Benchmarking Data Warehouse Architectures

A Feature-based Modelling and Evaluation Methodology

A thesis submitted in fulfilment of the requirements for the award of

Doctor of Philosophy (Ph.D.)

Qishan Yang

B.Sc., M.Sc.

Supervised by: Professor Markus Helfert

DUBLIN CITY UNIVERSITY

School of Computing

September 2019

Declaration

I hereby certify that this material, which I now submit for assessment on the programme of study leading to the award of Doctor of Philosophy is entirely my own work, that I have exercised reasonable care to ensure that the work is original, and does not to the best of my knowledge breach any law of copyright, and has not been taken from the work of others save and to the extent that such work has been cited and acknowledged within the text of my work.

Student Name: ..... Student ID: .....

Signature: ..... Date: .....

Acknowledgments

Firstly, I would like to thank my supervisor Professor Markus Helfert for his continuous guidance, invaluable support, patient assistance, great enthusism and responsible commitment to this project. I would also like to thank in particular Associate Professor Mouzhi Ge who supported this research and provided feedback over the past years.

I really want to thank Professor Alan Smeaton and Associate Professor Sobah Abbas for their agreement of being examiners and their precious time on my viva. I truly appreciate the funding supported by the Science Foundation Ireland (SFI) grant SFI/12/RC/2289 to Insight Centre for Data Analytics. INSIGHT is funded under the SFI Research Centres Programme and is co-funded under the European Regional Development Fund.

I also wish to thank the members of the Business Informatics Group (BIG) for their help, support and care (particularly, they frequently asked me how is your thesis going.). I wish to thank Zohreh Pourzolfaghar, Manoj Kesavulu, Viviana Angely Bastidas Melo, Claudia Elizabeth Roessing Rocha, Giovanni Maccani, Iyas Alloush, Plamen Petkov, Artem Bielozorov, Umut Sezen, Esma Mine Yildiz, Gultekin Cakir, Tai Mai, Rehan Iftikhar, Douglas Cirqueira, Priyanka Singh, Stephanie Lewellen, Gabriel Hogan, Heneghan Beatrice, Marco Alfano, Paolo Bolzoni, Owen Foley, Phlim Murnion, etc. In addition, I want to express my gratitude for the support and help from Zhengwei Wang, Liting Zhou, Dongyu Lie, Congcong, Xing, Michael Scriney, Suzanne McCarthy, Marlon Oliveira, Rashmi Gupta, Deirdre McCabe and other colleagues from INSIGHT.

Furthermore, I am very grateful to all the participants at each stage of this project, for the time and efforts that they have committed to this project. Their participation and feedback are extremely significant and beneficial.

I also want to thank my family for their encouragement and patience throughout these years. I would like to express my particular gratitude to my parents Xiaolian Yang and Lanxiu Hu, who made all possible to me and supported my education throughout my life. They have supported and understood me during my Master and PhD without any complain in past 7 years even I only have gone home

twice and stayed with them for 1 month in total. Last but not least, I would like to thank the support and care from my parents-in-law Jingang Wang and Fang Wen, and especially my life and soul mate Mingze Wang who has not complained and shown great patience of awakening at deep nights and many lonely weekends.

Publication List

- **Book Chapter:**

  Yang, Qishan, Ge, Mouzhi and Helfert, Markus (2018). Data Quality Management for Data Integration based on TPC-DI Benchmark. Enterprise Information Systems. Springer

- **Conference:**

  Yang, Qishan, Ge, Mouzhi, and Helfert, Markus (2019). Analysis of Data Warehouse Architectures: Modeling and Classification. In: The International Conference on Enterprise Information System. (ICEIS), 3-5 May, 2019, Heraklion, Crete, Greece. ISBN 978-989-758-372-8

  Yang, Qishan, Ge, Mouzhi and Helfert, Markus (2017). Guidelines of data quality issues for data integration in the context of the TPC-DI benchmark. In: The International Conference on Enterprise Information System (ICEIS), 26-29 Apr, 2017, Porto, Portugal. ISBN 978-989-758-247-9

  Yang, Qishan and Helfert, Markus (2016). Revisiting arguments for a three layered data warehousing architecture in the context of the Hadoop platform. In: The 6th International Conference on Cloud Computing and Services Science (CLOSER), 23-25 Apr 2016, Roma, Italy. ISBN 978-989-758-182-3

  Yang, Qishan and Helfert, Markus (2016). Data quality for web log data using a Hadoop environment. In: 21st International Conference on Information Quality (ICIQ), 22-23 Jun 2016, Ciudad Real, Spain.

Table of Contents

List of Abbreviations

Abbreviation	Full Text
BDWH[1]	Big Data Warehouse
BDWA[2]	Big Data Warehouse Architecture
BI	Business Intelligence
BPMN	Business Process Modelling and Notation
C/CDW	Centralised Data Warehouse
CDWA	Centralised Data Warehouse Architecture
DBMS	Database Management System
DFS	Distributed File System
DL	Data Lake
DLA	Data Lake Architecture
DM	Data Marts
DMB	Data Mart Bus Data Warehouse
DMBA	Data Mart Bus Data Warehouse Architecture
DSRM	Design Science Research Methodology
DWH	Data Warehouse
DWHA	Data Warehouse Architecture
DDL	Data Warehouse with Data Lake
DDLA	Data Warehouse with Data Lake Architecture
DSRM	Design Science Research Methodology
ER	Entity Relationship
ETL	Extraction Transformation Loading
FDWH	Federated Data Warehouse
FDWA	Federated Data Warehouse Architecture
HAS	Hub-and-spoke Data Warehouse
HASA	Hub-and-spoke Data Warehouse Architecture
HDFS	Hadoop Distributed File System
HOLAP	Hybrid Online Analytical Processing
IDM	Independent Data Mart
IDMA	Independent Data Mart Architecture

Abbreviation	Full Text
IS	Information Systems
LR	Literature Review
MOLAP	Multidimensional Online Analytical Processing
MRQ	Main Research Question
OLAP	Online Analytical Processing
OLTP	Online Transaction Processing
OSS	Object Storage Service
PDWH	Parallel Data Warehouse
PDWA	Parallel Data Warehouse Architecture
ROLAP	Relational Online Analytical Processing
SRQ	Sub-research Question
TPC	Transaction Processing Performance Council
TPC-DI	A Benchmark for Data Integration
TPC-DS	A Decision Support Benchmark
TPC-E	An On-Line Transaction Processing Benchmark
UML	Unified Modeling Language
V/VDWH	Virtual Data Warehouse
VDWA	Virtual Data Warehouse Architecture

* The difference between 1 (BDWH: Big Data Warehouse) and 2 (BDWA: Big Data Warehouse Architecture) is that 1 stands for the warehouse only, while 2 stands for the architecture containing this warehouse and other components. This explanation is analogised to other abbreviations such as CDW, CDWA, DMB, DMBA, etc.

List of Tables

List of Figures

Abstract

Benchmarking Data Warehouse Architectures
A Feature-based Modelling and Evaluation Methodology

Qishan Yang

With decades of development, data warehouses architectures (DWHAs) have been extended to a variety of derivatives. They have been built in multiple environments for achieving different organisations' requirements. Data warehouse projects may undergo disparate stages during their life cycle (e.g. design, development, maintenance, melioration, migration and discarding). In each stage, some issues may arise to obstruct operations or new requirements may need to be achieved in order to address new challenges and competition. It is therefore necessary to evaluate DWHAs to obtain insights into these derivatives. However, limited research studies have been conducted on how to explicitly and efficiently evaluate DWHAs. They normally require experts for support during evaluation, which makes the processes involved time-consuming and costly. A diversity of features are retrieved and measured in previous research studies when evaluating DWHAs. As a consequence, the evaluation results may be inconsistent and inadequate, due to much manual intervention under different scopes of DWHAs and features.

This research proposes an explicit and efficient method for DWHA evaluation. This method is aimed at reducing the ambiguity of results and the interpretation from experts. It is based on two methods also proposed in this research which systematically classify representative DWHAs and identify reliable features. These DWHAs and features are frequently and widely concerned in the domain of DWHA research, which are further interpreted and organised to scientifically evaluate the modelled DWHAs. Furthermore, the evaluation method is implemented with these interpreted DWHAs and features. Finally, this method is evaluated in multiple cases and empirical research through industry and experiments. Consequently, this research outputs a methodology with three methods and makes contributions to facilitate academia and industry on feature identification, DWHA modelling and evaluation.

1 Introduction

This chapter presents background information on the concepts and development of data warehouses (DWHs) and data warehouse architectures (DWHAs). Subsequently, some issues, research questions and objectives are identified and explained, which should be addressed in this research and make contributions both in relation to academia and industry. In addition, an overview and a breakdown of the structure of this thesis are provided in the last section.

1.1 Research Background

This section introduces the background of DWHs and DWHAs. Firstly, it discusses the differences between DWHs and operational systems. Furthermore, the concept of DWHs is explained as being important for systems referring to data analysis and decision-making in the context of business activities. Finally, the concept of DWHAs is discussed as an indispensable framework to contain and manage DWHs.

1.1.1 The Relationships between Operational Databases and Data Warehouses

Information is one of the most important assets for organisations. This valuable asset is managed in two ways, namely the operational system or the DWH, and they are placed at where the data is input and output, respectively [118]. The operational system can be built in a database management system (DBMS) and

can provide services on data analysis to some extent, although it serves the purpose of running daily business transactions and managing active interactions on time. Thus, the main purpose of the operational system is swiftly processing transaction activities. It should strike a balance between efficiency in transaction processing and query requirements, although it has a small number of resources to be optimised for analysing services such as OLAP (on-line analytical processing), decision-making and data mining [102]. DWHs are designed for other purposes, and they can be built on DBMSs, but they are implemented for data querying and analysis [176]. In addition, these two systems provide services for different types of users. Users almost always manipulate one record at a time in an operational system. However, in a DWH, users almost never retrieve one row at a time, whose questions often need hundreds or thousands of rows to be retrieved and analysed [118]. Therefore, it is necessary to establish DWHs for managing integrated business data and supporting strategic decisions for an organisation.

1.1.2 Data Warehouses

In an organisation, data can be generated from diverse services and allocated to different departments. When conducting analysis at an organisational level, it is necessary to extract these heterogeneous and multiple data sources, then provide consistent information for different purposes. Thus, a centralised repository is required to maintain and manipulate multiple sourced data with a consistent format. A DWH is the conglomerate of all sub-source systems within the company and that information is always maintained with the dimensional model [119]. A DWH is a subject-oriented, integrated, non-volatile and time-variant collection of data that provides capabilities for decision-making [99, 281]. It has the ability to collect data from different data sources, which provides a single, complete and consistent store of data and makes it available to end users in business activities [58]. With more than 20 years' development and evolution, it has become the core of the modern system for decision-making, extracting and integrating infor-

mation from various and heterogeneous data sources for analysis to enhance the users' knowledge of businesses [73]. This centralised data system governs and maintains data for analysis to meet an organisation's requirements. It normally contains historical, current even real-time data, and the analytic output can indicate business trends and perform further data mining in different fields, such as agriculture [1], social media [93], insurance [119], healthcare [272] and others.

1.1.3 Data Warehouse Development

DWHs are continually developed in order to address diverse requirements and challenges faced at different stages. Before the big data era, DWHs normally addressed size-controlled and structured data [113]. Conflicts arise between increasing data analysis requirements and arduous operational transactions. Initially, operational databases acted as omnipotent systems to provide operational and analytical services, or DWHs were developed in the same place as the operational databases, such as the virtual DWH [58]. This affects the performance and speed of the operational databases, as it retards the running of the business activity. Alternatively, DWHs' functions or locations are gradually separated from operational databases in order to provide efficient query-response abilities with a lower influence from operational business transactions for following systems such as data marts (DMs) [86]. Furthermore, in order to cover enterprise-wide data for advanced analysis, they are designed to extract and integrate data from multiple sources, then save data in centralised repositories [64]. At this stage, DWHs are managing and manipulating structured data extracted from DBMSs. In order to better organise data and less convert data, these DWHs are established on DBMSs (e.g. Oracle and DB2) such as [58, 97, 99], which are referred to as traditional DWHs in this research.

Nowadays, data is growing explosively, and such data is usually referred to as big data with 5V characteristics [55]. In particular, unstructured data occupies a large portion of data volumes. This type of data drives the innovation which

should be desired if advanced or further analysis is necessary, as it contributes 90% of information [123, 182], including social media text, audio, video, sensors, and click stream data, etc. Meanwhile, the traditional DWH systems described above are inefficient and are even unable to scale and handle this exponentially increasing unstructured data [182]. In order to address big data challenges, DWHs have become innovative and have been extended with state-of-the-art technologies, via which big data platforms or concepts (e.g. Hadoop and Hive) are employed to establish DWHs with large-scale and high-speed processing capacities [242].

In addition, as an emerging centralised data repository, the data lake (DL) is used to address big data issues. There are different descriptions and understandings between DLs and DWHs. DL systems are designed and developed to provide solutions for solving big data issues with a loose schema repository for raw data and a compatible interface for further data analysis [79]. A data lake is normally established to manipulate high throughput of unstructured data with large volumes rather than the a DWH which mainly addresses structured data [155]. It has the ability to capture and maintain various types of raw data at a low cost, operate transformations on the data, define the schema at the time when it is needed, perform new types of data processing and conduct ad hoc analysis based on specific cases [48]. It more serves for the most highly skilled analysts to explore data. According to the literature, a DL can not only be applied to manage data solely, but can be associated with a DWH. The DL and the DWH provide a synergy of abilities to provide diversified services and deliver accelerating responses [48]. Hence, they work together to present their competences. This combination enables people to deal with enormous and multiple-source data with different types of formats and granularities.

1.1.4 Data Warehouse Architectures

DWHAs are established synchronously along with the progress of DWHs to support DWHs and synthesise other components together for data analysis. DWHAs have different descriptions and interpretations in the literature. A DWHA is an assembly which is comprised of all components, including a DWH as the main part [195]. The main areas include data acquisition, data storage and information delivery. It is composed of different types of layers, which are identified as having three distinct patterns: the single-layer, two-layer and three-layer [58]. Wrembel [276] asserts that the DWHA is constructed to gather and integrate data extracted from multiple heterogeneous, distributed, and autonomous external data sources, as well as to provide services for advanced analysis. The major components of the architecture include the following: an external data source layer, an extraction-transformation-loading (ETL) layer, a data warehouse layer and an OLAP layer. According to [99, 119], there are two main approaches (top-down and bottom-up) in relation to establishing DWHAs in practice. In the top-down approach, a centralised DWH is developed first, then dependent DMs are built for individual departments. However, in the bottom-up approach, DMs are built first then the DWH is created by combining these DMs.

1.1.5 Data Warehouse Architecture Development

The evolution of DWHAs occurs along with the development of the DWHs' road map, as they support and serve DWHs to achieve goals from an architectural perspective. DWHAs can be built on different types of environments or systems to organise and manipulate data. Initially, DWHAs are based on DBMSs to build DWHs, which manipulate and store limited sizes of structured data [44]. These systems (e.g. Oracle and DB2) have been utilised for a long period of time to establish and maintain DWHs [58, 97], as they enable users to store and retrieve data from multiple tables with complicated join queries. During this time, different types of DWHAs are designed in order to meet frequently changed requirements,

which can be categorised and summarised by investigating them from different perspectives. For instance, DWHAs can be categorised into three different types (the one-layer, two-layer and three-layer) [58]. They can be divided into five different types based on the relationships of the components, including the independent data marts architecture (IDMA), data mart bus architecture (DMBA), hub-and-spoke DWHA (HASA), centralised DWHA (CDWA), and federated DWHA (FDWA) [19].

DWHAs are extended to the big data environment, as large-volume, complex, growing and autonomous data sets need to be manipulated [43, 277]. For instance, Yang and Helfert [279] build a DWHA that is similar to three-layer DWHA but based on a big data platform. Thusoo et al. [241] implement a big DWH on big data platforms having the ability to store and process extremely large data sets on multiple nodes, which can be queried by SQL-like languages. In addition, emerging architectures with DLs are created and built in the big data context. Pasupuleti and Purra [182], meanwhile, point out that there are multiple interpretations of the DLA involving several factors (e.g. the type of data ingestion, structured data storage, the ability to search for data, etc.). These architectures (Big DWHAs and DLAs) cannot be slighted and have been accepted by a number of companies (e.g. IBM, Apple, eBay, LinkedIn, Yahoo, Facebook and PWC) [93, 228, 239]. Therefore, when discussing architectures in big data environments, only the architectures with DWHs or both DWHs and DLs are considered in this research.

1.2 Motivation

As discussed above, DWH projects undergo different situations and should fulfil various requirements. It is necessary to better interpret and understand DWHAs through evaluating their features in relation to choosing a suitable one or improving their services. If evaluation can be simply conducted and evaluated results can be easily understood, these will benefit people to efficiently evaluate DWHAs, even they are not the experts in this domain. However, previous research studies

do not provide an efficient method by which to evaluate DWHAs. This motivates to provide an explicit and efficient method for DWHA evaluation. In order to conduct the evaluation from inward and outward perspectives, this research intends to output a method in which different types of features are interpreted and measured to comprehensively evaluate DWHAs. These features evaluated in this method should be reliable, critical, frequently adopted and widely accepted in the domain of DWHAs. This method is briefly designed as follows. Firstly, DWHAs are modelled to better describe their components and structures. It is beneficial to extract features via observing the modelled DWHAs, then evaluate them through this method. The evaluated results should be generated with less expert support and ambiguity. This research should facilitate academia and industry in relation to DWHA investigations, modelling, evaluation and differentiation from a feature perspective.

1.3 Problem Statement

When discussing DWH projects, there is interest in a wide range of aspects, as various projects are at different stages of their life circle. Hence, it is necessary to discuss various situations and problems of a DWH project under different stages before designing the evaluation method. There are three broad phases identified by reviewing the literature, which are described as follows.

1.3.1 Design and Development

In the phase of designing and developing a DWH project, different organisations may have different requirements and circumstances, which require different architectures to achieve their goals. For example, an organisation with a simple operational system and basic data analysis demands may not need a complicated enterprise-wide DWHA, especially with a limited time and budget. A virtual data warehouse architecture (VDWA) may fulfil its requirements, as it has intrinsic

preponderances and is easy to build [181]. Conversely, an international enterprise with heterogeneous internal and external massive data sources may need a more sophisticated architecture, as an simplified architecture (e.g. VDWA) cannot cover all of its requirements. In this case, an enterprise-wide DWHA (e.g. HASA) should be designed and developed, as it has the ability to gather and centralise data from multiple sources [19]. These situations are discussed in previous research, and they are vital to the success of a DWH project. Laney [126] states that the architecture selection is one of the key factors influencing the success of a DWH project. Strange and Friedman [230] identify the architecture selection decision as one of the five problem areas associated with DWH projects. During the selection, it is essential to evaluate potentially suitable DWHAs in order to systematically measure their differences and characteristics.

1.3.2 Maintenance and Melioration

After a DWHA is established, the emphasis is converted from development to operation. In this phase, some issues may occur to affect the performance of delivering, manipulating or analysing data. For instance, ETL is a set of processes to extract, transform and load data from multiple sources to DWHs. These processes must be finished in a certain period of time before DWHs are available to end users again. Thus, it is necessary to meliorate their execution time [222]. If the full loading strategy is used for refreshing new data, it may inefficiently load all records including old or already loaded records, it is necessary to develop an efficient way for loading to reduce the execution time. The delta/incremental loading can be an option to optimise these processes. This kind of issue is sometime not easy to identify in practice. For instance, several interviews are conducted to discuss issues with IT managers who maintain or operate DWH systems in different departments. Their DWHs take a long time for loading data from operational systems. After evaluation, the full loading strategy is identified for daily data loading, while the delta loading strategy should be leveraged to speed up the refreshment

process. As for a shortage of documentation and some other reasons, these issues cannot be easily realised.

1.3.3 Migration and Discarding

Migration focuses on the aspect of hardware perspectives, which discards previous components or the whole architecture, then upgrades components or migrates to a new architecture and the legacy architecture may be discarded. However, the melioration discussed above is more related to meliorating current DWHA without hardware changes. In the business environment, the requirements are normally changed due to new competition or business services. For example, in the big data era, new platforms and techniques can be leveraged to manage and manipulate different types of data in order to further analyse potential information which is valuable to generate strategies by analysing big data. However, traditional DWHs built in DBMSs are not adept at addressing explosively increased data [182]. If a company's DWH system is built in a traditional environment and it intends to upgrade some components or migrate to a big data DWHA, this situation is similar to the situation mentioned above, but it can use more advanced DWHAs to improve the current system with stronger capacities. This issue is rarely discussed in academia, while it is explored in the industry, where some methods and platforms can be found to address this issue such as [193, 238]. Before migration, it is necessary to identify which components or DWHAs should be selected from candidates, so evaluation should be conducted to select a suitable one.

1.3.4 Discussion

It is necessary to evaluate DWHAs in order to investigate their details and select a suitable one when designing and building a DWH. However, some DWHAs are difficult to distinguish as they have similar structure and provide similar services to end users. For example, CDWA and HASA are very similar from an

architectural perspective, and they are able to extract and integrate data from multiple sources to provide enterprise-wide compatible data for end users. If using modelling languages to describe them, it is easy to identify differences between them and help even individuals without rich background knowledge of DWHAs to understand and make suitable decisions. In addition, when a legacy DWHA has some undetectable issues or cannot fulfil its mission as requirements are increased, it is necessary to evaluate DWHAs to uncover issues and identify how to meliorate or migrate the current architecture with new components, or even new architectures. Some issues (e.g. loading and data flow issues) are not easily or directly recognised from the surface by observing the modelled DWHAs explained above. More complicated and underlying investigations (e.g. evaluating data flow or throughput) should be conducted to discover these issues and further provide solutions to solve them.

The DWHA migration is another strategy by which to meliorate, but involve more thorough improvements from physical or hardware-oriented perspectives. For instance, if the legacy DWHA intends to improve its ability to address unstructured data, the current system may need to migrate to a state-of-the-art platform. In this case, limited improvements can be made by upgrading the system in relation to software aspects. It is easy to identify differences or improvements by modelling architectures and measuring their performances from different perspectives.

DWHAs may need to be outwardly and inwardly evaluated in order to identify issues. Some previous studies have investigated DWHAs based on various features [20, 136, 266]. The problem is that most of the features are chosen based on researchers' perceptions. Different individuals may propose disparate features and inadequate rationales are given for why these features are chosen to evaluate DWHAs, in addition to how they are extracted. Besides, most of these methods require experts during the evaluation, which is time-consuming and exorbitant in this process. The results are subjective or biased as different people may have different understandings of DWHAs. The results proposed in these studies are brief

and abstract without adequate diagrams or concrete information enabling people to better understand them.

1.4 Challenges and Objectives

Following decades of development, a great number of DWHAs with various components and characteristics in a diversity of environments have been developed to match diverse requirements. In addition, as discussed above, even in a DWH project, it may undergo different phases and situations during its life cycle. These DWHAs and different phases challenge and complicate the investigation and evaluation of DWHAs. It is time-consuming to investigate all of them. In order to intuitively describe DWHAs, this research intends to directly present DWHAs by using a modelling language to further facilitate the investigation and evaluation of their features (e.g. components, characteristics, performances, etc.). These modelled DWHAs are input into the method which, subsequently, output the evaluation results. This method should be efficiently executed without much support from experts, and results should be explicitly shown without much ambiguity. Although the outline or road map for solving this problem is explained above, some challenges can be addressed during the conducting of this research, which are summarised as follows:

- How to properly present DWHAs in a consistent way via a modelling language for investigating and evaluating their features;
- How to identify reliable and critical features derived from these DWHAs leveraged in different situations or with different components for different purposes;
- How to investigate DWHAs explicitly and efficiently by modelling them and measuring their features in order to output obvious evaluated results with less expert support and ambiguity.

Therefore, the objectives of this research cover two aspects. Firstly, it will generate a method as a theory to systematically present and evaluate DWHAs by investigating and measuring reliable and critical features. This method should facilitate DWHA evaluation in different phases or situations of DWH projects, which should be executed with less expert support during the evaluation processes and provide less ambiguous results. This method should be examined both from industrial and academic perspectives. It is important to ensure the validity of this method and the reliability of the results generated by this method. In addition, it should be examined and demonstrated in the presence of people from industry and academia who can offer feedback for iteratively upgrading this method. Therefore, the objectives of this research are summarised and listed below:

- Creating a scientific method with abundant theoretical evidences supported to systematically evaluate different DWHAs by measuring reliable and critical features;

- Reducing the need for interpretation from experts and simplifying the processes of conducting the evaluation of DWHAs using this method in real DWH projects;

- Examining the concepts, processes and results of this method and demonstrating them to people from industry and academia in meetings, workshops or conferences;

- Providing clear and less ambiguous results to directly present the DWHAs by graphically modelling their architectures and measuring in detail some of their features together.

1.5 Hypothesis and Research Questions

To address the challenges and achieve the objectives, a hypothesis and a main research question with three sub-research questions are proposed below.

Hypothesis: Data warehouse architectures can be evaluated by feature modelling and measurement to reduce the ambiguity of results and the need for interpretation from experts.

The general processes of this method for how to evaluate DWHAs are designed as described in figure 1.1 below. Firstly, it involves modelling and extracting features from organisational requirements which should be achieved or a DWHA which needs to be evaluated, then inputting into the feature framework concerning reliable and critical features. These features are further divided into sensitive and insensitive features for evaluating this DWHA from different perspectives, finally the evaluated results are output along with some diagrams to profile this DWHA. In terms of the feature framework, the sensitive features are used for measuring the characteristics which can easily be affected, while the insensitive features refer to some components where their changes cannot easily be affected. In other words, the sensitive features focus more on the internal or inward characteristics of a DWHA's performances (e.g. data flow and throughput), while the insensitive features are more related to the external or outward appearances of DWHA components (e.g. ETL and DM).

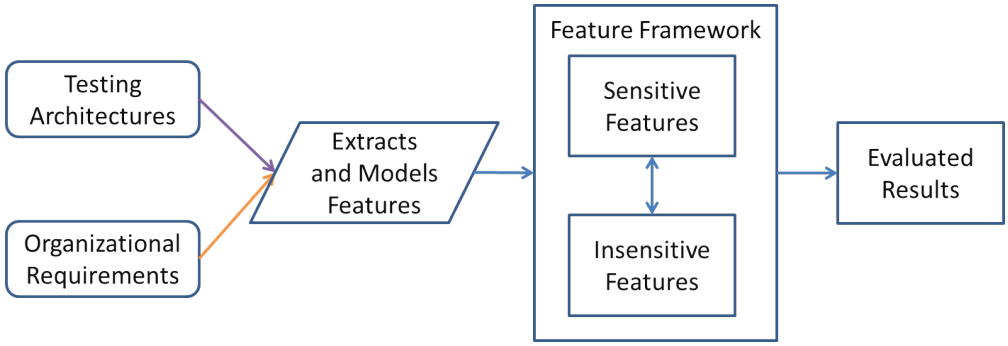


Figure 1.1: The brief processes of the evaluation method

In order to prove the hypothesis, the main research question is proposed below, which synthesises the concerns of the objectives. The method developed for this research should have the ability to address this question.

Main research question (MRQ): What features are reliable and critical to explicitly and efficiently evaluate modelled data warehouse architectures?

As this MRQ is complicated and difficult to answer directly, it is divided into three sub-research questions (SRQs). By researching and answering these SRQs, the MRQ can be answered simultaneously. They are individually presented and explained below. The first SRQ and two aspects are presented below, which focus on investigating different types of DWHAs. As there are a vast number of DWHAs which can be found in the literature, it is necessarily to analyse and generalise them to identify their differences and similarities, then summarise how many types of representative DWHAs should be considered in this research, the details are described in chapter 4.

Sub-research question 1 (SRQ 1): What are representative types of data warehouse architectures and how can these be described?

- How to collect and identify data from multiple sources referring to DWHAs;
- How to systematically describe and identify these representatives from a vast number of DWHAs.

This research is aimed at evaluating DWHAs from a feature perspective, so the DWHA and feature are two critical topics being considered. The first SRQ places emphasis on the DWHA, then the second SRQ focuses on analysing what reliable and critical features should be determined. These features should be frequently adopted and widely accepted in the domain of DWHAs. The method of how to identify these features is explained in chapter 5. They should be also discriminative to present and evaluate DWHAs. As discussed in the following chapter, a diversity of key or typical features are extracted and demonstrated in table 2.2 based on previous research studies. They reflect different aspects, components, functions or performance of DWHAs. For instance, the ETL feature is more related to a component in DWHA to extract data from data source, transform and

load it into DWH systems, while the direct cost or indirect cost focus on financial or business aspects. This research concentrates on the features referring to discriminatively present and measure DWHs from the architectural perspective for evaluation, which is concretely explained in section 6.1.

Sub-research question 2 (SRQ 2): What reliable and critical features can be used to describe and evaluate data warehouse architectures?

- How to derive and distinguish reliable and critical features out of various potential features from miscellaneous DWHAs in different circumstances;
- How to describe and interpret these features for further DWHA description and evaluation.

After representative DWHAs and features are identified as the outputs to answer the first and second sub-research questions, it is necessary to consider how to synthesise and measure them for the purpose of evaluating DWHAs. Hence, the third SRQ is proposed below, which is the main part to design and develop the evaluation method. The details are explained in chapter 6. Besides, the explicit and efficient characteristics of this method should be evaluated, which is conducted in chapter 7.

Sub-research question 3 (SRQ 3): How should these features be organised and measured in order to explicitly and efficiently describe and evaluate data warehouse architectures?

- How to design the method and its processes to describe and measure these features for the DWHA evaluation;
- How to explicitly and efficiently evaluate DWHAs to generate clear results and require less expert support.

1.6 Contributions

This research develops a methodology to explicitly and efficiently evaluate modelled DWHAs by measuring reliable and critical features. The contributions are summarised and listed from both academic and industrial perspectives.

Academic Contributions:

- It proposes a method to extend the knowledge of identifying typical DWHAs.
- It implements a method to contribute the knowledge for generating reliable features and relevant sub-features.
- It develops a feature-based modelling and evaluation method supported by the two methods results to generate theories for systematically benchmarking DWHAs.

Industrial Contributions:

- A DWHA classification model is pro-posed with four levels and the nine typical architectures based on the literature.
- A taxonomy of DWHA features is produced with four levels for different purposes.
- Modelled DWHAs, framework, tables and radar charts are designed to explicitly demonstrate the evaluated DWHAs.

1.7 Thesis Structure

This research is guided by the design science research methodology (DSRM), combined with the concepts and techniques of qualitative and quantitative research methodologies. Hence, the chapters of thesis are organised according to

the main concepts and processes of design science. The remaining contents of the thesis are structured as follows:

Chapter 2 describes literature reviews in relation to DWHs, DWHAs, DWHA features and the methods adopted for evaluating DWHAs, illustrating the detailed background of the progress made in this domain. It provides in-depth analysis of different aspects to explain the current situation and problems faced in the area of DWHA evaluation. Chapter 3 introduces different concepts, theories, techniques and research methodologies which are beneficial to systematically conduct this research, then explains how to select some of them and adapt to this research.

Chapter 4 and chapter 5 investigate DWHAs and features based on the data collected from the literature, which provide materials (e.g. representative DWHAs, reliable and critical features) to generate the method for DWHA evaluation. These two chapters investigate DWHAs and features in depth based on scientific methods and techniques (e.g. systematic literature review, TextRank, Word2Vec, etc.), aiming to address and answer SRQs 1 and 2, respectively.

Subsequently, in chapter 6, the method is designed and developed based on the results generated in chapter 4 and chapter 5. This chapter partly answers SRQ 3 concerning how these reliable and critical features are operated to evaluate DWHAs, focusing on organising these features and enabling them to cooperate. In chapter 7, this method is executed and assessed to examine if it achieves the objectives. The concepts of multiple case studies and empirical research are leveraged to comprehensively test this method in different situations. This chapter aims to answer SRQ 3 regarding the explicitness and efficiency of this method.

Chapter 8 concludes this research and the contributions to prove that this research achieves the objectives and answers the MRQ. It analyses the limitations and challenges of this research, and provides an outlook on future research. In order to present an overview of this research, the following table demonstrates the main focus, the methods and techniques applied, along with the results gained in each chapter associated with the MRQ and SRQs.

Table 1.1: Overview of this research

Chapter	MRQ or SRQs	Content	Output
1	-	Background of DWHs & DWHAs, problem statement and motivation	Objectives and research questions
2	-	Related work on DWHs & DWHAs, features and DWHA evaluation methods	DWHAs in different circumstances, summaries of features and evaluation methods from previous research studies
3	-	Importance of research methodologies in research, investigation of different research methodologies and literature review methods	The DSRM is chosen to underpin this research
4	SRQ 1	Design and development of a method with systematic processes and algorithms for DWHA classification	A systematic classification method and a DWHA classification model with a hierarchical structure

Chapter	MRQ or SRQs	Content	Output
5	SRQ 2	Designing and developing a method with systematic processes and algorithms for identifying reliable and critical features, in addition to generating a taxonomy based on these features	A method to identify features and generate taxonomies, reliable and critical DWHA features, a DWHA feature taxonomy
6	SRQ 3	Designing and developing a DWHA evaluation method with a feature framework	A method with systematic processes to evaluate DWHAs based on modelled features
7	SRQ 3	Assessing this research in multiple case studies and empirical research	Results of assessment
8	MRQ	Discussing contributions made by this research, analysing limitations and the future work of this research	Contributions, limitations and future work

2 Literature Review

For the sake of further clarifying objectives and problems, this chapter reviews related research studies on DWHs, DWHAs and features. There are different interpretations of these concepts found in the literature. By analysing and summarising these theories, some definitions are proposed and used in the present research. In addition, this chapter also investigates the literature for preparation of solving SRQs in following chapters. For answering SRQ 1 in chapter 4, some architecture classification methods in previous research are summarised to identify the current situation for better designing the DWHA classification method. For answering SRQ 2 in chapter 5, the key or typical features are extracted from literature and further analysed the issue. Subsequently, the taxonomy methods and related literature review methods are explained in order to well organise features into a taxonomy. For answering SRQ 3 in chapter 6, some methods relating to the evaluation of DWHAs and other IT architectures in previous research studies are investigated, which assists the identification of current situations and benefit the development of the evaluation method.

2.1 Data Warehouses

DWHs may adopt various roles in different circumstances. For instance, before putting data into DWHs, it may be located in different locations with heterogeneous formats which provide inconsistent information to end users. Thus, an organisation needs a systematic channel to maintain and manipulate its data with

a mutually agreeable format for supporting decision-making. DWHs gather raw data from various sources and provide consistent data for end users [58]. They may extract data from other decision-making systems to extend the ability of providing more data supported services. For instance, Kimball and Ross [118] describe how DWHs extract data from DMs with conformed dimension tables to combine these separated marts. These conformed dimensions enable facts to be organised and presented in the same way across multiple marts to generate a uniform data format at the enterprise-wide level. DWHs may help other systems to achieve advanced or extra services. For instance, Dedi and Stanier [54] note that a DWH is a system used for reporting and data analysis, and it is considered as a core component to provide data for business intelligence (BI) systems. In other words, consistent data should be retrieved from a DWH when the following systems need to conduct data analysis.

2.1.1 Data Warehouse Definition

As discussed above, DWHs play different roles and present diverse capabilities. Thus, they may be understood and defined with different patterns. To further investigate and analyse the definitions proposed in different circumstances, [58, 99] describe a DWH from a data perspective, in which data should be collected and integrated in a consistent format under special characteristics (e.g. subject-oriented, time-variant, etc.). Dedi et al. [54, 118] consider components in a DWH system, unlike previous definitions, DMs are the main components needed to form a DWH and the BI system is one of main components to provide services for end users. Dedi and Stanier [54] place an emphasis on the functions and following systems (e.g. BI) with lesser emphasis on DWHs, whereas these definitions do not adequately describe the DWH development track, and new DWHs have emerged in recent years.

DWHs are continually evolving in order to address a spectrum of requirements and challenges. Initially, DWHs were developed in place of operational systems

such as the virtual DWH [58]. They gradually separated from operational systems to provide efficient query and response abilities with less of an influence from daily operational business transactions for individual departments such as the DM [86]. In order to address data analysis especially at the enterprise level, they are designed to extract and integrate data from different systems, then populate it into DWHs [64]. In the big data era, data is explosively generated and new requirements are required to acquire more valuable information by analysing more data. With these requirements to be achieved and emerging technologies to be applied, DWHs are developed and extended to Big DWHs and DLs with less integrated and formatted data but more flexible and efficient analysis [93, 101]. This research analyses the previous definitions and interpretations, and considers the development track of DWHs from operational database to Big DWHs and DLs which is concretely discussed in 1.1.3. The definition of the DWH is extended and applied in the present research, which is described as follows.

A data warehouse is a system to physically or logically collect, manage and manipulate data providing valuable information for decision-making and data analysis.

2.1.2 Data Warehouse Representations

When developing DWHs, they can be categorised into two main patterns: top-down and bottom-up. These two representations are proposed by Bill Inmon [99] and Ralph Kimball [119], respectively, and they are widely accepted in this domain. These two concepts are frequently used in different projects to design and develop DWHs depending on their requirements and situations. The sections below discuss the details and differences of these concepts.

2.1.2.1 Top-down Data Warehouse

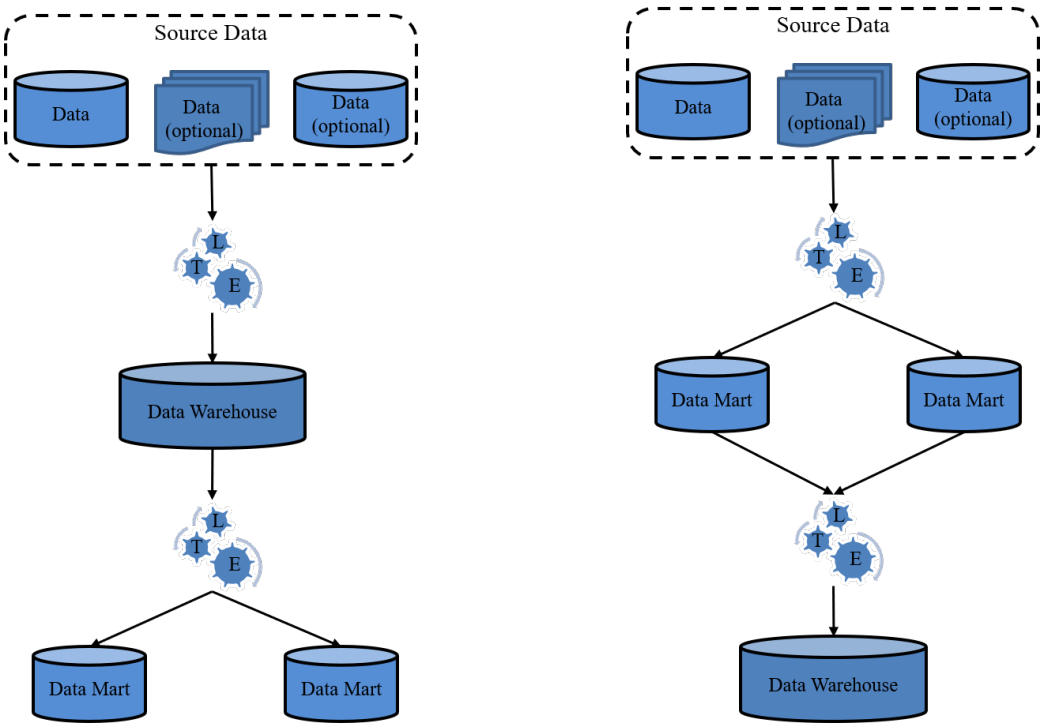
In a top-down DWH, data is extracted from a single source or multiple sources and loaded into a DWH in which data is integrated and pre-processed to a consistent

format. DMs retrieve data from this DWH for different purposes. Therefore, this DWH system maintains data flows from data source to DMs through the DWH and two ETL systems. In this situation, the DWH is firstly designed and developed, then DMs are developed and should retrieve data from this DWH [110]. As most of the data quality issues and pre-processes have already been solved in the first ETL, data in the DWH is cleansed, integrated and consistent. Therefore, the second ETL focuses more on preparing and further formatting data for different DMs, putting less effort on data quality and integration issues. As it is time-consuming to design and develop this DWH, and DMs cannot be queried before the DWH is ready, this representation needs more time to provide data and services for end users [137]. However, the DWH is a hub or centralised repository to provide consistent data to fulfil enterprise-wide requirements.

2.1.2.2 Bottom-up Data Warehouse

In a bottom-up DWH, data is extracted by a ETL system from a single source or multiple sources. The difference is that the extracted data is loaded into the corresponding DMs. After this process, data from these DMs is extracted, transformed and loaded into a DWH. Therefore, in this system, the data flow is opposite compared with the data flow in the top-down DWH. Data flows from data sources to DWH via DMs and ETLs [46]. Thus, DMs are firstly designed, developed and populated to achieve concrete and specific requirements for different departments or ad hoc analysis. The first ETL makes the most effort to cleanse data quality issues and prepare data for DMs, while the second ETL focuses on managing and responding to requests from the DWH. When DMs are built and loaded, they can be queried by end users for data analysis even when the DWH is still unavailable, so this type of DWH can provide services earlier than the top-down DWH [110]. However, in order to prepare enterprise-wide data for the DWH, the data in their dimension tables need to be conformed, which allows data in these DMs to be linked together [119]. In addition, this DWH may need to retrieve data from mul-

tiple DMs for a query, so it may be complicated and time-consuming to provide enterprise-wide data for end users.



(a) The top-down DWH **(b) The bottom-up DWH**

Figure 2.1: Two typical representations of DWHs adapted from [99, 119]

2.2 Data Warehouse Architectures

DWHs need to rely on architectures and other components assisting them to solve data quality issues, manipulate, integrate and transfer data. DWHAs are developed synchronously along with DWHs. The correlational literature shows that the definition of the DWHA has different explanations and interpretations. Ponniah [195] points out three main parts of DWHAs based on different types of data involved. Devlin and Cote [58] classify DWHAs into three sub-architectures based on different data layers. Wrembel [276] describes data in a DWHA and summarises the four layers based on the data flow of a DWHA, which clearly explains the

data and components of the DWHA. There is another concept (data warehousing) which is similar to DWHA. Widom [267] states that data warehousing encompasses architectures, algorithms and tools for bringing together selected data from multiple databases or other information sources into a single repository. As can be observed, the DWHA focuses more on organising components related to a DWH and their relationships. Data warehousing involves architectures but places greater emphasis on the process of building a DWH project. Therefore, this concept is not involved in this research.

2.2.1 Data Warehouse Architecture Definition

The present research intends to investigate and evaluate DWHAs through measuring their features. Besides, it considers the following circumstances. When more data is required, source data may be extracted from internal, or even external, sources. DWHAs may need to manage and address big data, in which data may be unstructured. In order to address the unstructured big data challenges, DWHs can be built with state-of-the-art technologies. For example, the big data platforms or concepts (e.g. Hadoop and Hive) are leveraged to build DWHs with high-speed processing capacities [242]. In terms of the definition of DWHAs, the descriptions proposed by [58, 195, 276] discuss DWHAs from different perspectives. However, previous research studies have not adequately emphasised architectures referring to circumstances discussed above. Therefore, based on the previous interpretations and present intentions, a new definition of the DWHA is proposed and applied in this research, which is described below.

A data warehouse architecture is a principled system with fundamental properties and rational relationships to manipulate, manage and analyse data from internal, and even external, sources with various formats, which may include the source layer, ETL layer, data storage layer and data presentation layer.

2.2.2 Architectures with Different Representations

DWHAs can be represented with different patterns to meet changeable requirements. By analysing DWHAs from different perspectives, they can be summarised and classified into three scenarios described below, which can aid further identification and the ability to determine the scope of the DWHAs being investigated and evaluated in this research.

2.2.2.1 Architectures Based on the Number of Layers

In terms of the layers of DWHAs, they can be divided into three classifications: single-layer DWHA (real-time layer), two-layer DWHA (real-time layer and derived data layer) and three-layer DWHA (real-time layer, reconciled data layer and derived data layer) [58, 97, 219, 279]. The primary mechanism of the single-layer (real-time layer) involves all data sets being stockpiled only once [99]. In other words, an operational system and a DWH system share completely identical data which is stored in the same place. The two-layer architecture adds a derived data layer which enables operational and informational requirements to be fulfilled separately. The three-layer architecture is comprised of a real-time data layer, a reconciled data layer and a derived data layer [97]. It has a reconciled layer as the new layer, compared with the two-layer architecture. This new layer cleanses and integrates operational data sets and saves them together in a consistent format.

2.2.2.2 Architectures Based on Components

When investigating DWHAs from a component perspective, they can be characterised into five predominant architectures: IDM, DMB, HAS, CDW, and FDW [218]. These five architectures are comprised of different types of components which include primary portions to provide basic functions such as ETL, data warehouses, data marts, etc. Although other architectures (e.g. hybrid) are mentioned

in the literature, they tend to be variations on these five [18, 216, 261]. Ariyachandra and Watson [19] conduct a web-based survey involving companies ranging from small (less than $10 million in revenue) to large (in excess of $10 billion). Most of them are located in the United States (60%), covering a variety of industries. Matouk and Owoc [147] conduct a similar survey which collects data from some firms listed in the Warsaw Stock Exchange. The ranks of these DWHAs are the same in both surveys. The HAS occupies the broadest domain, followed by the DMB, then by the CDW, IDM, and finally the FDW.

2.2.2.3 Architectures Based on Big Data

Kimball and Ross [118] point out several limitations of DWHAs built on DBMSs and advocates an alternative in which DWHAs can be built on MapReduce/Hadoop to solve big data issues. Thus, it is necessary to consider big data issues when designing and developing DWHAs in order to fit the demands in the future. Many correlational studies establish DWHs in the context of big data [210, 211, 278, 290]. In addition, from the literature, there are some DWHAs with DLs, which are built on big data platforms to manipulate data for advanced analysis. Based on [48, 101, 182], the relationships between the DWH and DL in these architectures can be summarised as involving two situations. The first is a DL fed by a centralised data warehouse. It can be used in a company which already has a legacy-centralised DWH and attempts to build a DL for the purpose of meeting further requirements. The second is a DWH fed by a DL. This architecture is suitable for optimising the legacy DWH data flows. It can be used to extend the DWH to meet advanced requirements.

2.2.3 Data Warehouse Architecture Classification

As DWHAs have different presentations, in order to systematically investigate their characteristics and structures, it is necessary to identify their similarities and differences, which is beneficial for gaining insight into these DWHAs in relation

to further research. Therefore, the sub-sections below explain current classification situations, different terminologies of Classifications and methods used to classify DWHAs.

2.2.3.1 Data Warehouse Classification Situation

With decades of evolution, DWHAs have been developed and extended to a vast number of derivatives in different circumstances. In order to investigate them, it is necessary to classify them for analysing insight into their relationships and differences. A classification is a fundamental mechanism for structuring knowledge [259]. It can be defined as the action or result of systematically distributing, allocating, or arranging things in a number of distinct classes based on their characteristics or relationships, or it is a category of something belonging to [177]. Although there are some ad-hoc studies on DWHAs investigation and classifications, these studies do not provide adequate reasons or methods on how to classify them. There is still a lack of a systematic method to build up an evidence supported classification with concrete steps of how they are classified. Besides, data is explosively generated in the big data era. More emerging architectures and technologies are established to manipulate and manage data in this domain. It is therefore valuable to systematically revisit and classify previous and emerging architectures under a method.

2.2.3.2 Different Terminologies of Classifications

In order to fully collect and investigate previous classification methods, it is necessary to identify them which may be presented with synonymous terminologies. The concept or theory of classification is applied in a wide spectrum of domains and disciplines in different domains. For this reason, diversified terminologies regarding classification are used in different research studies. For example, the terms “framework”, “typology”, “taxonomy” and “clustering” can be employed in the process or at the results stage of classifying objects [61, 166].

In terms of the framework, it is an approach to understand the research within a body of knowledge through a set of assumptions, concepts, values and practices [217]. It is a synonymous expression of the classification to organise knowledge within a well-structured pattern for better understanding. The typology is the study of analysis or classification regarding types or categories [151]. It is another phraseology of the classification, and can be used interchangeably to refer to the same thing in different ways in a topic, e.g. [180]. The taxonomy can arrange concepts with regard to relationships in order for them to be clearly presented and applied by researchers [148]. It can facilitate the process of mastering the science behind design principles of observed artefacts [269]. This term is originally derived from biological science and provides a rigorous and typological structure supporting further work or research. The concept of the clustering has been introduced and applied to classify objects for the last decades [26]. It groups or distributes unclassified or unlabelled objects into different clusters with systematic and algorithmic analyses based on their dimensions or affiliations [3].

2.2.3.3 Data Warehouse Architecture Classification Methods

Regarding the related literature on DWHA classifications, there are some research studies, but the focus as being less on the processes or methods of generating these classifications. Therefore, it is necessary to review the previous research into classifying architectures in the domain of information systems (IS) with the terms (classification, framework, typology, taxonomy and clustering) discussed above to identify related studies. These studies can provide references to design and develop the method for DWHA classification in this research.

Tang et al. [235] recognise the need for and the importance of classifying enterprise service-oriented architectures for the improved design of IS, satisfying business requirements and reducing enterprise IT complexity and cost. In their research, these architectures are classified into six sub-styles based on the domain model and the abstract model. Aier et al. [4] are based on an exploratory em-

pirical analysis to classify three different enterprise architecture scenarios using the hierarchical clustering algorithm. Lantow et al. [130, 233] propose a system and a framework, respectively, to analyse and classify different enterprise architectures based on dimensions, which benefits managers in easily making decisions as regards selecting enterprise architectures and identifying possible directions for future research in this domain. Pruijt et al. [200] generates a typology to specify and exemplify responsibilities in each layer of the software architectures from different perspectives, which provides an overview of responsibility types and allocations in the software of business IS and enhances the analysability, reusability and portability of the system. Dukaric and Juric [60] propose a unified taxonomy of IaaS (infrastructure as a service) architectures based on their fundamental components for analysing and classifying them into ordered categories or layers, and applying this taxonomy to design an IaaS architectural framework. Zhang et al. [284], meanwhile, create a new hybrid algorithm based on weighted and directed class as the fundamental entity for clustering, which is for object-oriented software architecture recovery. The methods proposed in these research studies are tailored to certain circumstances or are empirical in order to configure dimensions for architecture classification, which are illustrated in table 2.1 below.

Table 2.1: Some architecture classification methods

Research	Terminology	Sub-domain	Mechanism
[58]	Type	Data warehouse architecture	Based on types of data
[18]	Classification	Data warehouse architecture	Literature and survey
[218]	Classification	Data warehouse architecture	Components and their relationships
[235]	Classification	Enterprise service-oriented architecture	Domain model and abstract model

Research	Terminology	Sub-domain	Mechanism
[4]	Classification	Enterprise architecture	Exploratory empirical analysis and hierarchical clustering
[130, 233]	Framework	Enterprise architecture	Based on dimensions
[200]	Typology	Software architecture	Specifying and exemplifying responsibilities
[60]	Taxonomy	Infrastructure as a service architecture	Based on fundamental components
[284]	Cluster	Object-oriented software architecture	Based on the weighted and directed class as the fundamental entity for clustering

2.2.3.4 Discussion

As discussed above, a method is required to classify DWHAs. However, limited research studies can be identified referring to these methods on how to classify DWHAs in the literature. Hence, architecture classification methods in IS are investigated hereby. A traditional LR is conducted to collect related publications and investigate what methods they used to classify architectures. During this LR, the keywords (e.g. data warehouse architecture, classification, type, framework, etc.) are used to search some online data sets such as scholar, scopus, IEEE, etc. There are some studies which analyse and classify IS architectures such as [200, 233], it is still difficult to prove whether a classification is consistent and reliable,

because much manual and empirical intervention may affect classifications, and it is not well proved whether these classifications are consistent and reliable or not. Furthermore, different architectures might be described in various patterns such as third-party or self-defined modelling languages, which may complicate and affect architecture classifications. To the best of knowledge, there is no unified method to collect, model, and classify architectures based on modelled components.

Different terminologies are leveraged to describe architecture classification in different research studies covering multiple sub-domains of IS (e.g. DWHA, enterprise architecture, software architecture, etc.). In these research studies, different mechanisms are applied to identify and classify architectures. Most of them are based on features or components to generate classifications. Therefore, this research will leverage a similar mechanism to analyse and measure some features when classifying DWHAs.

2.3 Data Warehouse Architecture Features

Features are essential to evaluate DWHAs in this research. The Institute of Electrical and Electronics Engineers defines a feature in IEEE 829 as "A distinguishing characteristic of a software item" [15]. Another definition of a feature, from [178], is "a typical quality or an important part of something", and it has a variety of synonyms: "characteristic", "attribute", "quality", "property", "aspect", "facet", "side", "point", "detail", "factor", "ingredient", "component", "constituent", "element", "theme", "peculiarity", "quirk", "oddity", etc. For instance, Goar [72] proposes the main features required to successfully implement a DWH. Some studies in the DWH domain use the term of criteria to describe the DWH project characteristics [136, 229]. Some studies identifies the critical factors for determining the performance of DWHs in an organisation [96]. By reviewing the related literature, the potential synonyms of DWHA features may include "criterion", "factor", "characteristic", "attribute", "aspect", "component" and "element", which can be used to identify DWHA features in further investigations.

2.3.1 Data Warehouse Architecture Feature Definition

Clearly, different terms have been used in previous research studies when investigating DWHAs, which reflects the different aspects of DWHAs. In addition, the definitions proposed by [15, 178] are general and broad for wide domains, which cannot be used to accurately describe the characteristics and attributes of a DWHA. In order to effectively identify DWHA features, a definition of the DWHA feature is proposed and applied in this research. The proposed definition and potential synonyms of feature will be applied to extract and identify potential features, which can facilitate the accurate identification and collection of features.

A feature in a data warehouse architecture is an important portion or a distinguishing characteristic which is reflected as a component or attribute to affect performance.

2.3.2 Data Warehouse Architecture Feature Identification

By reviewing the literature with the definition of a feature and its synonyms presented above, different features can be identified from previous research studies. For instance, Arnott [21] provides a set of factors that influence the success of a DWH and its business intelligence system. Ariyachandra and Watson [20] conduct research to investigate features which influences the evaluation of an organisational DWHA. It suggests that different combinations of specific factors affect the likelihood of selecting one architecture over another. Choudhary [45] provides some features which may potentially affect the evaluation of architectures. As a consequence, these features are identified and summarised in table 2.2 below.

Table 2.2: Key or typical features from previous research.

Research	Year	Features
[37]	2004	ETL, marketing and change management, business, data model, executive sponsor, project team, users

Research	Year	Features
[96]	2004	Support from the top management, size of the organisation, effect of champion, internal needs, competitive pressure
[18]	2006	Information quality, system quality, individual impacts, organisational impacts
[136]	2007	Display interface, analysis tools, query functionality, compatibility, integration, database support, ETL functionality, data quality checks, metadata management, DWH administration, direct cost, indirect cost, vendor reputation, vendor stability, vendor support, vendor experience
[142]	2008	Resource consumption (labour usage and computing budgets) system usage measures (users and queries), quality measures (data age and availability), size measure (change data amount), flexibility ratio(ad hoc)
[21]	2008	Committed and informed executive sponsor, widespread management support, appropriate team, appropriate technology, adequate resources, effective data management, clear link with business objectives, well-defined information and systems requirements, evolutionary development, management of project scope
[39]	2008	Organisational implementation success, operational, technical (strategic alignment, management support, external environment, champion, resources, user participation, training operational, project planification, prototype, external consultant, team skills, source systems, technical skills

Research	Year	Features
[20]	2010	Information interdependence, urgency, task routineness, strategic view, resource constraints, perceived ability of the IT staff, sponsorship level
[72]	2011	Performance effect factor (valuable communication, education, training and documentation, call centre, network management, ETL, materialised view)
[270]	2011	Ad-hoc access, near real-time actuality, flexibility in usage, actuality of data, speed of data provision, availability of data, system stability
[45]	2012	Information interdependence between organizational units, management's information needs, urgency of need for a DWH, nature of end user tasks, constraints on resources, view of the DWH prior to implementation, expert influence, compatibility with existing systems, the perceived ability of the in-house IT staff, source of sponsorship, technical issues
[192]	2013	Organisation culture, technical tools, management support, user involvement, quality of data sources, self-efficacy, knowledge sharing, clear objective, scope and goals
[165]	2015	Business process, requirement management, measurement and evaluation, DWH life cycle, business requirement definition, technical architecture design, produce selection and installation, dimension modelling, physical design, data staging design and development, end-user application specification, end-user application development, project manager, deployment

Research	Year	Features
[283]	2016	Data quality, data quality management practices, external expertise, ETL tools, jobs' loading planning, top management commitment, teamwork, schedule
[59]	2017	ETL, logical modelling (ROLAP, MOLAP, HOLAP and non-relational), multidimensional schema, join-less schemata, star or snow-flake schemata, volume, velocity, variety, veracity, value

2.3.3 Taxonomy for Feature Structuring

Since a large number of features may potentially affect DWHA investigations, it is worthy to design an approach to identify features which should be reliable and critical. After obtaining these features, it is valuable to further organise them into a taxonomy, because a taxonomy provides a well-organised, rigorous and typological structure supporting further work or research navigation. Taxonomies can organise concepts in order and be applied by researchers to postulate on the relationships among the concepts [148]. Williams et al.[269] illustrate taxonomies can be leveraged to understand the science behind design principles of observed artefacts. Taxonomies classify the large number of focused information into well-defined and easily understood categories for future developments [112]. Nickerson et al. [166] denote a primary issue in many disciplines is how to classify objects into taxonomies whose process is complicated and not adequately explained in the IS literature.

2.3.3.1 Taxonomy Generation Methods

In information system research, taxonomies are usually derived from the researcher's perceptions. For example, Anu et al. [16] are based on researchers' understanding and perception to empirically generate and evaluate the human error taxonomy

for software requirements. Triguero et al. [246] empirically conduct an exhaustive research that involves a large number of data sets, with different ratios of labelled data to measure their transductive and inductive classification capabilities. Nickerson et al. [166] conduct research using combined method (conceptual-to-empirical and empirical-to-conceptual) with data from the literature to generate mobile application taxonomies. Astillo et al. [23] apply a systematic literature review (LR) to identified a knowledge-based taxonomy of critical factors for adopting electronic health record systems. Hatami-Marbini et al. [83] conduct a LR to generate a taxonomy on fuzzy data envelopment analysis. Prat et al. [198] execute the conceptual-to-empirical approach based on a LR to create the taxonomy of evaluation methods. Machchhar and Kosta [139] collect classification techniques by systematically and exhaustively reviewing related literature and creates a taxonomy in the bioinformatics fields. In order to achieve the purpose, this research designs and develops a set of method-driven processes to generate a taxonomy for better identifying reliable and critical features ragarding the SRQ 2. Hence, it is necessary to discuss what taxonomy methods can be applied. These methods are summarised in table 2.3.

Table 2.3: The summary of taxonomy methods derived from [166]

Taxonomy Method	**Description**	**Relevant Publications**
Inductive taxonomy method	Inductive taxonomy method involves observing empirical cases to analyse and determine dimensions and relationships.	[115, 246]

Taxonomy Method	Description	Relevant Publications
Deductive taxonomy method	Deductive taxonomy method is extended from theory or conceptualisation rather than derived from empirical cases. This approach may be followed by an analysis of empirical cases to evaluate and perhaps modify the created taxonomy.	[145, 225]
Mixed inductive and deductive taxonomy method	It combines inductive and deductive methods to generate a taxonomy.	[166, 198]
Intuitive taxonomy method	Intuitive taxonomy method is a group of ad hoc or self-defined approaches created based on researchers' understanding or specialised for certain issues.	[205, 214]
Others	Other approaches do not belong to these four categories illustrated above.	[56, 71]

2.3.3.2 Literature Review Methods for Taxonomies

The well-defined taxonomy should be supported by a plethora of evidences and documents from multiple resources. LR is a way to search relevant available research and non-research literature on the topic with an objective, thorough summary and critical analysis [52]. Webster and Watson [262] state a relevant review is essential for any academic project. An effective review is the cornerstone for advancing knowledge discovery, which benefits theory development and uncovers

research field if necessary. LR provide potentially useful implications which are significantly vital [221]. LR is a rational choice to robustly generate a mature taxonomy with traceable and repeatable processes. It collects relevant data to create this taxonomy by reviewing previous literature. There are different LR and taxonomy methods which can be applied. Hence, it is necessary to discuss relationships of their methods in order to choose suitable LR and taxonomy methods for this research. The links between taxonomy methods and LR methods are demonstrated in table 2.4.

Table 2.4: Links between taxonomy and literature review methods

LR Method	Taxonomy Method	Explanation
Meta-synthesis LR	Inductive taxonomy method	Qualitative research
Meta-analysis LR, theoretical LR	Deductive taxonomy method	Quantitative research
Traditional or narrative LR, argumentative LR	Intuitive taxonomy method	They are more depending on the human intuition and understanding.

2.3.4 Discussion

Different studies propose different features based on their requirements or circumstances. These features cover different aspects of DWHA features. To be more specific, bottom-up and top-down are more closely related to the structure of DWHAs, while the cube and DM are components in DWH systems. Some features refer to aspects of human resources (e.g. executive sponsors, project teams, users, etc.), whilst, some features focus on business levels (e.g. internal needs, competitive pressure and the benefit to the business). It is necessary to systematically analyse and organise them to better investigate them for further evaluation. In addition, by analysing these research studies, they do not provide clear or adequate information or explanations on how these terms are identified and de-

termined as the key or typical features rather than others. Inadequate evidence may affect the identification of these critical features, which are only important or critical in some certain circumstances, being less important in others. Although, these features are identified, it is necessary to organise them into a taxonomy for better understanding their relationships and using them in evaluation.

2.4 Data Warehouse Architecture Evaluation Studies and Methods

In this section, some studies are discussed to investigate the evaluation of DWHAs or DWH projects from a feature perspective. Furthermore, some methods referring to evaluating DWHAs and other architectures are discussed to deeply understand them for further design and development of the evaluation method proposed in the present research.

2.4.1 Research Studies on Feature-based Evaluation

There are many research studies on how to evaluate DWH systems by measuring features. These features are normally identified by manual intervention or from specific situations. For instance, Wixom and Watson [273] process an empirical investigation of the factors influencing success of DWH projects, in which a search model with a set of abstract features (e.g. resources, development technology, data quality, etc.) is proposed to present DWH success. During the phase of design, it is necessary to conduct evaluations based on features. For instance, Mohsen and Hassan [159] propose a framework with several features (e.g. modularity, adaptability, compactness, etc.) to assist designers and warehouse managers to make informed decisions. Hsieh and Tsai [95] operate a study on optimum design of a warehouse system for efficiently picking orders, which considers the effects on the system performance for factors.

There are several studies which propose some approaches to benchmark different aspects of DWHAs. These approaches can be easily re-operated with detailed

steps and guidelines. For instance, as ETL is an essential part in a DWHA, TPC (transaction processing performance council) proposes the first industry benchmark (TPC-DI) for ETL evaluations [194], which is a non-profit corporation founded to define transaction processing and database benchmarks. Before the TPC-DI, there are some self-defined benchmarks, such as Efficiency Evaluation of Open Source ETL Tools [140] and the Data Warehouse Engineering Benchmark [53]. However, there is a lack of an industrial standardised ETL benchmark, which can be used to evaluate performances of ETL tools [163]. Besides, TPC proposes other benchmarks e.g. TPC-E, TPC-DS, etc. to evaluate on-line transaction processing, decision support systems, queries of large volumes of data and some others [164]. As can be seen, these benchmarks are normally emphasis on a specific aspects of a decision support system. There is no method or benchmark which investigates DWHAs by evaluating features from the architectural perspective.

2.4.2 Methods for Feature-based Evaluation

In order to systematically conduct this research, it is necessary to investigate methods used to evaluate DWHAs in previous research. By reviewing the related literature,some methods can be identified from previous research studies to evaluate DWHAs. To be more specific, Lin et al. [136] establish a fuzzy-based decision-making procedure for DWH system evaluation and differentiation. Ariyachandra and Watson [20] are based on hypotheses and several key organisational factors to evaluate and differentiate DWHAs. Wibowo et al. [266] propose a group of decision-making procedures to evaluate DWH systems. As it is not adequate to investigate DWHA evaluation methods only in the domain of the DWHA, A further review is conducted by searching architecture evaluations in IS. In order to collect these methods, a LR is processed by searching the relevant literature, in which 79 papers are retrieved over a time frame stretching from 1981 to 2019 and covering a variety of domains (e.g. decision support systems, CASE tools, COTS, DWH systems, etc.). Various methods are proposed to evaluate general or

specific systems. Table 2.5 illustrates these methods, which are interpreted under four aspects. In this table, the evaluation means how does a method evaluate architectures; the expert indicates whether a method needs expert support; the feature stands for whether a method uses the features to evaluate architectures; the goal explains what the purpose of a method.

Table 2.5: Architecture evaluation method summary based on LR results and [183, 220]

Name	Evaluation	Expert	Feature	Goal
AHP	Hierarchically structuring a decision-making problem	Decision-makers or expert team(s)	Features are involved	Helping people to understand and simplify the problem
Criteria	Evaluation can be done for different purposes	Expert(s) to form Criteria	Features should be considered in the Criteria	Setting Criteria or using Criteria to achieve the goals
Fuzzy-based	Decision-makers can use linguistic terms to evaluate alternatives	Decision-makers or experts	Features are involved to make decisions	The process of evaluating alternatives is intuitive
Weight	Ease of use	Experts for weighting features	Features need to be weighted	Evaluating and selecting alternatives using weighted attributes

Name	Evaluation	Expert	Feature	Goal
SAAM	Scenario based functionality and analysis	A expert evaluation team	Features show in scenarios	Risk identification, suitability analysis
ATAM	Hybrid approach: questioning & measuring	A expert evaluation team	Features are derived from questions	Sensitivity and trade-off analysis
ARID	Combination of scenario-based and active design reviews	A expert evaluation team	Features are derived from scenarios and design reviews	Validating the viability of design for insights
SAAMER	Information modelling and scenario-based	Not specified	Features influence the information modelling	Assessing SA for reuse and evolution
ALMA	Dependent on the analysis goal	Not specified	Not specified	Changing impact analysis, predicting maintenance effort
SBAR	Multiple approaches	Not specified	Features are involved	Evaluation is conducted based on a scenario

Name	Evaluation	Expert	Feature	Goal
SAAMCS	Complex scenarios	Not specified	Features appear in complex scenarios	Predicting context relative flexibility, risk assessment
ISAAMCR	Scenario-based	Not specified	Features appear in scenarios	Analysing flexibility for reusability

2.5 Conclusion

This chapter systematically analyses different aspects regarding this research via reviewing the relevant literature to further make progress based on chapter 1. It discusses different types of DWHs, DWHAs, classifications and features involved in previous research studies, and proposes some definitions applied in the present research. Furthermore, a set of methods to evaluate DWHAs and architectures in other IS domains. As can be observed, the evaluation concepts are diversified, but almost all of them are based on features to process evaluation. Therefore, in this research, features are applied and measured to evaluate DWHAs. However, as discussed in the previous section, features proposed in previous research studies are varied and loosely supported by evidence. As this research is based on features to evaluate DWHAs, it is necessary to identify key or typical features with rich evidence supported for further evaluation. In addition, half of them require experts or decision-makers to evaluate architectures or systems. Nevertheless, almost all DWHA evaluation methods (e.g. AHP, criteria, fuzzy-based and weight) need experts' or decision makers' support during the processes of evaluation. As it is normally time-consuming and exorbitant to organise and conduct evaluation with a group of experts or decision-makers who should have rich experiences in this domain, it is worth designing and developing a method to explicitly and efficiently evaluate DWHAs with less expert involvement and interpretation. The

main purpose of this research is to create a method to systematically address the MRQ and achieve objectives, which will be concretely explained in the following chapters.

3 Research Methodology

This chapter discusses the concepts to present why research and research methodologies are necessary to systematically solve problems in section 3.1 and 3.2. As this research should collect and analyse data based on literature, hence, different types of LR methods are explained and tabulated in section 3.3. Subsequently, through reviewing different types of research methodologies in qualitative and quantitative research in section 3.4, the DSRM along with six steps is chosen to systematically drive the procedure of this research explained in 3.5, 3.6 and 3.7.

This chapter is significant to provide methods and methodologies for systematically conducting this research with theoretical support. Some LR methods presented in this chapter are applied in different stages of this research. For instance, the Meta-analysis and Historical LRs are used in chapter 4 and chapter 5 to collect data referring to DWHAs and features. The DSRM plays a vital role to guide this research by following its steps. The structure of this thesis is also organised based on these steps. chapter 1, 2 and 3 are related to the first two steps; chapter 4, 5 and 6 refer to third and fourth steps; chapter 7 and 8 link to the last two steps. The details are illustrated in figure 3.1.

3.1 The Importance of the Research

Research can be applied to identify or confirm facts, strengthen the results of previous work, address existing or emerging problems, support theorems or generate new theories [149]. Hevner and Chatterjee [88] explain that research is a process

supported by data which answers a question, resolves a problem or serves to better understand a phenomenon. Johnston [107] defines research as a logical process of steps used to collect and analyse data to understand a topic or issue in order to solve a problem or improve knowledge. Creswell [50] states that research is a process of collecting and analysing information to gain an understanding of a topic or issue, which includes three stages: raising or presenting a question, gathering data to answer the question and demonstrating the answer to the question. The goal of research is to investigate from the unknown to the known regarding a specific question being required in any field (regarding the natural, the artificial or the behavioural world) which should be answerable, and it should intend to fill a gap in existing knowledge [141].

3.2 The Importance of Research Methodologies

Research requires a methodology to systematically guide the processes and ensure scientific outputs. A research methodology is a systematical means to address the research problem [122]. It is a subject having several ways with academic disciplines to facilitate research in multiple domains [124]. Another concept, namely the research method, is similar to the research methodology. There are interchangeable or mixed used in some research such as [249, 252]. However, research methods are understood as approaches or techniques that are used for the conducting research. They are more closely related to the behaviour and instruments used in research, such as selecting and constructing research techniques to make observations and analyse data [122]. It is essential to use a research methodology or method if a research study should be systematically conducted. In the present research, research methodologies and research methods are understood having different levels of details in a research study. A research methodology is more macroscopic and has underlying theories in order to guide the steps of a research study as a whole. A research method focuses on what to conduct as part of research on a microscopic level. In the following sections, several re-

search methodologies applied in previous research are discussed for the purpose of choosing suitable methods and methodologies to conduct this research.

3.3 Methods for Literature Review

As this research should be conducted with relevant information and data supported, LRs are useful approaches to understand the current situations and collect data in this domain. A methodological review of the past literature is an essential feature for any research work [262]. The importance of uncovering what is already known in the body of knowledge should not be underestimated [80, 134]. An effective review is the cornerstone for advancing knowledge discovery, which benefits theory development and uncovers the research field if necessary. LRs are of crucial importance to the hierarchy of evidence, as they provide potential implications which are most important information [221]. Therefore, in the initial stage of this project, the literature in relation to DWH, DWHAs, evaluation methods and features are reviewed. In order to better understand LRs, based on [75, 76, 206, 208, 258, 263], some LR methods are discussed and summarised in table 3.1 below. In this research, the traditional LR is applied to collect data and identify the research question and the gap in chapter 1 and 2. Subsequently, meta-analysis LR and historical LR are used to facilitate further investigation on DWHAs classification and feature identification in chapter 4 and 5.

Table 3.1: Literature review methods

LR Method	Description
Traditional or narrative LR	Traditional or narrative LR is based on research questions to discuss and summarise the literature, then propose conclusions. It gives a whole picture of the current situation, which facilitates the identification of gaps or inconsistencies in this field.

LR Method	Description
Systematic LR	Systematic LR is a more meticulous approach with well-arranged steps and timeframes compared with other LRs. It is question-oriented to collect the literature for reporting, data analysis, the conducting of further research, theory generation and so on.
Meta-analysis LR	Meta-analysis LR is a sub-branch of systematic LR and is based on statistical techniques to analyse the results from a large amount of literature in order to better understand theories or prove some phenomena. It is more related to quantitative research than qualitative research.
Meta-synthesis LR	Meta-synthesis LR is a sub-branch of systematic LR and is the opposite of meta-analysis LR, which is based on non-statistical techniques. It is more related to qualitative research for aggregating findings into new interpretations.
Integrative LR	Integrative LR forms new results by integrating previous results with reviewed and critiqued new findings from the literature about the same topic or a related research topic which are reviewed and critiqued.
Theoretical LR	Theoretical LR further investigates already existing theories derived from issues, concepts or phenomena. It can be leveraged to develop new hypotheses, strengthen or even refute current theories.

LR Method	Description
Historical LR	Historical LR considers retrospectively an issue, concept, theory or phenomena throughout a period of time beginning from when it first appeared or in a momentous time node. It tracks the evolution of a topic from the initial stage to a certain stage, which allows the reader to better understand the history of the topic and guide the direction of future research.

3.4 Methodologies for Research Guidance

This research should be systematically conducted via well-organised processes. Hence, a research methodology is required to guide the processes and guarantee the quality of the output generated in this research. Whilst there are different types of research methodologies and they can be described from different perspectives. Heterogeneous research methodologies are generated and leveraged in different domains of research. It is necessary to discuss these methodologies proposed in previous research and to choose an appropriate one to guide this research. Consequently, several feasible research methodologies are identified in the domain of IS. These methodologies frequently appear in the literature, and they can be broadly classified into qualitative research, quantitative research, mixed research and others, which include action research, grounded theory, case study, DSRM and others.

3.4.1 Qualitative Research

Qualitative research is inherently inductive and allows researchers to generally explore meanings and insights in a given situation [133, 232]. It is more subjective compared with the quantitative research methodology and uses various methods (e.g. in-depth interviews) to gather information. For instance, in quali-

tative research, a small number of people or groups are interviewed to investigate and understand a phenomenon in depth [202]. Several types of concepts can be identified when applying this methodology, including phenomenology and case study. Phenomenology intends to understand a phenomenon observed by one or more individuals. Case study places an emphasis on one or more cases to comprehend issues or phenomena by analysing data in detail [74, 152]. In addition, this methodology is a broad theory including numerous sub-qualitative methodologies (e.g., action research and grounded theory), which are leveraged in IS research [51, 157]. In order to better understand and utilise this methodology, its sub-methodologies are discussed below.

3.4.1.1 Action Research

It focuses more on addressing problems related to a specific client's concerns [98]. It is initiated to address an immediate problem reflect a progressive problem-solving process [63]. Thus, the purpose of action research is to address a particular problem and to generate guidelines applied in practice [57]. This methodology is able to integrate academic and industrial concerns together. Hence, it is affected by industrial concerns and intended to address a socio-technical problem by generating a new solution fitting the organisational situation. Consequently, this research methodology leads to non-universal solutions, as it is biased or influenced by industry-related concerns.

3.4.1.2 Grounded Theory

Grounded theory has certain characteristics which are based on existing theoretical frameworks and which indicate how the theory does or does not apply to the phenomenon being studied by collecting and analysing data [9]. It is a methodology used to construct theories through systematically gathering and analysing data in social sciences [65]. This methodology is considered a powerful means of rigorous theory development, as it grounds a theory through the analysis of actual

settings and processes [250]. Therefore, grounded theory is focused on a question, or even just collecting qualitative data. With new data gathered and analysed, the raw theories extracted may be proven and rebuilt as the basis for new theories. This can involve a systematic qualitative approach to conduct research, focusing on generating middle-range theory from substantive data [5, 70].

3.4.2 Quantitative Research

Qualitative research focuses on facilitating analytical induction by investigating individual data or groups of data sets, while quantitative research involves studies to generate induction by analysing numerical data [35]. It normally generates results by conducting statistical analysis based on quantities or numbers [184]. It analyses numerical data and executes mathematical calculations to discover their properties for certain primary purposes, such as conducting measurements, making comparisons, examining relationships, making forecasts, verifying hypotheses, generating concepts and theories and so on [256]. Muijs [161] lists four situations involving the use of quantitative methods: intending to calculate a quantitative answer; where numerical change must be accurately analysed; planning to discover the state of something or explain phenomena (e.g. predicting scores on a factor or a variable); and testing hypotheses via numerical or quantitative evidence. It summarises two main types of quantitative research: experimental research and non-experimental research.

3.4.2.1 Experimental Research

Experimental research is based on experiments, in which a research study is conducted within controlled conditions to demonstrate a known truth or validate a hypothesis. The key characteristic of experimental research is that it involves controlled conditions when doing an experiment, controlling the environment as much as possible and investigating only certain variables of concern in the research [161]. There are two types of experimental research: true experimental and

quasi-experimental [49, 78]. In true experimental research, the cause-effect relationship among a group of variables is built via a research study, and all variables (dependent variables) are controlled except the one (the independent variable) which should be tested. This type of research applies systemic approaches to collect quantitative data and uses mathematical models in the analyses. Meanwhile, in quasi-experimental research, researchers do not allocate the control group randomly [30, 161].

3.4.2.2 Non-experimental Research

Non-experimental research lacks the manipulation of controlled variables, in which researchers conduct research studies by investigating variables occurring naturally [185]. Non-experimental research does not need further effects on setting up experiments to manipulate data and find relationships between variables. It is varied and includes some sub-research methodologies which can support correlational research and observational research [185]. Correlational research focuses on identifying relationships between two or more variables in the same population or the same variables in two populations [131]. It applies statistical techniques to investigate relationships between variables which are measured but not manipulated [185]. Observational research is variable and flexible, and it observes a great number of situations in a variety of ways. A survey is conducted to collect data, in which standard questionnaire forms are presented by telephone, in person, by post, via web-based forms or via emails. Furthermore, the collected data is analysed using observational methods to summarise the results [161].

3.4.3 Mixed Research

The mixed research methodology has been rapidly developed alongside the debate concerning qualitative and quantitative research, as it combines qualitative and quantitative methodologies to bridge their shortcomings and address a research question [81]. This type of research study combines the concepts and tech-

niques of quantitative and qualitative research. The fundamental mechanism of this methodology is that it utilises the collected data more completely and synergistically than by separately conducting quantitative and qualitative data collection and analysis [271]. This mixed methodology is typically traced to the multitrait and multi-method approaches [237]. Johnson and Onwuegbuzie [106] note that, in this type of research, researchers integrate and utilise techniques, methods, approaches, concepts or language from quantitative and qualitative research into a single study. In this type of research, data should be collected from multiple sources with mixed strategies and methods to strengthen research and ensure there is no overlap regarding weaknesses, which enables this mixed methodology to adequately investigate more insights than data collected by only qualitative or quantitative research [105]. In other words, it provides an opportunity to negate or complement weaknesses and counteract biases associated with a methodology, gathering strengths from both quantitative and qualitative methodologies [77]. In order to better understand this combined methodology, a specific mixed methodology (case study research) is discussed below.

3.4.3.1 Case Study Research

Case study research makes use of the case as a specific illustration to comprehend an issue, problem or phenomenon [227]. It is a research strategy allowing investigators to inquire into a specific situation or multiple cases through detailed, in-depth data collected from different sources of information and reports [191]. It can be used in single case study and multiple case studies with quantitative data from multiple sources, which means that it benefits from the prior establishment of theoretical conceptions. It is based on a mix of quantitative and qualitative evidence to conduct research studies [282]. The cases in case study research may be presented differently in a specific time and place, and they may involve individuals, organisations, events or actions. The cases may be mixed with different types of objects. The case is the subject of many research methods when it is

applied in abstract situations [114]. Therefore, a case study research methodology may involve quantitative or both qualitative and quantitative data to extract knowledgeable information for better understanding issues and phenomena based on cases which involve objects in reality or subjects in conception.

3.4.4 Other Methodologies

In addition to the research methodologies discussed above, there are some other research methodologies in the field of IS. The behavioural science and design science are discussed in order to investigate insight into their characteristics.

3.4.4.1 Behavioural science

Behavioural science explains or predicts human or organisational behaviour in order to develop and verify theories [89]. It relies on truth or the discovery of truth and normally involves collecting data to testify something valid or invalid [111]. Thus, it enables an understanding of the true reality with data supported by observing behaviour, then finally generates theories. It derives from natural science and intends to discover the truth, normally initiates from a hypothesis and gathers data to verify this hypothesis [88, 34]. Therefore, it is based on the reality of phenomena and hypothesis already provided to develop and justify theories. It more focuses on real activities or characteristics presented by human or organisations for generating general theories to enrich knowledge which is undiscovered before.

3.4.4.2 Design Science

DSRM creates new and innovative artefacts to extend the boundaries of human and organisational capabilities [88]. It seeks to create what is effective to strengthen or broaden current knowledge with artefacts [31]. It is an objective-oriented approach to produce an artefact which needs to be designed and evaluated [88, 175].

This research methodology creates things that fulfil purposes and solves problems. Hence, it focuses on creating and evaluating innovative artefacts that extend the ability to address important unsolved tasks. The main objective of design science is to generate knowledge which can be used to design solutions for problems [253]. Its artefacts includes, but are not limited to constructs, models, methodologies, algorithms, interfaces, etc. [103, 251].

3.5 Research Methodology Selection for Guiding this Research

In the previous section, different types of research methodologies are discussed and these characteristics are summarised in table 3.2 based on [9, 89, 161, 185, 98, 227]. DSRM is chosen as the main methodology to direct this research and the main reasons are explained as follows. Firstly, the mainstream of IS has accepted and leveraged DSRM for more than a decade [257]. Secondly, this present research needs to create a method with identified features by investigating different types of DWHAs. Thirdly, this methodology involves an artefact as the output of this research to achieve the objectives. Fourthly, DSRM includes processes concerning how to evaluate and demonstrate artefacts, which can be leveraged to assess this methodology. In addition, some concepts and techniques from other research methodologies can be used to supplement and strengthen this research, which are explained in the last column of this table.

Table 3.2: Research methodology discussion

Research Methodology	**Purpose**	**Output**	**Application in This Research**
Design science (Selected as the primary methodology)	Generating knowledge and creating artefacts to solve problems	Knowledge and artefacts	Main research methodology to guide this research

Research Methodology	Purpose	Output	Application in This Research
Action research	Combining theory and practice to address specific issues	Solutions for particular problems	Using its principles to analyse current situations and issues from academia and industry perspectives
Grounded theory	Proving and rebuilding new theories from raw theories based on qualitative data	Forming new theories	Some previous feature measurements are used but extended
Experimental research	Experiments are conducted within controlled conditions to demonstrate a known truth or validate a hypothesis	Truth or validated hypothesis	The proposed evaluation method will be evaluated by cases derived from experiments under controlled conditions
Non-experimental research	Investigating relationships between variables in natural ways	Relationships between variables	-

Research Methodology	Purpose	Output	Application in This Research
Case study research (Selected to evaluate this research)	Using cases to comprehend an issue, problem or phenomenon	Knowledgeable information for better understanding issues and phenomena	It is used to conduct evaluation
Behavioural science	Explaining or predicting human or organisational behaviour to develop and verify theories	Theories	-

3.6 Design Science Processes

With decades of development, different types of DSRMs have been proposed and applied in the literature, which shows different procedures and models to organise the processes of research studies. Hevner [87] proposes three cycles to manage research projects, which includes the relevance cycle, the design cycle and the rigour cycle. Offermann et al. [169] provide 11 processes in three phases: the problem identification, solution design, and evaluation. Takeda et al. [234] divide the processes into five stages: the enumeration of problems, suggestion, development, evaluation to confirm the solution and a decision on a solution to be adopted. March and Smith [143] advocate a framework with two dimensions, in which the first dimension is based on activities (building, evaluating, theorising and justifying), while the second dimension is based on outputs (representational constructs, models, methods and instantiations). Peffers et al. [189] define a methodology as involving the following six steps: problem identification and motivation; def-

inition of the objectives for a solution; design and development; demonstration; evaluation; and communication. All of these methodologies have similar processes by which to design and evaluate artefacts. In order to contextualise design science, the methodology proposed by [34, 189, 190] is adopted to drive and conduct this research. The processes of this DSRM fit this research because it states the problem, motivation, objectives and the creation of an artefact. In addition, the artefact created in this research should be evaluated by people not only from research communities, but from industry, which is crucial for achieving the contributions and proving its validity. The details and processes of this research under the six-step methodology are illustrated in figure 3.1.

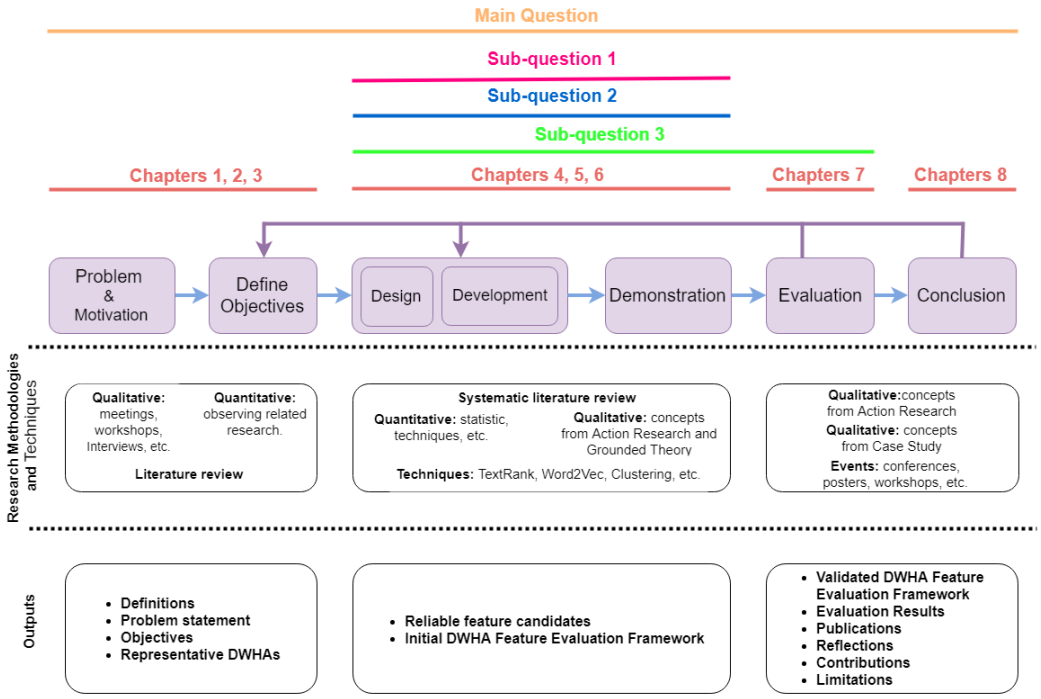


Figure 3.1: The research methodology adopted for this research

3.7 Research Application Based on Design Science

As the processes of the DSRM are divided into six steps, in order to better conduct this study, the sections below demonstrate how to apply this methodology in each

step along with techniques and concepts from other methods or methodologies (e.g. LR, statistics, action research, etc.). These six steps are grouped into three main parts based on the research questions and other aspects such as problems, motivation, objectives and the means of producing the artefact. These steps are deeply explained along with their location in chapters.

3.7.1 Identifying the Problem, Motivation and Research Objectives

These steps occur in the first two chapters. In chapter 1, through investigating previous research studies, the current issues and requirements are identified. Meetings, interviews and workshops are held to identify demands and concerns in industry and academia. Based on these issues and requirements, the objectives are determined, then the main research question and the sub-research questions are proposed for this research. In chapter 2, via reviewing the related literature, the definitions of the DWH, DWHA and features. The current issues and the research gap are further analysed to get insight into current situations referring to the methods of DWHA evaluation in this domain, which can be beneficial to solve the problem and achieve the objectives.

3.7.2 Design & Development and Demonstration

These steps are conducted in chapters 4, 5 and 6, in which the three SRQs are answered along with three proposed methods. For the SRQ 1, a method is implemented to identify representative DWHAs and the scope of this research. For the SRQ 2, a method is developed to determine reliable DWHA features which are important and should be measured in evaluation. For the SRQ 3, a feature-based modelling and evaluation method is created to systematically benchmark DWHAs.

In order to answer these SRQs with data supported, this research based literature review methods to collect data and use some machine learning algorithms

and two widely used algorithms in natural language processing to manipulate and analyse the data. For solving the SRQ 1, the first method classifies DWHAs through the clustering, which is a set of algorithms in machine learning to assort data into different groups [121]. This research applies the hierarchical clustering algorithm and the details of processes how to conduct the classification can be found in chapter 4. For answering the SRQ 2, the second method use two algorithms (TextRank and Word2Vec) to identify features and sub-features, finally generate a DWHA feature based taxonomy. The TextRank is originally derived from the PageRank algorithm which ranks web pages by the hyperlinks to identify their importance [286]. The Word2Vec is an algorithm to convert words to vectors based on Neural Networks which is also an machine learning algorithm [82, 204]. The details of how to execute these two algorithms are concretely explained in chapter 5. For address the SRQ 3, the third method is created and it does not use machine learning algorithms, but it based on the results of the first and second methods to form a framework for DWHA evaluation, which is presented in chapter 6. Therefore, machine learning is significant referring to operate data and generate results for supporting this research. The outlines of these three methods are briefly presented as follows.

3.7.2.1 DWHA Classification Method for SRQ 1

The first SRQ is related to identifying and describing these representative DWHAs and the scope of this research. This SRQ can be answered by identifying these representatives, but robust and quantitative evidence should be provided to support the answer. When designing this method, a quantitative research methodology associated with the meta-analysis LR is conducted to scientifically collect information from publications related to DWHAs. Statistical techniques are used to support and confirm these representatives with robust numerical proof. In order to generate the reliable result, this method is designed to iteratively upgrade the classification of DWHAs until it is stable. In other words, a consistent and reliable

classification model should be generated without variation between two continued iterations. The method is designed and allocated in a loop with four phases, which is diagrammed in figure 3.2.

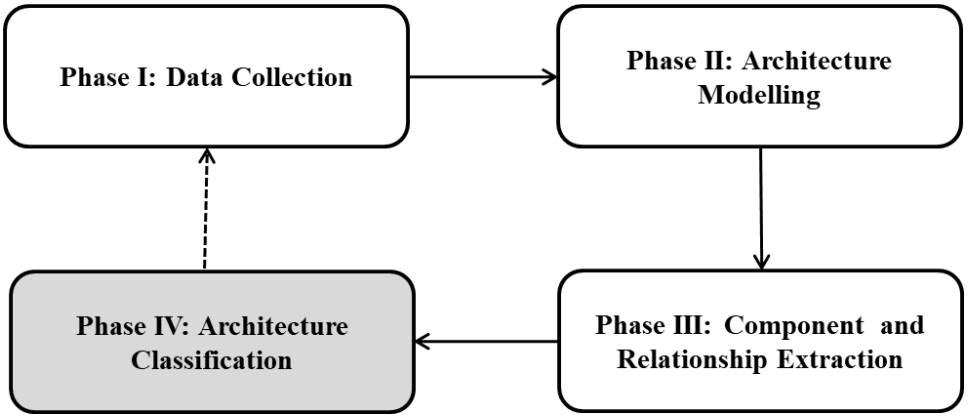


Figure 3.2: The architecture classification method with four phases

The first phase aims to collect DWHAs as raw materials. The second phase is to model these architectures in a unified way, because these collected architectures are represented with different styles and patterns. The third phase extracts components from modelled architectures to form a component matrix. In the fourth phase, the matrix is used as input to run algorithms and further classify these architectures. If the classification does not fit objectives or requirements, the process may flow back to phase I. This method is designed to be versatile, as it can be invoked to classify architectures without pre-condition or restriction on what domains of architectures belong to. Hence, DWHAs can be classified by executing this method. The details of how to identify these representative DWHAs is presented in chapter 4.

3.7.2.2 Reliable DWHA Feature Identification Method for SRQ 2

The second SRQ refers to deriving distinct features and using them to evaluate representative DWHAs. In order to investigate underlying and insight into features and their relationships, a combined LR with the Meta-analysis LR and the

Historical LR is conducted to trace the time at which DWHAs first appeared in the literature. After the relevant data is collected, a mixed taxonomy method is proposed and applied to deductively and inductively generate a DWHA feature taxonomy. During the building up of this taxonomy, two algorithms (TextRank and Word2Vec) are used as the main mechanisms to facilitate feature extraction and organisation, which are based on statistic concepts to analyse how important a word in documents and the relationships between them. This method aggregates these two algorithms' characteristics with well-organised steps embedded in the program flow to identify reliable features and further form the taxonomy with these features and their relevant terms (sub-features). The framework of this method is briefly designed with three phases, which is demonstrated in figure 3.3.

Figure 3.3: Phases of the feature and taxonomy framework method

In the phase I, the combined LR is conducted to collect data which is related to DWHA features. As for Phase II and III, their processes are based on previous concepts and theories, but they are redesigned and rebuilt to fulfil the purpose of the present research. To be more specific, the TextRank and Word2Vec approaches are applied in this method to assist feature extractions, analyses and interpretations. These two algorithms are derived from Google projects based on the core algorithm of PageRank and the theory of neural network, respectively, which are highly accepted and widely applied to identify keywords, generate abstracts, cluster terms, interpret relationships of terms, etc. Hence, the main mechanism of this method is to aggregate the statistical concept and these two algorithms to identify potential features and generate the initial taxonomy. It generates an initial taxonomy with reliable features in a hierarchical structure. Furthermore, this initial taxonomy is refined by experts. The processes of Phases I, II and III are concretely explained in chapter 5.

3.7.2.3 A Feature-based Modelling and Evaluation Method for SRQ 3

After representative DWHAs and reliable features are analysed and identified based on these two methods, they are used as the materials to generate the framework as the main engine of the evaluation method. The concepts of action research and grounded theory are applied during these processes. During designing and developing this framework, meetings and workshops are held with individuals from industry and academia in order to obtain suggestions and feedback for upgrade. In other words, the framework is demonstrated in front of people from time to time and is subject to suggestions for further iterative development and upgrading. For example, these features are further explained with their own sub-features. If a feature's sub-features receive negative feedback from meetings or fail to evaluate DWHAs well, this feature is replaced by other features or is described by other sub-features, depending on the circumstances.

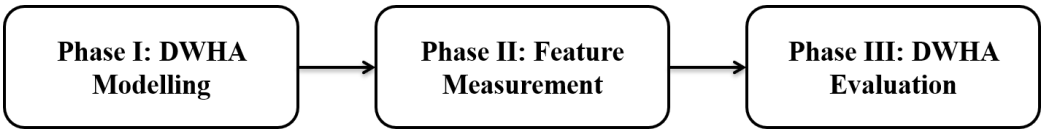


Figure 3.4: Phases of the modelling and evaluation method

After these features, sub-features and their measurements are determined, they are organised into a framework to evaluate DWHAs. For the sake of systematically conducting evaluation, a method is designed along with three phases which are briefly illustrated in figure 3.4. In the first phase, a DWHA needs to be modelled in order to depict its components, then identify some insensitive features. In the second phase, some sensitive features are measured, these features normally cannot be measured by directly observing the modelled architecture presented in the first phase, for which measured results should be calculated based on other information (such as time, data size, throughput, etc.). After these two phases, all features are measured and the results are put into the third phase. In the last phase, the framework is leveraged to evaluate this DWHA and output the results.

The structure of the framework and the details of these three phases are explained and demonstrated in chapter 6.

3.7.3 Evaluation and Communication

Evaluation is for observing and measuring how much the artefact supports a solution to the problem through comparing the objectives of a solution to actual observed results from the use of the artefact [190]. In design science research, it is crucial to evaluate artefacts within different environments [88, 187]. Artefacts should be evaluated in the context of their respective applications and implementation environments regarding the requirements [188]. The initial evaluation of an artefact may simply show that it works and produces adequate solutions [167]. The problem is ambiguous in evaluation, some researchers may think that this evaluation is adequate, while others may believe it is inadequate [167]. There are a plethora of theories and methods regarding the evaluation of artefacts in design science research [85, 188]. Based on [88], they can be broadly categorised into two major philosophical groupings: objectivist and subjectivist. They may correspond to the concepts from quantitative and qualitative research, respectively.

Each evaluation theory or approach has its own advantages and disadvantages [141]. Therefore, in this research, a combined evaluation method is applied to test and validate this proposed method, in which some concepts in qualitative and quantitative research (e.g. case study, interviews, survey, etc.) are used in order to adequately evaluate this method. The evaluation here is aimed at answering the sub-research question 3 and estimating how well the objectives have been achieved. According to [197] and the taxonomy shown in figure 3.5, two aspects (understandability and validity) are chosen as goal dimensions which are measured in the evaluation. Understandability is related to interpretability and unambiguity on whether this method is **explicit** or not, which is associated with sub-research question 3. According to [197, 231], validity can be assessed by measuring the reliability of an artefact. In addition, this framework is designed to

evaluate DWHAs and output results in an **efficient** way, which is associated with sub-research question 3. This can be examined by if this method can be used by individuals who are not experts in this domain. If it can be used by non-experts, then it believes that this method can be efficiently to evaluate DWHAs. If DWHAs can be efficiently evaluated by this method and the results are reliable, then the validity of this methodology is proved. Furthermore, if this method and its generated results pass the examinations in relation to this aspect. This method can be proved valuable to evaluate DWHAs and output as the artefact. The main RQ can be properly answered, and objectives of this research can be achieved.

During the evaluation, cases are collected from open sources, this research and a cooperative project to measure this method from a qualitative perspective. At the same time, interviews are conducted to collect data from people with experience in industry and academia to measure this method and generate numerical results to support this framework from a quantitative perspective. If the framework is not executed well or the objectives are not achieved during the evaluation phrase, it may be directly upgraded based on the results or this research may return to a previous stage for the purpose of iteratively upgrading this framework.

Overall, it is necessary to present artefacts in front of people from technical and organisational backgrounds [90, 141]. It is significant to associate with researchers and industrial people to present the utility and novelty of this research and its artefact [190]. During the conducting of this research, several articles and posts have been published to present different aspects and stages of this research in academic conferences, industrial events and workshops. These events provide valuable opportunities to discuss this research with people having different backgrounds.

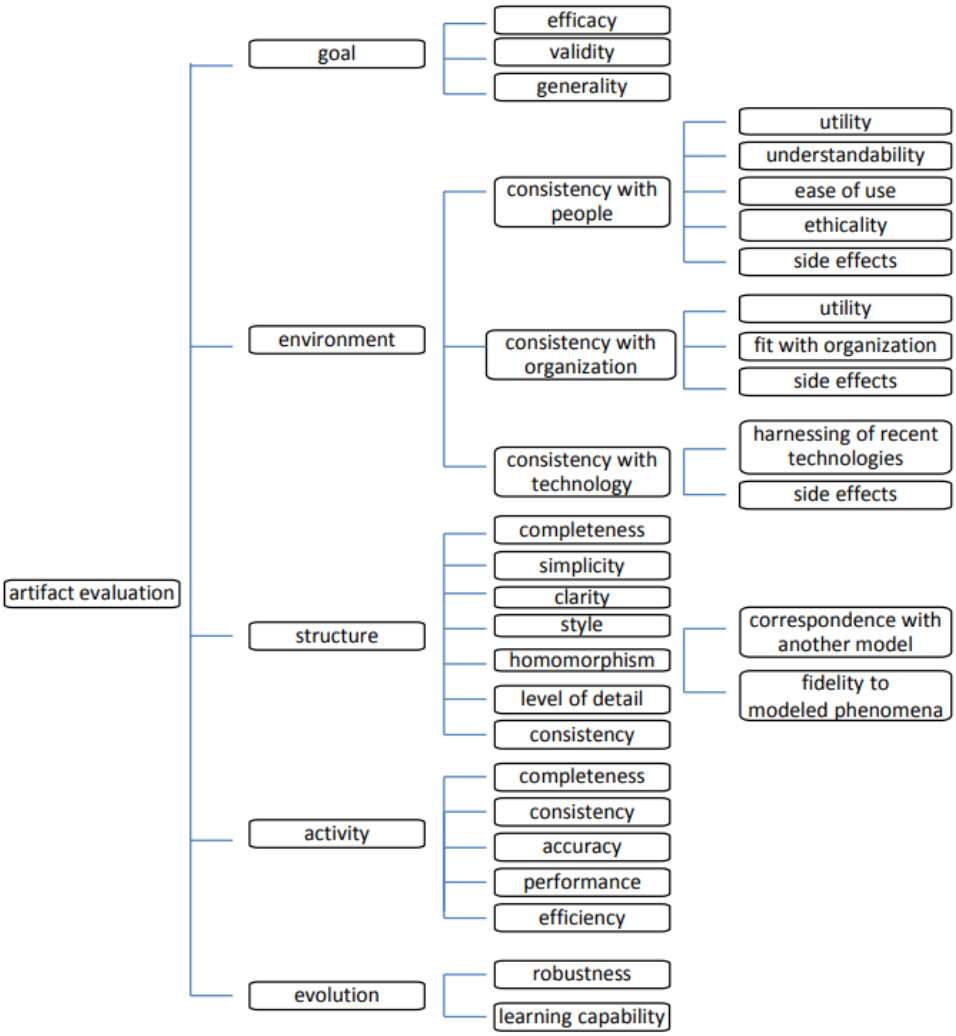


Figure 3.5: Hierarchy of criteria for IS artefact evaluation derived from [197]

4 Result of SRQ 1 - Method for Data Warehouse Architecture Classification

This chapter is related to SRQ 1 and three DSRM phases (design & development and demonstration). It designs and develops a method with a set of processes and algorithms to systematically classify architectures. This method involves four phases, namely architecture collection, architecture modelling, architecture digitisation and architecture classification. By executing it, DWHAs are firstly collected from relevant online sources by conducting a systematic LR, then they are modelled by a modelling language, digitised and classified through the rest of phases.

4.1 Architecture Classification Method Design and Development

This method is divided into four phases. The first phase is to collect DWHAs from the literature, the second phase refers to models all these collected DWHAs with a modelling language. The third phase digitises these modelled DWHAs, which converts them to machine-readable. The fourth phase classifies these digitised DWHAs through a set of processes. The details of each phase are described as follows.

4.1.1 Phase I: Architecture Collection

When classifying architectures, it is necessary to collect and analyse them to provide valuable information for generating a data supported and reliable architecture classification. As these architectures are normally distributed in different data sources, it is important to determinate how to systematically collect and prepare data for further phases. There are a variety of methods or techniques which can be applied to collect data (e.g. experiment, interviews, questionnaire and survey, observation etc.) [94, 179]. These methods can be summarised and broadly distributed into three types based on their collected data and relationships between them: method for self-created data (experiment, interview, questionnaire and survey, observation, case study), method for semi-created data (document and record review, LR) and method for pre-created data (data source re-usage). As a consequence, there are three types of scenarios when collecting architectures for classifications, in which different data collection methods should be executed: If there is not related data available, then the related methods need to be applied to self-create the data; If some existing data sources are found but not completely match requirements, then extra transformations should be conducted to prepare data; If data sources fulfil all requirements and are reused directly, but these sources are not sufficient. In terms of this research, there is no existing data which can be pre-processed or directly used. However, there are a number of architectures in the literature. A LR is conducted to collect semi-created architectural data from publications rather than experiments or self-creation.

As discussed above, there are different methods and scenarios to collect and prepare data. As pre-created data is limited, self-created or semi-created data needs to be collected from the literature. In addition, architectures may be described and presented with different patterns (e.g. diagrams or only text). Thus, in this step, data should be collected as detailed as possible for further thoroughly understanding them, especially for the self-created data. These collected architectures should be adequate to cover all possible classes when generating classi-

fications. Data should be collected from multiple sources to reduce the bias of classifications especially when collecting semi-created data. After data is collected, it is essential to roughly pre-process data for cleansing, transforming and deleting dirty data.

4.1.2 Phase II: Architecture Modelling

When describing architectures, people may use different ways to describe their structures and components. Hence, these collected architectures may be described and diagrammed via different tools or approaches (e.g. self-defined, process flow, modelling languages, etc.), which complicates the process of the classification. In order to unify these different presentations, an architecture modelling language should be applied to describe these architectures. As there are different types of modelling languages which can be found in the literature, it is necessary to choose a proper one which should be widely accepted in this domain and have the ability to illustrate architectures, components and their relationships. Once a modelling language is selected, it should be applied to model all these collected architectures to output the modelled architectures with a consistent pattern. As for the pre-created data, the provided architectures are already modelled and fit the objectives. As for the self-created and semi-created data, architectures in these types may need to be extracted from text or figures, then modelled manually. This step is crucial but necessary to uniformly their formats from different descriptions and visualise them. It serves as a fundamental and indispensable part for the next phase to facilitate the component extraction and matrix generation.

4.1.3 Phase III: Architecture Digitisation

After the collected architectures are modelled, they are converted to machine-readable formats for further classifying them. The output in this phase is a component matrix, which contains a list of architectures and a comprehensive set of components arrays for each of them. To be more specific, there is a variety of

potential components extracted from modelled architectures, which should be organised and added into a matrix. There are two scenarios to be considered when doing this adding: If an architecture has new components, then these new components are added into the matrix as new columns, a new row is created in the matrix for this architecture and all related slots of its components are marked; If an architecture has some components, but all components are already in the matrix, then it does not need to add new columns, just mark all slots of components for this architecture in a new row of the matrix. Besides, a component may include sub-components, even the sub-components may have sub-components, so the component level needs to be decided before generating the matrix. This decision may be affected by the requirements, objectives or the volume of data. For example, if the number of architectures is not sufficient, the component level may need to be set much deeper to enrich the component matrix. If three levels of components are extracted, it means main components, sub-components and sub-sub-components should be extracted. If plenty of DWHAs are collected, maybe it is sufficient to extract two levels of components to classify architectures, but the granularity for the component extraction should depend on situations or requirements.

4.1.4 Phase IV: Architecture Classification

This phase is the key step to identify the similarities and differences of DWHAs through analysing the component matrix. The mechanism of this method is designed to firstly retrieve a portion of data to build an initial classification model and iteratively examine it via rest of collected data. In order to strengthen the reliability of this classification, this classification should be structurally steady in at least two consecutive iterations. If the classification is firstly confirmed that there is no change compared with the last result, an extra iteration should be conducted to secondly examine its stability. If the classification model does not change in these two consecutive iterations, then the classification can be confirmed and proved consistent and reliable. In this design, a problem arises on how many iterations

are required and expected to generate this double confirmed classification, which should be determined before allocating collected data for iterations. In order to analyse the expected iterations, its mathematical expectation is expressed and explained in the following equations derived from [66].

$$E(x) = x_1 \cdot p_1 + x_2 \cdot p_2 + ... + x_{n-1} \cdot p_{n-1} + x_n \cdot p_n \Rightarrow E(x) = \sum_{i=1}^{\infty} x_i \cdot p_i \quad (4.1)$$

$$E(x) \leq N - M \quad (4.2)$$

$$x_n p_n = \begin{cases} n \times 0, & \text{if } n = 1 \\ n \cdot \frac{Fibonacci[n-1]}{2^n}, & \text{if } n \geq 2 \end{cases} \Rightarrow E(x) = \sum_{n=2}^{\infty} n \cdot \frac{Fibonacci[n-1]}{2^n} \quad (4.3)$$

If x_i and p_i are set to present a certain number of the iterations and the possibility to match the ending conditions in each iteration, respectively. The mathematical expectation of the iterations *E(x)* can be presented in equation 4.1. x_i is a set of continuous natural numbers starting from 1 (e.g. 1, 2, 3, 4, etc.). While p_i is changeable, which is influenced by the confirmation times. In other words, the confirmation times indirectly affect the mathematical expectation of the iterations. If N is set to the number of segments in which the collected architectures are randomly distributed into; M is allocated to show the number of segments for developing the initial classification; the number of iterations can be calculated by *(N - M)*, which should be adequate and not less than the mathematical expectation of the iterations *E(x)*. The relationships among N, M and *E(x)* can be formulated in equation 4.2. When examining the classification using a testing folder, there are two possible results (changed or unchanged), if both of their possibilities are equal to 50%, the formula of the general term in the *E(x)* can be presented in equation

4.3. Then when n tends to be infinity, according to [66] and calculated by [274], the number of the expected iterations is 6, which means the double continuous confirmations may happen during 6 iterations. For example, if the collected data is divided into 10 segments, it is better to leave at least 6 segments for testing the initial classification and strengthening it until obtaining a consistent and reliable classification. In order to generate the classification without much manual intervention, this method is designed with a set of steps in a program. In order to better explain this program, an example of the process flow is illustrated in figure 4.1.

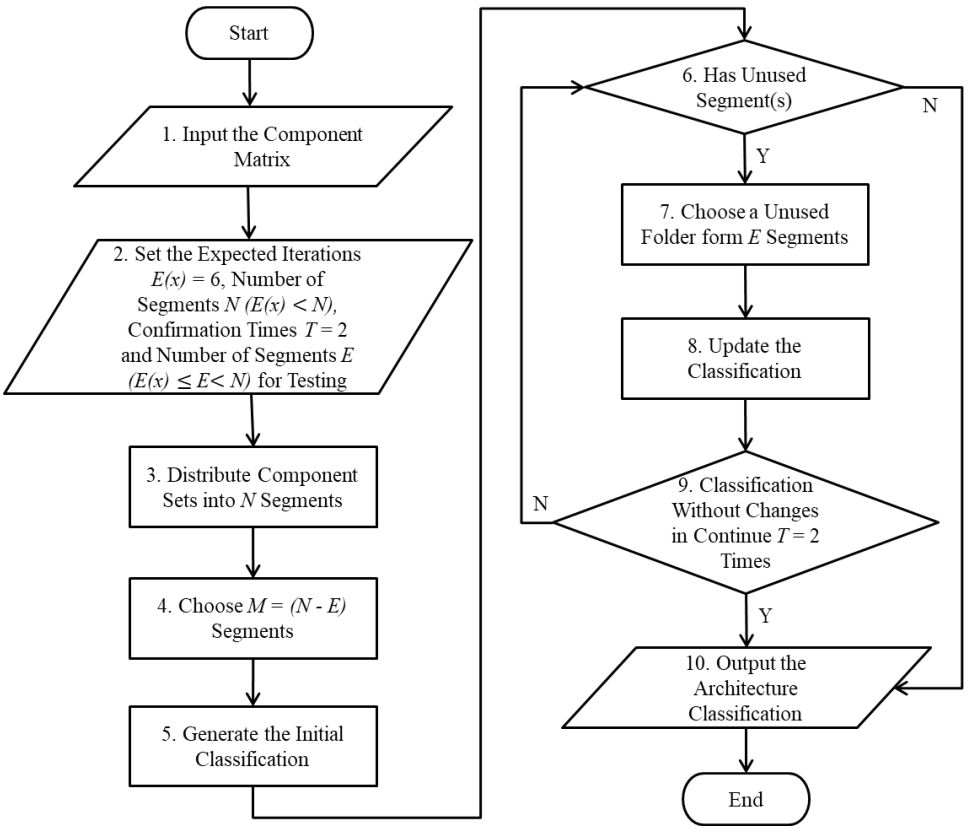


Figure 4.1: The process flow of the phase IV in the classification method

As can be seen, the processes of producing the architecture classification are divided into 10 steps. The first step is for inputting the component matrix into this program. In step 2, the theoretical expected iterations $E(x)$ is set to 6; the number of segments is set to N; the confirmation time T is set to 2, which means

the classification should be consistent and reliable in two continued iterations; the number of segments for iterations of real tests is set to E. After this, the architectures need to be randomly distributed into N segments in step 3. Step 4 is to choose M segments for generating an initial classification. Thus, the initial version of the architecture classification is formed in step 5 using some algorithms (e.g. hierarchical clustering, K-means etc.). Step 6 adjusts whether there is an unused segment left for testing or not. In step 7 and 8, the program flows into a loop to test and update the current classification using an unused segment each time. The updated classification should be compared with the last classification. If the current classification fits the ending condition of two continuous confirmations without changes, the program flows to step 10, then outputs the classification and terminates. If the classification does not fit the ending condition and there are still some testing folders are available, then this program goes back to step 6. If no segment can be used for testing, then it will go to step 10 and output the architecture classification, then terminate this program.

4.2 Method Evaluation for Data Warehouse Architecture Classification

This section is to evaluate the validation of this proposed method and demonstrate how to execute it to classify DWHAs. In this research, 116 publications are collected from multiple sources. By removing duplicative DWHAs, 68 disparate DWHAs are finally left and modelled by ArchiMate, then they are digitised and grouped into a classification. This classification can guide researchers and practitioners to identify, analyse and compare differences and trends of DWHAs from a component perspective. The details of these four phases are described below.

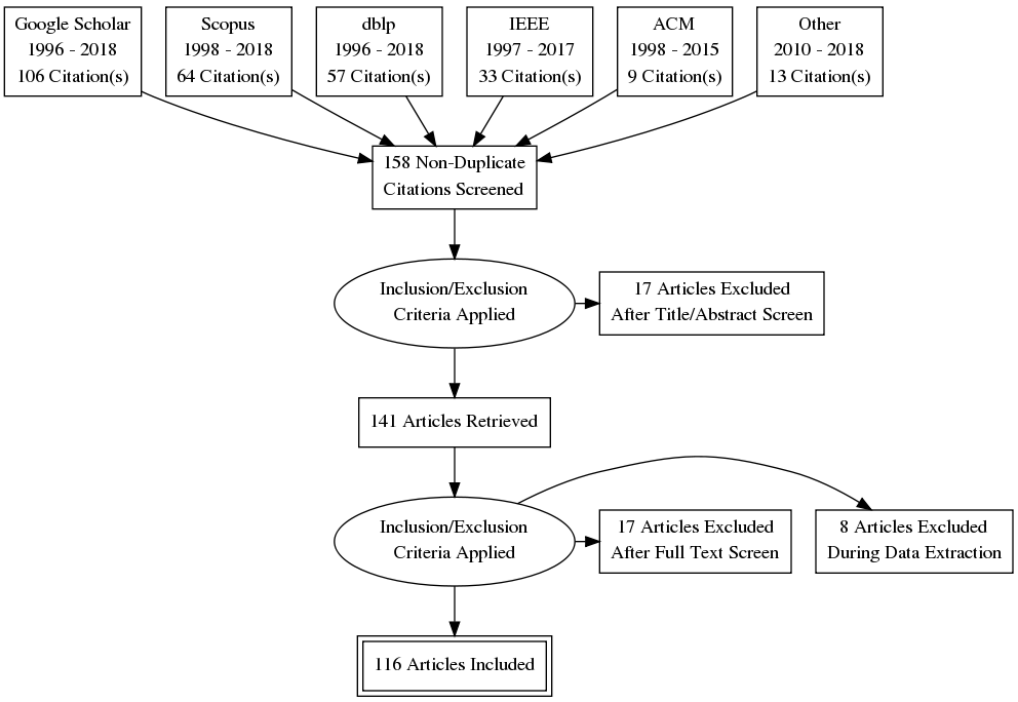


Figure 4.2: The literature review process flow for collecting DWHAs

4.2.1 Data Warehouse Architecture Collection

In order to collect DWHAs, a meta-analysis LR is conducted to review and analyse DWHAs. This LR method can systematically generate results from the literature in order to better understand theories or prove observations [146]. The processes of this LR are organised into 5 main steps based on [221]: (1) observing some phenomena or problems that some new DWHAs are not included in the previous DWHA classifications and how to generate a consistent and reliable classification; (2) identifying the search terms to retrieve related publications from the literature; (3) searching at least two different electronic databases by searching the terms identified in step 2; (4) screening their titles and abstracts to distinguish potentially eligible for inclusion; (5) analysing full text of related articles. The related online resources (Google Scholar, ACM, dblp, IEEE and Scopus) are selected hereby. The key word of the "Data Warehouse Architecture" is searched through these resources. In addition, some publications involving "Data Lake Architecture"

and “Big Data Warehouse Architecture” are added as supplementary materials. Even though there are plenty of papers that can be found in the literature, for some reasons, a large number of them are not accessible. Based on [158, 199], the results of these processes in this LR are demonstrated in figure 4.2. As can be seen, initially, 282 potential publications are searched from 6 online sources. When accessing and downloading them, only 158 publications are available. By screening their titles and abstracts, 141 articles are remained. By further screening full text and data extraction, 116 articles are finally identified and left for the further investigation.

4.2.2 Data Warehouse Architecture Modelling

After these DWHAs are collected, they should be restated by modelling from a component perspective in order to further conduct the analysis and classification. There are different modelling languages applied to diagram architectures, such as unified modeling language (UML), business process modelling and notation (BPMN), and ArchiMate [127]. Compared with ArchiMate, UML and BPMN show narrow fields, which more focus on software and business processes, respectively [129]. ArchiMate is partly based on the concepts of the IEEE 1471 standard [92] and supported by various tool vendors and consulting firms [171]. ArchiMate is the open source modelling toolkit for all levels of enterprise architects and modellers [17]. This modelling language can be combined with TOGAF [128] and it is widely used to model and implement architectures [213]. Hence, in this research, ArchiMate is used to model the collected DWHAs.

In terms of the modelling, some DWHAs have already been modelled by using other or self-defined modelling languages. In order to unify them, they are remodelled using ArchiMate. As some collected DWHAs are only briefly described and diagrammed, in this circumstance, it is necessary to understand these architectures by interpreting their related contents, extracting components and modelling them. Some collected articles are discarded, in which DWHAs are duplicated

or cannot be interpreted and analysed even reading contents. As a consequence, 68 DWHAs are modelled using ArchiMate in this research, Three of them are presented in figure 4.3. These modelled DWHAs can facilitate to observe and analyse similarities and differences between them. In total, 26 symbols and two relations provided by ArchiMate are used in this research to diagram DWHAs, in which most of these used symbols are mainly from the core layers (Business, Application, Technology) and these two relations present data flows in DWHAs.

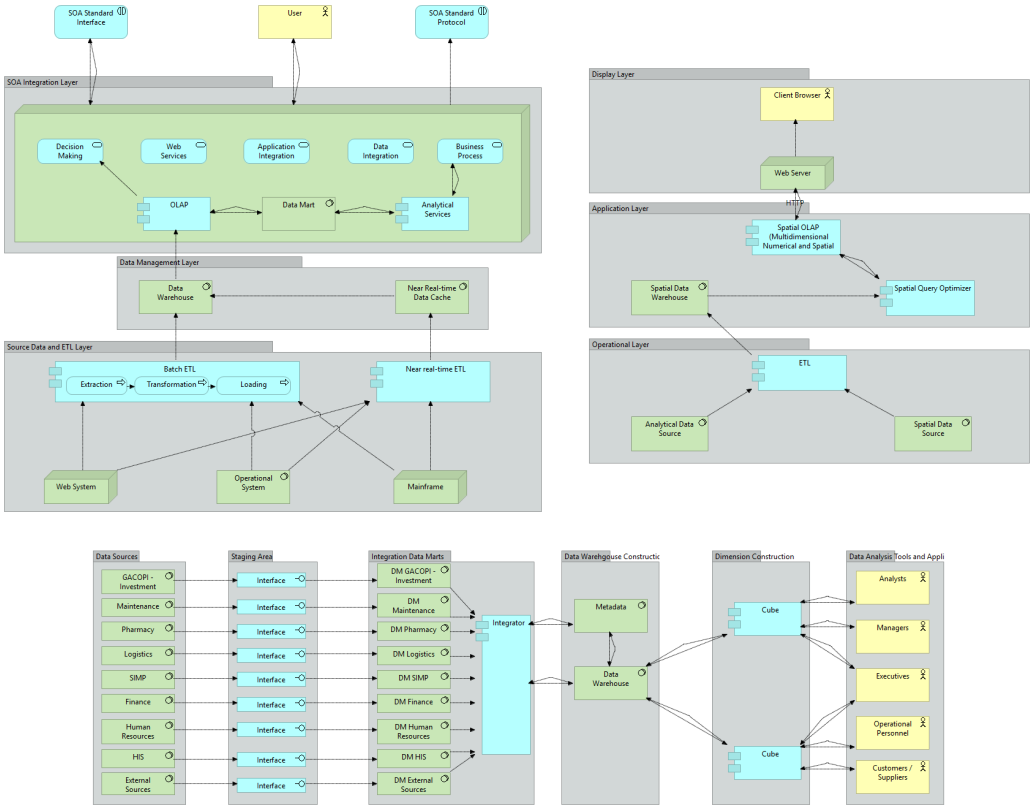


Figure 4.3: Three samples of modelled DWHAs

4.2.3 Data Warehouse Architecture Digitisation

This phase is to convert these modelled DWHAs from graphical format to digital format which is the machine-readable pattern for further analysing them and automatically generating the classification. The detailed processes are designed

below. The first step is retrieving a modelled DWHA, extracting components and marking the related slot in the matrix if a component already exists in the matrix, or adding a new column with the new component if this component does not exist in the previous DWHAs. Some examples of DWHAs are given in table 4.1 below. For example, in this table, the first column is the ID of DWHAs. The first DWHA contains structured data, then putting 1 into the related slot of the structured data, if the column of the structured data is not included while the first DWHA has this feature, then adding this feature in this table. Before conducting these processes, the component level should be predetermined, as a component may contain sub-components, and these sub-components may incorporate sub-sub-components. Hence, the component level should be consistent to execute the extraction from different hierarchical components depending on the purposes and requirements. In this case, the component level is set to 2, which means a component and its sub-components are considered and added into the matrix, if its sub-components have sub-sub-components, these components in the third level are not taken into account. Therefore, in the matrix, each row stands for a modelled DWHAs, if a DWHAs have several components and sub-components, these related slots are assigned with 1, if this DWHA does not have some components but others have, then the related slots are assigned with 0.

DWHA ID	Structured Data	Unstructured Data	E	EL	...	Meta Data
1	1	0	0	1	...	0
2	1	0	0	0	...	0
3	1	0	0	0	...	0
4	1	0	0	0	...	0
5	1	0	1	0	...	0
...	...	...	...	...	...	...

Table 4.1: Some samples of digitised DWHAs

4.2.4 Data Warehouse Architecture Classification

In previous phase, the component matrix is generated with these modelled DWHAs and components in each row. This phase aims to automatically derive their relationships and classify them into different clusters. As the number of clusters is unknown and there is no label which marks the group for each DWHA, this issue is referred to the unsupervised learning for clustering. There are different types of algorithms that can be applied such as hierarchical clustering, DBSCAN, K-means clustering, fuzzy clustering, association rule, etc. [27]. According to [10], choosing the clustering algorithm depends on the clustering objects and the clustering task. In this research, the algorithm of the hierarchical clustering is used to investigate the insight into the DWHA classification, because it allows people to easily observe relationships of these architectures with multiple levels. It can flexibly present the classification with hierarchical or taxonomical structure [224]. This clustering does not pre-set the number of clusters or threshold values to classify DWHAs. In other words, it requires less artificial intervention during execution.

The initial component matrix of DWHAs has 47 dimensions, and 25 dimensions are selected for further analyses because other dimensions are optional and auxiliary (e.g. reporting tool). Therefore, the component matrix is pre-processed to 68 DWHAs with 25 dimensions, then input into the program. In order to fulfil the double continuous confirmed tests and easily manage each segment, these 68 DWHAs are randomly divided into 14 segments, in other words, each segment has 5 DWHAs except the last one which has 3 DWHAs. Then, 7 segments are used to form the initial classification. The rest of 7 segments are applied for testing, which is greater than the expected number of 6 iterations. When the classification is not changed in two continuous tests or all data is used, then the final result will be output and the program will be terminated. In order to look insight into how to generate a set of classifications, an example is given with 60 DWHAs in figure 4.4.

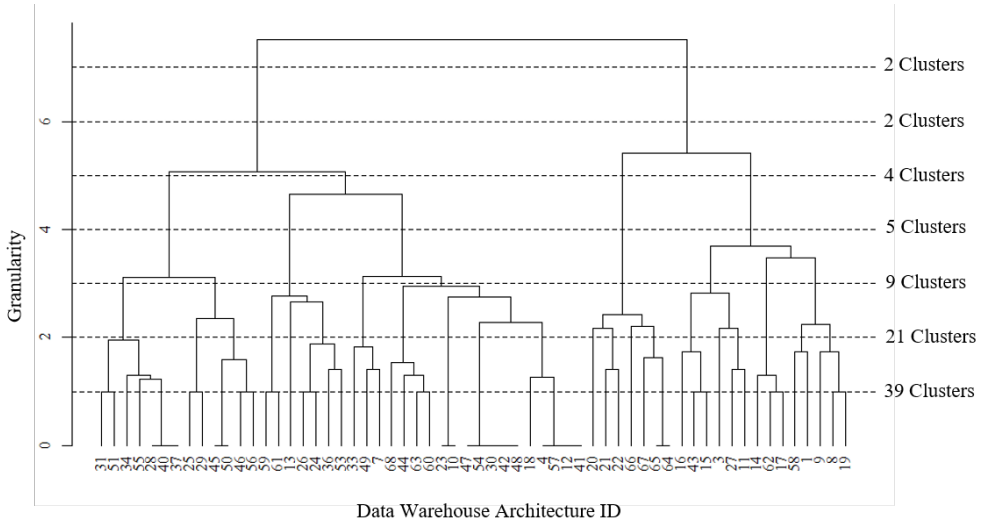


Figure 4.4: The classification with 60 DWHAs

It can be seen that the relationships between granularities and the number of clusters can be illustrated with two sets: [1, 2, 3, 4, 5, 6, 7] $\Rightarrow$ [39, 21, 9, 5, 4, 2, 2], in which the granularity is a parameter to measure the distance between clusters. For instance, if the granularity is 1, then the number of clusters is 39. In addition, when analysing the number of clusters, if a granularity line cross or near the line in the hierarchical clustering, then a balance is struck depending on the sizes of other clusters. In order to concretely observe the pathway of how the number of clusters changes along with granularities (e.g. granularity 1, granularity 2, etc.), each segment (e.g. 5, 10, 15, etc.) is measured and the results are tabulated in table 4.2.

4.3 Data Warehouse Architecture Classification Result

In order to analyse the trends of these granularities, the changes of clusters in each granularity with the different number of DWHAs are displayed in figure 4.5. The x-axis is the number of DWHA inputs with the increment of 5. The y-axis is the number of resulted clusters generated by the hierarchical clustering algorithm. Each line represents clusters under a certain granularity. By observing this figure,

Table 4.2: Clusters with different numbers of DWHAs and granularities

	5	**10**	**15**	**20**	**25**	**30**	**35**	**40**	**45**	**50**	**55**	**60**	**65**	**68**
G 1	5	9	13	15	19	21	26	28	30	32	34	39	39	41
G 2	3	5	8	11	13	14	16	17	19	20	20	21	24	25
G 3	1	2	3	4	5	7	8	9	9	9	9	9	9	9
G 4	1	1	1	2	2	3	3	4	4	5	5	5	5	5
G 5	1	1	1	1	2	2	2	2	2	2	3	4	3	4
G 6	1	1	1	1	1	1	1	1	2	2	2	2	2	2
G 7	1	1	1	1	1	1	1	1	1	1	1	2	2	2
G 8	1	1	1	1	1	1	1	1	1	1	1	1	1	2
G 9	1	1	1	1	1	1	1	1	1	1	1	1	1	1

it is expected that the low granularity has more clusters than the high granularity. For instance, the clusters of granularity 1 is greater than the clusters of granularity 2, et cetera. From top to down, the slopes of these lines are metamorphic from sharp incline to gentle incline, which implies that the low cluster granularity is harder to obtain the reliable classification than the high cluster granularity.

Therefore, when searching the stable classification, a top-bottom rule is used to observe figure 4.5. To be more specifically, it starts from the granularity 1. As can be seen, the lines of granularity 1 and 2 are sharply and synchronously increased along with the number of DWHAs, and no reliable status can be achieved. When granularity is 3, 4 or 5, the trend of lines and the number of clusters in these classifications undergo two stages from gradually increased or finely fluctuant to relatively reliable. Besides, the number of clusters in these classifications is manageable for further analysis. Whilst, if granularity is set to 6, 7, 8 or 9, the trend and number of clusters are slightly changed, which may have limited values for further analysing DWHAs from these classifications. Therefore, based on the double continuous confirmation rule described in the previous section and the top-bottom rule, the granularity 3 is further considered as a proper cluster distance and these DWHAs can be classified into 9 classes. In this granularity, after inputting 40 DWHAs, the trend of the line tends to be reliable.

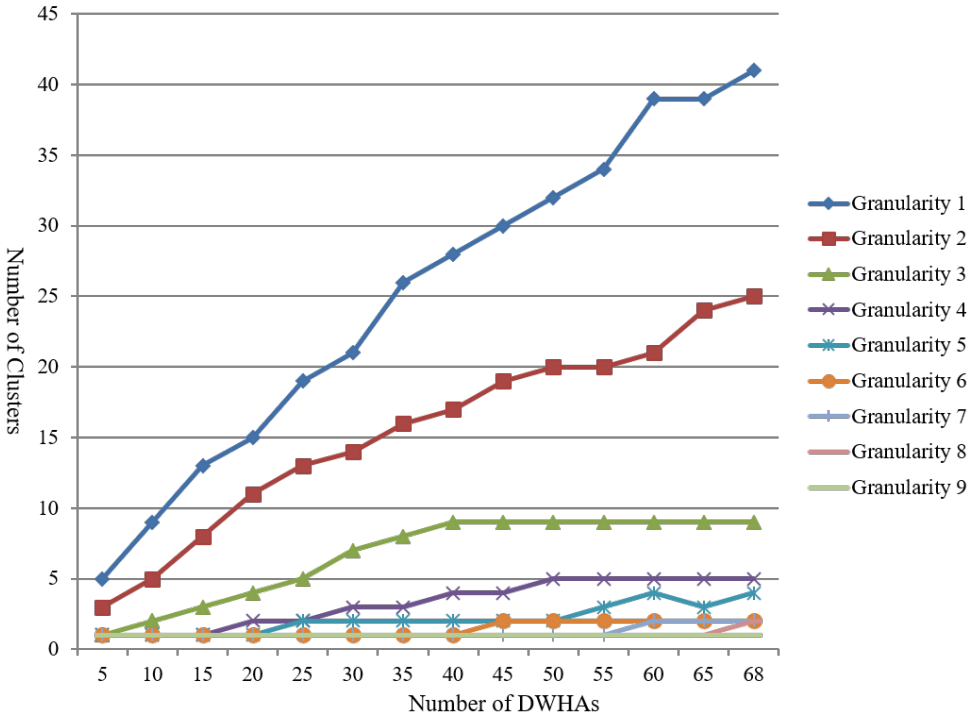


Figure 4.5: The DWHA inputs and classifications for each granularity

4.3.1 Representative Data Warehouse Architectures

As a result, these DWHAs can be classified into 9 consistent and reliable clusters when the granularity is 3. In order to investigate the characteristics of these DWHAs, they are identified and generalised by analysing their components in each cluster. When observing the DWHAs in these clusters, some clusters have a types of DWHA, while some clusters have mixed types of DWHAs. Inadequate information of some DWHAs is provided or they have many same dimensions (e.g. data stage and operational data store), which affect clustering. Subsequently, 9 typical DWHAs are identified and generalised. Some of them are included in the [18, 147], while, others (e.g. VDWA, PDWA, BDWA and DLA) are excluded. At the same time, ArchiMate is applied to model these representative DWHAs. In order to better demonstrate and distinguish the differences among them, these modelled DWHAs are generalised and merged together in figure 4.6.

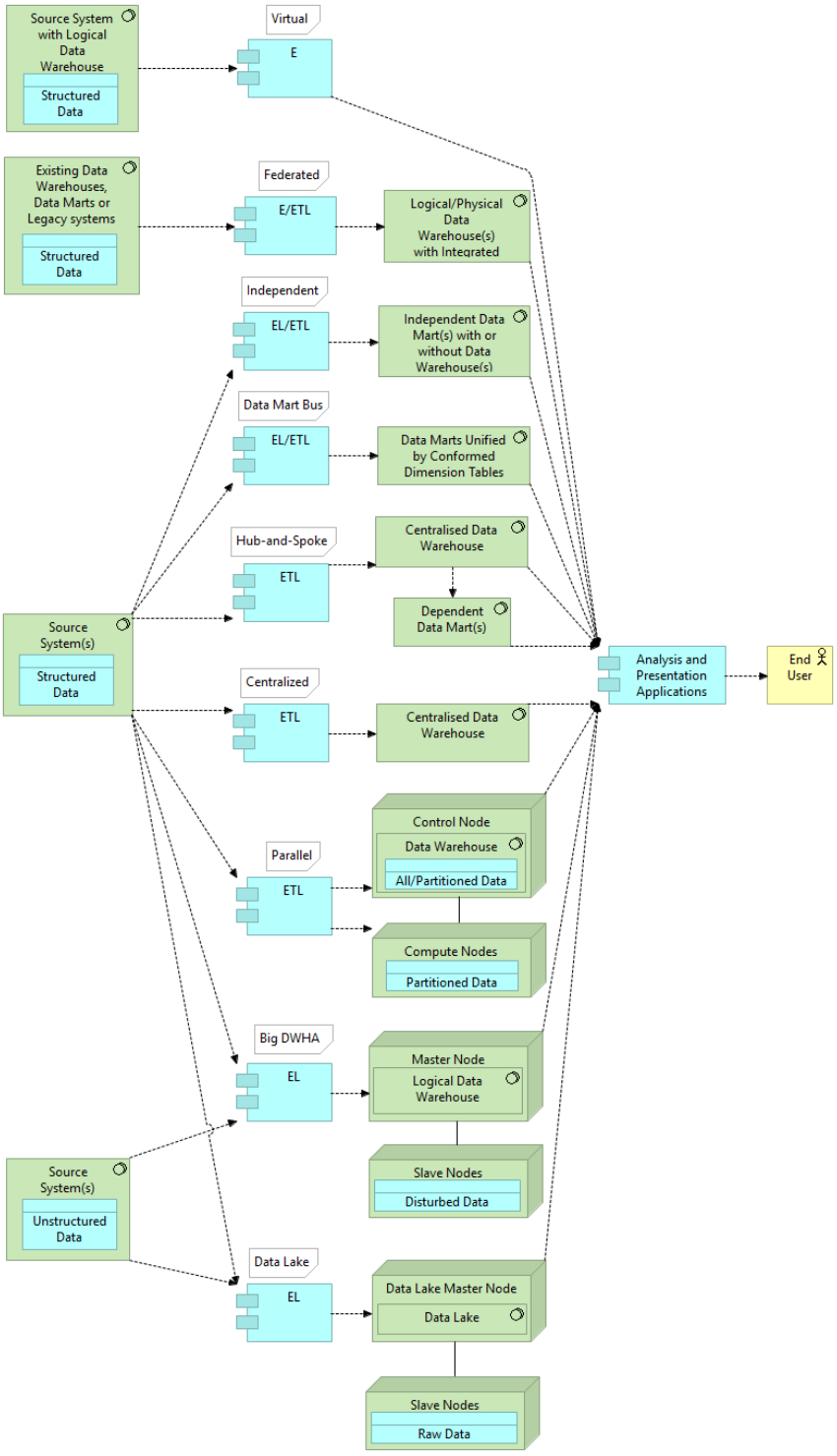


Figure 4.6: The big picture of the 9 typical DWHAs

The Virtual DWHA: The VDWH manipulates and analyses operational data sources with limited data integration functions and summary views of the DWH depending on the complexity of requirements [41]. Middleware may be leveraged as an interface between front-end tools and the operational database to provide services for end users [24]. The DWH in VDWH is a logical warehouse allowing users to directly access multiple operational data through middleware tools. It does not necessarily store copies of the operational data with different formats compared with other DWHAs [58].

The Federated DWHA: The FDWA is partly based on the bottom-up implementation, in which the legacy decision-making or other systems are integrated together to form DWHs in order to provide enterprise-wide analytical solutions [104]. The federated DWH model can be acted as the added data staging layer that enables easier implementation of analytical solutions to hide the complexities of operational data sources [11].

The Independent DWHA: In this research, the IDWA is broader rather than just the independent DMs in previous research. The independent DMs consist of physically/logically separated and irrelevant DMs, while the independent DWHA may contain hybrid and less-coupling components including DWHs and DMs. This type of architecture may be called as the DM architecture which has the independent DM(s) or distributed DWHA that contains the mixed but independent DWH(s) and DM(s) [212].

The Data Mart Bus DWHA: Ralph Kimball [119] proposes the DMBA with dimensional modelling, in which a data modelling approach is unique to the DWH. The enterprise-wide cohesion is accomplished by a data bus standard [36]. It is a bottom-up approach to develop a DWH. Conformed dimensional tables and DWH bus matrixes play vital roles to provide consistent dimensions for implementation of DMs asynchronously.

The Hub and Spoke DWHA: The HASA is suitable for traditional relational database tools to fulfil the development requirements of an enterprise-wide DWH

[36, 117]. It is a top-down approach to develop a DWH and DMs. A centralised DWH is built first, then DMs are built after this DWH. This type of representative DWHA is widely used in organisations with multiple departments to provides consistent data for analysis.

The Centralised DWHA: The CDWA collects heterogeneous data into the enterprise-level cross-functional information system, which enables the separate sources to be used together and stored with the presentation-ready format for requests from end users or further analysis. The topology of this DWH is not complicated to maintain warehouse data for many end-users and applications throughout an organisation [33].

The Parallel DWHA: The PDWA herein includes DWHs working in parallel or built on multi-node cloud computing platforms, in which data are pre-processed and physically distributed in DBMS (e.g Oracle, DB2 etc.) with a predefined schema. This architecture fragments the warehouse schema using partitioning algorithms [186] and allocates the generated fragments over the compute nodes using particular algorithms [150]. In order to gain availability of data and the high performance of the system, the generated fragments may be duplicated and sent to computing nodes for further manipulation or analysis [288].

The Big DWHA: The BDWA is built on a big data platform (e.g. Hadoop), in which the distributed file system and some other mechanisms (e.g. MapReduce) are applied to store and deal with data [279]. This DWH can address big data issues (e.g. petabyte level) for reports and ad-hoc analyses by using SQL-like language (e.g Hive) [241]. It is easily executed by people with SQL experience. This logical DWH does not need to strictly pre-process data consisting with a predefined schema. It can manage and analyse unstructured data provided by a logical DWH [278].

The Data Lake Architecture: This architecture can be leveraged directly as a centralised raw data repository providing information for further analysis [203]. The DWH and DL can cooperate together to address the big data issues

and achieve requirements in relation to data analysis e.g. [101]. As discussed in chapter 2, there are two situations referring to this combined architecture. In the first situation, the DWH populates the DL, which means the DWH is built first or a company already has a legacy DWH, then a DL is built. Another situation is that the DWH is fed by the DL . The DWH obtains data from the DL, which is suitable for building a DL followed by a DWH, optimising a legacy DWH data flows or extending a DWHA with a DL to meet intangible requirements.

4.3.2 Data Warehouse Architecture Distribution

A statistic is made to analyse the distribution of these DWHAs based on the literature, and the result is compared with the two previous research studies [18] and [11] presented in table 4.3. As can be seen, there is no big conflict among these three results. The first three popular DWHAs (HASA, DMBA and CDWA) are similar. In the first two research studies, the DMBA's ratio is greater than the CDWA's, while in this paper, the result is opposite. The reason might be observing the domain from different angles (e.g. industry or academia). According to the result, it implies that these three popular DWHAs are widely accepted in industry and academia. When choosing or building DWHAs, these three architectures should be concerned. But it does not mean that other DWHAs do not need to be pondered, especially in some cases that only simple DWHs or DMs, even virtual DWHs are required and can fulfil requirements. In addition, most of these papers referring to DLAs have been published since 2015, so it shares less proportion in this statistic. If adding the PDWA, BDWA and DLA as a whole, these three architectures' ratios are high and cannot be neglected, which may imply that big data issues are increasingly significant and inescapable in the context of DWHAs. Hence, when designing and developing DWHAs, it is necessary to consider these three big data oriented architectures.

Table 4.3: The DWHA distribution

	DWH Architecture Distribution								
	Hub and Spoke	Data Mart Bus	Centralised	Independent	Federated	Virtual	Parallel	Big DWHA	Data Lake
[18]	39%	26%	17%	12%	4%				
[11]	35%	23%	20%	15%	7%				
This Study	29%	9.3%	27%	8.7%	6.4%	2.9%	8.7%	3.5%	5.2%

4.4 Data Warehouse Architecture Classification Discussion

There are two main DWHA classifications based on the literature. The layer-based classification is general and abstract, which gives an outline of architectures. The component-based classification is derived from cases in practice. However, they are some inter-relations between them. Some DWHAs (e.g. HASA, DMBA and CDWA) may need a ETL tool to extract, transform and load data. The FDWA is special, which may not need ETL tools/staging areas, while its legacy DWHs or DMs may obtain pre-processed data from ETL tools. In order to build the logical/-physical federated DWH, these DWHAs or DMs should be reconciled to provide cleansed and integrated data. Therefore, the four architectures (HASA, DMBA, CDWA and FDWA) have the similar structure as the three-layered DWHA has. The IDWA is extended from the concept of independent DMs and it undergoes different situations: If it consists of independent DMs for individual department applications, then it may belong to traditional two-layered DWHA; If it is mixed with IDWs and DMs, it belongs to the three-layered DWHA as it may need to reconcile data from different sources. Therefore, the relationship of these two main classifications (layer-based classification and component-based classification) can be illustrated: Two-Layer $\supset$ {Independent [DM]}; Three-Layer $\supset$ {HASA, DMBA, CDWA, Independent [DWH with DM] and FDWA}.

Table 4.4: The DWHA Classification Model

DWHA Classification Model		Data Warehouse Architecture									
		Traditional DWHA							Big Data DWHA		
		One Layer	Two Layer	Three Layer					Big		Syn.[12]
		Virtual	Independent [DM]	Independent [DWH with DM]	Centralised	Data Mart Bus	Hub-and-Spoke	Federated	Parallel	Big DWHA	Data Lake
Data Type	SD[1]	✗	✗	✗	✗	✗	✗	✗	✗		
	S&UD[2]									✗	✗
Schema	Sin.[3]	✗	✗								
	Mul.[4]			✗	✗	✗	✗	✗	✗	✗	✗
ETL[5]	E	✗						✗			
	EL		✗			✗					✗
	ET										
	ELT									✗	
	ETL			✗	✗		✗		✗		
Data Mart	CDM[6]					✗					
	IDM[7]		✗	✗							
	DDM[8]						✗				
	Unk.[9]	✗			✗			✗	✗	✗	✗
Storage	Dis.[10]								✗	✗	✗
	Agg.[11]	✗	✗	✗	✗	✗	✗	✗			

[1] Structured Data
[2] Structured & Unstructured Data
[3] Single Schema
[4] Multiple Schema
[5] Extract, Transform, Load
[6] Conformed Data Mart
[7] Independent Data Mart
[8] Dependent Data Mart
[9] Unknown or Null
[10] Distributed Storage
[11] Aggregated Storage
[12] Synthesis

4.4.1 Data Warehouse Architecture Classification Model

According to the method, a new classification model of DWHAs is generated based on their components. To further demonstrate them, they are organised into

a hierarchical structure presented at the top of table 4.4 with bold font. It can be observed that the model has four levels: the first level is the root of the DWHA; the second level has two branches, traditional data and big data oriented architectures; the third level contains the abstracted architectures; the fourth level is the actual or concreted architectures used in practice. Nevertheless, the DL in the last column can cooperate either with or without DWH(s), so only the DL is presented in this model.

4.4.2 Data Warehouse Architecture Classification Model Discussion

Even DWHAs are classified in this model, there are still some indistinct characteristics or similar components which are indistinguishable. In order to identify and explain differences of these DWHAs, five features and fifteen sub-features are chosen and discussed, which are derived from generalised and modelled DWHAs in figure 4.6. Some typical DWHAs collected from the literature are selected and discussed as cases hereby. The table 4.4 under the classification model analyses and summarises differences of these cases. The ✗ mark in this table means that a certain DWHA has this sub-feature or can address this sub-feature. For example, the VDWA can manipulate structured data and then a mark is placed in the related cell; the BDWA has abilities to easily operate structured data and unstructured data, then a mark is drawn in the S&SU cell. It does not mean the other DWHAs without marks in the S&UD row cannot address unstructured data, they can operate unstructured data under some assistance, but they are not originally designed for this type of data especially in these cases. This situation may fit other features and sub-features as if choosing different or ad-hoc DWHAs which may present different characteristics.

From this table, differences can be easily analysed from a feature perspective. It can be beneficial to provide useful information in relation to investigating DWHAs. For example, most of DWHAs are normally not designed to process unstructured data except the BDWA and DLA. Hence, if a company needs to anal-

yse the large amount of unstructured data directly, these two architectures may be more suitable than others. The differences can be analysed from architecture perspectives. The CDWA and HASA are very similar, but in the feature of the "Data Mart", the HASA has DM(s), while the CDWA does not necessarily need DM(s). Hence, if a company only needs an enterprise DWH, then the CDWA is chosen and developed. If dependent DMs are required, the architecture can be upgraded to the HASA. Besides, in order to distinguish the differences and better understand these DWHAs, some DWHAs are analysed together, as they have similar components or in the same branches. The details are presented as follows.

VDWA, FDWA and BDWA: As can be seen in figure 4.6, the VDWA, FDWA and BDWA can contain a logical DWH, but based on table 4.4, the VDWA generally manages data with a single data schema in a data source system, while, the FDWA and BDWA normally extract data from multiple data sources. In addition, the BDWA stores data in the distributed file system (DFS); the FDWA may store and retrieve data from different systems, but they have different topics and are not tightly connected unlike the DFS.

Independent [DWH with DM], CDWA, DMBA, HASA and FDWA: According to the classification model, the CDWA, DMBA, HASA, and FDWA are categorised in the three-layer branch. The differences can be discerned through the "ETL" and "Data Mart" features. For example, the DMBA has separated DMs but connected with conformed dimension tables; the HASA has depending DMs, but these DMs do not need to be strictly linked by conformed dimension tables. The independent [DWH with DM] has DWH(s) and DM(s), but they are independent.

PDWA, BDWA and DLA: The PDWA, BDWA and DLA belong to the same main branch (big data DWHA). The data in the BDWA can be extracted, loaded and transformed, as the BDWA has powerful computing platforms (e.g. Hadoop) to manipulate data. In addition, its big DWH (e.g. Hive) may not necessarily pre-load data, which only needs to be informed directories of data stored in DFS.

However, data in the PDWA may need to be extracted, transformed then loaded (ETL) into a DWH with a predefined schema. Besides, the DL in the DLA stores row data with detailed information, which extracts and loads (EL) row data from sources with less pre-processing. It normally needs extra components or functions to analyse data, which more focuses on storing, maintaining and providing raw data to other systems.

4.5 Conclusions

In this part, a method is proposed to produce a consistent and reliable classification model. This method contains four main phases including data collection, architecture modelling, architecture digitisation, and architecture classification. The architecture classification is the critical and complicated phase, which is further designed and executed with step-by-step program processes. In order to demonstrate the proposed method, a case is leveraged to produce a consistent and reliable DWHA classification. During this case, 116 publications are collected from the literature and 68 disparate DWHAs are modelled using ArchiMate. These modelled DWHAs act as materials to run this method, then 9 representative DWHAs are identified and further summarised into the "big picture" for distinguishing their similarities and differences. A statistic is conducted to analyse the distribution of these DWHAs. Furthermore, a new DWHA classification model is proposed to better identify, analyse and compare different DWHAs from featured and structural perspectives.

This proposed method provides the ways to digitise architectures from diagrams to machine-readable patterns and automatically measure them for further classifications. It makes architectures more comparable especially from feature perspectives. In terms of the generalisation, it is executed to generate the classification of DWHAs without restriction or pre-condition. If architectures from other domains can be modelled and digitised, then they can be classified via this method. Hence, it is possible to develop consistent and reliable classifications for

other types of IS architectures through this method. However, extra work should be conducted to test its generalisation.

5 Result of SRQ 2 - Method for Identify Data Warehouse Architecture Features

This chapter is related to SRQ 2 and three DSRM phases (design & development and demonstration). It is aimed at designing and developing a method to identify reliable features across different DWHAs and generate structural knowledge of a taxonomy for further investigation. This taxonomy is validated by measuring the DWHAs derived from industry. Some features in this taxonomy are further used to form another method for DWHAs evaluation explained in following chapters.

5.1 Features and Taxonomy Method Design and Development

This method is designed with three phases which are articulated in chapter 3. The data for extracting reliable features is collected and prepared in the first phase. Reliable features are identified in the second phase through a set of processes. After these features are acquired, they are further manipulated in the third phase as input and sub-features are generated as output. Subsequently, these features and sub-features form an initial taxonomy. This taxonomy is refined by experts to better demonstrate these features and sub-features. The refine process is not included into the three phases, as it is designed as an optional part depending on requirements. The detailed descriptions are illustrated below.

5.1.1 Phase I: Data Collection

In order to precisely identify these features with adequate data supported, a Meta-analysis LR is conducted based on [221, 262]. By searching related literature, there is not similar systematic review has already been operated to answer this research question, so it is worth to conduct this LR. The main processes of this LR are listed below:

1. The first step aims to generate the research question or purpose based on the requirements of research.

2. The research question or purpose is broken down into individual concepts to create search terms.

3. When searching relevant publications, the scope of time-frame should be determined (e.g. years from 2000 to 2019).

4. The search should be taken in at least two different electronic databases using the terms identified in step 2.

5. Identification is made by reading the title and abstract to distinguish potentially eligible for inclusion.

6. Examination is made by full text screen.

5.1.2 Phase II: Feature Identification

Phase II has a program with the steps from 1 to 8 which are designed and explained as follows: (1) The papers collected from the LR are input in this program and the condition of top T nearest terms are determined. The T is set for Phase II, which may be a number (e.g. 5 or 10) or a threshold of the similarity between a feature and its relevant terms. (2) As the algorithm of TextRank is used to manipulate

and analyse a paper each time, this step randomly retrieve a paper each time to prepare data for following steps. (3) TextRank is executed to iteratively identify features from each paper; (4) After potential features are extracted, they are added into a pool which includes all potential features generated by previous iterations. Furthermore, the frequencies of these features are calculated to make or update a frequency model. This frequent model will be leveraged to obtain top K features in step 8 and identify the importance of top T terms in step 11. (5) In order to analyse whether top K features are reliable or not, the changes of the top K features are tracked and saved. For example, if K is set to 5, in the i iteration, top K features are $TF_{(k,i)} = \{f_a, f_b, f_c, f_d, f_e\}$, in the i+1 iteration, the top K features are $TF_{(k,i+1)} = \{f_a, f_b, f_c, f_f, f_g\}$. Then the identical features in these two iterations are $TF_{(k,i)} \cap TF_{(k,i+1)} = \{f_a, f_b, f_c\}$, in which the number is 3 and the change can be calculated to 2 which is 5 minus 3. An example is given below to concretely explain this process. The changes of K are saved in a set in order to easily compare them and facilitate step 7 for finding the top K stable and reliable features. (6) This step is to check if all papers are manipulated. (7) K is determined and input into this program based on the results in step 4 and 5. (8) The top K reliable features are selected according to the input in step 7.

5.1.3 Phase III: Taxonomy Generation

Phase III includes the steps from 9 to 16 to generate the taxonomy with sub-features based on the reliable features identified in Phase II. These steps are designed as follows: (9) This step executes Word2Vec which is an algorithm to build a model for managing similarities between terms. These terms include reliable features and a massive of sub-features which are relevant to the reliable features. (10) After this similarity mode is built, the top T nearest terms are identified by search each reliable feature through the model. In this step, the T should be defined to a certain number (e.g. 5, 10, 15 etc.) or even a threshold of the similarity between two terms. (11) As some of relevant terms may be not inconsequential,

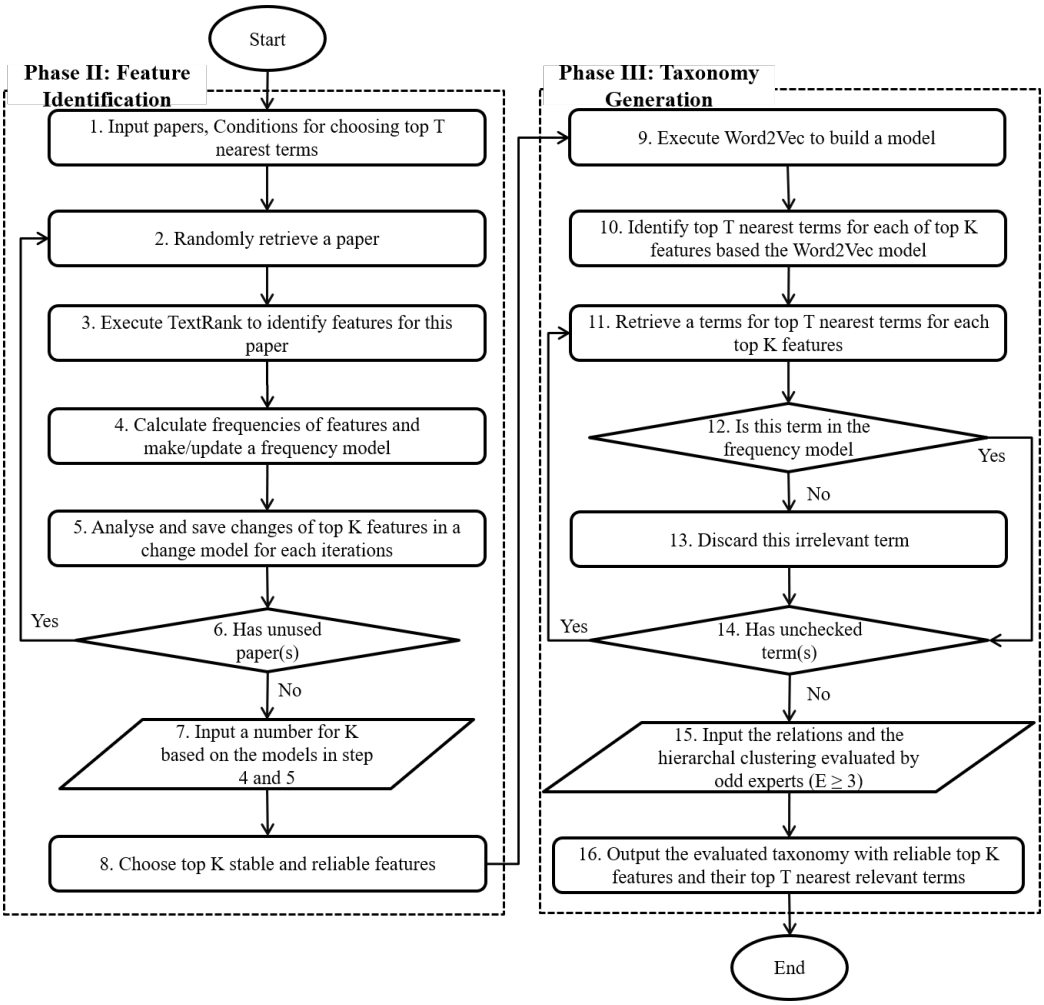


Figure 5.1: The processes for reliable features and the taxonomy

each of them should be estimated by the frequency model managed in step 4. So this step retrieves a untested term each time to prepare data for following steps. (12) This step examines if a term fulfils a threshold or other requirements. For instance, the condition can be set as whether the term can be found in the frequency model or not. (13) If a term' frequency does not fulfil a threshold or other requirements, then discard it. (14) If all terms have already been checked, then the program flows to next step, otherwise, reverses back to step 11. (15) At this stage, the initial taxonomy with top K features and their relevant terms are generated. It is necessary to evaluate it with experts to further discard unimportant and

irrelevant terms. It encourages to conduct this step with odd number of experts. In order to guarantee the quality, the taxonomy should be evaluated or ameliorated by at least 3 experts. (16) This is the last step to output the hierarchically structured and evaluated taxonomy with reliable features and relevant terms.

5.2 Method Evaluation with Data Warehouse Architectures

This method is evaluated by identifying reliable features and generating a taxonomy for DWHAs. In this research, 220 papers are collected in Phase I. The reliable features are identified and the taxonomy is generated in Phase II and III respectively. The details of how to conduct these three phases are demonstrated as follows.

5.2.1 Data Warehouse Architecture Feature Collection

As the purpose and research question are proposed in chapter 1, the terms for searching are extracted from them, which include "data warehouse", "data warehouse architecture" and "data warehouse evaluation", "data warehouse feature" and some synonyms of the feature explained in chapter 2 (e.g. characteristic, attribute, property, component, element). As for the time scope of publications, it tracks to relevant topics firstly appear in the literature, which covers the time from 1996 to 2019. The publications are originally extracted from multiple sources including IEEE, dblp, ACM, Scopus, Google Scholar and other online resource listed below. Initially, 611 articles are identified from these sources, which are potentially relevant to this research. By eliminating duplicates, 332 articles are available and downloaded. After screening titles, abstracts and full-contents, 220 publications are remained for further investigation. The detailed processes of this LR are illustrated in 5.2 figure below.

- https://scholar.google.com

- https://dl.acm.org
- https://dblp.org/search
- https://ieeexplore.ieee.org/Xplore/home.jsp
- https://www.scopus.com/search/form.uri?display=basic
- https://www.google.com

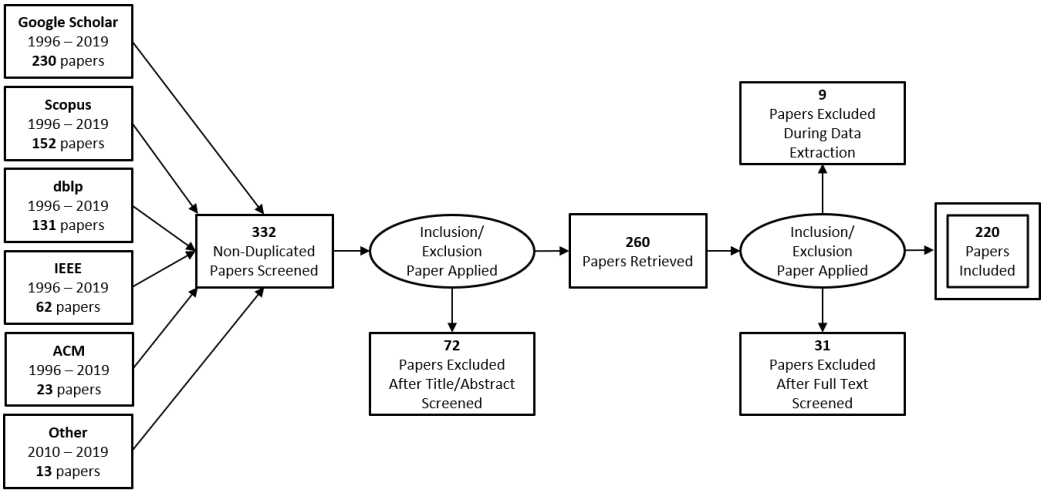


Figure 5.2: The literature review flow diagram for DWHA features

5.2.2 Reliable Feature Identification

In phase I, one of the most important steps is step 7 as it needs to input a number for K in order to choose top K features from the frequency model. It is necessary to concretely discuss how to analyse and determine the K and choose top K frequent features. In this research, 220 papers are iteratively analysed by TextRank. In step 4, a frequency model is managed and arranged by the descending order based on their frequencies, which is upgraded after a new features generated. For instance, if $FM_{(k,i)}$ and $TF_{(k,i)}$ stand for the frequency model and top K features after the

i iteration. (f_a, n) is a feature and n is the total frequencies of this feature so far. If K is 5 and their changes and relationships in the i and i+1 iterations can be illustrated in equation (5.1), (5.2) and (5.3) below. In step 5, the change of top K features are analysed and saved in each iteration. If CF_k is a set of changes for top K features in each iteration, $c_{(k,i)}$ is the number of the changed top K features after the i iteration. Then their relationship can be presented in equation (5.4) and the $c_{(k,i+1)}$ can be calculated based on the $TF_{(k,i)}$ and $TF_{(k,i+1)}$.

$$FM_{(k,i)} = \{(f_a,30),(f_b,26),(f_c,25),(f_d,24),(f_e,23),(f_f,23),...\} \tag{5.1}$$

$$FM_{(k,i+1)} = \{(f_a,31),(f_b,27),(f_c,25),(f_d,25),(f_f,24),(f_e,23),...\} \tag{5.2}$$

$$\Rightarrow TF_{(k,i)} = \{f_a,f_b,f_c,f_d,f_e\};\ TF_{(k,i+1)} = \{f_a,f_b,f_c,f_d,f_f\} \tag{5.3}$$

$$CF_k = \{c_{(k,i)}|K - Count(TF_{(k,i-1)}) \cap TF_{(k,i)})\} \Rightarrow c_{(k,i+1)} = 1 \tag{5.4}$$

5.2.2.1 Stable Position Identification

In order to identify reliable features, it is necessary to investigate how many iterations can stabilise top K features without changes. To be more specific, it needs to find that $c_{(k,i)}$ and its subsequent items/changes should be stable to 0. Therefore, in order to analyse their relationship, a set of experiments are conducted to calculate each $c_{(k,i)}$ for CF_k with different K from 1 to 60 based on the equation 1, 2, 3 and 4. In these experiments, stable positions for each top K features are found by analysing CF_k and some samples are given in figure 5.3. When K are 5, 10, 15 and 20, their top K features are stable after 129, 220, 157 and 216 iterations. It

indicates that top 5 and 15 features are reliable to be retrieved after 129 and 157 iterations as they are stationary and not easy to break their reliable statuses. In this paper, these circumstances are called the **Steady State** under this volume of collected papers and their top K features are reliable to be retrieved. When K is 10, top 10 features are volatile and converted until the last iteration. By further analyses its neighbours (K = 9 and 11), the features on the 9^{th}, 10^{th} and 11^{th} positions are exchanged in the **Competitive State**, which affect the stability of these top K features. The top K features in the competitive State are not reliable. In addition, when K is 20, the stable position/iteration is 216 which is reaching to the last iteration (220). It is uncertain to predicate that these top 20 features are reliable after 216 iterations as these features are proved stable only with 4 iterations, which is hereby called the **Uncertain State**. These three states may occur in any top K features.

5.2.2.2 Trend of Stable Positions

After analysed all reliable position for each top K feature, it is still unclear to distinguish which state does a certain K belong to. If the top K features are identified in the steady state, it will be useful to retrieve reliable features from these top K features. In addition, if a certain number of papers are collected, It will be helpful to know how many reliable features can be extracted from these papers. If the relationship between top K reliable features' state and the trend of $CF_{(k)}$ can be interpreted, then it is possible to identify steady states and determine how many reliable features can be obtained under a certain number of iterations/papers. Therefore, the reliable positions with top K features are diagrammed in figure 5.4. In this figure, x-axis and y-axis stand for top K features and stable iterations, the points mean K features are stable after a certain iteration. For instance, the point (5, 129) means top 5 features are stable after the 129^{th} iteration. After putting all points on the canvas, the distributions of these points show a trend, which is analysed and presented as the curve in this diagram. If $SP_{(k)}$ means the stable position

for a certain K, this curve can be expressed as a natural exponential function in equation (5.5), in which e is a mathematical constant and approximately equals to 2.71828.

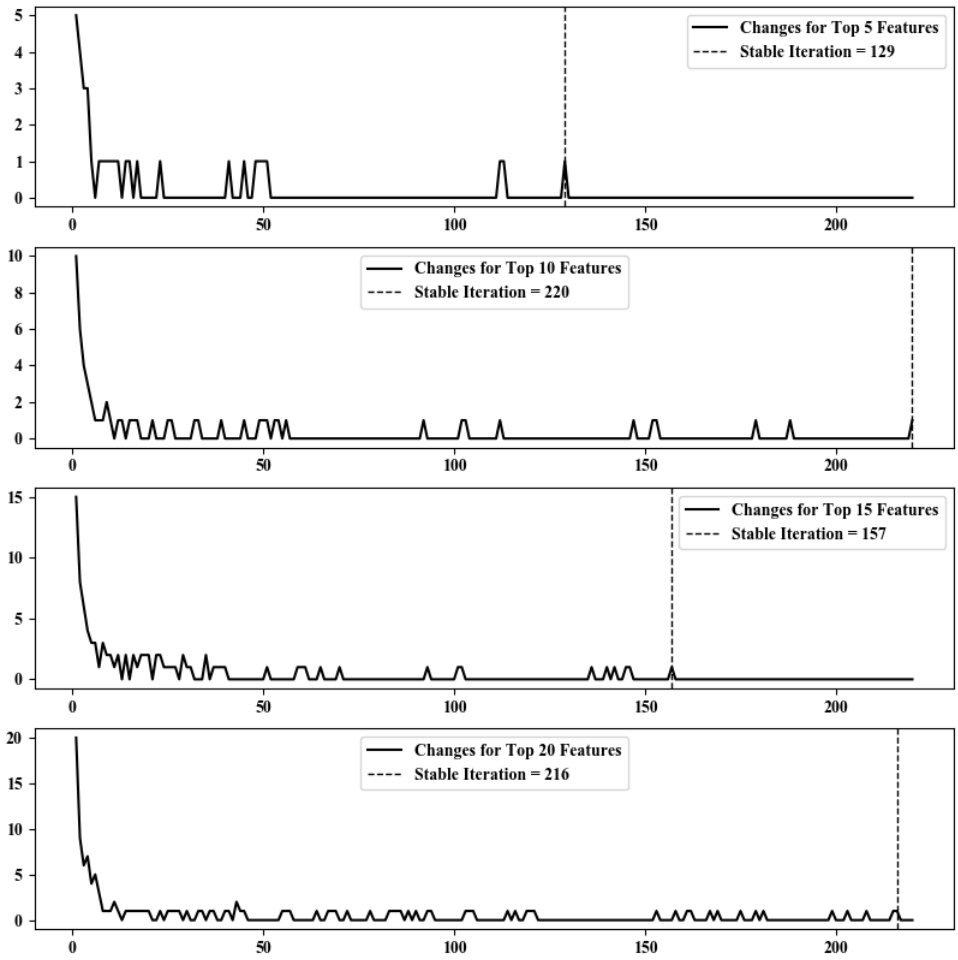


Figure 5.3: The change tracks of some top K features for identifying reliable features

As can be seen, the slope of the curve is sharply increased and gradually trends to gentle incline after K is greater than 30, most of these points' stable position almost reaches the cap of iterations (220), which cannot be distinguished whether is stable or not as insufficient data and shortage of stable iterations proved. Hence, these points are grouped into the uncertain state. When K is not greater than 30,

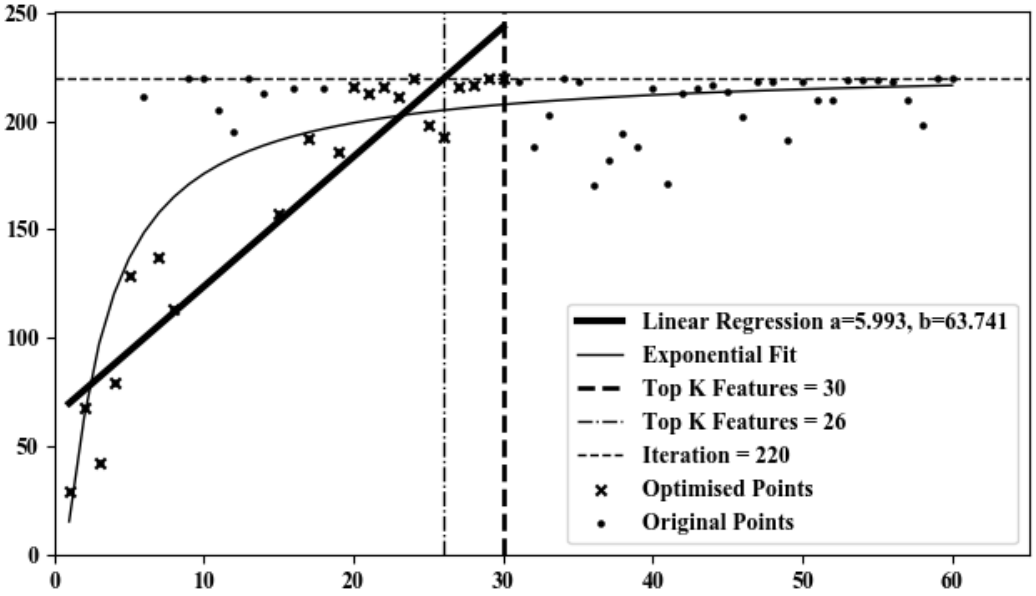


Figure 5.4: The relationship of top K reliable features and iterations

these points belong to the three states as discussed in the previous. For instance, by tracking changes of some points roughly ranging from 9 to 16, a common phenomenon can be identified, most of their features are stable, only a small number of features are vulnerable and exchanged with neighbours, especially the 9^{th} and 10^{th} features are transposed frequently until the end. Hence, these points are grouped to the competitive state. Rest of points may belong to the steady state or uncertain state. For instance, when K is 15, these features are unchanged after the 157^{th} iteration and 63 (220 - 157) iterations prove these top 15 features are stable and reliable, while, when K is in the range from 17 to 30, by analysing their tracks, there is not much evidence to show they are in competitive state and prove they are in the stable state as inadequate stable iterations. In addition, their states cannot be analysed by equation (5.5). These two issues motivate to further investigate insight of their relationships to predict whether they are reliable or not. If only analysing the points in the stable state and uncertain state, a trend can be identified. In order to simplify this trend, a linear function is applied hereby to demonstrate their relationships in equation (5.6), in which $\mathrm{SPL}_{(k)}$ means the sta-

ble position in the linear situation. By using these two functions, it is possible to analyse and calculate values of top K reliable features and the number of iterations. For instance, if SPL is 220, by putting this value into equation (5.6), the K can be calculated to 26.074 presented in equation (5.7), which indicates that top 26 features is maximum reliable features which can be extracted from these 220 papers.

$$SP_K = 230.795 \times e^{\frac{2.429}{K}} + 5.312 \tag{5.5}$$

$$SPL_K = 5.993 \times K + 63.741 \tag{5.6}$$

$$\Rightarrow K = \frac{(SPL_K - 63.741)}{5.993} \Rightarrow K = 26.074 \tag{5.7}$$

5.2.3 Feature Taxonomy for Data Warehouse Architectures

After these 26 reliable features are identified and retrieved, the process flow transfers into Phase III for the taxonomy generation. Firstly, a model is created and upgraded by executing Word2Vec with the collected papers. This model maintains vectors for a vast number of terms and similarities among these terms. Then, top T nearest terms are identified by inputting these reliable features into this model. In this research, T is set to 10, which means 10 most similar terms are retrieved for a certain reliable feature. As a consequence, 260 terms are identified in total. In order to guarantee the quality of these similar terms, they are examined by the frequency model generated by TextRank in step 4, in which if a term cannot be found, then this term is discarded as it not important as a potential features. After this process, 182 terms are retained for further investigation. All the inductive steps from step 1 to step 14 in figure 5.1 are completed, in which the initial taxonomy with reliable features and relevant terms are generated. In this research, the deductive processes are conducted by 3 experts who rate each reliable feature

and its relevant terms whether they are important or not. Then a feature or term is discarded if its important rate is less than 2. Finally, 22 reliable features and 111 relevant terms are selected as sub-features based on their feedback, which are illustrated in figure 5.5. As can be seen from this hierarchical figure, the first node is the root of DWHA features. The second level is comprised of 22 reliable features. The third level includes 111 sub-features. For each reliable feature, its sub-features are able to explain or interpret what the meanings or concrete aspects of this feature.

5.3 Refining the Feature Taxonomy

By observing this taxonomy, there are some relationships among these features and relevant terms. In order to better organise them and enable this taxonomy easy to be used, 3 experts are asked to group these 22 reliable features based on their understanding of these features and relevant terms. The steps of this group experiment is presented on the right side of figure 5.6 and designed as follows: (1) Retrieve a feature from the feature list which initially includes all 22 reliable features; (2) Look for similar features for this feature from the feature list; (3) Delete this feature and similar feature(s) from the list and joins them into a group; (4) Check whether there is a feature left in the list, if so, then go back to step 1, otherwise go to next step; (5) Output the grounded features made by a expert. After this, three types of groups are proposed by 3 experts, which are further manipulated by setting the relationship/weight of two features to 1 if they are in a group. Finally, the summarised groups are generated by linking features whose relationships/weights not less than 2. As a consequence, these features are categorised into 4 groups, which are named as the device driver, requirement and support, data structure, process and service.

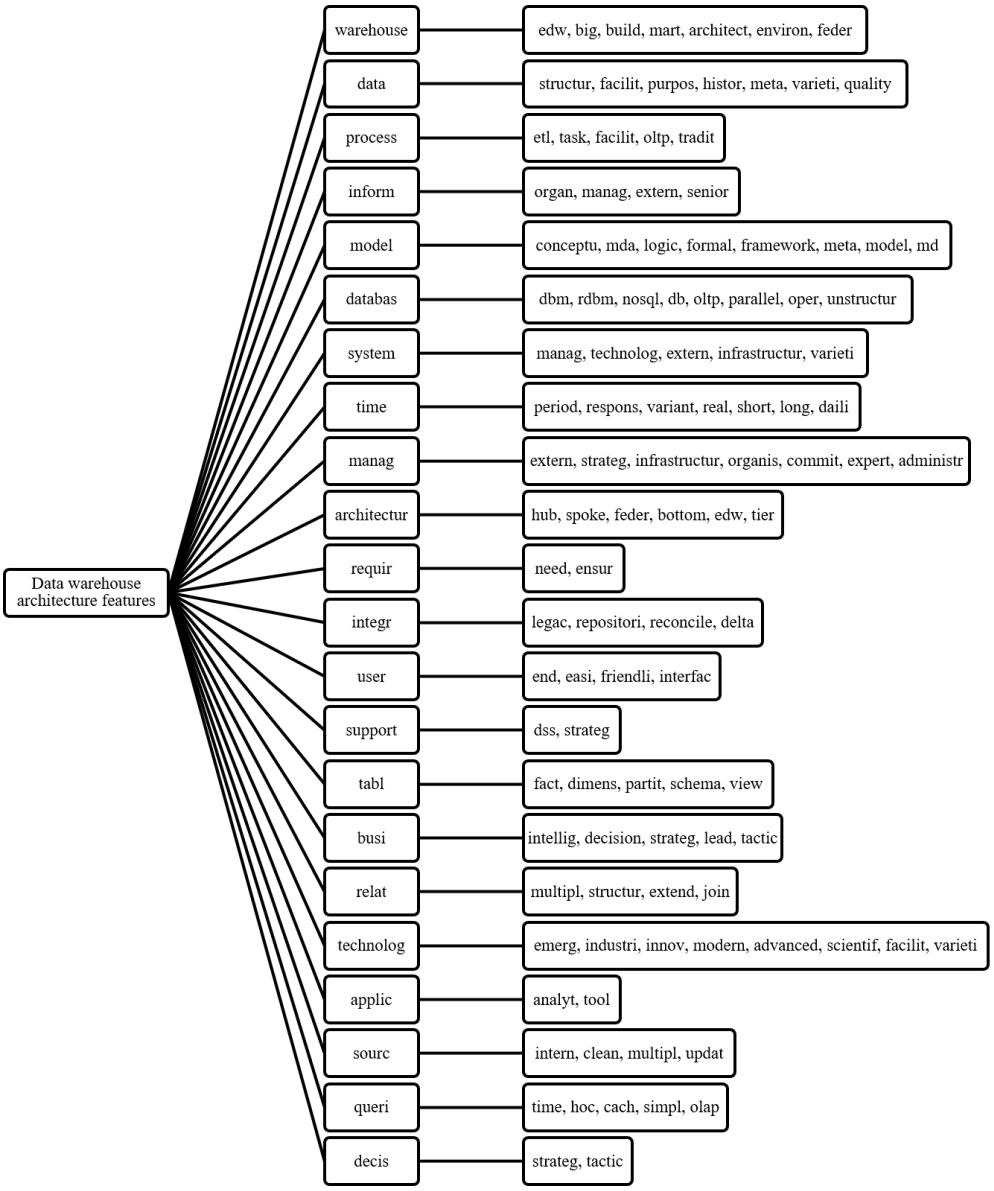


Figure 5.5: The taxonomy with reliable features and their sub-features

5.4 Feature Taxonomy Discussion

As defined in 2.3.1, features in DWHAs are important portions or a distinguishing characteristics reflecting components or attributes. By comparing the reliable features in the taxonomy and features extracted from previous research summarised in table 2.2, most of reliable features or related terms can be found in the table.

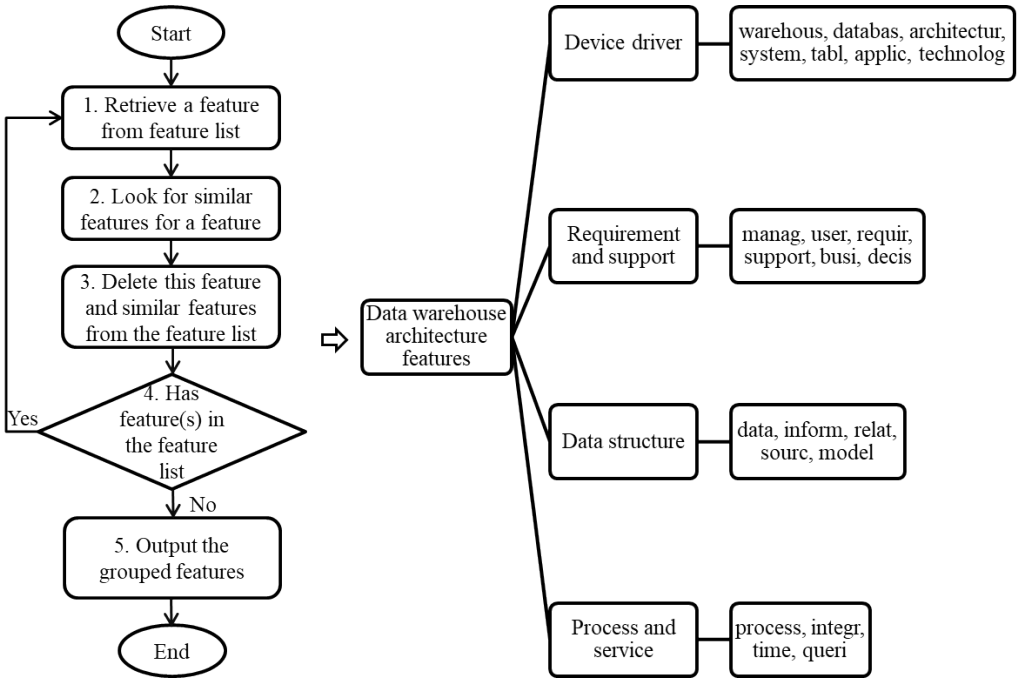


Figure 5.6: The refined feature taxonomy

When analysing the features in the table, different features are proposed in previous research studies, there is no consensus that what features are more important than others. The reliable features are derived from literature by analysing people's concerns. They are initially identified as keywords and frequently considered in literature. They show critical in this field and are further chosen based on the feature definition. Therefore, the features proposed in this research concern broader than the features proposed in previous research.

To evaluate this taxonomy, it is validated by investigating and matching the features and sub-features extracted from two cases of DWHAs built by IBM and Facebook based on the feature definition as well. The DWHAs described in [242, 265] present IBM and facebook architectures, which are chosen as the first and second cases. These two DWHAs are interpreted by experts to extract features through investigating content and the figure of the architecture. If $EF_{k,i}$ = $\{f_{k,i,a}, f_{k,i,b}, f_{k,i,c}, f_{k,i,d}, ...\}$ means the feature set identified by the i^{th} expert from the k^{th} case. Then the overlapped features proposed by experts can be presented

as $EF_k = \{\bigcap_1^n EF_{k,i}\}$. Therefore, two sets of important features are identified from experts listed below: the overlapped features for the first DWHA can be presented as EF_1 ={operational data, external data, ETL, edw, mart, other information storage, interface, decision support, business intelligence, administration, metadata, management}; the overlapped features for the second DWHA can be presented as EF_2 ={oltp, big data warehouse, federated, rdbm, ad hoc big data warehouse}. In order to test how many features extracted from these two architectures are covered by this taxonomy, the recall is measured hereby to demonstrate its validation. By counting the number of similar features or terms between EF_1 and EF_2 with the taxonomy, recalls for the first and second architectures are 91.67% and 80% respectively.

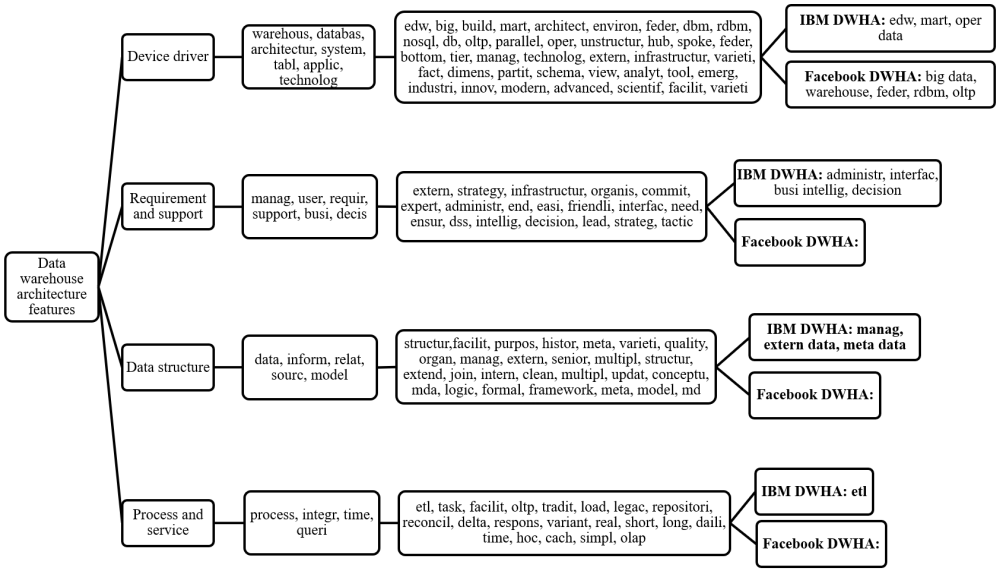


Figure 5.7: The measured results based on the feature taxonomy

Therefore, according to recalls of features for these DWHAs, the taxonomy covers most of common features and some features are not included especially ad hoc features. To be more specific, based on the description of the first DWHA, the feature named “other information storage” is a component extracting data from data marts to meet the requirements of individual users or a specific application, which may be a analytic tool or application for ad hoc queries. But without much

detailed description, this feature is not counted into the recalled features. In the second DWHA, the feature of "ad hoc big data warehouse" are not counted into the recalled features with strict rules, as the "ad hoc" feature in the taxonomy is more relevant to query or application rather than a DWH, even this customised DWH directly provides ad hoc query services. In addition, relevant terms in this taxonomy are still unclear or abstract, which should be further explained. The Word2Vec model created in step 9 of figure 5.1 can be used as a DWHA feature related digital dictionary to interpret a feature or term by searching its relevant terms. Hence, it is helpful for individuals to learn or better understand this taxonomy with this dictionary. As this taxonomy is generated by a set of steps and most of them are automatically executed, it enables people who do not have rich experience in this domain to build a initial taxonomy based on these steps. The reliable features and taxonomy are easily and swiftly updated with new papers involved, as it can be automatically conducted by updating the frequency model and the Word2Vec model to identify new features and generate the new taxonomy.

5.5 Conclusion

In this chapter, a method is proposed to extract reliable features and output a hierarchical taxonomy. This method is designed to form a taxonomy with reliable features for the domain of DWHAs, but it can be migrated to other domains to identify reliable features and taxonomies as it has low mutual coupling with a certain topic. Nevertheless, more cases should be involved to test its generalisation. In this method, a systematic LR is conducted to collect data, TextRank and Word2Vec are applied to extract reliable features and their sub-features, respectively. Since in previous research studies, most of the features are derived by researchers' perceptions, this proposed method decreases manual intervention and significantly increases the efficiency and reliability of features and taxonomy. By executing this method with DWHAs related articles, the sub-research question 2 is answered with these reliable features and their sub-features. To the best

of knowledge, this is the first work that is to address the reliability issue of features and taxonomy for DWHAs. The taxonomy generated by this method can be used to further evaluate DWHAs. In order to validate the derived taxonomy, this study measures two DWHAs from IBM and Facebook, whereby the measurement shows the validity of this taxonomy covering most of their features.

6 Result of SRQ 3 - Method for Data Warehouse Architecture Evaluation

This chapter is related to SRQ 3 and three DSRM phases (design & development and demonstration). It proposes a method to systematically evaluate DWHAs based on the results generated in chapters 4 and 5 (e.g. DWHA classification, reliable features and taxonomy). Some features are selected and associated with measurements to build a framework, which is the core part of this evaluation method. During the development of this method, some concepts from TPC are adopted to evaluate some features.

6.1 Features in the Evaluation Method

Before designing and developing this method, it is worth to discuss features which act as the primary materials for evaluation. The DWHA feature taxonomy with 22 reliable features and a set of relevant sub-features are identified in the previous chapter. The feature for DWHA evaluation should be also discriminative. As in this research, the evaluation is conducted to measure features from the architectural perspective. By iteratively testing these features to evaluate DWHAs and updating them based on acquired feedback in workshops and meetings, some of them show more discriminative when conducting the evaluation. In addition, as discussed in 2.2.1. the DWHA is defined with four layers (source layer, ETL layer, data storage layer and data presentation layer). In order to better associate with

DWHAs, These features are selected and filled into these layers. Some features do not belong to these layers, hence they are allocated to the other.

6.1.1 Source Layer

In this layer, source and data are chosen as the main features to evaluate DWHAs. In addition, operational data normally has data quality issues, so they are measured in this layer. The following sub-sections explain the details of some options or aspects being considered when evaluating DWHAs from the source layer perspective.

6.1.1.1 Source - Data Source

A DWH is a centralised data repository to store and maintain data which may be derived from a single or multiple data sources. Kwon et al. [125] denote data can be divided into internal data and external data. Some data may be directly obtained from internal operational or archive systems, this source is named **Internal Source**. Data in this kind of the data source is stored in a physically/logically same location or device with DWHs placed. Hence, it does not need to care more about remote transmission and data leakage. There are different types of DWHs which may extract data from the internal source, e.g. virtual DWHA, independent DM, centralised DWH, hub-and-spoke DWH etc. [18, 22, 40, 58]. DWHs extract data from different data sources which belong to disparate organisations. This kind of data source is called **External Source**. As data from different data sources may be formatted miscellaneously and remote transmissions may be required to collect data from different locations, extra efforts should be made to build DWHs with external sources. External data may highly provide valuable information for decision-making or increasing business knowledge [43]. There are some DWHs obtaining data from external sources, e.g. centralised DWH, Hub-and-Spoke DWH, big DWH etc. [144, 160, 279]. As can be seen, there is no boundary or regulation whether a DWH is allowed to have internal sources,

external sources or both of them. For instance, a centralised DWH may extract data from an internal source, it may extract data from an external source or both of them depending on the requirements or achievements of a DWH project.

6.1.1.2 Data - Data Format

Structured data: This type of data is defined by the data which can be directly readable or manipulated by computing devices [25]. It is typically stored in or derived from relational databases [135]. Data in these systems are normally organised in tables with structured rows and columns to provide services of storing, managing and retrieving data. It is a type of data which can be addressed in efficient ways. In terms of benefits for DWHs, this formatted data can be readily undertaken, but it only occupies a small portion (around 15% - 20%) in all data [32, 68]. As it is supported by the database, it can be effectively and efficiently operated for handling and analysing [209]. According to [68, 135], the structured data may be allocated in or extracted from different types of sources summarised as follows: relational databases, spreadsheets (e.g. Excel), online transaction processing systems, BI systems, DWHs, etc.

Unstructured data: It stands for the data which cannot be populated into database tables with rows and columns directly. This type of data may be stored in a BLOB (binary large object) [32]. Unstructured data has no pre-defined format with data model and is not suitable for mainstream relational databases [68]. As for the domain of the DWH, this type of data could be found in BI systems and analytical applications for further analysis [32]. As this type of data lacks consistent formats, so extra efforts should be made to pre-process it to machine-readable version for further analysis. It can be presented in different patterns including web pages, images, videos, reports, surveys, emails, pdf, log, etc [32, 69]. There is another concept which is called the semi-structured data. This type of data is not stored in well-organised tables or relational database systems, but with some structured fields benefiting for querying [38]. Hence, there is no big difference between un-

structured data and semi-structured data from the rational database perspective. In order to simplify the measurement for the data type, in this research, unstructured data includes semi-structured data.

6.1.1.3 Data - Data Quality

Solving data quality issues is important for DWH construction and operation. Data quality issues usually come first when data needs to be analysed. It occupies approximately 80% of the total data engineering effort in practice [285]. A large amount of time is spent on how to pre-process unformatted or defective data. After this, the processed data can provide sufficient information, and be easily analysed, mined or utilised to make decisions. The data pre-processing might be the last inspection to handle data quality problems before data is delivered into DWHs. The data from relational database systems normally has referential integrity constraints which enforce the business rules associated with databases and prevent the entry of invalid information into tables [173]. It has pre-defined models to organise data in a structured pattern. Hence, structured data could be easier to assure its quality. However, the volume of unstructured data is approximately five times the volume of structured data [100]. Unstructured data occupies 85% or more of all data [207], so it is not sufficient for decision-makers or data analysts to only use structured data if intending to get more useful information from data. Therefore, data quality is an essential and crucial feature which should be considered in evaluation. In this research, the concepts and the data set of TPC-DI benchmark are leveraged to investigate typical data quality along with the classic data quality dimensions proposed by Wang and Strong [260]. Finally, six typical data quality problems are identified and illustrated below.

Missing Values: There are two types of missing value problems in data sources. On one hand, the data in one field appears to be null or empty. This type of missing values is defined as direct incompleteness, which means this can be directly detected by rule-based queries. On the other hand, the data could be missing be-

cause of the data operations such as data update. This type of missing values is defined as indirect incompleteness.

Conflict of Entities: In this research, the definition of an entity is a record or object stored in a table. The difference between the entity and the record is that a record may contain one entity or multiple entities. The conflicts of entities mean that there is more than one valid or active record with the same identifier in a table. The records in tables need to agree with each other and avoid conflicts between them.

Format Incompatibility: This issue appears frequently in the date format. Date format conflicts are triggered by inconsistent date styles between data sources and DWHs. For instance, in the TPC-DI data set, the field of EffectiveDate means the effective date of a certain record. The date retrieved from the data source is the string with the format YYYY-MM-DDTHH24:MI:SS which includes the date and time split by the capital T. If the DWH is built in the Oracle database system, the date format is DD-Mon-YY HH.MI.SS.000000000 AM/PM which uses a different date and time format compared with the format in the data source.

Multi-resource or Mixed Records: In raw data sources, a record may contain more than one table's entities. These entities in a record normally have referential or dependent relationships. For example, in the TPC-DI data set, the CustomerMgmt.xml file has a record which may contain two dimension tables' entities (DimCustomer and DimAccount tables). An account must belong to a certain customer and a customer could have more than one account (One-to-Many relationship). For each record, there is a field named an action type, which shows the purpose of this record. When inserting a record to create or update a new account, the value of the Action Type is New or UPDACCT respectively. This record includes two entities which contain customer and account's information. Whereas, when only updating the customer information, the value of action type is UPDCUST. In this case, the record only contains one entity.

Multi-table Files: In raw data sources, some files contain more than one ta-

ble's records. This situation may happen when records in tables are collected from one system. For instance, in the TPC-DI data set, a file may contain three tables' data: CMP, SEC and FIN. The CMP records are related to the DimCompany table; the SEC is for the DimSecurty table; the FIN feeds the Financial table. Based on types of records, the data extracted from this data source file can be divided into several branches. Each branch may have sub-branches for different purposes as they can be further split into different sub-branches (e.g. ACTV and INAC). Then there are several branches and sub-branches which need to be considered in the process of loading data. If dependencies and links exist among these tables, the sequence of loading the data into tables needs to be prioritised as some tables may depend on other tables via foreign keys. If ignoring this sequence, errors may be triggered as the foreign keys are not found.

Multi-meaning Attributes: In data sources, an attribute or a field may allow containing different types of data which could have different meanings. It could be difficult to avoid ambiguities with improper differentiation. For instance, in the TPC-DI data set, an attribute named the CoNameOrCIK may carry the company identification code (10 chars) or company name (60 chars). Some rows may use company identification codes, while some rows may use companies' name. In a table of the TPC-DI benchmark, there is an attribute called SK_CompanyID which is the primary key of the DimCompany table as well as the foreign key of the Financial table. Thus, when inserting a record into this table, it could either use the company identification code or company name to look up related dimension tables to find the primary key and then insert it into the Financial table as a foreign key. It could be confused or make mistakes with multiple formatted values in an attribute during conducting the loop-up and insertion processes.

6.1.2 ETL Layer

A DWH may need an ETL system to transfer data from one point to another. An ETL system has the ability to gather separate sources and finally delivers data in

DWHs with a presentation-ready and unified format. Even though an ELT system is invisible to end users as a black box, it could cost 70 percent of the resources in the DWH implementation and maintenance [116]. The process of the ETL can be described by extracting, transforming and loading data from one data source or multiple data sources into a data repository [194]. In a DWH, the ETL system is the bridge for the data migration from data sources to the destination. There are three phases (E, T and L) in ETL systems, the E phase is an indispensable part and normally comes first, the T and L phases are dispensable in some DWH systems. In addition, different orders of these three phases present different sequential processes. For instance, the ETL system means extracting data first, then transforming data and finally loading data. While, the ELT system operates processes with extracting data first, then loading data into a DWH, finally transforming data. Therefore, the possibility of this system could be presented in five cases $\text{Set}_{ETL} = \{\text{E, ET, EL, ETL, ELT}\}$. Each of them is explained in the following contents below.

6.1.2.1 Process - ETL Type

E: Some DWH systems only have the E phase, especially the DWHs built in one layer DWHAs. This kind of architectures stores data once and once only, which avoid the data management problems with multiple and synchronical copies of data. These systems directly extract data from operational systems and provide real-time data as DWHs for data analysis [58].

ET: This type of systems extracts and transforms data, which do not necessarily load the transformed data into other systems for data analysis. The ET system conducts these two processes in the same place as the E system does in one layer DWHAs. For instance, a DWHA manipulates and analyses operational data with limited data integration, then analyse results and save them into views [41].

EL: Extracted data may not need to be transformed, which may be retrieved from the same operational system or multiple systems with confirmed data for-

mats. This system could be found in some DWHAs with relatively pure data sets. For instance, the architecture with an independent DM does not require much data transformation. However, these DMs are not formally advocated in the industry. Nevertheless, they exist and are used as a DWH solution in organisations [11].

ETL: This type of DI systems is frequently and widely applied in DWHAs to gather data and transform data with mutually agreeable formats, then load transformed data into DWHs especially with multiple data sources. For example, in a Hub-and-Spoke DWHA, several different data sources are retrieved by the ETL system and transferred into a centralised DWH, then split into several DMs to provide data for further analysis [287]. This ETL system is a traditional approach to uniform data with consistent formats, in which three phases normally occur before the DWH systems. In other words, they independently provide ETL services without assistance from DWH systems, which is different to the ELT processes explained below.

ELT: This system is a novel structure compared with other ETL structures. The main difference between them is the order of these three phases occurred. The processes of the ELT may take place in different systems and populate data into DWH systems without conducting data transformation, which normally appear in some innovative DWHAs. For instance, referring to the DWHA proposed in [279], data is firstly extracted from data sources and loaded into a big data framework, data is further transformed in this framework, which has high-speed ability to manipulate data. After data is loaded into the framework, it may be not necessary to transform data immediately, which could be conducted when data are queried.

6.1.2.2 Process - Loading Strategy

Updating data is essential to keep data fresh and complete. DWHs normally do not need data to be real-time except some special cases, but they still need to be updated with new data sets in each loading period which could be 24 hours or sev-

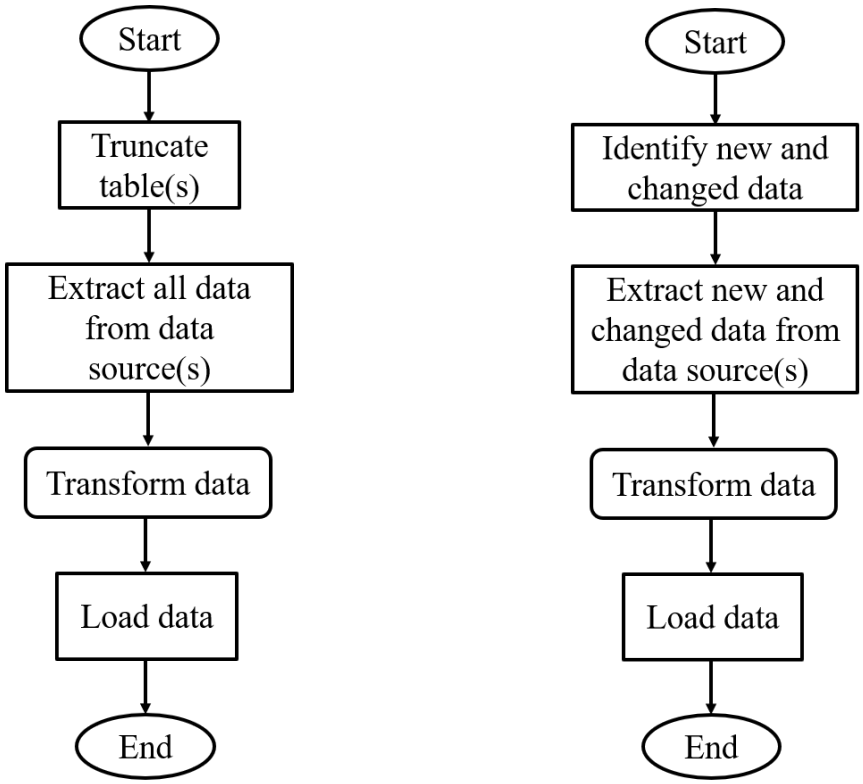
eral days depending on business requirements. There are normally two strategies for new data loadings including full load (or destructive load) and incremental load (or delta load) [42, 196, 201], which are discussed as follows.

Full Loading: It is a set of processes to fully delete the existing data in DWHs and populate all historical and new data into DWHs [254]. Hence, a lot of data without any change is still expurgated and reloaded into DWHs in each full loading. In this strategy, the ETL system does not necessarily identify which data is new or update, which has simple logic and just extracts all data from data sources. So this strategy is easy to be conducted and it is straightforward to guarantee data consistency. But it needs more time to transfer data which has already loaded from data sources to DWHs [42], especially in cases with a vast number of historical data and a fraction of new data.

Delta/Incremental Loading: The delta/incremental loading strategy has more complicated processes than the full loading strategy, which does not erase loaded data in DWHs. Through analysing data, this strategy only extracts new and updates data sets from data sources and sends to DWHs [108]. It takes less time than full loading as avoiding already loaded data, but it is more complicated to implement this strategy and it could weight the data integration system when identifying new and updating data sets [42]. Therefore, this loading is beneficial when most data is stable and a small portion of data is changed or inserted. In a DWH system, it is possible to use these two strategies for loading different tables depending on requirements and situations. The brief processes of these two loading strategies are illustrated in flow charts 6.1 below.

6.1.2.3 Process - Loading Throughput

Based on the taxonomy, loadings are crucial to populate data into DWHs. The loading processes occur in ETL layer, in which data is extracted, prepared and loaded into DWHs. Instead of loading new and changed data into DWHs, there is another loading named historical loading which is essential to establish DWH



(a) The full loading strategy **(b) The incremental loading strategy**

Figure 6.1: Two update strategies for DWH loading

systems. But this loading arises when a DWH has been implemented and is ready for obtaining historical data. This type of data is legacy information generated before, which may hold decades of relevant data in a DWH project. While the data populated during the update loading is relatively fresh compared with the data populated during the historical loading. These two loadings provide adequate and fresh information to DWHs for further analysis. Their throughput is significant as a crucial feature to measure their performance (e.g. [194, 245]). In this research, the concept of [244] on how to measure loading throughput is adopted but with more flexible patterns, which is illustrated in equation 6.1 below. $Records_i$ or $DataSize_i$ are number of records or the size of data transferred in ETL layer in a loading process. $Max(Time_i, 180)$ is the time taken in a loading. The loading throughput is designed as the average value in mutiple loadings, hence, n stands

for the number of loadings.

In TPC-DI, the throughput only measures the total number of rows of source data and it has one historical and two update loadings, while, in equation 6.1, either the total number of rows or total volume of data size can be leveraged to calculate the average throughput. It is extended to some scenarios which have more than two loading processes. In addition, the TPC-DI only allows using specific data sets generated by DIGen to measure the throughput, while this equation can be applied with customised data sets. It is possible to separately calculate the historical loading throughput and update loading throughput depending on requirements.

$$LoadingThroughput = \frac{\sum_{i=1}^{n} \frac{Records_i \vee DataSize_i}{Max(Time_i, 180)}}{n} \tag{6.1}$$

6.1.3 Data Storage Layer

This layer is the primary part referring to different concepts and mechanisms used in the storage Layer, in which DWHs are significantly vital components to manipulate, manage and provide data for the following systems or tools. Based on the taxonomy, three features are chosen to be evaluated in this layer, which include the types of DWHs, their tables and time available for querying in a period.

6.1.3.1 Warehouse - Data Warehouse Type

Different types of DWHs may have different components and characteristics, which may affect their performances and abilities. Hence, the DWH type is leveraged as one of the sub-features to evaluate DWHAs. As discussed in the chapter 4, there are 9 typical DWH systems identified from the literature by executing the proposed classification method, which include virtual DWH (**VDWH**), centralised DWH (**C**), hub-and-spoke DWH (**HAS**), data mart bus DWH (**DMB**), independent data marts DWH (**IDM**), federated DWH (**FDWH**), parallel DWH

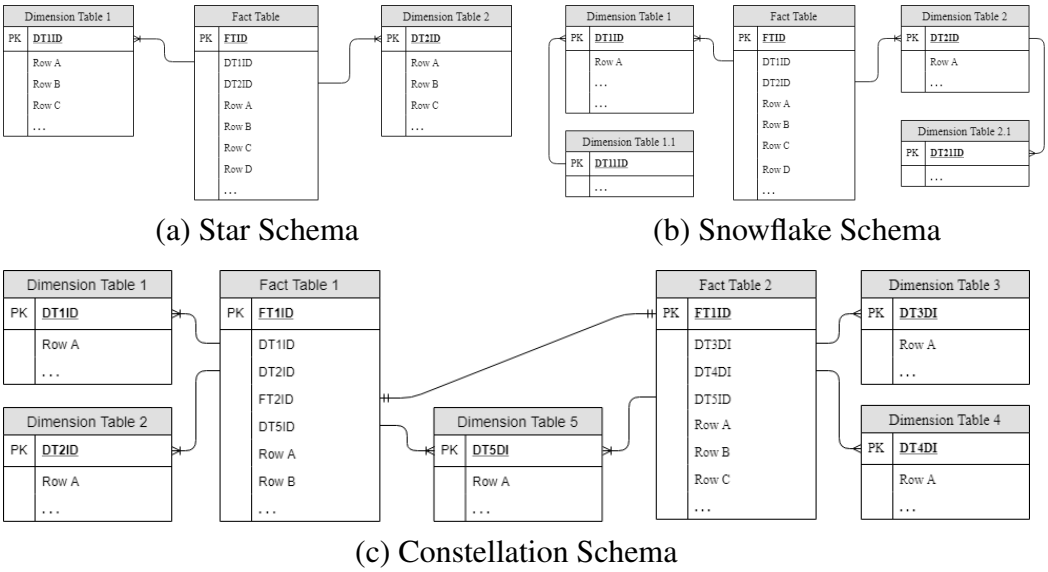
(**PDWH**), big DWH (**BDWH**) and data lake. However, in this research, data lakes are considered and linked with DWHs explained in 4.3.1, which form the DWH with data lake (**DWHDL**). These DWH systems could be used to measure a DWH along with the classification method proposed in chapter 4 to identify which type of this DWH belongs to.

6.1.3.2 Table - Schema Type

The schema is a sub-feature of a reliable feature (table), which is fundamental information to show how data is organised and stored in a DWH system. There are three schemata identified from the literature, which include **Star Schema**, **Snowflake Schema** and **Constellation Schema** [174, 248]. Most DWH systems apply the concept of the star schema to build their data model [42]. In the star schema, a fact table is surrounded and linked by a set of dimension tables and there is no link between these dimension tables. A DWH system with snowflake schema is comprised of a fact table and a set of dimension tables, but some dimension tables may have sub-dimension tables to provide details information [132]. The fact constellation schema is more complicated than the star and snowflake schema, which has several fact tables and a set of dimension tables. It maintains a group of star schemata with hierarchical links between fact tables, which enable data to be drilled down with different levels and granularity [156]. This schema has stronger abilities to model and provide different dimensional information for data analysis. But it requires higher level skills to implement and maintain this complex schema compared with other schemata. Three samples are illustrated in figure 6.2 below.

6.1.3.3 Time - Period

Time is one of reliable features in the taxonomy, which has several sub-features including loading, period, response, etc. As for the DWH layer, the period is chosen to measure how long a DWH is available to provide services for responding queries from end users in a period. This measurement is essential to present the



(a) Star Schema (b) Snowflake Schema

(c) Constellation Schema

Figure 6.2: The samples of three typical schemata used in DWHs

availability of a DWH. If a DWH needs a considerably long time to update data in a period, it implies less time available to provide services for end users. In this research study, the evaluating indicator is named **Availability Per Period**, which is obtained by calculating the average of available time in n periods to reduce deviations. The details of how to generate this indicator is presented in equation 6.2. *Period* is the time interval between two loadings. $UpdateTime_i$ is the time consumption in a loading. n is the number of loadings in order to calculate the average of the availability per period. As for a DWH, the period is the interval between two updates, which is normally fixed, so to get this result, the main work is recording how much time is taken in each update process. For instance, if a DWH updates data daily (24 hours), and the time in 4 updates is 5, 4.5, 5.5 and 5 hours, then the **Availability Per Period** can be calculated by [(24-5)/24+(24-4.5)/24+(24-5.5)/24+(24-5)/24]/4, which is 79.17%.

$$AvailabilityPerPeriod = \frac{\sum_{i=1}^{n} \frac{Period - UpdateTime_i}{Period}}{n} \tag{6.2}$$

6.1.4 Presentation Layer

The presentation layer places emphasis on further organising data to provide upper services for data analysis. In this research, it does not refer how to present or visualise results responded by DWHs as these are a vast number of tools or platforms to show data with different patterns, which is more related to DWHs on how to efficiently and effectively provide data and respond queries. In this layer, three reliable features are leveraged to evaluate DWHAs, which include business intelligence, table and query. Based on these, three sub-features (OLAP, table view and query throughput) are utilised to measure DWHAs from the presentation perspective.

6.1.4.1 Business Intelligence - OLAP

As for business intelligence, its sub-feature, namely the OLAP, is selected to do the measurement. It is applied to acquire important business-critical knowledge from data crossing enterprises, which is an essential component for BI system [2]. This component allows end users to query ad hoc analysis of multiple dimensional data, which can offer insight and meet the need for better decision making [170]. There are several typical operations including roll-up, drill-down, slice, dice and pivot when retrieving data from OLAP systems. Roll-up and drill-down analyse high levels of dimensions with coarse granularity and low levels of dimensions with fine granularity in details respectively. The slice and dice are related to obtaining data from different number of dimensions to further manipulate data for specific targets. Pivot is to rotate the multidimensional view of data to show different aspects to end users [42]. There are three main typical OLAP systems: **ROLAP**, **MOLAP** and **HOLAP**, which are explained below. There are other OLAP systems existed to analyse data, but they are less popular kinds of OLAP technologies or do not really exist any longer (e.g. DOLAP, WOLAP, Mobile OLAP, SOLAP, etc.) [170], which are not considered in this research.

ROLAP is an acronym for relational on-line analytical processing, which enables to directly retrieve data stored in relational databases by using SQL (structured query language) which is the standard database language to define and manipulate data in an RDBMS. ROLAP can reduce resources by enabling organisations to use their existing RDBMS rather than invest or purchase other tools [170]. MOLAP is the multidimensional OLAP, which provides faster services for customers especially who have ad hoc queries. It organises and maintains data based on an ad hoc logical model that can represent multidimensional data and operations directly [223]. As ROLAP is not designed for ad hoc queries and MOLAP only holds less scale and only a limited amount of data [29], HOLAP is the abbreviation of the hybrid online analytical processing and is proposed to bridge the gap between the two models, which combines the ROLAP and MOLAP into a single architecture [62]. HOLAP has the ability to convert from the cube down to the relational tables for detailed data, which shows advantages on scalability, quick data manipulation and flexibility in accessing data sources [170].

6.1.4.2 Table - View

A view acts as a virtual table [162], which does not really store data in a table. so it does not require a block of space to locate data. A view could benefit to simplify and customise data for users and provide a mechanism to protect original data by only allowing users to retrieve data through the view without permissions to directly access the base tables. It offer a compatible interface and is transparent to users for querying data even through base tables' schema has changed [153]. In DWH systems, views are important to increase the speed of responding end users queries. When a view is visited, it is still necessary to go back and obtain data from base tables, which may take a long period of time until results generated as the volume of data in DWHs may be more enormous than data in normal databases. Hence, the concept of the materialised view is adopted to avoid this look-back process, in which data is allocated with some space and end users can retrieve

data from it directly. However, materialising views acquire extra financial, spatial and operational resources. Therefore, there are different types of strategies and algorithms to find the trade-off on selecting what views should be materialised [240]. As this research focuses on architectural perspectives rather than how to choose and materialise views, so two sub-features (**View** and **Materialised View**) are chosen to measure DWHAs for this table feature.

6.1.4.3 Query - Query Throughput

The OLAP and view can reduce query response time [28]. The consumed time for responding queries is fundamental to present the capability of analysing data, which affects the satisfaction of end users. In this research, the **Query Throughput** is selected to measure the number of responding queries in a period of time referring to the average of query response time with a set of queries. The processes of how to measure this indicator are illustrated in equation 6.3, 6.4 and 6.5 which are designed based on the concept of [245]. At the beginning, time consumption in each query ($QueryResponseTime_i$) is recorded and put into equation 6.3 for calculating the average of query response time in n queries. In this step, the second is applied as the unit. Then the average of query response time is input into equation 6.5 to calculate how many queries can be responded in an hour under a certain data size. The size factor in equation 6.4 is set to several GB-based levels (1, 10, 100, etc.). For instance, if the *DataSize* of a DWH is 56.4 GB, then its size factor level is 56. If the *DataSize* of a DWH has 56.5 GB data, then its level is 57.

As for the equation 6.5, 3600 means 3600 seconds in a hour. n is the number of queries, which is the same with the n in the equation 6.3. $SizFactor$ is the result of the equation 6.4. $QueryResponseTime_i$ stands for the same as the valuable in the equation 6.3. In order to enable results greater than 1, the unit of the query throughput is queries/hour. If using queries/minute or queries/second, the results could be less than 1 as the query response time normally is several seconds even

several minutes especially with ad hoc queries. For instance, in [7, 47, 243], the query response time is various from several seconds (e.g 4.7 seconds) to thousands of seconds (e.g 7,069.7 seconds). Therefore, the hour acts as the granularity to measure this indicator. However, this granularity is flexible and could be converted to the minute or second depending on specific requirements and situations. While, when evaluating several DWHAs at the same time, the unit should be consistent. In addition, it encourages to compare DWHAs with similar data size, their values of the size factor could be fairly equivalent. Otherwise, the query throughput of DWH systems with small data sizes could be affected with small values of size factors.

$$AverageTime = \frac{\sum_{i=1}^{n} QueryResponseTime_i}{n} \tag{6.3}$$

$$SizeFactor = Around(\frac{DataSize}{1GB}) \tag{6.4}$$

$$QueryThroughput = \frac{3600 \times SizeFactor}{AverageTime} = \frac{3600 \times n \cdot SizeFactor}{\sum_{i=1}^{n} QueryResponseTime_i} \tag{6.5}$$

6.1.5 Other

Some other features and sub-features are significant to present DWHAs' performance. As these features could be in any or outside of these four layers, they are allocated into the other part. In this part, three aspects (user type, meta data, architecture complexity) based on three reliable features (user, data, architecture) are selected to evaluate DWHAs as supplementary indicators other than the sub-features in the four layers. The details of these three sub-features are explained in the following contents below.

6.1.5.1 User - User Type

As for a DWH project, investigating end users who use this system is necessary and valuable to avoid superfluous functions or components being added and implemented the DWH system, which could reduce financial and human resources. it is an important process to interpret users' requirements and expectations of a DWH system. According to the taxonomy, there are several relevant sub-features (e,g, expert and administrator) referring to different types of roles who operate or manage DWHAs. As administrators are normally essential and indispensable for DWHA management and maintenance, so this type of user is not considered and measured in this sub-feature. By reviewing and investigating previous research studies, there are different concepts to classify DWH users.

Users can be divided into four types: content viewer, data discoverer, content creator and query expert [172]. According to it, content viewers occupy the largest user population, who query DWHs or other components for ordinary requirements or presupposed requests (e.g. report). Data discoverers are more interactive than content viewers, who discover hidden information from data in DWHs. Content creators further manipulate data and intend to create visualisations, reports and dashboards. Query experts normally are less than previous two types of users, but require most technical skills especially in programming, mathematics, statistics, etc. Taylor [236] summarises five types of users (explorers, farmers, tourists, miners and operators), which uses metaphors to explain these roles for the usage of DWHs. Explorers are more related to being queried by experts, who explore unknown patterns and unsuspected information but valuable for organisations. Farmers are similar to data discoverers or content creators, who have clear targets or goals of what they expect to require from DWHs. Tourists are relation to decision makers who are interested in an enterprise-wide view for operations. Miners are a group of analysts who should address ad hoc requests with advanced skills (e.g. data mining). Operators are administrators or staff who operate DWH systems.

In order to simplify types, in this research, users are categorised into two main types (**Content User** and **Knowledge User**). The content user does not necessarily require rich technical or other upper skills to use DWH systems, who just retrieve already-prepared results or invoke pre-implemented functions stored in DWHs, while the knowledge user should have a high level of manipulating or analysing data with further operations, who may implement algorithms or use other tools (e.g. data mining tools) to provide results or functions for the content user or themselves. To be more specific, the content user is similar to the content viewer and data discoverer in [172], the tourist and farmer in [236]. The knowledge user is analogous to the content creator and query expert in [172], the explorer in [236].

6.1.5.2 Data - MetaData

Metadata is information referring to data of how it is created, accessed and used [58]. In a DWHA, data is extracted from the data source, manipulated and loaded in a DWH, finally queried by end users, which may be converted to different formats or patterns during these phases. The volume and throughput of data in a DWHA could be enormous, which should be well-organised and scheduled to avoid congestion or crash. Inmon [100] denotes that metadata is auxiliary to explain data to provide information for managers, users and the analysts, which is one of the major and critically important components as for DWH systems. Different understandings and concepts are proposed in previous research studies. To be more specific, Devlin and Cote [58] classify metadata with three types (build-time, control metadata and usage metadata) in DWH systems. The build-time metadata refers to the data which is created and used in the processes of the application, design and construction. The control metadata is leveraged to control and manage DWHs during the operation time. The usage metadata is used to support users and improve services or productivity. According to [168], metadata can be generally classified into descriptive metadata, structural metadata and administrative

metadata. Vassiliadis [255] categorises metadata with five types, which include information on contents of DWHs including locations and structures; information on processes (e.g. refreshment); information on implicit semantics of data; information on infrastructure and physical aspects of components and sources; information on security and management data.

Based on these concepts referring to metadata, it can be summarised to four types in this research including **Configuration Metadata**, **Log Metadata**, **Management Metadata** and **Query Metadata**. **Configuration Metadata** stores information referring to the design, development, maintenance, melioration, migration and discard and other information related to configured activities on DWHs for each layer and component. This metadata contains information regarding most of DWHs' phases mentioned in chapter 1 and all layers of DWHAs. **Log Metadata** tracks details of operation information, which could be used for better understanding the current situation and performance, diagnosing and solving problems. This metadata puts emphasis on operating movements in the ETL layer and DWH layer especially for data update. **Management Metadata** records activities conducted by engineers and managers who maintain and administrate DWHAs, which may cover the information referring to four layers. **Query Metadata** stores the query information from users, which could be further analysed to improve DWH services. This metadata obtains data from DWH layer and presentation layer. Therefore, these four types of metadata cover all phases and layers of DWHAs. Different types of metadata may be produced in the same layer. For instance, the DWH layer almost can generate all types of metadata. Hence, there are some intersections between them, but they focus on different aspects and hold different types of information. The coverage of metadata aligned with different phases of DWHAs is shown in table 6.1. The ✗ mark in a cell means that a certain type of metadata covers a kind of DWHA activities.

When evaluating metadata, the coverage can be measured by calculating how many types of metadata are covered in different layers. In here the phase related metadata is not counted, as if counting all of them in table 6.1, there are some

	Design and Development	Maintenance and Melioration	Migration and Discard	Source Layer	ETL Layer	DWH Layer	Presentation Layer
Configuration	✗	✗	✗	✗	✗	✗	✗
Log		✗			✗	✗	✗
Management		✗	✗	✗	✗	✗	✗
Query		✗				✗	✗

Table 6.1: The coverage of metadata in different phases and layers

overlaps between phase and layer related metadata, and a DWHA is normally in a phase in a certain period of time, which is unfair to other DWHAs in different phases with different types of metadata. For instance, The maitaining and meliration phase has four types of metadata, while rest of them only have one or two types of metadata. However, DWHAs could have multiple layers which avoid the unbalanced types of metadata. The metadata coverage only counts the layer oriented metadata with green background in table 6.1 and can be calculated as described below. If a DWHA has two types of metadata (configuration and log metadata). Configuration metadata records all phases and layers of metadata, then the Configuration metadata covers 4 aspects. Log metadata has 3 aspects of information. Hence, the metadata of this DWHA covers 7 aspects. Based on table 6.1, the total types are 13. Therefore, the metadata coverage of this DWHA can be calculated with 7 divided by 13, which is around 53.85%. The details of these processes can be summarised and demonstrated in equation 6.6, in which the $TypesOfMetadataintheLayer_i$ is the number of metadata types in a layer and 13 is the total types of metadata in four layers.

$$MetadataCoverage = \frac{\sum_{i=1}^{n} TypesOfMetadataintheLayer_i}{13} \ (n = 4) \qquad (6.6)$$

6.1.5.3 Architecture - Architecture Complexity

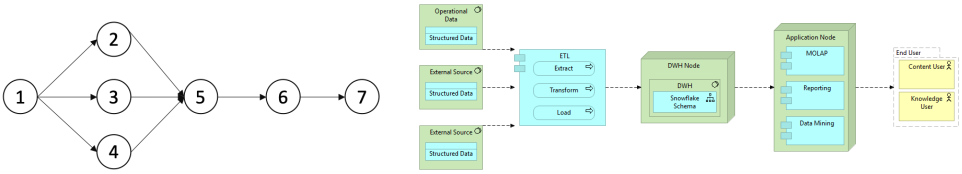
In this part, the **Architecture Complexity** is investigated and measured, which provides valuable information for whom wants to design, develop and maintain DWHAs. The theory derived from the cyclomatic complexity is leveraged and revised to fit the situation and requirements of this research. The cyclomatic complexity is widely accepted in the domain of computer science to measure complexity of software such as [14, 138]. According to [215], the mechanism of how to calculate the cyclomatic complexity is demonstrated in equation 6.7 below. In this equation, the E and N mean the number of edges and nodes, respectively. The original equation is modified to adapt calculating the complexity of DWHAs, which is presented in equation 6.8. In this modified equation, the E and N are the same to the original one, which mean the number of edges and nodes, respectively. The S stands for the number of sources which provide data for a DWH system.

$$V(G) = E - N + 2 \qquad (6.7)$$

$$AchitectureComplexity = E - N + S + 1 \qquad (6.8)$$

It is not complicated to observe and count the number of edges, nodes and sources from modelled DWHAs rather than just identify them from text. In order to align with the cyclomatic complexity and output results greater than 1, when modelling DWHAs, it is necessary to describe them from data sources to end users. If a DWHA has several source data, they should be separately presented for easily counting numbers of sources. If a DWHA has several types of end users, they should be merged and counted one. Based on the graph theory, graphs

can be classified into undirected and directed graphs [264]. As the data flow has direction and the ArchiMate is chosen to model DWHAs in this research, two types of relationships presented by dotted or broken arrow are counted as edges in a modelled DWHA, as these two types of arrows describe data flow from a node to another. Two examples are given in figure 6.3 below to calculate the cyclomatic complexity and architecture complexity. The first figure is a generalised graph, the numbers of its edges and nodes are 8 and 7 respectively. Based on equation 6.7, its cyclomatic complexity is 3. The second figure is a DWHA modelled by ArchiMate, it has 6 edges, 7 nodes and 3 sources. Based on equation 6.8, its architecture complexity is 3. As can be seen, they have the similar structures.



(a) A example of the graph (b) A example of a modelled DWHA

Figure 6.3: Example for calculating architecture complexity

6.2 Framework for the Evaluation Method

As discussed above, some reliable features and sub-features are chosen and they are further allocated into a framework in which features and sub-features are split into 5 groups including four layers and other. The detailed structure of the framework is designed in figure 6.4 below. As can be seen from this framework, in the source layer, data source, the data format and data quality are measured; in the ETL layer, different types of ETL systems, data update strategies and loading throughput are measured; in the DWH layer, DWHA types, table schema and availability per period are assessed; in the presentation layer, different types of OLAP systems, table views and query throughput are benchmarked; in the other group, different types of users, metadata and architecture complexity are evalu-

ated. There are some options or equations to measure each sub-feature, and an additional option Unknown/Null is allocated to some sub-features in case these sub-features could not be measured or their information is not given. If a sub-feature has several options or fixed scopes, these options are divided or normalised into 4 levels except for some sub-features (e.g loading throughput and query throughput) which do not have fixed scopes.

The Framework

	Feature	Sub Feature	Feature Measurement									
Source Layer	Source	Data Source	Unknown	Internal Source	External Source	Both						
	Data	Data Format	Unknown	Structured Data	Unstructured Data	Both						
		Data Quality	$DataQuality = Round(\frac{Count(D_i) - Subsistent(D_i)}{Count(D_i)} \times 3)$ $D_i \in$ (Missing Values, Conflict of Entities, Format Incompatibility, Multi-Resource, Multi-Table Files, Multi-Meaning Attributes)									
ETL Layer	Process	ETL Type	Unknown/None	E	EL	ET	ELT	ETL				
		Loading Strategy	Unknown/None	Full	Delta	Both						
		Loading Throughput	$LoadingThroughput = \frac{\sum_{i=1}^{n} \frac{Records_i \; V \; DataSize_i}{Max(Time_i, 180)}}{n}$									
Data Storage Layer	Warehouse	Data Warehouse Type	Unknown /Other	V	IDM	C	DMB	HAS	FDWH	PDWH	BDWH	DWHDL
	Table	Schema Type	Unknown/Other	Star	Snowflake	Constellation						
	Time	Time Period Rate	$AvailabilityPerPeriod = \frac{\sum_{i=1}^{n} \frac{Period - UpdateTime_i}{Period}}{n} \times 3$									
Presentation Layer	Business Intelligence	OLAP Type	Unknown/None	ROLAP	MOLAP	HOLAP						
	Table	View	Unknown/None	View	Materialised View	Both						
	Query	Query Throughput	$QueryThroughput@SizeFactor = \frac{3600 \times n \cdot SizeFactor}{\sum_{i=1}^{n} QueryResponseTime_i}$									
Other	User	User Type	Unknown/Other	Content User	Knowledge User	Both						
	Data	Meta Data Type	$Round(\frac{MetadataCoverage = \sum_{i=1}^{n} TypesofMetadataintheLayer_i}{13} \times 3)$ $(n = 4)$	0	1	2	3					
	Architecture	Architecture Complexity	ArchitectureComplexity = E – N + S + 1									

V: Virtual, IDM: Independent Data Marts, C: Centralised, DMB: Data Mart Bus, HAS: Hub and Spoke, FDW: Federated DWH, PDW: Parallel DWH, BDWH: Big DHW, DWHDL: DWH with Data Lake

Figure 6.4: The DWHA feature evaluation framework[1]

The sub-feature and their measurements in this framework are further divided. Some sub-features with green colour show macroscopic/outer characteristics or can be identified directly by observations, which hereby are named the **macro-**

[1] Text with the purple colour: the reliable & critical features from the taxonomy.
Text with the green colour: the insensitive sub-features derived from the taxonomy.
Text with the orange colour: the sensitive sub-features derived from the taxonomy.
Each feature measurement has four levels with the colours from the light red to brownish red.

scopic features and **macroscopic measurements**. While some sub-features with orange colour and their measurements show microscopic/inner characteristics or should be investigated and calculated, which could not be directly identified by just observations. They are named the **microscopic features** and **microscopic measurements**. The macroscopic features are architectural or physical components which are more related to structural indicators to describe what constitute the main portions of a DWHA. Hence, their macroscopic measurements more focus on components which are implemented before operations and not changed frequently. In other words, these macroscopic features are relatively stable and **insensitive**. However, the microscopic features are more related to data manipulations and transmissions, whose performances are frequently changeable and **sensitive**. In this research, these two sets of names are applied in different circumstances and their relationships are summarised below: **macroscopic features** are equal to **insensitive features**; **microscopic features** are equal to **sensitive features**.

6.3 Evaluation Method Design & Development and Demonstration

All materials for evaluation are identified and displayed in the framework. This part places an emphasis on how to systematically evaluate DWHAs. In order to achieve this, a method is designed with three phases including DWHA modelling, feature measurement, DWHA evaluation. Each phase is concretely explained as follows.

6.3.1 Phase I: Data Warehouse Architecture Modelling

This phase is for diagramming DWHAs and describing macroscopical/insensitive features. In this research, ArchiMate is chosen and leveraged to model DWHAs, which could benefit for feature extraction and evaluation. For example, when describing the sub-feature of the data source, it is better to manifest whether a source

is internal source or external source. Some reliable features and their relevant sub-features can be easily identified if they are directly modelled by ArchiMate with original symbols, while, some sub-features should be further investigated, analysed or calculated by measuring or checking related indicators (e.g. schema, data quality, load speed, etc.), which could not be modelled by ArchiMate with original symbols. Therefore, some symbols are proposed to extend the capabilities of ArchiMate, which allow it to articulate these features in the figures of modelled DWHAs. An example derived from [154] with some extended information is given in in figure 6.5 below to demonstrate a modelled DWHA with an extended symbol. As can be seen, the schema of a DWHA can be presented, which could facilitate DWHA description and evaluation. However, this symbol could not be found in the original ArchiMate modelling tool kit.

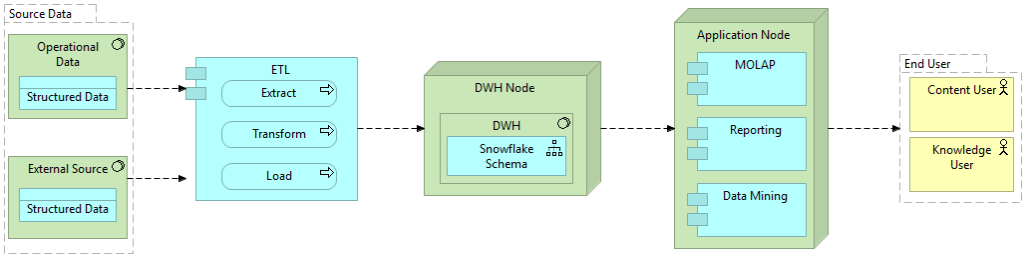


Figure 6.5: An example of a modelled DWHA with extended modelling symbols

6.3.2 Phase II: Feature Measurement

This phase measures microscopical/sensitive features and further investigates some macroscopical/insensitive features which are still unclear. In order to explain how features are measured, the example of the modelled DWHA in figure 6.5 is reused hereby to present details of these processes. As can be seen from this modelled DWHA, it is easy to identify components (e.g. ETL, DWH, end user, etc.) from this figure. There are three main types of relationships being considered hereby, which are presented as the flow, composition and coordination. For instance, there

is an arrow from the ETL to the DWH, which is for transferring data and referring to activities of loading data to the DWH. In the DWH, there is schema to organise data with the snowflake structure. The relationship between the DWH and schema is composition, in other word, the DWH contains this schema which is a sub-component of the DWH. In terms of the coordination, the ETL system has several processes to manipulate data, so this ETL includes these processes or sub-components. The relationships between the ETL and these processes are composition, while the relationships among these processes are coordination. They cooperate together to present capabilities of operating data for the ETL.

Except these features explained above, some features still need further investigation or calculation to obtain the evaluated results. Some extra information (such as loading throughput and query throughput) are supposed in order to describe the details of measuring these features. Two examples are given to explain the investigation and calculation to measure features. This system extracts data from both internal and external data sources. If the data sources have 2 types of data issues, then the score for the data quality is 3. The loading throughput, loading strategy, availability per period and query throughput are supposed and added into the modelled DWHA which is demonstrated in figure 6.6. Compared with the architecture in figure 6.5, this figure provides measured results as extra information into the original modelled DWHA, which could benefit to evaluate this DWHA in the following phase.

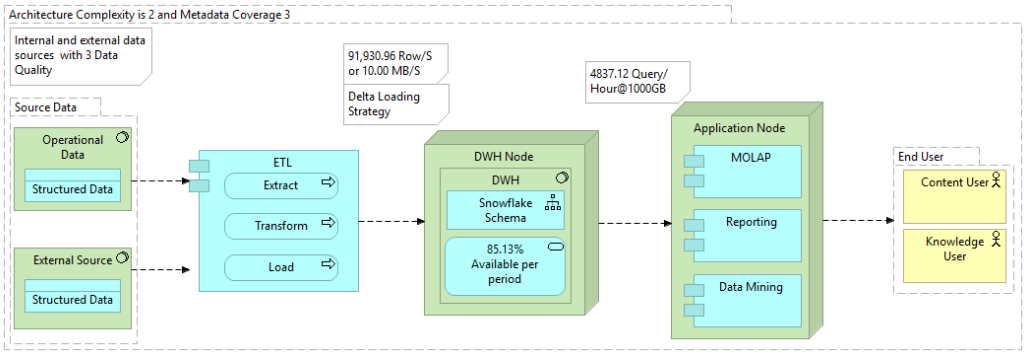


Figure 6.6: An example of a measured DWHA

6.3.3 Phase III: Data Warehouse Architecture Evaluation

After this DWHA is modelled and all features are measured, the measured results are filled into the framework. The details of the evaluated results for each sub-feature are demonstrated in figure 6.7. As can be seen from this framework, this DWHA is evaluated from five main parts. In each part, three sub-features are evaluated, in which two of them are macroscopical/insensitive features with green colour, one is a microscopical/sensitive feature with orange colour. All results of macroscopical features are graded into four levels with values from 0 to 3. The higher level means more complicated or powerful, but may need more resources. Some microscopical features having fixed ranges (such as data quality and availability per period) are normalised to the range from 0 to 3. Some other microscopical features (such as loading throughput, query throughput and architecture complexity) do not have fixed ranges, so they are not normalised. However, if they have a set of results, they could still be normalised to the range from 0 to 3. The measured results added in the framework could be a profile to fully describe this DWHA. In order to further graph the evaluated results, this DWHA can be described in to diagram in figure 6.8 below.

6.4 Conclusion

This chapter demonstrates different reliable features, their sub-features and options, and proposes measurements to evaluate these sub-features. They are further organised into a framework in which they are divided into five parts (source layer, ETL layer, DWH layer, presentation layer and other) based on their locations. In addition, These sub-features are classified into two groups (macroscopic and microscopic) based on their characteristics. Macroscopic sub-features or measurements are more related to architectural or physical components. Microscopic sub-features or measurements represent performances on data throughput, manipulations, queries and transmissions. Then a method is designed with three phases

	Feature	Sub Feature	Feature Measurement									
Source Layer	Source	Data Source	Unknown	Internal Source	External Source	Both						
	Data	Data Format	Unknown	Structured Data	Unstructured Data	Both						
		Data Quality	$DataQuality = Round(\frac{Count(D_i) - Subsistent(D_i)}{Count(D_i)} \times 3)$ $D_i \in$ (Missing Values, Conflict of Entities, Format Incompatibility, Multi-Resource, Multi-Table Files, Multi-Meaning Attributes) 3									
ETL Layer	Process	ETL Type	Unknown/None	E	EL	ET	ELT	ETL				
		Loading Strategy	Unknown/None	Full	Delta	Both						
		Loading Throughput	$LoadingThroughput = \frac{\sum_{i=1}^{n} \frac{Records_i \vee DataSize_i}{Max(Time_i, 180)}}{n}$ 91,930.96 Row/S or 10.00 MB/S									
Data Storage Layer	Warehouse	Data Warehouse Type	Unknown /Other	V	IDM	C	DMB	HAS	FDWH	PDWH	BDWH	DWHDL
	Table	Schema Type	Unknown/Other	Star	Snowflake	Constellation						
	Time	Time Period Rate	$AvailabilityPerPeriod = \frac{\sum_{i=1}^{n} \frac{Period - UpdateTime_i}{Period}}{n} \times 3$ 2.554									
Presentation Layer	Business Intelligence	OLAP Type	Unknown/None	ROLAP	MOLAP	HOLAP						
	Table	View	Unknown/None	View	Materialised View	Both						
	Query	Query Throughput	$QueryThroughput@SizeFactor = \frac{3600 \times n \cdot SizeFactor}{\sum_{i=1}^{n} QueryResponseTime_i}$ 4837.12 Query/Hour@1000GB									
Other	User	User Type	Unknown/Other	Content User	Knowledge User	Both						
	Data	Meta Data Type	$MetadataCoverage = Round(\frac{\sum_{i=1}^{n} TypesofMetadataintheLayer_i}{13} \times 3)$ $(n = 4)$			0	1	2	3			
	Architecture	Architecture Complexity	C = E – N + S + 1 2									

V: Virtual, IDM: Independent Data Marts, C: Centralised, DMB: Data Mart Bus, HAS: Hub and Spoke, FDW: Federated DWH, PDW: Parallel DWH, BDWH: Big DHW, DWHDL: DWH with Data Lake

Figure 6.7: An example of the evaluated DWHA

Figure 6.8: The evaluated macroscopical features of an example

to evaluate DWHA, in which DWHAs should be modelled based on the framework, features are extracted from modelled DWHAs, finally these features are input into the framework and demonstrated in a radar chart. In order to fully describe DWHAs and facilitate evaluations, some symbols and extra information are created to extend the modelling language ArchiMate, which allow the modelled DWHAs easily to be understood and evaluated.

7 Evaluation

In the previous chapters, the three SRQs are answered with the three methods which idenfies 9 typical DWHAs, a DWHAs feature-based taxonomy and a framework for DWHA evaluation, respectively. This research aims to develop a methodology to explicitly and efficiently evaluate modelled DWHAs through measuring features. The three methods and their outputs contribute to generate this methodology and from both academic and industrial perspectives.

This chapter is related to the evaluation phase in DSRM. After the DWHA feature-based modelling and evaluation methodology has been designed and implemented in previous chapters, it is essential to evaluate its understandability and validity. The third method of this methodology created in chapter 6 is the main part to evaluate DWHAs. Hence, it is evaluated by several cases collected from a public data set, this research and an organisation. Furthermore, empirical research is conducted to validate whether the objects of the present research are achieved or not.

7.1 Evaluating Data Warehouse Architectures from a public data set

TPC-DI and TPC-DS publish some results in relation to the tested results of DWHAs. In these results, DWH systems are reported along with detailed information of system configuration information, data set size, throughput, time consumption for queries and loads, system prices, etc. Three reports are provided in the official website of the TPC-DS benchmark, which assess three DWH sys-

tems built on Cisco UCS Integrated Infrastructure for Big Data, Alibaba Cloud E-MapReduce, Alibaba Cloud AnalyticDB and published on 5th Mar, 2018, 19th March, 2019 and 26th April, 2019, respectively [12, 47, 243]. In this section, these three DWH systems will be measured via the evaluation method based on these reports and detailed information provided in the TPC-DI website. These three DWH systems use the same data sets produced by a data generation engine and build the same schemata to organise these data sets. Hence, it benefits to discuss and compare them under the same data conditions but different architectures. The sub-sections below will fully describe how these DWHAs are evaluated in details via the evaluation method.

7.1.1 Evaluating the Data Warehouse Architecture in Case 1

Technical Environment: The first case of the DWHA is built on the Cisco UCS Integrated Infrastructure. According to the report [47], the data for this system is generated by the DBGen engine and stored in flat files with the volume of 10,000GB. These files are then loaded into HDFS which is the acronym of the Hadoop Distributed File System and it supplies high throughput access to operate data [279]. This infrastructure is established on the operating system of the Red Hat Enterprise Linux Server Release 6.7 with Transwarp Data Hub V5.1 and 17 Cisco UCS C240 M4 Servers to manipulate, store and analyse data. After these files are uploaded into the big data platform, tables of this DWH for managing data are created on Hive which is a DWH tool to promote the performance of querying and managing large data sets in distributed storage [91]. The following processes are counted into the loading time consumption. Firstly, data starts to load into these tables, audit scripts are run and statistics are conducted for loading checks, and the loading environments are cleansed. After data is populated into the DWH, the loading stage is finished and data is available for queries and further analysis. In this architecture, the staging area is not used and data is transformed and prepared in the big data platform directly. The evaluation method is executed

and the details of the processes are demonstrated in the following steps.

7.1.1.1 Modelling the Architecture in Case 1

According to the description of the benchmark report and the TPC-DS specification, this DWHA can be presented in figure 7.1. Initially, data is derived from five internal sources including information systems of Store, WEB, Catalog, Inventory and Promotions. In this scenario, data is generated by DBGen and stored in flat files, then put into HDFS. Hence, it omits some parts of ETL processes and data flows from data sources to the DWH. In order to fully display and understand the whole processes, the five data sources and the ETL system are demonstrated in the modelled DWHA. The flat files act as the extracted data prepared by a ETL system, but the data in these flat files is transformed and cleansed in the Hadoop platform. Thus, the ETL system only operates two actions (Extract and Load). After data is uploaded into the platform, it is organised and maintained in multiple snowflake schemata in Hive. The DWH system in this case provides views and MOLAP to speed up the query efficiency. It offers various services for different types of end users for reports, ad hoc queries, data mining, etc. Hence, it could fulfil requirements both from content and knowledge users.

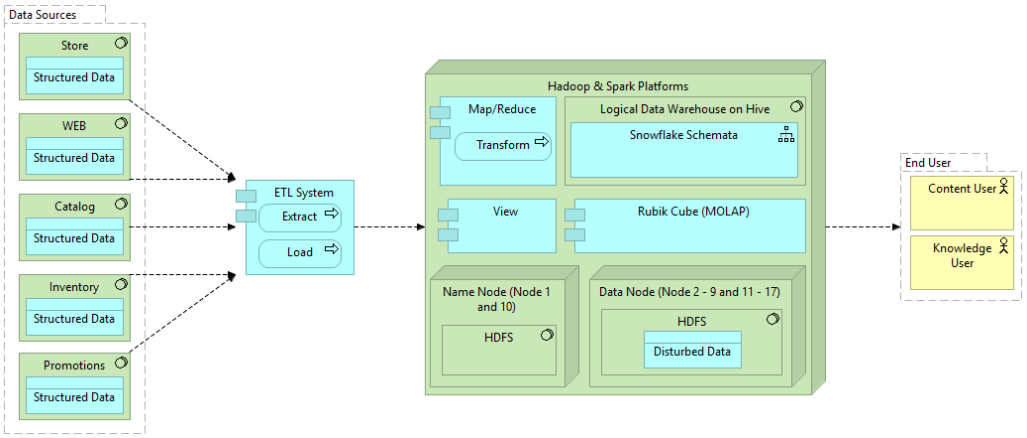


Figure 7.1: The modelled DWHA in the case 1

7.1.1.2 Measuring the Features in Case 1

Most of the macroscopic features are demonstrated in figure 7.1. The main purpose of this phase is to identify and measure microscopic features and the rest of macroscopic features in different layers. According to the report [47] and the TPC-DS specification, the detailed information can be obtained as follows. In the source layer, through examining these data sources, they have 2 types of data quality issues including missing values and format incompatibility. Based on the results in chapter 6, there are totally 6 types of data quality issues involved in the present research, so its data quality score can be calculated by the equation of the "Data Quality" in the framework, which is 2. In the ETL layer, the ELT mechanism is leveraged to extract raw data, load into the Hadoop platform and transform data in this platform. Only new and changed records are updated into the DWH via the delta strategy. During the data update, 31075.79836 MBs data or 285,715,667 rows are transferred from data sources to the Hadoop platform, and it takes 1,502.9 and 1,471.0 seconds in two tests, so the load speed (192175.48 Row/S or 20.90 MB/S) can be calculated by the equation of the "Loading Throughput" in the framework. In the DWH layer, as this DWH is built in Hive and based on the big data platform, it belongs to the big DWH with snowflake schemata. This DWH is updated daily and the average time consumption of data updating is 1486.95 seconds, so the available rate in a period can be calculated by the equation of the "Availability Per Period" in the framework, which is 2.948.

In the Presentation Layer, it maintains a MOLAP and creates views for efficiently responding to queries. In the benchmark, 99 queries are conducted and take 67306.635 seconds to analyse and response them. Hence, the query throughput is 52951.69 Query/Hour@SizeFactor presented in the equation of the "Query Throughput@SizeFactor" in the framework. In terms of others, this DWH system serves for both content users and knowledge users. It does not provide detailed information in relation to the metadata which only includes 4 types of metadata. Hence, the metadata coverage is 1 based on the equation in the framework pro-

posed in chapter 6. According to figure 7.1, the number of edges, nodes, and sources are 7, 8 and 5, respectively. Hence, the data flow complexity is 5 based on the equation of the "Architecture Complexity" in the framework. After these processes, most of features are extracted and measured based on the modelled DWHA and related documents. The measured results are added into its modelled pattern which is presented in figure 7.2.

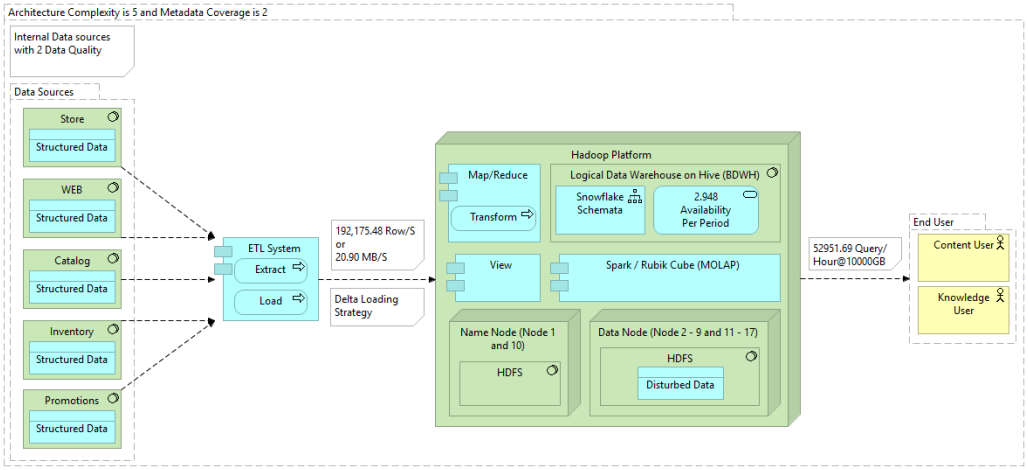


Figure 7.2: The modelled DWHA of the case 1 with measured results

7.1.1.3 Evaluated Result for Case 1

In this part, all measured results are organised together to evaluate and demonstrate this DWHA. The evaluation framework proposed in chapter 6 is applied hereby to achieve this task. After putting all measured results into this framework, the output is presented in figure 7.3. As can be observed from this figure, there are totally 10 reliable features in this framework and 15 relevant sub-features are measured to evaluate and describe this DWHA from macroscopic and microscopic perspectives. The evaluated results are extracted and illustrated in the list below. The evaluated results are separate in different layers. For instance, the evaluated results in the source layer are (1, 1, [2]), the values in round brackets are evaluated results for macroscopic features such as the value 1, while the

value in square brackets inside the round brackets is the result for a microscopic feature. Furthermore, the evaluated results of macroscopic features are further demonstrated in figure 7.4 including 10 normalised values scoping from 0 to 3.

Evaluated Results in the Source Layer: (1, 1, [2])

Evaluated Results in the ETL Layer: (3, 2, [192175.48, 20.90])

Evaluated Results in the DWH Layer: (3, 2, [2.948])

Evaluated Results in the Presentation Layer: (2, 1,[52951.69])

Evaluated Results in the other (3, 1, [5])

Case 1 - Cisco UCS Integrated Infrastructure for Big Data

	Feature	Sub Feature	Feature Measurement									
Source Layer	Source	Data Source	Unknown	Internal Source	External Source	Both						
	Data	Data Format	Unknown	Structured Data	Unstructured Data	Both						
		Data Quality	$DataQuality = Round(\frac{Count(D_i) - Subsistent(D_i)}{Count(D_i)} \times 3)$ $D_i \in$ (Missing Values, Conflict of Entities, Format Incompatibility, Multi-Resource, Multi-Table Files, Multi-Meaning Attributes) 2									
ETL Layer	Process	ETL Type	Unknown/None	E	EL	ET	ELT	ETL				
		Loading Strategy	Unknown/None	Full	Delta	Both						
		Loading Throughput	$LoadingThroughput = \frac{\sum_{i=1}^{n} \frac{Records_i \, V \, DataSize_i}{Max(Time_i, 180)}}{n}$ 192175.48 Row/S or 20.90 MB/S									
Data Storage Layer	Warehouse	Data Warehouse Type	Unknown /Other	V	IDM	C	DMB	HAS	FDWH	PDWH	BDWH	DWHDL
	Table	Schema Type	Unknown/Other	Star	Snowflake	Constellation						
	Time	Time Period Rate	$AvailabilityPerPeriod = \frac{\sum_{i=1}^{n} \frac{Period - UpdateTime_i}{Period}}{n} \times 3$ 2.948									
Presentation Layer	Business Intelligence	OLAP Type	Unknown/None	ROLAP	MOLAP	HOLAP						
	Table	View	Unknown/None	View	Materialised View	Both						
	Query	Query Throughput	$QueryThroughput@SizeFactor = \frac{3600 \times n \cdot SizeFactor}{\sum_{i=1}^{n} QueryResponseTime_i}$ 52951.69 Query/Hour@10000GB									
Other	User	User Type	Unknown/Other	Content User	Knowledge User	Both						
	Data	Meta Data Type	$Round(\frac{\sum_{i=1}^{n} TypesofMetadataintheLayer_i}{13} \times 3)$ $(n = 4)$	0	1	2	3					
	Architecture	Architecture Complexity	ArchitectureComplexity = E – N + S + 1 5									

V: Virtual, IDM: Independent Data Marts, C: Centralised, DMB: Data Mart Bus, HAS: Hub and Spoke, FDW: Federated DWH, PDW: Parallel DWH, BDWH: Big DHW, DWHDL: DWH with Data Lake

Figure 7.3: The evaluated results of the case 1

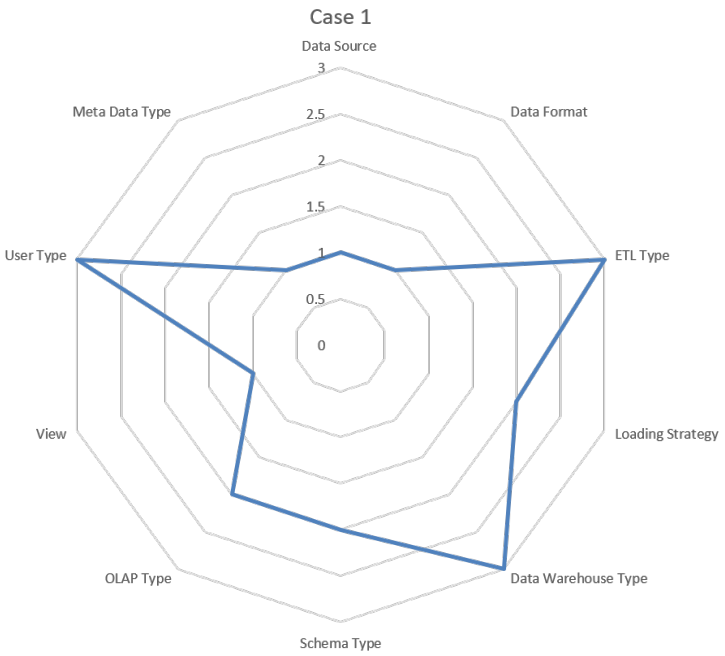


Figure 7.4: The macroscopic feature evaluated results of the case 1

7.1.2 Evaluating the Data Warehouse Architecture in Case 2

Technical Environment: This case is provided by Alibaba which is the biggest Chinese e-commerce enterprise covering multiple domains particularly in retail, internet, technology, etc. [247, 268]. According to the benchmark report [243], this DWH system is established on Alibaba Cloud E-MapReduce 3.16.1 which is a cloud elastic compute service server and is operated by CentOS Linux Release 7.4. The data is generated by the DBGen containing the volume of 10,000 GB. Similar to the first case, this system is based on the Hadoop big data platform to build the DWH on Hive and use Spark to analyse data for responding to queries. However, in this case, 19 servers and an OSS (object storage service) are used to build the main part of this system, which include 1 node of ecs.sn2ne.8xlarge, 18 nodes of ecs.i1.14xlarge with 4,160 GB total memories and 64,856 GB total storages, and one OSS standard storage with the 10 TB storage capacity and 1,000,000 daily read & write API requests. In this Hadoop-based system, one node acts as

the name node for HDFS and Spark, 18 nodes act as data nodes for HDFS and executors for Spark. In order to improve its reliability, a replica of table data is stored in the OSS which is a remote web server.

7.1.2.1 Modelling the Architecture in Case 2

Based on the report and TPC-DS specification, the detailed information of the modelled DWHA is illustrated in figure 7.5 below. As described above, the data for this DWH is derived from five data sources, which is generated by DBGen with the volume of 100000 GB in flat files. As these files are originally derived from operational systems, so the five operational systems are described in this modelled DWHA. These files are further put into the HDFS, then populated into the Hive-based DWH. The difference is that this whole system utilise 18 nodes and OSS to maintain data, while the previous DWHA in case 1 uses 17 nodes without OSS. This may complicate the architectural complexity of this DWH system but improve the data reliability when some unexpected accidents occur and the HDFS is not available for querying. In addition, this DWH leverages snowflake schemata to organise and provide efficient way to retrieve and analyse data for queries from content and knowledge users. The OLAP system is not presented in the benchmark report [243], so it is not modelled in this figure.

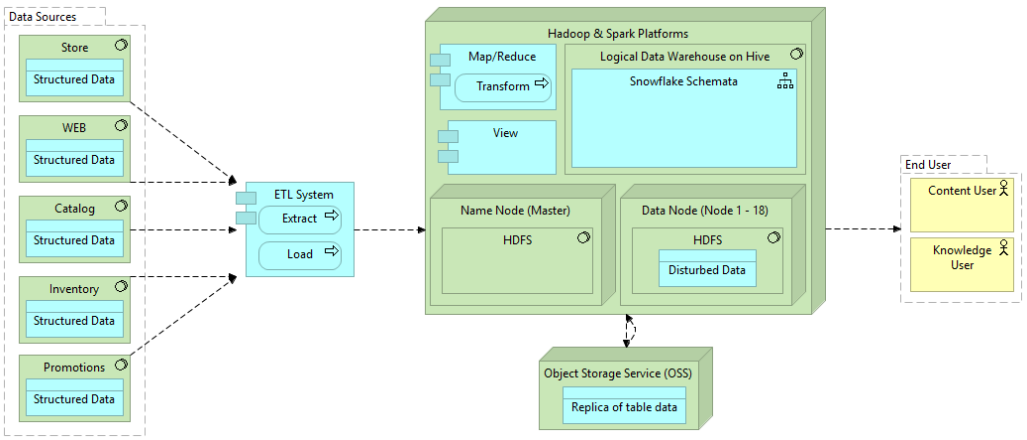


Figure 7.5: The modelled DWHA of the case 2

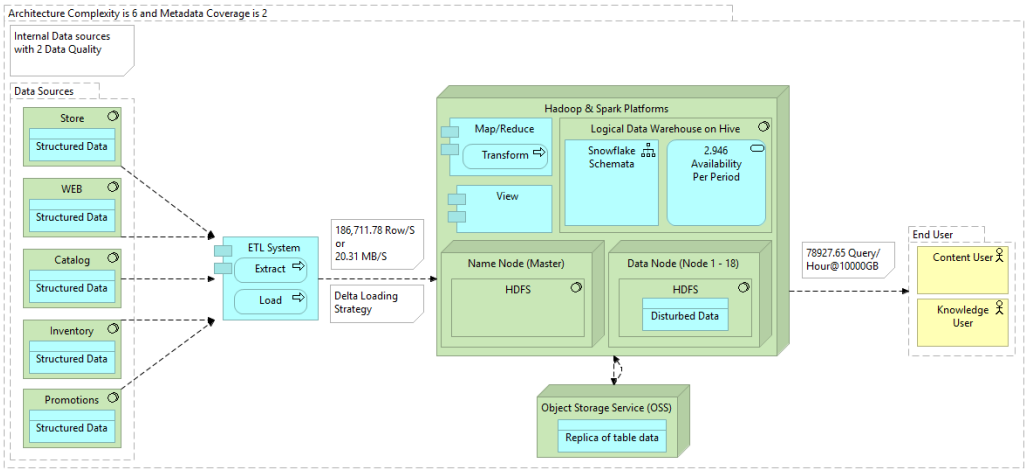


Figure 7.6: The modelled DWHA of the case 2 with measured results

7.1.2.2 Measuring the Features in Case 2

Some features can be identified and extracted from the modelled DWHA in figure 7.5. The measured results of these macroscopic features are similar to the results in the first case. For instance, the data in these five operational systems is extracted from internal sources. Both of them are based the concept of ELT to update DWHs. It utilises snowflakes and views to organise data for efficient queries. But the difference is that it does not provide much information on what OLAP is built in this DWH system. In terms of the microscopic features, this system is measured with the data generated by DBGen, so the data quality is 2. Based on the benchmark report [243], two updates are measured, which take 1,530.8 and 1,529.7 seconds. Hence, the average load throughput is 186711.78 Row/S or 20.31 MB/S. This DWH should be updated daily, so the score of the availability per period is 2.946. For the query throughput, 99 queries are executed in two tests which take 45,836.8 and 44,485.2 seconds, respectively. Thus it can be analysed to 78927.65 Query/Hour@10000GB. This system provides services for both content users and knowledge users. By analysing the documents, it covers 7 types of metadata, so the metadata coverage is 2. The architecture complexity can be

calculated to 6, in which the numbers of edges, nodes and sources are 9, 9 and 5, respectively. The details are illustrated in figure 7.6.

7.1.2.3 Evaluated Result for Case 2

After all features are measured in previous phase, they are input into the evaluation framework and the results are illustrated in figure 7.7. In this case, 14 sub-features are evaluated in total. Most of evaluated results for macroscopic features are similar to the first case except the OLAP sub-feature. There is no huge distinctions and conflicts of the evaluated results for microscopic features between case 1 and case 2. This DWHA is more complicated than the DWHA in case 1 referring to the architecture complexity, as it utilises the OSS to save a replica of data in order to improve the reliability of accessing data. More complicated DWHA means when developing and operating this DWHA, an extra stream of a data flow should be built and maintained. The evaluated results are extracted and listed below. In addition, the evaluated results of macroscopic features are presented in figure 7.8.

Evaluated Results in the Source Layer: (1, 1, [2])

Evaluated Results in the ETL Layer: (3, 2, [186711.78, 20.31])

Evaluated Results in the DWH Layer: (3, 2, [2.946])

Evaluated Results in the Presentation Layer: (0, 1, [78927.65])

Evaluated Results in the other (3, 2, [6])

Case 2 - Alibaba Cloud E-MapReduce											
	Feature	Sub Feature	Feature Measurement								
Source Layer	Source	Data Source	Unknown	Internal Source	External Source	Both					
	Data	Data Format	Unknown	Structured Data	Unstructured Data	Both					
		Data Quality	$DataQuality = Round(\frac{Count(D_i) - Subsistent(D_i)}{Count(D_i)} \times 3)$ $D_i \in$ (Missing Values, Conflict of Entities, Format Incompatibility, Multi-Resource, Multi-Table Files, Multi-Meaning Attributes) 2								
ETL Layer	Process	ETL Type	Unknown/None	E	EL	ET	EET	ETL			
		Loading Strategy	Unknown/None	Full	Delta	Both					
		Loading Throughput	$LoadingThroughput = \frac{\sum_{i=1}^{n} \frac{Records_i \ V \ DataSize_i}{Max(Time_i, 180)}}{n}$ 186711.78 Row/S or 20.31 MB/S								
Data Storage Layer	Warehouse	Data Warehouse Type	Unknown /Other	V	IDM	C	DMB	HAS	FDWH	PDWH	BDWH	DWHDL
	Table	Schema Type	Unknown/Other	Star	Snowflake	Constellation					
	Time	Time Period Rate	$AvailabilityPerPeriod = \frac{\sum_{i=1}^{n} \frac{Period - UpdateTime_i}{Period}}{n} \times 3$ 2.946								
Presentation Layer	Business Intelligence	OLAP Type	Unknown/None	ROLAP	MOLAP	HOLAP					
	Table	View	Unknown/None	View	Materialised View	Both					
	Query	Query Throughput	$QueryThroughput@SizeFactor = \frac{3600 \times n \cdot SizeFactor}{\sum_{i=1}^{n} QueryResponseTime_i}$ 78927.65 Query/Hour@10000GB								
Other	User	User Type	Unknown/Other	Content User	Knowledge User	Both					
	Data	Meta Data Type	$MetadataCoverage = Round(\frac{\sum_{i=1}^{n} TypesofMetadataintheLayer_i}{13} \times 3)$ $(n = 4)$	0	1	2	3				
	Architecture	Architecture Complexity	C = E – N + S + 1 6								
V: Virtual, IDM: Independent Data Marts, C: Centralised, DMB: Data Mart Bus, HAS: Hub and Spoke, FDW: Federated DWH, PDW: Parallel DWH, BDWH: Big DHW, DWHDL: DWH with Data Lake											

Figure 7.7: The evaluated results of the case 2

Figure 7.8: The macroscopic feature evaluated results of the case 2

7.1.3 Evaluating the Data Warehouse Architecture in Case 3

Technical Environment: According to the report [12], this DWH system is built on the Alibaba Cloud AnalyticDB platform and operated by the Alibaba Group Enterprise Linux. It employs 16 Alibaba Cloud AnalyticDB (ADB) servers/instances to read/write data and form a cluster, in which 2 instances are used as coordinators. A cloud storage platform named Pangu is applied to store and maintain all table data and its replicas. This storage is a distributed file system and provides high reliability, availability and performance [6]. These ADB servers are not necessary to maintain original table data, which just store event and transaction log, temp files and cache of table data. This system has 10,000 GB data generated by DBGen. The Pangu system act as the main container, which provides 81,920 GB space to manage this data set. These 16 ADB servers provide 39,072 GB in total to temporarily store some potentially queried data for quick responses. The AnalyticDB is based on the MYSQL mechanism to manage data, which is a high-concurrency and low-latency analytical database to solve PB-level data [8].

7.1.3.1 Modelling the Architecture in Case 3

The data in this case is transferred from sources to the Pangu cloud storage organised by AnalyticDB. As discussed above, these 16 ADB nodes do not store original warehouse data, which are for responding to queries. Based on [12], end users submit SQL queries to the ADB clients, then the ADB coordinator compiles all queries and assigns them to different ADB Read/Write nodes. Each node individually addresses assignments, finally results are gathered by the coordinator and sent to end users. In terms of the DWH, the snowflake is utilised to organise data with two types of tables including dimension tables and ordinary tables. The size of the dimension table is limited, but the size of the ordinary table could be bulky and need to be split. Hence, there are two levels to store data in normal tables. Data by default is allocated to the first-level table. If more data should be imported, the second-level table may be created to address the increased data

[13]. As can be seen, the dimension tables and normal tables in AnalyticDB are the same to the concepts of dimension tables and fact tables in DWH schema, respectively. As the data in normal tables could be partitioned and stored in different places, which is similar to the main mechanism of the schema in a PDWH (parallel Data Warehouse). In this type of DWH, data belonging to a table may be separated into different places, so this DWHA is classified to the PDWH. This system applies view and OLAP to swiftly answer queries. As the OLAP is built on the relational database and no more information is provided, it is classified into ROLAP. Based on the different types of queries, there are two types of end users (content and knowledge users). Therefore, the modelled DWHA in this case is presented in figure 7.9.

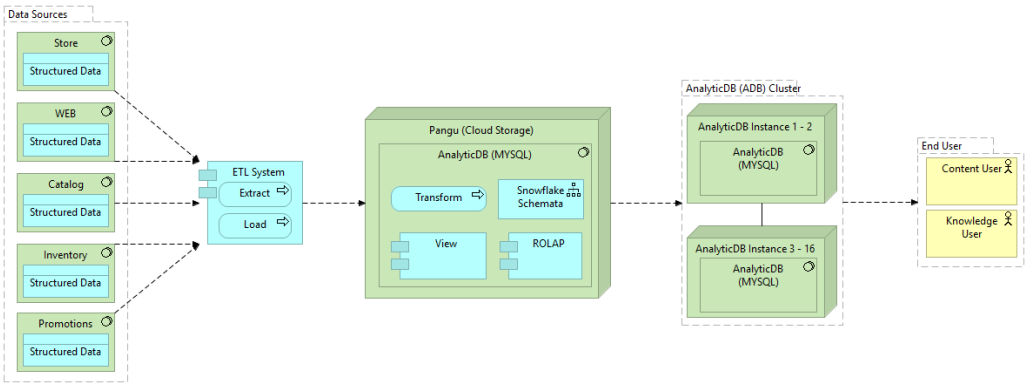


Figure 7.9: The modelled DWHA of the case 3

7.1.3.2 Measuring the Features in Case 3

Most of macroscopical features are discussed and modelled in the previous section above. According to [12], case 1, case 2 and case 3 use the same data sets, so the evaluated result of data quality is 2. In terms of the load throughput, around 31075.79836 MB data or 285,715,667 rows are transferred in 1,831.57 and 1,833.71 seconds in two tests. Hence, the average of throughput is around 16.96 MB/Second or 155,903.92 Rows/Second. This DWH needs to be updated daily,

so the availability per period is around 2.947. When measuring its query performance, two tests are conducted, which take 54,653.48 and 65,205.20 seconds with 99 queries. Hence, the query throughput is 59470.04 Query/Hour@10000GB. In addition, this system provides services for content user and knowledge users and covers 5 types of metadata based on the report. According to the modelled DWHA in figure 7.9, the number of edges, nodes and sources are 8, 9 and 5 respectively. Hence, the architecture complexity is 5. These measured results are added into this modelled DWHA and presented in figure 7.10.

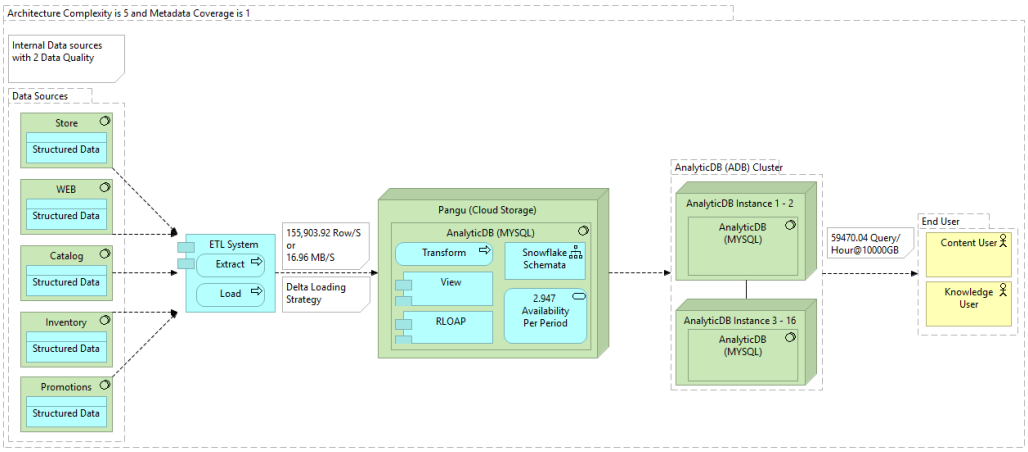


Figure 7.10: The modelled DWHA of the case 3 with measured results

7.1.3.3 Evaluated Result for Case 3

In this case, 15 sub-features are measures and the results are input into the framework illustrated in figure 7.11. As can be seen, most of macroscopical features are similar to case 1 and case 2, while microscopical features show different abilities and performance. It indicates that macroscopical features are insensitive to compare some cases but microscopical features are more sensitive to identify their differences. To be more specific, in the source layer, all data in these three cases is derived from five internal sources which are structured with two types of data quality issues. But these three cases' loading throughput are different. If only

comparing evaluated results of microscopical features, it is easy to identify differences between cases, but evaluated results could not adequately interpret and describe DWHAs. It implies these two types of features should cooperate together to evaluate DWHAs from different aspects. In addition, all evaluated results of these sub-features are listed below. Furthermore, the evaluated results of macroscopical features are normalised and demonstrated in figure 7.12.

Case 3 - Alibaba Cloud AnalyticDB

	Feature	Sub Feature	Feature Measurement									
Source Layer	Source	Data Source	Unknown	Internal Source	External Source	Both						
	Data	Data Format	Unknown	Structured Data	Unstructured Data	Both						
		Data Quality	$DataQuality = Round(\frac{Count(D_i) - Subsistent(D_i)}{Count(D_i)} \times 3)$ $D_i \in$ (Missing Values, Conflict of Entities, Format Incompatibility, Multi-Resource, Multi-Table Files, Multi-Meaning Attributes) 2									
ETL Layer	Process	ETL Type	Unknown/None	E	EL	ET	ELT	ETL				
		Loading Strategy	Unknown/None	Full	Delta	Both						
		Loading Throughput	$LoadingThroughput = \frac{\sum_{i=1}^{n} \frac{Records_i \ V \ DataSize_i}{Max(Time_i, 180)}}{n}$ 155903.92 Row/S or 16.96 MB/S									
Data Storage Layer	Warehouse	Data Warehouse Type	Unknown /Other	V	IDM	C	DMB	HAS	FDWH	PDWH	BDWH	DWHDL
	Table	Schema Type	Unknown/Other	Star	Snowflake	Constellation						
	Time	Time Period Rate	$AvailabilityPerPeriod = \frac{\sum_{i=1}^{n} \frac{Period - UpdateTime_i}{Period}}{n} \times 3$ 2.947									
Presentation Layer	Business Intelligence	OLAP Type	Unknown/None	ROLAP	MOLAP	HOLAP						
	Table	View	Unknown/None	View	Materialised View	Both						
	Query	Query Throughput	$QueryThroughput@SizeFactor = \frac{3600 \times n \cdot SizeFactor}{\sum_{i=1}^{n} QueryResponseTime_i}$ 59470.04 Query/Hour@10000GB									
Other	User	User Type	Unknown/Other	Content User	Knowledge User	Both						
	Data	Meta Data Type	$MetadataCoverage = Round(\frac{\sum_{i=1}^{n} TypesofMetadataintheLayer_i}{13} \times 3)$ $(n = 4)$	0	1	2	3					
	Architecture	Architecture Complexity	$C = E - N + S + 1$ 5									

V: Virtual, IDM: Independent Data Marts, C: Centralised, DMB: Data Mart Bus, HAS: Hub and Spoke, FDW: Federated DWH, PDW: Parallel DWH, BDWH: Big DHW, DWHDL: DWH with Data Lake

Figure 7.11: The evaluated results of the case 3

Evaluated Results in the Source Layer: (1, 1, [2])

Evaluated Results in the ETL Layer: (3, 2, [1555903.92, 16.96])

Evaluated Results in the DWH Layer: (3, 2, [2.947])

Evaluated Results in the Presentation Layer: (1, 1,[59470.04])

Evaluated Results in the other (3, 1, [5])

Figure 7.12: The macroscopic feature evaluated results of the case 3

7.1.4 Comparison of Cases from a Public Data Set

When gathering the radar charts of DWHAs from these three cases, it is easy to identify the differences between them. The gathered diagram is illustrated in figure 7.13. As can be seen, in terms of the meta data, case 2 covers more types than case 1 and 2, while case 1 present better than case 2 and 3 referring to the OLAP. By further investigating and comparing their documents, when measuring the meta data types, the DWHA in case 2 provides evidence that it manage and store more meta data types that other 2 DWHAs. As discussed in section 6.1.1.2, meta data is essential and significant to record information on how a DWH system is created, accessed and used; The case 1 uses MOLAP. Case 2 does not provide

detailed information on this feature, so it is measured with a low score. case 3 uses ROLAP, which is lightweight compared with MOLAP. In this part, three DWHAs from case 1, 2 and 3 are compared as an example to demonstrate the processes and analyse the reason. This gathered radar chart can also be leveraged to compare DWHAs in other cases.

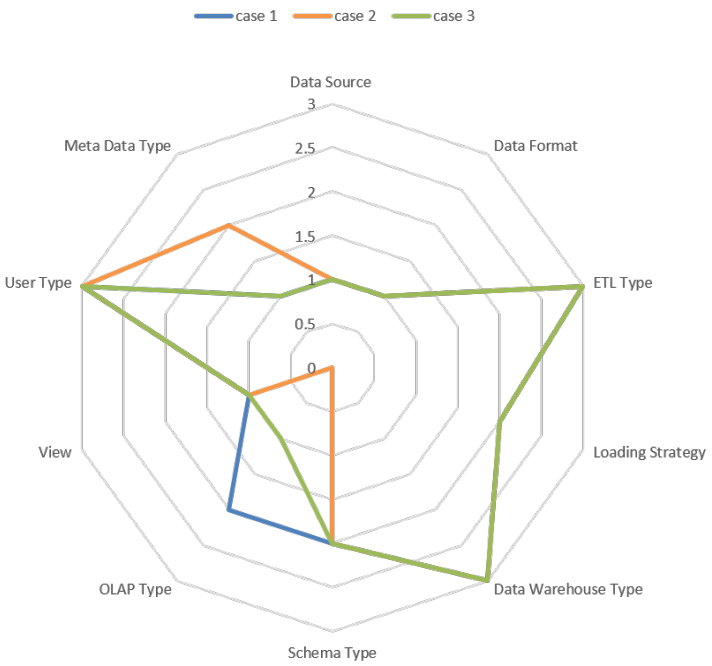


Figure 7.13: The evaluated results of the cases from a public data set

7.2 Evaluating Data Warehouse Architectures Developed in This Research

Two DWH systems are developed in this research and their architectures are based on [278, 279]. However, some changes are made to better fit this method. These two DWHs are established in the Hadoop environment and Oracle. Log data is involved and analysed to understand browsed web resources and visitors' activities. In order to evaluate and compare these two DWHAs, the same data sets and hardware are leveraged in these two architectures. In addition, a schema with

the same structure of fact and dimension tables is applied in both DWH systems. Unlike the partitioned DWHs with a hierarchical schema and split fact tables in [278], these two DWH systems only have one fact table and five dimension tables, and their structure and relationships are shown in [279].

7.2.1 Evaluating the Data Warehouse Architecture in Case 4

Technical Environment: This system is based on the Hadoop environment which is built in a machine (Intel Core(TM) i7 CPU 3.60GHz, 16 GB RAM, Disk 931.51 GB) associated with nine Linux virtual machines. Four of them are set up as data resources. The rest of them are used to establish the Hadoop platform. Their detailed configuration is presented in table 7.1. HDFS is used in this system, which provides high availability in relation to accessing data. The Yarn is built in Node 5 to separate the work of resource management and schedule into different daemons [280]. Two name nodes are configured in Node 6 and Node 7 to master other nodes and enable them working together. The data is placed in date nodes which are located at Node 8, Node 9 and Node 10 in this system. These data nodes not only store data but need to manipulate or analyse data. The data set is derived from the 1998 World Cup website [275], which includes log of requests made in ten days from 30th April, 1998 to 9th May, 1998. The volume of this 10 days data log is around 1 GB.

7.2.1.1 Modelling the Architecture in Case 4

The data is originally from website log, which is classified as unstructured data based on the concept proposed by [289]. In this system, Flume is applied to fetch and transfer web log files from physically independent servers into HDFS, but it does not cleanse and format data. This tool has the ability to provide a distributed service for efficiently extracting and delivering large amounts of log data from data sources to destinations, which can be used to work in collaboration with other tools in Hadoop environments [67]. Hive is employed in this case to

Node Name	Operation System	Processor	Memory	Software
Node 1	Centos 7 X86_64	1	1GB	-
Node 2	Centos 7 X86_64	1	1GB	-
Node 3	Centos 7 X86_64	1	1GB	-
Node 4	Centos 7 X86_64	1	1GB	-
Node 5	Centos 7 X86_64	1	1GB	JDK 1.7, Hadoop, HBase, Hive
Node 6	Centos 7 X86_64	1	1GB	JDK 1.7, Hadoop, HBase
Node 7	Centos 7 X86_64	1	1GB	JDK 1.7, Hadoop, HBase
Node 8	Centos 7 X86_64	1	1GB	JDK 1.7, Hadoop, HBase, Zookeeper
Node 9	Centos 7 X86_64	1	1GB	JDK 1.7, Hadoop, HBase, Zookeeper
Node 10	Centos 7 X86_64	1	1GB	JDK 1.7, Hadoop, HBase, Zookeeper

Table 7.1: The configuration of the system in case 4

establish a DWH. MYSQL database is allocated in the Hadoop system to save metadata for this DWH. HBase is applied in this system to generate the LogId, PathId, DomainId and other ID when preparing data for this DWH. It is a NoSQL database system to manage large data sets and large tables with billions of rows millions of columns, which can be seamlessly associated with other mechanisms and tools (MapReduce, HDFS, Yarn, etc.) in Hadoop environments [84]. When new records should be populated into the DWH, they need to be checked if they are already stored in the dimension tables.

In addition, the OLAP cubes are stored in HBase and MYSQL located inside and outside of the hadoop platform for different purposes. Views are created and materialised in the MYSQL database system. Sqoop is used to transfer data between the Hadoop platform and the MYSQL system. It can exchange data between a relational database management system and HDFS [226]. If the OLAP and views have limited data and MYSQL could maintain all of them, then they can locate in the MYSQL, otherwise, they could be distributed in the HBase sys-

tem. The architecture of this DWH system is similar to the architecture presented in [279] but with four data sources to simulate HTTP servers at four geographic locations (Paris, France; Plano, Texas; Herndon, Virginia; and Santa Clara, California) as described in [275]. This whole system is designed to offer different types of services and functions including basic queries and advanced analysis. The detailed information of this system is modelled and presented in figure 7.14.

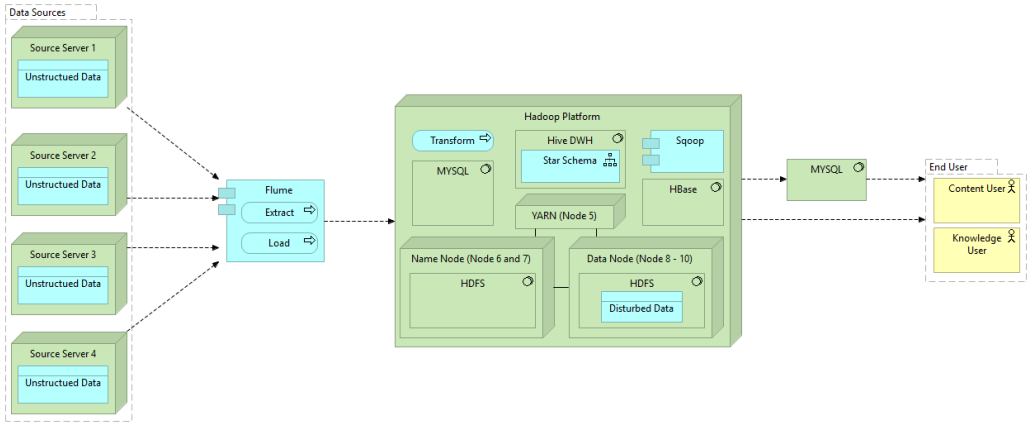


Figure 7.14: The modelled DWHA in case 4

7.2.1.2 Measuring the Features in Case 4

According to the description in [275], The log data is extracted from data sources located at different places, so it matches the conditions of external sources and unstructured data. By examining the log, there are 3 types of data quality issues in the raw data, so the score for data quality is 2. These log records are extracted from data sources and transferred to HDFS, then they are transformed in the hadoop platform. Hence, the ELT is applied in this system. This DWH should be updated with new records daily, and the delta load strategy is applied for update. The data in this DWH system is organised by a star schema in Hive, so the BDWH and star schema match its situation. As for the load speed, two updates are measured in last two days (8^{th} May, 1998 and 9^{th} May, 1998). In the first update, around

141.88 MB or 1,646,315 records are transferred and manipulated in 8,311.27 seconds. In the second update, around 84.10 MB or 975,879 records are operated in 5,476.83 seconds. Based on the relevant equation in the framework, the load speed is 190.18 Row/Second or 0.01639 MB/Second. As the period of update is 24 hours, so the average of the availability per period is 2.761. The OLAP is created and views are materialised in the MYSQL database. In terms of the query speed, some normal and ad-hoc queries (e.g. the daily and weekly page views and unique visitors) are tested. 25 queries are measured in which 15.94 and 16.41 minutes are taken in two tests. Hence, the query speed is 92.74 Query/Hour. This DWH system provides different types of services for content users and knowledge users. By checking the metadata in this system, 10 different types of metadata are identified in this DWH system, so the score for the metadata coverage is 2. In this architecture, there are 8 edges, 8 nodes and 4 sources, thus, the complexity of this system is 5. Furthermore, these measured results are added in the modelled DWHA which is demonstrated in figure 7.15.

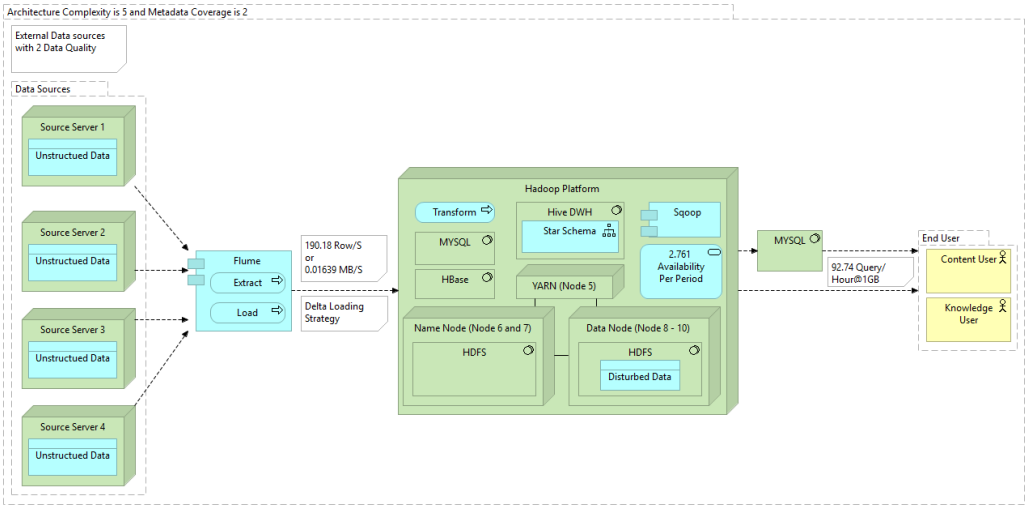


Figure 7.15: The modelled DWHA of the case 4 with measured results

7.2.1.3 Evaluated Result for Case 4

After extracting and measuring all sub-features, these measured results are filled into the framework illustrated in figure 7.16. Subsequently, the evaluated results are extracted from the framework and listed below. The normalised results of macroscopical features are diagrammed in figure 7.17. As can be seen, this DWH system has relatively high scores referring to the macroscopical features such as the ETL type and DWH type, as it builds an advanced DWH in the big data platform. However, by comparing its microscopical features with previous cases', this DWHA shows lower ability to transfer, manipulate and analyse data. As can be observed from this case, there is a loose link between macroscopical features and microscopical features referring to their evaluated results. Even macroscopical features are evaluated with high scores, but its microscopical features could performance low scores.

Evaluated Results in the Source Layer: (2, 2, [2])

Evaluated Results in the ETL Layer: (3, 2, [190.18, 0.01639])

Evaluated Results in the DWH Layer: (3, 1, [2.761])

Evaluated Results in the Presentation Layer: (1, 2,[92.74])

Evaluated Results in the other (3, 2, [5])

Case 4 - DWHA in Hadoop (Experiment 1)											
	Feature	Sub Feature	Feature Measurement								
Source Layer	Source	Data Source	Unknown	Internal Source	External Source	Both					
	Data	Data Format	Unknown	Structured Data	Unstructured Data	Both					
		Data Quality	$DataQuality = Round(\frac{Count(D_i) - Subsistent(D_i)}{Count(D_i)} \times 3)$ $D_i \in$ (Missing Values, Conflict of Entities, Format Incompatibility, Multi-Resource, Multi-Table Files, Multi-Meaning Attributes) 2								
ETL Layer	Process	ETL Type	Unknown/None	E	EL	ET	EET	ETL			
		Loading Strategy	Unknown/None	Full	Delta	Both					
		Loading Throughput	$LoadingThroughput = \frac{\sum_{i=1}^{n} \frac{Records_i \ V \ DataSize_i}{Max(Time_i, 180)}}{n}$ 190.18 Row/S or 0.01639 MB/S								
Data Storage Layer	Warehouse	Data Warehouse Type	Unknown /Other	V	IDM	C	DMB	HAS	FDWH	PDWH	BDWH	DWHDL
	Table	Schema Type	Unknown/Other	Star	Snowflake	Constellation					
	Time	Time Period Rate	$AvailabilityPerPeriod = \frac{\sum_{i=1}^{n} \frac{Period - UpdateTime_i}{Period}}{n} \times 3$ 2.761								
Presentation Layer	Business Intelligence	OLAP Type	Unknown/None	ROLAP	MOLAP	HOLAP					
	Table	View	Unknown/None	View	Materialised View	Both					
	Query	Query Throughput	$QueryThroughput@SizeFactor = \frac{3600 \times n \cdot SizeFactor}{\sum_{i=1}^{n} QueryResponseTime_i}$ 92.74 Query/Hour@1GB								
Other	User	User Type	Unknown/Other	Content User	Knowledge User	Both					
	Data	Meta Data Type	$MetadataCoverage = Round(\frac{\sum_{i=1}^{n} TypesofMetadataintheLayer_i}{13} \times 3)$ $(n = 4)$			0	1	2	3		
	Architecture	Architecture Complexity	C = E – N + S + 1 5								
V: Virtual, IDM: Independent Data Marts, C: Centralised, DMB: Data Mart Bus, HAS: Hub and Spoke, FDW: Federated DWH, PDW: Parallel DWH, BDWH: Big DHW, DWHDL: DWH with Data Lake											

Figure 7.16: The evaluated results of the case 4

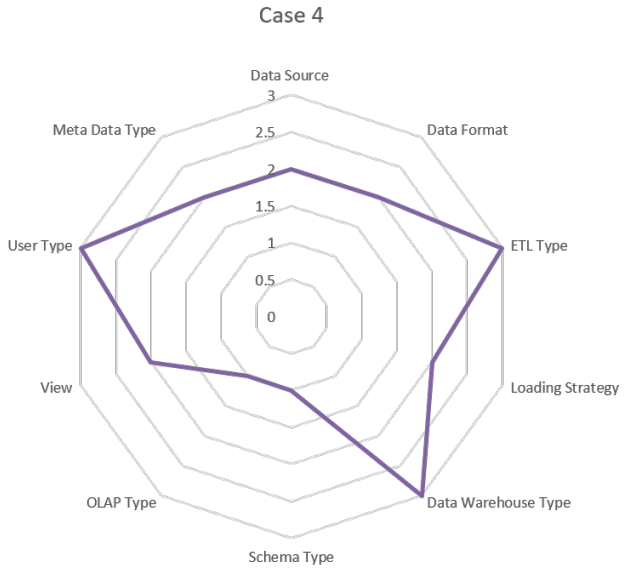


Figure 7.17: The macroscopic feature evaluated results of the case 4

7.2.2 Evaluating the Data Warehouse Architecture in Case 5

Technical Environment: The DWH system in this case is built in the same physical machine used in case 4. These two cases are mutually exclusive in two experiments, when one experiment is conducted, another one should be terminated without occupying resource. Unlike case 4 having ten nodes, in this case, seven virtual machines are established and share the resource of this machine. Four of them (Node 1, Node 2, Node 3 and Node 4) are set as data sources, one server (Node 4) is employed to build the DWH system, two servers (Node 6 and Node 7) are leveraged to build two DMs for different purposes, which obtain a portion of data from the DWH. Hence, in terms of the relationship, these two DMs should depend on this DWH. All virtual machines are operated by Linux under Centos 7. The Oracle database is utilised to organise and manage data for this DWH and these two DMs. In order to make a trade-off between this case and case 4, the total memories (10 GB) allocated to these virtual machines are the same in these two cases, so Node 5, 6 and 7 acquire 2 GB memories and the rest of them occupy 1 GB. The details of each node are presented in table 7.2. In addition, the data executed in this case is the same as log records in case 4.

Node Name	Operation System	Processor	Memory	Software
Node 1	Centos 7 X86_64	1	1GB	-
Node 2	Centos 7 X86_64	1	1GB	-
Node 3	Centos 7 X86_64	1	1GB	-
Node 4	Centos 7 X86_64	1	1GB	-
Node 5	Centos 7 X86_64	1	2GB	Oracle
Node 6	Centos 7 X86_64	1	2GB	Oracle
Node 7	Centos 7 X86_64	1	2GB	Oracle

Table 7.2: The configuration of the system in case 5

7.2.2.1 Modelling the Architecture in Case 5

This DWH system is modelled and presented in figure 7.20. It is designed to acquire log files from four sources located in physically separated servers. They act as external data sources and the data in these servers is unstructured. As for the ETL layer, it is designed to provide services to extract, transform and load data from these four sources to the DWH system, so it relies on the normal ETL mechanism to transfer and manipulate data. When testing its update, new records generated in two days (8^{th} and 9^{th} May, 1998) are loaded via the delta load strategy. As for the data management in the DWH, a star schema is created and has the identical structure as the schema in case 4, which has one fact table and five dimension tables. Subsequently, two DMs are established to offer special services, which focus on analysing two web page views (index.html and playing/body.html). The same star schema is used in these DMs to organise data, but these two DMs do not store all data in the DWH, rather only maintain relevant records for page view analysis. Therefore, this system relies on the mechanism of the Hub and Spoke structure to build this DWH. Since these two DMs do not have all data, end users may need to visit the main DWH for some ad hoc queries. The materialised views and ROLAP are designed and distributed in either the DWH or DMs depending on their purposes. This system provides services for both content and knowledge users.

7.2.2.2 Measuring the Features in Case 5

By examining raw data, there are three typical quality issues (conflict of entities, format incompatibility and multi-table files), so the data quality is 2. In order to test the update speed, two updates are conducted with the same data sets used in case 4. In the first update, 93.41 minutes are taken for populating 141.88 MB with 1,646,315 records. In the second update, 52.53 minutes are consumed. Thus, its update speed is 299.46 Row/Second or 0.02581 MB/Second. This DWH system should be refreshed daily, so it provides 94.93% of the time available to end users

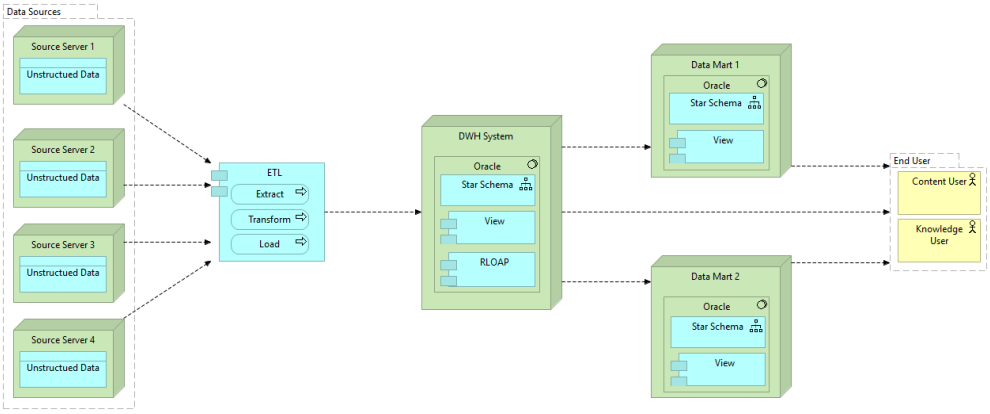


Figure 7.18: The modelled DWHA in case 5

for querying in a period, so the score of the the availability per period is around 2.848. When measuring the query speed, the same queries in case 4 are executed to test its performance in normal and ad-hoc situations, in which 14.77 and 14.18 minutes are taken in two tests. Thus, the query speed can be calculated, which is around 103.63 Query/Hour@1GB. By observing the modelled pattern, there are 10 edges, 9 nodes and 4 sources, the result is 6 based on the architecture complexity equation. Furthermore, this DWHA is re-modelled and demonstrated in figure 7.19.

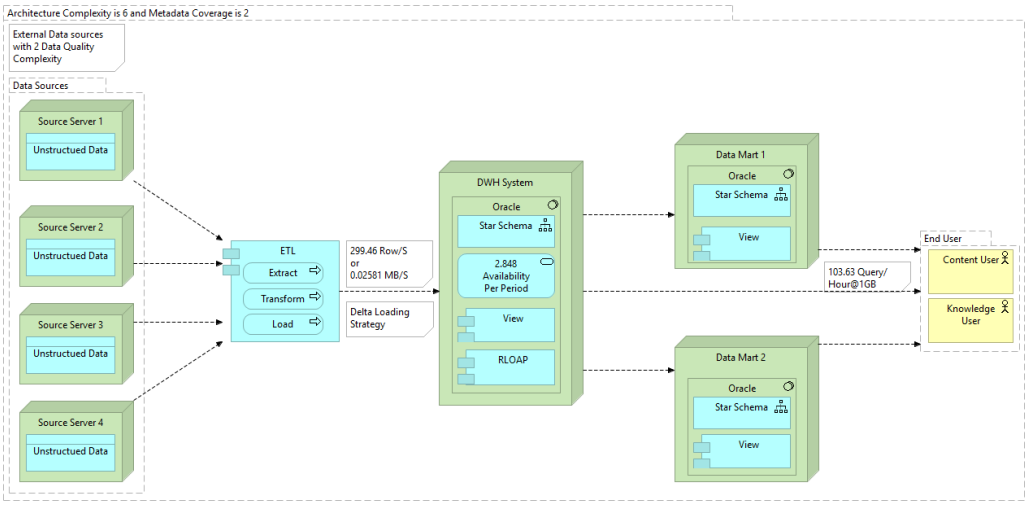


Figure 7.19: The modelled DWHA of the case 5 with measured results

7.2.2.3 *Evaluated Result for Case 5*

These measured results are fed into the evaluation framework as shown in figure 7.20. Subsequently, the evaluated results are extracted and listed below. In addition, the results for macroscopic features are extracted and demonstrated in figure 7.21. As can be observed, it has high scores in some sub-features in relation to the ETL and User. Although this system only manages 1 BG data, its architectural complexity is 6 which is the same to case 2 having 10000GB. This value means when testing the whole system' data flows, at least 6 tests should be conducted to cover all flows. It indicates there is a loose link between the architecture complexity and the data set. A DWHA could be designed complex even when the volume of its data set is not enormous.

Case 5 - DWHA in Oracle (Experiment 2)

	Feature	Sub Feature	Feature Measurement									
Source Layer	Source	Data Source	Unknown	Internal Source	External Source	Both						
	Data	Data Format	Unknown	Structured Data	Unstructured Data	Both						
		Data Quality	$DataQuality = Round(\frac{Count(D_i) - Subsistent(D_i)}{Count(D_i)} \times 3)$ $D_i \in$ (Missing Values, Conflict of Entities, Format Incompatibility, Multi-Resource, Multi-Table Files, Multi-Meaning Attributes) 2									
ETL Layer	Process	ETL Type	Unknown/None	E	EL	ET	ELT	ETL				
		Loading Strategy	Unknown/None	Full	Delta	Both						
		Loading Throughput	$LoadingThroughput = \frac{\sum_{i=1}^{n} \frac{Records_i \ V \ DataSize_i}{Max(Time_i, 180)}}{n}$ 299.46 Row/S or 0.02581 MB/S									
Data Storage Layer	Warehouse	Data Warehouse Type	Unknown /Other	V	IDM	C	DMB	HAS	FDWH	PDWH	BDWH	DWHDL
	Table	Schema Type	Unknown/Other	Star	Snowflake	Constellation						
	Time	Time Period Rate	$AvailabilityPerPeriod = \frac{\sum_{i=1}^{n} \frac{Period - UpdateTime_i}{Period}}{n} \times 3$ 2.848									
Presentation Layer	Business Intelligence	OLAP Type	Unknown/None	ROLAP	MOLAP	HOLAP						
	Table	View	Unknown/None	View	Materialised View	Both						
	Query	Query Throughput	$Query\ Throughput@SizeFactor = \frac{3600 \times n \cdot SizeFactor}{\sum_{i=1}^{n} QueryResponseTime_i}$ 103.63 Query/Hour@1GB									
Other	User	User Type	Unknown/Other	Content User	Knowledge User	Both						
	Data	Meta Data Type	$MetadataCoverage = Round(\frac{\sum_{i=1}^{n} TypesofMetadataintheLayer_i}{13} \times 3)$ $(n = 4)$	0	1	2	3					
	Architecture	Architecture Complexity	C = E – N + S + 1 6									

V: Virtual, IDM: Independent Data Marts, C: Centralised, DMB: Data Mart Bus, HAS: Hub and Spoke, FDW: Federated DWH, PDW: Parallel DWH, BDWH: Big DHW, DWHDL: DWH with Data Lake

Figure 7.20: The evaluated results of the case 5

Evaluated Results in the Source Layer: (2, 2, [2])

Evaluated Results in the ETL Layer: (3, 2, [299.46, 0.02581])

Evaluated Results in the DWH Layer: (2, 1, [2.848])

Evaluated Results in the Presentation Layer: (1, 2, [103.63])

Evaluated Results in the other (3, 2, [6])

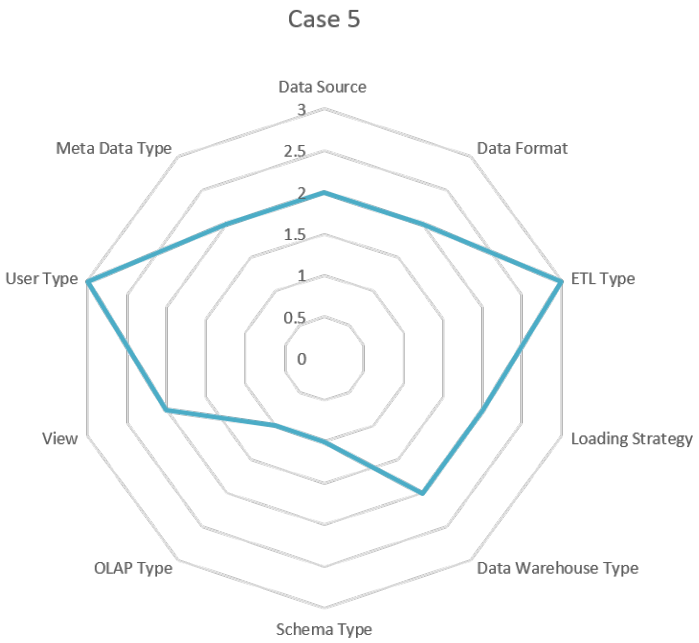


Figure 7.21: The macroscopic feature evaluated results of the case 5

7.2.3 Evaluating Data Warehouse Architecture in Case 6

Technical Environment: The whole system is established in five separated servers from the operational system to the BI system using Unix and Oracle to operate these systems and store data respectively. Data is generated by clients in transactions and transferred to the DWH and BI system in work days. Comparing with

the previous DWHs presented in this chapter, this system has a data staging area in a separated node for daily data update, which leverages an Oracle database management system to temporarily store and transform data being prepared for the DWH. There is no specific ETL tool applied in this architecture to manipulate and transfer data between physically divided servers, which relies on commands, functions or procedures to conduct these processes.

7.2.3.1 Modelling the Architecture in Case 6

This DWH system obtains data from a single data source to provide services such as reports and data analysis. This DWH system does not have data from external sources and only has an internal data source in which data is structured. This system extracts original data and puts into the staging area for transformation, then populates into the DWH. Thus, it relies on the ETL mechanism to coduct these processes. However, in this system, the data for this DWH is not directly extracted from the operational system. Between the operational system and staging area, there is a server, which copies all operational data in order to reduce the effect of the operational system when updating data. As for the loading strategy, the full loading is executed daily except weekends. As this DWH obtains data from a single data source and there is no DM in this system, so its DWH is classified into the Idependent DM. The data in this DWH system is organised in a star schema including a fact table and multiple dimension tables. In the BI system, the ROLAP and views are created for efficient responding to queries inquired by end users. As this system is for normal requirements and no advanced requirements and tasks (Data Mining) should be achieved, it provides services for content users. This DWHA is modelled and demonstrated in figure 7.22.

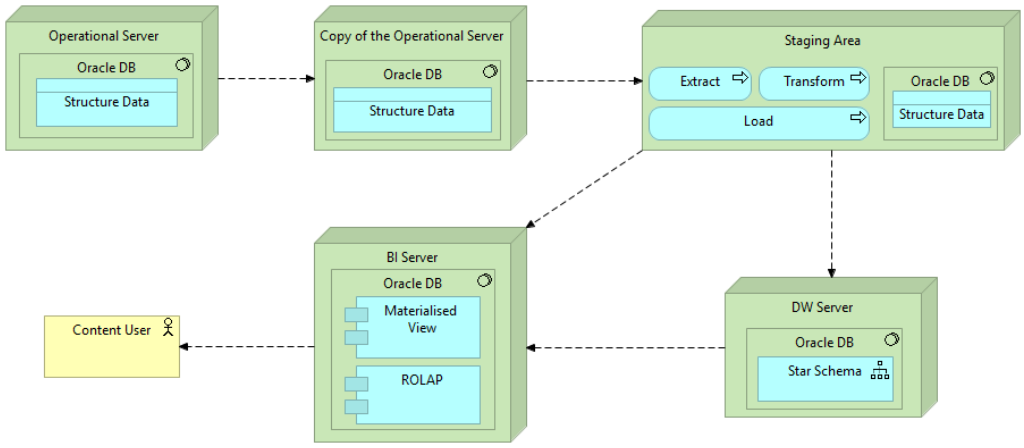


Figure 7.22: The modelled DWHA of the case 6

7.2.3.2 Measuring the Features in Case 6

Through examining this data set, three typical data quality issues are identified, including missing values, format incompatibility and multi resources. In the ETL layer, in order to test the throughput of the update, two updates are randomly chosen. There are around 94,277,379 records transferred in each of two updates taking 39,012.26 and 38,694.32 seconds, so the load throughput is around 2,426.50 Row/Second. There is a data flow from the staging area to the BI server, but this flow is few compared with the main flow, so its loading throughput is not considered hereby. In the DWH layer, this DWH is updated daily except weekends, so the period is daily and the average time consumption is 38,853.29 seconds. Thus, the availability per period is around 1.651. As this DWH system is deployed and available to end users, so the query throughput is not measured, otherwise it could affect the performance of the whole system. According to the modelled pattern in figure 7.22, this architecture has 6 edges, 6 nodes and 1 source, consequently, the architecture complexity of this DWHA is 2. After these sub-features are measured, it is remodelled with these evaluated results and demonstrated in figure 7.23.

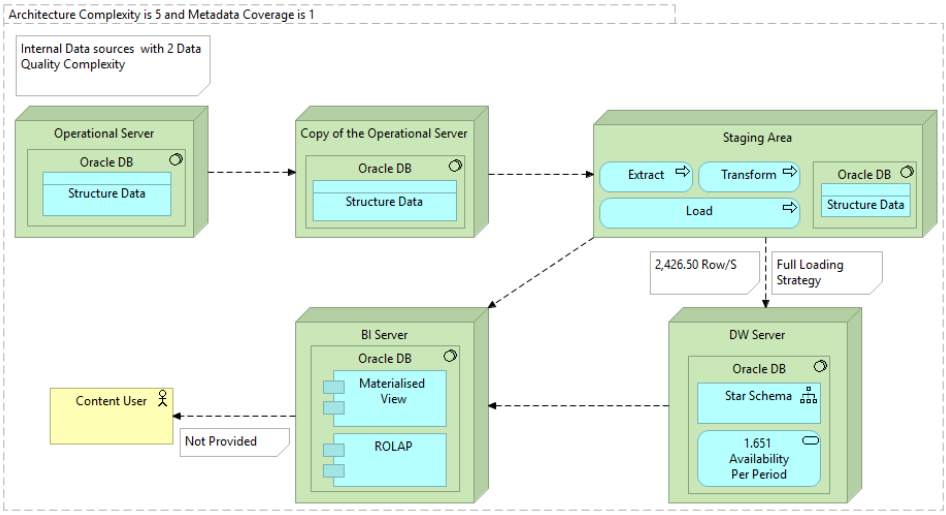


Figure 7.23: The modelled DWHA of the case 6 with measured results

Case 6 - A DWHA from A Cooperative Company											
	Feature	Sub Feature	Feature Measurement								
Source Layer	Source	Data Source	Unknown	Internal Source	External Source	Both					
	Data	Data Format	Unknown	Structured Data	Unstructured Data	Both					
		Data Quality	$DataQuality = Round(\frac{Count(D_i) - Subsistent(D_i)}{Count(D_i)} \times 3)$ $D_i \in$ (Missing Values, Conflict of Entities, Format Incompatibility, Multi-Resource, Multi-Table Files, Multi-Meaning Attributes) 2								
ETL Layer	Process	ETL Type	Unknown/None	E	EL	ET	ELT	ETL			
		Loading Strategy	Unknown/None	Full	Delta	Both					
		Loading Throughput	$LoadingThroughput = \frac{\sum_{i=1}^{n} \frac{Records_i \; V \; DataSize_i}{Max(Time_i, 180)}}{n}$ 2426.50 Row/S								
Data Storage Layer	Warehouse	Data Warehouse Type	Unknown /Other	V	IDM	C	DMB	HAS	FDWH	PDWH	BDWH	DWHDL
	Table	Schema Type	Unknown/Other	Star	Snowflake	Constellation					
	Time	Time Period Rate	$AvailabilityPerPeriod = \frac{\sum_{i=1}^{n} \frac{Period - UpdateTime_i}{Period}}{n} \times 3$ 1.651								
Presentation Layer	Business Intelligence	OLAP Type	Unknown/None	ROLAP	MOLAP	HOLAP					
	Table	View	Unknown/None	View	Materialised View	Both					
	Query	Query Throughput	$QueryThroughput@SizeFactor = \frac{3600 \times n \cdot SizeFactor}{\sum_{i=1}^{n} QueryResponseTime_i}$ Not Measured								
Other	User	User Type	Unknown/Other	Content User	Knowledge User	Both					
	Data	Meta Data Type	$MetadataCoverage = Round(\frac{\sum_{i=1}^{n} TypesofMetadataintheLayer_i}{13} \times 3)$ $(n = 4)$	0	1	2	3				
	Architecture	Architecture Complexity	$C = E - N + S + 1$ 2								

V: Virtual, IDM: Independent Data Marts, C: Centralised, DMB: Data Mart Bus, HAS: Hub and Spoke, FDW: Federated DWH, PDW: Parallel DWH, BDWH: Big DHW, DWHDL: DWH with Data Lake

Figure 7.24: The evaluated results of the case 6

7.2.3.3 Evaluated Result for Case 6

This DWH is evaluated with 14 aspects as the query throughput is not provided and the results are put into the framework presented in figure 7.24. Subsequently. the evaluated results are extracted and listed below. In addition, the results of the macroscopic features are extracted and demonstrated in figure 7.25. By comparison to the previous cases, the macroscopical features of this DWHA do not require high scores, which means that it may not need high level of these related components such as an advanced DWH or an external data source. Besides, the availability per period is lower compared with values in previous cases, which is 1.651, while others are normally above 2.5.

Figure 7.25: The macroscopic feature evaluated results of the case 6

Evaluated Results in the Source Layer: (1, 1, [2])

Evaluated Results in the ETL Layer: (3, 1, [2426.50])

Evaluated Results in the DWH Layer: (1, 1, [1.651])

Evaluated Results in the Presentation Layer: (1, 2, [])

Evaluated Results in the other (1, 1, [2])

Indicator	Case 1	Case 2	Case 3	Case 4	Case 5	Case 6
Source & Format Ave.	1	1	1	2	2	1
Data Quality	2	2	2	2	2	2
ETL & Load Ave.	2.5	2.5	2.5	2.5	2.5	2
loading throughput	2.6123	2.539	2.12	0.002048	0.003227	0.9105
DWH & Schema Ave.	2.5	2.5	2.5	2	1.5	1
Time Period Rate	2.948	2.946	2.947	2.761	2.848	1.6508
OLAP & View Ave.	1.5	0.5	1	1.5	1.5	1.5
Query Throughput	1.589	2.368	1.784	0.0278	0.03108	0
User & Meta Ave.	2	2.5	2	2.5	2.5	1
Architecture Complexity	2.5	3	3	2.5	3	1

Table 7.3: The normalised evaluated results of the DWHAs in the 6 cases

7.2.4 Comparison of the DWHAs in Cases

After these 6 DWHAs are evaluated, their results are normalised and tabulated in table 7.3. As sensitive features have 10 aspects and insensitive features have 5 aspects, in order to strike a balance, every 2 sensitive features in a layer are replaced by the average of their values. As can be seen, the results are scoped into the range from 0 to 3 and there are compressed into 10 indicators. Subsequently, they are illustrated and presented in the figure 7.26. Through the table and radar chart, these DWHAs can be easily compared. To be more specific, when comparing the case 1, 2 and 3, the sensitive features (e.g. loading throughput) show more distincter than the insensitive features (discussed in section 7.1.4). Furthermore, when comparing the DWHAs across different environments and data sizes from case 1 to case 6. The insensitive features may show similar between them, but the sensitive features reflect wide differences. For instance, the loading throughput

and query throughput in case 1, 2 and 3 are greater than case 4, 5 and 6. The reason is that they are designed for systems with different size of data or they have specific purposes.

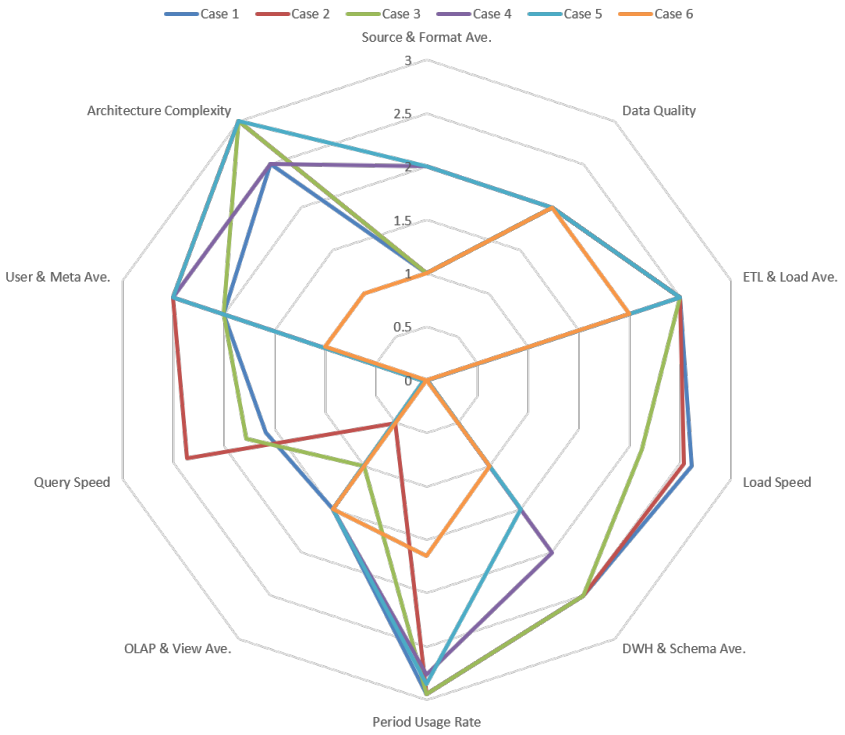


Figure 7.26: The radar chart with 6 DWHAs

7.3 Empirical Research for Method Evaluation

Empirical research can make contributions to design oriented research with the variety of artefacts and consistently form knowledge with empirical validation supported [109]. In empirical research, the interview is one of the approaches to collect data and identify empirical evidence to evaluate output [120]. In this research, interviews are conducted to assess the results of these evaluated cases and measure whether the objectives of the present research are achieved or not. Based on the circumstances and configurations, these cases are briefly divided into 3 categories, in which case 1, 2 and 3 are allocated to group A, case 4 and 5

are placed in group B, case 6 is assigned to group C. All cases in the same group leverage the same data and queries to conduct the evaluation.

The interview is designed with several close-ended questions and open-ended questions which are related to the explicitness of the modelled and evaluated results, the comparison of evaluated DWHAs and the efficiency of this method. 3 individuals with different background are invited to iteratively update and test these questions and tasks in order to enable these questions and tasks easily to understand and respond. Considering the workload, a participant is asked to answer these questions with two cases each time. These two cases are acquired from two parts described as follows: (1) The first part is for comparing cases with different circumstances, which randomly draws one from group A, B and C, then assign two of them in a comparison. In this step, case 1, 5 and 6 are drawn, so the comparative sets are (1,5),(1,6),(5,6). (2) The second part is for comparing cases in the same circumstances in order to make a balance. three sets are randomly selected, including (1,2),(2,3),(4,5). So all sets are determined, which will be used in interviews. As it is not adequately just comparing each of these 6 two-case sets once, in this research, 18 people are invited to conduct interviews, in which each set can be assessed by 3 participants and results can be more convinced. They are allowed to ask questions referring to basic background and knowledge which does not affect their answers. The processes of the interviews are designed as follows:

1. Participants read and sign a consent form in which their data will not be revealed, no personal/sensitive data or no ethical question is involved in this interview.

2. Participants are given four modelled DWHAs referring to two cases, including two modelled DWHAs with original information and two modelled DWHAs with extended information. They answer the questions in part 1 based on these modelled DWHAs.

3. Participants are given the evaluated results of these two cases generated by

the evaluation method, in which these results are extracted, normalised and presented in a table and radar chart. Based on these resources, they answer the questions in part 2.

4. Participants answer the questions in part 3 based on the information obtained in this interview, in addition to the knowledge required before.

7.3.1 Participants

In this research, 18 participants are invited by sending emails and messages. These interviews are conducted based on the processes described above, in which participants are asked to answer the questions and complete the tasks. Participants who are invited to join these interviews should have basic IT background on knowledge of databases and modelling languages. In order to strike a balance and reduce bias, half of them (9) are male and half of them (9) are female. In addition, half of them are experts or professionals having several years industrial experience and took higher education (at least awarded bachelor degree) in this domain. The detailed distribution of these participants is summarised in table 7.4. As can be seen, participants only having IT background are 4 females and 5 males. Participants having professional experience in this domain are 5 females and 4 males. Around 88.89% of participants have industrial experience and all of them took higher education. They are randomly assigned to analyse and compare different case sets and each case set is assessed to three participants.

Table 7.4: The summary of participants' information

Participant	Gender	Expert/Professional	Industrial	Higher Education
1	Female	No	No	Yes
2	Female	No	Yes	Yes
3	Female	No	Yes	Yes
4	Female	No	Yes	Yes

participant	Gender	Expert/Professional	Industrial	Higher Education
5	Male	No	Yes	Yes
6	Male	No	Yes	Yes
7	Male	No	Yes	Yes
8	Male	No	Yes	Yes
9	Male	No	Yes	Yes
10	Female	Yes	Yes	Yes
11	Female	Yes	Yes	Yes
12	Female	Yes	No	Yes
13	Female	Yes	Yes	Yes
14	Female	Yes	Yes	Yes
15	Male	Yes	Yes	Yes
16	Male	Yes	Yes	Yes
17	Male	Yes	Yes	Yes
18	Male	Yes	Yes	Yes

7.3.2 Conducting Interviews

In order to clearly describe these interviews, an example is provided along with the questions and tasks listed below. The materials in a interview are attached in appendix A with a participant consent form, a question and task form, modelled DWHAs in case 2 and 3, evaluated results in a table and a radar chart. After participants read the form, they are asked to observe four modelled DWHAs. These four DWHAs include two cases presented by the modelling language (ArchiMate) with original elements and extra information (such as loading speed, query speed, schema, etc.). In terms of the question and task form, the first part has two close-ended questions and one open-ended question, which refers to the usefulness of the extra information and the extended component. Participants need to choose one answer from four options for each case and are requested to provide a brief

reason for their choices. In the second part, these two cases are evaluated by the evaluation method. Taking case 2 and 3 as examples, the results are tabulated in a big table and illustrated in a radar chart which are attached in appendix A in table A.1 and figure A.5. This radar chart have 10 indicators, in which 5 of them are evaluated results of combined macroscopical features in each layer, 5 of them are evaluated results of microscopical features but with normalised scope from 0 to 3. For example, in this chart, a indicator (Source & Format Ave.) is obtained by calculating the average of the evaluated results referring to the sub-features of the data source and data format. Participants are asked to compare which architecture they will choose based on these results and give a brief reason of their determination. In the third part, the questions are more open to broadly assess explicitness of the evaluated results and confidence of using this method.

1. **The first part**

    1.1 Please compare the first and second modelled architectures referring to page 1 and page 2. Do you think the additional information in orange boxes is useful to better understand/compare data warehouse architectures? [Options: Yes | Maybe | Unsure | No]

    1.2 Please compare the first and second modelled architectures referring to page 1 and page 2. Do you think the extended component (schema) in purple boxes is useful to better understand/compare data warehouse architectures? [Options: Yes | Maybe | Unsure | No]

    1.3 Please give a brief reason.

2. **The second part**

    2.1 Referring to the table and radar chart in page 3, which architecture do you choose? [Options: First Case | Second Case | Same | Unsure]

    2.2 Please give a brief reason.

3. **The third part**

   3.1 Do you think the evaluated results are explicit and easy to understand? [Options: Yes | Maybe | Unsure | No]

   3.2 How much confidence do you have to evaluate/compare data warehouse architectures using this method?

   3.3 Please give a brief reason.

7.3.3 Result Discussion

Participants are isolated to conduct interviews in order to avoid interactions between them. They normally take around 10 to 20 minutes to finish a interview. Finally, 18 forms are completed and collected for further analysis.

7.3.3.1 Results for the Understandability

The first part is to evaluate whether the modelled architectures along with measured results are understandable or not. The question 1.1 receives answers including 16 "Yes", 1 "Maybe" and 1 "No", none of the participants chooses the option "Unsure". Hence, 88.89% participants think the additional information is useful. Regarding the question 1.2, the responses receive 13 "Yes", 3 "Maybe", 1 "Unsure" and 1 "No" from participants. Around 72.22% of them believe that the extended component are definitely useful and 6.67% of them believe that it may be useful. Different open reasons and suggestions are received referring to the question 1.3. These answers are summarised and presented as follows, which may be helpful to improve this research in future work.

- The extra information provides details to efficiently understand describe DWHAs.

- The modelled DWHAs with extra information is useful for managers and developers to better understand DWHAs.

- The type of the schema information is useful to know the structure of how tables are organised in DWHs.

- The extra information can be applied to describe architectures and systems for other domains.

7.3.3.2 Results for the Validity of Evaluated Results

The second part evaluates the validity of the evaluated results through observing the results and diagrams to rank DWHAs, then comparing with initial results. The question 2.1 is responded by 16 Participants who make their estimations based on the evaluated results and choose a certain DWHA between two cases. 2 Participants choose unsure to determine which one is good. Hence, 88.89% participants can make choices based the evaluated results generated by this method. And approximately 11.11% of them are unsure to choose one from these two DWHAs based on the results. Before conducting these interviews, initial result referring to the ranking of these cases are determined based on the total scores calculated by adding all sub-features' score. The results generated by participants and initial results are contrasted and all of them are matched. For instance, when comparing the case 2 and 3, two participants choose case 2, one tester is unsure. Hence, the final result from participants is case 2, as it has more supporters than case 3. By observing the initial results, the score of case 2 is greater than the score of case 3. Hence, these two results are estimated to be consistent. By further analysing the compared results between internal cases (such as case 1 and 2) which are tested in the same data set and queries and external cases (such as case 1 and 6) which are tested with different data set and queries, there is no distinct difference between them. To be more specific, external cases can be comparable even when they are running in different environments. In terms of the open-ended question 2.2, different opinions and feedback are provided by participants and some of them are summarised below.

- When comparing DWHAs, it is easy to understand which architecture is better through the evaluated results.
- The radar chart is helpful to make decisions to choose a DWHA.
- The decision can be made by comparing each sub-features and counting which one has more sub-features having greater scores.
- A case is chosen based on a sub-feature, e.g. a case with the higher loading/query throughput.

7.3.3.3 Results for the Explicitness and Efficiency of the Evaluation Method

The third part refers to evaluate the explicitness and efficiency of this evaluation method. In terms of the results, the question 3.1 receives answers including 15 "Yes", 2 "Maybe" and 1 "No". Hence, 83% of participants believe the evaluated results are explicit and easy to understand. 11.11% of participants hesitate to make decisions. 5.56% of participants denote that the results are not easy to understand. In terms of the question 3.2, through gathering and calculating the rates of the confidence of using this method, the average rate of all participants is 79.17%. If considering their background, the average confidence rate of professionals is 82.78%, while the average rate of participants who only have the IT background is 75.00%. Some of them believe if more time or training is given, they can increase their confidence. It is insufficient for them to obtain details of this method only through this test (around 10 to 20 minutes). In terms of the open-ended question 3.3, diverse reasons and feedback are provided by participants, which are summarised below.

- This method offers a systematic way for people who are not experts in this domain to evaluate DWHAs.

- The results are easy to understand and the procedure of evaluating DWHAs is easy to follow.
- This method covers the measurements of metadata for DWHs and distributed systems which are important.
- This research could make a contribution referring to generate proposals for new DWH projects.
- This method can be extended to evaluate customised DWHAs with other features and sub-features in the taxonomy.

7.4 Conclusion

This chapter evaluates the proposed evaluation method with 6 cases built in multiple types of architectures and environments. This method evaluates DWHAs through three primary phases, firstly, a DWHA is modelled. During this time, some insensitive sub-features are identified and measured. After this, some other sub-features named sensitive features are measured. Finally, these results are input into the framework to evaluate this DWHA. The evaluated results are normalised and demonstrated in a table and a radar chart. The results generated by this method are evaluated in empirical research through interviews conducted with 18 participants. The results generated in these interviews are further analysed and summarised as follows. Around 88.89% and 72.22% participants denote the extra information and the extended component are useful to better understand/evaluate DWHAs. Approximately 88.89% of participants can make choices based the evaluated results generated by this method. All chosen results match the initial results generated by comparing the total scores of these cases. In terms of the explicitness, around 83.33% of participants agree the evaluated results are easy to understand. It means the evaluated results are clear without ambiguities. Participants have the average of 79.17% confidence to execute this method to evaluate

DWHAs, which prove this method is efficiently executed without much expert interpretation.

The third method of this methodology as the main part for evaluation is proved understandabe and valid in case studies and emprical research. As discussed in chapter 1, the contributions made by this methodology refer to both academia and industry. They show differences in these two sides when using this methodology regarding. In terms of the usage in academia, this methodology can be acted as a theory and provides three methods to identify typical DWHAs and features, then evaluate DWHAs. The methodology with these three methods can be extended to other domains. However, referring to usage in industry, it more focuses on the outputs of these three methods (9 typical DWHAs, DWHA feature-based taxonomy and the framework) and how to use this methodology to evaluate DWHAs. The details will be articulated in the following chapter.

8 Conclusions and Further Research Directions

This chapter is related to the conclusion phase in DSRM. It discusses the contributions made in this research and further directions. Three methods are generated as primary outputs. The first two methods are used to classify DWHAs and identify reliable features. Based on the results generated by these two methods, the method for DWHA evaluation is designed and developed as the artefact regarding DSRM. Furthermore, the limitations of this research are discussed to clarify the scope of this research and criticise the insufficiency of this research. In view of these limitations, research directions are analysed and articulated to enhance this research in the future.

8.1 Summary

This research is guided by the DSRM under six phases to systematically identify the problem, determine objectives, design, develop and evaluate the artefact (the evaluation method) propose in this research. Firstly, the background of DWHs and DWHAs is discussed to identify the problem which should be addressed in this research. Based on the background and problem, the motivation is clarified, then the challenges and objectives of this research are defined. Subsequently, the main research question and three relevant sub-research questions are put forward aligned with the problem and objectives. Through further investigating the current situa-

tion and observing related previous research, several LRs are conducted regarding DWHs, DWHAs, their features and methods for DWHA evaluation. After these fundamental investigations, two methods are created to clarify the scope of this research with 9 typical DWHAs and identify a set of reliable features which are frequently concerned in this domain. Based on the outputs of these two methods, the method for DWHA evaluation is designed and developed with these typical DWHAs, some reliable features and some sub-features. Finally, this evaluation method is assessed with 6 cases and empirical research.

By investigating insights into this methodology and its three method, they show some pros and cons referring to their work. For instance, the evaluation methodology works well with adequate DWHAs information, may not work well with inadequate DWHAs information or with out scope of these 9 typical DWHAs. The three methods also should be executed with sufficient data supported. The details and other aspects are articulated in following sections.

8.2 Main Research Contributions

Regarding contributions, this research is aligned with research questions. In the first three chapters, the MRQ and three SRQs are identified by investigating current situations and issues in this domain based on the literature. The DSRM is decided to guide this research in order to systematically address these questions. Subsequently, the following three chapters (from chapter 4 to chapter 6) investigate and answer these three sub-research questions by generating three methods. Chapter 7 evaluates the evaluation methodology and whether the objectives are achieved or not through case studies and empirical research. The main contributions and main outputs are summarised and listed from both academic and industrial sides.

8.2.1 Academic Contributions

For MRQ: It develops a methodology with three methods to explicitly and efficiently evaluate DWHAs by measuring reliable and critical features.

- This methodology can evaluate DWHAs with less expert support and the results can easily be understood, which is proved in 6 cases and empirical research.
- This methodology provides three methods which can be extended to evaluate other architectures.

For SRQ 1: It proposes a method to extend the knowledge of identifying typical DWHAs.

- New DWHAs (such as virtual and big DWHAs) are considered and added into the typical DWHAs and the classification model.
- The distribution of these DWHAs is produced by conducting statistic analysis based on collected data in a systematic LR.

For SRQ 2: It implements a method to contribute the knowledge for generating reliable features and relevant sub-features.

- It generates two matrices to provide frequent terms and relationships between these terms referring to DWHAs.
- 22 reliable features and 111 terms as relevant sub-features are identified based on these terms derived from the data collected via a systematic LR.

For SRQ 3: It develops a feature-based modelling and evaluation method supported by the two methods' results to generate theories for systematically benchmarking DWHAs.

- A framework is generated to organise selected reliable features, sub-features and their measurements to concretely evaluate DWHAs and grade results.
- A set of processes is designed and developed to systematically evaluate DWHAs by the measurements in the framework.

8.2.2 Industrial Contributions

For the proposed methodology: A methodology with three methods are proposed to provide systematic processes for DWHA investigation and evaluation.

- It offers an artefact to conduct evaluation DWHAs with less human resource and cost.
- The principle of this methodology can be as a reference to form evaluation methods in other domains.

For the DWHAs classification method: A DWHA classification model is proposed with four levels and the nine typical architectures based on the literature.

- The typical DWHAs and their "big picture" can facilitate companies to determine what types of DWHAs can be chosen and built.
- The trend of DWH systems to address big data issues is identified based on the distribution of these DWHAs. If companies need to establish DWHAs, the BDWA, PDWA and DDLA should be considered.

For the reliable features identification method: A taxonomy of DWHA features is produced with four levels for different purposes.

- The DWHA feature taxonomy can simplify the procedure of evaluating DWHAs with other reliable and critical features and sub-features in practice.

- The method and taxonomy can be leveraged to identify and interpret critical features from DWH projects which are being developed or have already operated in companies.

For the feature-based modelling and evaluation method: Modelled DWHAs, framework, tables and radar charts are designed to explicitly demonstrate the evaluated DWHAs.

- Through this method, companies can better understand their current DWHAs via modelling their architectures and evaluating these features.

- This method can assist companies to evaluate then choose suitable DWHAs, when they are in the design and development of DWH projects.

Therefore, these three methods answer these three sub-research questions and make contributions from both academic and industrial perspectives. In addition, the evaluation method should be tested whether it can be efficiently conducted with less experts' interpretation and be used to explicitly demonstrate results with interpretability and unambiguity. These two aspects are examined and proved in chapter 7 in which 6 DWHAs are evaluated by executing the evaluation method and empirical research is conducted by carrying out interviews. In these interviews, 18 participants are invited to answer some questions and tasks. Around 83.33% of participants agree the evaluated results are explicit and unambiguous. Participants have around 79.17% of confidence to execute this method to evaluate DWHAs. The average of the confidence rate from professionals is 82.78%, while the average rate from testers who have IT background is 75.00% after these interviews. There is no big gap (7.78 percent) or contradiction of confidence to conduct this method between professionals and normal IT participants. Therefore, the efficiency and explicitness of this method are proved through these interviews.

8.3 Limitations and Challenges

Due to limited time and resource, this research study shows some limitations and challenges which should be discussed in order to better use this method and identify work being conducted in the future. These limitations and Challenges can affect the scope to narrow the executions of this method and attenuate the value of the results. The details are presented below.

8.3.1 Limitations referring to SRQs

For MRQ: The limitations refer to the contributions and outputs.

- 6 cases are involved in the evaluation and 18 participants are invited to conduct interviews in the empirical research to assess whether the objectives are achieved or not. This small group of cases and people may be not adequate to conduct this assessment.
- Some features and sub-features in the evaluation framework may be not suitable or important for evaluating some DWHAs, especially some DWHAs have special architectures, slef-defined components or customised requirements.

For SRQ 1: The limitations refer to the contributions and outputs.

- The data collected by the systematic LR may not cover all DWHAs and all relevant information, which may affect the results of the typical DWHAs.
- Some other DWHAs are described in the literature or used in practice. However, the evaluation method does not cover these DWHAs and focuses on the 9 typical DWHAs, which limits the scope of the evaluation method.

For SRQ 2: The limitations refer to the contributions and outputs.

- This research does not exhaust to search and collect all DWHA feature related data. Some features may be important referring to potentially reliable features.

- Although these features and sub-features are reliable and critical in the taxonomy, but it does not prove if they are adequate as potential features to evaluate DWHAs. Maybe more features and sub-features should be involved.

For SRQ 3: The limitations refer to the contributions and outputs.

- This evaluation method only measures 10 reliable features and 15 sub-features derived from the taxonomy to constitute the evaluation framework. The rest of them can be significant to evaluate DWHAs.

- More information should be provided regarding the framework on how to choose these features and sub-features from the taxonomy.

8.3.2 Methodological Challenges

This evaluation method is developed based on the typical DWHAs and reliable features identified by the two methods proposed in chapter 4 and 5. These methods are designed to be applied in other domains, but it shows some challenges to prove their generalisation with adequate cases. In addition, the evaluation method proposed in chapter 6 should be executed under different types of cases. The detailed challenges are listed below.

- The architecture classification method proposed in chapter 4 can be used for other architecture classifications, but it is a challenge to provide adequate evidence to prove its validity.

- The method in chapter 5 is invoked to identify reliable features and generate the taxonomy in relation to the domain of DWHAs, which is designed to identify reliable features and the taxonomy in other domains, but it is time-consuming to prove this extension.

- The evaluation method is conducted under 6 cases. It is a challenge to conduct it with adequately cases. Because it is not easy to find suitable cases with detailed information for evaluation.

8.3.3 Domain Specific Challenges

DWHA evaluation belongs to the sub-domain of IS research, but this domain has specific characteristics which challenge the progress and results of this research. Some of them are articulated and listed below.

- DWHAs or DWH projects normally have enormous components and relationships between them, which are complicated to model their architectures and evaluate their features.

- DWHs extract data from operational systems or other sources and they normally contains decades of data, so the volume of data is large to challenge some features' evaluation (e.g. throughput).

- The data in DWHs general contains valuable even sensitive information, so it is difficult to be allowed accessing these system and conducting evaluation.

8.3.4 Others

There are some other challenges which affect the progress and results of this research. According to the DSRM, after the evaluation method is designed and

developed, it should be evaluated and 6 case studies are used in the present research. It is a challenge to find more cases from different environments and the details are explained below.

- Limited DWHAs with detailed information can be acquired from open sources. There are a vast number of DWHAs in the literature, but almost all of them provide inadequate information for evaluation.

- It is time-consuming and requires hardware resource to build DWHAs, so the DWHAs built in research are normally uncomplicated and manageable to maintain DWHs and other components with limited data and simple structures.

- It is expensive and resource-consuming to implement and maintain a DWH system. Thus, it is not easy to find organisations which have DWHAs as cases to evaluate this method.

8.4 Future Work and Research Directions

The limitations and challenges are discussed above. In order to improve them, some supplementary work will be considered and conducted in the future work. Although the main research question and three sub-research questions are solved, some extra jobs should be continued to enhance the contributions and outputs of this research, especially these three methods. In addition, this research should be continued to further develop and assess for resolving these limitations and challenges. The future work and research directions are summarised and demonstrated in the list below.

For MRQ: The future work refers to the methodology with the three methods.

- The methodology generated in this research be as the theory to will be strengthened and further guide related work regarding architectural evaluation.

- The processes of methodology on how to create the framework as the core part will be organised together for developing frameworks in other domains in a short term.

- The three methods in this methodology can also be separately used for different purposes and will be extended to theories.

For SRQ 1: The future work refers to the DWHAs classification method.

- More DWHAs will be collected from other data sources for reinforcing the identification of typical DWHAs.

- More investigations will be conducted to prove if other DWHAs are typical and should be considered other than these 9 typical DWHAs.

- The architecture classification method will be tested in other domains (such as software system and enterprise architecture).

For SRQ 2: The future work refers to the reliable features identification method.

- More data referring to DWHA features will be collected to further investigate and identify reliable features.

- In order to further strengthen this method, more reliable features and sub-features will be considered or involved to extend the taxonomy.

- The method is designed to identify reliable features and generate the taxonomy in other domains. Extra work will be done to prove its generalisation and validation when executing it.

For SRQ 3: The future work refers to the evaluation method.

- This method will be executed with more cases to further examine its capabilities of DWHA evaluation.

- The results generated by this method will be assessed by more participants with different backgrounds.

- This method will be enhanced to customise its framework by adding or reducing features and sub-features for special DWHAs and purposes.

References

[1] Abdullah, A., Brobst, S., Umer, M., and Khan, M. (2004). The case for an agri data warehouse: Enabling analytical exploration of integrated agricultural data. In *Databases and Applications (pp. 139–144)*.

[2] Abelló, A., Romero, O., Pedersen, T. B., Berlanga, R., Nebot, V., Aramburu, M. J., and Simitsis, A. (2015). Using semantic web technologies for exploratory olap: a survey. *IEEE transactions on knowledge and data engineering*, 27(2):571–588.

[3] Aggarwal, C. and Reddy, C. (2013). *Data Clustering: Algorithms and Applications (Chapman & Hall/CRC Data Mining and Knowledge Discovery Series)*. Chapman and Hall/CRC; 1 edition.

[4] Aier, S., Riege, C., and Winter, R. (2008). Classification of enterprise architecture scenarios - an exploratory analysis. In *Enterprise Modelling and Information Systems Architectures, 3(1), pp.14–23*.

[5] Aldiabat, K. and Navenec, L. (2011). Philosophical roots of classical grounded theory: Its foundations in symbolic interactionism. In *The Qualitative Report, 16(4), pp.1063–1080*.

[6] Alibaba (2018, October 10). Pangu. *The High Performance Distributed File System by Alibaba Cloud.* Retrieved from

https://www.alibabacloud.com/blog/pangu-the-high-performance-distributed-file-system-by-alibaba-cloud_594059.

[7] Alibaba (2019, April 26). Alibaba cloud analyticdb. *TPC-DS Advanced Sort Results List*. Retrieved from http://www.tpc.org/tpcds/results/tpcds_advanced_sort.asp.

[8] Aliyun (2019, August 21). Alibaba cloud analyticdb. *Aliyun*. Retrieved from https://cn.aliyun.com/product/list.

[9] Allan, G. (2003). A critique of using grounded theory as a research method. In *Electronic Journal of Business Research Methods, vol. 2, no.1 pp. 1–10*.

[10] Almalawi, A., Fahad, A., Tari, Z., Cheema, M., and Khalil, I. (2016). Knnvwc: An efficient k-nearest neighbors approach based on various-widths clustering. In *IEEE Transactions on Knowledge and Data Engineering, 28(1), pp.68–81*.

[11] Alsqour, M., K, M., and Owoc, M. L. (2012). A survey of data warehouse architectures: preliminary results. In *InFedCSIS (pp.1121–1126)*.

[12] AnalyticDB (2019, April 26). Alibaba cloud analyticdb. *TPC-DS Advanced Sort Results List*. Retrieved from http://www.tpc.org/tpcds/results/tpcds_result_detail.asp?id=119042601.

[13] AnalyticDB (2019, August 21). Analyticdb - a cloud based olap data analytics service that enables big data processing to gain actionable insights. *Aliyun*. Retrieved from https://www.alibabacloud.com/product/analyticdb?spm=a2c63.p38356.879954.30.13deccffNbFQy0.

[14] Anand, R. V. and Dinakaran, M. (2018). Improved scrum method through staging priority and cyclomatic complexity to enhance software process and

quality. *International Journal of Internet Technology and Secured Transactions*, 8(2):150–166.

[15] ANSI (1983). Ieee std 829-1983, standard for software test documentation. In *Institute of Electrical and Electronics Engineers, New York.*

[16] Anu, V., Hu, W., Carver, J. C., Walia, G. S., and Bradshaw, G. (2018). Development of a human error taxonomy for software requirements: a systematic literature review. *Information and Software Technology*, 103:112–124.

[17] ArchiMate (2019, August 21). Archi open source ArchiMate modelling. *Archimatetool.* Retrieved from Available at: https://www.archimatetool.com/.

[18] Ariyachandra, T. and Watson, H. (2006). Which data warehouse architecture is most successful? In *Business intelligence journal, 11(1), p.4*.

[19] Ariyachandra, T. and Watson, H. (2008). Which data warehouse architecture is the best? In *Communications of the ACM, 51(10), p.146*.

[20] Ariyachandra, T. and Watson, H. (2010). Key organizational factors in data warehouse architecture selection. In *Decision support systems. 31;49(2):200–12*.

[21] Arnott, D. (2008). Success factors for data warehouse and business intelligence systems. In *ACIS 2008 Proceedings, p.16*.

[22] Arnrich, B., Walter, J. A., Albert, A., Ennker, J., and Ritter, H. J. (2004). Data mart based research in heart surgery: challenges and benefit. In *Medinfo*, pages 8–12.

[23] Astillo, V., Martnez-Garca, A., and Pulido, J. (2010). A knowledge-based taxonomy of critical factors for adopting electronic health record systems by physicians: a systematic literature review. *BMC medical informatics and decision making*, 10(1):60.

[24] Ayman, A., Zaiane Osmar, R., and Ji, Y. (2001). Towards framework for the virtual data warehouse. In *British National conference on databases (pp. 202–218)*.

[25] Baars, H. and Kemper, H. G. (2008). Management support with structured and unstructured dataan integrated business intelligence framework. *Information Systems Management*, 25(2):132–148.

[26] Bailey, K. (1994). Numerical taxonomy and cluster analysis. In *Typologies and Taxonomies. p. 34. ISBN 9780803952591*.

[27] Berkhin, P. (2006). A survey of clustering data mining techniques. In *Grouping multidimensional data (pp. 25–71). Springer, Berlin, Heidelberg*.

[28] Bernardino, J. and Madeira, H. (2002). Data warehouse striping: Improved query response time. *Enterprise Information Systems III*, 3:75.

[29] Bhatnagar, S., Singh, P., Yadav, S., and Jahan, R. (2018). Analytical processing in data warehouse: A review. *International Journal of Advanced Research in Computer Science*, 9(Special Issue 2):49.

[30] Bhawna1 and Gobind (2015). Research methodology and approaches. In *IOSR Journal of Research & Method in Education (IOSR-JRME) e-ISSN: 23207388,p-ISSN: 2320737X Volume 5, Issue 3 Ver. IV (May - Jun. 2015), PP 48-51*.

[31] Bichler, M. (2006). Design science in information systems research. In *Wirtschaftsinformatik, 48(2), pp.133–135*.

[32] Blumberg, R. and Atre, S. (2003). The problem with unstructured data. *Dm Review*, 13(42–49):62.

[33] Bontempo, C. and Zagelow, G. (1998). The IBM data warehouse architecture. In *Communications of the ACM, 1;41(9):38–48*. ACM.

[34] Brandtner, P. (2017). *Design and evaluation of a process model for the early stages of product innovation*. Doctoral dissertation, Dublin City University.

[35] Brannen, J. (2017). *Mixing methods: Qualitative and quantitative research*. Routledge.

[36] Breslin, M. (2004). Data warehousing battle of the giants. *Business Intelligence Journal*, 7:6–20.

[37] Brown, M. (2004). 8 characteristics of a successful data warehouse. In *Proceedings of the Twenty-Ninth Annual SAS Users Group International Conference*.

[38] Buneman, P. (1997). Semistructured data. In *Proceedings of the sixteenth ACM SIGACT-SIGMOD-SIGART symposium on Principles of database systems*, pages 117–121. ACM.

[39] Candal-Vicente, I. (2009). Factors that affect the successful implementation of a data warehouse. In *2009 Fourth International Multi-Conference on Computing in the Global Information Technology (pp.1–6). IEEE.*

[40] Carino Jr, F., Kostamaa, P., Kaufmann, A., and Burgess, J. (2001). Storhouse metanoia-new applications for database, storage & data warehousing. In *ACM SIGMOD Record*, volume 30, pages 521–531. ACM.

[41] Chaki, N. and Sarkar, B. (2010). Virtual data warehouse modeling using petri nets for distributed decision making. In *Journal of Convergence Information Technology, 5(5), pp.8–21*.

[42] Chaudhuri, S. and Dayal, U. (1997). An overview of data warehousing and olap technology. *ACM Sigmod record*, 26(1):65–74.

[43] Chen, H., Chiang, R., and Storey, V. Business intelligence and analytics: From big data to big impact. In *MIS quarterly. 2012 Dec 1;36(4):1165-88*.

[44] Chen, W., Andrus, S., Balaji, B., Cialini, E., Kwok, M., Melnyk, R., and Rockwood, J. (2014). Leveraging db2 10 for high performance of your data warehouse. In *Integrations of Data Warehousing, Data Mining and Database Technologies: Innovative Approaches (pp. 106-147)*. IBM Redbooks.

[45] Choudhary, R. (2012). Key organizational factors in data warehouse architecture selection. In *Vivekananda Journal of Research, 1(1), pp.24–32.*

[46] Chowdhury, R. and Pal, B. (2010). Proposed hybrid data warehouse architecture based on data model. In *International Journal of Computer Science and Communication, 1(2), pp.211–213*.

[47] Cisco (2018, March 5). Cisco ucs integrated infrastructure for big data. *TPC-DS Advanced Sort Results List*. Retrieved from http://www.tpc.org/tpcds/results/tpcds_result_detail.asp?id=118030501.

[48] CITO (2014). Putting the data lake to work a guide to best practices. In *CITO Research Advancing the craft of technology leadership*.

[49] Cook, T. D. (2015). Quasi-experimental design. *Wiley Encyclopedia of Management*, pages 1–2.

[50] Creswell, J. (2008). Educational research: Planning, conducting, and evaluating quantitative and qualitative research (3rd ed.).

[51] Creswell, J. (2009). *Research Design: Qualitative, Quantitative and Mixed Method Approaches (3rd ed.)*. Los Angeles: SAGE Publications.

[52] Cronin, P., Ryan, F., and Coughlan, M. (2008). Undertaking a literature review: a step-by-step approach. In *British journal of nursing, 17(1), pp.38–43*.

[53] Darmont, J., Bentayeb, F., and Boussaïd, O. (2005). Dweb: A data warehouse engineering benchmark. In *International Conference on Data Warehousing and Knowledge Discovery*, pages 85–94. Springer.

[54] Dedi, N. and Stanier, C. (2016). An evaluation of the challenges of multilingualism in data warehouse development. In *Proceedings of the 18th International Conference on Enterprise Information Systems. 1. SciTePress. pp. 196206.*

[55] Demchenko, Y., Grosso, P., De Laat, C., and Membrey, P. (2013). Addressing big data issues in scientific data infrastructure. In *2013 International Conference on Collaboration Technologies and Systems (CTS)*, pages 48–55. IEEE.

[56] DeMeo, F. E., Binzel, R. P., Slivan, S. M., and Bus, S. J. (2009). An extension of the bus asteroid taxonomy into the near-infrared. *Icarus*, 202(1):160–180.

[57] Denscombe, M. (2014). *The good research guide: for small-scale social research projects*. McGraw-Hill Education (UK).

[58] Devlin, B. and Cote, L. (1996). Data warehouse: from architecture to implementation. In *Addison-Wesley Longman Publishing Co., Inc.*

[59] Di Tria, F., Lefons, E., and Tangorra, F. (2017). Evaluation of data warehouse design methodologies in the context of big data. In *International Conference on Big Data Analytics and Knowledge Discovery (pp. 3–18). Springer, Cham*.

[60] Dukaric, R. and Juric, M. (2013). Towards a unified taxonomy and architecture of cloud frameworks. In *Future Generation Computer Systems, 29(5), pp.1196–1210*.

[61] Dutt, A., Aghabozrgi, S., Ismail, M., and Mahroeian, H. (2015). Clustering algorithms applied in educational data mining. In *International Journal of Information and Electronics Engineering, 5(2), p.112*.

[62] El Moukhi, N., El Azami, I., and Mouloudi, A. (2015). Data warehouse state of the art and future challenges. In *2015 International Conference on Cloud Technologies and Applications (CloudTech)*, pages 1–6. IEEE.

[63] Ernest, T. S. (2014). *Action research*. Thousand Oaks, California : SAGE.

[64] Evans, R., Lloyd, J., and Pierce, L. (2012). Clinical use of an enterprise data warehouse. In *AMIA Annual Symposium Proceedings (Vol. 2012, p. 189). American Medical Informatics Association*.

[65] Faggiolani, C. (2011). Perceived identity: applying grounded theory in libraries. In *University of Florence. JLIS.it. . 2 (1). doi:10.4403/jlis.it-4592*.

[66] Finkelstein, M. and Whitley, R. (1978). Fibonacci numbers in coin tossing sequences. *Journal of Fibonacci Quarterly*, pages 16(6), ISSN 0015–0517.

[67] Flume (2019, January 8). Flume. *Apache*. Retrieved from https://flume.apache.org/.

[68] Geeksforgeeks (2019, August 21a). Difference between structured, semi-structured and unstructured data. *A computer science portal for geeks*. Retrieved from https://www.geeksforgeeks.org/difference-between-structured-semi-structured-and-unstructured-data/.

[69] Geeksforgeeks (2019, August 21b). What is unstructured data. *A computer science portal for geeks*. In *Retrieved from https://www.geeksforgeeks.org/what-is-unstructured-data/*.

[70] Glaser, B. (1978). Theoretical sensitivity: Advances in the methodology of grounded theory. In *Mill Valley, CA: Sociology Press*.

[71] Gleicher, I., Ren, Y. J., and John, C. P. S. (2017). Method and system for defining an extension taxonomy. Google Patents. US Patent 9,811,604.

[72] Goar, V. (2011). The main feature for successful implementation a data warehouse. In *Vishal Kumar Goar et al. Indian Journal of Computer Science and Engineering (IJCSE)*.

[73] Golfarelli, M. and Rizzi, S. (2018). From star schemas to big data: 20+ years of data warehouse research. In *A Comprehensive Guide Through the Italian Database Research Over the Last 25 Years (pp. 93-107). Springer, Cham*.

[74] Goulding, C. (2005). Grounded theory, ethnography and phenomenology: A comparative analysis of three qualitative strategies for marketing research. *European journal of Marketing*, 39(3/4):294–308.

[75] Grant, M. J. and Booth, A. (2009). A typology of reviews: an analysis of 14 review types and associated methodologies. *Health Information & Libraries Journal*, 26(2):91–108.

[76] Graves, B. A. (2008). Integrative literature review: a review of literature related to geographical information systems, healthcare access, and health outcomes. *Perspectives in Health Information Management/AHIMA, American Health Information Management Association*, 5.

[77] Greene, J. (2007). *Mixed methods in social inquiry (Vol. 9)*. John Wiley & Sons.

[78] Gribbons, B. (1997). True and quasi-experimental designs. *Practical Assessment, Research & Evaluation*. Volume 5. ISSN:1531-7714.

[79] Hai, R., Geisler, S., and Quix, C. (2016). Constance: An intelligent data lake system. In *Proceedings of the 2016 International Conference on Management of Data*, pages 2097–2100. ACM.

[80] Hart, C. (2018). *Doing a literature review: Releasing the research imagination*. Sage.

[81] Harwell, M. (2011). Research design: Qualitative, quantitative, and mixed methods: Pursuing ideas as the keystone of exemplary inquire. In *C. Conrad, & R. C. Serlin (Eds.), The Sage handbook for research in education: Pursuing ideas as the keystone of exemplary inquir (Second Edition ed.). Thousand Oaks, CA: Sage*.

[82] Hassanpour, S., Tomita, N., DeLise, T., Crosier, B., and Marsch, L. A. (2019). Identifying substance use risk based on deep neural networks and instagram social media data. *Neuropsychopharmacology*, 44(3):487.

[83] Hatami-Marbini, A., Emrouznejad, A., and Tavana, M. (2011). A taxonomy and review of the fuzzy data envelopment analysis literature: two decades in the making. In *European journal of operational research*, volume 214, pages 457–472. Elsevier.

[84] HBase (2019, June 10). Hbase. *Apache*. Retrieved from https://hbase.apache.org/.

[85] Helfert, M., Donnellan, B., and Ostrowski, L. (2012). The case for design science utility and quality - evaluation of design science artifact within the sustainable ict capability maturity framework. In *An International Journal on Information Technology, Action, Communication and Workpractices 6 (1), pp. 46–66*.

[86] Henigman, T. and Demesa, J. (2010). Data mart generation and use in association with an operations intelligence platform. In *U.S. Patent 7,840,607*.

[87] Hevner, A. (2007). A three cycle view of design science research. In *Scandinavian journal of information systems, 19(2), p.4*.

[88] Hevner, A. and Chatterjee, S. (2010). *Design research in information systems: theory and practice*. Springer Science & Business Media.

[89] Hevner, A., March, S., Park, J., and Ram, S. (2008). Design science in information systems research. In *Management Information Systems Quarterly, 28(1), p.6*.

[90] Hevner, A. R., March, S. T., Park, J., and Ram, S. (2004). Design science in information systems research. In *MIS Quarterly, 75–106*.

[91] Hive (2019, August 21). Hive. *Apache*. Retrieved from https://hive.apache.org/.

[92] Hoidn, P. and Schwidder, K. (2014). Enterprise IT architectures IT architecture standards, togaf and omg in more detail, key architecture work products. *University of Zurich, Zurich, Switzerland*.

[93] Holmes, A. (2012). *Hadoop in practice*. Manning Publications Co.

[94] Hox, J. and Boeije, H. R. (2005). Data collection, primary versus secondary. In *Encyclopedia of Social Measurement, Volume 1. Elsevier Inc. All Right Reserved*.

[95] Hsieh, L. and Tsai, L. (2006). The optimum design of a warehouse system on order picking efficiency. *The International Journal of Advanced Manufacturing Technology*, 28(5-6):626–637.

[96] Hwang, H., Kua, C., Yen, D., and Cheng, C. (2004). Critical factors influencing the adoption of data warehouse technology: a study of the banking industry in taiwan. In *Decis. Support Syst. 37, pp.1*.

[97] IBM (1993). Information warehouse architecture I. *IBM Corporation*.

[98] Iivari, J. and Venable, J. (2009). Action research and design science research seemingly similar but decisively dissimilar. In *17th European Conference on Information Systems*.

[99] Inmon, B. (1997). What is a data warehouse? In *Prism Tech. Topic 1(1)*.

[100] Inmon, B. (2006). Dw 2.0; architecture for the next generation of data warehousing. *Information Management*, 16(4):8.

[101] Inmon, B. (2016). *Data Lake Architecture: Designing the Data Lake and avoiding the garbage dump*. Technics publications.

[102] Jalaja, T. and Shailaja, M. (2015). A comparative study on operational database, data warehouse and hadoop file system. In *International Journal of Engineering and Techniques - Volume I Issue 5*.

[103] Järvinen, P. (2007). Action research is similar to design science. *Quality & Quantity*, 41(1):37–54.

[104] Jindal, R. and Acharya, A. (2004). Federated data warehouse architecture. *Wipro Technologies white paper*.

[105] Johnson, B. and Turner, L. A. (2003). Data collection strategies in mixed methods research. *Handbook of mixed methods in social and behavioral research*, pages 297–319.

[106] Johnson, R. and Onwuegbuzie, A. (2004). Mixed methods research: A research paradigm whose time has come. In *Educational researcher, 33(7), pp.14–26*.

[107] Johnston, A. (2014). Rigour in research: theory in the research approach. In *European Business Review 26 (3), pp. 206–217*.

[108] Jörg, T. and Deßloch, S. (2008). Towards generating etl processes for incremental loading. In *Proceedings of the 2008 international symposium on Database engineering & applications*, pages 101–110. ACM.

[109] Juristo, N. and Moreno, A. M. (2013). *Basics of software engineering experimentation*. Springer Science & Business Media.

[110] Kanchi, S. (2009). Selecting the right architectures for successful data warehouses. *Tata Consultancy Services*.

[111] Kaplan, A. (2017). *The conduct of inquiry: Methodology for Behavioural Science*. Routledge, eBook ISBN 9781351484510.

[112] Karie, N. M. and Venter, H. S. (2015). Taxonomy of challenges for digital forensics. *Journal of forensic sciences*, 60(4):885–893.

[113] Katal, A., Wazid, M., and Goudar, R. (2013). Big data: issues, challenges, tools and good practices. In *Contemporary Computing (IC3), 2013 Sixth International Conference on (pp. 404–409). IEEE*.

[114] Kessler, R. and Stafford, D. (2008). *Collaborative medicine case studies: Evidence in practice*. Springer Science & Business Media.

[115] Khelil, N. (2016). The many faces of entrepreneurial failure: Insights from an empirical taxonomy. *Journal of Business Venturing*, 31(1):72–94.

[116] Kimball, R. and Caserta, J. (2011). *The data warehouse ETL toolkit: practical techniques for extracting, cleaning, conforming, and delivering data*. John Wiley & Sons.

[117] Kimball, R. and Ross, M. (2005). *Building the data warehouse*. John wiley & Sons.

[118] Kimball, R. and Ross, M. (2011). *The data warehouse toolkit: the complete guide to dimensional modeling*. John Wiley & Sons.

[119] Kimball, R. and Ross, M. (2013). *The data warehouse toolkit: The definitive guide to dimensional modeling*. John Wiley & Sons.

[120] Klein, J. D. (2002). Empirical research on performance improvement. *Performance Improvement Quarterly*, 15(1):99–110.

[121] Knox, S. W. (2018). *Machine learning: a concise introduction*, volume 285. John Wiley & Sons.

[122] Kothari, C. R. (2004). *Research methodology methods and techniques*. New Age International (P) Limited.

[123] Kumar, A., Dabas, V., and Hooda, P. (2018). Text classification algorithms for mining unstructured data: a swot analysis. *International Journal of Information Technology*, pages 1–11.

[124] Kumar, R. (2019). *Research methodology: A step-by-step guide for beginners*. Sage Publications Limited.

[125] Kwon, O., Lee, N., and Shin, B. (2014). Data quality management, data usage experience and acquisition intention of big data analytics. *International journal of information management*, 34(3):387–394.

[126] Laney, D. Warehouse factors to address for success. In *HP Professional 14 (5) (2000)*.

[127] Lankhorst, M., Proper, H., and Jonkers, H. (2009). The architecture of the ArchiMate language. In *Enterprise, Business-Process and Information Systems Modeling (pp. 367–380). Springer, Berlin, Heidelberg*.

[128] Lankhorst, M. and van Drunen, H. (2007). Enterprise architecture development and modelling–combining togaf and archimate. *Via Nova Architectura*, 21.

[129] Lankhorst, M. M., Aldea, A., and Niehof, J. (2017). Combining archimate with other standards and approaches. In *Enterprise Architecture at Work*, pages 123–140. Springer.

[130] Lantow, B., Jugel, D., Wiotzki, M., Lehmann, B., Zimmermann, O., and Sandkuhl, K. (2016). Towards a classification framework for approaches to enterprise architecture analysis. In *IFIP Working Conference on The Practice of Enterprise Modeling (pp. 335–343). Springer, Cham*.

[131] Leedy, P. and Ormrod, J. (2010). Practical research: Planning and design. In *9th edn. Pearson Educational International, Boston*.

[132] Levene, M. and Loizou, G. (2003). Why is the snowflake schema a good data warehouse design? *Information Systems*, 28(3):225–240.

[133] Levitt, H. M., Motulsky, S. L., Wertz, F. J., Morrow, S. L., and Ponterotto, J. G. (2017). Recommendations for designing and reviewing qualitative research in psychology: Promoting methodological integrity. In *Qualitative Psychology, 4(1), 222*.

[134] Levy, Y. and Ellis, T. (2006). Towards a framework of literature review process in support of information systems research. In *Proceedings of the 2006 Informing Science and IT Education Joint Conference (Vol. 26)*.

[135] Li, G., Ooi, B. C., Feng, J., Wang, J., and Zhou, L. (2008). Ease: an effective 3-in-1 keyword search method for unstructured, semi-structured and structured data. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 903–914. ACM.

[136] Lin, H., Hsu, P., and Sheen, G. (2007). A fuzzy-based decision-making procedure for data warehouse system selection. In *Expert systems with applications, 32(3), pp.939–953*.

[137] List, B., Bruckner, R., Machaczek, K., and Schiefer, J. (2002). A comparison of data warehouse development methodologies case study of the process warehouse. In *International Conference on Database and Expert Systems Applications (pp.203–215)*. Springer, Berlin, Heidelberg.

[138] Liu, H., Gong, X., Liao, L., and Li, B. (2018). Evaluate how cyclomatic complexity changes in the context of software evolution. In *2018 IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC)*, volume 2, pages 756–761. IEEE.

[139] Machchhar, S. and Kosta, Y. (2017). Approaches to gene expression classification using knowledge discovery techniques: A taxonomy and systematic literature review. In *2017 International Conference on Intelligent Sustainable Systems (ICISS)*, pages 922–926. IEEE.

[140] Majchrzak, T. A., Jansen, T., and Kuchen, H. (2011). Efficiency evaluation of open source etl tools. In *Proceedings of the 2011 ACM Symposium on Applied Computing*, pages 287–294. ACM.

[141] Mani, N. (2018). *A configuration-based domain-specific rule generation framework for process model customization*. Doctoral dissertation, Dublin City University.

[142] Mannino, M., Hong, S., and Choi, I. (2008). Efficiency evaluation of data warehouse operations. In *Decision Support Systems, 44(4), pp.883-898*.

[143] March, S. and Smith, G. (1995). Design and natural science research on information technology. In *Decision support systems, 15(4), pp.251–266*.

[144] March, S. T. and Hevner, A. R. (2007). Integrated decision support systems: A data warehousing perspective. In *Decision support systems, 43 (3), pp. 1031–1043. Elsevier*.

[145] Massad, N. and Beachboard, J. (2008). A taxonomy of service failures in electronic retailing. In *Proceedings of the 41st Annual Hawaii International Conference on System Sciences (HICSS 2008)*, page 285. IEEE.

[146] Mathirajan, M. and Sivakumar, A. (2006). A literature review, classification and simple meta-analysis on scheduling of batch processors in semiconductor. In *The International Journal of Advanced Manufacturing Technology, 29(9-10), pp.990–1001*.

[147] Matouk, K. and Owoc, M. (2012). A survey of data warehouse architectures preliminary results. In *Computer Science and Information Systems (FedCSIS), 2012 Federated Conference on (pp.1121–1126). IEEE.*

[148] Mcknight, D. and Chervany, N. (2001). What trust means in e-commerce customer relationships: an interdisciplinary conceptual typology, 6(2), 3559. *International Journal of Electronic Commerce*.

[149] Mehta, A. (2017). *A comparative study on corporate governance disclosure practices of GMM pfaudler limited and elecon engineering company limited, vitthal udyognagar*. RED'SHINE Publication. Pvt. Ltd.

[150] Menon, S. (2005). Allocating fragments in distributed databases. In *IEEE transactions on parallel and distributed systems, 16(7), pp.577–585*.

[151] Merriam (2019, August 21). Typology. *Merriam Webster Dictionary.* Retrieved from https://www.merriam-webster.com/dictionary/typology#learn-more.

[152] Merriam, S. B. (1988). *Case study research in education: A qualitative approach.* Jossey-Bass.

[153] Microsoft (2018, October 10). Create view (transact-sql). *Selected Version SQL Server 2017*. Retrieved from https://docs.microsoft.com/en-us/sql/t-sql/statements/create-view-transact-sql?view=sql-server-2017.

[154] Miele, R. and Mola, F. (2005). An open source based data warehouse architecture to support decision making in the tourism sector. *No. 142.2005, Fondazione Eni Enrico Mattei (FEEM), Milano*.

[155] Miloslavskaya, N. and Tolstoy, A. (2016). Big data, fast data and data lake concepts. *Procedia Computer Science*, 88:300–305.

[156] Mishra, D., Yazici, A., and Basaran, B. P. (2008). A case study of data models in data warehousing. In *2008 First International Conference on the Applications of Digital Information and Web Technologies (ICADIWT)*, pages 314–319. IEEE.

[157] Mohajan, H. (2018). Qualitative research methodology in social sciences and related subjects. In *Journal of Economic Development, Environment and People, Vol-7, Issue 01, 2018, pp.23–48*.

[158] Moher, D., Shamseer, L., Clarke, M., Ghersi, D., Liberati, A., Petticrew, M., Shekelle, P., and Stewart, L. (2015). Preferred reporting items for systematic review and meta-analysis protocols (prisma-p) 2015 statement. In *Systematic reviews, 4(1), p.1*.

[159] Mohsen and Hassan, M. (2002). A framework for the design of warehouse layout. *Facilities*, 20(13/14):432–440.

[160] Moody, D. L. and Kortink, M. A. (2000). From enterprise models to dimensional models: a methodology for data warehouse and data mart design. In *DMDW*, page 5.

[161] Muijs, D. (2010). *Doing quantitative research in education with SPSS*. Taylor & Francis. New York and London.

[162] MySQL (2019, August 21). Using views. *MySQL Server*. Retrieved from https://dev.mysql.com/doc/refman/5.7/en/views.html.

[163] Nambiar, R. and Poess, M. (2009). *Performance evaluation and benchmarking*. Springer.

[164] Nambiar, R. and Poess, M. (2017). Industry standards for the analytics era: Tpc roadmap. In *Technology Conference on Performance Evaluation and Benchmarking*, pages 1–8. Springer.

[165] Nathal, A. and Jos, D. (2015). An integrated strategy based in processes, requirements, measurement and evaluation for the formalization of necessities in data warehouse projects. In *International Workshop on Data Mining with Industrial Applications (DMIA) (pp. 8–17). IEEE*.

[166] Nickerson, R., Varshney, U., and Muntermann, J. (2013). A method for taxonomy development and its application in information systems. *European Journal of Information Systems, 22(3), pp.336-359*.

[167] Niederman, F. and March, S. T. (2012). Design science and the accumulation of knowledge in the information systems discipline. In *ACM Transactions on Management Information Systems. 3 (1), p. 15*.

[168] NISO (2019, August 21). Definitions. *Metadata types and functions*. Retrieved from http://marciazeng.slis.kent.edu/metadatabasics/types.htm.

[169] Offermann, P., Levina, O., Schonherr, M., and Bub, U. (2009). Outline of a design science research process. In *Proceedings of the 4th International Conference on Design Science Research in Information Systems and Technology (p. 7). ACM*.

[170] OLAP (2019, August 21). Easy olap definition. *What is the definition of OLAP?* Retrieved from http://olap.com/olap-definition/.

[171] Opengroup (2019, August 21). ArchiMate certification. *The open Group.* Retrieved from https://www.opengroup.org/certifications/archimate.

[172] Opentext (2019, August 21). Understanding 4 types of data users. *Opentext*. Retrieved from https://blogs.opentext.com/understanding-4-types-of-data-users/.

[173] Oracle (2019, August 21a). Database concepts: 21 data integrity. *Oracle Help Center*. Retrieved from https://docs.oracle.com/cd/B19306_01/server.102/b14220/data_int.htm.

[174] Oracle (2019, August 21b). Schema modeling techniques. *Oracle Help Center*. Retrieved from https://docs.oracle.com/cd/B10500_01/server.920/a96520/schemas.htm.

[175] Ostrowski, L. and Helfert, M. (2012). Design science evaluation example of experimental design. In *Journal of Emerging Trends in Computing and Information Sciences. 3(9):253-62*.

[176] Oueslati, W. and Akaichi, J. (2010). A survey on data warehouse evolution. In *International Journal of Database Management Systems (IJDMS), 2(4), pp. 11–24*.

[177] Oxford (2019, August 21a). Classification. *Oxford English dictionary.* Retrieved from http://www.oed.com.

[178] Oxford (2019, August 21b). Feature. *Oxford English dictionary.* Retrieved from https://en.oxforddictionaries.com/definition/feature.

[179] Pan, M. (2016). *Preparing literature reviews: Qualitative and quantitative approaches*. Routledge.

[180] Par, G., Trudel, M., Jaana, M., and Kitsiou, S. (2015). Synthesizing information systems knowledge: A typology of literature reviews. In *Information & Management, 52(2), pp.183-199*.

[181] Pasquale, C. C., Ionescu, B., Ionescu, D., Gurerero, A., et al. (2008). Virtual data warehouse architecture for real-time webgis. In *2008 IEEE Conference on Virtual Environments, Human-Computer Interfaces and Measurement Systems*, pages 80–85. IEEE.

[182] Pasupuleti, P. and Purra, B. (2015). Data lake development with big data. In *Packt Publishing Ltd*.

[183] Patidar, A. and Suman, U. (2015). A survey on software architecture evaluation methods. In *2015 2nd International Conference on Computing for Sustainable Global Development (INDIACom)*, pages 967–972. IEEE.

[184] Patten, M. L. and Newhart, M. (2018). *Understanding Research Methods: An Overview of the Essentials*. Taylor & Francis. New York and London.

[185] Paul, C. P., Jhangiani, R., Chiang, I. A., Dana, C., and Cuttler, C. (2019, August 21). Overview of non-experimental research. *Research methods in psychology*. Retrieved from https://opentext.wsu.edu/carriecuttler/chapter/overview-of-non-experimental-research/.

[186] Pavlo, A., Curino, C., and Zdonik, S. (2012). Skew-aware automatic database partitioning in shared-nothing, parallel OLTP systems. In *Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data (pp. 61–72)*. ACM.

[187] Peffers, K., Rothenberger, M., and Kuechler, B. (2012a). Design science research in information systems. In *Advances in Theory and Practice: Springer Berlin Heidelberg (Lecture Notes in Computer Science)*.

[188] Peffers, K., Rothenberger, M., Tuunanen, T., and Vaezi, R. (2012b). Design science research evaluation. In *Design Science Re-search in Information Systems. Advances in Theory and Practice: Springer Berlin Heidelberg (Lecture Notes in Computer Science), pp. 398–410.*

[189] Peffers, K., Tuunanen, T., Gengler, C. E., Rossi, M., Hui, W., Virtanen, V., and Bragge, J. (2006). The design science research process: a model for producing bragge, j., presenting information systems research. In *Proceedings of the first international conference on design science research in information systems, and Inc.. technology (DESRIST 2006) (pp. 83–106). ME Sharpe*.

[190] Peffers, K., Tuunanen, T., Rothenberger, M. A., and Chatterjee, S. (2007). A design science research methodology for information systems research. In *Journal of management information systems, vol. 24, pp. 45–77*.

[191] Petkov, P. (2016). *Assessing and Analysing Data Quality in Service Ori-*

ented Architectures Developing a Data Quality Process. Doctoral dissertation, Dublin City University.

[192] Phiriyayotha, T. and Rotchanakitumnuai, S. (2013). Data warehouse implementation success factors and the impact of leadership and personality on the relationship between success factors. *Journal of Business and Economics, ISSN 2155-7950, USA, Volume 4, No. 10, pp. 948–956*.

[193] Plunkett, T., Macdonald, B., Nelson, B., Hornick, M., Sun, H., Mohiuddin, K., Harding, D., Mishra, G., Stackowiak, R., Laker, K., et al. (2013). *Oracle big data handbook*. McGraw-Hill Osborne Media.

[194] Poess, M., Rabl, T., Jacobsen, H. A., and Caufield, B. (2014). TPC-DI: the first industry benchmark for data integration. In *Proceedings of the VLDB Endowment*, volume 7, pages 1367–1378. VLDB Endowment.

[195] Ponniah, P. (2004). *Data warehousing fundamentals: a comprehensive guide for IT professionals*. John Wiley & Sons.

[196] Post, A. R., Ai, M., Pai, A. K., Overcash, M., and Stephens, D. S. (2017). Architecting the data loading process for an i2b2 research data warehouse: Full reload versus incremental updating. In *AMIA Annual Symposium Proceedings*, volume 2017, page 1411. American Medical Informatics Association.

[197] Prat, N., Comyn-Wattiau, I., and Akoka, J. (2014). Artifact evaluation in information systems design science research - a holistic view. In *PACIS (p. 23)*.

[198] Prat, N., Comyn-Wattiau, I., and Akoka, J. (2015). A taxonomy of evaluation methods for information systems artifacts. *Journal of Management Information Systems*, 32(3):229–267.

[199] PRISMA (2019, August 21). Prisma. *Flow Diagram Generator.* Retrieved from http://prisma.thetacollaborative.ca/.

[200] Pruijt, L., Wiersema, W., and Brinkkemper, S. (2013). A typology based approach to assign responsibilities to software layers. In *Proceedings of the 20th Conference on Pattern Languages of Programs*, page 2. The Hillside Group.

[201] Qin, H., Jin, X., and Zhang, X. (2012). Research on extract, transform and load (etl) in land and resources star schema data warehouse. In *2012 Fifth International Symposium on Computational Intelligence and Design*, volume 1, pages 120–123. IEEE.

[202] Qu, S. Q. and Dumay, J. (2011). The qualitative research interview. *Qualitative research in accounting & management*, 8(3):238–264.

[203] Rangarajan, S., Liu, H., Wang, H., and Wang, C. (2015). Scalable architecture for personalized healthcare service recommendation using big data lake. In *Service Research and Innovation (pp. 65-79)*. Springer, Cham.

[204] Raschka, S. (2015). *Python machine learning*. Packt Publishing Ltd.

[205] Read, D. W. (2016). *Artifact classification: a conceptual and methodological approach*. Routledge.

[206] Ridley, D. (2012). *The literature review: A step-by-step guide for students*. Sage.

[207] Robb, D. (2017, July 3). Semi-structured data. *Datamation*. Retrieved from https://www.datamation.com/big-data/semi-structured-data.html.

[208] Rouhani, S., Rotbei, S., and Shamizanjani, M. (2017). Meta-synthesis of big data impacts on information systems development. *Journal of Management Analytics*, 4(2):182–201.

[209] Rouse, M. (2019, August 21). Structured data. *Whatis*. Retrieved from https://whatis.techtarget.com/definition/structured-data.

[210] Russom, P. (2011). Big data analytics. In *TDWI Best Practices Report, Fourth Quarter. 2011 Sep 18:1–35*.

[211] Sagiroglu, S. and Sinanc, D. (2013). Big data: A review. In *Collaboration Technologies and Systems (CTS). International Conference on 2013 May 20 (pp. 42–47). IEEE*.

[212] Sahama, T. and Croll, P. (2007). A data warehouse architecture for clinical data warehousing. In *Proceedings of the fifth Australasian symposium on ACSW frontiers-Volume 68 (pp. 227–232)*. Australian Computer Society, Inc.

[213] Sales, T. P., Roelens, B., Poels, G., Guizzardi, G., Guarino, N., and Mylopoulos, J. (2019). A pattern language for value modeling in archimate. In *International Conference on Advanced Information Systems Engineering*, pages 230–245. Springer.

[214] Salinas, G. and Ambler, T. (2009). A taxonomy of brand valuation practice: Methodologies and purposes. *Journal of Brand Management*, 17(1):39–61.

[215] Sarwar, M. M. S., Shahzad, S., and Ahmad, I. (2013). Cyclomatic complexity: The nesting problem. In *Eighth International Conference on Digital Information Management (ICDIM 2013)*, pages 274–279. IEEE.

[216] Scheibe, K. and Nilakanta, S. (2008). Dimensional issues in agricultural data warehouse designs. In *Computers Rai, A., and pp.263-278. electronics in agriculture, 60(2)*.

[217] Schwarz, A., Mehta, M., Johnson, N., and Chin, W. W. (2007). Understanding frameworks and reviews: a commentary to assist us in moving our

field forward by analyzing our past. *ACM SIGMIS Database: the database for Advances in Information Systems*, 38(3):29–50.

[218] Sen, A. and Sinha, A. (2005). A comparison of data warehousing methodologies. In *Communications of the ACM, 48(3), pp.79–84*.

[219] Senapati, R. and Anil Kumar, D. (2014). A survey on data warehouse architecture. In *International Journal of Innovative Research in Computer and Communication Engineering. ISSN ONLINE(2320–9801), PRINT (2320–9798)*.

[220] Shanmugapriya, P. and Suresh, R. (2012). Software architecture evaluation methods-a survey. *International Journal of Computer Applications*, 49(16).

[221] Siddaway, A. (2014). What is a systematic literature review and how do i do one. *University of Stirling: Stirling, UK*, (Volume 1):1–13.

[222] Simitsis, A., Vassiliadis, P., and Sellis, T. (2005). Optimizing etl processes in data warehouses. In *21st International Conference on Data Engineering (ICDE'05)*, pages 564–575. IEEE.

[223] Sinha, A. K. (2001). *Data Warehousing (Connectivity Series)*. Prompt (DPI - 8/01), 1 edition.

[224] Sisodia, D., Singh, L., Sisodia, S., and Saxena, K. (2012). Clustering techniques: a brief survey of different clustering algorithms. In *International Journal of Latest Trends in Engineering and Technology (IJLTET), 1(3), pp.82–87*.

[225] Snow, N. M. and Reck, J. L. (2016). Developing a government reporting taxonomy. *Journal of Information Systems*, 30(2):49–81.

[226] Sqoop (2019, January 18). Sqoop. *Apache*. Retrieved from https://sqoop.apache.org/.

[227] Stake, R. (2013). *Multiple Case Study Analysis*. Guilford Press.

[228] Stein, B. and Morrison, A. (2014). The enterprise data lake: Better integration and deeper analytics. In *PwC Technology Forecast: Rethinking integration, 1, pp.1–9*.

[229] Stewart, R. and Mohamed, S. (2002). IT/IS projects selection using multi-criteria utility theory. *Logistics Information Management*, 15(4):254–270.

[230] Strange, K. and Friedman, T. (2003). Making BI and data warehousing strategic: The key issues. *Stamford, CT: Gartner Group*.

[231] Straub, D., Boudreau, M., and Gefen, D. (2004). Validation guidelines for is positivist research. In *Communications of the Association for Information systems, 13(1), p.24*.

[232] Strauss, A. and Corbin, J. (2008). *Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory (3rd Ed.)*. Thousand Oaks, London, New Delhi: SAGE Publications.

[233] Sultanow, E., Brockmann, C., Schroeder, K., and Cox, S. (2016). A multi-dimensional classification of 55 enterprise architecture frameworks. In *Twenty-second Americas Conference on Information Systems, San Diego*.

[234] Takeda, H., Veerkamp, P., and Yoshikawa, H. (1990). Modeling design process. In *AI magazine, 11(4), p.37*.

[235] Tang, L., Dong, J., Zhao, Y., and Tsai, W. (2010). A classification of enterprise service-oriented architecture. In *2010 Fifth IEEE International Symposium on Service Oriented System Engineering*, pages 74–81. IEEE.

[236] Taylor, B. (2009). *A case study of business intelligence applications for business users*. Research Dissertation, University of Chester.

[237] Teddlie, C. and Tashakkori, A. (2009). *Foundations of mixed methods research: Integrating quantitative and qualitative approaches in the social and behavioral sciences*. Sage.

[238] Teradata (2019, August 21). Teradata data warehouse migration programs. *Teradata.* Retrieved from https://www.teradata.com/Products/Software/Database/Data-Warehouse-Migration.

[239] Terrizzano, I. G., Schwarz, P. M., Roth, M., and Colino, J. E. (2015). Data wrangling: The challenging yourney from the wild to the lake. In *7th Biennial Conference on Innovative Data Systems Research (CIDR 15), Asilomar, California, USA*.

[240] Teschke, M. and Ulbrich, A. (1997). Using materialized views to speed up data warehousing. *University Erlangen-Nuremberg (IMMD VI): Relatório Técnico*.

[241] Thusoo, A., Sarma, J., Jain, N., Shao, Z., Chakka, P., Zhang, N., Antony, S., Liu, H., and Murthy, R. (2010a). Hive-a petabyte scale data warehouse using hadoop. In *Data Engineering (ICDE), 2010 IEEE 26th International Conference on (pp. 996-1005)*. IEEE.

[242] Thusoo, A., Shao, Z., Anthony, S., Borthakur, D., Jain, N., Sarma, J. S., Murthy, R., and Liu, H. (2010b). Data warehousing and analytics infrastructure at facebook. In *Proceedings of the 2010 ACM SIGMOD International Conference on Management of data (pp.1013–1020)*.

[243] TPC (2019, March 19). TPC-DS Result Highlights: Alibaba Cloud E-MapReduce. *TPC-DS Advanced Sort Results List*. Retrieved from http://www.tpc.org/tpcds/results/tpcds_result_detail.asp?id=119031901.

[244] TPC-DI (2019, August 21). Tpc-di is a benchmark for data integration. *TPC-DI*. Retrieved from http://www.tpc.org/tpcdi/default.asp.

[245] TPC-DS (2019, August 21). Tpc-ds is a decision support benchmark. *TPC-DS*. Retrieved from http://www.tpc.org/tpcds/default.asp.

[246] Triguero, I., García, S., and Herrera, F. (2015). Self-labeled techniques for semi-supervised learning: taxonomy, software and empirical study. *Knowledge and Information systems*, 42(2):245–284.

[247] Tung, M. (2018, January 19). Alibaba named to fortunes worlds most-admired companies list. *Alizila*. Retrieved from https://www.alizila.com/alibaba-named-fortunes-worlds-admired-companies-list/.

[248] Tutorialspoint (2019, August 21). Schemas. *Data Warehousing*. Retrieved from https://www.tutorialspoint.com/dwh/dwh_schemas.htm.

[249] Urquhart, C. and Fernández, W. (2016). Using grounded theory method in information systems: the researcher as blank slate and other myths. In *Enacting Research Methods in Information Systems: Volume 1*, pages 129–156. Springer.

[250] Urquhart, C., Lehmann, H., and Myers, M. D. (2010). Putting the theory-back into grounded theory: guidelines for grounded theory studies in information systems. In *Information systems journal, vol. 20, pp. 357–381*.

[251] Vaishnavi, V., K. W. and Petter, S. (2004). Design science research in information systems. *AIS*. Retrieved from http://desrist.org/design-research-in-information-systems.

[252] Vaishnavi, V. K. and Kuechler, W. (2015). *Design science research methods and patterns: innovating information and communication technology*. Crc Press.

[253] Van Aken, J. E. (2005). Management research as a design science: Articulating the research products of mode 2 knowledge production in management. *British journal of management*, 16(1):19–36.

[254] Vanga, S. (2015, September 08). What is the difference between full load and incremental load. *sql server central*. Retrieved from https://www.sqlservercentral.com/blogs/what-is-the-difference-between-full-load-and-incremental-load.

[255] Vassiliadis, P. (2009). Data warehouse metadata. *Encyclopedia of Database Systems*, pages 669–675.

[256] Walliman, N. (2017). *Research methods: The basics*. Routledge.

[257] Walls, J., Widmeyer, G., and El Sawy, O. (2004). Assessing information system design theory in perspective: how useful was our 1992 initial rendition? In *JITTA: Journal of Information Technology Theory and Application. 6(2):43*.

[258] Walsh, D. and Downe, S. (2005). Meta-synthesis method for qualitative research: a literature review. *Journal of advanced nursing*, 50(2):204–211.

[259] Wand, Y., Monarchi, D., Parsons, J., and Woo, C. (1995). Theoretical foundations for conceptual modelling in information systems development. In *Decision Support Systems, 15(4), pp.285–304*.

[260] Wang, R. Y. and Strong, D. M. (1996). Beyond accuracy: What data quality means to data consumers. In *Journal of management information systems*, volume 12, pages 5–33. Taylor & Francis.

[261] Watson, H. and Ariyachandra, T. (2005). Data warehouse architectures: Factors in the selection decision and the success of the architectures. In *Terry College of Business, University of Georgia, pp.1–55*.

[262] Webster, J. and Watson, R. (2002). Analyzing the past to prepare for the future: Writing a literature review. In *MIS quarterly, pp.xiii–xxiii*.

[263] Weishäupl, E., Yasasin, E., and Schryen, G. (2015). A multi-theoretical literature review on information security investments using the resource-based view and the organizational learning theory. *Thirty Sixth International Conference on Information Systems, Fort Worth*.

[264] West, D. B. et al. (1996). *Introduction to graph theory*, volume 2. Prentice hall Upper Saddle River, NJ.

[265] White, C. J. (2000). The IBM business intelligence software solution. *Data Base Associates, Version*, 4.

[266] Wibowo, S., Deng, H., and Zhang, X. (2012). A group decision making procedure for selecting data warehouse systems. In *International Conference on Artificial Intelligence and Computational Intelligence*, pages 301–308. Springer.

[267] Widom, J. (1995). Research problems in data warehousing. In *CIKM '95 Proceedings of the fourth international conference on Information and knowledge management Pages 25–30*.

[268] Wikipedia (2019, June 7). Alibaba group. *Wikipedia*. Retrieved from https://en.wikipedia.org/wiki/Alibaba_Group.

[269] Williams, K., Chatterjee, S., and Rossi, M. (2010). Design of emerging digital services: a taxonomy. In *Design Research in Information Systems*, pages 235–253. Springer.

[270] Winsemann, T. (2011). Valuation factors for the necessity of data persistence in enterprise data warehouses on in-memory databases. In *CAiSE (Doctoral Consortium)*, pages 3–14. Citeseer.

[271] Wisdom, J. and Creswell, J. (2013). Integrating quantitative and qualitative data collection and analysis while studying patient-centered medical home models. In *Agency for Healthcare Reseach and Quality, pp.13–28*.

[272] Wisniewski, M. F., Kieszkowski, P., Zagorski, B. M., Trick, W. E., Sommers, M., and Weinstein, R. A. (2003). Development of a clinical data warehouse for hospital infection control. *Journal of the American Medical Informatics Association*, 10(5):454–462.

[273] Wixom, B. H. and Watson, H. J. (2001). An empirical investigation of the factors affecting data warehousing success. In *MIS quarterly*, pages 17–41. JSTOR.

[274] Wolfram (2019, August 21). Wolfram. *Mathematica.* Retrieved from https://www.wolfram.com/mathematica/.

[275] WorldCup98 (2019, August 21). Worldcup98. Retrieved from ftp://ita.ee.lbl.gov/html/contrib/WorldCup.html.

[276] Wrembel, R. (2011). On handling the evolution of external data sources in a data warehouse architecture. In *Integrations of data warehousing, data mining and database technologies: innovative approaches*, pages 106–147. IGI Global.

[277] Wu, X., Zhu, X., Wu, G.-Q., and Ding, W. (2013). Data mining with big data. *IEEE transactions on knowledge and data engineering*, 26(1):97–107.

[278] Yang, Q. and Helfert, M. (2016a). Data quality for web log data using a hadoop environment. In *21st International Conference on Information Quality. Ciudad Real, Spain*.

[279] Yang, Q. and Helfert, M. (2016b). Revisiting arguments for a three layered data warehousing architecture in the context of the hadoop platform. In *The 6th*

International Conference on Cloud Computing and Services Science (Closer 2016), Roma, Italy. ISBN 978-989-758-182-3.

[280] Yarn (2018, November 13). Apache hadoop yarn. *Apache Hadoop*. Retrieved from https://hadoop.apache.org/docs/current/hadoop-yarn/hadoop-yarn-site/YARN.html.

[281] Yeoh, W. and Koronios, A. (2010). Critical success factors for business intelligence systems. *Journal of computer information systems*, 50(3):23–32.

[282] Yin, R. (2009). *Case study research: Design and methods (applied social research methods)*. London and Singapore: Sage.

[283] Zellal, N. and Zaouia, A. (2016). A measurement model for factors influencing data quality in data warehouse. In *2016 4th IEEE International Colloquium on Information Science and Technology (CiSt)*, pages 46–51. IEEE.

[284] Zhang, Q., Qiu, D., Tian, Q., and Sun, L. (2010). Object-oriented software architecture recovery using a new hybrid clustering algorithm. In *2010 Seventh International Conference on Fuzzy Systems and Knowledge Discovery*, volume 6, pages 2546–2550. IEEE.

[285] Zhang, S., Zhang, C., and Yang, Q. (2003). Data preparation for data mining. *Applied artificial intelligence*, 17(5-6):375–381.

[286] Zhang, X., Wang, Y., and Wu, L. (2019). Research on cross language text keyword extraction based on information entropy and textrank. In *2019 IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)*, pages 16–19. IEEE.

[287] Zhou, Q. and Sun, L.-j. (2009). The architecture and design strategy for data warehouse of highway management. In *2009 Second International Con-*

ference on Intelligent Computation Technology and Automation, volume 4, pages 459–462. IEEE.

[288] Zhu, H., Gu, P., and Wang, J. (2008). Shifted declustering: a placement-ideal layout scheme for multi-way replication storage architecture. In *Proceedings of the 22nd annual international conference on Supercomputing (pp. 134–144)*. ACM.

[289] Zicari, R. V. (2014). Big data: Challenges and opportunities. *Big data computing*, 1:103–128.

[290] Zikopoulos, P., Eaton, C., et al. (2011). *Understanding big data: Analytics for enterprise class hadoop and streaming data*. McGraw-Hill Osborne Media.

A Appendix - Materials in an Interview

A.1 Participant Consent Form

I am _____ and voluntarily agree to participate in the research study entitled **Benchmarking Data Warehouse Architectures A Feature Based Modelling and Evaluation Methodology**.

I understand that even if I consent to participate now, I can still withdraw and leave during this experiment.

I understand that this study does not ask any ethical question and I can reject to respond any question.

I understand that this study will not collect any personal/sensitive data and I can decline any question referring to this type of data.

I agree to provide information for this study and it will only be used in this research.

I agree that my identity will be anonymised in any document on the results of this research. This information will be confidential and I cannot be identified.

I understand that I would be contacted to seek further clarification and information.

I have read this document and have understood details of the study explained to me.

Signature of research participant:

Signature of participant: _____

Date: _____

Signature of researcher:

Signature of participant: _____

Date: _____

A.2 Questions and Tasks Form

Part	Questions and Tasks	Answer			
1	1.1 Please compare the first and second modelled architectures refer-ring to page 1 and page 2. Do you think the additional information in orange boxes is useful to better understand/evaluate data warehouse architectures?	Yes	Maybe	Unsure	No
	1.2 Referring to Page 1 and 2, Please compare the first and second modelled architectures refer-ring to page 1 and page 2. Do you think the extended component (Schema) in the purple box is useful to better understand/evaluate a data warehouse architecture?	Yes	Maybe	Unsure	No
	1.3 Please give a brief reason.				
2	2.1 Referring to the table and radar chart in Page 3, which architecture do you choose?	Case 2	Case 3	Same	Unsure
	2.2 Please give a brief reason.				
3	3.1 Do you think the evaluated results are explicit and easy to understand?	Yes	Maybe	Unsure	No
	3.2 How much confidence do you have to use this methodology for a data warehouse architecture evaluation? (From 0 to 100%)				
	3.3 Please give a brief reason.				

A.3 Case 2

Page 1

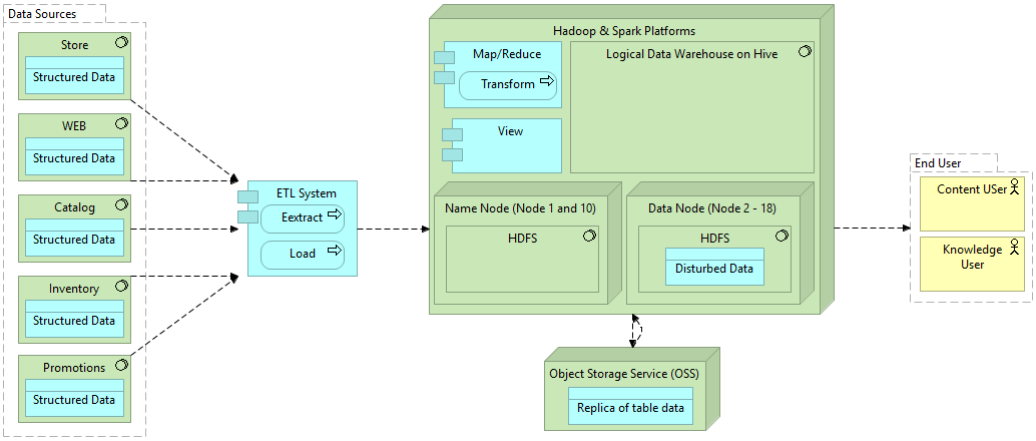


Figure A.1: The modelled data warehouse architecture of case 2 with extended elements and extra information

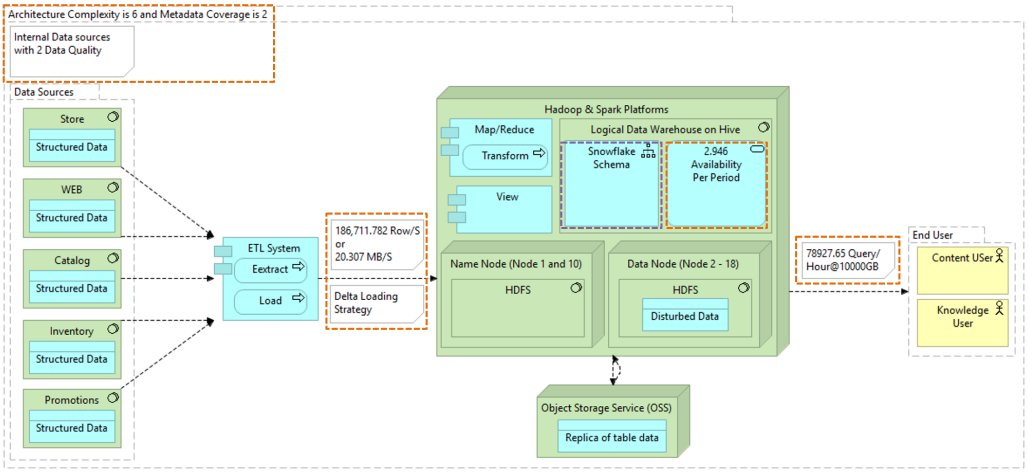


Figure A.2: The modelled data warehouse architecture of case 2 with default elements

A.4 Case 3

Page 2

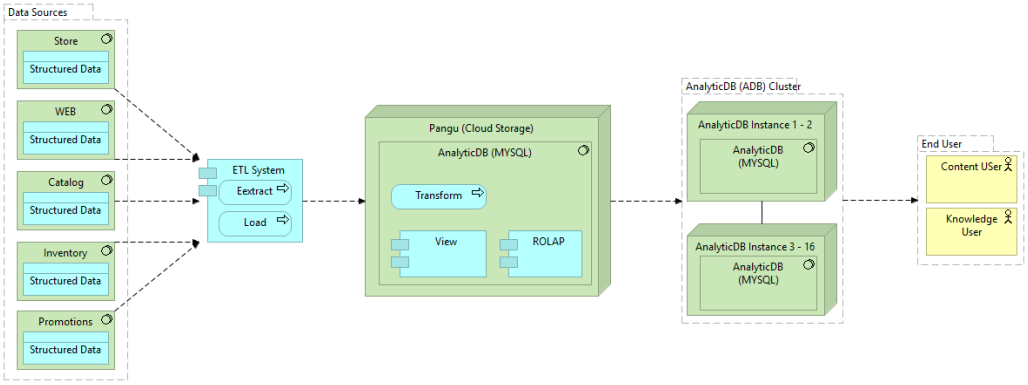


Figure A.3: The modelled data warehouse architecture of case 3 with extended elements and extra information

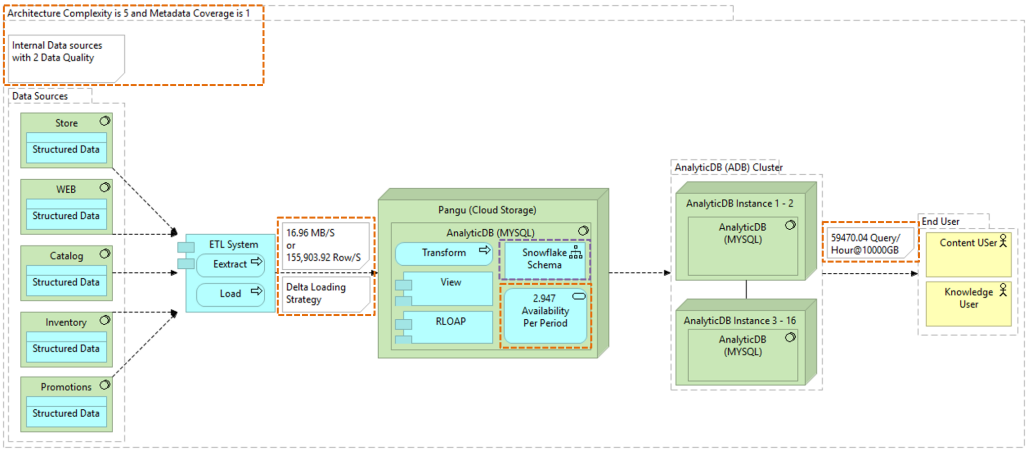


Figure A.4: The modelled data warehouse architecture of case 3 with default elements

A.5 Evaluated Results

Page 3

Layer	Indicator	Case 2	Case 3
Source Layer	Source & Format Ave.	1	1
	Data Quality	2	2
ETL Layer	ETL & Load Ave.	2.5	2.5
	Loading Throughput	2.539	2.12
DWH Layer	DWH & Schema Ave.	2.5	2.5
	Time Period Rate	2.946	2.947
Presentation Layer	OLAP & View Ave.	0.5	1
	Query Throughput	2.368	1.784
Other	User & Meta Ave.	2.5	2
	Architecture Complexity	3	3

Table A.1: The evaluated results of cases 2 and 3 with 10 indicators

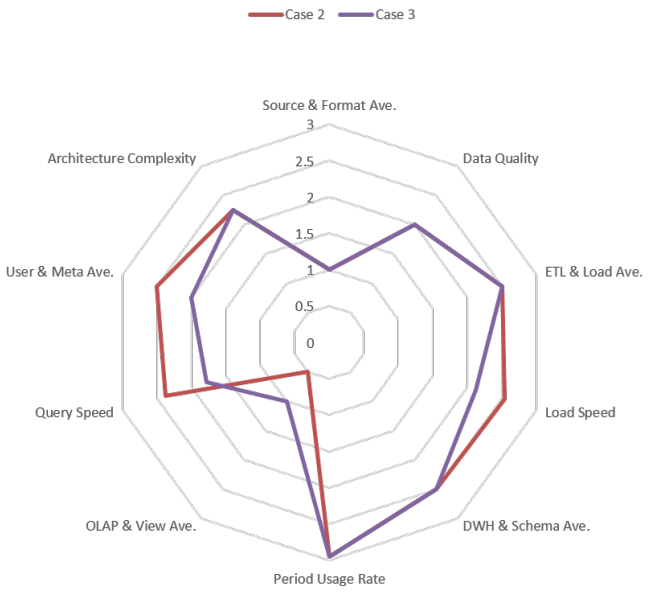


Figure A.5: Radar charts of cases 2 and 3 for comparison