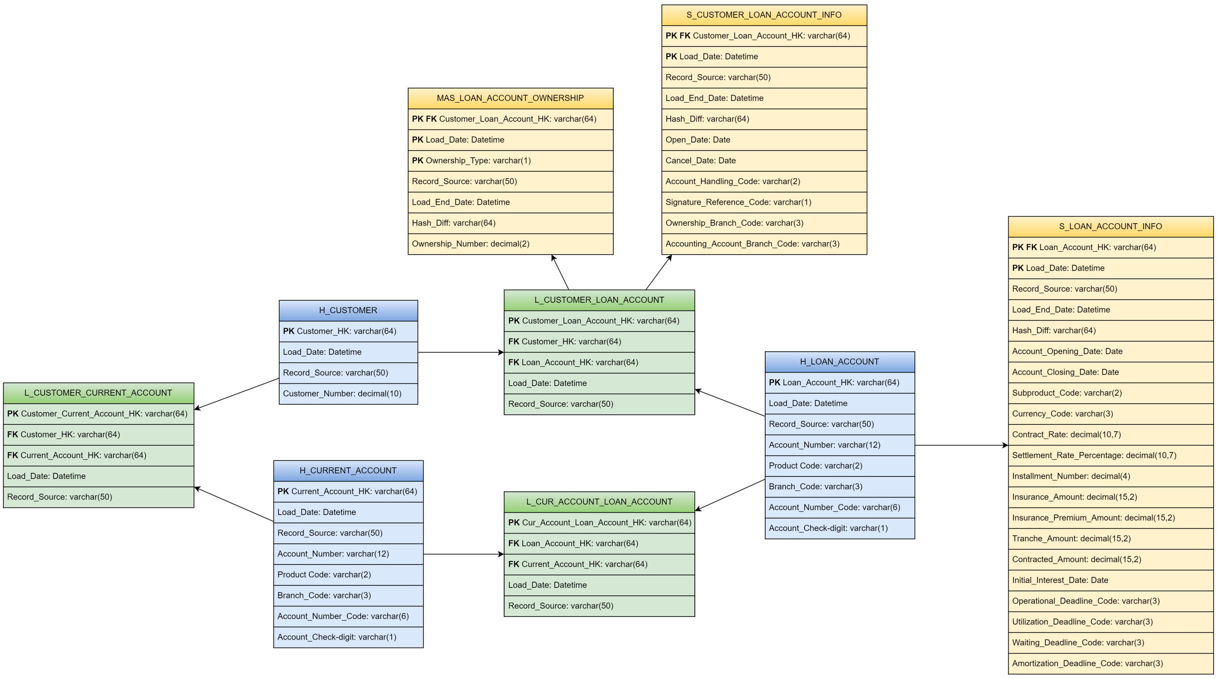

S_CUSTOMER_LOAN_ACCOUNT_INFO
PK FK Customer_Loan_Account_HK: varchar(64)
PK Load_Date: Datetime
Record_Source: varchar(50)
Load_End_Date: Datetime
Hash_Diff: varchar(64)
Open_Date: Date
Cancel_Date: Date
Account_Handling_Code: varchar(2)
Signature_Reference_Code: varchar(1)
Ownership_Branch_Code: varchar(3)
Accounting_Account_Branch_Code: varchar(3)
MAS_LOAN_ACCOUNT_OWNERSHIP
PK FK Customer_Loan_Account_HK: varchar(64)
PK Load_Date: Datetime
PK Ownership_Type: varchar(1)
Record_Source: varchar(50)
Load_End_Date: Datetime
Hash_Diff: varchar(64)
Ownership_Number: decimal(2)
S_LOAN_ACCOUNT_INFO
PK FK Loan_Account_HK: varchar(64)
PK Load_Date: Datetime
Record_Source: varchar(50)
Load_End_Date: Datetime
Hash_Diff: varchar(64)
Account_Opening_Date: Date
Account_Closing_Date: Date
Subproduct_Code: varchar(2)
Currency_Code: varchar(3)
Contract_Rate: decimal(10,7)
Settlement_Rate_Percentage: decimal(10,7)
Installment_Number: decimal(4)
Insurance_Amount: decimal(15,2)
Insurance_Premium_Amount: decimal(15,2)
Tranche_Amount: decimal(15,2)
Contracted_Amount: decimal(15,2)
Initial_Interest_Date: Date
Operational_Deadline_Code: varchar(3)
Utilization_Deadline_Code: varchar(3)
Waiting_Deadline_Code: varchar(3)
Amortization_Deadline_Code: varchar(3)
H_CUSTOMER
PK Customer_HK: varchar(64)
Load_Date: Datetime
Record_Source: varchar(50)
Customer_Number: decimal(10)
L_CUSTOMER_LOAN_ACCOUNT
PK Customer_Loan_Account_HK: varchar(64)
FK Customer_HK: varchar(64)
FK Loan_Account_HK: varchar(64)
Load_Date: Datetime
Record_Source: varchar(50)
H_LOAN_ACCOUNT
PK Loan_Account_HK: varchar(64)
Load_Date: Datetime
Record_Source: varchar(50)
Account_Number: varchar(12)
Product Code: varchar(2)
Branch_Code: varchar(3)
Account_Number_Code: varchar(6)
Account_Check-digit: varchar(1)
L_CUSTOMER_CURRENT_ACCOUNT
PK Customer_Current_Account_HK: varchar(64)
FK Customer_HK: varchar(64)
FK Current_Account_HK: varchar(64)
Load_Date: Datetime
Record_Source: varchar(50)
H_CURRENT_ACCOUNT
PK Current_Account_HK: varchar(64)
Load_Date: Datetime
Record_Source: varchar(50)
Account_Number: varchar(12)
Product Code: varchar(2)
Branch_Code: varchar(3)
Account_Number_Code: varchar(6)
Account_Check-digit: varchar(1)
L_CUR_ACCOUNT_LOAN_ACCOUNT
PK Cur_Account_Loan_Account_HK: varchar(64)
FK Loan_Account_HK: varchar(64)
FK Current_Account_HK: varchar(64)
Load_Date: Datetime
Record_Source: varchar(50)

Figure 20: Addition of a loan account and its relationships with existing business objects

5.1.3. New attributes added to loan account

In the current context, the organization is analyzing the adoption of Environmental, Social, and Governance (ESG) policies. One notable concept in this realm is ESG-linked loans. These are general-purpose loans where pricing terms are associated with the ESG performance of the borrowing entity (Kim et al., 2022). The loan spreads of these loans are directly linked to key performance indicators (KPIs) related to sustainability goals (Kim et al., 2022). These KPIs may take the form of ESG scores, often provided by external rating agencies. Alternatively, they could represent specific metrics such as the greenhouse gas emissions of the borrowing entity or employee welfare score. (Kim et al., 2022)

This additional information is easily added to the Data Vault model, by adding a new satellite connected to the loan account hub, with the ESG-related attributes (Linstedt and Olschimke, 2015, p. 135). It is crucial to mention that these attributes are hypothetical and do not represent actual attributes employed by the organization, considering that the relevant policies have not been implemented yet. Altering the existing structure is unnecessary and could even introduce challenges. It could complicate tracking changes and create inconsistency, as new satellite entries would include additional fields. Additionally, redefining the attributes used to calculate the hash diff value would be necessary. (Linstedt and Olschimke, 2015, p. 139)

The updated model of the loan accounts, with the added descriptive data, is depicted in Figure 21.

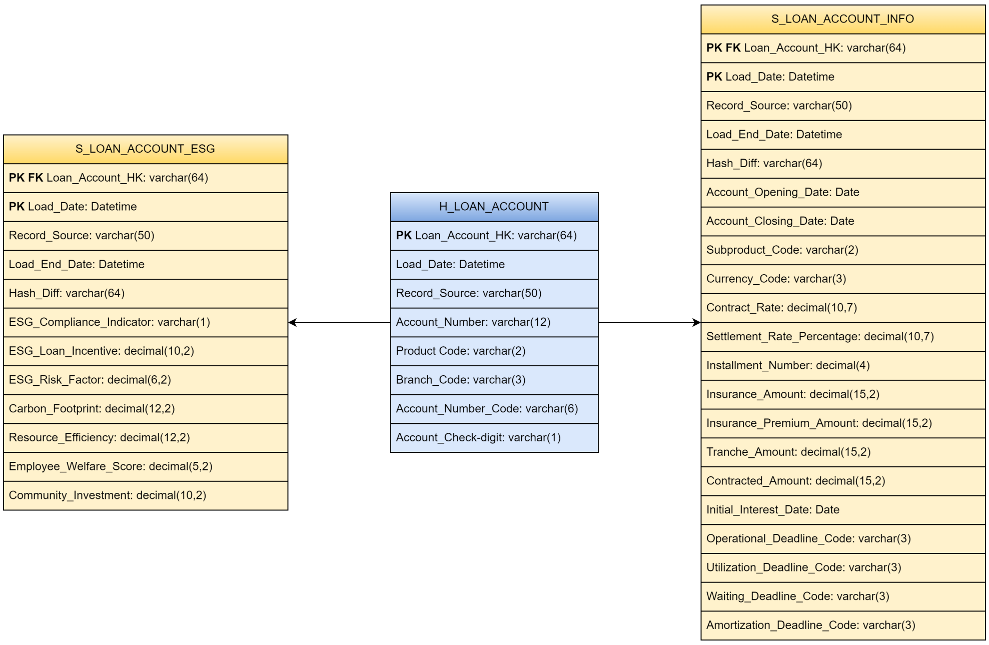

S_LOAN_ACCOUNT_ESG
PK FK Loan_Account_HK: varchar(64)
PK Load_Date: Datetime
Record_Source: varchar(50)
Load_End_Date: Datetime
Hash_Diff: varchar(64)
ESG_Compliance_Indicator: varchar(1)
ESG_Loan_Incentive: decimal(10,2)
ESG_Risk_Factor: decimal(6,2)
Carbon_Footprint: decimal(12,2)
Resource_Efficiency: decimal(12,2)
Employee_Welfare_Score: decimal(5,2)
Community_Investment: decimal(10,2)
H_LOAN_ACCOUNT
PK Loan_Account_HK: varchar(64)
Load_Date: Datetime
Record_Source: varchar(50)
Account_Number: varchar(12)
Product Code: varchar(2)
Branch_Code: varchar(3)
Account_Number_Code: varchar(6)
Account_Check-digit: varchar(1)
S_LOAN_ACCOUNT_INFO
PK FK Loan_Account_HK: varchar(64)
PK Load_Date: Datetime
Record_Source: varchar(50)
Load_End_Date: Datetime
Hash_Diff: varchar(64)
Account_Opening_Date: Date
Account_Closing_Date: Date
Subproduct_Code: varchar(2)
Currency_Code: varchar(3)
Contract_Rate: decimal(10,7)
Settlement_Rate_Percentage: decimal(10,7)
Installment_Number: decimal(4)
Insurance_Amount: decimal(15,2)
Insurance_Premium_Amount: decimal(15,2)
Tranche_Amount: decimal(15,2)
Contracted_Amount: decimal(15,2)
Initial_Interest_Date: Date
Operational_Deadline_Code: varchar(3)
Utilization_Deadline_Code: varchar(3)
Waiting_Deadline_Code: varchar(3)
Amortization_Deadline_Code: varchar(3)

Figure 21: Modelling of the updated loan account, with additional attributes

5.1.4. Adaption to new business logic

Currently, the business logic of the organization stipulates that a loan account is associated with a singular current account, while a current account may be linked to multiple loan accounts. This constitutes a one-to-many relationship. If the organization were to modify this structure to allow a loan account to be connected with multiple current accounts, the relationship would transition to a many-to-many type.

This transformation would have no impact on the representation in the Data Vault Model. All relationships within the Data Vault modelling paradigm are inherently considered to be many-to-many (Linstedt and Olschimke, 2015, p. 12). Therefore, no structural alterations would be required, and the existing model would effectively accommodate this revised business logic.

Figure 22 depicts the section of the Data Vault model representing the relationship between loan accounts and current accounts, under the revised business logic.

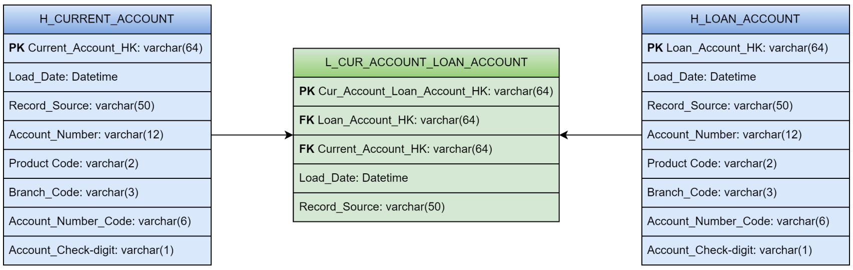

H_CURRENT_ACCOUNT
PK Current_Account_HK: varchar(64)
Load_Date: Datetime
Record_Source: varchar(50)
Account_Number: varchar(12)
Product Code: varchar(2)
Branch_Code: varchar(3)
Account_Number_Code: varchar(6)
Account_Check-digit: varchar(1)
L_CUR_ACCOUNT_LOAN_ACCOUNT
PK Cur_Account_Loan_Account_HK: varchar(64)
FK Loan_Account_HK: varchar(64)
FK Current_Account_HK: varchar(64)
Load_Date: Datetime
Record_Source: varchar(50)
H_LOAN_ACCOUNT
PK Loan_Account_HK: varchar(64)
Load_Date: Datetime
Record_Source: varchar(50)
Account_Number: varchar(12)
Product Code: varchar(2)
Branch_Code: varchar(3)
Account_Number_Code: varchar(6)
Account_Check-digit: varchar(1)

Figure 22: Modelling the relationship between current and loan accounts, with updated business logic

5.2. FULLY ADAPTED DATA VAULT MODEL

A finalized version of the model, with all the changes described in the preceding subsections included, is depicted in Figure 23.

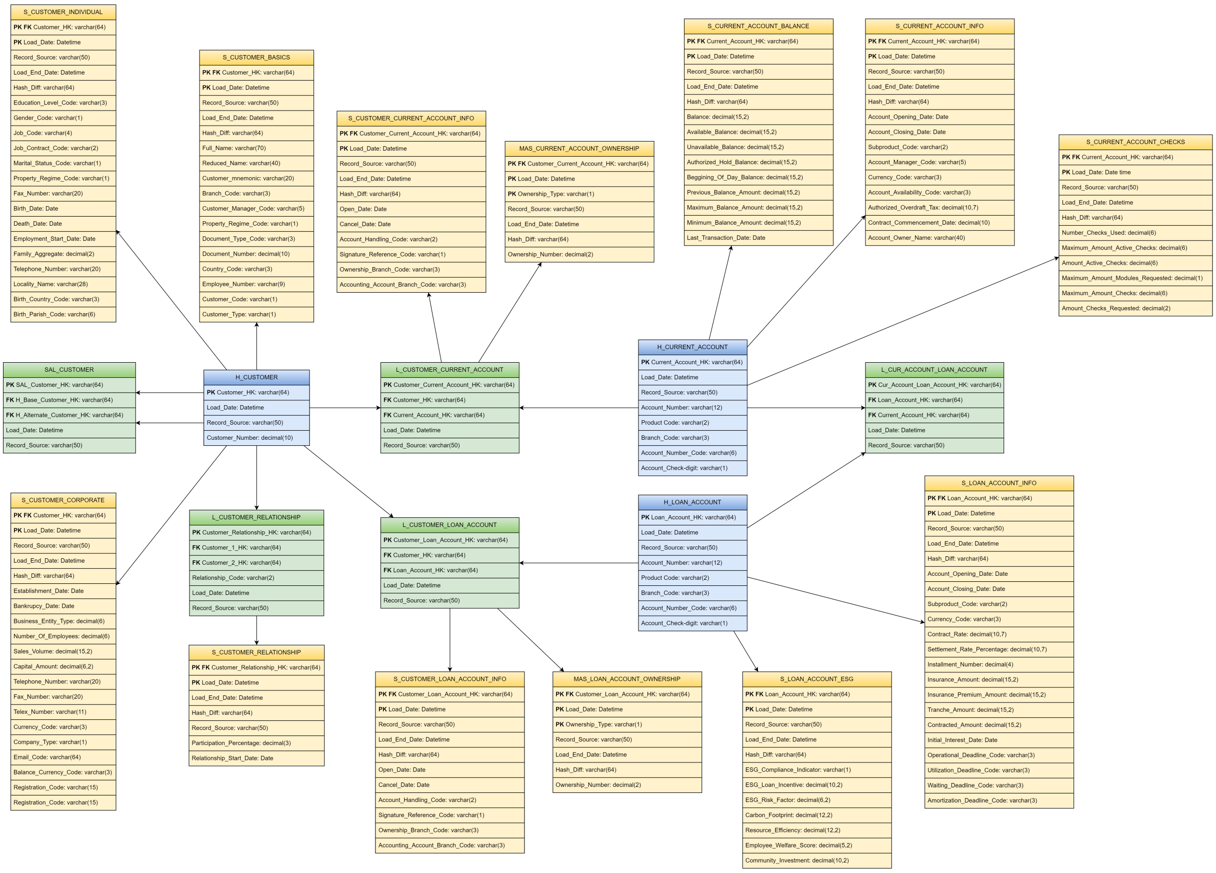

S_CUSTOMER_INDIVIDUAL
PK FK Customer_HK: varchar(64)
PK Load_Date: Datetime
Record_Source: varchar(50)
Load_End_Date: Datetime
Hash_Diff: varchar(64)
Education_Level_Code: varchar(3)
Gender_Code: varchar(1)
Job_Code: varchar(4)
Job_Contract_Code: varchar(2)
Marital_Status_Code: varchar(1)
Property_Regime_Code: varchar(1)
Fax_Number: varchar(20)
Birth_Date: Date
Death_Date: Date
Employment_Start_Date: Date
Family_Aggregate: decimal(2)
Telephone_Number: varchar(20)
Locality_Name: varchar(28)
Birth_Country_Code: varchar(3)
Birth_Parish_Code: varchar(6)
S_CUSTOMER_BASICS
PK FK Customer_HK: varchar(64)
PK Load_Date: Datetime
Record_Source: varchar(50)
Load_End_Date: Datetime
Hash_Diff: varchar(64)
Full_Name: varchar(70)
Reduced_Name: varchar(40)
Customer_mnemonic: varchar(20)
Branch_Code: varchar(3)
Customer_Manager_Code: varchar(5)
Property_Regime_Code: varchar(1)
Document_Type_Code: varchar(3)
Document_Number: decimal(10)
Country_Code: varchar(3)
Employee_Number: varchar(9)
Customer_Code: varchar(1)
Customer_Type: varchar(1)
S_CUSTOMER_CURRENT_ACCOUNT_INFO
PK FK Customer_Current_Account_HK: varchar(64)
PK Load_Date: Datetime
Record_Source: varchar(50)
Load_End_Date: Datetime
Hash_Diff: varchar(64)
Open_Date: Date
Cancel_Date: Date
Account_Handling_Code: varchar(2)
Signature_Reference_Code: varchar(1)
Ownership_Branch_Code: varchar(3)
Accounting_Account_Branch_Code: varchar(3)
MAS_CURRENT_ACCOUNT_OWNERSHIP
PK FK Customer_Current_Account_HK: varchar(64)
PK Load_Date: Datetime
PK Ownership_Type: varchar(1)
Record_Source: varchar(50)
Load_End_Date: Datetime
Hash_Diff: varchar(64)
Ownership_Number: decimal(2)
S_CURRENT_ACCOUNT_BALANCE
PK FK Current_Account_HK: varchar(64)
PK Load_Date: Datetime
Record_Source: varchar(50)
Load_End_Date: Datetime
Hash_Diff: varchar(64)
Balance: decimal(15,2)
Available_Balance: decimal(15,2)
Unavailable_Balance: decimal(15,2)
Authorized_Hold_Balance: decimal(15,2)
Beggining_Of_Day_Balance: decimal(15,2)
Previous_Balance_Amount: decimal(15,2)
Maximum_Balance_Amount: decimal(15,2)
Minimum_Balance_Amount: decimal(15,2)
Last_Transaction_Date: Date
S_CURRENT_ACCOUNT_INFO
PK FK Current_Account_HK: varchar(64)
PK Load_Date: Datetime
Record_Source: varchar(50)
Load_End_Date: Datetime
Hash_Diff: varchar(64)
Account_Opening_Date: Date
Account_Closing_Date: Date
Subproduct_Code: varchar(2)
Account_Manager_Code: varchar(5)
Currency_Code: varchar(3)
Account_Availability_Code: varchar(3)
Authorized_Overdraft_Tax: decimal(10,7)
Contract_Commencement_Date: decimal(10)
Account_Owner_Name: varchar(40)
S_CURRENT_ACCOUNT_CHECKS
PK FK Current_Account_HK: varchar(64)
PK Load_Date: Date time
Record_Source: varchar(50)
Load_End_Date: Datetime
Hash_Diff: varchar(64)
Number_Checks_Used: decimal(6)
Maximum_Amount_Active_Checks: decimal(6)
Amount_Active_Checks: decimal(6)
Maximum_Amount_Modules_Requested: decimal(1)
Maximum_Amount_Checks: decimal(6)
Amount_Checks_Requested: decimal(2)
SAL_CUSTOMER
PK SAL_Customer_HK: varchar(64)
FK H_Base_Customer_HK: varchar(64)
FK H_Alternate_Customer_HK: varchar(64)
Load_Date: Datetime
Record_Source: varchar(50)
H_CUSTOMER
PK Customer_HK: varchar(64)
Load_Date: Datetime
Record_Source: varchar(50)
Customer_Number: decimal(10)
L_CUSTOMER_CURRENT_ACCOUNT
PK Customer_Current_Account_HK: varchar(64)
FK Customer_HK: varchar(64)
FK Current_Account_HK: varchar(64)
Load_Date: Datetime
Record_Source: varchar(50)
H_CURRENT_ACCOUNT
PK Current_Account_HK: varchar(64)
Load_Date: Datetime
Record_Source: varchar(50)
Account_Number: varchar(12)
Product Code: varchar(2)
Branch_Code: varchar(3)
Account_Number_Code: varchar(6)
Account_Check-digit: varchar(1)
L_CUR_ACCOUNT_LOAN_ACCOUNT
PK Cur_Account_Loan_Account_HK: varchar(64)
FK Loan_Account_HK: varchar(64)
FK Current_Account_HK: varchar(64)
Load_Date: Datetime
Record_Source: varchar(50)
S_CUSTOMER_CORPORATE
PK FK Customer_HK: varchar(64)
PK Load_Date: Datetime
Record_Source: varchar(50)
Load_End_Date: Datetime
Hash_Diff: varchar(64)
Establishment_Date: Date
Bankrupcy_Date: Date
Business_Entity_Type: decimal(6)
Number_Of_Employees: decimal(6)
Sales_Volume: decimal(15,2)
Capital_Amount: decimal(6,2)
Telephone_Number: varchar(20)
Fax_Number: varchar(20)
Telex_Number: varchar(11)
Currency_Code: varchar(3)
Company_Type: varchar(1)
Email_Code: varchar(64)
Balance_Currency_Code: varchar(3)
Registration_Code: varchar(15)
Registration_Code: varchar(15)
L_CUSTOMER_RELATIONSHIP
PK Customer_Relationship_HK: varchar(64)
FK Customer_1_HK: varchar(64)
FK Customer_2_HK: varchar(64)
Relationship_Code: varchar(2)
Load_Date: Datetime
Record_Source: varchar(50)
L_CUSTOMER_LOAN_ACCOUNT
PK Customer_Loan_Account_HK: varchar(64)
FK Customer_HK: varchar(64)
FK Loan_Account_HK: varchar(64)
Load_Date: Datetime
Record_Source: varchar(50)
H_LOAN_ACCOUNT
PK Loan_Account_HK: varchar(64)
Load_Date: Datetime
Record_Source: varchar(50)
Account_Number: varchar(12)
Product Code: varchar(2)
Branch_Code: varchar(3)
Account_Number_Code: varchar(6)
Account_Check-digit: varchar(1)
S_LOAN_ACCOUNT_INFO
PK FK Loan_Account_HK: varchar(64)
PK Load_Date: Datetime
Record_Source: varchar(50)
Load_End_Date: Datetime
Hash_Diff: varchar(64)
Account_Opening_Date: Date
Account_Closing_Date: Date
Subproduct_Code: varchar(2)
Currency_Code: varchar(3)
Contract_Rate: decimal(10,7)
Settlement_Rate_Percentage: decimal(10,7)
Installment_Number: decimal(4)
Insurance_Amount: decimal(15,2)
Insurance_Premium_Amount: decimal(15,2)
Tranche_Amount: decimal(15,2)
Contracted_Amount: decimal(15,2)
Initial_Interest_Date: Date
Operational_Deadline_Code: varchar(3)
Utilization_Deadline_Code: varchar(3)
Waiting_Deadline_Code: varchar(3)
Amortization_Deadline_Code: varchar(3)
S_CUSTOMER_RELATIONSHIP
PK FK Customer_Relationship_HK: varchar(64)
PK Load_Date: Datetime
Load_End_Date: Datetime
Hash_Diff: varchar(64)
Record_Source: varchar(50)
Participation_Percentage: decimal(3)
Relationship_Start_Date: Date
S_CUSTOMER_LOAN_ACCOUNT_INFO
PK FK Customer_Loan_Account_HK: varchar(64)
PK Load_Date: Datetime
Record_Source: varchar(50)
Load_End_Date: Datetime
Hash_Diff: varchar(64)
Open_Date: Date
Cancel_Date: Date
Account_Handling_Code: varchar(2)
Signature_Reference_Code: varchar(1)
Ownership_Branch_Code: varchar(3)
Accounting_Account_Branch_Code: varchar(3)
MAS_LOAN_ACCOUNT_OWNERSHIP
PK FK Customer_Loan_Account_HK: varchar(64)
PK Load_Date: Datetime
PK Ownership_Type: varchar(1)
Record_Source: varchar(50)
Load_End_Date: Datetime
Hash_Diff: varchar(64)
Ownership_Number: decimal(2)
S_LOAN_ACCOUNT_ESG
PK FK Loan_Account_HK: varchar(64)
PK Load_Date: Datetime
Record_Source: varchar(50)
Load_End_Date: Datetime
Hash_Diff: varchar(64)
ESG_Compliance_Indicator: varchar(1)
ESG_Loan_Incentive: decimal(10,2)
ESG_Risk_Factor: decimal(6,2)
Carbon_Footprint: decimal(12,2)
Resource_Efficiency: decimal(12,2)
Employee_Welfare_Score: decimal(5,2)
Community_Investment: decimal(10,2)

Figure 23: Data Vault model including all adaptations

In this section, we successfully demonstrated the ability of the model to adapt to an array of new business requirements. The model exhibited robustness and versatility in adapting to different requirements through the application of diverse Data Vault modelling concepts of varying complexity. Notably, we utilized approaches that ranged from maintaining the original model structure to accommodate altering business logic, to the addition of a same-as link table for mapping incompatible customer numbers from different sources.

Despite these advancements, a key limitation was identified in the form of the escalating number of tables required to accommodate new requirements. The initial model consisted of 13 tables, but following the adaptations, this number increased to 21, representing a significant increment. This factor may act as a deterrent to an enterprise-wide implementation of a Data Vault 2.0 EDW, given the potentially overwhelming complexity and management challenges posed by such an expansion.

In a broader context, the flexibility and robustness demonstrated by the adapted model highlight the viability of Data Vault modelling in evolving business environments. The body of solutions offered by the adapted model could potentially serve as a proof of concept for organizations looking to adapt their data infrastructure in response to rapidly changing business requirements, by implementing a Data Vault 2.0 EDW.

6. EVALUATION

In this section, we delve into the Evaluation phase of the DSR. The principal objective of this phase is to assess whether the proposed artefact fulfils the purpose for which it was designed, as outlined by Venable et al. (2012).

The structure of this section unfolds as follows: Initially, Section 6.1. outlines the methodology utilized during the evaluation process. Subsequently, an analysis of the questions posed to the participants unfolds across Sections 6.2., 6.3., and 6.4.. Finally, Section 6.5. provides a discussion on the results of the evaluation.

6.1. METHODOLOGY

This study employed semi-structured one-on-one interviews as the primary method to evaluate our artefact, a Data Vault model. The interviewees consisted of experienced members from the collaborating organization. These individuals were identified as key evaluators due to their deep experience and insights into the organization's operations and data infrastructure. Their extensive knowledge of the banking industry and the organization's internal operations and data infrastructure equip them to evaluate the model's accuracy in portraying the business, the efficacy of the model, and its fit with the organization's context.

6.1.1. Participant Characteristics

The participants in the interviews were experienced members from diverse roles within the organization and the broader banking industry. Each brought a unique level of experience and seniority, leading to a variety of perspectives on the proposed model. Their roles, spanning multiple data-related fields, allowed for a broad spectrum of opinions. This diversity not only enriched the value of the evaluation but also provided unique insights since each participant would interact differently with the model, prioritizing and raising concerns based on their individual experiences. Participant details are listed in Table 21.

Table 21: Profile of the participants

Identifier	Age	Gender	Role within the organization	Area	Banking experience	Data-related experience
Participant 1	49	Male	Grade III Technician	Data Quality	29 years	29 years
Participant 2	55	Female	Grade II Technician	Data Quality and Governance	12 years	19 years
Participant 3	37	Female	External Consultant - Analyst/Programmer	Projects	9 years	9 years
Participant 4	56	Male	External Consultant - Analyst/Programmer	Projects	32 years	15 years
Participant 5	51	Female	Grade III Technician	Data Governance	22 years	5 years
Participant 6	48	Female	Grade II Technician	Data Governance	25 years	2 years
Participant 7	54	Male	Chief Data Officer	Data Quality and Governance	23 years	18 years

6.1.2. Interview design

The duration of each interview ranged from 35 to 60 minutes, with an average length of 45 minutes. The specific duration varied depending on the length of the participants' answers, allowing flexibility while ensuring comprehensive data collection. To facilitate the interview process, we divided it into two distinct parts. In the initial 15 minutes, we provided a brief overview of the data vault model and demonstrated its ability to adapt to new business requirements. This introductory phase aimed to familiarize participants with the constructed model and allow participants to give more insightful feedback. Following the introduction, the next 30 minutes were dedicated to the interview questions, allowing ample time for in-depth discussions. In total, 11 questions were asked to each of the participants. To ensure smooth execution, mock interviews were conducted as a rehearsal before the actual interviews, which confirmed that a 45-minute duration provided an ideal balance between obtaining comprehensive insights and maintaining participant engagement.

All the interviews were conducted in Portuguese, as it is the native language of all the interviewees, thereby facilitating a more accurate and nuanced expression of their views. Subsequently, the quotes presented during this section from both questions and answers received were translated from Portuguese to English for the purpose of this Dissertation. A full transcript of each interview is found, in Portuguese, in the annex of this document. The conducting, transcribing, and analyzing of the interviews were executed according to the guidelines set by Rubin and Rubin (2005). The details of the interview structure are presented in Table 22.

Table 22: Details of the interview

Interview details	**Description**
Number of participants	7
Interview duration	45 minutes
Number of posed questions	11
Interview language	Portuguese
Interview method	Online video-call

6.1.3. Interview questions

Nine questions were directed to each of the interviewees, alongside two open-ended questions at the end of the interview, in order to cover any additional thoughts the participant might have. We constructed the questions based on Prat et al.'s hierarchy of evaluation criteria (2014), ensuring a balanced and complete evaluation of all aspects of the model. The evaluation focused on three interconnected aspects of the model: the accuracy of the model in mirroring the business, the model's efficacy, and its alignment with the organization. These combined criteria offer a thorough evaluation of the model, providing detailed insights while keeping the interviews to a manageable length and maintaining participant engagement. The questions, arranged according to the specific criteria they address, are presented in Table 22.

Table 23: Questions posed to the participants, grouped by criteria

Criteria	Sub-criteria	Question
Accuracy of the model	**Account ownership**	In terms of account ownership, does the model reflect the business in an accurate way?
	Customer Interrelationships	In terms of the relationship between customers, does the model reflect the business in an accurate way?
	Overall accuracy of the model	Overall, does the model reflect the business in an accurate way, considering the represented business concepts and associated relationships, while complying with the recommended practices of Data Vault 2.0
Efficacy of the model	**Completeness of the model**	Considering only the business domains modeled, how would you classify the model in terms of completeness?
	Simplicity of the model	In your opinion, is the proposed model simple to understand and use?
	Robustness of the model	How would you classify the robustness of the model, in its ability to adapt to both the presented business requirements and those that may arise in the future?
Fit with organization	**Model as a proof of concept**	In your opinion, does the model represent a good proof of concept for a future implementation?
	Importance of the model in the context of the organization	In what way do you consider the existence of the proposed artefact pertinent and/or important within the context of the organization?
	Utility to data users within the organization	In your opinion, can the proposed model be useful for data architects, engineers, and analysts of the organization?
Miscellaneous		What recommendations or suggestions would you give to improve the model?
		What other comments can you provide about the model?

For each criterion, represented by a subsection in this document, the rationale for every question is explained, complemented by an analysis of the significance of the participants' areas of expertise and their responses. A comprehensive overview of the collective perceptions for each question is provided, emphasizing noteworthy comments for deeper insight. Points of divergence are also identified and discussed, highlighting potential areas for improvement in the model. Each subsection concludes with a succinct summary of the findings and a discussion of their implications, providing valuable directions for future research or improvement.

6.2. ACCURACY OF THE MODEL

A few questions were asked to the participants regarding the model itself, and how it portrays the organization's business concepts. First, the participants were inquired about certain aspects of the model, namely the aspects which are important, but less easily comprehensible, these being, account ownership and the relationships between customers. Then, the participants are inquired on the overall ability of the model to accurately reflect the business.

6.2.1. Account Ownership

The method employed to depict the relationship between customers and their current accounts, as described in Section 4.3.4., incorporates an account ownership aspect. This facet, while crucial, isn't as immediately comprehensible as other parts of the model. Given that the participants all boast an extensive and experienced knowledge of the organization's data and business operations, they serve as insightful evaluators on the accuracy of the approach applied to this particular segment of the model. The particular question posed to the participants in order to assess this, was the following, ”In terms of account ownership, does the model reflect the business in an accurate way?”

All seven participants agreed that the model aptly captured the nuances of account ownership. Their responses were generally concise, demonstrating their confidence in the approach. For instance, Participant 2 commended the model's clarity, stating, ”There is a customer that has an ownership type, and links with the account, with that ownership type. Perfect”. Participant 6 acknowledged the innovation of the approach, stating, ”Yes, I think it was quite an innovative way to overcome the issues”.

These affirmations from experienced insiders serve as validation for the approach adopted to model this particular segment. Their confidence serves as a robust indicator of the model's adequacy and its ability to accurately reflect the intricacies of account ownership within the business context.

6.2.2. Customer Interrelationships

The approach used to model the relationship between the organization's customers is described in detail in Section 4.3.2.. This aspect of the model isn't as easy to comprehend as most other parts of the model, mainly due to the use of a link table that connects two of the same type of hubs, and a dependent child key, a lesser-known and complex tool used in Data Vault modelling. It is through the comprehensive knowledge of the participants about both the organization's data and business operations that we can validate this approach for modelling customer relationships. The participants were asked, ”In terms of the relationship between customers, does the model reflect the business in an accurate way?”

All seven participants concurred that the Data Vault model adequately represents the relationships between customers. The responses were succinct and delivered with conviction, though a significant portion of the discussion was devoted to clarifying the term ”relationships between customers”. Of particular note, Participant 2 emphasized the potential benefits of the approach, stating, ”Perfect. In a relationship between a customer and another customer, there we have tutors, guardians, we have several options. Or even with a corporate customer and an individual. What is their function? Are they an administrator? A manager? A partner? It's great”.

Participant 3 found the modelling of the relationship adequate, and added a noteworthy suggestion, ”What could be interesting, is how you have the reference tables, which you ended up not using a lot. Those types of codes (Relationship_Code) could be decodified...For users who don't know what the code means, they could benefit if they had the code descriptor.”

The feedback received from the participants serves as a robust validation of the methodology employed to model the relationships between the organization's customers. A prevalent enthusiasm was detected regarding the potential of the model to represent a diverse set of relationships between customers. Future improvements could incorporate the proposition from Participant 3, regarding the inclusion of a ”Relationship_Code” reference table, which could be considered to further optimize the model's user-friendliness.

6.2.3. Overall accuracy of the model

In the final part of the interview concerning the participants' perceptions of the model and the modeling decisions made, we sought their views on the model's overall accuracy. Specifically, we asked, ”Overall, does the model reflect the business accurately, considering the represented business concepts and associated relationships, while complying with the recommended practices of Data Vault 2.0?”

The seven participants participant agreed that the model faithfully represented the business. The majority of the responses were concise affirmations of the model's accuracy. Participant 2 gave an extensive answer and provided valuable insight by emphasizing the superior functionality of satellite tables compared to the current Data Warehouse implementation, stating, ”One of the biggest issues we have with the Data Warehouse is that we treat all tables with the same importance when they're not equally important...we don't need to spend time updating tables that do not need to be updated. I think it's great.”

Participant 3 suggested improvements related to customer data integration from various sources, expressing a need for an easier way to identify the source of customer data. They noted, ”Given the frequent instances of customer data loading from multiple organizations, there should be a field specifying the source company code, as a description of the source.”

In summary, the model received overwhelmingly positive feedback, validating the soundness of the modeling decisions and its accurate representation of the business. Participant 3's suggestion provides an intriguing potential for future improvements of the model.

6.3. EFFICACY OF THE MODEL

To measure the efficacy of the proposed model, a comprehensive assessment was undertaken with a focus on three principal factors: the model's completeness, simplicity, and adaptability. The adoption of the model as a fully realized Data Vault model predicates upon these aspects, as they play a pivotal role in characterizing the organizational business logic, facilitating straightforward implementation, and allowing for dynamic response to constant business changes.

In the following subsections, the opinions of the participants regarding these factors are thoroughly analyzed.

6.3.1. Completeness of the model

All interviewees possess a medium to expert knowledge of the bank's data and operations, particularly concerning business concepts such as Customers and Current Accounts represented in the model. This expertise equips them to provide informed opinions about the accuracy and completeness of the model's representation of business concepts. Each participant was asked, ”Considering only the business domains modeled, how would you classify the model in terms of completeness?”

All of the seven participants considered the modelling of the business concepts complete. In particular, Participant 7 exclaimed, ”I think it is very complete, it addressed all of the relevant topics and more. Even the tricky ones, that are less obvious, I think were well modelled and transposed.” Participant 4 stated, ”I think it is quite complete...especially the main concepts like having the customers well defined, the accounts defined, and the relationships between these two concepts, I think that is very complete.”

While Participant 7 viewed the model as complete, they noted some limitations due to the lack of complete access to the bank's data structure. They observed, ”I think that a lot of basic things are missing, if you don't have access to the entirety of the bank's structure, I think that imposed some limitations...I would say it's complete within what was provided to you.”

6.3.2. Simplicity of the model

A crucial factor in evaluating the proposed model is its simplicity, which significantly influences its adoption across the enterprise. Participants were asked to assess the model's ease of understanding and use. The specific question posed was: ”In your opinion, is the proposed model simple to understand and use?”

Opinions varied regarding the perceived simplicity of the model. Five out of seven participants deemed the model simple to understand and use, with Participant 2 noting, ”It is simple to understand and simple to alter and modify anything that needs to be adjusted.”

It was notably apparent that participants more familiar with Data Vault modelling exhibited greater confidence in the model's simplicity. Participants 2 and 7, who had the most knowledge about Data Vault modelling, respectively stated, ”The characteristic of separating keys to one side, attributes to another side, and attributes to another, that vision to me is very simple” and ”To me, using this model, even for data exploration, I do not think it's something extremely complex. On the contrary, you basically have to know what the hubs are, and where they are. All the paths start with the hubs...The model is fundamentally hubs and links; the satellites are just additional...But the model is direct.”

Conversely, three out of the seven participants expressed reservations, with particular emphasis on doubts regarding the Data Vault modelling methodology itself, rather than the specific characteristics of the actual model. Participant 4 noted, ”It has a lot of tables, a lot of links. I think it needs to be thoroughly studied to be understood.” Participant 1 claimed, ”I think the model you presented was simple, considering we're talking about Data Vault.”

6.3.3. Robustness of the model

One of the primary advantages attributed to the Data Vault is its ability to seamlessly adapt to emerging business requirements Linstedt and Olschimke, 2015, p. 108. To assess this adaptability, participants were prompted to consider not only the business requirements presented during the demonstration phase of this study but also potential future requirements that might arise. The question asked to each participant, about this topic is the following: ”How would you classify the robustness of the model, in its ability to adapt to both the presented business requirements and those that may arise in the future?”

All twelve participants agreed that the model demonstrates strong adaptability to the business requirements presented. Additionally, the majority also considered it well-positioned to meet emergent business requirements in the future. Participant 5 asserted, ”The model appears 100% adaptable, any change easy to resolve.” Similarly, Participant 7 noted, ”I think it was evident that the model can easily adapt to new business needs”.

While there was a consensus on the model's adaptability to the business requirements demonstrated, three participants expressed reservations concerning its ability to handle unanticipated future requirements. Participant 1 noted, ”It appears to me that the model has a certain capacity for adaptability and scalability. However, I can't make predictions of the future, if it's going to adapt or not, I don't know.”

6.4. FIT WITH THE ORGANIZATION

0One of the evaluation criteria presented by Prat et al., 2014 (2014) is fit with the organization, characterized as the alignment of the artefact with its organizational environment (Hevner et al., 2004). This is a particularly important facet to evaluate our model on, considering it was developed in collaboration with an organization in the banking sector. The interview participants, with their knowledge of both the organization and the broader banking industry, are ideally suited to assess this aspect.

In the following subsections, we provide a detailed analysis of the participants' viewpoints on how well the model fits with the organization.

6.4.1. Model as a proof of concept

As discussed in previous sections, this project does not cover the full scope of modelling the entirety of the bank's data, instead focusing on Customers and their relationships with Current Accounts. It is important to assess whether the modelling of these concepts represents a good proof of concept for modelling the entirety of the bank's data. The question posed to participants regarding this subject was as follows: ”In your opinion, does the model represent a good proof of concept for a future implementation?”

All seven participants agreed that the model serves as an effective proof of concept for a future enterprise-wide Data Vault 2.0 model implementation. The responses to this question were generally succinct. Participant 3 offered the most extensive comment, stating, ”That would be the great challenge...I think it is an excellent starting point to start the project.”

This unanimous positive response suggests a strong consensus among participants about the potential value of this model as a proof of concept for future implementations.

6.4.2. Importance of the model in the context of the organization

The participants of this evaluation represent a diverse array of roles within the organization, each bringing unique experience and insight. Their individual perspectives on the proposed model's importance, shaped by their unique roles and experiences, are invaluable to this assessment. The question posed to the participants was as follows, ”In what way do you consider the existence of the proposed artefact pertinent and/or important within the context of the organization?”

Each participant acknowledged the model's significance, at least to some extent. This question prompted the most detailed responses, as participants frequently linked the model's importance to their struggles with the existing system. For instance, Participant 3 emphasized its ease of modification as a significant advantage, stating, ”Currently, introducing or modifying an attribute is an extensive process. This should be a more linear operation.” Similarly, Participant 4 considered the model's potential for performance enhancement as a critical factor, suggesting that it could lead to ”a significant improvement” in the bank's operations.

Moreover, Participant 1 highlighted that even if the Data Vault model is not adopted across the entire organization, its key concepts could still be extracted and implemented, noting that the model offers ”solutions that could be repurposed for other applications, and even at an object level...There are certain benefits for sure, even to apply to other situations.”

Additionally, two participants compared the Data Vault methodology's merits to other widely-used data warehouse methodologies. Participant 2 believed the new model would be ”much faster to implement” compared to the ones based on the Inmon approach. Echoing this sentiment, Participant 7, the Chief Data Officer, identified flexibility as a key advantage of the Data Vault. They elaborated on the rigidity of Kimball approaches and the inability to adapt to change without a complete overhaul. However, with Data Vault, they saw the potential for evolution, stating, ”If a change arises, we can easily discard the dimensional model and create a new one, based on the updated Data Vault structure. At present, our progress is hindered by our inability to repurpose resources until the model is fully defined.”

In conclusion, the feedback to the question posed shows a largely positive response towards the proposed Data Vault model, primarily due to its flexibility, ease of modification, and potential for improving performance. The responses also suggest that even if Data Vault modelling is not adopted across the enterprise, key concepts could be extracted and utilized.

6.4.3. Utility to data users within the organization

As the participants of the interviews hold a variety of roles within the organization, spanning multiple departments such as Data Quality, Data Governance, and Project Management, their perspective is uniquely valuable. These individuals are not only familiar with the intricacies of their respective departments but are also experienced data users themselves. Hence, their insights carry significant weight when it comes to evaluating the utility of the proposed model for data users across the organization. To evaluate this, a specific question was posed to the participants: ”In your opinion, can the proposed model be useful for data architects, engineers, and analysts of the organization?”

Out of seven participants, six agreed that the proposed model would be of benefit to the organization’s data users. Although most responses were concise, one participant offered a more detailed perspective on the model’s usefulness, emphasizing its diverse applications across different data-related roles. Participant 2 stated, ”I think that for someone that has the job of modelling a repository for a company, this model is extremely important. To the remaining users, I believe it takes a while. To data analysts, in particular, it will take a while to understand how the concepts are distributed throughout so many tables...I believe it’s a matter of habit”

In contrast to the majority opinion, there was one participant who did not confidently assert that the model would be beneficial for data users. Participant 4 demonstrated hesitation, stating, ”I think so, but I don’t know, I can’t answer that question.”

6.5. RESULTS DISCUSSION

The conducted interviews yielded valuable insights about the proposed model’s accuracy, efficacy, and fit within the organization. The primary insights include:

- The importance of the model in the context of the organization was a particularly vital aspect of the evaluation. Irrespective of the model satisfying the other evaluated criteria, its perceived importance among its primary users ultimately determines its value. All participants recognized the model’s importance to the organization to varying degrees.

Participants pointed out the model's potential to improve on certain deficiencies of the current system. Specifically, they valued its flexibility to change existing attributes, enhancing the agility of a currently tedious process. Moreover, the speed at which it can be implemented and modified greatly outperforms the current system, promising faster value delivery and more efficient resource utilization.

- Participants unanimously viewed the model as a promising proof of concept for a future implementation of an organization-wide Data Vault 2.0 EDW. A vast majority agreed it would benefit data users, such as architects, engineers, and analysts. However, one participant noted that data analysts would initially struggle to adapt, given the numerous tables generated by the Data Vault model.

- The opinions on the simplicity of the model were varied. While the majority of the participants agreed that the model itself was simple to understand, there were some doubts regarding the simplicity of Data Vault modelling as a concept. Participants who were more familiarized with Data Vault 2.0 found the model itself and Data Vault modelling as a whole simple to understand and use.

- All of the participants found the model complete, considering the business domains modelled. A majority acknowledged that the scope of the project inherently restricted the model. However, they agreed that considering the limited access we had to the bank's data infrastructure, the model was indeed complete.

- The robustness of the model, characterized by its ability to adapt to new business requirements is one of the key advantages of Data Vault modelling. This feature was generally appreciated among the participants. However, a few of the participants, despite recognizing the ability to adapt to the presented business requirements, had some reservations regarding the ability to adapt to future business requirements.

- The consensus among all participants was that the model accurately reflected the business and its specific nuances, namely the account ownership and the relationships between customers aspects.

In conclusion, the results of the interviews demonstrate overall positive perceptions of the proposed model. The participants acknowledged that within the scope of the project, the model was very complete in its portrayal of the modelled business concepts. It accurately reflected the business logic and its specific nuances. Additionally, the model was viewed as a valuable proof of concept for the future implementation of an enterprise-wide Data Vault 2.0 EDW and was deemed important within the context of the organization.

7. CONCLUSIONS

This research aimed to explore the use of a Data Vault 2.0 EDW within the context of the banking industry. Specifically, this research aimed to address the question: ”How can a Data Vault 2.0 EDW address the primary data management challenges of an organization within the banking industry, being served by a Data Lake with a proper architecture”. To maximize the real-world application of our research and allows us to provide valuable and relevant results, this research was conducted in collaboration with a Portuguese bank that serves a customer base exceeding three million.

To explore the application of a Data Vault 2.0 EDW, we developed a Data Vault 2.0 model based on d organization’s data, particularly customer and current account data. The goal of the model was to present an accurate representation of the underlying business logic, including specific nuances such as customer interrelationships, and the intricacies of account ownership.

Following the creation of the model, we exhibited its adaptability to changing business requirements, a characteristic intrinsic to the rapidly evolving banking industry. This industry, characterized by the wide array of services provided and stringent regulatory measures, necessitates continuous adaptability. To assess this, the model was subject to realistic business requirements devised collaboratively with members from within the banking organization.

Upon the development and adaptation of the model, we conducted semi-structured one-on-one interviews with experienced members of the partner organization. These participants offered valuable insights into several aspects of the model. Their feedback particularly focused on its accuracy in representing business concepts and their relationships, its simplicity, adaptability, and completeness. Additionally, they elaborated on the model’s importance within the context of the organization.

Our findings suggest that Data Vault 2.0 modelling can effectively mirror an organization’s business logic while remaining simple to understand, thereby facilitating a wide-scale adoption within large enterprise environments.

The Data Vault 2.0 model proved highly adaptable to new business requirements, from adding attributes to a table to the introduction of new business concepts or modification of existing business logic. This adaptability is a vital feature for the banking industry, which continuously grapples with rapidly evolving business requirements and the need to adapt to new regulatory measures.

A significant advantage of Data Vault 2.0, outlined by members of the organization, lies in its iterative development process. Changes can be made without necessitating a complete model restructuring, enabling an organization to start delivering value sooner without waiting for the model to be fully defined. This methodology not only enhances resource efficiency but also bolsters the agility of associated processes.

Based on our research, we conclude that a Data Vault 2.0 EDW is a viable and advantageous Data Warehousing methodology for organizations in the banking industry. It addresses numerous prevalent data management issues faced in the industry and thus has significant potential to improve operational efficiency within the banking industry.

7.1. LIMITATIONS

Despite the overall positive results of this research, it is important to refer the limitations of this research.

One such limitation arose from the methodological constraints tied to the evaluation process. Although we obtained valuable qualitative data through one-on-one semi-structured interviews with experienced members of the collaborating organization, we were unable to perform a quantitative evaluation. This was primarily due to the restrictive nature of the banking sector, with its emphasis on privacy and security, which rendered the required data access impossible for us as externals to the organization. Ideally, we would have conducted performance and scalability benchmarks to gain a more in-depth understanding of the model, especially when comparing it with the current implementation. While the qualitative interviews provided significant insights, a combination of both qualitative and quantitative evaluations would have permitted a more complete assessment of the model.

Secondly, due to the extensive nature of banking data, we were only able to model a segment of the organization's data, specifically customers and current accounts. Fully designing a Data Vault model to represent the entirety of the bank's operations was beyond the scope of this project and would have been unfeasible given the constraints. An expansion of the model to include more areas of the bank's operations would potentially offer more comprehensive insights regarding its utility and value.

Lastly, this research was conducted in the context of limited existing literature on Data Vault, particularly Data Vault 2.0, and its integration with data sources. This scarcity of previous studies posed a challenge for this research. Our understanding of Data Vault 2.0 and its potential integration strategies would have undoubtedly benefitted from a more robust body of academic literature. It emphasizes the need for further research in this area to support future studies.

7.2. RECOMMENDATIONS FOR FUTURE WORK

In light of the findings from this study, several avenues for future research emerge.

As our research was limited to a segment of the collaborating organization's data, one promising area involves the complete implementation and subsequent performance evaluation of an enterprise-wide Data Vault 2.0 EDW within a large-scale organization. This assessment could utilize both quantitative and qualitative evaluation methodologies to compare its performance against previously implemented data warehouses.

Our research was conducted in collaboration with a more than century-old organization, as a result, a Data Vault 2.0 EDW implementation would necessitate accommodating years of historical data from legacy systems. An interesting contrast could be the implementation of a Data Vault 2.0 EDW within a recently founded organization, one devoid of pre-existing data infrastructure systems. This scenario could yield valuable insights into the usability of Data Vault 2.0 when an organization starts from scratch. Specifically, it would be valuable to explore the iterative implementation of a Data Vault 2.0 EDW and its scalability as the organization grows.

Moreover, although the current literature acknowledges the benefits of integrating data lakes and data warehouses, it tends to overlook the aspect of their mutual design considerations. Future work could delve into this intersection, proposing best practices for designing a data lake intended as a source for a data warehouse. This exploration would be particularly valuable in the context of a Data Vault 2.0 EDW.

Finally, considering the limited existing literature on Data Vault 2.0, further research in this field is necessary. This would not only contribute to our theoretical understanding of the system but also facilitate its practical applications across different contexts.

BIBLIOGRAPHICAL REFERENCES

Bhatt, S., Shaikh, T., & Wiebe, G. (2022). Prescriptive Guidance for Implementing a Data Vault Model on the Databricks Lakehouse Platform. Retrieved January 16, 2023, from https://www.databricks.com/blog/2022/06/24/prescriptive-guidance-for-implementing-a-data-vault-model-on-the-databricks-lakehouse-platform.html

Databricks. (2022). What is the medallion lakehouse architecture? — Databricks on AWS. Retrieved December 21, 2022, from https://docs.databricks.com/lakehouse/medallion.html

Databricks. (2023). What is Delta Lake? — Databricks on AWS. Retrieved February 1, 2023, from https://docs.databricks.com/delta/index.html

Dehghani, Z. (2019). How to Move Beyond a Monolithic Data Lake to a Distributed Data Mesh. Retrieved January 7, 2023, from https://martinfowler.com/articles/data-monolith-to-mesh.html

Diener, F., & Špaček, M. (2021). Digital transformation in banking: A managerial perspective on barriers to change. *Sustainability (Switzerland)*, *13*(4), 1–26. https://doi.org/10.3390/su13042032

Garousi, V., Felderer, M., & Mäntylä, M. V. (2019). Guidelines for including grey literature and conducting multivocal literature reviews in software engineering. *Information and Software Technology*, *106*, 101–121. https://doi.org/10.1016/j.infsof.2018.09.006

Giebler, C., Groger, C., Hoos, E., Schwarz, H., & Mitschang, B. (2020). A Zone Reference Model for Enterprise-Grade Data Lake Management. *2020 IEEE 24th International Enterprise Distributed Object Computing Conference (EDOC)*, 57–66. https://doi.org/10.1109/EDOC49727.2020.00017

Hai, R., Geisler, S., & Quix, C. (2016). Constance: An intelligent data lake system [Type: Conference paper]. *Proceedings of the ACM SIGMOD International Conference on Management of Data*, *26-June-2016*, 2097–2100. https://doi.org/10.1145/2882903.2899389

Herden, O. (2020). Architectural Patterns for Integrating Data Lakes into Data Warehouse Architectures [Series Title: Lecture Notes in Computer Science]. In L. Bellatreche, V. Goyal, H. Fujita, A. Mondal, & P. K. Reddy (Eds.), *Big Data Analytics* (pp. 12–27). Springer International Publishing. https://doi.org/10.1007/978-3-030-66665-1_2

Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design Science in Information Systems Research [Publisher: Management Information Systems Research Center, University of Minnesota]. *MIS Quarterly*, *28*(1), 75–105. Retrieved June 11, 2023, from http://www.jstor.org/stable/25148625

Hu, H., Wen, Y., Chua, T.-S., & Li, X. (2014). Toward Scalable Systems for Big Data Analytics: A Technology Tutorial. *IEEE Access*, *2*, 652–687. https://doi.org/10.1109/ACCESS.2014.2332453

Hultgren, H. (2012). *Modeling the agile data warehouse with data vault*. New Hamilton.

Inmon, B. (2016). *Data Lake Architecture: Designing the Data Lake and Avoiding the Garbage Dump* (1st). Technics Publications, LLC.

Inmon, W. H. (1992). *Building the data warehouse* (1st ed). Wiley.

Jemmali, R., Abdelhedi, F., & Zurfluh, G. (2022). DLToDW: Transferring Relational and NoSQL Databases from a Data Lake. *SN Computer Science*, *3*(5), 381. https://doi.org/10.1007/s42979-022-01287-7

Jia, J. (2020). Lambda, Kappa and now Delta. Retrieved February 1, 2023, from https://jixjia.com/delta-architecture/

Kim, S., Kumar, N., Lee, J., & Oh, J. (2022). ESG lending. *Proceedings of Paris December 2021 Finance Meeting EUROFIDAI-ESSEC*.

Kimball, R., Ross, M., & Anisimov, A. A. (2003). The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling (2nd Edition) [Type: Article]. *SIGMOD Record*, *32*(3), 101–102. https://doi.org/10.1145/945721.945741

Kiran, M., Murphy, P., Monga, I., Dugan, J., & Baveja, S. S. (2015). Lambda architecture for cost-effective batch and speed big data processing. *2015 IEEE International Conference on Big Data (Big Data)*, 2785–2792. https://doi.org/10.1109/BigData.2015.7364082

Kitchenham, B. (2004). Procedures for Performing Systematic Reviews, 33.

Kreps, J. (2014). Questioning the Lambda Architecture. Retrieved December 21, 2022, from https://www.oreilly.com/radar/questioning-the-lambda-architecture/

Lee, D. (2020). Beyond Lambda: Introducing Delta Architecture. Retrieved February 1, 2023, from https://www.databricks.com/discover/getting-started-with-delta-lake-tech-talks/beyond-lambda-introducing-delta-architecture

Li, Y., Zhang, A., Zhang, X., & Wu, Z. (2018). A Data Lake Architecture for Monitoring and Diagnosis System of Power Grid. *Proceedings of the 2018 Artificial Intelligence and Cloud Computing Conference on ZZZ - AICCC '18*, 192–198. https://doi.org/10.1145/3299819.3299850

Linstedt, D. (2002). Data Vault Series 1 – Data Vault Overview. Retrieved June 14, 2023, from https://tdan.com/data-vault-series-1-data-vault-overview/5054

Linstedt, D., & Olschimke, M. (2015). *Building a Scalable Data Warehouse with Data Vault 2.0* [Publication Title: Building a Scalable Data Warehouse with Data Vault 2.0 Type: Book]. https://doi.org/10.1016/C2014-0-02486-0

Liu, P., Loudcher, S., Darmont, J., & Noûs, C. (2021). ArchaeoDAL: A Data Lake for Archaeological Data Management and Analytics [arXiv:2107.11157 [cs]]. *25th International Database Engineering & Applications Symposium*, 252–262. https://doi.org/10.1145/3472163.3472266

Mitruś, P. (2021). Data Lake Architecture: How to create a well Designed Data Lake. Retrieved December 21, 2022, from https://lingarogroup.com/blog/data-lake-architecture

Orescanin, D., & Hlupic, T. (2021). Data Lakehouse - a Novel Step in Analytics Architecture. *2021 44th International Convention on Information, Communication and Electronic Technology (MIPRO)*, 1242–1246. https://doi.org/10.23919/MIPRO52101.2021.9597091

Oukhouya, L., El haddadi, A., Er-raha, B., & Asri, H. (2021). A generic metadata management model for heterogeneous sources in a data warehouse (S. Krit, Ed.). *E3S Web of Conferences*, *297*, 01069. https://doi.org/10.1051/e3sconf/202129701069

Oukhouya, L., El haddadi, A., Er-raha, B., Asri, H., & Laaz, N. (2023). A Proposed Big Data Architecture Using Data Lakes for Education Systems [Series Title: Lecture Notes on Data Engineering and Communications Technologies]. In M. Ben Ahmed, B. A. Abdelhakim, B. K. Ane, & D. Rosiyadi (Eds.), *Emerging Trends in Intelligent Systems & Network Security* (pp. 53–62). Springer International Publishing. https://doi.org/10.1007/978-3-031-15191-0_6

Peffers, K., Rothenberger, M., Tuunanen, T., & Vaezi, R. (2012). Design Science Research Evaluation [Series Title: Lecture Notes in Computer Science]. In D. Hutchison, T. Kanade, J. Kittler, J. M. Kleinberg, F. Mattern, J. C. Mitchell, M. Naor, O. Nierstrasz, C. Pandu Rangan, B. Steffen, M. Sudan, D. Terzopoulos, D. Tygar, M. Y. Vardi, G. Weikum, K. Peffers, M. Rothenberger, & B. Kuechler (Eds.), *Design Science Research in Information Systems. Advances in Theory and Practice* (pp. 398–410). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-29863-9_29

Peffers, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2007). A Design Science Research Methodology for Information Systems Research. *Journal of Management Information Systems*, *24*(3), 45–77. https://doi.org/10.2753/MIS0742-1222240302

Pisoni, G., Molnár, B., & Tarcsi, Á. (2021). Data Science for Finance: Best-Suited Methods and Enterprise Architectures. *Applied System Innovation*, *4*(3), 69. https://doi.org/10.3390/asi4030069

Prat, N., Comyn-Wattiau, I., & Akoka, J. (2014). ARTIFACT EVALUATION IN INFORMATION SYSTEMS DESIGN-SCIENCE RESEARCH – A HOLISTIC VIEW.

Ravat, F., & Zhao, Y. (2019). Data Lakes: Trends and Perspectives [Series Title: Lecture Notes in Computer Science]. In S. Hartmann, J. Küng, S. Chakravarthy, G. Anderst-Kotsis, A. M. Tjoa, & I. Khalil (Eds.), *Database and Expert Systems Applications* (pp. 304–313). Springer International Publishing. https://doi.org/10.1007/978-3-030-27615-7_23

Ribeiro, A., Silva, A., & Da Silva, A. R. (2015). Data Modeling and Data Analytics: A Survey from a Big Data Perspective. *Journal of Software Engineering and Applications*, *08*(12), 617–634. https://doi.org/10.4236/jsea.2015.812058

Rubin, H., & Rubin, I. (2005). *Qualitative Interviewing (2nd ed.): The Art of Hearing Data*. SAGE Publications, Inc. https://doi.org/10.4135/9781452226651

Saddad, E., El-Bastawissy, A., M., H., & Hazman, M. (2020). Lake Data Warehouse Architecture for Big Data Solutions. *International Journal of Advanced Computer Science and Applications*, *11*(8). https://doi.org/10.14569/IJACSA.2020.0110854

Sakr, S., & Zomaya, A. Y. (Eds.). (2019). *Encyclopedia of Big Data Technologies*. Springer International Publishing. https://doi.org/10.1007/978-3-319-77525-8

Sarramia, D., Claude, A., Ogereau, F., Mezhoud, J., & Mailhot, G. (2022). CEBA: A Data Lake for Data Sharing and Environmental Monitoring. *Sensors*, *22*(7), 2733. https://doi.org/10.3390/s22072733

Sharma, B. (2018). Architecting Data Lakes, 50.

Singh, J., Singh, G., & Bhati, B. S. (2022). The Implication of Data Lake in Enterprises: A Deeper Analytics. *2022 8th International Conference on Advanced Computing and Communication Systems (ICACCS)*, 530–534. https://doi.org/10.1109/ICACCS54159.2022.9784986

Sivarajah, U., Kamal, M. M., Irani, Z., & Weerakkody, V. (2017). Critical analysis of Big Data challenges and analytical methods. *Journal of Business Research*, *70*, 263–286. https://doi.org/10.1016/j.jbusres.2016.08.001

van 't Westeinde, A. (2022). Unifying batch and stream processing in a data lakehouse. Retrieved February 1, 2023, from https://www.alten.nl/2022/09/26/unifying-batch-and-stream-processing-in-a-data-lakehouse/

Venable, J., Pries-Heje, J., & Baskerville, R. (2012). A Comprehensive Framework for Evaluation in Design Science Research [Series Title: Lecture Notes in Computer Science]. In D. Hutchison, T. Kanade, J. Kittler, J. M. Kleinberg, F. Mattern, J. C. Mitchell, M. Naor, O. Nierstrasz, C. Pandu Rangan, B. Steffen, M. Sudan, D. Terzopoulos, D. Tygar, M. Y. Vardi, G. Weikum, K. Peffers, M. Rothenberger, & B. Kuechler (Eds.), *Design Science Research in Information Systems. Advances in Theory and Practice* (pp. 423–438). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-29863-9_31

Wohlin, C., Runeson, P., Höst, M., Ohlsson, M. C., Regnell, B., & Wesslén, A. (2012). *Experimentation in Software Engineering*. Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-29044-2

Zhao, Y., Megdiche, I., Ravat, F., & Dang, V.-n. (2021). A Zone-Based Data Lake Architecture for IoT, Small and Big Data. *25th International Database Engineering & Applications Symposium*, 94–102. https://doi.org/10.1145/3472163.3472185

ANNEXES

Interviewee: Participant 1
Interview Date: 26/06/2023

I: Em termos dos beneficiários das contas cartão e essa parte do modelo, considera que o modelo reflete o negócio de forma precisa?

> *P1: Pah, essa pergunta é uma pergunta com rasteira, porque teria de fazer eu uma análise para ver se de facto se isso, se isso era a melhor solução ou não. Agora, nos pressupostos daquilo que vocês me apresentaram sim. Esses pressupostos sim.*

I: Em termos da titularidade das contas, e daquela questão dos satélites multi-ativos, o modelo reflete o negócio de forma precisa?

> *P1: Sim, sim. Lá está, sempre com as reservas dos pressupostos não é –*
>
> I: Claro, claro –
>
> *P1: Por mim, por exemplo aí (. . .) a introdução de, no caso da empresa, nesse caso não, mas e outras podia fazer outro tipo de coisa, mas nos pressupostos que vocês teem sim.*
>
> I: Sim, sim, sim. Não, mas esses comentários também são importantes para nós porque também, para nós sabermos que há sempre espaço para melhorar.

I: Em termos de relação entre clientes. Acha que o negócio é refletido de forma precisa?

> *P1: O que é que queres dizer com relação entre clientes?*
>
> I: Eu posso mostrar. Temos uma tabela link que vai relacionar dois clientes. Eu posso mostrar, é mais fácil, sem mostrar não, ok. Ok portanto, ali em baixo temos uma customer, uma interrelação que vai pegar em dois customers, vai haver uma chave extra, que vai ser a tal titularidade, vai ter os atributos descritivos, da granularidade dum cliente com o outro e –
>
> *P1: Sim, sim, sim, sim (. . .) sim, está bem, sim, sim. A resposta é sim.*

I: Agora a questão é no geral, se considera que o modelo reflete o negócio de forma precisa, considerando os conceitos de negócios apresentados, em cumprimento com os conceitos de Data Vault 2.0?

> *P1: Lá está, a minha questão é só com o código de empresa não estar aqui dentro do modelo, mas não vos tendo sido passado de alguma forma esse, esse, esse, esse, a importância desse dado, digamos que sim, eu considero que sim, dentro daquilo que vocês apresentaram considero que sim.*

I: Considerando então, apenas os conceitos de negócio modelados e aquilo que nós apresentámos, como é que classificaria o modelo em termos de completude?

> *P1: Em termos de completude, isso é um bocado, isso é um bocado (...) tás a ver, eu dentro daquilo que vocês mostraram, fazia-me falta ter os subprodutos, mas vocês não vão a esse nível de detalhe, fazia-me falta ter a empresa, mas vocês não teem esse nível de detalhe. Se me perguntas, a completude do modelo apresentado, pah, no abstrato eu diria que falta aí muita coisa base, que tá associada à, aos próprios modelos que vocês apresentaram, agora se não vos foi dada essa informação, se vocês não teem acesso, à totalidade da estrutura do banco, etc, penso que deve ter havido aí algumas limitações, eu teria que dizer que sim, que tá complicado dentro daquilo que eventualmente vos foi facultado.*

I: Na sua opinião, o modelo é simples de entender e utilizar?

> *P1: Estamos a falar de Data Vault, acho que dentro do Data Vault não há nada simples de utilizar, nem, nem, no caso vocês meteram aí aquela questão daquela bridge. A bridge que é tipo o snapshot não é, vocês meteram isso. Gosto disso de alguma maneira, não é, principalmente eu que tou na qualidade de dados, não é. Tenho que fazer testes sobre os dados que estão a ser disponibilizados, e isso, esse modelo do Data Vault, embora o [confidencial] goste muito dele, e eu acredito que ele em termos de futuro desde que seja bem documentado, desde que hajam ferramentas que simplificam de alguma forma a gestão desse modelo, não é. E a forma como nós vamos buscar os dados desse modelo. Que as ferramentas é importante, já nesse nível de complexidade, desde que isso exista pah, eu diria que o modelo que vocês apresentaram é simples qb (que baste) e (...) considerando que tamos a falar de Data Vault, não é. Considerando que estamos a falar de Data Vault acho que sim. O modelo que vocês apresentaram é simples.*

I: Como classificaria a robustez do modelo, na sua capacidade de adaptar tanto aos requisitos de negócios apresentados, tanto aos que possam surgir no futuro?

> *P1: Sim, aqueles que vocês apresentaram(...) [atendeu uma chamada] (...) epah aquilo que vocês apresentaram, pareceu-me que sim, não é. Agora, a novos outros requisitos, epah isso já é entrarmos na bola de cristal, não é. Depende do requisito que surja agora no futuro. Parece-me haver uma certa margem de adaptabilidade e exponencialidade do modelo, agora, tar aqui a fazer previsões no futuro, e se vai-se adaptar ou não, não sei. Agora no presente sim, e aqueles casos que vocês apresentaram, pareceu-me, pareceu-me bem.*

I: Na sua opinião, o modelo representa uma boa prova de conceito para uma futura implementação?

> *P1: Sim, prova de conceito sim, falta aí muita coisa como é evidente, como já vos disse, mas sim, aquilo que vocês reforçaram teem umas ideias giras. Até porque nós já tivemos aí, nós, quer dizer, a parte que tá aí com os modelos de arquitetura, etc. Já teve umas*

abordagens e a coisa não foi assim muito produtiva, portanto apresentaram aí. Portanto, o que vocês apresentaram gostei de ver, parece uma coisa simples, não fui minuciosamente ver as relações que vocês teem, etc, isso já levava muito mais tempo, mas daquilo que olhei pareceu-me tudo bem.

I: De que forma considera a existência do modelo proposto pertinente e/ou importante?

P1: O modelo proposto por quem, por vocês ou –

I: Sim, o que estivemos a apresentar, o que nós apresentamos de que forma é que ele pode ser pertinente ou importante, dentro do contexto do banco?

P1: É assim, dentro dos meus conhecimentos, que são assim [partes?] relativamente a Data Vault. Partes, quero dizer que conheço alguns mas não conheço na profundidade, que vocês com certeza ficaram a conhecer (...) O modelo tem relevância, parece-me que há aí soluções que podem ser aproveitadas até para outras coisas e mesmo a nível de objetos. Vocês acho que exploraram bem os objetos do modelo Data Vault, pah, portanto, há aí mais valias aí sem dúvida, para aplicar até a outras situações.

I: Na sua opinião, o modelo proposto pode ser útil para arquitetos, engenheiros, e analistas de dados da organização?

P1: Se nós adotarmos o modelo do Data Vault, sim com certeza.

I: Que recomendações/sugestões daria para melhorar o modelo?

I: Sendo que, já deu algumas, mas pronto.

P1: Sim, epah, tirando aquilo que eu disse, é como vos digo, tirando aquilo que para mim foi mais impactante naquilo que vi, tudo o resto já carecia de uma análise muito mais profunda, e sinceramente eu não a fiz. E portanto, eu não vos posso dar esse contributo, dessa forma não vou tar aqui a inventar coisas para vos tar a dizer.

I: Que outros comentários pode fornecer sobre o modelo proposto?

I: Se tiver algo a dizer –

P1: Epah não, não. Aquilo que tinha a dizer já fui dizendo. Vocês estudaram isso melhor que eu, portanto há coisas que já me foram apresentadas e eu concordo. Vocês de certeza tiveram a ver milhares de soluções (...) portanto, acho que aquilo que vocês apresentaram está bom, relativamente àquilo que vos foi proposto. Não tenho assim mais a acrescentar.

Interviewee: Participant 2
Interview Date: 26/06/2023

I: Em termos de beneficiários de contas-cartão e as relações que estão associadas a isso, considera que o modelo reflete o negócio de forma precisa?

P2: Sim, né? Eu precisava ver o modelo agora sim, se pudesse me mostrar, para dar uma olhada.

I: Sim, claro, claro (. . .).

P2: Parte dos beneficiários, né? Pronto, então temos aqui, vamos pegar pelo hub principal, que foi aquela parte complexa. Temos o cliente, o cliente pode ser um beneficiário, mas um beneficiário (. . .) nem todos os beneficiários são clientes, né? Aqui o meu colega (eliminação de conteúdo sensível) até comentou, eu não sei se é o caso, eu não conheço profundamente essa área de cartões, mas uma empresa (. . .) ela pode ter um cartão, né? Um cartão da empresa, mas que é utilizada por exemplo por vários sócios, então sim, são vários beneficiários, né, mas o cartão provavelmente é da empresa. O cliente aqui seria um cliente empresa e os beneficiários não, né? Seria um beneficiário, mas eu não sei exatamente como funciona em termos de negócio, mas sim, do meu ponto de vista, em termos de desenho, tá correto, que é, um cliente é- pode ser sempre um beneficiário, agora um beneficiário nem sempre será um cliente, não é?

I: Daí este número de beneficiário

P2: Exatamente. Certo. Depois temos aqui em cima a conta corrente, que tá ligada ao cliente. Temos aqui a parte toda, conta corrente, o hub de conta corrente e temos o hub do beneficiário com a conta corrente, certo, que é esse link, tá bem. E vai dar no de cima (. . .) Pode subir um pouquinho mais para eu ver? Certo. E é o. . . a conta com os beneficiários da conta corrente. Certo.

I: Mm, mm. Sim, é basicamente. . . é aquela tabela auxiliar dos beneficiários.

P2: A (eliminação de conteúdo sensível), sim.

I: Sim, essa aí já. . . esses atributos já têm todos a relação com uma certa conta corrente e portanto pusemos como atributos deste link, porque esta tabela já tem tudo. Tem beneficiários, tem a conta-cartão e tem a conta corrente.

P2: Tá-me aqui a fazer confusão é que um beneficiário, não sendo um cliente, não tem conta corrente, certo?

I: Certo.

P2: É o cliente quem tem conta corrente, né?

I: Sim, sim.

P2: Temos que ter esse cuidado

I: E o cliente tá ligado aqui por este link.

P2: Certo, então (...)

I: Aqui é só porque a nível da fonte de dados e dos metadados que nós vimos (...)

P2: Sim.

I: Onde há a tal chave estrangeira da conta DO-

P2: Então ele não é beneficiário da conta corrente, ele é beneficiário da conta-cartão.

I: Sim, sim, sempre da conta-cartão. É só porque a relação com a DO é feita nesta tabela que também tem informação dos beneficiários.

P2: Ah, certo! Tá bem.

I: Nós decidimos manter assim só porque tamos a representar o negócio como ele existe agora-

P2: Certo. Tá bem. Certo. Tem os beneficiários do hub beneficiários e tem o satélite dos beneficiários conta corrente, que se ligam com a conta corrente, né? Certo. Percebi. Tá bem, tá bem. É porque eu tou ainda aqui com o desenho do Data Vault ainda me... familiarizando. Percebi. Temos o satélite só do beneficiário e temos o satélite do link beneficiários com (...)

I: Conta corrente.

P2: Conta corrente.

I: Sim, sim, sim. Exatamente.

P2: Com o hub de conta corrente. Que é esse que eu tenho um pouquinho de dúvida. Certo.

I: Mas sim, mas também, pode fazer comentários que não sejam tão positivos (riu-se).

P2: Sim, é que essa questão aqui ainda tá me fazendo espécie, certo? Porque, como disse, o beneficiário não me parece (...) que essa parte de cima exista. Que é (...) eu tenho (...) Esse link L beneficiários sim, é beneficiário com customer, certo? E se esse... esse satélite que tá lá em cima, ele pra mim tinha que tar aqui nos beneficiários, no L beneficiários, e não no beneficiários current account, percebe? Porque eu não tenho link dos beneficiários com conta corrente. Percebe?

I: Sim, sim, sim, nós-

P2: Eles têm é- O beneficiário é um cliente também, então ele tá nesse link customer conta corrente, current account, mas não há uma relação do beneficiário, da business key *beneficiário, com a conta corrente.*

I: Ok-

P2: Percebe? Temos que ter esse cuidado, eu não tive na análise da (eliminação de conteúdo sensível) e nem conheço bem essa área, mas tá me fazendo espécie, porque se eu posso ter um beneficiário (que) não é cliente, como é que existe esse hub L beneficiários current account, percebe?

I: Nós também nos fez um bocado de espécie, nós também não percebemos muito bem, mas a verdade é que era essa tabela que tinha como chave o número de beneficiário-

P2: Certo.

I: O número da conta-cartão-

P2: E o- Pois é.

I: Essa tabela é que tinha a conta DO.

P2: Certo.

I: Era essa.

P2: Tou percebendo.

I: Então pronto-

P2: Que foi a bendita (eliminação de conteúdo sensível). Percebo.

I: Sim.

P2: Percebo.

I: Embora não faça muita... muito sentido, era como estava.

P2: Certo. Tá bem, é isso mesmo.

I: Mas pronto, eu percebo, eu percebo.

P2: Percebe né? Como é que (...) Então é porque eu não respondi (...) Na verdade a preocupação que eu tava quando comecei a entrevista com vocês era exatamente porque nós nunca chegámos a confirmar se todos os beneficiários eram clientes, percebe? Porque é como você- Tá certa. A (eliminação de conteúdo sensível) é onde tem o trio, e quem é o trio? O trio é a conta DO, a conta-cartão e o beneficiário.

I: Mm, mm. Exatamente.

P2: E é estranho, é isso que é estranho, ele podia estar aí (...) Nós temos que ver mas *é (segmento de texto incompreensível) É se esse beneficiário é o cliente. O beneficiário cliente.* I: Sim, sim, sim. Até pode ser.

P2: Ou então, como eu não cheguei a responder a vocês, que é, todos os beneficiários são clientes? São obrigados a ser clientes? Percebe?

I: Pois, até é possível. Pois.

P2: Pronto, se depois quiserem ainda resolver essa dúvida a gente depois fala, mas sim, já percebi, tá ótimo. É isso mesmo!

I: Pronto, então em termos dest-

P2: Sim, sim. Indo de acordo com as nossas fontes de dados, sim.

I: Exato.

P2: Certo.

I: Ok, ok, pronto. Depois, em termos de titularidade de contas, no geral, considera que o modelo reflete o negócio de forma precisa?

P2: Sim, esse não há dúvidas.

I: Ok, ok (riu-se).

P2: Esse não há dúvidas né, porque esse não tem nenhum problema. Há um cliente que tem uma titularidade e se relaciona com a conta, com essa titularidade. Perfeito.

I: E neste caso pode haver vários tipos.

P2: Exatamente.

I: Ok, em termos da relação entre os clientes, considera que o modelo reflete o negócio de forma precisa?

P2: Também perfeito. Que é uma relação de um cliente com outro cliente. E aí temos tutores, temos responsáveis, temos vários (. . .) Ou mesmo numa conta empresa, um cliente é conta empresa, qual é a função dele? É administrador? É sócio-gerente? É gerente? (. . .) Tá ótimo. É (segmento de texto incompreensível)? Tá ótimo.

I: Agora, de uma forma geral, tendo em conta os conceitos de Data Vault, acha que os conceitos de negócio modelados foram representados de forma precisa?

P2: Pode repetir? Desculpa. Pode repetir?

I: É aquela pergunta, só de uma forma geral, se o modelo de forma geral representa o negócio de forma precisa.

> *P2: Sim, eu acho que-*

> I: Respeitando as práticas de Data Vault.

> *P2: Sim, com perfeição. Principalmente na distribuição dos satélites, porque um dos maiores problemas que nós temos no Data Warehouse é tratarmos as tabelas com a democracia que não se aplica, ou seja, eu trato tabelas sempre com a mesma importância quando elas não têm. Elas têm importâncias distintas. E trato também... trato todas as tabelas com o mesmo grau de mudança, ou seja, de guardar histórico, e trato todos os campos com essa mesma importância, quando na verdade o Data Vault tem aqui alguma ênfase em... o que é que eu utilizo mais e o que é que eu utilizo menos... pra eu separar, pra eu não tar precisando taxar tempo e fazendo atualização de tabelas que não são necessárias. Eu acho ótimo, tá muito bom.*

I: Então, considerando apenas os conceitos de negócio que foram modelados, como é que classificaria o modelo em termos de completude?

> *P2: Sim, então vamos falar um pouquinho sobre isso. Nós colocámos clientes e nós colocámos a conta corrente, que nós chamamos costumeiramente a conta depósito à ordem, colocámos os cartões, principalmente cartão de crédito (segmento de texto incompreensível). Depois colocaram também a conta-os empréstimos, não é? E colocaram (...) Ou seja, dentro do cofre, nós colocamos a parte que entra, que é a parte de depósitos, depósitos à ordem e depósitos a prazo. Em termos de saída, nós colocamos empréstimos e os cartões. Eu acho que sim. Duas entradas e duas saídas e uma conta que administra essas entradas e saídas. Eu acho que tá ótimo.*

I: Ok. Na sua opinião, o modelo é simples de entender e utilizar?

> *P2: Para mim sim, eu (...) e até quando escolhi o meu curso era porque gostava muito de matemática e (...) mesmo quando fiz a faculdade, a parte que mexeu mais comigo foi a parte de base de dados, porque pra mim era quase teoria dos conjuntos e pra mim era muito fácil ver bases de dados. Quando veio agora o Data Vault separando esses conjuntos com essas características, para mim ficou até mais simples do que os outros relacionais, porque essa característica de separar chaves para um lado, atributos para o outro e relações para o outro, essa visão de conjunto matemática pra mim fica muito simples.*

I: Agora, como classificaria a robustez do modelo quanto à sua capacidade de se adaptar tanto aos requisitos de negócio apresentados, quanto aos que possam ser apresentados no futuro?

> *P2: Também eu acho, do que eu li, essa é a mais valia do Data Vault. Tem que lembrar sempre, e eu sou apologista disso, e defendo essa ideia, que é (...) parece que há duas correntes muito distintas no que diz respeito a guardar dados e utilizar dados que é (...) Para mim o Data Warehouse, ele tem uma linha que divide muito bem que é guardar os dados, e outra coisa é utilizar esses dados, e para mim essa linha, ela separa muito bem a*

forma de eu guardar e depois a forma de eu utilizar. Para mim, o Data Vault, do que eu li e do que eu conheço das outras modelagens, ele parece-me, e eu não tenho a experiência, ser a melhor forma de guardar. É mais difícil na hora de tirar? Pode ser. Mas hoje há tantas outras ferramentas para tirar dados e se desenvolveram muito mais ferramentas de extrair dados, de manipular dados, do que metodologias de guardar e metodologias de desenhar os dados para serem guardados. Então sim, para mim tá robusto e faz todo o sentido para guardar os dados de uma companhia, de uma empresa. Para extrair não, vamos para outros modelos. Mas para guardar, eu acho que é extremamente flexível e robusto.

I: Agora algumas perguntas mais gerais. De que forma é que considera a existência do modelo proposto pertinente e/ou importante?

P2: A minha dúvida maior sobre o Data Vault é que (. . .) normalmente nós já temos um Data Warehouse e custa as pessoas, principalmente pela multiplicidade das tabelas né, pela dimensão que as pessoas pensam em tabelas, mas eu acho que se fosse me dito hoje que eu tinha que criar um Data Warehouse novo para uma empresa, eu não teria menor dúvida que escolheria o Data Vault nesse sentido, por ser o mais flexível e o mais simples matematicamente de desenhar. É o que eu acho. E eu acho que ele não é mais utilizado, porque quem já tem um Data Warehouse tem dentro de um modelo, ou de um modelo Inmon ou de um modelo Kimball, e aí você fazer migração de modelos eu nunca vi e aqui tá quase acontecendo. Mas você criar um novo para mim fazia todo o sentido que fosse no Data Vault. Não sei se respondi bem à pergunta.

I: Era mais no sentido de ser importante (. . .) Qual era a importância ou pertinência se isto fosse implementado ou uma versão mais completa disto.

P2: Ah, sim. Novamente para mim, eu acho que da experiência que eu tenho com outros modelos, principalmente com o Inmon, eu penso que esse modelo seja muito mais rápido de implementar.

I: Ok, pronto, é a vantagem maior.

P2: É.

I: Na sua opinião, o modelo proposto pode ser útil para arquitetos, engenheiros, analistas de dados da organização?

P2: Para arquitetos (. . .) Eu não sei qual é a nomenclatura correta hoje utilizada. É para ser um modelar de repositório de dados para o uso analítico. Porque hoje os nomes tão mudando, Data Lake, Data Warehouse, Lakehouse, etc. Eu acho é que a profissão que é o modelador do repositório de uma companhia, esse modelo é extremamente importante. Ou seja, qual é o nome que nós tamos dando se é engenheiro de dados, se é arquiteto de dados, se é o modelador de dados do Data Warehouse ou do repositório de Warehouse... Para os demais, eu acho que vai demorar um pouquinho, que é, para quem faz análises de *dados, eles vão- Vai custar a perceber, vai custar a entender como é que nós distribuímos em tantas tabelas. Porque ele vai achar que picou ainda mais né, que normalizou demais, mas eu acho que é tudo também uma questão de hábito, porque nos primórdios, quando eu comecei a trabalhar quase nem havia modelagem relacional, era quase tudo muito flat, né, era flat files. É uma questão de hábito. Agora, para um modelador, a profissão que se identifica como sendo a pessoa que vai desenhar o modelo para conter toda a informação de uma empresa, com histórico, ele tem que entender hoje disso, tem que conhecer o Data Vault.*

I: Então acha que é mais útil para esses arquitetos de dados e pessoas de modela-

P2: Para os arquitetos de dados.

I: . . . do que propriamente para os analistas?

P2: É, não, um analista ele vai. . . Quer dizer, hoje nem tanto porque nós tamos também voltando tanto. . . O Data Lake agora é tudo tão distribuído que eu acho que eles não vão sentir tanto mas eu acho que o Data Vault é direcionado para modeladores de repositórios de dados, que são utilizados para tudo, para BI, para IA, para tudo o que for analytics. Porque aqui você consegue guardar rapidamente, se adaptar rapidamente, entrando ESGs, entrando o RGPD, entrando qualquer outro tipo de característica. Nos modelos atuais que nós temos, os Inmons e os Kimballs, o impacto dessas mudanças é muito sentido e acho que aqui no Data Vault ele entra normal. . . não tem impacto. Como ele é mais rápido, no meu entender, você rapidamente disponibiliza rápido para os analistas de dados, porque nos Inmons da vida e nos Kimballs você demora para ter dados prontos para entregar aos analistas de dados porque você vai meter no modelo, vai ter que entender toda a lógica.

I: Que recomendações/sugestões daria para melhorar o modelo?

P2: Pronto, ah, como falamos da profissão, para fazer um modelo, nós temos de conhecer o negócio, né, então, a sugestão é que seja sempre muito acompanhado, o modelador seja acompanhado dos responsáveis técnicos das aplicações. Eu acho que, sendo para mim o Data Vault um modelo muito matemático, de fácil distribuição das tabelas até lendo para as fontes e distribuindo, mas eu acho que o ideal é sempre muito bom, muito bom, em qualquer modelagem que seja, indiferente de ser Data Vault ou não, é bom você conhecer o negócio. Porque essas dificuldades que eu penso que nós tivemos aqui veio muito da falta do meu próprio conhecimento do negócio, né, de ajudá-los. Então, é sempre bom que fique na cabeça de todos, eles e o modelador, ele tar muito próximo das fontes de dados e das pessoas responsáveis por essas fontes de dados. Se forem pessoas com um bom conhecimento de bases de dados, mesmo que seja relacional, ela pode identificar inclusive os problemas que ela teve enquanto base de dados relacional, como nós tivemos aqui essa questão do beneficiário, ou mesmo a questão do cartão. O cartão tá ligado a uma conta só ou tenho vários cartões para uma conta de cartão, uma conta DO? Então a recomendação é que, sendo um arquiteto de dados, ele tem que tar sempre muito aberto *a ouvir os responsáveis pelas fontes. Não pode ser também essa matemática tão simples, olha, chave para um lado, atributo para o outro e relações para o outro. Não, porque, eu acho que a otimização do modelo, ela á muito ligada também ao conhecimento do negócio, então... arquitetos... sempre... Porque você modelar um banco é uma coisa, você modelar uma seguradora é outra coisa, você modelar uma empresa de telecomunicações é outra coisa... Então, eu acho que o Data Vault é muito democrático nessa parte matemática, isso é maravilhoso, é o que também traz muita rapidez, mas eu acho que, como... Porque uma coisa é você modelar uma aplicação. Quando você modela uma aplicação, por exemplo, se vocês, imagina, nós pegamos aqui cartão de crédito, você vai modelar cartão de crédito, só cartão de crédito. Liga com as outras, tudo bem, mas você tá modelando cartão de crédito, você se especializa em cartão de crédito. Quando você é um modelador de um repositório de dados, que antigamente se chamava Data Warehouse, você é um modelador de um tipo de indústria. Você não é um modelador de empréstimos, você não é um modelador de depósitos a prazo, você é um modelador de banca, você é um modelador de telecomunicações. Então, é ter consciência também de que um arquiteto de dados, um modelador de dados, um engenheiro de dados, de modelagem de repositório de insdústria, mesmo sem ter Data Vault, ele vai ter que escolher uma área para também se especializar, porque se você modelar uma insdústria farmacêutica, né, você vai modelar em Data Vault o repositório de dados de uma insdústria farmacêutica... completamente diferente, apesar do modelo ser matemático, é diferente de você modelar um outro tipo de insdústria. Então a recomendação é, também procurem a área que você se identifique ou... não sei... e se especialize. Porque quando nós vamos ao mercado de repositório de dados, existem já modelos criados para determinadas indústrias. Eu vou buscar um modelo Data Warehouse para a banca, eu vou buscar um modelo Data Warehouse para a indústria farmacêutica, eu vou buscar um modelo Data Warehouse para aviação, percebe? O modelo tá ligado à indústria.*

I: Certo. E há alguma recomendação assim mais específica para o nosso modelo, para o que nós propusemos?

P2: Não, eu acho que fizemos bem e foi como eu disse, eu acho que nós fizemos partes pequenas do banco, mais fizemos, né, fizemos algumas partes mais comuns, que é, a minha conta corrente, depois o meu cartão, que é o que hoje todo o mundo tem, depois um depósito a prazo, que é muito comum hoje as pessoas pensarem na reforma, depois empréstimo, que vocês provavelmente daqui a pouco vão tar a entrar numa conta dessa (riu-se), então assim, eu acho que ficou bem colocado. Começámos pela parte simples... claro que aí nas table reference é um mundo à parte, porque nós usamos duas aí que elas vão para além de table reference, que são os produtos e os balcões, eles fazem parte de outras estruturas, que é a própria estrutura orgânica do banco, os produtos também que é uma dimensão muito maior (...)

I: Há alguma recomendação mais específica?

P2: Não, não, não, porque um banco vende muita, muita, muita, muita coisa que nós não fazemos ideia e eu acho que o vosso modelo está na modelagem do nosso dia-a-dia. Quem é que não tem esses negócios de banco que vocês puseram aí? Nós podíamos dizer que outros tantos, né, cofres... O banco vende um cofre para você guardar os seus tesouros e isso é uma forma e você vai ter que modelar isso. Os cofres, o que é que tem nos co- O que é que tem não, mas a administração que o banco aluga, arrenda para você guardar... Entre outras coisas de ações e muitas, muitas outras coisas, mas são coisas que, por exemplo, eu não tenho, eu nunca utilizei, então ainda seria mais difícil para eu falar sobre esse tipo de negócio. Essas eu acho que tão ótimas.

I: Que outros comentários pode fornecer sobre o modelo proposto?

P2: Não tenho, eu acho que vocês fizeram um ótimo trabalho, principalmente vendo essa parte que vocês colocaram aí de outras características que o Data Vault (...) Ou seja, deixando claro que o Data Vault não deixa nada de fora e inclusive que o criador tem sempre o cuidado de dizer "sim, temos solução para isso, tenham cuidado com a utilização, pode não ser bom para a performance, etc.". Temos as bridges, que é um facilitador entre as duas áreas que eu falei anteriormente que é a área de guardar dados com a área que vai utilizar dados. Eu acho que tá tudo.

Interviewee: Participant 3
Interview Date: 26/06/2023

I: Em termos de beneficiários de contas-cartão, considera que o modelo reflete o negócio de forma precisa?

P3: É uma boa pergunta. Podemos voltar se calhar lá aos beneficiários só para (...)

I: Claro, tendo em conta que o modelo é bastante simples, mas de acordo com o que está apresentado, basicamente é essa a ideia.

P3: Sim.

I: (apresenta o diagrama do modelo) É mais aqui esta parte, diria.

P3: Sim, eu aqui só tenho aquele tema que tu até já tinhas referido, Inês, que seria... o ideal seria termos aqui umas entidades, não é, algum hub de entidades (...) se calhar faria aqui um bocadinho mais de sentido, até para agregarmos todas as entidades que temos aqui no banco, mas para aquilo que vocês têm aqui parece-me tar bem, não tenho aqui nada que acrescentar. Até porque também, a nível de beneficiários, confesso que não é bem aqui a... (riu-se).

I: Tudo bem (riu-se). É de acordo com o conhecimento que tem.

P3: Sim, sim, sim.

I: Em termos da titularidade de contas, aquela questão que nós modelámos com os satélites multi-ativos, considera que o modelo reflete o negócio de forma precisa?

P3: Sim. É assim, conforme estavas a referir, um cliente pode ter n tipos de intervenção numa conta e em diferentes contas, por isso, ao estar a contemplar, recorda-me lá aqui em cima...

I: Era a questão do ownership_type, é uma coluna que faz parte também da chave, para além da chave da relação de cliente com conta.

P3: Mm, mm.

I: Esta chave customer_credit_card_account que passa para aqui já tem o cliente e a conta-cartão e depois, pronto, para além da load_date, que faz sempre parte da chave nos satélites, é mesmo o standard, ainda temos o ownership_type, que assim permite que para cada cliente-conta possa haver vários tipos de titularidade. Se não tivéssemos essa chave, isso não poderia acontecer.

P3: Sim. Mas tu não estavas a falar do tipo de titularidade que vem da... posso estar a fazer confusão... da (eliminação de conteúdo sensível)?

I: Sim, que é da relação cliente-conta, ou seja, neste caso é cliente com conta-cartão, mas também podemos ver da à ordem, mas a ideia é a mesma.

P3: Sim, ou seja, qualquer que seja o tipo de titularidade, tá ali associada a-

I: Sim, exatamente. E a ideia seria, qualquer outra conta que fosse, fosse à ordem ou o que fosse, haveria sempre este satélite.

P3: Esse cenário. No fundo, dá para adaptar a qualquer cenário.

I: Exato.

P3: Sim. É isso. Ok.

I: Em termos e relação entre clientes, considera que o modelo reflete o negócio de forma precisa?

P3: Qual é que é? Re-

I: Esta aqui, customer com customer.

P3: Mm, mm. Depois vocês no satélite têm o quê? A data início e fim de relação e-

I: Sim, para já só temos estes, que é a percentagem de participação, início de relação e fim de relação. Se não forem todos falta um, por aí.

P3: Ah, ok. Sim, a nível de relação, também é muito simples, não é? No fundo, é identificar o tipo de relação que existe entre os dois clientes, sim.

I: E o código de relação até acaba por ter já essa informação de qual é que é a relação.

P3: O que depois poderia ser interessante, mas isto lá está, como vocês têm aquelas tabelas de descodificação, que vocês acabaram por não usar muito, aquelas reference tables, este tipo de código, por exemplo, é uma informação que pode ser descodificada.

I: Sim.

P3: Tudo o que é códigos depois pode ter essa abordagem também.

I: Sim, focámo-nos nos principais, os mais importantes.

P3: Sim. Sim, mas os mais importantes... Até porque quem vai ver depois não sabe o que é que significa aquele código... Se tiver o descritivo do código...

I: Claro. De forma geral, tendo em conta os conceitos de negócio apresentados, considera que o modelo reflete o negócio de forma precisa, cumprindo as regras do Data Vault?

P3: Sim, eu acho que sim. Eu só deixava aqui aquelas notas que falámos aqui durante a apresentação. Aquela situação de multi-empresa que acontece muito, mas que, na minha opinião, deveria existir ali um código empresa, uma descrição da fonte, por exemplo (eliminação de conteúdo sensível). Acho que talvez fizesse sentido existir esse código empresa. E aquilo que me salta mais aqui, que embora não esteja aqui no âmbito, e eu percebo isso, o que me salta aqui mais à vista são estas descodificações que acreditem que depois têm muito impacto em quem vai fazer as análises.

I: Pois.

P3: São as duas coisas que me saltam aqui mais logo assim (...) São esses dois pontos.

I: Considerando então apenas os conceitos de negócio modelados, mais uma vez, como classificaria o modelo em termos de completude?

P3: Mas... como assim? Que classificação (...)

I: Quão completo é que considera o modelo tendo em conta o que nos foi fornecido de informação?

P3: Sim. Eu acho que vocês tiveram aqui muito trabalho e tá bastante completo para a informação que vocês tiveram e para a informação que vos foi fornecida.

I: Na sua opinião, o modelo é simples de entender e utilizar?

P3: É simples de entender, é simples de utilizar e é simples de alterar e modificar alguma coisa que seja necessário de ajustar, também é simples de o fazer.

I: Como classificaria a robustez do modelo, na sua capacidade de se adaptar tanto aos requisitos de negócio apresentados, como a requisitos que possam ser apresentados no futuro?

> *P3: É assim, restringindo só mesmo aqui a este âmbito aqui, eu acho que, seguindo aqui um bocadinho aquilo que eu estava a dizer há pouco, acho que é muito simples de incluir e de alterar aqui qualquer ponto. Nesse sentido, acho que também tá bem conseguido.*

I: Na sua opinião, o modelo representa uma boa prova de conceito para uma futura implementação?

> *P3: Isso é que seria o grande desafio (riu-se). Mas sim, acho que era um excelente ponto de partida para começar assim um projeto.*

I: De que forma considera a existência do modelo proposto pertinente e/ou importante?

> *P3: Mas o que é que vocês pretendem saber exatamente aqui? Como assim?*
>
> I: Como é que poderia ser aplicado ou qual é que seria a importância se fosse implementado dentro da empresa ou de que forma é que poderia ajudar. E mesmo que não seja feita a tal implementação, mas conceitos extraídos que possam ser aplicados...
>
> *P3: É assim, considerando... Eu vou comparar um bocadinho também com o que temos hoje em dia implementado, ok? E, eu conhecendo mais ou menos o que existe hoje e olhando para o vosso modelo aqui, para mim a grande mais valia e a grande importância que isto poderia ter seria mesmo a facilidade com que seria possível efetuar alterações. Vocês podem não ter bem noção mas hoje em dia, cada vez que queremos alterar ou incluir um atributo, aquilo é um processo que ainda é longo e é uma coisa que devia ser muito mais linear do que é hoje em dia e, sem ter visto nada implementado do Data Vault, mas olhando aqui para o modelo, se isto funcionasse como eu acho que funcionaria, de facto ia ser uma mais valia e... tanto que existia até o objetivo aqui inicialmente no () ou há uns tempos atrás mudarmos aqui um bocadinho também para o Data Vault exatamente por vermos essas mais valias aqui neste tipo de modelação.*

I: Na sua opinião o modelo proposto pode ser útil para arquitetos, engenheiros, analistas de dados da organização?

> *P3: Sim, isso pode ser de certeza absoluta. Muito útil, a sério. Acho que vocês fizeram aqui um excelente trabalho.*

I: Obrigada. Últimas duas perguntas. Que recomendações ou sugestões daria para melhorar o modelo?

> *P3: Eu acho que isso já respondi mais ou menos, não é? Eu continuo sem perceber, olhando para estes aqui, que a Inês me disse que estavam aqui estes dois separados aqui em baixo, mas continuo com alguma dúvida como é que isto depois se podia integrar no- É que não faz sentido estarem ali duas tabelas soltas ali. Aquilo de alguma forma tinha de ser ali interligado, para conseguirmos fazer a análise depois dessa informação. E no*

fundo as melhorias, foram aquelas que já referi, assim de repente não me recordo de mais nada.

I: Que outros comentários pode fornecer sobre o modelo proposto? Se existirem.

P3: Não tenho assim mais comentários.

Interviewee: Participant 4
Interview Date: 26/06/2023

I: Em termos da parte dos beneficiários de contas cartão e relações associadas, considera que o modelo representa o negócio de forma precisa.

P4: Era aquilo que eu dizia, o que vocês fizeram tenho de estudar, tenho de estudar isto realmente, porque aqui a parte em que vocês fazem a ligação entre conta, conta do, conta cartão e beneficiário, parece tar muito complicado, e parece que é mais simples pelo que eu vi nos dados. Não consigo responder a dizer que está, eventualmente estará, mas acho que pode ser simplificado.

I: Em termos de titularidade de contas, ou seja, aquela questão dos satélites multi-ativos que nós explicamos para guardar a titularidade do cliente com uma conta, considera que o modelo reflete o negócio de forma precisa?

P4: Pareceu-me que sim, digo pareceu-me porque não tou, não tamos a ver os dados.

I: Se quiser podemos ir mostrando o modelo.

P4: Vocês não fizeram, não fizeram, não implementaram pois não, é tudo muito pela teórica?

I: Não, infelizmente não tivemos acesso ao Data Box que era suposto (. . .)

P4: Mas pronto, pareceu-me que estava ok.

I: Em termos da relação entre clientes, considera que o modelo reflete o negócio de forma precisa? Portanto, é a questão do cliente com cliente.

P4: Em termos de relação com clientes (. . .)

I: - Entre dois clientes. Eu posso partilhar outra vez se for preciso.

P4: Relação dos clientes (. . .). Para o mesmo cliente (. . .)

I – Era este link aqui basicamente que faz tal a relação entre dois clientes que veem do hub customer, e depois dalhe um código de relação para distinguir a relação entre eles –

> *P4: À sim, sim, essa descodificação dessa relação, pareceu-me porreiro, tava a pensar mais por terem o mesmo numero de contribuinte e terem, e ser o mesmo cliente, aí a relação tava-me a fazer um bocado de confusão. Aqui é para clientes e clientes, com uma relação. Sim, acho que tá ok, acho que é isso mesmo.*

I: De forma geral, tendo em conta os conceitos de negócio apresentados, considera que o modelo reflete o negócio de forma precisa? Considerando o Data Vault?

> *P4: Sim, é nesta parte que tamos assim aqui a ver, relação de clientes com contas, relação de clientes com clientes, relação de (..) das contas com os cartões, parece-me que sim, salvaguardando aquela parte dos beneficiários, é aí que eu tou com dúvidas só nessa parte. O negócio de empréstimos é muito mais complicado do que isto, do que a parte que mostraram certo?*
>
> I: Sim, sim, claro
>
> *P4: Pronto, à bocado estavam a mostrar ali o parte dos empréstimos, se implementarem empréstimos, o empréstimos é uma aplicação, só por si é uma aplicação.*
>
> I: Pois –
>
> *P4: Certo, por isso é que eu estou a dizer, tá aqui assim bastante mais simplificado, é lógico que vocês aqui já estão (. . .) também estou a exagerar um bocado, também porque eu aqui tou dentro de tudo (. . .) mas vocês vão já aí buscar os dados depois do tratamento e carregar as tabelas, que é só isso que teem a fazer mais nada, por isso tá, é isso que tá bem.*

I: Considerando apenas os conceitos de negocio modelados, ou seja tendo em conta o âmbito bastante reduzido va do nosso modelo, como é que classificaria o modelo em termos de completude. Ou seja, quão completo é que é, tendo em conta as circunstancias?

> *P4: Sim, eu acho que está bastante completo, para o que foi apresentado que está completo, é isso que teem de fazer, especialmente o principal acho que é ter os clientes bem definidos, as contas definidas, a relação desses dois e acho que isso aí tá muito completo.*

I: Na sua opinião, o modelo é simples de entender e utilizar?

> *P4: Hmmm, mais ou menos [risos], pronto é assim, eu ainda tenho de estudar para perceber bem, mas é, tem muitas tabelas, e as ligações dos links, muitos links, aquilo tem de se estudar bem para perceber, mas acho (. . .) acho que tenho de estudar, não posso dizer ou agora que sim ou que não.*

I: Como classificaria a robustez do modelo, na sua capacidade de se adaptar tanto aos requisitos demonstrados, tanto a futuros requisitos que possam existir?

> *P4: Se for assim tão simples, tão direto como vocês apresentaram, acho que tá, tá bem, é*

so acrescentar (. . .) um hub e um link, para acrescentar novos, novos dados, parece-me que está (. . .) está robusto, mas isto aqui tem que ser sempre testado não é?

I: Claro, claro, e idealmente seria isso, mas não conseguimos. Ok

I: Na sua opinião, o modelo representa uma boa prova de conceito para uma futura implementação?

P4: Sim representa, acho que sim

I: Hmmm, ok, esta aqui é um bocadinho mais, pronto, subjetiva, mas, de que forma considera a existência do modelo proposto pertinente e/ou importante?

P4: Pah, eu acho que era muito importante e pertinente, mas isto aqui foi importante, inicialmente era suposto acontecer aqui, aqui no [confidencial], e era isso que fazia sentido, vocês iam fazer um protótipo basicamente, iam fazer isso para nós implementarmos uma aplicação bancária inteira, por isso acho que é muito pertinente, e muito importante. E especialmente, porque acho que em termos de performance, isto ia melhorar muito aqui assim o banco, por isso é que eu perguntei essa performance, porque a ideia que eu tinha é que ia ser o principal (. . .) era a performance.

I: Sim, especialmente da parte de guardar os dados, não tanto talvez na extração, mas –

P4: Eu acho que o acesso é mais difícil –

I: Sim, sim

P4: É mais difícil, mas ao mesmo tempo vocês também demonstraram que teem assim, hipóteses, de que, de que esse, de criar tabelas auxiliares para facilitar esses acessos, por isso acho que haverá soluções para muita coisa.

I: Na sua opinião, o modelo proposto pode ser útil para arquitetos, engenheiros e analistas de dados da organização?

P4: Sim, já é uma pergunta assim um bocado, sim acho que sim, desde que se saiba, que consiga aceder aos dados (. . .) sim, não sei, não sei, não sei responder a esta aqui assim.

I: Sim, isto é perguntas um bocado de resposta aberta –

P4: Pois, é que esta aqui assim, para análise de dados, para análise de dados (. . .) não sei, não sei não vou responder, não sabe não responde [risos]

I: Ok, só faltam mais duas. Então

I: Que recomendações ou sugestões daria para melhorar o modelo?

P4: É aquela parte que já falei, vou analisar para ver se consigo fazer assim uma sugestão para melhorar em termos daquela relação de beneficiários e a cartões e contas, mas é, é uma incerteza que tenho neste momento, náo sei se estou com razão em ver algum *problema ou não.*

I: Ok, por último, é mais aberta. Que outros comentários pode fornecer sobre o modelo proposto?

P4: Ui (. . .) Não tenho –

I: Se houverem, sim

P4: [risos] não sei [risos], acho que já não, já não posso acrescentar mais nada, [incompreensível] acho que falámos quase tudo.

I: Sim, também acho que sim. Mas pronto é aquela pergunta final, só para ver se falta alguma coisa.

P4: Não, mas acho que está porreiro, para pronto, acho que faltava mesmo era uma implementação com alguns de dados e para ver se, é muito mais fácil fazer demonstrações dados.

I: Claro, sem dúvida, mas pronto, em falta de melhor.

Interviewee: Participant 5
Interview Date: 27/06/2023

I: Em termos de beneficiários de contas-cartão e relações associadas, considera que o modelo reflete o negócio de forma precisa?

P5: Sim, acho que sim (. . .)

I: Pode elaborar, se quiser.

P5: Sim, sim. Mas é assim, vamos lá ver uma coisa, eu não sou propriamente desta área.

I: Ok, ok.

P5: Eu estou na área de governo, portanto isto para mim é tudo tão novo, como para vocês, não é (riu-se).

I: Sim, acredito. Nós também com a Andreína, também tivemos um bocado de dificuldade porque ela às vezes certas coisas também tinha que ir procurar e tudo mais, portanto. . .

I: Mm, mm. Sim. Então e, em termos de titularidade de contas, considera que o modelo reflete o negócio de forma precisa?

P5: Reflete, reflete, reflete. Conseguem aí obter toda. . . toda a informação de titularidade, de contas DO. É assim, isto é só contas DO de empresas ou (. . .) Falaram do banco *(eliminação de conteúdo sensível)… é particulares e empresas… é só empresas… ?*

I: É tudo (segmento de texto incompreensível)

P5: São todas as contas DO, ok.

I: Sim, sim. A ideia era modelar todo o tipo de clientes.

P5: Ok.

I: Ou seja, todo o tipo de contas também DO… desses clientes.

P5: Ok.

I: Em termos da relação entre os clientes, considera que o modelo reflete o negócio de forma precisa?

P5: Sim, eu acho que o modelo tá de forma precisa em tudo aquilo que vocês abordaram, portanto…

I: De forma geral, tendo em conta os conceitos de negócio apresentados, considera que o modelo reflete o negócio de forma precisa, cumprindo as práticas recomendadas de Data Vault 2.0?

P5: Reflete. Reflete o negócio de forma precisa, em tudo aquilo que vocês abordaram. Claro que há aqui outras questões, mas… não foram abordadas, portanto não…

I: Sim, mas se houver alguma coisa dentro do âmb-

P5: Não, não. Não, dentro do… Acho que tá perfeito.

I: Considerando apenas os conceitos de negócio modelados, como classificaria o modelo em termos de completude?

P5: O que é que vocês querem dizer com classificaria… Bom, mau, médio…

I: Sim, sim, não tem de se-

P5: Excelente…

I: Sim, pode ser.

P5: Eu acho que o modelo tá correto. Tá… tá… tá bastante bom. Tá bastante bom.

I: Na sua opinião o modelo é simples de entender e utilizar?

P5: Ah… Lá está, simples de entender e utilizar (…) médio, vá.

I: Por causa da questão das tabelas?

P5: Sim.

I: Como classificaria a robustez do modelo na sua capacidade de se adaptar tanto aos requisitos de negócio apresentados, quanto aos que possam surgir no futuro?

> *P5: Eu acho que... eu acho que o modelo é cem porcento adaptável. Isto... por aquilo que vocês explicaram, qualquer alteração é fácil de resolver... qualquer... qualquer (...) alteração, qualquer... Não há aqui nada que não seja fácil de se fazer, não é, pelos vistos.*
>
> I: Sim, do que nós mostrámos pelo menos.
>
> *P5: Sim, do que vocês mostraram, logicamente.*

I: Na sua opinião, o modelo representa uma boa prova de conceito para uma futura implementação?

> *P5: Sim, claro que sim. Sim.*

I: De que forma considera a existência do modelo proposto pertinente e/ou importante, dentro do contexto da organização?

> *P5: É muito importante. Devia ser... Devia ser... Daquilo que eu estou a ver, devia ser uma das coisas a ser utilizada.*

I: Na sua opinião, o modelo proposto pode ser útil para arquitetos, engenheiros e analistas de dados da organização?

> *P5: Sim, acho que sim.*

I: Que recomendações ou sugestões daria para melhorar o modelo?

> *P5: É assim... de repente... não tou a ver nada (riu-se) que possa sugerir, porque como vos disse, não é a minha área, mas eu acho que o modelo tá bastante flexível e bastante... É entendível e acho que resolve várias questões da organização.*

I: Que outros comentários pode fornecer sobre o modelo proposto?

> *P5: Nenhum (riu-se).*

Interviewee: Participant 6
Interview Date: 27/06/2023

I: Em termos dos beneficiários de contas-cartão e das relações associadas, considera que o modelo reflete o negócio de forma precisa?

> *P6: É assim, a minha área não é bem esta. Eu tou mais ligada ao data governance mas, do que eu vi da apresentação, pareceu-me que sim.*
>
> I: Sim, pronto, pelo menos focando naquilo que nós falamos do que eram as chaves e tudo

mais.

P6: Mm, mm. Certo.

I: Em termos da titularidade das contas, considera que o modelo reflete o negócio de forma precisa?

P6: Sim, até achei que era uma… uma… uma forma de ultrapassar as questões. Uma forma até inovadora.

I: Em termos da relação entre os clientes, considera que o modelo reflete o negócio de forma precisa, portanto, a relação entre dois clientes?

P6: Vocês trataram só clientes particulares, não é?

I: Não, na verdade, a ideia era que todos fossem carregados na mesma tabela, naquela do hub customer, e depois diferenciamos com os satélites, onde tem os atributos específicos, aí é que temos para empresa, para individuais ou particulares, que é assim que acho que chamam.

P6: É. Nós temos os particulares e empresa. E temos depois outra questão que é o multi-empresa. Nós temos várias empresas, não só o banco, como as seguradoras, como (segmento de texto incompreensível) empresas, pronto várias empresas à volta, e normalmente nos nossos esquemas de dados temos de ter sempre em consideração o multi-empresa e saber exatamente qual é a empresa que tamos a trabalhar… Mas no vosso caso, vocês tão a focar só uma… uma pequena parte, não é.

I: Sim, mas por exemplo, aquela questão que estava agora a perguntar, que é da relação entre clientes…

P6: Mm, mm.

I: A nossa ideia, pelo menos, e lá está, isto está aberto a críticas, era modelar qualquer tipo de relação, ou seja, seja ela entre empresas ou entre uma pessoa e uma empresa, por exemplo… ou entre empregados de uma empresa, do género, ser um gerente do outro… A ideia é que o modelo seja flexível o suficiente para conseguir (…) Nós temos aquele relationship_type_code que faz com que qualquer que seja o tipo de relação, dá para fazer várias relações e até entre os clientes, pronto, a ideia era essa.

P6: Ok.

I: De forma geral, tendo em conta os conceitos de negócio apresentados, considera que o modelo reflete o negócio de forma precisa, cumprindo as práticas recomendadas de Data Vault 2.0?

P6: Sim, sim. De uma maneira geral, sim.

I: Considerando apenas os conceitos de negócio modelados, como classificaria o modelo em termos de completude?

> *P6: Vocês têm uma escala ou. . . ?*
>
> I: Não, é mesmo resposta aberta. . . Opinião geral.
>
> *P6: Então, bom.*

I: Na sua opinião, o modelo é simples de entender e utilizar?

> *P6: Sim, pareceu-me que, da forma como vocês apresentaram e explicaram, que tá relativamente simples.*

I: Como classificaria a robustez do modelo, na sua capacidade de se adaptar tanto aos requisitos de negócio apresentados como futuros, que possam surgir?

> *P6: Bom. Até porque vocês apresentaram naqueles requisitos, aqueles novos requisitos, várias situações e mostravam que se podia adaptar, portanto bom.*

I: Na sua opinião, o modelo representa uma boa prova de conceito para uma futura implementação?

> *P6: Sim.*

I: De que forma considera a existência do modelo proposto pertinente e/ou importante, no contexto da organização?

> *P6: Muito importante, porque nós tamos também a dar os primeiros passos (riu-se) em implementação, por isso é muito importante ter este tipo de situações que nos ajudam também a ver como é que se pode implementar, não é. É pedagógico.*

I: Na sua opinião, o modelo proposto pode ser útil para arquitetos, engenheiros e analistas de dados da organização?

> *P6: Sim, sim.*

I: Que recomendações ou sugestões daria para melhorar o modelo?

> *P6: É assim, vocês já. . . já tiveram contacto aqui com várias pessoas que estão até mais habilitadas a dar esse tipo de sugestões. Acho que, tendo em consideração isso, de certeza que já vos deram algumas sugestões nesse sentido. Eu de facto. . . não é exatamente a minha área, portanto não vos sei dizer assim nada concreto, por assim dizer.*

I: Por último, que outros comentários pode fornecer sobre o modelo proposto?

> *P6: Que outros (. . .)*
>
> I: Se houverem.
>
> *P6: Eu não tenho assim nenhum. . . agora. . . (riu-se) presente.*

Interviewee: Participant 7
Interview Date: 27/06/2023

I: Em termos de beneficiários de contas cartão e relações associadas, considero que o modelo reflete o negócio de forma precisa?

> *P7: Sim, a mim parece-me que reflete, e falámos aqui de alguns exemplos. E também reflete outra coisa que talvez no futuro pode vir a ser necessário refletir, que é a dados errados do operacional que se conseguem ainda assim integrar depois no próprio Data Warehouse e marcá-los como errados, isto é. Não impedir a integração desses dados, porque as vezes acontece até de outras fontes, não é. Se pensarmos em consolidar outras fontes de outros bancos do grupo ou de outras empresas do grupo, pode ser relevante. E por acaso estamos a pensar fazê-lo, agora com o [confidencial].*

I: Em termos de titularidade de contas, considera que o modelo reflete o negócio de forma precisa?

> *P7: Reflete*

I: Em termos da relação entre os clientes, considera que o modelo reflete o negócio de forma precisa?

> *P7: Em relação entre clientes, vocês tinham um satélite que tinha (...) tinha desculpam um link que ia e vinha –*
>
> I: Posso mostrar –
>
> *P7: Mas era isso, não era? Aqui em baixo. Isso, exato. Sim, claro.*

I: De forma geral, tendo em conta os conceitos de negócio apresentados, considera que o modelo reflete os conceitos de negócio de forma precisa, cumprindo simultaneamente com as práticas de Data Vault 2.0?

> *P7: Sim, parece-me que sim.*

I: Considerando apenas os conceitos de negócio modelados, como classificaria o modelo em termos de completude?

> *P7: Eu acho que ele tá muito completo, eu acho que ele tá muito completo, acho que abordou todos os temas relevantes e aliás ainda mais, os tricky, aqueles que são menos óbvios acho que modelou bem, acho que tao bem transpostos.*

I: Na sua opinião, o modelo é simples de entender e utilizar?

> *P7: Na minha opinião é. Lamento dizer-vos, mas na minha opinião é. Porque (...) o exemplo que costumo dar é o exemplo da numeração, não é. Os números existem tantos quantos aqueles que nós quisermos, e entre quaisquer dois números infinitamente próximos, existe*

uma infinidade de outros números. E o facto de existir muitos não o torna mais complexo, desde que tenhamos uma regra e uma lógica sempre subjacente na sua utilização. Portanto, para mim utilizar este modelo até para exploração de dados não acho que seja uma coisa extremamente complexa, antes pelo contrário. Ele basicamente o que tem de saber é onde estão os hubs, ou quais são os hubs. E todo o meu caminho é feito a partir dos hubs. Vou dos hubs, tenho os links, percebo como é que estas entidades se relacionam. E o meu modelo é fundamentalmente hubs e links. Os satélites são só o adicional, ou aquilo que necessito de ir buscar para responder a uma questão muito concreta. Mas o modelo é direto, se pensarmos noutro modelo relacional, ou num modelo de data warehouse mais convencional, ou até, se quisermos pensar mesmo nos factos e dimensões aproxima-se um bocadinho do tema factos e dimensões, em que tenho as dimensões da análise e tenho os factos. Também é simples por isso, mas essa é simples também porque tem poucas tabelas. E depois tem outras limitações. Mas sim, por isso.

I: Como classificaria a robustez do modelo, na sua capacidade de se adaptar tanto aos requisitos adaptados, tanto aos que possam aparecer?

P7: O modelo, esse que vocês apresentaram, acho que ficou, acho que ficou evidente que ele é adaptável a novas situações de negócio. Também houve um tema que vocês não trouxeram, mas depois se calhar na vossa discussão podem levar. Eu não sei porque é que vocês me estão a fazer estas perguntas, nem sei como é que elas entram aqui, nem sei depois como é que vocês vão utilizá-las. Se vão meter isto no relatório ou se vão utilizar para se preparar também, para a discussão. Mas (...) há uma situação que pode ser interessante que é, todas as abordagens que vocês apresentaram não implicaram fazer alterações à estrutura do Data Vault, no entanto, o Data Vault também é muito (...) muito, como é que vou dizer, permissivo se quiserem, a alterações da própria estrutura, redesenhos de estrutura (...) sem perder todo o conteúdo que já tinham nas estruturas anteriores.

I: Na sua opinião, o modelo representa uma boa prova de conceito para uma futura implementação?

P7: Sim, eu acho que sim.

I: De que forma considera a existência do modelo proposto pertinente e ou importante, no contexto da organização?

P7: Bom, no nosso contexto acho que ele era super importante que nós tivéssemos adotado esta ideologia, por um motivo muito simples, que é nós estamos num processo de migração e queremos começar a entregar (...) e atualmente e estamos a viver esse problema hoje, ok. Optou-se fazer data warehouses Kimball. E o problema do kimball é que ou se já tem completamente fechado o tema das dimensões todas muito bem definidas e os factos que estão, ou então vai estar constantemente a alterar o modelo, que significa alterações em tudo o que está à frente do modelo. E portanto, não prevê essa ideia evolutiva. Eu acho que uma das grandes vantagens do Data Vault é exatamente essa, eu posso começar por *trabalhar clientes, e até expor modelos dimensionais ou de utilizadores finais, da lógica Kimball, sem stress. Se eu amanhã tiver uma evolução ao negócio, posso fazer o, descartar aquele modelo e recriar um modelo novo dimensional, com base nas minhas novas estruturas de Data Vault. Nós estamos a viver isso neste momento, e estamos a verificar precisamente que estamos com dificuldades a avançar, porque não existe a possibilidade de reaproveitar o quer que seja, enquanto o modelo não estiver fechado. E um exemplo simples, estamos a falar de (...) da recuperação, que ele projeta recuperação, em que temos clientes que estão para a recuperação de créditos, e é preciso ter a informação do cliente. E como é preciso, mas é preciso ser uma pequena parte da informação do cliente, que não tem a componente máxima, não tem nada disso. E, portanto, a dimensão cliente tinha que estar toda completa, segundo o nosso arquiteto de dados, para que possa implementar aquilo sem ter alterações no futuro, ou pelo menos alterações substanciais. E neste caso vai ter, portanto a vossa solução aqui quando disseram: "Ok, vou criar umas tabelas de reference data para aquilo que eu não vou modelar". Era isso que era expectável para um caso como este do projeto de (...) recuperação, que era, ok, vou criar um reference data temporário de clientes, com informação mínima necessária para trabalhar o tema da recuperação, e depois aprofundo o tema da recuperação em termos de modelo. Amanhã posso substituir esta reference data por um modelo adicional –*

I: Certo –

P7: Portanto, eu acho que era, era para mim o modelo ideal, o modelo correto, como vocês imaginam.

I: Considera que o modelo proposto pode ser útil para arquitetos, engenheiros, e analistas de dados da organização?

P7: Sim, pode.

I: Que recomendações ou sugestões daria, para melhorar o modelo?

P7: Não, eu para melhorar o modelo não, acho que não, honestamente não tou a ver. Mesmo analisando o modelo de repente, e conhecendo até razoavelmente o negócio (...) Não sei se faria alguma sugestão de alteração. Acho que os temas mais pertinentes foram endereçados corretamente, por exemplo, um dos temas que foi sempre muito discutido, que é o tema do cliente, e o cliente particular, e o cliente empresa. Aqui é direto, com uma abordagem como vocês fizeram, não é, que é ok termos satélites de empresa, termos satélite de particulares. E, portanto, acho que a abordagem é essa. Outra coisa tambem que o Data Vault nos traz é exatamente esse conforto, que é, eu não preciso de já ter o modelo perfeito antes de começar a implementá-lo. E, portanto, honestamente eu tambem não pensei muito sobre isso, vou pensando á medida que as coisas vão nascendo. Se modelarmos by the book, utilizando as regras de modelação que estão definidas, e quando não existe uma regra que seja aplicável, fazer aquilo que vocês fizeram, que foi pesquisar para ver que outras soluções existem, como o multi-active satellite, e o same-as, e os PITs, *que é os point-in-times, resolvem esses problemas, portanto, é algo que, eu acho que isso é mais uma grande vantagem no fundo de um modelo deste género.*

I: Que outros comentários pode fornecer sobre o modelo proposto?

P7: Já forneci, agora é vocês apanharem daí. Já forneci vários comentários.