MASTER IN BUSINESS ENGINEERING PROFESSIONAL FOCUS IN DATA SCIENCE

Paving The Way Towards Full ETL Automation: A Systematic Literature Review of ETL Services

ABRAS, Jordan

Award date:
2023

Awarding institution:
University of Namur

Paving The Way Towards Full ETL Automation: A Systematic Literature Review of ETL Services

Jordan ABRAS

Directeur: Prof. C. BURNAY

Mémoire présenté
en vue de l'obtention du titre de
Master 120 en ingénieur de gestion, à finalité spécialisée
en data science

ANNEE ACADEMIQUE 2022-2023

Université de Namur, ASBL
Faculté des Sciences économiques, sociales et de gestion – Département des Sciences de gestion

Rempart de la Vierge 8, B-5000 Namur, Belgique, Tel. +32 [0]81 72 49 58/48 41

Acknowledgement

I would like to express my deepest gratitude to the following individuals whose support and guidance have been instrumental in the completion of this master's thesis as well as of my academic journey as a whole.

First and foremost, I would like to sincerely thank my thesis supervisor, Prof. Corentin Burnay, for his invaluable guidance, expertise, and invaluable support throughout my research journey. His dedication, knowledge, insightful suggestions, and constructive feedback have greatly contributed to shaping the direction and quality of this thesis. I am truly grateful for his mentorship and the trust he placed in me.

I would also like to acknowledge Benito Giunta, for his assistance and cooperation. His prompt responses, technical expertise, and willingness to help have been invaluable in addressing my queries and resolving various challenges I encountered along the way of writing this master's thesis.

I am deeply indebted to my parents, as well as my sisters, for their unconditional love, encouragement, and constant support throughout all my studies. Their perpetual belief in my abilities and their sacrifices have been the driving force behind my academic achievements. I am grateful for their unwavering presence in my life, and I owe my success to their resolute faith in me.

To my dear grandparents, I extend my heartfelt appreciation for their love, wisdom, and unwavering support. Their encouragement and inspirational stories have instilled in me a strong sense of determination and resilience, motivating me to pursue my academic goals with relentless enthusiasm. Their belief in me has been a constant source of strength, and I am truly grateful for their presence in my life.

I would also like to express my deepest gratitude to my partner. Her love, patience, and understanding have been the cornerstone of my emotional well-being throughout this academic journey. Her belief in me, constant encouragement, and willingness to lend a listening ear during moments of doubt have provided me with the strength and motivation to overcome challenges and persevere. I am grateful for her support and for standing by my side every step of the way.

Finally, I extend my heartfelt thanks to my friends, for the wonderful years we have spent together and the mutual support we have shared during our academic challenges. Their friendship, camaraderie, and the countless moments of laughter and inspiration have enriched my academic experience beyond measure. I am grateful for their support, intellectual discussions, and shared memories, which have made this journey all the more meaningful and enjoyable.

To all those mentioned above, and to everyone else who has contributed to my academic and personal growth, thank you from the bottom of my heart. Your support, encouragement, and belief in me have been invaluable, and I am truly grateful for the impact you have had on my life.

Résumé/Summary

Résumé

Les systèmes de Business Intelligence (BI) sont de plus en plus utilisés pour la prise de décision, mais peuvent être perçus comme risqués et coûteux à mettre en place. Pour réduire les risques perçus, l'automatisation du processus extract-transform-load (ETL) est proposée. Pour cela, il sera nécessaire de standardiser les protocoles de fonctionnement de chaque service composant l'ETL, ce qui nécessite de dresser une liste exhaustive de tous les services ETL à considérer. Nous avons donc effectué une revue systématique de la littérature afin de récolter ces services, ce qui nous a permis d'établir une taxonomie étendue des services ETL qui guidera les futures recherches sur l'automatisation du processus ETL.

Summary

Nowadays, decision-making is increasingly relying on Business Intelligence (BI) systems. These systems however may be perceived risky due to the effort and investment needed to set them up. One way to reduce the perceived risk of deploying such systems is to automate their most effort-demanding and costly component: the extract-transform-load (ETL) process, responsible for data integration. To effectively automate this process, it will be necessary to define standardization protocols for each of its service, therefore leading to the need for a comprehensive list of all ETL services. To build up this exhaustive list, we performed a systematic literature review to retrieve each ETL service that is discussed in the ETL literature on BI systems. Using this methodology, we derived an extended taxonomy of ETL services that will guide us in the further researches on automating the ETL process.

Contents

List of Figures

List of Tables

Glossary

domain-specific modeling (DSM) corresponds to applying MDD to model a system using a formal language that has been developed especially for the system's domain [Petrović et al., 2017].

goal modeling Using formal models to represent the goals and objectives of an organization regarding a system [Giorgini et al., 2003].

model-driven architecture (MDA) Proposes the use of different models to represent a system. First, the Computation Independent Model (CIM), it abstracts the system under modelisation at the level of its environment and system requirements, disregarding any computerized aspects. Then, this CIM is transformed into a Platform-Independent Model (PIM). This consider the system under study in relation with its computerized components, taking into account the components of the system that will be computer-based, disregarding the technology to be used. The last model in MDA is the Platform-Specific Model, which models the system under study by taking into account specific aspects related to the technology on which it is to be deployed. The goal of MDA is to allow the study of a system considering different levels of abstraction that can be automatically transformed from one another, facilitating the development of systems [Pastor et al., 2008].

model-driven development (MDD) Applying the model-driven architecture to software development.

ontology Formal knowledge representation of a domain using the specific elements it is composed of as well as their interconnection [Happel and Seedorf, 2006].

query-view-transformation Family of models transformation languages allowing the automated transformation of models from one another. This family of language consider a query (source element to be transformed), a view (the end representation of the source element), and the rule for transforming the query into the view (transformation) [Muñoz et al., 2009].

Paving the Way Towards Full ETL Automation : A Systematic Literature Review of ETL Services

Jordan ABRAS

June 2023

1 Introduction

Our world is changing, frequently, rapidly, and is often labelled "VUCA": Volatile, Uncertain, Complex, Ambiguous [Mack et al., 2015]. A direct consequence of this for organizations is that making the right decision at the right moment has become critical in order to adopt the adequate reaction, seize opportunities and remain competitive [Bucher et al., 2009]. To tame this uncertainty and support the right-time decision-making process, it has become a common practice in organizations to deploy data-driven decision support systems, in particular, Business Intelligence systems [Michalczyk et al., 2020]. Business Intelligence (BI hereafter) systems encompass all the methodologies, tools, techniques, and applications used in an organization to integrate, analyze, and visualize the data available to support the decision-makers [Negash, 2004].

Setting up a BI system is a cumbersome process: the traditional BI architecture is composed of several layers, each requiring its own professionals and tools to be developed. Building a full BI system can therefore represent a huge investment for some companies, not only in terms of financial costs but also in terms of energy and time. This, in turn, can lead organizations to delay or avoid the implementation of BI, thereby decreasing their ability to manoeuvre in the VUCA world, and hence increasing the risk of missed opportunities offered by internal or external data, and the risk of erroneous and costly decisions [Olszak and Ziemba, 2012].

One way to make BI deployment easier is to automate its development. Automating part of the deployment effort would lower the number of IT specialists to hire, thereby decreasing the overall BI costs [Rodic and Baranovic, 2009]. Another benefit of automation is that tasks are performed in a standardized manner, improving their efficiency and the quality of their outcomes. A parallel is the Lean Manufacturing System of Toyota, where automation of the production line led to decreased costs for higher-quality cars [Gupta and Jain, 2013]. Automation of BI systems' deployment can also lead to an improved reusability and a better scalability as well as a reaching of the Self-Service BI paradigm [Alpar and Schulz, 2016]. Finally, an automated process is usually faster. Decreasing the time needed to set up the system allows a higher rate of success for BI projects, as time is a critical success factor in BI system development [Olszak and Ziemba, 2012].

Among all the layers of the traditional BI architecture, it is commonly recognized that the most costly and time- and energy-consuming is the Extract-Transform-Load (ETL) process [Dupor and Jovanović, 2014, Vassiliadis et al., 2002]. ETL is in charge of the extraction, cleansing, transformation, and loading of the data from their source to their destination. The fact that this process often accounts for 80% of the development effort and 30% of the deployment costs makes it a good candidate for automation, as an automation of this process can help drastically decrease the needed initial investment and effort and, by doing so, an organization's perceived risk of deploying such system.

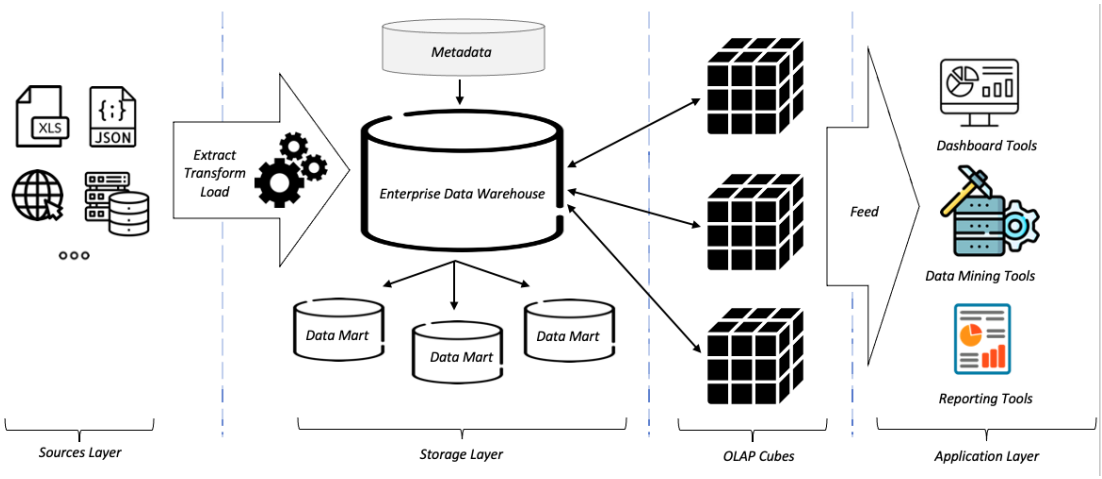

Figure 2.1: Traditional Business Intelligence Architecture

The first step towards automating the ETL process is to obtain a clear understanding of all the services and functionalities currently supported, in order to clarify the requirements for a future automated solution. This is not a trivial question, as there are numerous conceptual models proposed in the literature that cover various aspects, often complementary, related to data or metadata processing in the ETL flow. Industrial tools are also extremely rich and diverse, each with its own specificities. To the best of our knowledge, there is a lack of a clear, comprehensive, and standardized view of the services provided by ETL systems.

As an answer, this master's thesis intends to treat the following research question: "What are the important ETL services that have to be considered when designing and developing a fully automated ETL framework?". By ETL service, we mean any action, treatment, feature, manipulation offered by an ETL tool to treat data. To do so, we will perform a Systematic Literature Review (SLR) of all the articles dealing with the ETL process, following the methodology defined by Kitchenham in [Kitchenham, 2004].

The rest of this work is articulated as follows: Section 2 will present the traditional BI architecture and its impact on the ETL process. Then, we present in Section 3 the related works of this master's thesis. In Section 4, we present the methodology we followed to perform the SLR. The results from the SLR are presented in Section 5, as well as an articulation of the ETL services identified. In Section 6 we discuss the results from the SLR, the limitation of this study and the future research tracks. We then conclude.

2 Background - Traditional BI Architecture and ETL

As mentioned in the introduction, decision-making has become a key activity in all companies today. In fact, it can help gain market share, minimize losses, and seize opportunities [Michalczyk et al., 2020]. Therefore, decisions within organizations are increasingly data-driven, and these data must be made available to enable decision-makers to make fully informed decisions. To achieve this goal of giving access to data, more and more companies are turning to BI systems as data-driven decision support systems. Figure 2.1 depicts a typical BI architecture. The BI architecture is typically divided in four main layers: the source layer, the storage layer, the OLAP layer and the application layer [Habibu, 2013, Vaisman and Zimányi, 2014].

2.1 The Source Layer

The **source layer** is composed of all the data sources that an organization can access to extract the data it needs. This layer often consists of internal operational systems, containing data stored in various storage formats such as SQL databases or even flat files like CSV, XML or JSON, and external data sources originating from APIs, on which the company has no ownership.

2.2 The Storage Layer

The second layer to consider when building a BI system is the **storage layer**. In the common architecture presented by Inmon in [Inmon, 2005], this layer is composed of a unified repository called Enterprise Data Warehouse (EDW), which is often fragmented into multiple Data Marts, corresponding to departmental sub-DWs, and contains a Metadata Container storing data about the DW's multidimensional schema, about the ETL transformations, the business rules,...

A Data Warehouse (DW) is defined by Inmon and Kelley as: "a collection of subject-oriented, integrated, non-volatile and time-varying data to support management decision"[Inmon and Kelley, 1993]. In other words, a DW is a data store that centralizes data from multiple disparate sources (integrated) that are oriented towards the business's subjects of interest (subject-oriented), that are collected and stored over time (time variant) and that are immutable (non-volatile).

Data are represented in a DW using a multidimensional model consisting of interrelated facts and dimensions [Vaisman and Zimányi, 2014].

The **facts** correspond to the business events about which we want to collect and analyze data. For example, in a retail company, facts might represent data about sales. Fact tables typically contain numerical values related to these business events called **measures**. In the sales example, the fact table could store the price of each product purchased and the total amount of the receipt.

Dimensions, on the other hand, correspond to analysis perspectives according to which we want to analyse data. Dimension tables typically store textual values corresponding to details about the business events contained in the fact tables. The dimensions in the retail example may be time, geography, and product, which would be the time and location of the transaction as well as any related goods. A dimension often consists of a **hierarchy** of dimensional levels. For example, the geography dimension could be made up of the customer, store, and region levels, which would allow to evaluate sales data at the level of either the customer, the store, or the region from which the data were collected. To ensure consistency when moving up in a hierarchy, each measure in fact tables is associated with an **aggregation function** (SUM, AVG, MAX,...). In this way, data are aggregated to a higher level when the level of granularity of the analysis is decreased.

To sum up, data contained in a DW can be represented as a (hyper-)cube, where each cell of the the cube corresponds to a fact and each edge to a dimension.

At the logical level, however, there exists different ways to actually set up the storage of data. The most common method is to implement the specificities of the multidimensional model using a traditional relational approach, which is handled by the usual database management systems. The steps to transition from a multidimensional conceptual model to a relational model are presented in [Vaisman and Zimányi, 2014]. Nonetheless, there

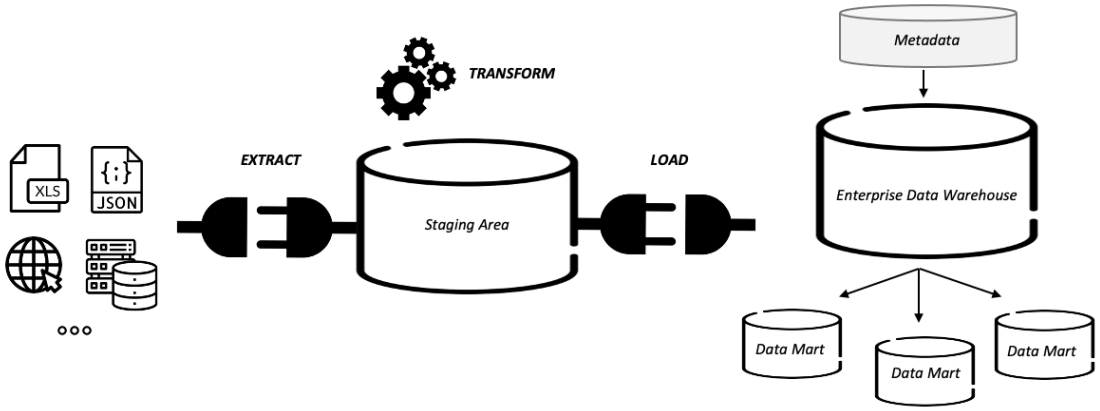

Figure 2.2: ETL architecture considering the staging area.

actually exists a multidimensional approach to store data, but this one is less efficient in terms of storage. There also exists hybrid approaches combining the advantages from relational and multidimensional data storage [Vaisman and Zimányi, 2014].

2.3 The Extract-Transform-Load Process

There are two challenges raised by the data sources from the source layer. First, the data are stored in extremely heterogenous forms, be it from a technical standpoint (databases, flat files, API response, etc.) or from a modeling perspective (relational, unstructured, graph databases, etc.). Accessing these data and ensuring that the system can process them in an integrated manner, regardless of where and how these are stored, represents a first important difficulty. Another challenge is that data referring to the same business object can be stored in different ways depending on their source. For example, the postal code of a customer's address can be stored as part of a full address field stored as a string in one source file, and as a separate integer field in another source database. Therefore, it is necessary to create a mapping between semantically identical data to remove type and encoding differences to ensure a unified data handling methodology.

These two challenges are one motive for the so-called Extract-Trasnform-Load (ETL) process, bridging the gap between the first and second layer of the BI architecture.

ETL processes are designed to access (extract) data from sources and allow a centralized and integrated workspace to work with them, called staging area. A staging area (SA) serves as a provisional storage space that houses the data tables during the extraction process (see Figure 2.2). Transformations (transform) are then applied to the tables within the SA and, once the transformation stage has come to an end, and that the data is deemed fit for loading, the tables are transported from the SA to the DW (load) and the SA is truncated [El-Sappagh et al., 2011]. The ultimate goal is to provide the rest of the BI system with a consistent, integrated, and unique data source that can be queried by the applications used by the decision-makers.

The ETL layer is likewise impacted by the fragmentation of the storage layer. Prior to loading data, one must first gather and retain metadata regarding the changes the source data went through. Additionally, one also needs to consider the destination of the data in the loading phase (EDW vs. data marts).

The multidimensional representation of data in a DW also affects the ETL process.

Knowing whether a piece of data belongs to a fact table or a dimension table is important since they are not loaded simultaneously in the same manner. In addition, dimensions may contain changing values that require special handling and processing (called slowly changing dimensions) [Faisal and Sarwar, 2014]. These particular aspects, among others, are additional motives for developing the ETL layer of a BI system.

In conclusion, the essence of the ETL process is to tackle the variations in source data locations, formats, and types when transferring them to the centralized and integrated DW. The ETL tier is also useful to bridge raw data from traditional databases and DW concepts specific to the multidimensional modeling used in BI systems.

It is important to mention that the architecture dealt in this master's thesis is the traditional BI architecture that considers an ETL process to handle data transformation and loading from the sources to the DW. There exists other variants of ETL that are used in alternative BI systems, each having its own advantages depending on the context it is used. We can cite the E-L-T process (Extract-Load-Transform), where the data are first loaded into a storage layer before being processed and transformed. This particular type of ETL is mainly used in BI environment using a data lake to store data from all sources available to the organization [Khine and Wang, 2018]. Another type of ETL is the "reverse ETL" where the principle is to extract data from the centralized DW to feed operational application with data coming from a single source of truth [Dash and Swayamsiddha, 2022]. These alternative integration processes deserve to be studied as well, but the point of this master's thesis is to remain general and to target the most common approach to data integration – the traditional ETL process – that has the greatest scope for automation.

2.4 OLAP Cubes and Application Layer

The next layers, while being very important in BI systems, are of lesser relevance in this master's thesis and are therefore presented in less details.

The application layer (rightmost part of the architecture in Figure 2.1) corresponds to the different data visualization and analysis tools (reporting interfaces, data mining tools, etc.) used or developed to support decision-makers.

These applications retrieve the data required by the decision-makers from OLAP servers (cubes). Each OLAP cube corresponds to a precomputed refinement of the data contained in the DW. OLAP cubes are obtained by performing OLAP (OnLine Analytical Processing) queries on the DW to pre-select a subset of data at some specific dimensional levels. The main purpose of using OLAP servers is mainly to facilitate and speed up the visualization of data on the application side [Chaudhuri and Dayal, 1997].

To conclude, the traditional BI architecture as presented in [Habibu, 2013, Vaisman and Zimányi, 2014] is composed of four layers. First, the source layer which consists in all data sources available to the organization. Then, the storage layer, consisting in a centralized multidimensional representation of data oriented towards decision-making called DW. An ETL process is grown in order to bridge data from the first layer to the second and to ensure high-quality data. The OLAP layer consists in sub-DWs (called OLAP cubes) that are precomputed to increase data delivery time to representation tools from the last layer – the application layer.

The goal of this section was to introduce the traditional BI architecture and give the scope of the ETL process, demonstrating its importance and the factors that impact it. Next section presents an overview of the works related to our field of research.

3 Related Work

In terms of related works, only few publications were found to actually deal with ETL automation, with rather specific scopes.

Firstly, as mentioned in [Mondal et al., 2020], the majority of articles discussing the automation of the ETL process concentrates on the conceptual level. Model-driven development approaches for creating ETL procedures are presented in a significant number of research papers. In [Petrović et al., 2017], a *domain-specific modeling* (DSM) approach is used to formalize the ETL procedures, allowing for a clear modeling of the ETL operations that can then be processed by the authors' built tool (ETL-PL) to produce executable code. In accordance with the same reasoning, [Tomingas et al., 2015] provides a model-driven (MD) method for developing metadata models that store details about the mappings between source and target data using XML schema filled out with *domain-specific language*. Additionally, they produce templates for common ETL SQL queries that may be executed by a template engine and reassembled with data from the metadata models.
[Chávez and Li, 2011] and [Theodorou et al., 2014] use ontologies as models in their articles to implement their MD approach to automating ETL procedures. [Chávez and Li, 2011] first suggests modeling the application domain of the data the ETL will process using ontologies. The ontology will then be processed by an ontology processor to produce a data model for the DW, a set of rules for extracting data from sources based on information about the source stored in the ontology, and a set of rules for loading data at destinations based on information held about the destination. [Theodorou et al., 2014] suggests a similar approach that incorporates goal modeling into the ontology to take into account business requirements. In [Muñoz et al., 2009], the authors suggest using modeling notations like UML class diagrams to create a high-level model of the ETL workflow. This model will then be transformed into platform-specific models by a set of formalized *query-view-transformation* transformations, which can then be processed by the target integration system to be used. The *Generation of Complex Integration Processes* (GCIP) framework, which is also a model-driven approach, is suggested in [Böhm et al., 2009]. It aims to model integration processes using UML or BPMN which can be automatically transformed into platform-specific models that can be processed by the intended integration system to be executed.
These papers do not focus on automating the ETL process itself, but on automatically transforming pre-defined ETL models into executable code.

Conversely, there are only few papers among the ones discovered that cover an ETL process automation in the manner that this master's thesis does. The first of the two articles we discovered is [Mondal et al., 2020], which suggests automating the data preparation step prior to the transformation phase to ensure adequate data extraction and treatment as well as data quality using machine learning algorithms. The second is [Radhakrishna et al., 2012]. By outlining the potential for inserting a layer composed of pre-written scripts between the data sources and the DW that would carry out the ETL processes in a planned way, the authors of this study open up the possibilities for future researches but does not provide any technological frame.

These two publications, the only ones that have been identified to do so, do not require the ETL process to be modelled in order to be fully automated. However, the suggested frameworks lack any practical applications or technological fixes and are essentially theoretical in nature.

Furthermore, as stated in the introduction, no source has been found that presents a full description of all the ETL services considered inside the framework it suggests. Vassiliadis et al. provide a brief summary of some typical ETL services in [Vassiliadis et al., 2003], although there are some activities like resolving duplicates and misspellings that are not covered but cited in other papers as [Mukherjee and Kar, 2017, Wijaya and Pudjoatmodjo, 2015]. [Nwokeji et al., 2018] undertakes a systematic literature review aiming at discovering the common approaches to implementing ETL solutions for handling "big data". [Kherdekar and Metkewar, 2016] and [Mukherjee and Kar, 2017], as other papers, discuss and review the numerous ETL tools available in the BI industry.

4 Methodology

Our SLR was conducted according to the methodology described in [Kitchenham, 2004]. Kitchenham defined three major stages for the review which are summarized in Figure 4.1: first, **planning** the review, then **conducting** the review, and finally, **reporting** the review. The details of each of these phases are presented below.

4.1 Planning the Review

The goal of the **planning** phase is to plan the way the review will be executed. In this phase are discussed the objectives and rationale of the SLR but also the review protocol that ought to guide the reviewing process.

4.1.1 Identifying the Need for a SLR

As a reminder, we found no scientific paper proposing a comprehensive review of all possible ETL services, but we found many articles and propositions advancing their own list of services. During our consultation of the literature, it was evident that some services were recurring while others were more specific and scarcely mentioned. In other words, there are numerous existing frameworks and models that are loosely aligned with each other, resulting in a global truth about ETL conception and development that is disseminated across several knowledge sources. We consider any scientific mention of a service is relevant to consider, so that a systematic listing of all existing services is the end-result we are looking for. Therefore, the adoption of an SLR strategy appears to be the way to go.

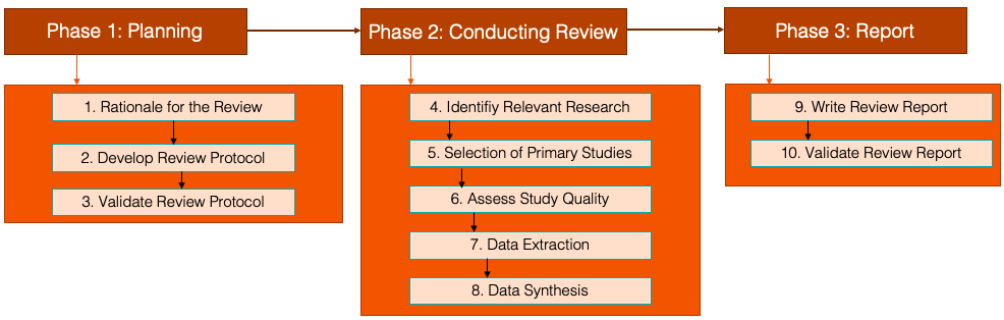

Figure 4.1: SLR procedure adapted from [Kitchenham, 2004]

4.1.2 Review Protocol

The next step in the planning phase is to develop a review protocol. The purpose of this protocol is to remove any selection bias from the research reviewing process but also to expose inclusion/exclusion criteria that are used to select the researches to be reviewed. It works as a guide for the SLR conduction.

The first component of a review protocol consists in **research questions** that the SLR intends to answer. In this work, the research question is the following:

RQ: *"What are all the ETL services mentioned in the scientific literature on data extraction, transformation, and loading applied to Business Intelligence?"*

A **search strategy** must then be established in order to find the studies that will be included in the SLR. We decided to perform our searches on the most prominent scientific digital libraries, renown for holding high quality studies on software engineering and development, namely:

- IEEE eXplore
- Scopus
- Springer
- ACM Communications Digital Library

To query these digital libraries, the next step was to elaborate search strings. To do so, we identified the following set of keywords:

- ETL
- Data Extraction ETL
- Data Transformation ETL
- Data Loading ETL
- Extract-Transform-Load
- ETL Data Integration
- ETL Services
- ETL Operations
- ETL Framework

Then, based on these keywords, we used boolean ANDs and ORs to elaborate a more sophisticated search string:

(etl) OR [etl AND (data extraction OR data transformation OR data integration OR services OR operations OR framework)] OR (extract-transform-load)

We undertook a twofold search strategy in this master's thesis: first a preliminary automated search and then a manual search to retrieve more potentially relevant researches.

To automate the retrieval of preliminary researches, we made the decision to create a Python script that would produce the various search strings specified above and use them to call the APIs of the various scientific platforms identified. Except for the ACM Communications Digital Library, for which we were unable to uncover any Python-callable API and had to proceed to manual searches, this approach was successful across all platforms.

	Inclusion Criteria
IC1	Papers written in English.
IC2	Papers written after the year 2000.
IC3	Papers presented in Journals or Conference Proceedings.
IC4	Parts of the search string are referred in the title.
IC5	Abstract, introduction, conclusion and main parts of the text give indication on the relevance of the paper for RQ.

Table 4.1: Inclusion criteria used to select papers to review.

	Exclusion Criteria
EC1	Papers written in another language than English.
EC2	Papers written before the year 2000.
EC3	Papers presented in other venue than Journals or Conference proceedings.
EC4	The title does not give indication that the article can answer RQ.
EC5	Abstract, introduction, conclusion and main parts of the text do not adress RQ.

Table 4.2: Exclusion criteria used to select papers to review.

After this automated search, we performed a manual search using a snowballing process as presented by [Wohlin, 2014]. The goal was to identify more potentially relevant papers by looking at either the list of references of one paper, or the list of papers citing this first paper.

To complete the search strategy, inclusion/exclusion criteria were applied to the identified researches to discard irrelevant papers and set a scope for this review. As presented in Tables 4.1 and 4.2, we decided in this study to only retain papers written in English with a publication date posterior to 2000. The point of only keeping articles written after the year 2000 is that, as mentioned in [Ain et al., 2019], the first academic papers discussing BI were published in that year. Also, we decided to only review papers presented in journals or conference proceedings to ensure the quality of the discussed researches. After all these criteria have been applied, we analyzed the title of the remaining candidates article and judge on their relevancy about the research question. We only kept researches for which we were sure, after having read the title, that they were discussing ETL services in BI systems. For the remaining researches, we scanned the whole work by looking at the abstract, the introduction, the conclusion and the most important parts. After that fast reading, we made a decision on the relevancy of the paper based on the intuition from this scan.

For the first automated search, we were able to automatically apply inclusion and exclusion criteria IC1 to IC3 and EC1 to EC3 when calling the websites' APIs, thus verifying the relevance of the papers on the go. For inclusion and exclusion criteria IC4-IC5 and EC4-EC5, we had to manually download the paper and read the parts of interest to remove from the tentative set of papers the ones that did not demonstrate enough relevance.

After this first search, we performed a snowballing procedure on the candidate papers for the SLR to retrieve more relevant papers [Wohlin, 2014]. As our start set of papers contained rather recent papers – with publication date ranging between 2009 and 2022 – it seemed relevant to proceed to a backward snowballing process, i.e., identify new papers from the reference list of already selected papers. In that way, we were able to unveil papers written before 2009 that were yet relevant. We however did not proceed

	Extracted Element	Description
E1	ID	An identifier that was generated for each research.
E2	Title	Title of the research.
E3	Type	Type of publication venue, either Journal or Conference Proceedings.
E4	Year	Publication year of the research.
E5	Author(s)	Name and initial of the author(s) of the research.
E6	Cites	(When available) Number of references made to this research.
E7	ETL Services	List of all the ETL services presented in the research.

Table 4.3: Table presenting the elements extracted from the reviewed researches.

to a forward snowballing procedure, i.e., searching new papers from citations. [Jalali and Wohlin, 2012] indeed mentions a potentially great overlap between papers from the start set, the identified papers extracted from the backward snowballing, and the papers extracted from the forward snowballing, resulting in a loss of time and energy.
We applied the same inclusion/exclusion criteria presented in Table 4.1 and 4.2 to the references identified, therefore only selecting the relevant referenced papers.

After having applied inclusion/exclusion criteria to the studies, we had to assess their quality. To do so, we identified three quality criteria:

QC1: Does the context of the study pertain to ETL process in BI systems?

QC2: Does the papers lead to the proposition of a conceptual model to represent the ETL process or to a formalized presentation of ETL services, regardless of the used format?

QC3: Does the paper present any case study or demonstration of the usability of its outcome in actual BI systems?

If the paper was compliant with at least one of these quality criteria, we kept it for the review, otherwise it was discarded.

After all the papers to review have been selected and their quality was assessed, the next step was to define a data extraction strategy, or the process that ought to be used to draw knowledge from the selected researches.

The approach taken in the context of this master's thesis was to read the papers first. Then, we extracted from these papers the following information: the title, the publication type, the publication year, the author(s), the number of references made to it (when available), and a list of all the ETL services covered in the paper. Each paper was assigned an identifier to ease the synthesis process. Table 4.3 summarizes these elements.

To integrate those data we used an Excel spreadsheet containing three matrices: the first one, the *Paper Matrix*, contained elements E1 to E6. The point of this matrix was to give metadata about the researches. The second one, the *ETL Category Matrix*, containing element E7, aimed at giving an overview of the ETL categories that were dealt in the selected researches. And the third one, the *ETL Service Matrix*, contained the element E8, that means, the list of all the ETL services considered in the researches. The complete matrices can be found in Section 5.

Finally, it is necessary to specify how the extracted data will be **synthesized**. After completion of the reviewing process, the identified ETL services were plotted on a high-level ETL workflow and assigned to a subcategory of services. This workflow as well as the explanations of the identified ETL services are provided in Section 5.

4.1.3 Validation of the Review Protocol

After this review protocol was developed, it was reviewed and validated by colleague researchers. The next phases of the SLR are to **conduct the review** and to **report the results**. The explanation on how the SLR was to be conducted are reported in the review protocol in Section 4.1.2, the results of the data extraction phase and the synthesis are presented in next section. The reporting phase, that is, reporting the results of the SLR, is composed of this whole master's thesis, as it aims at unveiling all the ETL services that were identified through the undertaken SLR.

5 Results

5.1 Conducting The Review

Following the two search strategies presented in our review protocol in Section 4.1.2, we identified 40 papers to review. Three of them were found to be "low quality" papers and therefore discarded. The others were assessed as "high quality" researches and met all inclusion criteria.

A trial of forward snowballing was conducted on a random sample of retrieved papers to examine the assumption proposed by [Jalali and Wohlin, 2012] regarding the redundancy of forward snowballing. The trial failed to identify any new relevant papers, leading to the termination of the paper retrieval process.

After having composed our research set, we proceeded with the data extraction process of the relevant information identified in the previous section. The resulting *Paper Matrix*, which holds metadata regarding the researches, is presented in Table 5.1. In addition, we compiled the *ETL Service Matrix* from the ETL services discussed in the reviewed researches. This service matrix is presented in Table 5.2, Table 5.3, and Table 5.4. These tables facilitated the answer to our research question by presenting a comprehensive matrix that encompasses all the ETL services extracted from our SLR. It is worth mentioning that some of the services presented in the matrix are derived from those mentioned in the reviewed studies. For example, one paper may mention "connection to external data sources," and we derived subsequent services such as "request access to the external source" and eventually "connect to the external source." Another example is the service "identify source type," which is intuitively derived from "connect to external data source," as the system must understand whether a source is internal or external to manage the connection in an adequate manner. The services extracted from the SLR and presented in the above-mentioned tables are elucidated in detail in the subsequent sections.

ID	Title	Type	Year	Author(s)	Cites
1	A Comparative Review of Data Warehousing ETL Tools with New Trends and Industry Insight	Conference	2017	Mukherjee R., Kar P.	19
2	An overview and implementation of extraction-transformation-loading (ETL) process in data warehouse (Case study: Department of agriculture)	Conference	2015	Rahmadi W., Bambang P.	5
3	An ETL Services Framework Based on Metadata	Conference	2010	Wang H., Ye Z.	10
4	Efficient incremental loading in ETL processing for real-time data integration	Journal	2020	Biswas N., Sarkar A., Mondal K. C.	13
5	From conceptual design to performance optimization of ETL workflows: current state of research and open problems	Journal	2017	Ali S. M., Wrembler R.	28
6	An approach to conceptual modelling of ETL processes	Conference	2014	Dupor S., Jovanic V.	1
7	Design and realization of an ETL method in business intelligence project	Conference	2018	Pan B., Zhang G., Qin X.	8
8	A Review on Traditional ETL Process for Better Approach in Business Intelligence	Conference	2018	Vuka E., Petritaj O.	1
9	A Taxonomy of ETL Activities	Conference	2009	Vassiliadis P., Simitsis A., Baikousi E.	41
10	Towards a Programmable Semantic Extract-Transform-Load Framework for Semantic Data Warehouses	Conference	2015	Deb Nath R. P., Hose K., Pedersen T. B.	34
11	pygrametl: a powerful programming framework for extract-transform-load programmers	Conference	2009	Pedersen T. B., Thomsen C.	32
12	A Survey of Extract-Transform-Load Technology	Journal	2009	Vassiliadis P.	NA
13	ELTA: New Approach in Designing Business Intelligence Solutions in Era of Big Data	Conference	2014	Marìn-Ortega P. M., Dmitriyevb V., Abilovb M., Marx Gmezb J.	0
14	On-Demand ELT Architecture for Right-Time BI: Extending the Vision	Journal	2013	Waas F., Wrembel R., Freudenreich T., Thiele M., Koncilia C., Furtado P.	NA
15	A Comparative Review of Extraction, Tranqformation, and Loading Tools	Journal	2013	Amanpartap Singh P., Khaira J. S.	NA
16	Towards Generating ETL Processes for Incremental Loading	Conference	2008	Jörg T., Dessloch S.	36
17	Formalizing ETL Jobs for Incremental Loading of Data Warehouses	Journal	2009	Jörg T., Dessloch S.	NA
18	Optimized incremental ETL jobs for maintaining data warehouses	Conference	2010	Behrend A., Jörg T.	12
19	Easy and Effective Parallel Programmable ETL	Conference	2011	Thomsen C., Pedersen T. B.	16
20	A UML Based Approach for Modeling ETL Processes in Data Warehouses	Conference	2003	Trujillo J., Luján-Mora S.	96
21	Designing ETL Processes Using Semantic Web Technologies	Conference	2006	Skoutas D., Simitsis A.	65
22	Ontology-based Conceptual Design of ETL Processes for Both Structured and Semi-Structured Data	Journal	2007	Skoutas D., Simitsis A.	NA
23	Conceptual Modeling For ETL Processes	Conference	2002	Vassiliadis P., Simitsis A., Skiadopoulos S.	218

Table 5.1 continued from previous page

24	A Framework for the Design of ETL Scenarios	Conference	2003	Vassiliadis P., Simitsis A., Georgantias P., Terrovitis M.	24
25	QETL: An Approach to On-Demand ETL from Non-Owned Data Sources	Journal	2017	Baldacci L., Golfarelli M., Graziani S., Rizzi S.	13
26	Data Cleaning: Problems and Current Approaches	Journal	2000	Rahm E., Do H. H.	NA
27	ETL Life Cycle	Journal	2015	Bindal P., Khurana P.	NA
28	Managing ETL Processes	Journal	2008	Albrecht A., Nauman F.	NA
29	A Generic and Customizable Frameworkfor the Design of ETL Scenarios	Journal	2005	Vassiliadis P., Simitsis A., Georgantias P., Terrovitis M., Skiadopoulos S.	106
30	BIcenter: A Collaborative Web ETL Solution Based on a Reflective Software Approach	Journal	2021	Almeida J. R., Coelho L., Oliveira J. L.	4
31	A Method For Modeling and Organizing ETL Processes	Conference	2012	Kabiri A., Chiadmi D.	13
32	A New Tool for ETL Processes	Conference	2012	Zuyi C., Taixiang Z.	1
33	A Web-Based ETL Tool for Data Integration Process	Conference	2013	Vijayendra N., Lu M.	1
34	Data Integration and ETL: A Theoretical Perspective	Conference	2021	Sreemathy J., Naveen Durai K., Lakshmi Priya E., Deebika R., Suganthi K., Aisshrawa P. T.	1
35	ETL Development using Patterns - A Service-Oriented Approach	Conference	2019	Oliveira B.,Oliveira Ó., Santos V., Belo O.	3
36	Overview of ETL Tools and Talend-Data Integration	Conference	2021	Sreemathy J., Brindha R., Selva Nagalakshmi M., Suvekha N., Karthick Ragul N., Praveennandha M.	3
37	Using BPMN for ETL Conceptual Modeling: A Case Study	Conference	2021	Oliveira B., Oliveira Ó., Belo O.	2

Table 5.1: Final paper matrix containing metadata about the reviewed studies.

Paper ID	ETL Services																								
	Detect Data Sources	Identify Source Type	Request Access to External Source	Connect to External Sources	Manage Database Credentials	Connect to Internal Source	Identify Source Format	Apply Wrapper	Detect Change (CDC Techniques)	Handle SCD	Perform Loss-Proof Encoding Conversion	Perform Full-Load	Perform Incremental Load	Generate and Assign Surrogate Key	Consult Metadata	Detect Illegal/Erroneous Values	Detect Violated Attribute Dependencies	Detect Referential Integrity Violation	Detect Uniqueness/PK Violation	Perform Lookup	Detect Domain Mismatches	Detect Data Inconsistencies	Detect Business Rules Violation	Fix/Remove Incorrect Data	Handle Missing Data
1		x	x	x											x										x
2	x	x	x	x						x					x			x			x	x			
3					x	x																			
4							x			x															
5														x				x							x
6							x			x		x													
7	x				x	x															x	x			x
8									x				x												
9																				x					
10																		x							
11										x										x					
12							x		x			x	x					x		x			x		x
13																x								x	x
14																x					x	x		x	x
15					x	x			x				x												
16									x			x	x				x	x		x			x		
17									x	x		x	x												
18																				x					
19					x	x	x			x		x								x					
20	x				x	x		x				x		x		x				x	x	x		x	x
21	x																								x
22	x	x	x	x			x					x													
23									x	x	x		x	x					x		x				x
24	x				x	x			x		x		x	x				x	x	x	x				x
25		x	x	x			x																		x
26								x							x			x			x	x			x
27									x	x			x	x		x		x	x	x	x	x	x	x	x
28																				x					
29	x								x				x	x				x	x	x	x		x		x
30	x						x									x								x	
31	x	x	x	x								x						x		x	x	x			x
32					x	x						x			x			x	x						x
33																									
34							x					x	x												
35									x	x				x											
36								x																	
37									x	x				x											

Table 5.2: Service Matrix containing the ETL services discussed by papers (Part I).

Paper ID	ETL Services																									
	Convert Data Format	Convert Data Type	Handle Duplicates	Handle Misspellings	Sort Data	Filter Data	Remove Column	Merge Datasets	Pivot/Unpivot Table	Use Conditions to Select Column	Retrieve Unique Values	Use Sampling Techniques	Determine Transformation to Apply	Use Condition to Apply Transformation	Parse Free-Format Attributes	Perform Usual String Operations	Rename Attribute	Determine Attribute To Generate	Generate Current Date/Time	Generate a Constant Value	Compute Traditional Aggregation Functions	Use User-Defined Function	Use Condition To Generate Attribute's Values	Compute Summaries of Data	Select Target Loading Location
1			x	x		x															x				
2			x	x									x								x				
3																									
4			x		x			x																	
5	x		x	x		x		x													x			x	
6						x		x	x					x							x		x		
7				x	x	x																			
8																									
9	x				x	x		x	x			x		x		x					x	x	x		
10			x				x	x		x							x	x							
11																									
12	x	x	x	x	x	x		x	x					x				x			x	x	x	x	
13																									
14			x			x																		x	
15								x	x																
16			x			x		x																	
17	x					x		x							x		x					x			
18			x		x			x							x						x	x			
19																									
20	x					x		x								x		x		x	x				x
21	x		x			x		x					x					x			x	x			
22	x		x			x		x					x								x				
23	x		x		x	x	x	x		x	x								x		x	x			
24					x	x	x	x		x	x										x	x			
25						x	x			x											x				
26	x		x	x	x	x		x							x	x					x	x			
27	x		x					x																	
28			x					x													x				
29	x				x			x			x										x	x			
30					x																x				
31	x	x	x	x	x	x		x							x										
32			x			x	x	x		x															
33		x	x					x								x		x							
34			x		x									x									x		
35	x																								
36						x		x																	
37								x																	

Table 5.3: Service Matrix containing the ETL services discussed by papers (Part II).

Paper ID	ETL Services					
	Use Conditional Routers	Specify Type of Loading	Check Target Database Constraints	Convert Encoding Format	Manage File/Data Transfer Operations	Perform Full/Incremental Loading
1						
2						
3						
4						
5						
6		x				x
7						
8						x
9	x					
10						
11						
12		x	x			x
13						
14						
15						
16	x		x			x
17						x
18						
19		x				x
20		x				x
21			x			
22		x				x
23				x	x	x
24				x	x	x
25						
26						
27						x
28						
29	x				x	x
30						
31						x
32						x
33						
34		x				x
35						
36						
37						

Table 5.4: Service Matrix containing the ETL services discussed by papers (Part III).

5.2 A Taxonomy of ETL Services

In order to report the outcomes of our SLR and the many services we identified, we decided to plot them on the well-known taxonomy of Extract-Transform-Load presented in Section 2. However, we found this taxonomy over-simplified to reach the level of granularity we wanted to achieve in this research. To handle this issue, we decided to enrich the ETL taxonomy by adding some components. We mainly drilled-down the "transform" phase to divide it into two phases: data validation and data transformation, that were in turn expanded as presented in Section 5.2.2. The identified sub-components of the ETL processes and their services are presented in details in the following sections.

5.2.1 Data Extraction

The initial phase of any ETL workflow involves the **extraction of data from its sources**. This phase aims to selectively retrieve relevant data from diverse sources and subsequently load them into the staging data store, where further transformations can be implemented (as illustrated in Figure 2.2).

The initial step in the extraction process is the **identification of relevant data sources** from which data is to be extracted. It is crucial to **determine the type**

of these sources, whether internal or external, **and the format** in which they are presented, whether structured or semi-structured/unstructured. These characteristics can indeed impact the methodology employed to extract the data. To obtain access to external data sources, **permission** to access the source system hosting the data **must be granted**. In the case of both internal and external sources, a system must be established to **manage the credentials and connection information** needed to connect to the data sources' systems. The format of the data sources is also significant, particularly in the case of semi-structured and unstructured files such as JSON or XML, mails, and texts. Due to their non-tabular nature, these types of files are challenging to handle, necessitating the **application of wrapper mechanisms** to process the data sources as structured tabular data.

After ensuring that all data are available in a consistent and integrated manner, they can be loaded into the SA of the DW. The initial data load is referred to as a **"full load"**, whereby all data contained in the source systems are transferred from their sources to the SA. After this first full load, to ensure and maintain the timeliness of the DW, recurring **"incremental loads"** are subsequently executed, which only refresh the data that have changed or are new. To facilitate the selective loading of the changed/new data, a **Change Data Capture** (CDC) mechanism must be implemented. Various methods for handling CDC in BI systems are presented in [Biswas et al., 2020]. For example, the transaction logs maintained by the source systems can be scraped to retrieve a history of database changes. Alternatively, an "audit column" may be used to store the last date on which data were loaded, allowing for the selective loading of data from source systems that have been added or updated since that date.
In all cases, whether performing a full or incremental load, a **conversion** operation should be taken into consideration to address differences in the encoding formats used by the source systems and the staging area/DW, such as the conversion from ASCII to UTF-8. The converter employed should **guarantee that no data is lost** during the conversion process.

An additional crucial service to consider during the extraction phase of the ETL process is the handling of Slowly Changing Dimensions (SCDs), which are dimensions that can undergo changes over time [Vaisman and Zimányi, 2014]. Such changes may manifest as modifications in the value of an attribute (e.g., a product label change) or in the attributes of a dimension (e.g., the decision to record marital status information for workers, which was previously omitted). Given the significance of keeping track of changes in dimensions in the context of BI systems, SCDs must be handled with care. Vaisman and Zimanyì [Vaisman and Zimányi, 2014] propose various techniques to address SCDs, including overwriting the attribute value (type I) or versioning the dimension table using validity attributes (type II).
To address SCDs in the ETL process, it is essential to specify how changes in dimension data will be reflected in the DW – i.e., type of SCD. To detect changes, the same CDC mechanisms introduced by Biswas et al. [Biswas et al., 2020] may be utilized.

Finally, in situations where a table lacks a natural identifier or where the natural identifier is prone to change or not guaranteed to be unique, the **generation and assignment of a surrogate key** – an artificially created unique key – can be deemed necessary to provide a unique identifier in the data.

The graph presented in Figure 5.1 presents the above-mentioned services as a function of the number of occurrences of each of them.

Based on the given graph, it is evident that the majority of the attention in the

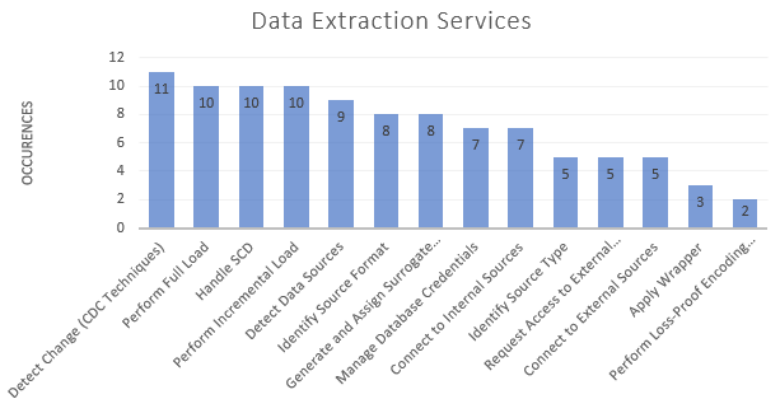

Figure 5.1: Histogram presentation of the data extraction services and their occurrences in the reviewed researches.

extraction phase of the ETL literature is focused on change management. Approximately one quarter of the reviewed papers address change detection to ensure continuous efficient incremental loading. The same proportion applies to papers discussing the handling of SCDs, which is indeed a crucial topic in ETL research since their proper management is essential for accurate historization of the data and their historical values.

Then, the graph shows that the identification of the relevant data sources and their format is also highly discussed in ETL papers. It will be interesting in an attempt for automation to have knowledge on how these tasks are performed. These are indeed tasks that are predominantly undertaken by business experts, who decide which data sources are to be used and then identify their format in order to apply the right extraction mechanism. These tasks will therefore be challenging to automate but the significant amount of literature on the subject is reassuring.

Furthermore, ETL literature also emphasizes source detection and format identification, as these tasks are predominantly performed by humans. Business experts usually decide which database to use and understand its format in advance, allowing them to apply the correct extraction mechanisms. This is relevant in the context of automation trials, and the significant amount of literature on the subject is reassuring.

On the other hand, few papers are dedicated to the application of wrapper mechanisms. It is surprising given the increase in the number of semi- or unstructured files being stored in our the current "big data world". It is indeed important, in a centralization purpose, to use mechanisms that facilitate integrated data management from all sources. This lack of interest in these mechanisms in ETL processes indicate a gap between academic literature and current business practices since many integration tools now support automated application of wrapper mechanisms, enabling for example the transformation of JSON files to tabular data.

Finally, encoding concerns are given less attention than other services, despite the fact that errors in encoding format can lead to data loss or decoding errors. For instance, transitioning from ASCII encoding to UTF-8 encoding is feasible, but the reverse is not always true.

5.2.2 Data Quality, Validation, and Transformation

After the data have been loaded in the SA, they are made available for transformation.

In the transformation step, two processes are performed concurrently: **data validation** and **data transformation**.

The aim of data validation is to ensure high data quality by conducting multiple checks during both the loading of data from the source to the SA and their subsequent transformation, and a consistent shape to answer business needs. The initial stage of data validation involves the execution of multiple **checks** on the data.

First, **single-source validation** checks, where quality checks are applied on data emanating from one single source [Rahm et al., 2000]. The data will be evaluated in order to **detect illegal/erroneous values** – ensuring so-called domain integrity. Then, we verify that **no attribute dependency has been violated**, meaning that the relationship between the values of several field is holding. For example, we check that the age contained in the age field is consistent with the date of birth contained in another field. To know whether or not a value is illegal, or which attribute are related, we have to **access the metadata** contained in the metadata container of the DW because it is where integrity rules are stored. Another check is also performed at the source level to **avoid referential integrity violation** – prevent an attribute to reference an attribute value that does not exist. Lastly, we check that there is **no uniqueness/primary key violation**, meaning that unique attributes are not duplicated in the data source.
The primary goal of single-source checks is to ensure the accuracy and consistency of data from each source before integrating them with data from other sources.

We then proceed to perform **multi-source validation** (MSV), a critical step aimed at verifying data that originate from multiple sources but are intended to be integrated. The primary objective of this step is to ensure the absence of data contradictions among disparate sources and to ensure the coherence of business objects represented in different ways.

To accomplish this, we utilize **Lookup** services to retrieve relevant information about a business object stored in one table that may be available in another table. For example, a Lookup service may be used to check the existence of a given customer or product in one database against its presence in another. This service plays a crucial role in facilitating inter-source verification.

An initial and critical aspect of the validation phase is to identify any **domain mismatches** that may arise due to the differing representation of semantically identical business objects across various data sources. For instance, consider the example discussed in Section 2, where the customer's address is represented as three separate attributes (i.e., street, number, and zipcode) in one database, while in another database, it is represented by a single attribute containing the entire address. Identifying such mismatches is essential to enable the mapping of sources and ensure that sources that were initially incompatible can eventually be integrated.

In the MSV procedure, a second validation step is undertaken to identify **data inconsistencies** that may arise from disparities in information across multiple data sources. Data inconsistencies materialize when information pertaining to a given business entity exhibit discrepancies from one data source to another. To illustrate, in the case of customer addresses, inconsistencies may emerge if the address information of a particular customer appears dissimilar between two distinct data sources.

The final set of assessments performed on the data sources entails ascertaining compliance with expert-defined **business rules** that reside within the metadata repository. For example, in the event that an attribute TURNOVER is derived during the transformation stage, a business rule may specify that it should be calculated as PRICE * VOLUME

SOLD. Therefore, while executing the ETL process for computing the attribute, the system ought to validate this rule.

The graph in Figure 5.2 summarizes the ETL services that should be considered in the "check" step of the data validation phase and presents the number of occurences of each of these services in the reviewed researches.

The second phase of data validation is focused on **data refinement**, which encompasses all transformations aimed at improving the quality and accuracy of the data. This step logically follows from the preceding data validation phase, since the observations generated therein inform the identification of the pertinent transformations to be employed.

During the refinement phase, the initial step involves the implementation of **data cleaning** procedures. The primary aim of this phase is to correct any potential violations that may have been detected in the previous step. To achieve this goal, **the incorrect data identified through the applied checks must be corrected or eliminated**. Subsequently, the focus shifts towards handling **missing data**. There are several approaches to handle missing data, which depend on the type of attribute under consideration. For numerical attributes, imputation mechanisms such as mean substitution, regression substitution, or multiple imputation techniques can be employed [Patrician, 2002]. In contrast, mathematical imputation techniques cannot be used for textual attributes. Instead, machine learning models that use probabilistic approaches to impute the most likely value based on the most similar other instances can be used [Rekatsinas et al., 2017]. Another alternative is to use functional dependencies to reveal the most probable textual value [Breve et al., 2022]. However, in both cases, the decision of simply removing instances containing missing data is available but this might lead to a loss of insightful data [Patrician, 2002].

Another service that must be considered during the cleaning phase is the **conversion of data formats**, especially in the presence of domain mismatches. When facing semantically similar objects represented in various way across different databases, it is crucial to ensure an integrated approach to working with them by transforming their format. The result is a uniform representation of the objects, regardless of their source databases. Data format conversion can apply to the address example we discussed above, but also to date formatting, currency conversion, etc. In addition to data format conversion, the **conversion of data types** is another essential service that must be taken into account, particularly in the presence of domain mismatches. For instance, in some databases, numerical values may have been encoded as textual values, requiring their conversion to the former data type to enable proper functionality when aggregating values, computing derived values, and other relevant operations.

Once the necessary conversion services have been performed, the next crucial step is to identify and effectively **manage duplicate values**. In most cases, when dealing with duplicate instances that share the same identifier, only one of them is kept, and the others are discarded [Biswas et al., 2020, Ali and Wrembel, 2017]. However, in the context of SCDs, duplicate rows may exist for the same identifier, representing a change in the value of an attribute for a specific object. In such cases, it is essential to apply appropriate SCD mechanisms, such as the one mentioned above, to handle the duplicates effectively. The final step in data cleaning is to **manage misspellings** in instances values. Natural language processing (NLP) algorithms exist that aim at detecting and correcting spelling errors in databases, see [Zamora, 1980, Nagata, 1996, Kim et al., 2021] for a presentation of some accepted techniques.

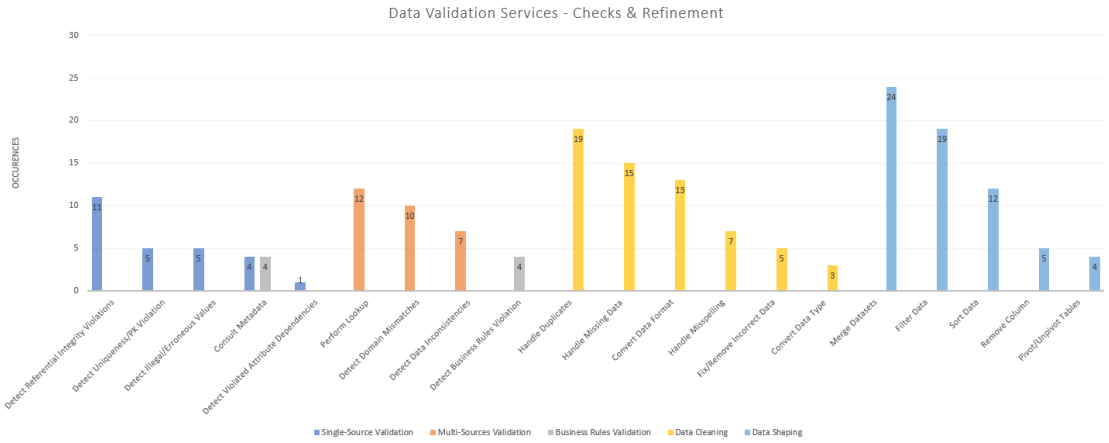

Figure 5.2: Histogram presentation of the data validation services and their occurrences in the reviewed researches.

The second step in the data refinement phase is to **shape the data**. Once the data have been cleaned and that they have been assessed as high quality, they can be reshaped in well-defined structured datasets to suit the further analyses that are to be made with them. This data shaping phase is critical for ensuring that data analysis and visualization efforts are accurate, efficient, and produce useful results. In this shaping phase, the goal is to give the "right shape" to the data, therefore these will be **sorted** and **filtered** to only select the relevant data for analysis. In this reshaping process, one can also decide to **remove irrelevant columns**. Then, datasets can be **merged** together following the different types of "join" (inner, outer, left, right, full). Finally, **pivoting** can be applied to data table to either "pivot" table – transforming rows into columns to create summary views of data, or "unpivot" table – converting summary views of data, where data has been pivoted into columns, back into a tabular format with rows and columns representing individual observations and their attributes [Vaisman and Zimányi, 2014].

As previously mentioned, **data transformation** is a concurrent process with data validation. This phase involves enriching raw data with the necessary transformations to achieve the final analysis goal. The first step in data transformation is **data selection**, which involves choosing the data to be used for decision-making purposes. It is important to note that there is a distinction between the data shaping phase and the data selection phase. In the former, we work with various raw sources from their respective systems and select the source attributes to be included and enriched in subsequent processes. In contrast, data selection involves choosing the data instances to be included and creating new attributes that are relevant for the final analysis. Data shaping is about shaping the datasets to include in the analysis, data selection is about creating the tables to include in the DW.

During the data selection phase, **conditions** are used to selectively **choose columns** that meet specific criteria. Furthermore, a **lookup** service must be provided to select data that meets certain criteria based on attributes from other tables. For instance, the lookup service could be used to retrieve only customers – from the CUSTOMER dimension table – who have purchased products from a specific category – which is stored in the PRODUCT dimension table. Additionally, a service must be available to **retrieve unique values** that an instance may take for a given attribute. This service is beneficial for computing aggregations for each unique value, such as calculating the average sale

for each product in a particular category. Finally, a service should propose **different data sampling methods**, e.g., row sampling or percentage sampling [Vassiliadis et al., 2009]. This kind of service would allow to retrieve a representative subset of the data for analysis. This can be really interesting in the case of very large datasets where it would be time-consuming or impossible to analyze the data as a whole.

The subsequent phase in the transformation process is the **data modification phase**, where selected data is modified to conform to the needs of decision-makers. The first essential step in this phase is to **determine the appropriate transformations** to apply to the data. **Conditions** are used to selectively **apply these transformations** to data that satisfies certain criteria. All instance-level transformation of textual values is performed in this phase. Firstly, **free-format textual attributes are parsed** to extract relevant data. This service ranges from simple retrieval of date information from string timestamps to the extraction of customer opinions from free-text fields. Text mining techniques may be employed to parse text and extract pertinent information [Justicia De La Torre et al., 2018]. Subsequently, the **standard string operations** are executed, such as converting text to lowercase or uppercase, truncating strings to a specified length, removing white spaces, and so on [Trujillo and Luján-Mora, 2003, Rahm et al., 2000]. Lastly, in this phase, **attributes are renamed**. This process involves assigning more meaningful and relevant names to attributes to facilitate further analysis and understanding of the data.

The final phase of data transformation is the **data creation phase**, where new attributes that were not previously available from the source systems are created. To accomplish this, it is essential to **identify the necessary attributes that need to be generated**. Once identified, various services must be available to compute these attributes. One important service required for generating new attributes is a function to **calculate the current date/time**. This service is necessary, for example, to compute a delay between a certain date and the current date. Another important service is one that **generates a constant value**, which can be used for computations or as a default value for newly created attributes. Additionally, a service **computing the traditional aggregation function** (such as average, sum, count, maximum, and minimum) must be available to compute aggregation from one attribute and use it as the value for another attribute. This is particularly important in the case of a pivot operation. A service that allows the **use of user-defined functions** to compute attribute values must also exist. Both the aggregation functions and the user-defined functions can be used to **generate derived attributes**, i.e., attributes that are computed from the values of other attributes. Moreover, it is necessary to make **use of conditions** to specify the value that the newly created attribute should hold. Finally, a service that **computes summaries** of data should be available at the end of the entire transformation process. This service is crucial to ensure that the creation of attributes and the manipulation of data did not result in any errors or outliers. Furthermore, the summaries can provide insight into the distribution of values, as well as identify patterns and trends, thereby allowing for a final check before loading the data into the data warehouse.

Although the data validation and data transformation phases, including their internal steps, may appear to be sequential and performed one after the other, it is worth noting that these phases are performed concurrently. For instance, the creation of a new attribute using an aggregation function may involve renaming it and verifying its compliance with the integrity constraints and business rules established during the validation phase. Consequently, we have opted to discuss data transformation and validation together in this section since they are highly interrelated and affect one another. This approach

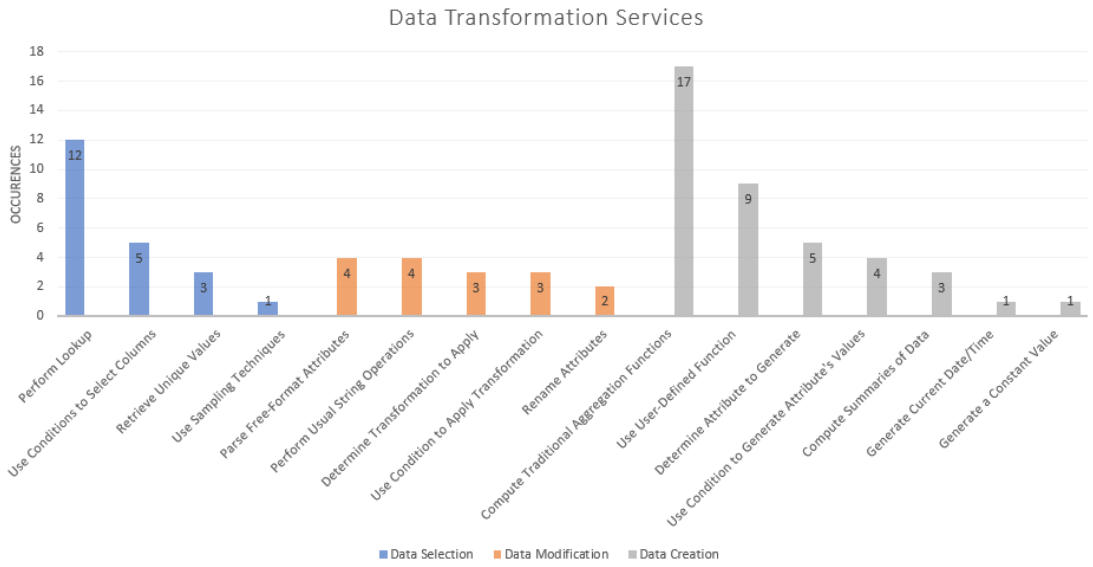

Figure 5.3: Histogram presentation of the data transformation services and their occurrences in the reviewed researches.

highlights the importance of considering the interdependence between data transformation and validation. The success of the data transformation phase is dependent on the quality and integrity of the source data, which are determined during the validation phase. Additionally, the validation phase must take into account any changes made to the data during the transformation phase, which may require additional validation checks.

Figures 5.2 and 5.3 provide a visual representation of the services that are presented in the data validation and transformation phases, along with the frequency of their occurrence in the reviewed literature. These have proved useful for guiding future research. Specifically, the histogram in Figure 5.2 revealed that, in the ETL literature, the majority of attention in the data validation phase is focused on data refinement, rather than on data checks. However, given the importance of good quality data for deriving proper and reliable analysis, there is a clear need to pursue further research in this field.

In the context of validation checks, it is noteworthy that only limited attention has been paid to the detection of violated attribute dependencies. The potential consequences of undetected attribute dependency violations are significant, as they can lead to data inconsistencies and errors, resulting in flawed decision-making processes based on erroneous data in the final data warehouse. The scarcity of literature addressing this issue may be due to the relative simplicity of detecting attribute dependency violations, which involves comparing the values of different fields within a dataset. Concerning our research of interest, as this task is typically performed by business experts, who possess advanced knowledge about the fields to be tested, automating this process can be challenging, and identifying which fields need to be tested can be problematic. As such, further research is necessary to develop automated methods for detecting correlated fields and identifying attribute dependencies.

The detection of business rule violations is another area that has received comparatively little attention in the ETL literature. This may be due, in part, to the fact that business rules are often defined by individual businesses and are not necessarily

generalizable across all ETL processes. Moreover, automating the validation of business rules poses a significant challenge, as there is limited literature devoted to this topic. However, in the business world, several tools and mechanisms have been developed to test data compliance with business rules, such as Oracle's Enterprise Data Quality module [Oracle, 2019], which enables the validation of data compliance with complex business rules. This further highlights the gap between professional practice and academic research in this domain.

Regarding the transformation step as we defined it, Figure 5.3 illustrates that a significant proportion of attention in the scientific literature was directed towards computing conventional aggregation functions and performing lookups, while other aspects were given less attention. This phenomenon can be attributed to the specific nature of this step, which involves selecting, modifying, and creating data, and therefore requires customization to suit the particular application and data being utilized. It will de facto be necessary, in an attempt to achieve a fully automated ETL, to establish a comprehensive taxonomy of these transformation services that clarifies when and how they should be performed according to the requirements of business experts.

5.2.3 Data Loading

The ultimate stage in nearly every ETL process is the loading of the reworked centralized data from the SA to the DW. During this loading phase, a number of essential services must be considered. First, it is necessary to carefully **select the target location for data loading**. In the case of a full load, simply specifying the path to the DW suffices. However, when performing incremental loading, **conditional routers** must be established to direct data to the appropriate existing table in the DW. Additionally, it is imperative to **specify the type of loading** (i.e., full or incremental) and **adjust the loading procedure accordingly**. The system should also **verify the target database constraints** to ensure that the data being loaded adhere to these constraints. Finally, the loading process should include a service that **converts the encoding format** to comply with the format used in the target database, as well as a service to **manage the various data/file transfer operations** (such as FTP protocols, encryption of the data, etc.). The aim of all these services is to ensure a secure and compliant loading of the transformed data.

The data loading services are summarized in Figure 5.4, which presents a histogram depicting the services and their frequency of occurrence in the reviewed literature. The graph reveals that surprisingly, apart from the loading services themselves, only a few studies mention the selection of the target loading location, the routing of data to the correct table, the checking for target database constraints, and the secure loading of data through encoding and transfer operations. It is indeed essential to give due attention to these services, as they are critical for ensuring a secure and compliant transfer of data from the sources to the destination databases.

Speaking of automation however, automating these services should not be a difficult task as they are not dependent on the application being created or on the data being used. With the exception of the conditional routing service, which must be aware of the schema of the data warehouse, these services are relatively static in nature and should not represent a significant challenge to automate.

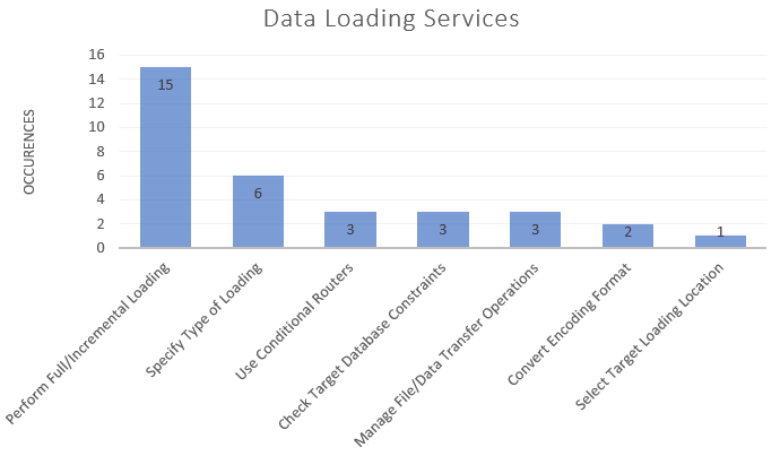

Figure 5.4: Histogram presentation of the data loading services and their occurrences in the reviewed researches.

5.2.4 A Summarized Taxonomy

To conclude, we have presented in this section an extended taxonomy of ETL services. This taxonomy has been divided to represent the ETL sub-processes that are performed during the whole ETL. We began by describing the services that should be considered in the data extraction phase where data are extracted from sources to the staging area. Then, we addressed the services related to two concurrent steps, namely data validation and data transformation. The former was divided into two phases: checks and refinement. Data checks were made up of three types of checks: single-source validation (considering one data source), multi-source validation (considering multiple sources) and business rules validation. In the refinement phase, we described the services that have to be included to clean and shape data. For data transformation, we discussed the different services related to the three component phases: data selection, data modification and data creation. Finally, we presented the services required for the loading of the data from the SA to the DW. A summary of all the services presented in this taxonomy is presented in Table 5.5.

Next section presents a discussion of the unveiled taxonomy and its potential for added-value. We also discuss limitations of the study alongside prospects for future research that are opened up by the current work.

ETL Services	Data Extraction Services			S1 Detect Data Sources S2 Identify Source Type S3 Request Access to External Sources S4 Connect to External Source S5 Manage Database Credentials S6 Connect to Internal Source S7 Identify Source Format S8 Apply Wrapper S9 Detect Change (CDC Techniques) S10 Handle SCDs S11 Perform Loss-Proof Encoding Conversion S12 Perform Full Load S13 Perform Incremental Load S14 Generate and Assign Surrogate Key
	Data Validation Services	Checks	Single-Source Validation	S15 Consult Metadata S16 Detect Illegal/Eronneous Values S17 Detect Violated Attribute Dependencies S18 Detect Referential Integrity Violation S19 Detect Uniqueness/Primary Key Violation
			Multi-Source Validation	S20 Perform Lookup S21 Detect Domain Mismatches S22 Detect Data Inconsistencies
			Business Rules Validation	S15 Consult Metadata S23 Detect Business Rules Violation
		Data Refinement	Data Cleaning	S24 Fix/Remove Incorrect Data S25 Handle Missing Data S26 Convert Data Format S27 Convert Data Type S28 Handle Duplicates S29 Handle Misspellings
			Data Shaping	S30 Sort Data S31 Filter Data S32 Remove Column S33 Merge Datasets S34 Pivot/Unpivot Tables
	Data Transformation Services	Data Selection		S35 Use Conditions to Select Columns S20 Perform Lookup S36 Retrieve Unique Values S37 Use Sampling Techniques
		Data Modification		S38 Determine Transformation to Apply S39 Use Condition to Apply the Transformation S40 Parse Free-Format Attribute S41 Perform Usual String Operations S42 Rename Attributes
		Data Creation		S43 Determine Attribute to Generate S44 Generate Current Date/Time S45 Generate a Constant Value S46 Compute Traditional Aggregation Functions S47 Use User-Defined Function S48 Use Condition to Generate Attribute's Values S49 Compute Summaries of Data
	Data Loading Services			S50 Select Target Loading Location S51 Use Conditional Routers S52 Specify Type of Loading S53 Check Target Database Constraints S54 Convert Encoding Format S55 Manage File/Data Transfer Operation S56 Perform Full/Incremental Loading

Table 5.5: Unified Taxonomy of ETL process services

6 Discussion and Future Works

The objective of this research was to derive a comprehensive list of the services that need to be considered in any ETL process from the scientific literature. To do so, we undertook a SLR to answer our research question RQ presented in Section 4.1.2. As an answer, we proposed an enriched taxonomy of the usual extract-transform-load phases presented in Table 5.5.

6.1 Discussing the Findings

It was observed in previous section that some services were more often discussed in the ETL literature than others. Specifically, the validation services were found to be the most frequently discussed on average. It was noted that the literature tends to give more importance to the data refinement phase over the check phase, which is somewhat unexpected given that data quality is a critical factor in enabling high-quality and error-free decision-making. In addition, it was found that in comparison with the other categories of services, little attention is devoted to the loading phase, which is a rather surprising finding given that improperly configured loading processes may result in errors and hinder access to critical data.

The preceding section also shed light on the ETL tasks that will certainly be more challenging to automate. Our analysis revealed that certain tasks, such as the detection of relevant data sources, identification of sources' format, handling of misspellings, and specification of the target loading location, are typically performed manually and remain lowly automated. These tasks are often dependent on the specific business domain and the applicable business rules. Consequently, they will require greater attention and further research to derive proper generalizations and establish comprehensive standards that would facilitate compliant and effective automation.

It is also important to mention that the preceding section exclusively focused on ETL services that relate to data integration. However, in modern ETL systems, a proper orchestration mechanism is also crucial. An orchestration system can be defined as a tool supporting the definition, the execution and the monitoring of the ETL services creating data flows [Matskin et al., 2021]. An adequate orchestration system is critical for the management and monitoring of ETL services. While our automation perspective concentrates on automating only the ETL services that relate to data integration, a monitoring system is still highly necessary to enable users to control and oversee the execution of services and their outputs, facilitate error tracing and debugging, and provide a lineage of the data transformation.

The papers we reviewed also discussed some orchestration services that are closely associated with ETL services. We categorized the orchestration services mentioned in the reviewed literature into three categories: operational reporting services, error detection and handling services, and other services.

To begin with, operational reporting services must provide users with a comprehensive view of their data. Two important orchestration services that should be considered are process monitoring and data lineage. The real-time monitoring of ETL activities is frequently discussed in the ETL literature (see [Mukherjee and Kar, 2017, Vassiliadis, 2009, Waas et al., 2013]). Additionally, data lineage possibilities, which allow the tracing of the transformations undergone by the data from source to destination, are also extensively discussed in the ETL literature (see [Mukherjee and Kar, 2017, Dupor

and Jovanović, 2014, Vassiliadis, 2009, Jörg and Dessloch, 2009, Albrecht and Naumann, 2008]).

Several papers also discuss what we have defined as error detection and handling services. This category of services pertains to the management of errors and aims to ensure a failure-proof ETL process. Detecting errors as they arise and enabling users to pre-configure a certain path to route incorrect and error-causing data are of utmost importance in an ETL tool [Wang and Ye, 2010, Vassiliadis, 2009, Vassiliadis et al., 2005, Pan et al., 2018, Oliveira et al., 2019]. Error logging is also a crucial service to consider, as it facilitates troubleshooting and continuous improvement [Wang and Ye, 2010, Biswas et al., 2020, Vassiliadis, 2009, Thomsen and Pedersen, 2011, Vassiliadis et al., 2003].

Finally, other orchestration services that were interesting to mention due to their recurrence were the support for security concerns through user access and for big data peculiarities. Security is indeed a crucial aspect of the ETL process, and policies should be put in place to ensure that only authorized users can access the data being processed and the ETL process itself. For instance, Microsoft Azure's integration component, Azure Data Factory, uses Role-Based Access Control to manage user access to data integration components [Familiar et al., 2017]. Several studies have addressed this security concern [Wijaya and Pudjoatmodjo, 2015, Pan et al., 2018, Almeida et al., 2021]. Another critical consideration is the handling of big data. The increasing volume and velocity of data make it challenging to perform ETL computations on a single machine, resulting in longer processing times that may not meet the needs of the BI system. To address this issue, many organizations are turning to distributed computing to enable the distribution of computation units and achieve faster and more efficient processing of data [Machado et al., 2019, Wang, 2017]. This is especially important for organizations with near-real-time analytics requirements, where data freshness is critical. The challenges of big data are discussed in [Dupor and Jovanović, 2014, Marín-Ortega et al., 2014, Oliveira et al., 2019].

Currently, many of the available ETL tools support the orchestration capabilities that we have mentioned above, such as Microsoft SSIS and Apache NiFi. Therefore, when attempting to build a new fully automated ETL tool, it will be necessary to also consider these orchestration services to ensure high usability potential.

6.2 Potential of the Unveiled Extended Taxonomy

This master's thesis is based on the premise that in order to achieve a fully automated ETL, an initial and essential step is to develop a comprehensive list of all ETL services from existing literature. We realized that this knowledge was available but disseminated across multiple researches, and therefore needed to be gathered. We therefore undertook a SLR to combine the identified papers and derive the needed list, what we achieved in Section 5.2.

The developed taxonomy and the Table 5.5 have successfully addressed the research question of the SLR by presenting a comprehensive list of ETL services that are crucial to consider when designing and developing an ETL solution. This taxonomy provides a useful framework for evaluating the completeness of existing ETL tools and for facilitating the design and development of new ETL systems.

We now have at disposition the complete list of ETL services that was required to achieve the final goal of this paper, that is, to pave the way towards a fully automated ETL process. Moreover, the discussion in Section 5.2 has identified areas that require

further research to automate certain services that are typically performed by humans.

In conclusion, this research has successfully addressed the research question by presenting an expanded taxonomy of ETL services. Furthermore, it has provided guidance for the development of a fully automated ETL system by identifying areas where full automation may be challenging due to the inherent nature of the considered services. Finally, the derived taxonomy can serve as a framework for the development and evaluation of new ETL tools.

6.3 Limitations

Even though we ensure the least potential for bias in our study through the well established objectivity of the SLR protocol, we can think about some possible limitations.

First, the number of papers reviewed might be considered as weak. Usual SLR indeed review more than a hundred papers, whereas in this work we only reviewed 37 papers. This might be due to over-restrictive selection and quality criteria. However, the constraint put on the papers using these criteria ensured the quality of our study. Moreover, we mentioned in the premise of this work that the ETL literature related to BI systems was rather scarce.

Then, although the selection protocol of the SLR was designed to maintain objectivity, the selection process ultimately relies on reviewer decisions, which may introduce selection bias, but to a limited extent. Furthermore, the absence of a peer review process for the selected papers, as well as the review being performed by only one individual, may have contributed to such bias. Nonetheless, any potential bias was minimized by the rigor with which criteria were applied to guide the paper selection process.

A last limitation of this study might be its narrow scope, as it exclusively focuses on the ETL process in the context of BI systems. The ETL process is indeed also used in various other contexts, such as migrating data from one system to another and transferring data between different storage providers. Although it is possible to discuss these contexts, our aim was to ensure consistency in this study as the primary concern was focusing on the automation of the ETL process in a BI context. Additionally, we solely considered traditional BI systems that utilize a DW as the central repository for storing data. However, many organizations are shifting towards using data lakes, for example, as their central storage point for all data. As a result, ETL operations conducted in such contexts should also be taken into consideration in future analyses. Exploring other ETL types is a potential avenue for future research.

6.4 Future Works

As mention in the introduction, this work aimed at defining a complete list of ETL services that would support the creation of fully automated ETL process. During this work, we made several findings.

First, it was observed that certain services were relatively understudied compared to others. However, in order to effectively automate these services, it is imperative to establish a comprehensive understanding of their working, necessitating further research on how these are generally performed.

Then, we identified some ETL services that would pose greater challenges to automation due to their reliance on the expertise of ETL developers. As such, it will be necessary to conduct further research in order to establish standardized procedures depicting the

functioning of these services, that can in turn be automated.

Another important step in our further research involves validating the taxonomy we developed by comparing it with professional ETL tools. This will help us verify the comprehensiveness of our taxonomy and its alignment with the industry practices. Additionally, it will provide an opportunity to identify any ETL services that have not been discussed in the literature, thus contributing towards the advancement and extension of the ETL research.

Our future research will also aim to investigate the most effective ways to automate the ETL process. One potential solution we have considered is the use of software bots, which have been defined by [Storey et al., 2020] as softwares that automate one or more feature, and/or perform one or more function(s) that human may do, and/or interact with humans or other agents. Software bots have already been successfully implemented in various industries, such as customer support chatbots [Storey et al., 2020], "devbots" for supporting developers in software development [Erlenhov et al., 2019], and social bots on social media platforms [Ferrara et al., 2016]. In these contexts, bots were always used for their ability to mimic human behavior to undertake repetitive tasks in an automated way.

Our research suggests that this technology could be well-suited for automating the ETL services we have identified, which are typically routine activities performed by humans. The use of bots has the potential to increase the efficiency and accuracy of these services, reduce the cost of BI projects, facilitate faster processing of large amounts of data, and adapt more easily to changing BI project requirements. This approach is relatively novel, as there has been limited research on using bots to automate BI components. To this end, we will investigate the development of a "bot ecosystem" to automate each of the ETL services we have identified and support the entire ETL process.

7 Conclusion

The primary goal of this master's thesis is to pave the way towards a fully automated ETL process to support an easier and more efficient set up of BI projects in risk-avert companies.

As presented in the introduction, business intelligence projects can be perceived as risky by some organizations due to the time needed to set them up and to the initial investment it requires [Olszak and Ziemba, 2012]. We identified that the BI component that was the most costly and time-consuming was the Extract-Transform-Load component, as it can account for 80% of the effort and 30% of the cost of development [Dupor and Jovanović, 2014, Vassiliadis et al., 2002]. We presented how an automation of this ETL component, responsible for the entire data integration phase, could help decrease the perceived risk of BI projects by decreasing the costs, thanks to lower human implication, and increasing the rate of success, thanks to higher standardization.

To achieve the automation of the ETL process, we made the postulate that standardization of each ETL service was imperative. To attain this standardization, a comprehensive catalog of all relevant ETL services was deemed necessary. We realized that this knowledge was accessible in the scientific literature, but disseminated across different articles. An extensive research effort was consequently necessary to retrieve and consolidate the knowledge from these disparate sources into a centralized knowledge

base.

We therefore conducted a systematic literature review to identify relevant researches pertaining to ETL in BI systems, and to extract the data relating to the ETL services they discussed. The research question we formulated was: "What are all the ETL services mentioned in the scientifi literature on data extration, transformation, and loading applied to Business Intelligence?" The methodology employed to address this question, as detailed in Section 4, identified 37 articles of high quality, which aided in the derivation of the comprehensive taxonomy presented in Section 5.

This taxonomy, as summarized in Table 5.5, contains the complete list of ETL services that was deemed to be essential to answer our research question. During the data extraction and synthesis procedures, we realized that some ETL services were underrepresented in the literature, and that some others would be inherently more difficult to automate due to their reliance on human intervention. The latter category will necessitate extensive research to establish representative standardization protocols to ensure their effective automation.

The ETL services being identified, the next step will be to confront our taxonomy to integration softwares to evaluate its completeness, and, if necessary, to complete it with services that are not discussed in the academic literature. Afterwards, a standardization effort will be initiated for these services, as a precursor to automation. To enable the automated execution of the ETL services in an effective and efficient manner, it will de facto be necessary to establish a framework that delineates the specific application, context, and manner in which each service should be executed.

The final research track will be to investigate how to automate the ETL services. One avenue of exploration that has been preliminarily examined is the use of bots. Specifically, we intend to create a bot ecosystem that emulates the interconnections between ETL services that integrate data from sources to the DW. Given the ability of bots in diverse industries to mimic human behavior when executing routine tasks, coupled with the scarcity of literature that examines bots in BI systems, we have chosen to investigate the potential of this technology in the context of BI projects.

References

[Ain et al., 2019] Ain, N., Vaia, G., DeLone, W. H., and Waheed, M. (2019). Two decades of research on business intelligence system adoption, utilization and success–a systematic literature review. *Decision Support Systems*, 125:113113.

[Albrecht and Naumann, 2008] Albrecht, A. and Naumann, F. (2008). Managing etl processes. *NTII*, 8(2008):12–15.

[Ali and Wrembel, 2017] Ali, S. M. F. and Wrembel, R. (2017). From conceptual design to performance optimization of etl workflows: current state of research and open problems. *The VLDB Journal*, 26(6):777–801.

[Almeida et al., 2021] Almeida, J. R., Coelho, L., and Oliveira, J. L. (2021). Bicenter: A collaborative web etl solution based on a reflective software approach. *SoftwareX*, 16:100892.

[Alpar and Schulz, 2016] Alpar, P. and Schulz, M. (2016). Self-service business intelligence. *Business & Information Systems Engineering*, 58:151–155.

[Biswas et al., 2020] Biswas, N., Sarkar, A., and Mondal, K. C. (2020). Efficient incremental loading in etl processing for real-time data integration. *Innovations in Systems and Software Engineering*, 16:53–61.

[Böhm et al., 2009] Böhm, M., Wloka, U., Habich, D., and Lehner, W. (2009). Gcip: Exploiting the generation and optimization of integration processes. In *Proceedings of the 12th International Conference on Extending Database Technology: Advances in Database Technology*, pages 1128–1131.

[Breve et al., 2022] Breve, B., Caruccio, L., Deufemia, V., and Polese, G. (2022). Renuver: A missing value imputation algorithm based on relaxed functional dependencies. In *EDBT*, pages 1–52.

[Bucher et al., 2009] Bucher, T., Gericke, A., and Sigg, S. (2009). Process-centric business intelligence. *Business Process Management Journal*, 15(3):408–429.

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

[Chávez and Li, 2011] Chávez, J. V. and Li, X. (2011). Ontology based etl process for creation of ontological data warehouse. In *2011 8th International Conference on Electrical Engineering, Computing Science and Automatic Control*, pages 1–6. IEEE.

[Dash and Swayamsiddha, 2022] Dash, B. and Swayamsiddha, S. (2022). Reverse etl for improved scalability, observability, and performance of modern operational analytics-a comparative review. In *2022 OITS International Conference on Information Technology (OCIT)*, pages 491–494. IEEE.

[Dupor and Jovanović, 2014] Dupor, S. and Jovanović, V. (2014). An approach to conceptual modelling of etl processes. In *2014 37th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, pages 1485–1490. IEEE.

[El-Sappagh et al., 2011] El-Sappagh, S. H. A., Hendawi, A. M. A., and El Bastawissy, A. H. (2011). A proposed model for data warehouse etl processes. *Journal of King Saud University-Computer and Information Sciences*, 23(2):91–104.

[Erlenhov et al., 2019] Erlenhov, L., de Oliveira Neto, F. G., Scandariato, R., and Leitner, P. (2019). Current and future bots in software development. In *2019 IEEE/ACM 1st International Workshop on Bots in Software Engineering (BotSE)*, pages 7–11. IEEE.

[Faisal and Sarwar, 2014] Faisal, S. and Sarwar, M. (2014). Handling slowly changing dimensions in data warehouses. *Journal of Systems and Software*, 94:151–160.

[Familiar et al., 2017] Familiar, B., Barnes, J., Familiar, B., and Barnes, J. (2017). Batch processing with data factory and data lake store. *Business in Real-Time Using Azure IoT and Cortana Intelligence Suite: Driving Your Digital Transformation*, pages 227–290.

[Ferrara et al., 2016] Ferrara, E., Varol, O., Davis, C., Menczer, F., and Flammini, A. (2016). The rise of social bots. *Communications of the ACM*, 59(7):96–104.

[Giorgini et al., 2003] Giorgini, P., Mylopoulos, J., Nicchiarelli, E., and Sebastiani, R. (2003). Reasoning with goal models. In *Conceptual Modeling—ER 2002: 21st International Conference on Conceptual Modeling Tampere, Finland, October 7–11, 2002 Proceedings 21*, pages 167–181. Springer.

[Gupta and Jain, 2013] Gupta, S. and Jain, S. K. (2013). A literature review of lean manufacturing. *International Journal of Management Science and Engineering Management*, 8(4):241–249.

[Habibu, 2013] Habibu, T. (2013). Parallel data analytics for business intelligence real-time online analytical processing (olap) for multi-core and cloud architectures.

[Happel and Seedorf, 2006] Happel, H.-J. and Seedorf, S. (2006). Applications of ontologies in software engineering. In *Proc. of Workshop on Sematic Web Enabled Software Engineering"(SWESE) on the ISWC*, pages 5–9. Citeseer.

[Inmon, 2005] Inmon, W. H. (2005). *Building the data warehouse*. John wiley & sons.

[Inmon and Kelley, 1993] Inmon, W. H. and Kelley, C. (1993). *Rdb/VMS: Developing the data warehouse*. John Wiley & Sons, Inc.

[Jalali and Wohlin, 2012] Jalali, S. and Wohlin, C. (2012). Systematic literature studies: database searches vs. backward snowballing. In *Proceedings of the ACM-IEEE international symposium on Empirical software engineering and measurement*, pages 29–38.

[Jörg and Dessloch, 2009] Jörg, T. and Dessloch, S. (2009). Formalizing etl jobs for incremental loading of data warehouses. *Datenbanksysteme in Business, Technologie und Web (BTW)–13. Fachtagung des GI-Fachbereichs" Datenbanken und Informationssysteme"(DBIS)*.

[Justicia De La Torre et al., 2018] Justicia De La Torre, C., Sánchez, D., Blanco, I., and Martín-Bautista, M. J. (2018). Text mining: techniques, applications, and challenges. *International journal of uncertainty, fuzziness and knowledge-based systems*, 26(04):553–582.

[Kherdekar and Metkewar, 2016] Kherdekar, V. A. and Metkewar, P. S. (2016). A technical comprehensive survey of etl tools. *International Journal of Applied Engineering Research*, 11(4):2557–2559.

[Khine and Wang, 2018] Khine, P. P. and Wang, Z. S. (2018). Data lake: a new ideology in big data era. In *ITM web of conferences*, volume 17, page 03025. EDP Sciences.

[Kim et al., 2021] Kim, T., Han, S. W., Kang, M., Lee, S. H., Kim, J.-H., Joo, H. J., and Sohn, J. W. (2021). Similarity-based unsupervised spelling correction using biowordvec: development and usability study of bacterial culture and antimicrobial susceptibility reports. *JMIR Medical Informatics*, 9(2):e25530.

[Kitchenham, 2004] Kitchenham, B. (2004). Procedures for performing systematic reviews. *Keele, UK, Keele University*, 33(2004):1–26.

[Machado et al., 2019] Machado, G. V., Cunha, Í., Pereira, A. C., and Oliveira, L. B. (2019). Dod-etl: distributed on-demand etl for near real-time business intelligence. *Journal of Internet Services and Applications*, 10:1–15.

[Mack et al., 2015] Mack, O., Khare, A., Krämer, A., and Burgartz, T. (2015). *Managing in a VUCA World*. Springer.

[Marín-Ortega et al., 2014] Marín-Ortega, P. M., Dmitriyev, V., Abilov, M., and Gómez, J. M. (2014). Elta: new approach in designing business intelligence solutions in era of big data. *Procedia technology*, 16:667–674.

[Matskin et al., 2021] Matskin, M., Tahmasebi, S., Layegh, A., Payberah, A. H., Thomas, A., Nikolov, N., and Roman, D. (2021). A survey of big data pipeline orchestration tools from the perspective of the datacloud project. In *Proc. 23rd Int. Conf. Data Analytics Management Data Intensive Domains (DAMDID/RCDL 2021)*, pages 63–78.

[Michalczyk et al., 2020] Michalczyk, S., Nadj, M., Azarfar, D., Maedche, A., and Gröger, C. (2020). A state-of-the-art overview and future research avenues of self-service business intelligence and analytics.

[Mondal et al., 2020] Mondal, K. C., Biswas, N., and Saha, S. (2020). Role of machine learning in etl automation. In *Proceedings of the 21st international conference on distributed computing and networking*, pages 1–6.

[Mukherjee and Kar, 2017] Mukherjee, R. and Kar, P. (2017). A comparative review of data warehousing etl tools with new trends and industry insight. In *2017 IEEE 7th International Advance Computing Conference (IACC)*, pages 943–948. IEEE.

[Muñoz et al., 2009] Muñoz, L., Mazón, J.-N., and Trujillo, J. (2009). Automatic generation of etl processes from conceptual models. In *Proceedings of the ACM twelfth international workshop on Data warehousing and OLAP*, pages 33–40.

[Nagata, 1996] Nagata, M. (1996). Context-based spelling correction for japanese ocr. In *COLING 1996 Volume 2: The 16th International Conference on Computational Linguistics*.

[Negash, 2004] Negash, S. (2004). Business intelligence. *Communications of the association for information systems*, 13(1):15.

[Nwokeji et al., 2018] Nwokeji, J. C., Aqlan, F., Anugu, A., and Olagunju, A. (2018). Big data etl implementation approaches: A systematic literature review (p). In *SEKE*, pages 714–713.

[Oliveira et al., 2019] Oliveira, B., Oliveira, Ó., Santos, V., and Belo, O. (2019). Etl development using patterns: A service-oriented approach. In *ICEIS (1)*, pages 216–222.

[Olszak and Ziemba, 2012] Olszak, C. M. and Ziemba, E. (2012). Critical success factors for implementing business intelligence systems in small and medium enterprises on the example of upper silesia, poland. *Interdisciplinary Journal of Information, Knowledge, and Management*, 7:129.

[Oracle, 2019] Oracle, C. (2019). Oracle enterprise data quality product family.

[Pan et al., 2018] Pan, B., Zhang, G., and Qin, X. (2018). Design and realization of an etl method in business intelligence project. In *2018 IEEE 3rd International Conference on Cloud Computing and Big Data Analysis (ICCCBDA)*, pages 275–279. IEEE.

[Pastor et al., 2008] Pastor, O., España, S., Panach, J. I., and Aquino, N. (2008). Model-driven development. *Informatik-Spektrum*, 31:394–407.

[Patrician, 2002] Patrician, P. A. (2002). Multiple imputation for missing data. *Research in nursing & health*, 25(1):76–84.

[Petrović et al., 2017] Petrović, M., Vučković, M., Turajlić, N., Babarogić, S., Aničić, N., and Marjanović, Z. (2017). Automating etl processes using the domain-specific modeling approach. *Information Systems and e-Business Management*, 15:425–460.

[Radhakrishna et al., 2012] Radhakrishna, V., SravanKiran, V., and Ravikiran, K. (2012). Automating etl process with scripting technology. In *2012 Nirma University International Conference on Engineering (NUiCONE)*, pages 1–4. IEEE.

[Rahm et al., 2000] Rahm, E., Do, H. H., et al. (2000). Data cleaning: Problems and current approaches. *IEEE Data Eng. Bull.*, 23(4):3–13.

[Rekatsinas et al., 2017] Rekatsinas, T., Chu, X., Ilyas, I. F., and Ré, C. (2017). Holoclean: Holistic data repairs with probabilistic inference. *arXiv preprint arXiv:1702.00820*.

[Rodic and Baranovic, 2009] Rodic, J. and Baranovic, M. (2009). Generating data quality rules and integration into etl process. In *Proceedings of the ACM twelfth international workshop on Data warehousing and OLAP*, pages 65–72.

[Storey et al., 2020] Storey, M.-A., Serebrenik, A., Rosé, C. P., Zimmermann, T., and Herbsleb, J. D. (2020). Botse: Bots in software engineering (dagstuhl seminar 19471). In *Dagstuhl Reports*, volume 9. Schloss Dagstuhl-Leibniz-Zentrum für Informatik.

[Theodorou et al., 2014] Theodorou, V., Abelló, A., Thiele, M., and Lehner, W. (2014). A framework for user-centered declarative etl. In *Proceedings of the 17th international workshop on data warehousing and OLAP*, pages 67–70.

[Thomsen and Pedersen, 2011] Thomsen, C. and Pedersen, T. B. (2011). Easy and effective parallel programmable etl. In *Proceedings of the ACM 14th international workshop on Data Warehousing and OLAP*, pages 37–44.

[Tomingas et al., 2015] Tomingas, K., Kliimask, M., and Tammet, T. (2015). Data integration patterns for data warehouse automation. In *New Trends in Database and Information Systems II: Selected papers of the 18th East European Conference on Advances in Databases and Information Systems and Associated Satellite Events, ADBIS 2014 Ohrid, Macedonia, September 7-10, 2014 Proceedings II*, pages 41–55. Springer.

[Trujillo and Luján-Mora, 2003] Trujillo, J. and Luján-Mora, S. (2003). A uml based approach for modeling etl processes in data warehouses. In *Conceptual Modeling-ER 2003: 22nd International Conference on Conceptual Modeling, Chicago, IL, USA, October 13-16, 2003. Proceedings 22*, pages 307–320. Springer.

[Vaisman and Zimányi, 2014] Vaisman, A. and Zimányi, E. (2014). Data warehouse systems. *Data-Centric Systems and Applications*.

[Vassiliadis, 2009] Vassiliadis, P. (2009). A survey of extract–transform–load technology. *International Journal of Data Warehousing and Mining (IJDWM)*, 5(3):1–27.

[Vassiliadis et al., 2009] Vassiliadis, P., Simitsis, A., and Baikousi, E. (2009). A taxonomy of etl activities. In *Proceedings of the ACM twelfth international workshop on Data warehousing and OLAP*, pages 25–32.

[Vassiliadis et al., 2003] Vassiliadis, P., Simitsis, A., Georgantas, P., and Terrovitis, M. (2003). A framework for the design of etl scenarios. In *Advanced Information Systems Engineering: 15th International Conference, CAiSE 2003 Klagenfurt/Velden, Austria, June 16–20, 2003 Proceedings 15*, pages 520–535. Springer.

[Vassiliadis et al., 2005] Vassiliadis, P., Simitsis, A., Georgantas, P., Terrovitis, M., and Skiadopoulos, S. (2005). A generic and customizable framework for the design of etl scenarios. *Information Systems*, 30(7):492–525.

[Vassiliadis et al., 2002] Vassiliadis, P., Simitsis, A., and Skiadopoulos, S. (2002). Conceptual modeling for etl processes. In *Proceedings of the 5th ACM international workshop on Data Warehousing and OLAP*, pages 14–21.

[Waas et al., 2013] Waas, F., Wrembel, R., Freudenreich, T., Thiele, M., Koncilia, C., and Furtado, P. (2013). On-demand elt architecture for right-time bi: extending the vision. *International Journal of Data Warehousing and Mining (IJDWM)*, 9(2):21–38.

[Wang and Ye, 2010] Wang, H. and Ye, Z. (2010). An etl services framework based on metadata. In *2010 2nd International Workshop on Intelligent Systems and Applications*, pages 1–4. IEEE.

[Wang, 2017] Wang, J. (2017). Distributed etl.

[Wijaya and Pudjoatmodjo, 2015] Wijaya, R. and Pudjoatmodjo, B. (2015). An overview and implementation of extraction-transformation-loading (etl) process in data warehouse (case study: Department of agriculture). In *2015 3rd International Conference on Information and Communication Technology (ICoICT)*, pages 70–74. IEEE.

[Wohlin, 2014] Wohlin, C. (2014). Guidelines for snowballing in systematic literature studies and a replication in software engineering. In *Proceedings of the 18th international conference on evaluation and assessment in software engineering*, pages 1–10.

[Zamora, 1980] Zamora, A. (1980). Automatic detection and correction of spelling errors in a large data base. *Journal of the American Society for Information Science*, 31(1):51–57.