POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

A metadata model for healthcare: the Health Big Data case study

Tesi di Laurea Magistrale in
Computer Science and Engineering - Ingegneria Informatica

Author: **Nives Maria Migotto**

Student ID: 928574
Advisor: Prof. Cinzia Cappiello
Co-advisors: Prof. Pierluigi Plebani, Prof. Letizia Tanca
Academic Year: 2021-22

Abstract

A considerable amount of information is generated and used in all healthcare applications, increasing together with the technological progress. This information includes patient personal information and medical history, stored in electronic health records, data from imaging and laboratory examinations, data from genomics-driven experiments, and data generated by monitoring devices. All of these data come in different formats and from different sources, for instance various healthcare facilities, medical laboratories, wearables. Consequently, the need for new systems to store and manage these data has arisen. In particular, it is paramount to store different types of medical data in a repository accessible by different treatment and research facilities, in order to create an organized and rich dataset. The system employed in this project is the data lake, a repository that supports structured, semi-structured and unstructured data at any scale. However, to effectively maintain the value of the data, the data lake, as well as the other solutions, needs to be structured and regulated. This can be done by virtue of a data catalog, which, in turn, relies on metadata, i.e. additional data describing the managed resources. The objective of this work is to identify a metadata model fit for this use case. Seeing how the metadata strongly depend on the data they characterize, the structure they exist in and the purposes of the data, they need to be tailored to the specific application. As a suited model could not be found in the literature, one had to be expressly defined to meet the needs of this project. After developing the metadata model, it was validated through a demo implementation based on the open source data catalog platform Apache Atlas. Atlas was chosen after reviewing several available solutions. Overall, this work is a step in the implementation of a complete metadata model and, eventually, a data lake architecture to be applied in the healthcare field.

Keywords: Big Data, Data Catalog, Metadata, Healthcare, Medical Data

Abstract in lingua italiana

Una considerevole quantità di informazioni viene generata e usata in tutte le applicazioni sanitarie, aumentando di pari passo con il progresso tecnologico. Tali informazioni comprendono anagrafica e anamnesi dei pazienti, conservati nelle cartelle cliniche elettroniche, dati derivanti da immagini ed esami di laboratorio, dati derivanti da esperimenti basati sulla genomica e dati generati da dispositivi di monitoraggio. Tutti questi dati hanno diversi formati e provengono da diverse fonti, per esempio varie strutture sanitarie, laboratori medici, dispositivi indossabili. Di conseguenza, è sorta la necessità di nuovi sistemi per conservare e gestire i dati. Nello specifico, è fondamentale mantenere diversi tipi di dati in un archivio accessibile da diversi istituti di cura e ricerca, in modo da creare un dataset organizzato e completo. Il sistema utilizzato all'interno di questo progetto è il data lake, una repository che supporta dati strutturati, semi strutturati e non strutturati su qualisasi scala. Tuttavia, per mantenere in modo efficace il valore dei dati, il data lake, come le altre soluzioni, deve essere strutturato e regolato. Questo può essere attuato grazie a un catalogo dati che, a sua volta, si basa sui metadati, ossia dati aggiuntivi che descrivono le risorse gestite. L'obiettivo di questo lavoro è l'identificazione di un modello di metadati idoneo a questo caso. Poiché i metadati dipendono strettamente dai dati che caratterizzano, dalla struttura in cui si trovano e dallo scopo dei dati, devono essere scelti appositamente per la specifica applicazione. Dal momento che non è stato possibile trovare un modello opportuno nella letteratura, è stato necessario definirne uno espressamente per soddisfare i bisogni e i requisiti di questo progetto. Dopo aver sviluppato il modello di metadati, è stato validato tramite un'implementazione demo basata sulla piattaforma di catalogo dati open source Apache Atlas. Atlas è stato scelto dopo aver vagliato diverse soluzioni disponibili. Nel complesso, questo lavoro rappresenta un passo nell'implementazione di un modello di metadati completo e, alla fine, di un'architettura di data lake che possa essere applicata nel settore sanitario.

Parole chiave: Big Data, Catalogo Dati, Metadati, Sanità, Dati Medici

Contents

1 | Introduction

Since the early 2000s, the use of the Internet and related technologies has been increasing more and more (e.g., social media, mobile phones, IoT devices, etc.), and so has the amount of generated data. These voluminous, varied and fast changing data are known as big data. They have an enormous potential, as they provide a remarkable quantity of information and knowledge, but they pose some challenges as well. Their very nature makes it difficult for traditional storage and analysis technologies to handle big data, as it increases the complexity of data management, requiring, for instance, additional storage, processing, and analytical power. New architectures were therefore developed as an answer to the need for innovative and efficient ways to manage big data.

One of the most popular emerging platforms is the data lake. However, while data lakes are fit to deal with big data, they still need to be structured and regulated in order to effectively maintain the value of the data. To this end, data catalogs hold a crucial role. They describe, organize and keep track of the data, ensuring straightforward access and preserving their quality. To perform these tasks, data catalogs rely on additional data describing the stored resources, called metadata. It is therefore paramount to establish an appropriate set of metadata and to manage it accordingly.

Seeing as metadata by definition refer to data objects, describing them and the related requirements, keeping track of their use, and managing their lifecycle, metadata depend on the application domain. Therefore, different domains, or even different applications within the same context, require a suitable metadata set, designed specifically with those data and uses in mind.

In particular, focusing on the healthcare domain, some aspects are more relevant than in other fields, among which privacy, data quality and timeliness, and specific data formats, and therefore need to be adequately represented by the metadata. The proposals available in the literature refer for the most part to precise sub-fields, such as electronic health records (EHR) [1], document exchange [26], medical images [8], and clinical trials [33], which are often too specific. On the other hand, too general metadata sets can be insufficient for the needs of the application at hand, as they fail to consider relevant aspects particular to a given project requirements. In both cases, it is possible to take a cue and integrate (part of) them in the new set, but it will probably be necessary to complete them, extending or refining them. Moreover, the healthcare field includes a variety of different data formats, all of which need their own metadata. This explains why it is difficult to find an appropriate metadata set for a given application in the healthcare domain.

1.1. Aim of the thesis

The aim of this thesis is to identify a set of metadata aimed at assisting in the management of healthcare big data in the context of institutes for treatment and research (Istituti di Ricovero e Cura a Carattere Scientifico - IRCCS). As the name suggests, the main focus points of IRCCSs are diagnostic-therapeutic activities aimed at patient treatment and high-level research activities in the biomedical field and in the organization and management of health services. The data, stored within each institute, should be accessible by all of them, allowing to have a greater interoperability among the IRRCSs and to take advantage of a sizable amount of data, as each structure would be able to use the data provided by all the other ones. To this end, the metadata model wants to provide a common way to describe all the data, as to ease their retrieval.

A suitable metadata set, integrated into a data catalog, would be helpful at different levels. First, it would aid in the integration and management of the data, crucial aspect at the base of all other services, since otherwise the data cannot be accessed and used. Furthermore, the data scientists and medical professionals at the institutes must be able to find useful information when searching all of these data, be it to enroll patients in a clinical trial, look for previous cases relevant to correctly formulate a diagnosis, examine laboratory results, connecting exams to the patient they belong to, and so on. To this end, it needs to be possible, for instance, to group data based on some attribute, link related objects, and filter search results. All of this can be achieved through properly defined and managed metadata, which allows to consider not only the data locally managed by a single institute, but also data managed by other institutes.

More in detail, the metadata should help integrate different types of data coming from different types sources within the various IRCCSs. Additionally, they have to support the management of the data throughout their lifecycle. This entails defining and enforcing legal requirements (privacy and security policies, access rights, etc.), guaranteeing data quality and authenticity, ensuring data preservation, recording data provenance, including all transformations and versions, and keeping track of user actions. Metadata also need to include the technical details necessary to access and use the data. Equally important, they should allow to retrieve the proper datasets when a query is performed, by suitably organizing and characterizing each dataset.

It is worth noticing that this thesis does not consider the privacy aspects related to data sharing. Nevertheless, we are aware of the need to define privacy-preserving mechanisms able to control whether a datum can be shared and whether transformations are required before sharing.

1.2. Description of the work

The focus of the thesis was to produce a metadata model for the management and analysis of data, in particular healthcare related data stored in a data lake. To this end, the first step was a review of the state of the art to explore existing standards and models. This was done considering a number of papers, both generally applicable and centered on the healthcare domain, providing metadata classifications. After analyzing the results of the review and assessing the needs and requirements of the application, it was possible to elaborate our own metadata set, tailored to this project. The proposed model is hierarchical and includes three main categories, which are in turn divided into sub-categories. In addition, a research on the major open source data catalog solutions was carried out. Among the available platforms, seven have been included, as they seemed, each with their advantages and disadvantages, to fit the application at hand. Out of these seven, Apache Atlas was chosen, primarily on account of its flexibility, to implement the metadata model. An illustrative application of the model was then performed, aimed to demonstrate how the identified metadata can assist in the search for useful datasets.

1.3. Structure of the thesis

This document is structured as follows.

- Chapter 2 includes the background of this work, framing the application at issue in its context. More in detail, first, the concept of big data is presented. Second, the chosen data storage and management platform, the data lake, is discussed. Then, the concepts of data catalog and metadata, which have the most direct bearing on the project, are described. Lastly, seven open source data catalog solutions are detailed.

- Chapter 3 presents a review of the state of the art pertinent to the project, that is papers providing metadata models both for general applications and specific for the healthcare field, followed by the relevant considerations.

- Chapter 4 details the proposed metadata model, including the employed method, a comparison with the literature, and significant observations.
- Chapter 5 offers additional insight into Apache Atlas functionalities and shows how the metadata model can be applied, using Atlas, to retrieve the datasets pertinent to a query of interest.
- Chapter 6 concludes the thesis, providing a brief recap of the work that has been done and discussing possible future steps.

2 | Background

This chapter provides the theoretical foundations required to understand the content of the thesis.

2.1. Big data

A number of definitions of big data can be found in the literature. However, the most popular and well-accepted one was given by Douglas Laney. According to Laney [23], data are growing in three different dimensions, namely volume, velocity and variety. Volume indicates the large amount of data. Velocity refers to the rate at which data are collected and made available for further analysis. Variety denotes the different types of structured and unstructured data that can be collected (e.g. text, multimedia, log files, etc.). Table 2.1 illustrates the three Vs.

More characteristics have been added to the first three over the years, the most accepted being veracity, meaning that “data must be trusted to be useful. Verify the quality and reliability of enormous amounts of data streaming into systems at high speed from multiple sources, in multiple formats” [18].

Put simply, big data means larger, more complex datasets, with respect to the previous standards, especially from new data sources, different from the traditional relational databases. These datasets are so voluminous that conventional data processing software just can’t manage them, as it would require an excessive storage and computing power. However, these massive volumes of data can be used to address business problems that could not be tackled before [28].

Volume	The amount of data matters. With big data, you will have to process high volumes of low-density, unstructured data. These can be data of unknown value, such as Twitter data feeds, clickstreams on a web page or a mobile app, or sensor-enabled equipment. For some organizations, this might be tens of terabytes of data. For others, it may be hundreds of petabytes.
Velocity	Velocity is the fast rate at which data are received and (perhaps) acted on. Some internet-enabled smart products operate in real time or near real time and will require real-time evaluation and action.
Variety	Variety refers to the many types of data that are available. Traditional data types were structured and fit neatly in a relational database. With the rise of big data, data come in new unstructured data types. Unstructured and semistructured data types, such as text, audio, and video, require additional preprocessing to derive meaning and support metadata.

Table 2.1: The three Vs: Volume, Velocity and Variety

These definitions underline several benefits. For one thing, thanks to the additional information, big data make it possible to obtain more complete answers, which in turn allows to make more accurate decisions.

However, big data also entail some challenges. Firstly, the massive increasing volume represents a challenge in of itself, as it is not easy to keep pace with the data and find ways to effectively store it. In addition, it is likely necessary to integrate data in different formats from different sources. Then, useful data need to be extracted and cleaned in a way that enables meaningful analysis. The analysis itself cannot be carried out with traditional methods, as they would require excessive storage capacity and computing power. Further, an appropriate data governance is required in order to maintain the value of data. Data governance refers to the process of managing the availability, usability, integrity and security of the data. Lastly, big data technology itself is changing at a rapid pace, and keeping up with it can be challenging.

An interesting domain to be considered for big data is represented by healthcare. Healthcare is a multi-dimensional system aimed at the prevention, diagnosis and treatment of health-related issues [10]. A considerable amount of information is generated and used in all healthcare applications, increasing together with the technological progress. Healthcare big data include patient personal information and medical history, which are stored in EHRs, data from imaging and laboratory examinations, data from genomics-driven experiments (e.g. genome sequencing, gene expression), and data generated by monitoring devices.

Big data in healthcare can bring various benefits. It can help patients make the right decision in a timely manner. Collecting different data from different sources can help researchers and developers by improving research on new diseases, therapies and technologies. Healthcare providers may recognize high risk populations and act accordingly (i.e. propose preventive measures), enhancing patient experience [6].

In the healthcare domain, some issues tied to big data are particularly relevant. For instance, most of healthcare data, such as doctor notes, prescriptions, images and radiograph films, and genomic data, are unstructured or semi-structured, which makes it harder for machines to store and analyze it. Further, a high level of security needs to be guaranteed, due to the large amount of sensitive data. Privacy must also be ensured, while at the same time allowing the exchange of information where useful or necessary, for instance between healthcare providers. Equally important, data quality is critical; the collected data need to be cleansed and preprocessed in order to be usable and meaningful.

Given these points, big data show once again great promise, but also challenges that need to be faced in order to take advantage of its benefits.

2.2. Data lake

To face the challenges posed by big data and to overcome the limitations that solutions based on data warehouses were facing when managing more heterogeneous and a greater amount of data, the concept of data lake was introduced. A data lake can be described as “a massively scalable storage repository that holds a vast amount of raw data in their native format until it is needed plus processing systems (engine) that can ingest data without compromising the data structure” [25]. The repository should be unique and accessible to all the applications.

The data lake has been selected for this project. The question arises: why choose a data lake over a data warehouse? William Inmon [20], the father of the data warehousing concept, defines a data warehouse as a "subject-orientated, integrated, time variant, non-volatile collection of data in support of management’s decision-making process". Breaking down this definition, the following characteristics can be detailed:

- **Subject-oriented**: it usually provides information on a specific topic

- **Integrated**: the data come from heterogeneous sources and is converted into a unified schema at extraction time
- **Time variant**: the time horizon is significantly longer than that of an operational system
- **Non-volatile**: the data are frequently accessed, but rarely updated

Table 2.2 summarizes the main differences between data lakes and data warehouses. As can be seen, the data lake is more flexible, low-cost and efficient, especially in the context of big data.

	Data warehouse	**Data lake**
Data	Structured, processed data	Structures, semi-structured, unstructured, raw
Schema	Schema-on-write: the schema is defined before loading the data	Schema-on-read: the data structure is defined at the time data are used
Processing	ETL (Extract, Transform, Load): after extraction from the source, the data are cleansed and transformed before being loaded in the DW	ELT (Extract, Load, Transform): data are ingested as is and transformed only when it needs to be used
Storage	Expensive for large data volumes	Designed for low-cost storage
Agility	Less agile, fixed configuration	More agile, can be (re)configured as needed

Table 2.2: Data warehouse vs data lake

Going into detail, data lakes support various capabilities [12]:

- They can store data at scale for a low cost
- They can store data coming from different sources and of any type, whether structured, semi-structured or unstructured, in the same repository

- They can define the structure of the data at the time they are used (schema on read paradigm)
- They can perform transformations on the data facilitating subsequent analyses

Notably, the data being acquired and stored as is lightens the ingestion process, making data lakes suitable to handle big data. The data undergo appropriate transformations and is then analyzed to extract useful knowledge only when it is needed.

While there are different kinds of data lake architectures, they have in common a partition in sections in which data are stored based on their characteristics, applications, lifetime or processing stage.

To summarize, four stages of the data lifecycle within a data lake can be identified: ingestion, storage, processing and access [24]. First, the data enter the data lake, where they are stored. Then, when the user needs to read the data, they are standardized and transformed in a way that allows for significant analysis. Finally, the data can be access and consumed from the data lake.

As described so far, massive amounts of data are entered in the data lake and kept there with little to none control. With this in mind, the need for a management system that catalogues data and ensures their quality and value through the whole data lifecycle is evident. During ingestion, data governance allows to set policies guaranteeing relevant properties, including security and privacy policies, data provenance, data cleansing standards. Storage and processing can be optimized so that the transition of data to different processing stages is efficiently regulated and a record of all transformations is kept.

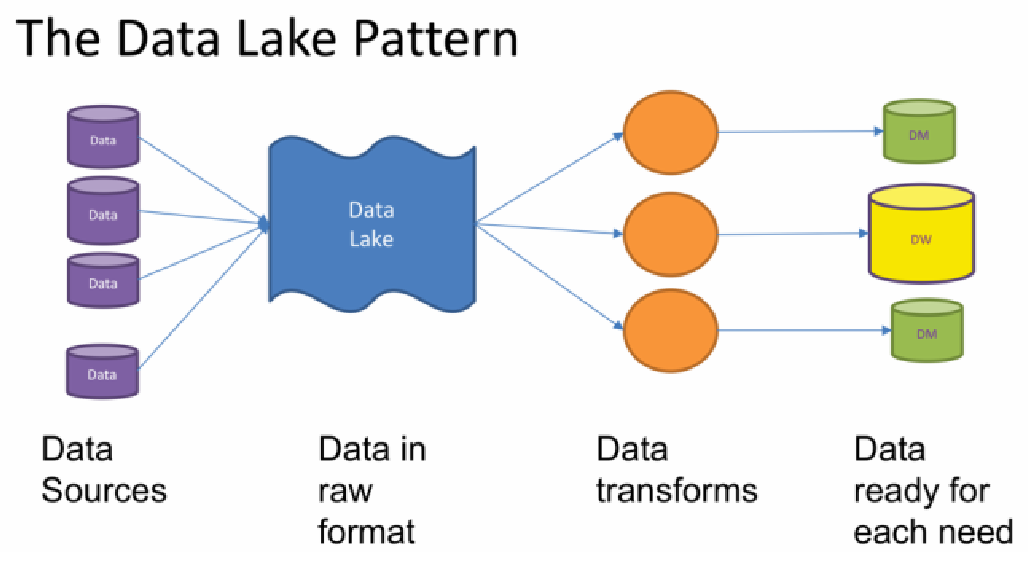


Figure 2.1: Data lake pattern

2.3. Data catalog

One of the issues arising from dealing with this kind of volume and variety of data stored in a single repository is finding and accessing data. Not only that, but also achieving it while respecting all the appropriate policies regulating data access and use. Data governance can be challenging as well, as it is critical to understand the kind of processed data, who uses hemt and to which purpose, and how they need to be protected, all without making data too hard to use.

More in detail, some of the issues related to data access are the following [29].

- Wasted time and effort on finding and accessing data: if data are not properly indexed and organized, retrieving the correct information is more difficult and time consuming. There could be useless copies of data or, conversely, inaccessible data
- Data lakes turning into data swamps: a data swamp is a badly designed, inadequately documented, or poorly maintained data lake. These deficiencies compromise the ability to retrieve data, and users are unable to analyze and exploit them efficiently.
- No common business vocabulary: it prevents related concepts and objects from being linked and grouped together, making it harder for a query to return useful and complete information
- Hard to understand structure: there is no clear way to know how the data are structured and organized
- Difficult to assess provenance, quality and trustworthiness: if these features cannot be ensured, through adequate data governance and data quality measures, the data cannot be trusted and hold little value
- No way to capture missing knowledge: it is not possible to infer if and which information is missing
- Difficult to reuse knowledge and data assets: without suitable structure and relationships between assets, knowledge derived from data analysis cannot be effectively stored and reused
- Manual and ad-hoc data prep efforts: made necessary by the lack of a systematic data cleansing and preparation process

A solution is the data catalog. Alation [2] gives a brief but exhaustive definition of data catalog, described as “a collection of metadata, combined with data management and search tools, that helps analysts and other data users to find the data that they need, serves as an inventory of available data, and provides information to evaluate fitness of data for intended uses".

Similarly, as specified by Gartner [34], "a data catalog maintains an inventory of data assets through the discovery, description and organization of datasets. The catalog provides context to enable data analysts, data scientists, data stewards and other data consumers to find and understand a relevant dataset for the purpose of extracting business value".

One particular use of data catalogs is to document and communicate the context, meaning and value of data that have been loaded to a data lake. According to Fraunhofer [22], the functionalities of a data catalog can be divided into the following 9 capabilities groups.

Data inventory: allows to register, organize and describe data. Data inventory capabilities include

- **Data registration**: allows to register data in the catalog
- **Metadata management**: allows to manage metadata throughout their lifecycle
- **Business glossary**: allows to describe data from the business point of view in order to understand its meaning and context
- **Data dictionary**: documents data from a technical viewpoint
- **Data lineage**: allows to track the history of data from point of origin to consumption
- **Data samples**: provides fragments of data, while respecting access policies, allowing for a better understanding of a dataset
- **Data access**: allows to access the data described in the catalog
- **Upload/link content**: allows to provide additional information for a dataset in the catalog

Data governance: support data governance activities. The main capabilities comprise

- **Roles and responsibilities**: allows to assign different roles and responsibilities to users
- **Handling sensitive data**: allows to identify sensitive data, who can access them and where they are used
- Additional capabilities include the definition and management of workflows, the creation and publishing pf policies to handle datasets, and the control and management of access to source data

Data assessment: aids the evaluation of data with respect to their fitness for use. The main capabilities comprise

- **Data risk**: allows to assess data related risks

- **Data profiling**: allows to automatically generate data profiles
- Additional capabilities include data usage tracking, data quality metrics definition, data value assessment, and data comparison and benchmarking

Data collaboration: assists in the collaboration of data-related user groups. The main capabilities comprise

- **Tagging**: allows to label data, facilitating their discovery and filtering
- **Textual explanations**: allows to textually describe data, making themmore understandable
- Additional capabilities include user communication, updates about data modifications, and data sharing

Data discovery: enables users to retrieve the data they need. The main capabilities comprise

- **Search**: allows to find data based on search terms, e.g. keywords, semantic text, etc.
- **Data exploration**: allows to browse data in a specific business area
- Additional capabilities include subscribing to data, downloading data from the catalog, and recommending data to users

Data analytics: supports the tasks of data analysts, data scientists and data engineers. Data analytics capabilities comprise writing scripts and designing data pipelines, and connecting the catalog to data application repositories.

Administration: supports the efficient and compliant usage, management and maintenance of the data catalog. The main capabilities comprise the configuration of the catalog functionalities and the management of users. The administrators are also able to monitor tasks and performance and analyze user behavior

Automation and artificial intelligence: facilitates data curation by supporting users and taking over manual tasks, allowing the data catalog to scale. The main capabilities comprise intelligent data labeling and automated data ingestion. Additional capabilities include intelligent data similarity and augmented discovery.

Visualization: expedites and assists the understanding and assessment of data. The main capability is the creation of data flow graphs, which help visualize the movement and transformation of data throughout their lifecycle. In addition, metrics used in profiling and data quality assessment, and knowledge graphs containing business terms and metadata can be displayed.

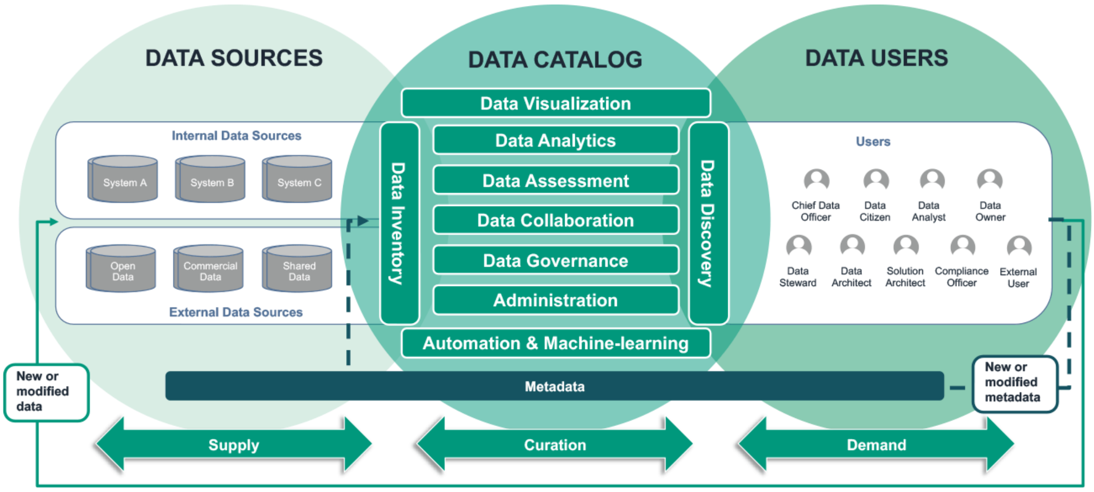


Figure 2.2: Data catalog functionalities

In a word, data catalogs increase search efficiency, give a better understanding of data meaning, context and value, regulate data access, keep track of data lineage and quality, and enhance data analysis and visualization.

2.4. Metadata

As mentioned before, a cornerstone of the data catalog is represented by metadata. A simple and brief definition often used to describe metadata is "data about data”. However, it is not particularly informative about the actual significance and role of metadata.

Focusing on the healthcare domain, which is the main interest of this work, as noted in Pocket Glossary of Health Information Management and Technology, AHIMA [1] defines metadata as "descriptive data that characterize other data to create a clearer understanding of their meaning and to achieve greater reliability and quality of information". According to Informatica [19], "Metadata is contextual information about a piece of data or a data set that is stored alongside the data. Metadata gives consumers of data, including applications and users, greater insight into the meaning and properties of that data".

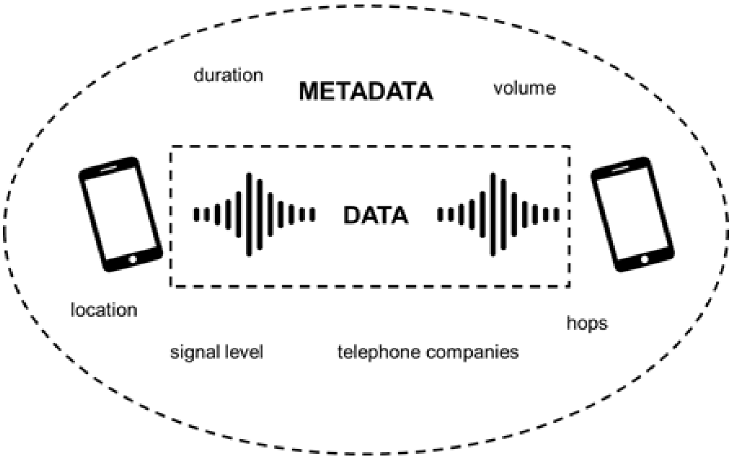


Figure 2.3: Data and metadata

Although the term metadata itself goes back to the '60s and has been widely adopted by computer scientists, metadata have been used well before the computer was invented. A clear example is libraries: the books and their contents represent the data, the catalogues used to describe them the metadata.

The main reason behind the development and use of metadata is memory, intended as retention of information. To explain, data need to be stored in order to be accurately preserved, but it is not enough: they have to be found again. For this, some sort of logical organization is needed and, in turn, metadata are necessary to make these filing systems work. Nevertheless, the role of metadata goes beyond the mere retrieval of a single piece of information. In fact, they allow to link data objects together to form knowledge, and eventually even wisdom.

A way to gain insight into the relationship between data, information, knowledge and wisdom is the DIKW (a.k.a. Ackoff) pyramid. At its base is the single unit of data, which holds little meaning on its own. When some meaning is inferred by looking into the relationships among data units, one can speak of information, the next level in the pyramid. Information may be thought of as organized data. The next step, to knowledge, occurs when context is added to information: a piece of information gains new meaning from the interaction with other units. For this to happen, patterns have to emerge which are themselves meaningful. Finally, wisdom is reached when this patterns are interpreted and analyzed so that new ones can emerge and everything learned until now can be applied in action. It can be said that if data and information are like a look back to the past, knowledge and wisdom are associated with what can be done now and what can be achieved in the future. In this model, metadata has a key role in enabling the pyramid to be constructed from the units of data up. In fact, to move from a level to the next it is crucial to establish links between the components of the current one.

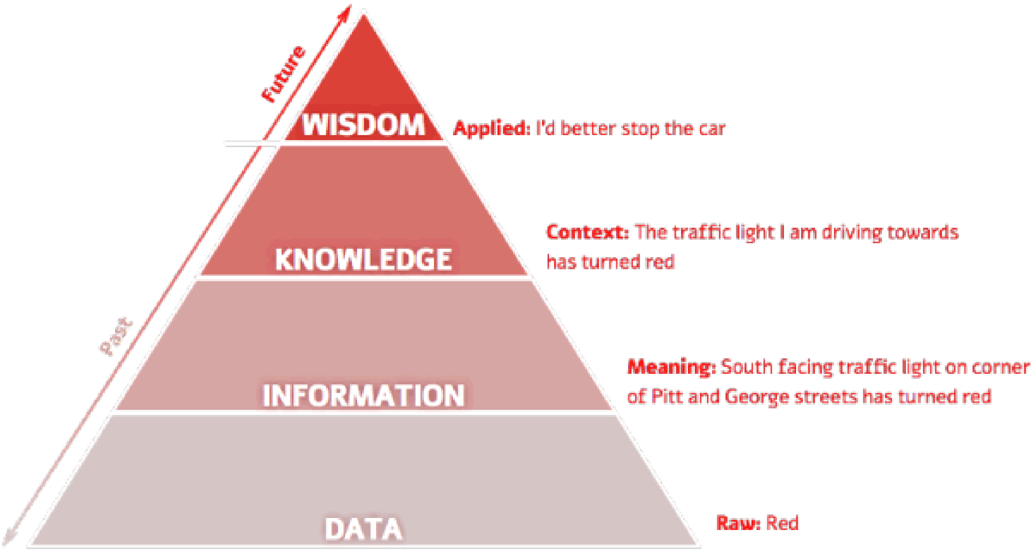

Future
Past
WISDOM
KNOWLEDGE
INFORMATION
DATA
Applied: I'd better stop the car
Context: The traffic light I am driving towards has turned red
Meaning: South facing traffic light on corner of Pitt and George streets has turned red
Raw: Red

Figure 2.4: DIKW (Ackoff) pyramid

A feature of metadata that is often overlooked is that they are a human construct. Metadata are designed by human beings for a particular purpose, thus the form they take strongly depends on their origin and they offer a subjective view about the objects they describe (aspects included and omitted, terms used, and so on) [14].

Metadata are usually divided into categories based on the different functions they carry out. Although these classifications vary based on the specific application, some categories are employed rather widely, if with slightly different characterization. A common broad division is between business and technical metadata. The former describe the meaning of data, aiding in their identification, understanding, and retrieval, while the latter refers to the technical attributes, such as the format and structure of the data, as needed by computer systems to deliver and render it properly. Another widely employed type is usually called administrative metadata, comprising all the background information necessary to store, preserve and access the data. The structures that bring together simple components into something larger that has meaning to a user are typically described by what is unsurprisingly named structural metadata. Metadata can be further partitioned according to the specific roles and applications.

In brief, "Metadata acts as a basis for information retrieval but can also have other functions, e.g. the management of user-access to resources, or preservation" [11]. Metadata therefore support every aspect of data upkeep and management, from governance policies to usage and versions tracking, as well as describe data and aid their retrieval.

2.5. Data catalog solutions

A variety of solutions offering data catalog, and data and metadata management services is already available on the market. In some cases, they are a part of a wider data platform, in others they exist as standalone services. Essentially all the main computer companies put on the market their own products, including big names as IBM, Microsoft, Oracle, and Amazon. However, for the purposes of this thesis, the focus will be on open-source software, developed to be freely used, modified, and distributed. In particular, seven solutions will be presented: Apache Atlas, Amundsen, CKAN, Kylo, Magda, Truedat, and iRODS, concentrating on the aspects more pertinent to metadata.

2.5.1. Apache Atlas

Apache Atlas [4], hosted by the Apache Software Foundation, is a metadata management and data governance tool that allows to ingest, discover, catalog, classify, and govern data from multiple data sources.

It employs a metadata model named ‘Type System’, which consists of definitions called types. Instances of types, called entities, represent the managed metadata objects. The Type system allows to define and manage types and entities. Metadata objects are persisted through a graph model, under the control of a Graph Engine. The Graph Engine creates the appropriate indices for the metadata objects as well. Further, ingest and export components are included. Two integration methods are provided. The primary mechanism to query and discover metadata is a REST API that enables types and entities to be created, updated and deleted. In addition, a messaging interfaced based on Kafka useful for communicating metadata objects to Atlas and to consume metadata change events from Atlas is in place. At this moment, the supported metadata sources include HBase, Hive, Sqoop, Storm, and Kafka.

Among the functionalities offered by Atlas, it is worth mentioning the presence of predefined metadata types, coupled with the possibility of defining new types. Each type can have attributes and objects references. Moreover, it is possible to dynamically create classifications and propagate them through data lineage. A basic search is available, which allows to query data by type, classification, attribute value or free text, as well as an advanced search based on a SQL-like language named Domain Specific Language (DSL). A rich REST API is available to search by complex criteria as well. It is also possible to filter the search results. Furthermore, Atlas provides a glossary of business terms, an intuitive UI to view lineage of data as they move through various processes and a fine-grained security for metadata access.

2.5.2. Amundsen

Amundsen [3] is a data discovery platform and metadata engine that was developed at Lyft. Similarly to Atlas, metadata are represented as a graph model.

Three of the main services are the search service, the metadata service, and the data ingestion library. The search service, responsible for searching metadata, serves Restful API and leverages Elasticsearch for most of its capabilities. It enables a page-rank style search based on usage patterns. The metadata service serves Restful API and is responsible for providing and updating metadata. The data ingestion library is responsible for building the metadata graph and search index. Users could either load the data with an ad hoc python script with the library or with an Airflow DAG importing the library.

2.5.3. CKAN

The Comprehensive Knowledge Archive Network (CKAN) [27] is a data management system (DMS) for the storage and distribution of data maintained by Open Knowledge Foundation.

A metadata set is provided for each dataset. CKAN offers a rich search experience that allows for quick Google-style keyword search as well as faceting by tags and browsing between related datasets. It is possible to search on dataset metadata, on full-text fields, for closely matching terms instead of exact matching (fuzzy-matching), via API, and via facets, such as tags, format, publisher. Faceted search allows to consecutively narrow the search by further facets, permitting to limit the search to datasets with specific formats or tags. CKAN allows users to register, update and refine datasets in a distributed authorization model called ‘Organizations’. This lets responsibility be distributed and authorization access managed by each department or agencies’ admins instead of centrally. Moreover, CKAN can be used to create a federated network of data portals that share data between each other. Further, CKAN has advanced geospatial features, covering data preview, search, and discovery.

There are over 200 open-source community extensions to CKAN services already available. In addition, CKAN has published an extending guide to help develop custom add-ons.

2.5.4. Kylo

Kylo [31], developed by Teradata, is an enterprise-ready data lake management software platform for self-service data ingest and data preparation with integrated metadata management, governance, security and best practices.

Kylo captures extensive business and technical metadata defined during the creation of feeds and categories. It processes lineage as relationships between feeds, sources, and sinks. Kylo automatically captures all operational metadata generated by feeds. In addition, it stores job and feed performance metadata and SLA metrics. Data profile statistics and samples are generated as well. A key part of Kylo's metadata architecture relies on the open-source JBoss ModeShape framework. The use of ModeShape makes the metadata model very extensible.

Kylo includes an integrated metadata repository and key capabilities for data exploration. Users can perform Google-like searches against data and metadata to discover entities of interest. Visual process lineage and provenance provide confidence in the origin of data. Automatic data profiling provides capabilities for data scientists and assurance in data quality. Some core features include dynamic schemas, which provide extensible features for extending schemas towards custom business metadata in the field, versioning, meaning the ability to track changes to metadata over time, text search, a flexible searching metastore, and portability, as Kylo can run on SQL and NoSQL databases. Web modules offer key data lake features such as metadata search, data discovery, data wrangling, data browse, and event-based feed execution.

2.5.5. Magda

Magda [9] is a data catalog system offered by the Australian Commonwealth Scientific and Industrial Research Organisation (CSIRO) that provides a single place where all of an organization's data can be catalogued, enriched, searched, tracked and prioritized.

The main services provided by Magda include discovery, federation, and metadata enhancement. Magda's discovery feature allows to retrieve higher-quality datasets above lower-quality ones, understand synonyms and acronyms, as well as search by time or geospatial extent. Federation means the ability to pull data from many different sources into one easily searchable catalog. Magda can accept metadata from its own cataloging process, existing Excel or CSV-based data inventories, existing metadata APIs such as CKAN or Data.json, or have data pushed to it via its REST API. Furthermore, Magda can automatically derive and enhance metadata, without the underlying data themselves ever being transmitted to a Magda server. This framework for enhancement is open and extensible, allowing to build custom enhancement processes using any language that can be deployed as a docker container.

While these functionalities are already complete and in use, other ones are still in the works. In particular, a guided, opinionated and heavily automated publishing process is being built into Magda, which will result in an easier time for those who publish data, and higher metadata quality to make it easier to search and use datasets for data users downstream. Additionally, an integrated, customizable authorization system will be added into Magda based on Open Policy Agent. It will allow datasets to be restricted based on established access-control frameworks (e.g. role-based), or custom policies specified by the organizations. It will also enable federated authorization, meaning that Magda will be able not only to pull data from an external source, but also mimic the same authorization policies, so that what can be seen from that system on Magda is the same as if one logged into it directly, and seamless integration with search, only getting back results in compliance with access policies.

Magda is designed as a set of microservices that allow extension by simply adding more services into the mix. Extensions to collect data from different data sources or enhance metadata in new ways can be written in any language and added or removed from a running deployment with little downtime and no effect on upgrades of the core product.

2.5.6. Truedat

Developed by Bluetab, Truedat [7] is a data cataloging and governance tool that allows to quickly unify and explore combined metadata from different sources on the same interface. It enables to organize and enrich information through configurable workflows and monitor data governance activity.

The metadata extracted from the sources are loaded into repositories called ‘systems’, accessible and manageable through the data catalog. The data catalog allows the consultation and enrichment of metadata obtained from the organization’s systems. For this purpose, search, filtering, and browsing options are provided through the catalog. Additionally, this module connects with both the glossary of concepts and the data quality glossary. It allows to discover metadata, filter, and search. In the business glossary, the business concepts are managed based on a predefined workflow that relies on configured permissions. New types of relationships between business concepts can be added. The data quality functionality allows the definition and implementation of quality rules, integrating with the glossary of concepts at the definition level and the data catalog at the implementation level. A global search engine is being effective on the business glossary and data catalog modules to query the data.

In addition, the data lineage service enables the visualization of the information life cycle, as well as the interconnection between each system of the organization, which in turn allows to have a complete traceability of the data, as well as impact analysis in the event of possible changes in data structures or processes. Three types of analysis are available: traceability, concerning the origin of the data, impact, relative to the use and effect of the data, and levels, related to the granularity of the data to be analyzed. Furthermore, profiling information can be run, which will be available to all authorized users through the data catalog. It is also possible to create a taxonomy defining the domains on which the information managed within the application is to be classified, as well as user permissions.

2.5.7. iRODS

The Integrated Rule-Oriented Data System (iRODS) [21] is a distributed, metadata driven, data centric data management software maintained by the iRODS Consortium.

In iRODS, metadata can include system or user-defined attributes associated with a Data-Object, Collection, Resource, etc., stored in the iCAT (iRODS Metadata Catalog) database. The metadata stored in the iCAT database are in the form of AVUs (attribute-value-unit tuples). Each of these three values is a string of characters in the database. The flexibility of this system can be used to interface with many different metadata standards and templates across different information domains. Metadata may be user-defined or applied automatically. Once metadata is applied, it can be used in various ways, for instance to trigger actions, based on rules defined in the iRODS rule engine. iRODS metadata can be searched as well. Complex queries can be generated using a subset of SQL operations. A search capability based on file contents has been implemented in an experimental capacity.

A policy framework around both full text and metadata indexing is present for the purposes of enhanced data discovery. Logical collections are annotated with metadata which indicates that any data objects or nested collections of data objects should be indexed given a particular indexing technology, index type, and index name. The iCAT catalog provides a means for persistently storing the state of the iRODS system. This includes the virtualization mapping as well as other system-metadata and user-defined metadata. iRODS V1.0 supports iCAT as Postgres or Oracle databases. It virtualizes the stages of the data lifecycle through policy evolution and the integrity service provides internal confidence that the data under management remain stable and good. Since every operation within an iRODS Zone (independent iRODS system) can be logged with an Audit Plugin, a well-formed query can discover every event associated with a particular data object, user, or resource. The results can be formed into a standardized target format and provide automated reporting for an organization. In addition, metadata templates give iRODS a friendly UI for specifying requirements, validation, and standardization.

Seven plugins are available to extend iRODS functionalities.

Table 2.3 summarizes the main aspects of each solution.

Data catalog solution	**Offered services**	**Metadata management**	**APIs**	**How to extend**	**Main languages**
Apache Atlas	Predefined metadata types and ability to define new types Dynamic classification creation and classification propagation UI to view data lineage Glossary of business terms Basic and advanced search Fine grained security for metadata access	Metadata model composed of definitions called types. Instances of types, called entities, represent the managed metadata objects. Metadata objects are persisted using a graph model	REST API allowing to create, update, delete and query metadata	N/A	Java JavaScript
Amundsen	Search service Metadata service Data ingestion library	Metadata are represented as a graph model	Restful API leveraging ElasticSearch for search Flask Restful API for metadata management	N/A	Python TypeScript

Data catalog solution	Offered services	Metadata management	APIs	How to extend	Main languages
CKAN	Multimodal search Metadata set provided for each dataset Distributed authorization model for data management Federation Advanced geospatial features	N/A	RPC-style API that exposes all of CKAN's core features to API clients	Extending guide for developing additional features 200+ open source community extensions	Python JavaScript
Kylo	Visual process lineage Automatic data profiling Dynamic schemas Versioning Text search Portability	The metadata architecture relies on the open-source JBoss ModeShape framework	REST API	N/A	Java TypeScript JavaScript
Magda	Discovery Federation Previews Metadata enhancement Automation (coming soon) Authentication (coming soon)	N/A	REST APIs (e.g. for authorization, content, indexer, registry, search and storage)	Extensions can be written in any language and easily added and removed	JavaScript TypeScript Scala

Data catalog solution	Offered services	Metadata management	APIs	How to extend	Main languages
Truedat	Global search engine Business glossary Data catalog Data profiling Data quality Data lineage Taxonomy	The metadata extracted from the sources are loaded into ‘systems’, accessible and manageable through the data catalog	APIs for data integration and task automation	N/A	Elixir
iRODS	Indexing Provenance Data lifecycle Integrity Metadata catalog Metadata templates	Metadata are stored in a relational database in the form of triplets consisting of an attribute field, a value field, and a unit field	RPC API REST API	Plugins (7 available)	C++ Python

Table 2.3: Data catalog solutions

3 | State of the art in metadata modeling

The first step toward the definition of a set of metadata was a thorough review of the literature, looking for classifications that could be useful for this application. Initially, the search was directed to non-specific categorizations, and then it focused on the healthcare domain, taking into account the data lake architecture. Even though several metadata partitions were found at a general level, nearly nothing turned up for data lakes specifically. On the other hand, several were available in the healthcare field, usually restricted to a particular topic (e.g., imaging, genomic data, clinical trials, etc.).

The core of the review was mainly published papers about metadata found on the Internet, both all-round and expressly about healthcare and medical applications. In addition, some online articles from qualified sources were considered. Sorting through the results, the focus was on the ones proposing a classification. Among them, the papers presenting the most pertinent metadata models for this project were selected.

In this chapter, the most relevant results of the review are presented, considering first non-specific schemas and then healthcare related ones. Observations relative to the findings follow.

3.1. Generic classifications

One of the most comprehensive and pertinent categorizations is the one proposed by Gilliland [15].

Administrative: Metadata used in managing and administering collections and information resources. Examples include acquisition information, rights and reproduction tracking, documentation of legal access requirements, location information, and selection criteria for digitization.

Descriptive: Metadata used to identify and describe collections and related information resources. Examples include cataloging records, finding aids, differentiations between versions, specialized indexes, curatorial information, hyperlinked relationships between resources, and annotations by creators and users.

Preservation: Metadata related to the preservation management of collections and information resources. Examples include documentation of physical condition of resources, documentation of actions taken to preserve physical and digital versions of resources (e.g., data refreshing and migration), and documentation of any changes occurring during digitization or preservation.

Technical: Metadata related to how a system functions or metadata behave. Examples include hardware and software documentation, technical digitization information (e.g., formats, compression ratios, scaling, routines, Tracking of system response times), and authentication and security data (e.g., encryption keys, passwords).

Use: Metadata related to the level and type of use of collections and information resources. Examples include circulation records, physical and digital exhibition records, use and user tracking, content reuse and multiversioning information, search logs, and rights metadata.

The US National Archives [32] offer a similar partition.

Administrative Metadata are used to manage collections of records. Examples include the Transfer Request (TR) Number, the Record Group, the name of the person authorized to transfer custody, etc.

Descriptive Metadata identify and describe records. Examples include a photograph's caption, the title of a book, or the composer of a song.

Preservation Metadata are the specialized set of information required to preserve and provide access to electronic records. Examples include the file format used to encode a file, the software necessary to view it, or an action taken to maintain it such as the results of a virus scan.

Technical Metadata describe aspects of electronic records important to their proper interpretation, rendering, or playback. The type of compression used with a digital image, the audio codec contained in a digital video, or the encryption algorithm used to digitally sign an email are all examples of technical metadata.

Use Metadata include information that describes how records can be accessed or circulated. Metadata identifying copyright status or security classification are examples of use metadata.

In their article on the FAIR principles (Findable, Accessible, Interoperable and Reusable) applied to metadata, Haux and Knaup [17] present a distinctive division.

Data Repository Metadata should be used to describe data provenance of the data source from that the analysis data sets were generated from. They represent an audit trail of the data that includes the point of data entry at the data consumer, data extraction and processing, and research output.

Catalogue Metadata comprise information about the organization of the data into groups, if present.

Dataset Metadata describe the analysis datasets that were provided by a data owner.

Distribution Metadata describe how data were made available for researchers.

Data Record Metadata contain information about the structure and the content of the dataset.

Even though their report focuses on images, the categories identified by Day and Patel [11] are still valid for our purposes. In fact, a portion of the data considered in this project is actually medical images.

The technical information required to view the image. These metadata would comprise information about image types - whether bitmaps, vector files or video - with information on particular file formats (e.g. TIFF, GIF, etc.), compression (e.g. JPEG) and colour metrics.

Information about the image capture process. They would include metadata about the type of image digitised with information about the size and dimensions of the original object. Technical information about the maker and model of scanner hardware used together with details of what has been done to each image would also be useful.

Information about the quality and veracity of an image. Users of images may need to know who was responsible for its digitisation. For example, a digital image created by a museum or art gallery from scanning the original or a high-quality surrogate may need to be differentiated from the same image scanned by a private individual at home.

Information about the original object. It will in many cases be important to know the precise nature and origin of the source object and link it with any surrogates.

Information about an image's authenticity. One of the problems with all digital information is that it is difficult to be sure that an information object is what it claims to be ("intellectual preservation"). How will users know that the digital object that they retrieve is the one that they want? How will administrators of digital repositories know that their holdings have not been subject to unauthorised changes, either accidental or deliberate? Solutions to these problems are likely to depend upon cryptographic techniques or implementations of digital signatures.

Information about rights management. Rights information is basic to the use and reuse of image resources. Rights metadata might include information on viewing and reproduction restrictions and contact information for the rights holders.

While the classification given by Gabriel, Hoppe and Pastwa [13] is, specific to data warehouses, most of the classes are applicable in the data lake context as well.

Terminology: Allow to identify the data objects and to standardize their labeling and meaning.

Data Analysis: Provide an overview of the processing and usage of object data.

Organization Reference: Describe where object data are created or imported, and how they are used in business processes. In addition, metadata of this category document access privileges as well as privacy classifications.

Data Quality: Provide information on the quality of the object data, which can generally be evaluated by two measures: the objective correctness of data values (e.g. precision, consistency) and the subjective aptitude of data to satisfy a certain information need.

Data Structure and Data Meaning: Describe the structures of data as well as their meaning.

Data Transformation: Describe the path object data take from the source systems to the analytical applications.

Metadata History: Record data changes over time with the help of a versioning system.

After citing the seven types of metadata suggested by Lagoze, Lynch and Daniel (1996), namely identification/description, terms and conditions, administrative data, content ratings, provenance, linkage/relationship data, and structural data, Greenberg [16] details her own schema.

Discovery metadata assist in the identification and retrieval of an object, and includes elements that represent both the physical and topical attributes of an information object. Discovery metadata include the elements that a user searches on when looking for an image. Examples include author/creator, title, and subject.

Use metadata permit the technical and intellectual exploitation of an information object. Technical exploitation includes system requirements, format, location (e.g., physical or virtual address), and other metadata that impact object access for the computer or an individual. Intellectual exploitation includes property rights, policy restrictions, and other terms and conditions metadata for content replication and publication citations. Together, technical and intellectual metadata determine the who can, what (e.g., a portion of an object), where, when, and how of object use. The class overlaps with what has been identified as structural metadata.

Authentication metadata support the evaluation of an information object's integrity, legitimacy and overall genuine quality. Source, relationship, version/edition, and digital signature are examples of metadata that help to determine the authenticity of an information object.

Administration (or administrative) metadata assist with the management and custodial care of an object. Provenance, date of acquisition, acquisition method (e.g., purchase/gift), restrictions, ownership, and preservation action metadata support administrative activities.

3.2. Healthcare related classifications

Moehrke [26] offers a view on metadata purposes in the healthcare field, in particular regarding document exchange models.

Provenance: Characteristics that describe where the data come from. These items are highly influenced by Medical Records regulations. This includes human author, identification of system that authored, the organization that authored, processor documents, successor documents, and the pathway that the data took.

Security & Privacy: Characteristics that are used by Privacy and Security rules to appropriately control the data. These values enable conformance to Privacy and Security regulations. These characteristics would be those referenced in Privacy or Security rules. These characteristics would also be used to protect against security risks to: confidentiality, integrity, and availability.

Descriptive: Characteristics that are used to describe the clinical value, so they are expressly healthcare specific. These values are critical for query models and to enable workflows in all exchange models. This group must be kept to a minimum so that it doesn't simply duplicate the data.

Exchange: Characteristics that enable the transfer of the data for both push type transfers, and pull type transfers. These characteristics are used for low level automated processing of the data. These values are not the workflow routing, but rather the administrative overhead necessary to make the transfer. This includes the document unique ID, location, size, mime types, and document format.

Object lifecycle: Characteristics that describe the current lifecycle state of the data including relationships to other data. This would include classic lifecycle states of created, published, replaced, transformed, deprecated.

In their article on a database of chronic wound images, Chakraborty, Gupta and Ghosh [8] concentrate mainly on descriptive metadata highly connected to the medical field.

Patient's personal data: PIDNUM, name, address, date of birth, birth place, sex etc.

Patient's medical data: plain text, image, textual, video etc.

Medical expert's personal data: doctor's personal information, unique identification code

System management related data: patient's list, password files, log files etc.

Pierson, Seitz, Duque and Montagnat [30] present the same center of attention, as do various other authors that will not be cited here.

Image-related metadata: image dimensions, voxels size, encoding, etc.

Acquisition-related metadata: acquisition device used, parameters set for the acquisition, acquisition date, etc.

Hospital-related metadata: radiology department responsible for this acquisition, radiologist, etc.

Medical record: anteriority, miscellaneous information explaining how to interpret this image, etc.

3.3. Discussion

All of these classifications are valid and functional. Nevertheless, comparing them is not as straightforward as it can appear, for a number of reasons. First, different criteria could be used. For instance, multiple sources classify metadata based on their function (descriptive, administrative, technical, usage, etc.), yet some choose to distinguish between domain-specific and domain-independent, user-related and organization-related or other features instead. Similarly, the partitions could cover different scopes; while one extends to all the metadata, another one might focus on descriptive metadata only, as it was often the case, especially with the healthcare oriented sources.

Some categories were frequently found in various sources, which seems to facilitate the comprehension and comparison of diverse classifications. Be that as it may, the same term is sometimes used by different authors to mean different things, which could turn up to be confusing. As an example, provenance usually refers to information concerning the origin and version history of managed data, but it sometimes is used to denote only the source. Conversely, the same concept may be defined with different expressions. In like manner, several times classes overlapped. To illustrate, two sources may use the same criterion, cover the same scope, and use at least partially the same categories, but assign a given item or subclass to different categories. For instance, one might place information about property rights under use metadata, while another one considers it part of administrative metadata.

On a different note, a considerable difference between general scope and healthcare related categorizations is the appearance of numerous domain specific classes and items in the latter. On one hand, the relevance of particular aspects, such as privacy and secure transfer of data, in this field are reflected in the presence corresponding metadata categories. On the other hand, especially within the so called descriptive metadata (i.e. business metadata describing the resources), multiple items are highly specific to the healthcare field and at times to even more particular areas, e.g. genomic data, clinical trials.

All the above mentioned issues can be reduced to the fact that each schema is tailored to the specific application at hand. In fact, different applications have different requirements and focus points, which reflects in the choice of relevant metadata. While some types have general purposes and are widely applicable, others are tied to particular fields or frameworks. Moreover, for the same classes, in one case some aspects might be more important (e.g. privacy in case of particularly sensitive data), leading to a more exhaustive and detailed metadata category.

4 | Proposed metadata model for healthcare data

Based on the proposed metadata models found in the literature, in this chapter a metadata model suitable for the healthcare domain is presented.

4.1. Method

Before going into detail on the rationale of the classification, it is appropriate to point out a couple of definitions. A dataset means a collection of homogeneous data; a database refers to a collection of datasets. With this in mind, metadata are intended as referred to datasets.

We identified three main categories of metadata, governance, data lifecycle management and descriptive, which in turn are be divided into three sub-categories, business/semantic, intrinsic, and inter-relationship. This was done by taking into account the most recurrent classes mentioned in the surveyed sources, the needs and requirements specific to this application, and some considerations about the metadata themselves.

Of the latter, the most relevant regards the so-called technical metadata, which represent the technical aspects of data that are necessary for data presentation, manipulation, and analysis. While, when mentioned, they are regarded as a category in and of itself (see [15] [33] [11] [16], [8]), we believe that they are better characterized as transversal. In fact, while each of the other classes identifies a specific topic, a 'what', technical metadata can be seen as the 'how' of all these classes. For instance, if governance metadata includes policies about access rights and authentication, passwords and encryption details represent the corresponding technical metadata, describing how these rights are ensured. In other words, instead of having a dedicated technical metadata category, every other one would have (if necessary) a section dedicated to technical details.

This is only an example of a more extensive challenge encountered while trying to define a precise classification of metadata. Specifically, we found that the classes often overlapped, regardless of the partition, which called for a hierarchical structure. To clarify, various groupings found in the literature covered approximately the same metadata types, but named and/or parted them in different ways. To present a few examples, both [15] and [13] cite multiversioning information, but the former assigns it to use metadata, while the latter keeps it as a standalone category, metadata history. While data quality, data authenticity, and rights management are so important to [11] that each is a class in an of itself, [13] considers only data quality and [15] includes just rights management within administrative metadata. [13], [17], and [26] all identify a category dedicated to data lifecycle and processing, but they named it in three different ways, data transformation, data repository and object lifecycle respectively.

In particular, the commonly named descriptive metadata are sometimes broadly used to denote anything describing data, from the content to the structure and relationships, while other times some of these aspects are regarded as individual classes. For instance, both [17] and [11] put together data meaning/content and structure (in data record metadata, and data structure and data meaning respectively), but still manage to have differences, with the former keeping information about the organization of data into groups separate (catalogue metadata), and the latter keeping a distinct class for terminology. Unlike [16], who calls them discovery metadata, [15] and [32] both use the usual term descriptive metadata, without setting many restrictions as to what they include. The solution that seemed more suited to resolve this issue is a hierarchical structure, in which descriptive metadata are again partitioned into subsets.

For other categories, specific sub-classes were attributed to one or the other based on the agreed definitions, although it could be argued that a specific item or subclass might be better suited for a different category. This is inevitable, since the metadata cover more a continuous spectrum than precisely disjoint sets; as a result, some items are bound to be attributable to more than one group.

4.2. Metadata model

As stated above, three general classes have been distinguished: governance, data lifecycle management and descriptive metadata. Descriptive metadata have been further divided into business/semantic, intrinsic, and inter-relationship metadata, due to the broadness of their scope.

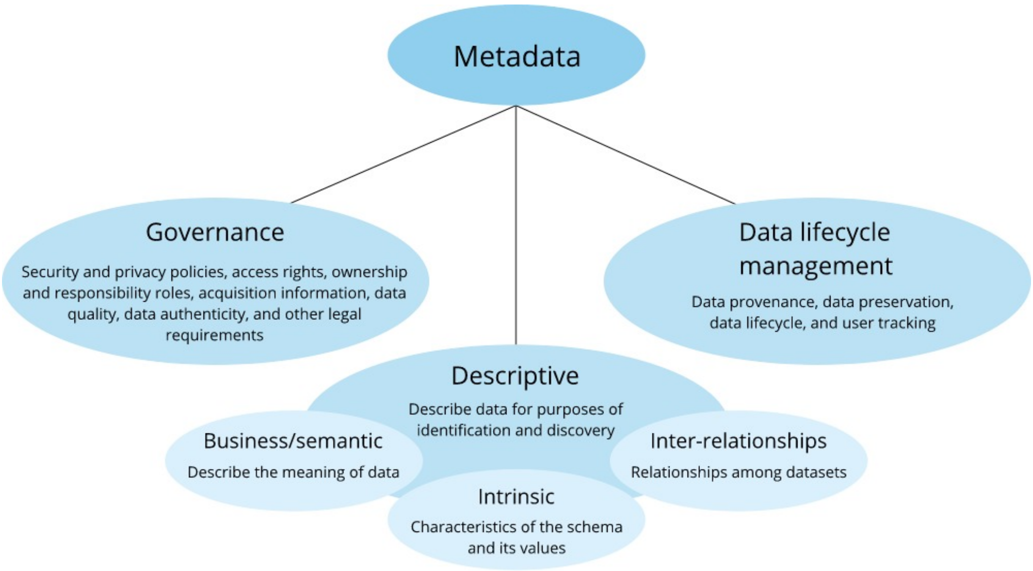

Metadata
Governance
Security and privacy policies, access rights, ownership and responsibility roles, acquisition information, data quality, data authenticity, and other legal requirements
Data lifecycle management
Data provenance, data preservation, data lifecycle, and user tracking
Descriptive
Describe data for purposes of identification and discovery
Business/semantic
Describe the meaning of data
Inter-relationships
Relationships among datasets
Intrinsic
Characteristics of the schema and its values

Figure 4.1: Metadata model

Governance metadata cover all security and privacy policies, access rights, ownership and responsibility roles, acquisition information, data quality, data authenticity, and other legal requirements.

Data lifecycle management metadata are related to data provenance, including the source of the data, all transformations performed and existing versions, usage tracking, and information required to preserve and use the data, including technical specifications. This group is approximately comparable to what is generally referred to as use metadata, but we felt that this term does not fully represent its scope.

Descriptive metadata describe data for purposes of identification and discovery. Since this is a very comprehensive definition, they are further divided into subcategories: business/semantic, intrinsic and inter-relationship metadata.

Business/semantic metadata describe the meaning of data through descriptions, tags, indexes, attributes, etc. In addition, they include constraints and other relationships within each datasets. Their usefulness can be increased by compiling a knowledge base and employing it to annotate data. As a result, different items referring to the same concept or connected by some semantic relationship can be connected and retrieved more easily.

Intrinsic metadata describe the characteristics of the schema and its values. They include profiling, statistics and descriptive-technical metadata.

Inter-relationship metadata pertain to relationships among datasets. For instance, they could be used to map foreign keys connecting datasets within a database. Similarly, they may outline relationships between datasets belonging to different databases.

Table 4.1 outlines the proposed metadata model, specifying for each category a brief definition and a number of examples of metadata items belonging to it.

For reference, table 4.2 presents a mapping of the main classifications found in the literature to the schema we defined. Each column represents a category from our model, each row a paper presenting their own classification, and each paragraph within a cell a category identified by the paper author(s). As can be seen, the categories never correspond exactly - sometimes they do not cover all of our types (or sub-types), in which case a column remains empty, at other times more than one fits in only one type, hence more paragraphs in a cell, or vice versa, the same category then appears in more than one column. Metadata examples for each category are included in parentheses. As an illustration, Gilliland's use and preservation metadata, from the first row, both fall within our data lifecycle management class. Conversely, the technical metadata span both governance and descriptive metadata. On the other hand, none of the categories identified by Haux and Knaup correspond to our governance metadata. This highlights how each model, based on its own context and purpose, while often including features similar to others, is organized in a unique way, that is not necessarily fit for different applications.

Type	Description	Examples
Governance	Comprises security and privacy policies, access rights, ownership and responsibility roles, acquisition information, data quality, data authenticity, and other legal requirements	Anonymization details, exchange authorization details, source and acquisition information, data authenticity and veracity information, information on viewing and reproduction restrictions Technical: authentication and security data, e.g., passwords, encryption keys, digital signatures and cryptographic techniques, data access logging
Data lifecycle management	Cover data provenance, data preservation, data lifecycle, and user tracking	Data storage option, usage tracking, content reuse and multiversioning information, transformations documentation Technical: details on how transformations are performed, log files, software and hardware specifications, actions performed to maintain the data, system requirements, version, location, acquisition parameters
Descriptive	Describe data for purposes of identification and discovery. Include business/semantic, intrinsic, and inter-relationships metadata	Indexes, annotations (e.g., descriptions, tags, keywords), intra-dataset relationships, profiling, statistics, foreign keys Technical: file format, compression ratio, size, data lineage

Table 4.1: Metadata for data analytics in healthcare

Source	Metadata		
	Governance	Data lifecycle	Descriptive
A. J. Gilliland (2008), Introduction to metadata	Administrative metadata: used in managing and administering collections and information resources (acquisition information, rights and reproduction tracking, documentation of legal access requirements, location information, selection criteria for digitization) Technical (hardware and software documentation, authentication and security data, e.g., encryption keys, passwords)	Use metadata: related to the level and type of use of collections and information resources (circulation records, physical and digital exhibition records, use and user tracking, content reuse and multiversioning information, search logs) Preservation metadata: related to the preservation management of collections and information resources (documentation of physical condition of resources, documentation of actions taken to preserve physical and digital versions, e.g., data refreshing and migration, documentation of any changes occurring during digitization or preservation)	Descriptive metadata: used to identify and describe collections and related information resources (cataloging records, finding aids, specialized indexes, curatorial information, annotations by creators and users) Technical (technical digitization information, e.g., formats, compression ratios, and scaling routines) Hyperlinked relationships between resources

Source	Metadata		
	Governance	Data lifecycle	Descriptive
US National Archives (2016), Metadata in Electronic Records Management	Administrative Metadata are used to manage collections of records (Transfer Request N., Record Group, name of the person authorized to transfer custody) Use Metadata include information that describes how records can be accessed or circulated (copyright status, security classification)	Preservation Metadata are the specialized set of information required to preserve and provide access to electronic records Technical (encryption algorithm used to digitally sign an email, software necessary to view it, an action taken to maintain it such as the results of a virus scan)	Descriptive Metadata identify and describe records Technical (compression used with a digital image, audio codec contained in a digital video, file format used to encode a file)
C. Haux and P. Knaup (2019), Using FAIR Metadata for Secondary Use of Administrative Claims Data		Data repository metadata: describe data provenance of the data source from which the analysis data sets were generated - audit trail of the data including point of entry at the consumer, data extraction and processing, and research output Distribution metadata: describe how data were made available for researchers	Dataset metadata: describe the analysis datasets that were provided by a data owner Data record metadata: contain information about the structure and the content of the data set Catalogue metadata: contain information about the organization of data into groups, if present

Source	**Metadata**		
	Governance	**Data lifecycle**	**Descriptive**
M. Day and M. Patel (2002), Metadata for images. A report for the FILTER project	Information about image quality (digitization supervisor) Information about rights management (viewing and reproduction restrictions, rights holders contact information) Technical Information about image authenticity (cryptographic techniques, digital signatures)	Technical Information required to view the image Information about the image capture process (type of image digitized, size and dimensions of original object, maker and model of scanner hardware)	Information about the original object Technical (image type, file format, compression, color metric)
R. Gabriel, T. Hoppe and A. Pastwa (2010), Classification of Metadata Categories in Data Warehousing - A Generic Approach	Organization Reference: Describe where object data are created or imported, and how they are used in business processes Data Quality: Provide information on the quality of the object data	Data Analysis: Provide an overview of the processing and usage of object data Data Transformation: Describe the path object data take from source to analytical applications Metadata history: Record data changes over time with the help of a versioning system	Terminology: Allow to identify the data objects and to standardize their labeling and meaning Data Structure and Data Meaning: data objects are grouped to data object types according to their relationships among each other

Source	Metadata		
	Governance	Data lifecycle	Descriptive
J. Greenberg (2001), A quantitative categorical analysis of metadata elements in image-applicable metadata schemas	Administration metadata assist with the management and custodial care of an object (date of acquisition, acquisition method, restrictions, ownership, source) Use metadata permit the technical and intellectual exploitation of an information object (property rights, policy restrictions, and other terms and conditions) Technical (digital signature)	Technical (system requirements, format, location, version/edition)	Discovery metadata assist in the identification and retrieval of an object, and includes elements that represent both the physical and topical attributes of an information object Linkage/ relationship data
J. Moehrke (2012), Healthcare Metadata	Security & Privacy: characteristics that are used by Privacy and Security rules to appropriately control the data Exchange: characteristics that enable the transfer of the data	Provenance: characteristics that describe where the data come from Object Lifecycle: characteristics that describe the current lifecycle state of the data	Descriptive: characteristics that are used to describe the clinical value, so they are expressly healthcare specific

Source	**Metadata**		
	Governance	**Data lifecycle**	**Descriptive**
C. Chakraborty, B. Gupta and S. K. Ghosh (2014), Mobile metadata assisted community database of chronic wound images	Technical (password files)	System management related data (patient's list, log files)	Patient's personal data (PIDNUM, name, address, date of birth, birth place, sex) Patient's medical data (plain text, image, textual, video) Medical expert's personal data (doctor's personal information, unique identification code)
J. Pierson, L. Seitz, H. Duque and J. Montagnat (2004), Metadata for efficient, secure and extensible access to data in a medical grid		History-related metadata (image sources, algorithms, parameters)	Hospital-related metadata (radiology department, radiologist) Medical record (anteriority, miscellaneous information explaining how to interpret this image)

Table 4.2: Source mapping for metadata

4.3. Observations

Point often overlooked, the distinction between data and metadata is not always clear. Not only that, but, on occasion, an object can be considered as both data and metadata within the same organization depending on the use and context. For instance, while examining the medical record of a patient, all patient information is undoubtedly data. On the other hand, if the object of interest is a medical image (e.g. MRI or CT scan) or other kinds of exam or laboratory results (e.g. ECG, genetic screening, etc.), the patient information becomes metadata providing additional insight to the image.

Inspecting the proposed metadata set, no reason comes to mind to use any of the listed items, except for the descriptive (semantic) metadata, as data. On the other hand, several data objects can assume the role of metadata with respect to another object. In particular, most of the times they would be part of the semantic metadata. This is because the other metadata are mainly service metadata, aimed at aiding the proper management and use of the data, and they do not belong to the same semantic domain as the data (i.e. medical and healthcare data). As a result of this dual nature, it would be useful to store data and metadata together in the same way.

5 | Implementation

Among the data catalog solutions listed in Chapter 2, Apache Atlas was chosen for the practical implementation of the metadata model, mainly because of its features, flexibility and ease of use and integration. After going into more detail about the tool, this chapter will present how Atlas was used to put into practice the metadata model, showing how it can be used to retrieve the datasets of interest.

5.1. Apache Atlas

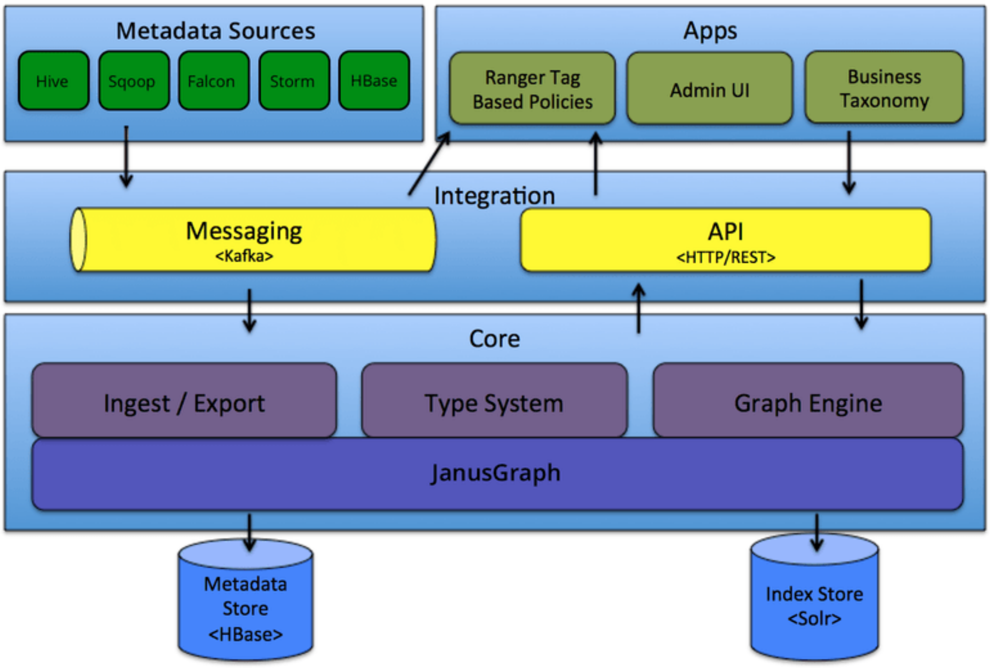


Figure 5.1: Atlas architecture

As is shown in Figure 5.1, Atlas [5] core include three components, Type System, Graph Engine and Ingest/Export. Atlas allows users to define a model for the metadata objects they want to manage. The model is composed of definitions called types. Instances of types, called entities, represent the actual metadata objects that are managed. The Type System is a component that allows users to define and manage the types and entities. All metadata objects managed by Atlas out of the box are modelled using types and represented as entities. One key point to note is that the generic nature of the modelling in Atlas allows data stewards and integrators to define both technical metadata and business metadata. It is also possible to define rich relationships between the two using features of Atlas.

The Graph Engine is responsible for translating between the types and entities of the Type System and the underlying graph persistence model. The graph approach provides great flexibility and enables the efficient handling of rich relationships between the metadata objects. In addition to managing the graph objects, the graph engine also creates the appropriate indices for the metadata objects so that they can be searched efficiently. Atlas uses the JanusGraph to store the metadata objects. The Ingest component allows metadata to be added to Atlas. Similarly, the Export component exposes metadata changes detected by Atlas to be raised as events. Consumers can consume these change events to react to metadata changes in real time.

Regarding integration, All functionality of Atlas is exposed to end users via a REST API that allows types and entities to be created, updated and deleted. It is also the primary mechanism to query and discover the types and entities managed by Atlas. In addition to the API, users can choose to integrate with Atlas using a messaging interface that is based on Kafka.

As it is particularly relevant to our purposes, the Type System will be discussed more in detail. A type is a definition of how a particular type of metadata objects is stored and accessed. A type represents one or a collection of attributes that define the properties for the metadata object. Each type, uniquely identified by a name, has a metatype. The metatype can be primitive (boolean, byte, short, int, long, float, double, biginteger, bigdecimal, string, date), an enumeration, a collection (array, map) or composite (Entity, Struct, Classification, Relationship). Entity and Classification types can extend from other types, called supertype - by virtue of this, it will get to include the attributes that are defined in the supertype as well. This allows modellers to define common attributes across related types. It is also possible for a type in Atlas to extend from multiple super types.

An entity is a specific value or instance of an Entity type and thus represents a specific metadata object in the real world. Every entity is identified by a unique identifier (GUID). The values of an entity are the values of the attributes defined during the corresponding type definition. Each attribute has additional properties beside a name and metatype, relative for instance to its cardinality and whether it is unique, mandatory and so on.

Atlas comes with a few predefined system types: Referenceable, Asset, Infrastructure, DataSet and Process. In this application the focus will be on DataSet. It extends Referenceable, which represents all entities that can be searched for using a unique attribute, and can be used to represent any type that stores data. In addition, entities of types that extend DataSet participate in data transformation and this transformation can be captured by Atlas via lineage (or provenance) graphs.

Typically, the described model captures technical attributes and metadata objects are created and updated by processes that monitor the real objects. It is often necessary to augment technical attributes with additional attributes to capture business details that can help organize, search and manage metadata entities. For this purpose, Business Metadata is available. Business Metadata is a type supported by Atlas Type System (similar to Entity, Struct, and Classification types). A business metadata type can have attributes of primitive type. Each business metadata attribute can be associated with a number of Entity types as well. Once a business metadata attribute is associated with an Entity type, Atlas allows values to be assigned to entities, and then entities to be searched based on such values, via UI and REST APIs.

Another tool available to organize and find entities is the classifications. Classifications are strings equipped with a certain complexity and structure that can be associated to entities. Classifications can be enriched with attributes in the form of key-value pairs and the value can be set to describe a particular entity. Further, they can have a tree structure, so that sub-classifications can inherit attributes from the super-classification. In addition, classifications can automatically propagate to additional entities through lineage relationships. They can also be used to drive access control policies.

A Glossary provides appropriate vocabularies for business users and it allows the terms (words) to be related to each other and categorized so that they can be understood in different contexts. These terms can be then mapped to entities and classifications. A term is a useful word for an enterprise. It can belong to zero or more categories, which allow to scope it into narrower or wider contexts. A term can be assigned to zero or more entities. It can be classified using classifications and the same classification is applied to the entities to which the term is assigned. A term can be linked to other terms through relationships such as synonyms, antonyms, preferred term, etc. A category is a way of organizing the terms so that their context can be enriched. A category may or may not contain hierarchies.

Apache Atlas offers to ways of querying entities: basic search and advanced search. The basic search allows to search by the type of an entity, by the associated classification and terms, and by text. It also supports filtering on the entity attributes (business metadata included) as well as the classification attributes. Attribute based filtering can be done on multiple attributes with AND/OR conditions. The basic search panel is shown in Figure 5.2.

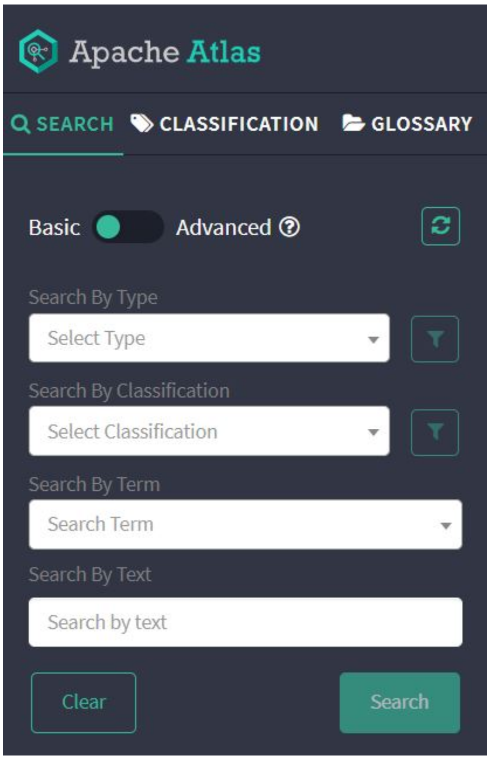


Figure 5.2: Atlas basic search panel

The advanced search is also referred to as DSL-based search, from the the query language it employs, the Domain Specific Language. DSL is a language with simple constructs that help users navigate Atlas data repository. Its syntax loosely emulates the popular Structured Query Language (SQL) from the relational database world. DSL provides users with an abstraction that helps them retrieve the data by being aware of just the types and their relationships within their dataset. It allows for a way to specify the desired output, it gives the possibility to group and aggregate results, and its syntax accounts for the use of classifications.

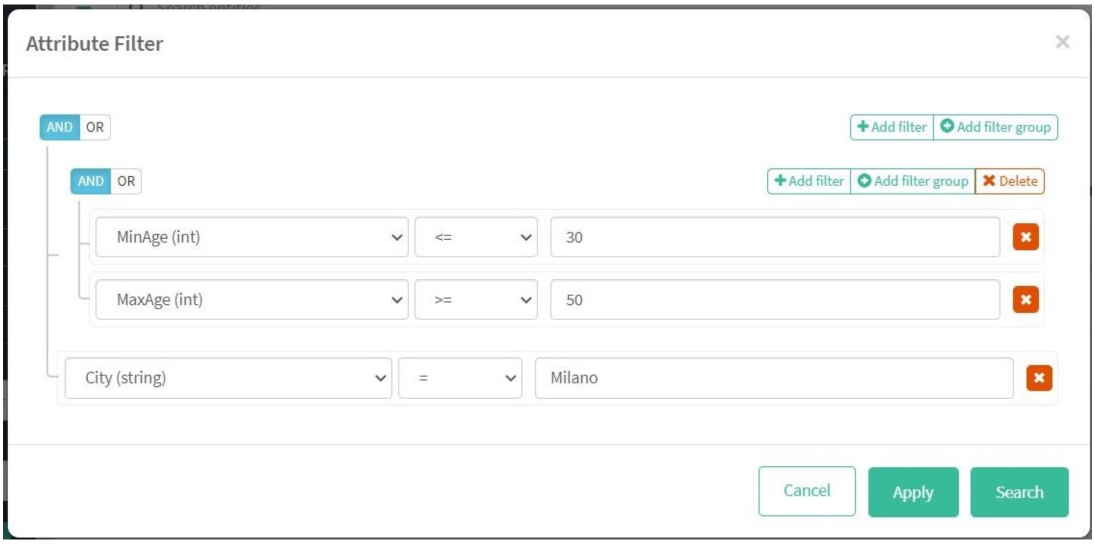
5.2. Metadata model implementation

As far as this application is concerned, first, we defined specific types based on the different kinds of data considered in the project in order to better categorize the entities. The four main types are patient data, image, signal and omics. Since they all represent types that store data, they all extend DataSet. In turn, image, signal and omics are provided with sub-types (e.g. MRI, CT, X-ray for image, ECG, EEG, EMG for signal, and genomics, proteomics, transcriptomics for omics). This way, each sub-type can inherit the parent attributes and add specific ones. At this time, the types do not have their particular attributes, since they are specific to each data format and out of the scope of this demonstrative implementation. Nonetheless, they will need to be added in order to effectively exploit this structure. The types definition can be found in Appendix A.

Seeing how technical the type attributes tend to be, the model was mapped to Atlas through the business metadata. Each of the three main categories (governance, data lifecycle management, and descriptive) is represented by a business metadata, and each metadata item by one of the associated attributes. While most metadata were applicable to all of the defined types, a few were fit for a particular one - for instance, 'body district' can only be pertinent to an image. For this reason, such attributes where made to only be assignable to entities of the related types.

To show the ability of the metadata model to aid in selecting the datasets relevant to a given query, two entities were added to the system. They represent a dataset of diagnostic images of patients with lung cancer at Istituto Nazionale dei Tumori and a dataset of medical information of patients at Ospedale San Raffaele respectively. The first one is of type image, the second one patient data. A list of metadata attributes was associated suitably to each entity.

Two example queries will be considered. The first one targets people between the ages of 30 and 50 living in Milan. As can be seen in Figure 5.3, it is possible to filter the results based on the values of the attributes.



Attribute Filter

AND OR

AND OR

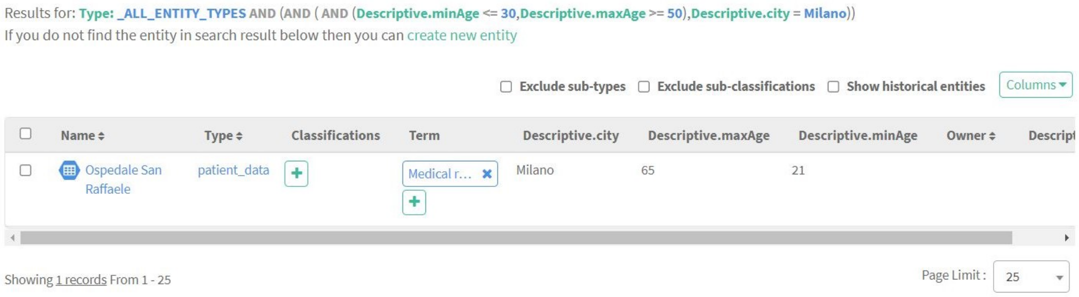
MinAge (int) <= 30

MaxAge (int) >= 50

City (string) = Milano

Figure 5.3: First query attribute filter

Figure 5.4 shows how the search results contain only the entity that meets the requirements.



Results for: Type: _ALL_ENTITY_TYPES AND (AND (AND (Descriptive.minAge <= 30,Descriptive.maxAge >= 50),Descriptive.city = Milano))
If you do not find the entity in search result below then you can create new entity

☐ Exclude sub-types ☐ Exclude sub-classifications ☐ Show historical entities

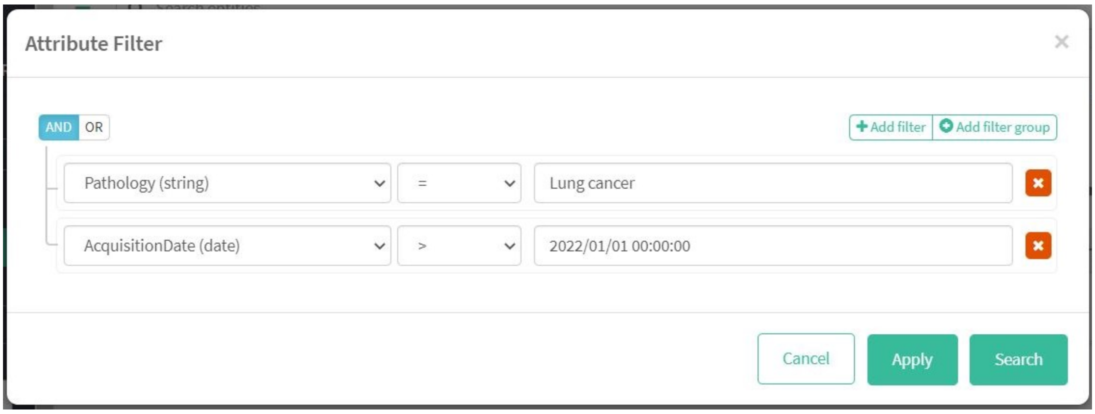
☐	Name ⬍	Type ⬍	Classifications	Term	Descriptive.city	Descriptive.maxAge	Descriptive.minAge	Owner ⬍	Descript
☐	Ospedale San Raffaele	patient_data	+	Medical r… ✖ +	Milano	65	21		

Showing 1 records From 1 - 25

Page Limit : 25

Figure 5.4: First query results

Likewise, it is possible to search for datasets regarding lung cancer uploaded in 2022 and, once again, the appropriate entity is shown in the results, as displayed in Figure 5.5 and Figure 5.6.



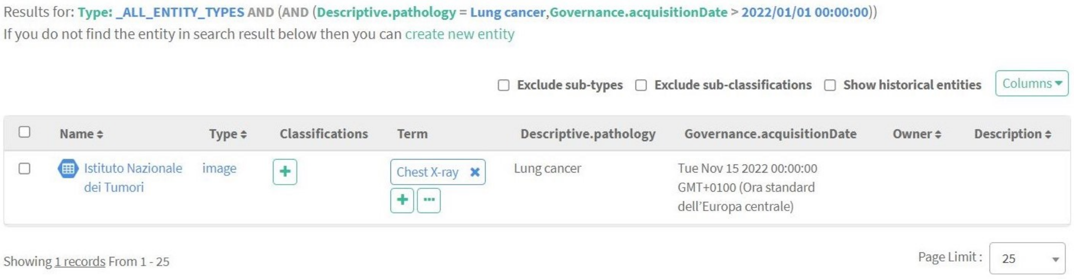
Attribute Filter

AND OR

Pathology (string) = Lung cancer

AcquisitionDate (date) > 2022/01/01 00:00:00

Figure 5.5: Second query attribute filter



Results for: Type: _ALL_ENTITY_TYPES AND (AND (Descriptive.pathology = Lung cancer,Governance.acquisitionDate > 2022/01/01 00:00:00))
If you do not find the entity in search result below then you can create new entity

□ Exclude sub-types □ Exclude sub-classifications □ Show historical entities

□	Name	Type	Classifications	Term	Descriptive.pathology	Governance.acquisitionDate	Owner	Description
□	Istituto Nazionale dei Tumori	image	+	Chest X-ray	Lung cancer	Tue Nov 15 2022 00:00:00 GMT+0100 (Ora standard dell'Europa centrale)		

Showing 1 records From 1 - 25

Page Limit : 25

Figure 5.6: Second query results

6 | Conclusions

The work presented in this thesis is a step in the design and implementation process of a storage system for medical data. The main objective of this work is to determine a metadata model functional for the storage, maintenance and use of medical data in a federated context.

The first step of the process was an analysis of the literature, in order to assess what is already available and what still needs to be developed. While this review brought to light a variety of metadata models, they were either too general or too tailored for the application they were designed for. This called for the development of our own model, elaborated expressly for this project and its needs. Nevertheless, the found proposals were still valuable, as they allowed for a better understanding of the current metadata scene, in particular in the healthcare domain, and it was possible to take a cue from them while building our model.

At this point, the project requisites were more accurately examined, focusing on the context, i.e. a federation of around 50 research and treatment institutes, the system architecture, i.e. the data lake, the different types of data, i.e. medical records, diagnostic images and signals, omics data, and the final uses, i.e. treatment and research, with the related requirements, e.g. high need for privacy while exchanging data to obtain the best possible results.

With all this in mind, it was possible to design an appropriate metadata model, consisting of different categories, each aimed at a particular functionality. More in detail, three classes were distinguished: governance metadata, focused on the regulation of data through privacy, security, quality and authenticity policies, ownership and access rights and other legal requirements, data lifecycle management, covering data provenance, from acquisition through all the transformations and transfers, data preservation and user tracking, and descriptive metadata, which describe data for identification and discovery purposes.

Once the model was finalized, it needed to be validated. To this end, we looked for an existing data catalog platform that would enable us to implement the metadata model.

Focusing on open source solutions and taking into account the requirements of the application, we identified 7 possible candidates, Apache Atlas, Amundsen, CKAN, Kylo, Magda, Truedat, and iRODS. Among them, Apache Atlas was chosen primarily for its flexibility, metadata management and overall features. After understanding better how Atlas works and can be used, a simple implementation of the metadata model was developed. This demo allowed us to verify that the chosen metadata are in fact helpful in the identification and retrieval of the datasets of interest for a given query.

In summary, a structured metadata model suitable for this application was designed and validated, considering the project requirements and the solutions and tools already available.

The model defined in this thesis is a first proposal of a metadata set in the described context of medical data management within a data lake for treatment and research purposes, seeing as nothing of the sort could be found in the literature. However, it still needs to be further developed. In particular, the metadata items should be better characterized, taking into account the specific features of the data and the system. The model should also be fully implemented, be it through Atlas or a better suited, possibly even ad hoc developed, tool.

Another important step that should be made is the identification of a minimum metadata set. To clarify, the metadata should be classified based on their relevance and usefulness, distinguishing between necessary, recommended and optional metadata. The necessary metadata, essential to the correct functioning of the system, make up the minimum metadata set.

Bibliography

[1] AHIMA. Rules for handling and maintaining metadata in the EHR. *Journal of AHIMA 84, no.5*, pages 50–54, 2013.

[2] Alation, Inc. What is a data catalog? data catalog features & benefits. URL https://www.alation.com/blog/what-is-a-data-catalog/.

[3] Amundsen Project Authors. Amundsen - Overview. URL https://www.amundsen.io/amundsen/.

[4] Apache Software Foundation. Apache atlas - Overview, . URL https://atlas.apache.org/#/.

[5] Apache Software Foundation. Apache atlas - Architecture, . URL https://atlas.apache.org/#/Architecture.

[6] H. Asri, H. Mousannif, H. A. Moatassime, and T. Noel. Big data in healthcare: Challenges and opportunities. In *2015 International Conference on Cloud Technologies and Applications (CloudTech)*, pages 1–7, 2015. doi: 10.1109/CloudTech.2015.7337020.

[7] Bluetab. User guide. URL https://docs.truedat.io/userguide.

[8] C. Chakraborty, B. Gupta, and S. K. Ghosh. Mobile metadata assisted community database of chronic wound images. *Wound Medicine*, 6:34–42, 2014. ISSN 2213-9095. doi: https://doi.org/10.1016/j.wndm.2014.09.002. URL https://www.sciencedirect.com/science/article/pii/S2213909514000445.

[9] CSIRO. Magda. A federated catalog for all of your data. URL https://magda.io/.

[10] S. Dash, S. K. Shakyawar, M. Sharma, and S. Kaushik. Big data in healthcare: management, analysis and future prospects. *Journal of Big Data*, 6, 2019. doi: 10.1186/s40537-019-0217-0. URL https://doi.org/10.1186/s40537-019-0217-0.

[11] M. Day and M. Patel. Metadata for images, Mar 2002. URL https://www.ukoln.ac.uk/metadata/filter/report/report.html.

[12] H. Fang. Managing data lakes in big data era: What's a data lake and why has it became popular in data management ecosystem. *2015 IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER)*, pages 820–824, 2015. doi: 10.1109/CYBER.2015.7288049.

[13] R. Gabriel, T. Hoppe, and A. Pastwa. Classification of metadata categories in data warehousing - a generic approach. *AIS Electronic Library (AISeL)*, 2010. URL `https://aisel.aisnet.org/amcis2010/133/`.

[14] R. Gartner. *Metadata. Shaping Knowledge from Antiquity to the Semantic Web.* Springer International Publishing, 2016. ISBN 978-3-319-40893-4.

[15] A. J. Gilliland. Setting the stage. In M. Baca, editor, *Introduction to metadata*, page 9. Getty Research Institute, Los Angeles, CA, second edition, 2008.

[16] J. Greenberg. A quantitative categorical analysis of metadata elements in image-applicable metadata schemas. *Journal of the American Society for Information Science and Technology*, 52(11):917–924, 2001. doi: https://doi.org/10.1002/asi.1170. URL `https://onlinelibrary.wiley.com/doi/abs/10.1002/asi.1170`.

[17] C. Haux and P. Knaup. Using FAIR metadata for secondary use of administrative claims data. In L. Ohno-Machado and B. Séroussi, editors, *MEDINFO 2019: Health and Wellbeing e-Networks for All - Proceedings of the 17th World Congress on Medical and Health Informatics, Lyon, France, 25-30 August 2019*, volume 264 of *Studies in Health Technology and Informatics*, pages 1472–1473. IOS Press, 2019. doi: 10.3233/SHTI190490.

[18] Informatica, Inc. What is big data?, . URL `https://www.informatica.com/services-and-training/glossary-of-terms/big-data-definition.html`.

[19] Informatica, Inc. What is metadata?, . URL `https://www.informatica.com/services-and-training/glossary-of-terms/metadata-definition.html`.

[20] W. H. Inmon. *Building the Data Warehouse.* John Wiley & Sons, New York, NY, first edition, January 1992. ISBN 978-0-471-56960-2.

[21] iRODS Consortium. iRODS technical overview. URL `https://irods.org/uploads/2016/06/technical-overview-2016-web.pdf`.

[22] N. Jahnke, M. Spiekermann, and B. Ramouzeh. Data catalogs - implementing capabilities for data curation, data enablement and regulatory compliance. Technical report, Fraunhofer Institute for Software and Systems Engineering ISST, August 2022.

[23] D. Laney et al. 3d data management: Controlling data volume, velocity and variety. *META group research note*, 6(70):1, 2001.

[24] A. LaPlante and B. Sharma. *Architecting data lakes.* O'Reilly Media, Sebastopol, CA, first edition, 2016. URL `https://learning.oreilly.com/library/view/-/9781492042518/?ar`.

[25] N. Miloslavskaya and A. Tolstoy. Big data, fast data and data lake concepts. *Procedia Computer Science*, 88:300–305, 2016. ISSN 1877-0509. doi: https://doi.org/10.1016/j.procs.2016.07.439. URL `https://www.sciencedirect.com/science/article/pii/S1877050916316957`. 7th Annual International Conference on Biologically Inspired Cognitive Architectures, BICA 2016, held July 16 to July 19, 2016 in New York City, NY, USA.

[26] J. Moehrke. Healthcare metadata, May 2014. URL `https://healthcaresecprivacy.blogspot.com/2012/05/healthcare-metadata.html`.

[27] Open Knowledge Foundation. A fully-featured, mature, and 100% open source DMS. URL `https://ckan.org/features`.

[28] Oracle Corporation. What is big data?, . URL `https://www.oracle.com/big-data/what-is-big-data/`.

[29] Oracle Corporation. What is a data catalog and why do you need one?, . URL `https://www.oracle.com/big-data/data-catalog/what-is-a-data-catalog/`.

[30] J. M. Pierson, L. Seitz, H. Duque, and J. Montagnat. Metadata for efficient, secure and extensible access to data in a medical grid. In *Proceedings. 15th International Workshop on Database and Expert Systems Applications*, pages 562–566, 2004. doi: 10.1109/DEXA.2004.1333534.

[31] Teradata Inc. Kylo is an open-source data lake management software platform. URL `https://kylo.io/`.

[32] US National Archives. Metadata in electronic records management, Nov 2016. URL `https://records-express.blogs.archives.gov/2016/11/21/metadata-in-electronic-records-management/`.

[33] US National Library of Medicine. Clinicaltrials.gov protocol registration data element definitions for interventional and observational studies, October 2020. URL `https://prsinfo.clinicaltrials.gov/definitions.html`.

[34] E. Zaidi, G. De Simoni, R. Edjlali, and A. D. Duncan. Data catalogs are the new black in data management and analytics. 2017. Gartner Research.

A | Appendix

Custom Atlas types definition.

```
{
    "entityDefs": [
        {
            "name": "patient_data",
            "description": "Patient data",
            "superTypes": [
                "DataSet"
            ],
            "typeVersion": "1.0",
            "attributeDefs": []
        },
        {
            "name": "image",
            "description": "Images",
            "superTypes": [
                "DataSet"
            ],
            "typeVersion": "1.0",
            "attributeDefs": []
        },
        {
            "name": "MRI",
            "description": "MRI images",
            "superTypes": [
                "image"
            ],
            "typeVersion": "1.0",
            "attributeDefs": []
        },
        {
            "name": "CT",
            "description": "CT images",
            "superTypes": [
                "image"
            ],
            "typeVersion": "1.0",
            "attributeDefs": []
        },
        {
            "name": "xray",
            "description": "X–ray images",
            "superTypes": [
                "image"
            ],
            "typeVersion": "1.0",
            "attributeDefs": []
        },
        {
            "name": "signal",
            "description": "Signals",
            "superTypes": [
                "DataSet"
            ],
            "typeVersion": "1.0",
            "attributeDefs": []
        },
        {
            "name": "ECG",
            "description": "ECG signals",
            "superTypes": [
                "signal"
            ],
            "typeVersion": "1.0",
            "attributeDefs": []
        },
    {
        "name": "EEG",
        "description": "EEG signals",
        "superTypes": [
            "signal"
        ],
        "typeVersion": "1.0",
        "attributeDefs": []
    },
    {
        "name": "EMG",
        "description": "EMG signals",
        "superTypes": [
            "signal"
        ],
        "typeVersion": "1.0",
        "attributeDefs": []
    },
    {
        "name": "omics",
        "description": "Omics data",
        "superTypes": [
            "DataSet"
        ],
        "typeVersion": "1.0",
        "attributeDefs": []
    },
    {
        "name": "genomics",
        "description": "Genomics data",
        "superTypes": [
            "omics"
        ],
        "typeVersion": "1.0",
        "attributeDefs": []
    },
    {
            "name": "proteomics",
            "description": "Proteomics data",
            "superTypes": [
                "omics"
            ],
            "typeVersion": "1.0",
            "attributeDefs": []
        },
        {
            "name": "transcriptomics",
            "description": "Proteomic data",
            "superTypes": [
                "omics"
            ],
            "typeVersion": "1.0",
            "attributeDefs": []
        },
        {
            "name": "radiomics",
            "description": "Radiomic data",
            "superTypes": [
                "omics"
            ],
            "typeVersion": "1.0",
            "attributeDefs": []
        }
    ]
}
```

List of Figures

List of Tables

List of Acronyms

AHIMA American Health Information Management Association

API Application Programming Interface

AVU Attribute-Value-Unit

CKAN Comprehensive Knowledge Archive Network

CSIRO Commonwealth Scientific and Industrial Research Organization

CSV Comma Separated Values

CT Computed Tomography

DAG Directed Acyclic Graph

DIKW Dada-Information-Knowledge-Wisdom

DMS Data Management System

DSL Domain Specific Language

DW Data Warehouse

ECG Electrocardiogram

EHR Electronic Health Record

ELT Extract, Load, Transform

ETL Extract, Transform, Load

FAIR Findable, Accessible, Interoperable and Reusable

GIF Graphics Interchange Format

GUID Globally Unique Identifier

IBM International Business Machines

iCAT iRODS Metadata Catalog

IoT Internet of Things

IRCCS Istituto di Ricovero e Cura a Carattere Scientifico

iRODS Integrated Rule-Oriented Data System

JPEG Joint Photographic Experts Group

MRI Magnetic Resonance Imaging

NoSQL Not Only SQL

PIDNUM Patient Identification Number

REST Representational State Transfer

SLA Service Level Agreement

SQL Structured Query Language

TIFF Tag Image File Format

UI User Interface

Acknowledgements

Having reached the end of my studies, an specifically of this thesis work, I would like to thank all the people who supported and accompanied me during these years and in particular in the last period.

First, I thank Prof. Cinzia Cappiello, Prof. Pierluigi Plebani and Prof. Letizia Tanca for the guidance and advice they offered me during these last months of work.

Then I want to thank my family, who supported and believed in me all through these (long) years of study, especially during the most difficult moments. My parents, without whom I could not have undertaken this journey, and my brothers, who offered me advice and support.

I also need to thank all my friends, with a special mention for Maria Teresa, Marta, Maria, Sara, and Giulia, always ready to help, listen with a sympathetic ear to my complaints, apprehension and excitement, encourage me, share breakfasts, dinners, lazy days, working days, and all the other moments we lived together (which somehow always end up including food).

Last, but by no means less important, my thanks go to Francesco as well, who in the last year has been standing by me, believing in what I can achieve even more than me, and giving me new serenity, energy and motivation to see the end of this phase of my life, while gifting me amazing experiences and memories.

Thank you all for being there for me, I hope I can be as good to you.