Temporal model to explore administrative healthcare databases

Johanne Bakalara

UNIVERSITÉ DE RENNES 1

THÈSE DE DOCTORAT DE

L'UNIVERSITÉ DE RENNES 1

ÉCOLE DOCTORALE N° 605
Biologie, Santé
Spécialité : *Analyse et Traitement de l'Information et des Images Médicales*

Par

Johanne BAKALARA

Temporal Model to explore Administrative Healthcare Databases

Thèse présentée et soutenue à Institut de Recherche en Informatique et Systèmes Aléatoires, le 23 juin 2022
Unité de recherche : IRISA UMR-6074 équipe LACODAM et REPERES EA-7449

Rapporteurs avant soutenance :

Anita Burgun	Professeur d'Université, Directrice de laboratoire, Sciences de l'information et médecine personnalisée,INSERM, Université Paris Descartes
John.H. Holmes	Professeur d'Université, Perelman "School of Medicine" University of Pennsylvania, DBEI (Department of Biostatistics, Epidemiology and Informatics)

Composition du Jury :

Président :	DAVID GROSS-AMBLARD	Professeur, Institut de Recherche en Informatique et Systèmes Aléatoires, Université Rennes 1
Examinateurs :	ANITA BURGUN	Professeur d'Université, Directrice de laboratoire, Sciences de l'information et médecine personnalisée, INSERM et Université Paris Descartes
	JOHN.H. HOLMES	Professeur d'Université, Perelman "School of Medicine" University of Pennsylvania DBEI (Department of Biostatistics, Epidemiology and Informatics)
	FLEUR MOUGIN	Professeur d'Université, Institut de Santé Publique, d'Epidémiologie et de Développement, Université de Bordeaux
Dir. de thèse :	EMMANUEL OGER	Professeur d'Université et Professeur Hospitalier Institut de recherche en santé, environnement et travail, Université de Rennes 1
Co-dir. de thèse :	THOMAS GUYET	HDR, Institut National de Recherche en Sciences et Technologies du Numérique

Table of Contents

INTRODUCTION

The rapid expansion of health data and the millions of patients they contain represents an unprecedented opportunity to improve public health. The massive amounts of records provide a lot of information and offer a wide spectrum of possibilities for epidemiological studies which aim at improving public health. For example, in France since 2008 French health insurance stores health care reimbursements and keeps them in a database. These data are extensive as they contain almost 99% of the French population with individual information: age, sex, location and health reimbursement information: drug deliveries, medical acts or medical visits and hospitalizations.

Even if the initial use of this database is the reimbursement of care products, epidemiologists have attempted to use these data for public health studies. The first large-scale study [Wei+10; Gua] with a high impact started with the suspension of benfluorex (*i.e.* mediator). This drug was controversial, gradually withdrawn from the market in Spain and Italy, and authorized in France until 2009, when it has been withdrawn from the market following the results of the study. This study compared French diabetic patients exposed to benfluorex with French diabetic patients not exposed to benfluorex. It concludes on a significantly increased risk of heart valve disease that can lead to death for the population treated with benfluorex. It became the first public study without conflict of interest, based on a large population, which concluded on the danger of benfluorex treatments. The example of benfluorex is a small example amid of the studies conducted by epidemiologists, but it is the precursor of the epidemiological studies conducted thanks to the French *Administrative Healthcare Databases* (AHDB), called *SNDS* (Sytème National des Données de Santé) [Sca+19].

Epidemiology aims to survey the state of public health, such as the identification of risk factors for a certain disease (*e.g.* the use of mediator for cardiovascular disease [Wei+10]) or the survey of a disease (*e.g.* survey the number of veinous trombo-embolism in Brittany [Og+00]) or the investigation of an epidemic (*e.g.* survey the evolution of covid-19). In the dictionary of epidemiology, Miquel Porta [Por14] defines epidemiology as the evaluation of the risk or benefits of an event on a population where these events are *any*

characteristic, or other definable entity, that brings about a change in a health condition or other defined characteristic. To conduct studies, the first challenge for epidemiologists is to find suitable health data. These data can be collected for the study (clinical data) or are already available thanks to the existence of registers (*e.g.* R.E.I.N[1] register in France), cohorts (*e.g.* Constances [Con]) or Administrative Health Care Databases (*e.g.* SNDS [Tup+17] in France or GePaRD [PA08] in Germany).

The clinical data seems ideal as they contain a lot of details such as medical procedures, medical reports, precise quantitative data, and even social data on the lifestyle of patients (diet, smoking etc...). However, no information system automatically acquires clinical data. Collecting such information to answer one specific epidemiological question requires a lot of time and is very expensive while existing AHDBs already contain a large population coverage. The existence of an exhaustive population issued from AHDB is a strong asset for the reliability of an epidemiological study. For benfluorex, the data were already available at a large scale whereas collecting clinical data would have required a time-consuming experimental setup and could only have concerned a small number of patients.

On the other hand, the extraction of a suitable population for epidemiological analysis from a large dataset of patients is a complex computer science task. In practice, epidemiologists describe a *care pathway* defining the population to extract, taking care to use the information provided by the database. The dataset of patients is a dataset of individuals where each individual has a set of medical events called a *care sequence*. Then, the description must be translated by a computer scientist specialized in databases to find patients who have followed this care pathway.

Let us illustrate a care sequence on Figure 1 on page 7 and a care pathway on Figure 2 on page 7. Figure 1 on page 7 illustrates a patient who has a care sequence composed of the following events : a drug delivery on the date "2021-01-30" as well as a visit to a general practitioner on the same date and then a laboratory exam the day after, then a two-day hospitalization from "2021-02-02" to "2021-02-03" and then three drug deliveries on the dates "2021-02-04", "2021-03-01" and "2021-04-01".

[1]The Epidemiology and Information Network in Nephrology (REIN) is an information system concerning the problems raised by the replacement therapy of chronic renal failure in the public health field `https://www.agence-biomedecine.fr/R-E-I-N-Reseau-Epidemiologique-et-Information-en-Nephrologie`

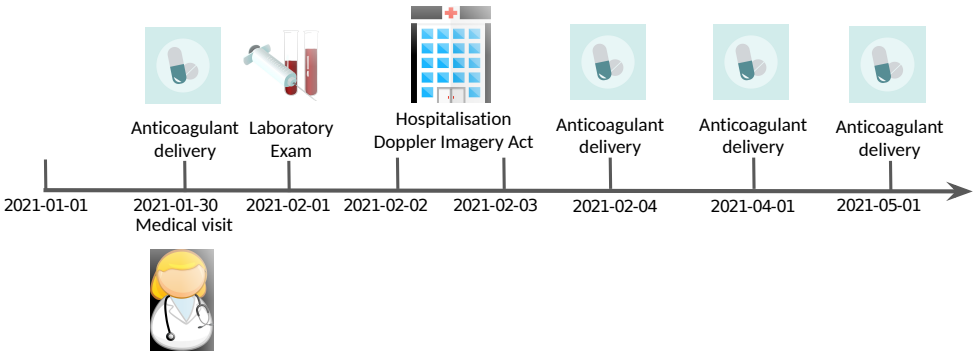


Figure 1 – Illustration of a care sequence

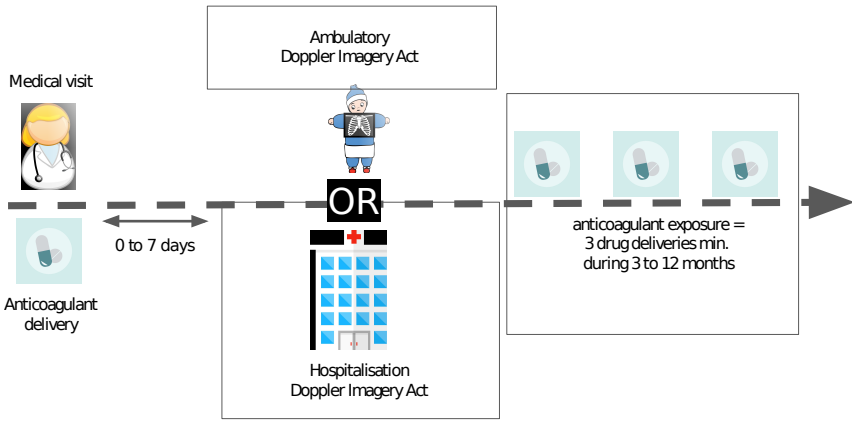


Figure 2 – Illustration of a care pathway also called phenotype to find patients suffering of VTE

Let us now imagine an epidemiologist who extracts a population suffering from venous trombo-embolism (VTE). He seeks to describe a typical care pathway for patients with VTE as illustrated in Figure 2 : A medical act (Doppler imagery) during or prior to anticoagulant deliveries for 0 to 7 days and delivery lasts a minimum of 3 months and a maximum of 12 months. The Doppler imagery can be executed in or out the hospital.

The care sequence presented Figure 1 verifies the care pathway presented Figure 2. Indeed, the care sequence contains an anticoagulant delivery followed by the Doppler imagery in the hospital, followed by three anticoagulant deliveries. Thus, the epidemiologist should find this patient in his extraction.

This thesis aims to query care sequences to extract only those verifying a given care pathway by proposing a formal approach combining expressiveness and efficiency. The notion of care pathway as a criterion for selecting a population is called *computational*

phenotype by Richesson *et al.* [Ric+20]. In epidemiology, it is used to formalize phenotypes with decision diagrams. In computer science, this formalism must be seen as a query.

To detail the notion of phenotype, we reuse the example of VTE Figure 2 on the preceding page. This example is issued from a concrete use case. It will illustrate several concepts throughout this thesis. Below, physicians propose phenotypes descriptions to identify patients suffering from VTE [Og+00].

Example 1 – phenotype to identify patients suffering of Venous Thromboembolism (VTE) [Og+00]

In clinical practice facing a suspicion of VTE physicians first prescribe antithrombotics and then confirm or not the diagnosis through specific medical procedures: e.g. *Doppler ultrasonography or CT-scan. Patients with suspected Pulmonary Embolism are often hospitalized whereas patients with suspected Deep Vein Thrombosis (DVT) are managed on an outpatient basis. On the one side, if the DVT suspicion is confirmed, antithrombotic deliveries continue for 3 to 12 months (once per month). Hence, the diagnosis (through the same medical procedures as above) is preceded or followed by initiating an anticoagulant treatment within a time window of at most 7 days. On the other side, Pulmonary Embolism suspicion leads to hospitalization during which medical procedures are performed to confirm the diagnosis and then anticoagulant delivery is observed only after the patient comes back home.*

There are 2 important dimensions in these descriptions:

- use of ontological concepts ("Doppler imagery act"/"anticoagulant"/"vascular specialist"): the code of the medical act provided to a patient is given, but is more precise than a criterion "anticoagulant drug" and symbolic domain knowledge is required to reconcile both. Here, "Anticoagulant" refers to a class of drugs that is described in the ATC taxonomy, an international classification.[2]

- use of temporal constraints between events ("during or prior 1 to 2 months", "within a time window of 1 to 2 days"): the temporal order of care and numerical duration/delays specify the temporal organization of the events.

This thesis will therefore propose a formalism to express phenotypes with these two dimensions while ensuring that the queries resulting from these formalisms are efficient.

[2] ATC: Anatomical Therapeutic Chemical Classification System.

In our context, data is not simply binary, textual information or number, it is an entity that can be part of several named groups also treated as entities. One of the major challenges in the medical field is the semantically rich use of data. Data have several levels of specificities, they are typed, classified, and sometimes refer to medical knowledge. One of the most telling examples that we will consider in this thesis is the classification of a drug provided by the ATC taxonomy: the element "box of Calciparine" is an entity belonging to a group: the group of "Heparine", itself belonging to the group "Antithrombotic agents", itself belonging to the group "Nervous system". Many international medical ontologies classify medical concepts.

Where there are no ontologies available, to select individuals who had been dispensed an anticoagulant we have to manually search (make a list) of each box sold on the market that is an anticoagulant. For example, instead of using the group "Antithrombotic agents", we list drug boxes of anticoagulants: previscan 20mg cpr 30, indione 50mg cpr 20, coumadin 10mg cpr 25, coumadin 10mg cpr 30, coumadin 5mg cpr 30, etc... In the VTE study, there is a list of nearly 130 boxes of anticoagulants.

The first objective of this thesis is to connect AHDB to the all drugs of its daughter classes. In a more general context, the Semantic Web enables to query data with Knowledge. In our case, medical knowledge from ontologies created by the community or by the owner of the study, one thinks of the ICD-10 diagnosis classification or even the SNOMED-CT ontology.

In Chapter 5 on page 69 we develop an ontology of the SDNS database. This ontology defines concepts related to health data, such as the concept of drug delivery or hospitalization. These concepts enable us to link this SNDS ontology to international ontologies and thus exploit existing medical knowledge. Moreover, this representation proposes to explicitly integrate the concepts of care sequence and care event. This helps to answer the following challenge aiming at querying care pathways on care sequences.

The second objective of this thesis is the expression of phenotypes, explained in Chapters 4 on page 49 and 6 on page 83. As seen previously, a phenotype must be able to express medical events linked to ontologies but also linked by temporal constraints. In our context, temporal constraints are temporal intervals that designate the time interval between one event and another event. For example, for VTE, we are interested in Doppler imaging followed within a week by an anticoagulant delivery. This constraint is essential, an anticoagulant delivery twelve months later could refer to another pathology than VTE.

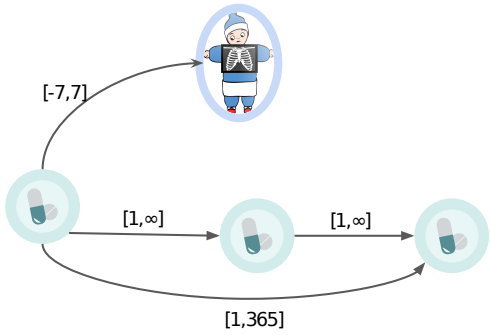


Figure 3 – Illustration of a Chronicle represented a VTE phenotype

In practice, it is easy to look for delivery of anticoagulant on the date “2021-01-30”", but it is more complex to find a delivery that took place after a hospitalization and that this same hospitalization was preceded by a visit to a specialist.

With the current query tools used (detailed in the state of the art Chapter 2 on page 19), there are no generic temporal models to formalize the structure of a phenotype. It is the computer scientists who work with the epidemiologists who build a query or a set of queries translating the description of a phenotype. Then, the construction of a query is time-consuming and requires a good knowledge of the database

In Chapter 4 on page 49, we propose the Chronicle model [DL07] to formalize the notion of phenotypes. This model defines event occurrences linked by temporal constraints. The existing algorithm of Chronicles enables to check sequences verifying the temporal constraints and events. We extend the event notion with taxonomy concepts and thus be able to exploit the information contained in the care sequences and international ontologies. This formalism is all the more interesting as it can be represented by a graph of constraints which constitutes a visual. A simple example of a Chronicle is shown in Figure 3. The Chronicle represents the phenotype of Figure 2 on page 7 where it designates that an anticoagulant delivery is preceded/followed between 7 days before and 7 days after by a Doppler imagery and this same anticoagulant is followed by two other anticoagulants within a year (1 to 365 days).

Finally, we have created the HyCOR tool to perform the extraction of care sequences verifying a Chronicle. It is a hybrid tool combining ontology management of the Semantic Web and the efficiency of Chronicles algorithms. HyCOR is very powerful, it finds sets of patients verifying a complex phenotype, using ontology and temporal constraints, in a few seconds.

Chapter 1

Problematic - Extract care sequences verifying a phenotype

In the introduction we have shown the two main axes of the challenge of this thesis: the challenge concerning the representation of the data and the challenge concerning the querying of these data with temporal phenotypes. These two main axes are linked as the content of the phenotype depends of the available data. To introduce the problematic of this thesis, we will start by describing the French database that serves as a framework. Then we will discuss the issues surrounding the notion of phenotype by giving a definition of a phenotype that serves as a formal support to query the data. Finally we will describe the problem statement that will guide us throughout the thesis.

1.1 Conducting pharmaco-epidemiological study with Administrative Healthcare Databases (AHDB)

The advantages of using AHDB for epidemiological purposes have been demonstrated in many studies but its use remains a major challenge. Indeed, contrary to EHR, registries, and cohorts data, a main issue to handle is: on one hand we can observe a large part of a population with a wide range of health events (and administrative information) but on the other one, there are no clinical criteria such as diagnosis and outcomes of care. This lack of diagnosis implies a difficulty in describing the population of interest. Indeed, it will not be possible to designate a population affected by deep venous thrombosis using a diagnosis stored in the database, but epidemiologists will have to describe a phenotype that is indicative of this disease. This phenotype will therefore be of a certain complexity and will have to be based on the available information.

In the article [Tup+17] Tuppin et al. detail the available data, these are described in Table 1.1. As a comparison, some other data, which would be useful for epidemiologists and clinicians, but not available in SNDS, are also presented in this table. It shows the gap between available data and the desired one. The SNDS contains French patients individual information:

- date of birth,
- sex,
- location,
- date of death

and health reimbursement information with dates of care:

- drug deliveries,
- medical acts,
- medical consultations,
- hospital stays

All the information are coded in a specific nomenclature/taxonomy but it **does not contain medical reports nor diagnosis out of hospital**. As we can see in Table 1.1, SNDS specifies the information of the procedures: the date of care and the code related to the procedure issued from the CCAM taxonomy, but the result of the procedure is not recorded. Patients with a broken arm will have the following record: arm radiology at a specific date but not the diagnosis of "broken arm". This lack of information is crucial in the use of AHDB data.

The introduction highlights that epidemiology is based on the study or comparison of populations. We propose to detail the different steps that are used today to conduct a pharmaco-epidemiological study in the SNDS.

We have identified four steps illustrated on Figure 1.1 on the facing page and detailed below:

1. ensure security and anonymisation of database and provide an access
2. pre-process data
3. select an interesting population
4. finally conduct a statistical study

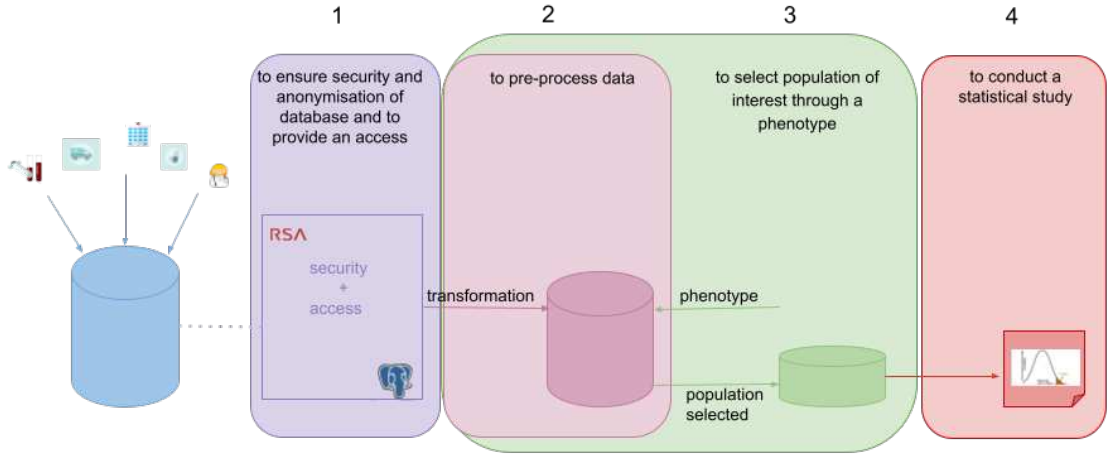


Figure 1.1 – Process to conduct an epidemiological studies on MADB from the data collection to the statistical analysis

Firstly, the data are provided according to a relational data schema [1]. The data are already pseudo-anonymized. In our context, there is no management of data acquisition, which means that there is no control over the available data, everything is based on the health insurance reimbursement system.

The second step concerns the usual data manipulation needed before selecting a population of interest. For example, in the pharmaco-epidemiological context, looking for patients exposed to a drug within a time window requires to create a new table in SQL(*view*) containing the exposure sequence for each patient. In the epidemiological context O'Connor *et al.* [OCo+09] explained that *these data-processing steps are often customized to a particular analysis and study plan, so the database methods are difficult to reuse across research projects.* This is especially true in the SNDS as data are initially collected for a reimbursement purpose and now used for epidemiology purposes.

The third step consists of selecting a population of interest where the population has a common phenotype. The difficulty for epidemiologists lies in the description of the phenotypes using available information. And the difficulty for the computer scientist lies in the construction of a complete query to select patients verifying this phenotype.

[1] https://documentation-snds.health-data-hub.fr/

Lastly, the pharmaco- epidemiologists can conduct the statistical analysis once selected the population of patients verifying a phenotype. The contributions of this thesis fall under points 2 and 4.

1.2 Phenotyping in pharmaco-epidemiology on AHDB

In the previous section, we explained there is a lack of diagnosis in AHDB and, on the contrary to EHR data, no information concerning results of clinical examinations is available. Selection of a population is consequently more difficult. More specifically, we do not reach the granularity information of a clinical data register. Epidemiologists have to overcome this lack of information [Pal17] to select population/subgroups of patients with a common set of criteria and more specifically, patients who experienced a health pattern called phenotype. In the book [Ric+20], Richesson *et al.* rethink the modern definition (see definition) of phenotype (*i.e.* computable phenotype) and give an overview of considerations for identifying, defining, and evaluating computable phenotypes, focusing on standardization efforts within the NIH Health Care Systems Research Collaboratory.

Definition 1 *A phenotype is a specification for identifying patients or populations with a given characteristic or condition of interest from EHRs using data that are routinely collected in EHRs or ancillary data sources. A computable phenotype definition consists of data elements and logical expressions (such as AND, OR, and NOT) that can be interpreted and executed by a computer. In other words, the syntax defining a computable phenotype is designed to be interpreted and executed programmatically without human intervention. Computable phenotype definitions often rely on value sets—lists of codes from standardized medical vocabularies that indicate a condition, drug exposure, or other clinical phenomenon of interest*

Even if the definition concerns EHR data, the definition of a phenotype does not change for AHDB or health data in general [Moo+19]. The Phenotype Knowledge Base website *PheKB* [Kir+16], a collaborative environment, references algorithms to identify patients from different databases through several characteristics. The objective is to facilitate the comparison of the results between different datasets concerning the same phenotypes. The French alternative is the redSIAM[2] which proposes algorithms guiding the selection of patients facing certain health cares in the SNDS. These communities proposing phenotypes

[2]https://www.redsiam.fr/

Table 1.1 – Information available or not available in the SNDS [Tup+17]

Available information	Nomenclature/ Taxonomy	Limited or no information
Patients		
Sex		Other socioeconomic characteristics: employment, type of job, income, marital status...
Date of birth		
Date of death		Risk factors: smoking, alcohol, sedentary lifestyle, nutrition, family history...
Causes of death		
CMUC, ACS		
Adult disability allowance (AAH)		
State medical aid (AME)		
Town of residence	Insee code	
Geographical social disadvantage index		
ALD	ICD-10	
Daily allowances (if > 6 months)	ICD-10	
Occupational diseases	ICD-10	
Disability	ICD-10	
Reimbursements with dates of care		
Drugs, Clinical pathology	ATC, NABM	Prescribed dosage of drugs (information limited to the quantity dispensed, requiring estimation of the daily dose)
Procedures, products and benefits	CCAM, LPP	Results of clinical examinations: blood pressure, body mass index... + Matching of treatment with test results (HT, lipids. . .) Laboratory test results, histology, pathology, radiology...
Health care professionals (type, place. . .)		Reasons for or diagnosis of medical or paramedical consultations
Hospitalisations (public and private) with dates		
Primary, related and associated diagnoses	CCAM	
Procedures, products, benefits	CCAM, LPP	
Excess expensive drugs and devices		Drugs delivered in hospital or long-stay wards, other than those on the excess list (SSR, PSY, Ehpad with internal pharmacy...), or in the context of lump sum payments (health clinics, health care networks, medical examination centres, etc.) whether or not the drugs are taken (information limited to reimbursement, with no data on adherence)
Stays billed directly to national health insurance (private clinics or associations, medical and social welfare centres...)		
Outpatient consultations		
Hospital dispensing		

are of primary importance. All the previous references explained that sharing phenotypes improves the reproducibility of the studies from multiple health databases, from multiple sites. In the best world, the algorithms proposed to express phenotypes ensures efficiency and ensures *that populations identified from different healthcare organizations have similar features, or were at least identified in the same way* [Ric+20].

We can notice that even for a simple case, the description given by Richesson [Ric+20] is incomplete, because phenotype description of cannot be limited to logical expressions (such as AND, OR, and NOT). For example, the use case of VTE presented in the introduction can not be expressed with this definition. The ontological concepts and use of temporal constraints are crucial in epidemiology, yet they are still topical issues in computer science. Futhermore, the temporalities in databases problems are known to deteriorate efficiency. To solve this problem, we are interested in linking the pattern mining domain to the semantic web domain.

1.3 Problem statement

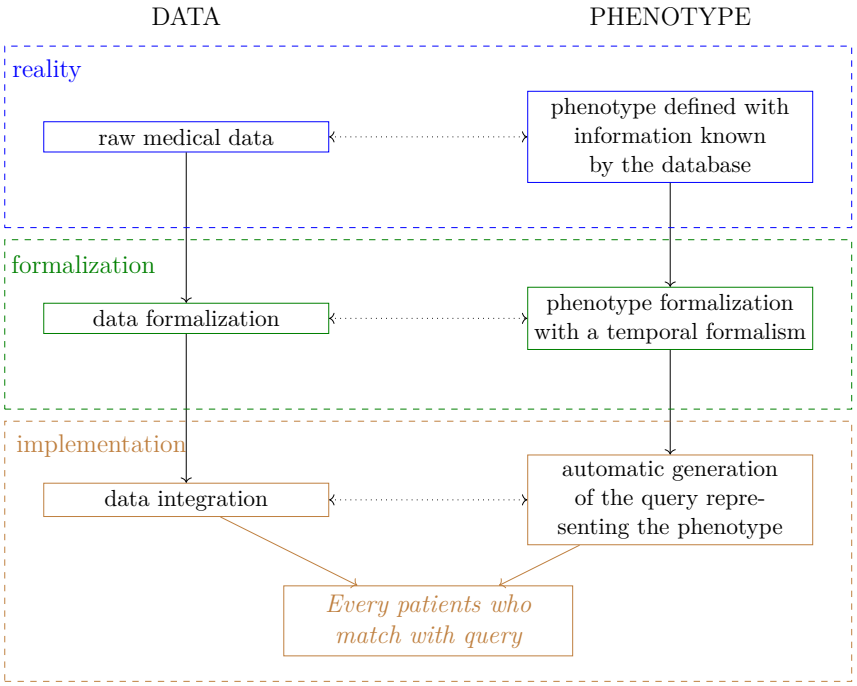
The VTE use case illustrated the problem of formalizing phenotype. The challenges we confront are to ease the extraction of population facing a given phenotype. It takes into account the temporal and ontological aspects. For the sake of generality, a formalism has to specify phenotype patterns of a broad range of care studies. Our work focuses on two aspects:

- the data representation
- the expression of temporal constraints in the description of phenotype with use of ontological concepts

The challenge we are facing is summarized in Figure 1.2 on the next page. Care sequences are on the left and phenotypes on the right. The overall goal is to find care sequences verifying a given phenotype.

In blue, the reality level: the epidemiologists define the phenotype with the information available in the database , while the database stores patient reimbursement information. In green, at the formalization level, data and queries are specified in a formalism. The latter must express a maximum amount of information to represent the complexity of

Figure 1.2 – Problem statement from data integration to temporal model to query data



the phenotype. In brown, the implementation level, data are integrated and we process them with a query automatically build from the phenotype. Then, we obtain all matching query patients. In current practice, SNDS stores and queries data with classical RDBMS (Relational Database Management Systems) such as MySQL and PostgreSQL with a given database schema[3]. There is no formalization step, a pre-processing data step is needed for each study. Moreover, the phenotype building is not covered by a defined formalism and specific algorithms have to be developed by computer scientists for each phenotype. Formalization will eliminate this pre-processing data step.

Finally, **the objective of the thesis is to propose a formal, efficient and strong theoretical foundation framework for querying care sequences with phenotype in the context of pharmaco-epidemiology.** As seen before, specifying phenotype requires manipulation of: temporal qualifiers (time constraints and time window), medical data, and symbolic knowledge (ontologies). The ideal formal framework should capture these dimensions, enable intuitive queries to be expressed for a wide range of pharmaco-epidemiological studies and be computationally efficient. It is of paramount importance to base choices on solid theoretical foundations. Expressiveness and efficiency are known

[3]https://gitlab.com/healthdatahub/dico-snds

Use taxonomies	
Expressing metric temporal constraints	
Use all types of criteria (diagnosis, medical acts, hospitalization, etc.)	

Table 1.2 – Criteria of evaluation for expressivity of the temporal model

A few minutes per phenotype for a large dataset	
No transformation of data at each phenotype	

Table 1.3 – Criteria of evaluation for efficacity of the temporal model

to be antagonist objectives [Lev86]. A theoretical approach would make possible future improvements possible and facilitate its application to a broad range of contexts (*i.e.*, various databases, queries).

To evaluate temporal model we compare temporal models around two main criteria:

- Expressivity: which criteria and conditions between those criteria can be used in phenotype ?
- Efficiency: expectations on execution time concerning the phenotype and the pre-processing data

This approach is original. Indeed, we propose to change the data analysts' habits by removing a time-consuming pre-processing phase. Few elements in literature address the problem in this sense, consequently we have few means comparison. We will see in the state of the art that other problems concerning health data have been addressed in the literature, neither the question of the phenotype expressivity, nor the problem of effectiveness according to the complexity of the proposed phenotype. Table 1.2 details the objective to reach expressivity criteria which concerns the type of determinant that can be used in the phenotype definition. Table 1.3 details the objective to reach efficiency criteria. As there are no references in the literature, we set ourselves a goal of a few seconds to find all the patients verifying a given phenotype by eliminating the pre-processing phase of the data. This includes a contribution to the data model adapted to query phenotypes.

Chapter 2

State of the art

In the previous chapter, we have defined the need to formalize the notion of phenotypes that will allow epidemiologists to define their target populations. They link inclusion criteria organized according to ontologies and linked together by temporal constraints. The criteria used in the phenotypes are available in the database. We are interested in the representation of data, and of the models that might be used. Then we query such data, taking into account ontological and temporal aspects. Lastly, we have to formally describe the temporal constraints that guides the phenotype content and which temporal model might adapted to represent them. We articulate the state of the art in two parts that we will progressively connect during this thesis:

- Section 1: to represent/store dated data with ontologies
- Section 2: to query them with queries containing temporal constraints and exploiting the information contained into ontologies

And those in a context related to an application on a health care database. Thus, we propose a state of the art dealing with these two aspects of the problem:

- **Data integration and query languages** to explore the different data management systems used to integrate and query medical data.
- **Temporal models** to explore formalisms used on timed data reasoning.

This state of the art should provide an overview of health care data management as currently done. We will first present the main database management systems used to integrate and query health care data. We compare them on several aspects:

- the type of data that can be integrated
- the database schema that orders them

- the data semantics and the associated query language

We will see that the data query systems are often more adaptable for EHR data than HADB data. We also summarize more technical aspects concerning the data format, the type of temporal unit linked to the data and whether it is possible to link ontologies issued from other resources to the data. We should consider the kind of temporal unit, since it guides the queries that can be sent on the data (*i.e.* intervals, fixe point). The container of the queries is guided by the needs in the expression of the phenotypes. Time management is generally a central point of literature at several levels: how to describe a mechanism that depends on time, how to "query/observe" it.

These questions have been treated in the field of temporal domains which constitutes our second part. At the end of this state of the art, the reader should have the relevant knowledge about the medical data current management and the existing types of temporal models. There are few bridges between these two parts, so we will deal with them in the rest of the manuscript.

2.1 Data integration systems and associated query languages

Data integration is the process of combining data from several resources following a precise format. Choosing a system to integrate data is choosing a format and a database schema to give a unified view of data and simplify the access [Koz+15].

Indeed, Venot [VBQ+13] explains the need to formalize medical language, the importance to define concepts to link reality with verbal expressions and to use terminology to have a common vocabulary. The use of ontologies is increasingly recommended to formally define knowledge models. Several sources insist on the need for specific semantic to use medical data from [TV12; SS06]. Both platforms *BioPortal*[1] and *OBO foundry*[2] ccollect more than 800 ontologies. They are open and collaborative [Whe+11; Smi+07] and they ensure a common syntax and relations based on ontology models. These points lead us to avoid any ambiguity and bias in medical database use.

We decided to distinguish three model categories to integrate data: Relation model (SQL), Common Data Model and Semantic Web model.

[1] https://bioportal.bioontology.org/
[2] http://www.obofoundry.org/

Here we propose a state of the art of addressing data format, data schema and query languages including time and ontologies management. détailler format de données, schéma de données et langage de requête. Expliquer ce découpage.

2.1.1 Relational databases - Temporal databases

Temporal databases and querying tools [Sno+86] extend the notion of database to timestamped data. This family encompasses the temporal extension of relational databases. Michael H. Böhlen *et al.* [Böh+17] give an overview of the state-of-the-art research results and technologies for storing, managing and processing temporal data in relational database management systems [JS99]. While designing a temporal data model, several aspects have to be considered, such as:

- different time dimensions or temporal aspects
- different timestamp types
- different forms of timestamps

In our context, time data only concerns *valid time* which captures the time when a fact is true. For example, the time when the drug delivery occurred. We could capture *transaction time* which would capture the time when the data has been recorded but it has no interest in the pharmaco-epidemiology context. We are not interested in the transaction time which focuses on storing data history.

We can highlight several types of timestamp which have been implemented in DBMS as: time points, time intervals/period and time element. Time points is the most basic where timestamps are atomic values and can be compared easily with the operators $=, \neq, >, <, \leq, \geq$. Time intervals can be continues or disjuncts (set of values) and can be compared with Allen's relations. Time element can define tuple/attribute timestamped with a finite union of intervals.

Various temporal query languages [Tan+93] have been proposed which extend SQL query language such as TQuel, TSQL2, SQL^{T}, IXSQL,SQL/TP, TOLAP, TOSQL, HSQL, ChronoSQL, SQL/Temporal, and so on. Seo-Young Noh [Noh04] compares TSQL2, IXSQL, SQL^{T} and SQL/TP. We resume the most cited in literature in few points by comparing them on the data model and the query language extension which are focus on temporal aspects.

- the data model of TSQL2 uses interval data type [Sno12] with time operators issued from Allen's interval relations [All83].
- IXSQL [LM97] is syntactically and semantically upwards consistent with SQL2. Data model uses *interval date* data type and provides query facilities to manipulate tables. The SQL-syntax is extended with 2 new operators: REFORMATS AS (reformats a table to a sequence of columns) and NORMALIZE ON (reformats a column list). It also proposes a new function: *unfold* which changes interval based structure to a point based structure.
- SQL/TP [Tom97] data model is a point-based model. Temporal attributes has a new data type for the timestamps which can be time instant or set of contiguous time instants. It conserves the SQL syntax omitting subqueries nested in the *where* clause and the *having clause*.
- SQL^T is based on a point-based temporal data model and on explicit time queries. The data model of SQL^T is similar to that of SQL/TP.

The temporal databases domain aims at manipulating dates into databases. It offers facilities to integrate such data and to manipulate them. A wide variety of DBMS have proposed SQL extensions which make this tool accessible, but performances of query with time is hardly evaluated. We notice that relational databases including time management do not include ontology management- a central element in medical context. However, the community of Common Data Model explored many solutions to deal with medical databases including medical ontologies. Most of these Common Data models use relational database models where data semantics and query language are inspired by SQL. We will detail them in the next section.

2.1.2 Common Data Model

In the medical area, common database schemas such as OMOP[3] (Observational Medical Outcomes Partnership - Common Data Model) have been proposed, but also data warehouses i2b2[4], openEHR[5], EHOP[6] and PCORnet[7] [WP19; Hri+15; DEL+15] which

[3]https://www.ohdsi.org/
[4]https://www.i2b2.org/
[5]https://www.openehr.org/
[6]https://centrededonneescliniques.univ-rennes1.fr/en/le-systeme-ehop-version-anglaise
[7]https://pcornet.org/

aim to achieve interoperability between various health analysis bases, whether clinical or AHDB. These common data models are developed with specific platforms to observe and query data. Database schemas are mostly developed on relation model schema but are not directly linked to temporal databases.

We detail each of them around a few elements:

- Medical data supported;
- Data format used;
- Time type (date, interval, point fix,...);
- Data schema;
- Data semantics;
- System used to query the data.

i2b2

- *Data:* Much of the discrete information that is stored in the EHR can be loaded such as: demographics (age, gender, race, etc.), diagnoses (ICD-9), allergies, procedures, medication orders, lab results and vitals.
- *Data format:* xml
- *Time type:* interval of dates
- *Data schema:* i2b2 star schema see Figure 2.1 on the following page where users define concepts
- *Data semantics:* only data coded as hierarchy (ICD-10). Non coded in terminology standard hierarchy data have to be organized, using other methods
- *Query data:* not "query language" but an advanced interface, with notably temporal query.

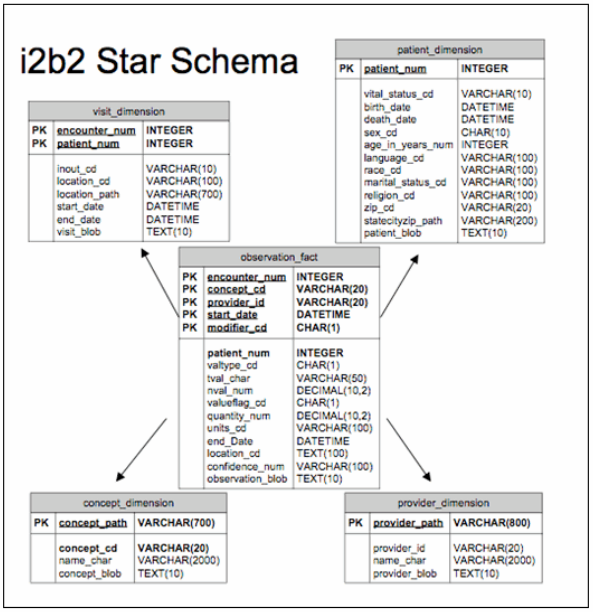


Figure 2.1 – i2b2 star schema

openEHR

- *Data:* EHR data, same as i2b2
- *Data format:* ADL, an human-readable and computer-processable language, gives archetype tool syntax/xml/json/yaml/MOFAS
- *Data schema:* each user designs their own template
- *Data semantics:* nearly the same as i2b2, only terminology. On- going ontologies OWL
- *Time type:* Date/Time types + any combination of values and interval
- *Query data:* AAQL : a declarative query language specifically developed for expressing queries used for searching and retrieving the data found in archetype-based repositories. The AQL specification is not bound to a specific Reference Model, but to use a given RM, it should comply with some requirements: it should be an Object Oriented Model and should follow the dual-model approach

PCORnet

- *Data:* daily data about more than 70 million people across the U.S.A. Include prescribing, procedures, dispensing, medication administration (includ- ing electronic medication administration and/or barcode)
- *Data format:* Relational data model (SQL)
- *Data schema:* see Figure 2.2, inspired from OMOP and W3C recommendation
- *Data semantics:* user chooses terminology
- *Query data:* PCORnet is first of all, a data warehouse. Query such data is managed by the platforms under prior request. Their web site states that query *can take from weeks to months depending on the complexity of the request, availability of the requestor to address questions, and number of other queries in the queue.*

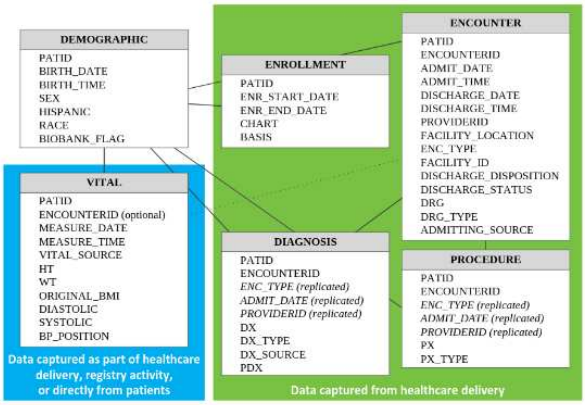


Figure 2.2 – PCORnet schema

OMOP

- *Data:* The OMOP schema aims at adapting a wide variety of data sources. It accommodates with both administrative claims and EHR data. The objective of this general schema is to allow assessing and analysing multiple data sources concurrently with a common data standard
- *data format:* Relational Model, mainly adapted for SQL but might be adapted to other formats provided that they can represent relational models and the types defined by the database schema.

- *database schema:* The book of OHDSI[8] efines an overview of the database schema where classic concepts of standardized clinical data (blue box) are defined with their derived elements (purple box). The orange box defines standardized vocabularies where the user can create his own concepts. Concepts are used to standardize the content of the records, it aims to include ontology.

- *Query data:* Standard queries against CDM may vary for local instances and date/-datetime configurations. This CMD is optimized for identifying populations with some healthcare intervention and outcomes and to characterize these population from parameters ; these criteria do not include time constraints and ontologies use. Usual relational database management systems (RDBMS) can be used to query data over OMOP database schema.

To fit real data with a common data model is a hard task. Data can be modified or lost when transformed into a new model [CB17]. Constraints absence makes it difficult to validate and reproduce data [CB17; Dan+19].

In practice, all these data warehouse have their own process to fit raw data into common data model, and make impossible any comparison and atomisation query. In fact, common models do not ensure the semantics, for example ontologies in i2b2 and OMOP do not propose formal semantics of the data but it is personalized during the data integration. [OCo+09] even explains that *A serious shortcoming of databases is that their representation of data, such as in a relational model, does not adequately support the representation of important biomedical domain concepts, such as hierarchies and complex representations* and discuss about a new general method to integrate *temporal domain knowledge* and a *temporal database schema* for clinical trial systems.

urthermore, time is not central in these data representation while the representation choice guides the type of query than can request data. Lastly, the data representation is closely linked to the use of these data: data schema will be differently optimized depending on the objective of use: hospital management, reimbursement system, patient follow-up or epidemiological studies. This state of the art shows that none of this CDM have been optimized for epidemiological studies on administrative data.

[8] https://ohdsi.github.io/TheBookOfOhdsi/CommonDataModel.html

Concerning the SNDS, it does not yet offer any of the previous standardization, the database schema does not include any ontologies from BioPortal and OBO foundry. However, some type of data (notably drugs and medical acts) used the international classification ATC and ICD-10 to code them. Let us note that there are works aiming at representing the SNDS data in OMOP format. It is interesting to exploit these data in this format in the long run. As the initially defined OMOP concepts were aimed at EHR data, it is therefore a real challenge to adapt administrative data such as the SNDS to this model. We are looking forward at more generic data management systems that could allow us to represent and exploit the available ontologies, taking advantage of the fact that the SNDS data follow an international classifications standardization.

2.1.3 Semantic Web

The Semantic web is an extension of the World Wide Web following recommendations set by the World Wide Web Consortium (W3C). It is a system of databases where its initial goal was to facilitate the link between the different sources of internet data and to make them machine-readable. Semantic Web technologies are interested in the meaning of the data before the structure of the data.

Semantic Web offers a different approach to store and query data through a three levels architecture:

- **RDF (Resource Description Framework)**: Semantic web data are stored and represented with this data format.
- **RDFS/OWL (Resource Description Framework Schema/ Ontology Web Language):** they are schema language, also called the knowledge representation (KR). OWL defines a semantic which enables to define knowledge with the notion of concept and complex relations between these concepts.
- **SPARQL (SPARQL Protocol and RDF Query Language)**: the query language of the Semantic Web. SPARQL enables to query RDF data and to reason with knowledge defined by the OWL schema.

RDF/RDFS RDF models data under graph model composed of triples. A RDF triple is a set $\langle subject, predicate, object \rangle$ where subjects and objects are nodes and predicates links between these nodes. The same node can be both subject of a triple and object of

another. So on, data can be represented graphically. For example figure is composed of three triples :

- $\langle Jean, lives, Paris \rangle$
- $\langle Jean, hasBought, Doliprane \rangle$
- $\langle Doliprane, hasActiveMolecule, paracetamol \rangle$

In this example, Jean is the subject of two different triples and Doliprane is both the subject of a triple and the object of another triple. Thus, RDF enables to represent interconnected data. Each resource is easily identified by a unique sequence of characters called URI (Uniform Resource Identifier) used to identify a logical or physical resource used by web technologies. The data semantics is ensured thanks to the W3C recommendation.

OWL OWL facilitates machine interpretability by providing additional vocabulary. It is based on the Description Logic (DL) formalism [Baa+03] a family of formal knowledge representation (KR). Many DL categories exist, not described in this section but that can be found in [Baa+17]. A DL system is characterized by four aspects [Bie16] :

- A set of concepts and roles :
  - Concepts with one or domain argument. It corresponds to the term of class in OWL.
  - Roles with two domain arguments. It corresponds to *object properties* and *data properties* in OWL. *Object properties* connect pairs of individuals and data properties connect individuals with datatypes. Datatypes are entities that refer to sets of data values. [9].
- A set of concept and roles assertions. This set is called an $ABox$ and the assertions are called "individuals". The $ABox$ ensures a direct link between concepts and data, as an individual is a data linked to a concept.
- A set of universally quantified assertions, called the $TBox$.
- Inference mechanisms for reasoning on both the $TBox$ and the $ABox$.

[9] `https://www.w3.org/TR/owl2-syntax/#Datatypes`

Axioms supported by OWL 2QL	**the OWL syntax**
subclass axioms	SubClassOf
class expression equivalence	EquivalentClasses
class expression disjointness	DisjointClasses
inverse object properties	InverseObjectProperties
property inclusion	SubObjectPropertyOf, SubDataPropertyOf
property equivalence	EquivalentObjectProperties, EquivalentDataProperties
property domain	ObjectPropertyDomain, DataPropertyDomain
property range	ObjectPropertyRange, DataPropertyRange
disjoint properties	DisjointObjectProperties, DisjointDataProperties
symmetric properties	SymmetricObjectProperty
reflexive properties	ReflexiveObjectProperty
irreflexive properties	IrreflexiveObjectProperty
asymmetric properties	AsymmetricObjectProperty
property assertions	DifferentIndividuals, ClassAssertion, ObjectPropertyAssertion, DataPropertyAssertion

Table 2.1 – Axioms supported by OWL 2QL and their corresponding syntax in OWL

An Ontology is thus described through an $ABox$, a $TBox$ and an inference mechanisms and OWL enables to manage this ontology. Ontology 2QL are based on the DL-Lite family $\mathcal{EL}$ and is under the format : $ABox, TBox$ defined in Definition 2.

Definition 2 *DL-LITE [Cal+05] DL-Lite concepts are defined as follows:*

$$B ::= A \mid \exists R \mid \exists R^{-} C ::= B \mid \neg B \mid C_1 \sqcup C_2$$

A denotes an atomic concept and R denotes an atomic role; B denotes a basic concept that can be either an atomic concept, a concept of the form $\exists R$, or a concept of the form $\exists R^{-}$, which involves an inverse role. C denotes a concept. Note that we use negation of basic concepts, and disjunction is not allowed. So, a DL-Lite knowledge base (KB) is constituted by two components: a $TBox$ used to represent knowledge, and an $ABox$, used to represent information. DL-Lite TBox assertions are of the form:

$$B \sqsubseteq C \text{ inclusion assertions } (\text{FUNCT } R), (\text{FUNCT } R^{-}) \text{ functionality assertions}$$

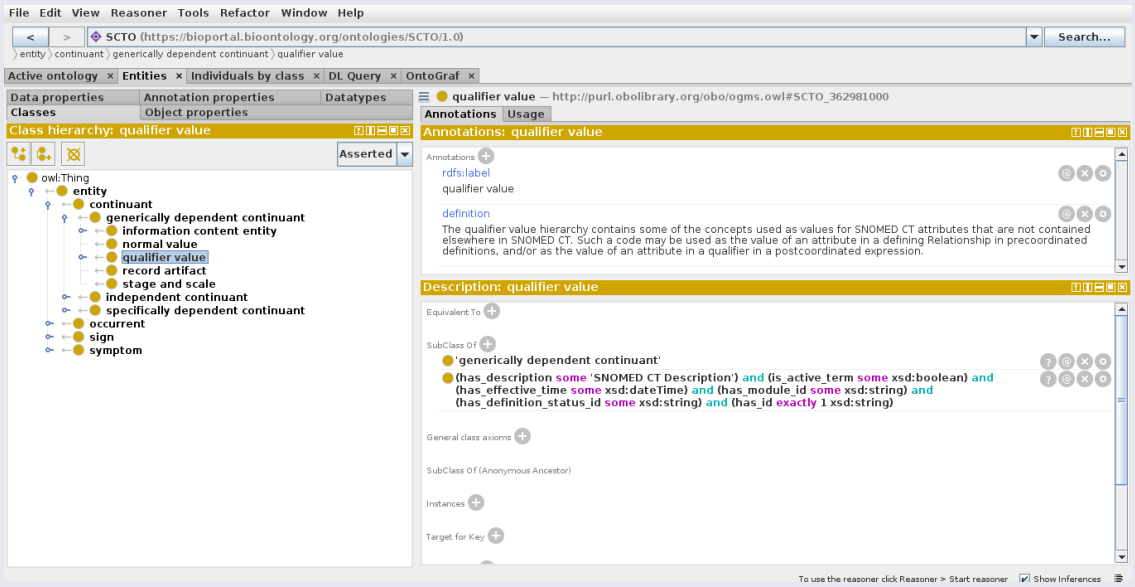
and $ABox$, assertions are of the form: $B(a), R(a, b)$membership assertions

As an example, we present the axioms of OWL 2 QL[10] in Table 2.1

[10] `https://www.w3.org/TR/owl2-profiles/#OWL_2_QL`

Example 2 – A view of the SNOMED CT 2QL ontology

As an example, the Systematized Nomenclature of Medicine Clinical Term Ontology (SCTO), proposed by Shaker El-Sappagh[ElS+18] attempts to create a OWL 2 ontology for SNOMED CT terminology. The SNOMED CT is an OWL 2QL ontology. Below, a view of SCTO with *Protégé*.



Thanks to OWL, data communities can classify information and bring knowledge to pre-existing data through ontologies. One can also reason about these ontologies and query the data linked to the ontologies thanks to the numerous tools available based on the OWL formalism.

The active community of the open-source software *Protégé*[11] [Hor+04; Mus15] has developed this tool to create ontologies OWL 2 QL, OWL 2 EL. In the example and in Table 2.1 on the preceding page, we present the *Protégé* editor which uses external reasoning engine (Hermit,...). We create classes, instances of these classes, link these instances with "data properties" and "object properties". We can add conditions such as those listed above. We can check the consistency of the schema and query ontology with DL-query. Of course, the defined ontology can be used with other applications , the ontology can be exported in RDF and thus queried with SPARQL. However, many platforms like *BioPortal* and *OBO foundry* provide OWL-ontology. We can note medical ontologies such as the international classification of drugs, the ATC, international classification of disease, the ICD-10 and the SNOMED-CT, which are all expressed in OWL 2QL.

[11] `https://protege.stanford.edu/`

SPARQL and queries SPARQL (SPARQL Protocol and RDF Query Language) is the standard language to query RDF data. SPARQL semantic is designed by the W3C. Unlike SQL, SPARQL enables to query data from multiple data stores. A SPARQL query is composed of triple patterns (subject, predicate and object) where each element of the triple can be a variable. The solution of the SPARQL query is the result of the mapping between the RDF triples and the triple pattern defined by the SPARQL query.

We can mention the OMQA (Ontology-Mediated Query Answering) domain which deals with governed data queries by ontologies defined via the Description Logic formalism. OMQA is not limited to SPARQL, but also deals with queries expressed with datalog, but we lose the genericity offered by OWL and RDF.

Concerning OMQA through temporal data, the formalism is explained in the section 2.2 on temporal models. In the context of temporal queries, some approaches propose to extend RDF/S- PARQL with temporal queries in a generic way For instance, Zhang et *et al.* [Zha+19] propose SPARQL[t] and EP-SPARQL [Ani+11a] which is a SPARQL extension of event processing.

ONTOP also proposes an ontology-based data access framework that has been extended for temporal data [Kal+19]. Semantic Web approaches also propose to combine query languages (*e.g.* SPARQL) extended to temporal data with Allen's relations [Wan+10].These approaches do not contain metric timed constraints and, as a consequence, are not relevant for the medical administrative databases query problems [Pac+18]. Moreover, they define the query language with a new syntax and a formal semantics and do not propose generic systems or API implementing this new semantics, which makes their use difficult.

In a more generic way, the usefulness of Semantic Web has been shown in a wide variety of areas: agriculture [Dru+19], biomedical [Smi+07], integrative biology [CYC13], healthcare [Zen+15] and it shows its efficiency to represent and query data in health informatics field [BZ12; RDL19; Sal+12; KK15]. Several efficient storage systems and APIs exist, here is a non exhaustive list: RDFox, Oracle Spatial and Graph 19c, Apache Jena, Virtuoso, GraphDB, QLever- where performances can be evaluated/compared according to the application. Therefore, the Semantic Web is a very active community, proposing several resources to query and reason on data with knowledge, including in health informatics.

As a conclusion, OWL is interesting for representing medical health data related to knowledge ontologies. The temporal aspect is treated in the data representation, but not much in the queries. The field has been extended by others communities proposing temporal models. They are interested in the search of information linked by temporal constraints, so in the following part we have selected existing temporal models relevant to our purpose.

2.2 Temporal Models

This chapter studies a selection of temporal models around the following aspects: predicates defined by the model, the syntax, the semantics and the temporal constraints. We distinguish the notion of temporal model from the notion of temporal constraints. For instance, the 13 Allen relations [All83] define temporal constraints while ETALIS defines a temporal model, where the predicates are linked to temporal constraints themselves. This distinction is important, it will allow us to work at the level of the granularity of the predicates which in our case are linked to health data.

This section introduces temporal models as defined in the literature. We are interested in models allowing data observation/verification, the description focuses on the temporal model definition and on data that can be queried/observed (depending of the formalisms) by these temporal models. We focused on four fields:

- Discrete Event Systems: automaton, Petri nets and their extension with time and temporal constraints
- Temporal logic: LTL, CTL, MTL, and so on...
- Complex Event Processing: chronicles and ETALIS
- Ontology-Mediated Query Answering (OMQA): Event Calculus, Description Logic, Sequence of events

For the sake of clarity, as the models come from different fields, we define the following terms that will allow us to describe models with a common vocabulary.

- *System:* group of entities with internal and external interaction.
- *Timed constraint:* constraints concerning an array of time

- *Temporal constraints:* constraints concerning a unique time instant
- *Event:* something that occurs, happens or changes the current state of affairs. [Ani+11b]
- *Sequence of events:* A *sequence* is an ordered set of timestamped events.

As follows, we will give an aspect of each field, according to their temporal model definition(figure 2.3 in green)and how these models are used to query/observe or verify timed data (figure 2.3 in blue). Temporal constraints, graphical representation and implementation of the temporal model have been described in the following for each of them.

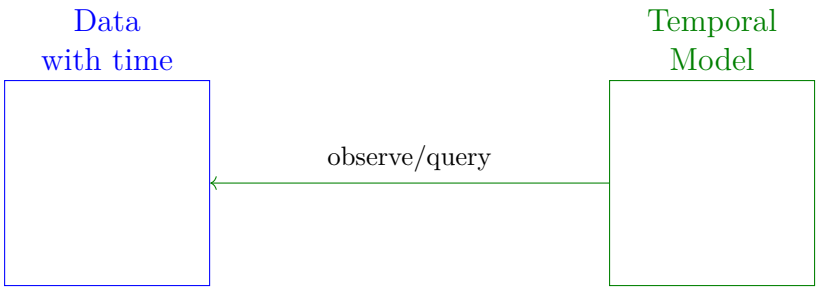


Figure 2.3 – data queried with temporal models

2.2.1 Model Checking - Temporal Modal Logic

MModel checking is the method by which a system is verified over a given model. In this section, we summarize the temporal logics as such model. They have been proved [Cla97] to be useful for such a task as describing the ordering of events without explicitly using time LTL, CTL [BK08]. However, real time systems extend these models with quantitative temporal properties: MTL [OW06; PD06], TPTL [BCM05], EventclockTL [HRS98].
The figure 2.4 on the next page shows in blue that data represented as sequences of events, or by automa- ton/Petri nets can be queried with model checking formula (in green).

Predicate Events are atomic events: they represent an occurrence at a point in time. They can be true or false.

Figure 2.4 – Model checking to query sequences of events or DES

Temporal constraints [PD06] Given a finite alphabet Σ of atomic events, the temporal logic formulas are built up from Σ by boolean connectives and temporal operators such as next ($\bigcirc$), eventually ($\Diamond$), always($\Box$) and until ($\sqcup$) in LTL. These are extended with intervals in MTL and with clocks in TPTL.

- LTL/CTL: ordering of events, no explicit time
- MTL: time constraint
- TPTL: temporal and time constraint with clocks

Negation A model checking formula can specify whether an event is true (occurs) or false (does not occur). It is noted $\neg$. So one the formula $\Box a$ means the event a is always true while $\Box\neg a$ means the event a is always false.

Graphical representation No graphical representation

Implementation UPPAAL proposes to use model checking over timed automaton, TINA [BRV04] proposes to use LTL to verify Petri nets, timed Petri Nets and temporal Petri nets, SPOT [DP04] proposes to use LTL to verify automata. There is no general algorithmic solution, but the constraint programming languages are suitable to solve this type of formula.

Discussion Model Checking is an interesting approach to reason on temporal model and formalize temporal constraints. This field has seen many theoretical contributions in the 80's and 90's, especially on decidability issues. However, the lack of efficient tools and graphical representation make it hardly accessible in our context.

2.2.2 Discrete Event Systems (DES)

DES are used to represent discrete process over time or to survey a behavior of a real time system. They are initially used for analysis and control systems [RW89; BS12] or diagnosis and detectability [SLT98; ZL13; BCD09]. Many formalisms are available , such as formal language [MAK12], Petri nets [Rei12] (extended with timed and temporal), automaton [CSK11] (extended with timed) and (Max,+) algebra [De 96].
IIn the case where DES are used to control/represent a system (blue part in Figure 2.5), temporal logics (Section 2.2.1 on page 33)can be used to verify the behaviour of the system, including checking the temporal constraints defined by the DES. Otherwise, DES are used to verify/observe sequences of events or even, a system (green part in Figure 2.6) represented with DES. This last point is related to diagnosis and detectability application.

Figure 2.5 – data represented with DES and observed with automaton

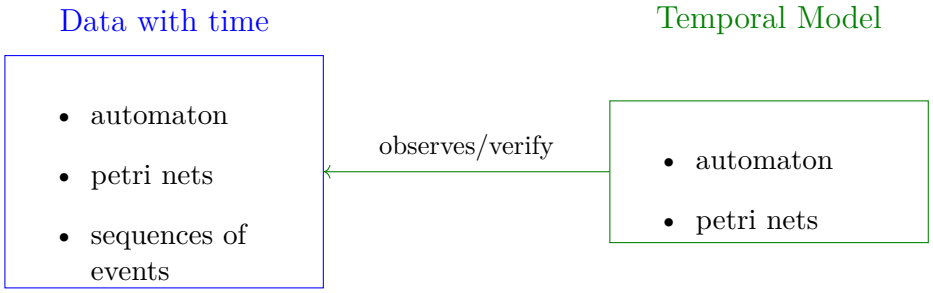


Figure 2.6 – sequences of events represented with sed and observed with automaton

Predicate There exists two predicates: event (an observation of fact) and action (an execution of a fact) where fact can be true or false. The notion of state is an abstraction of a discrete occurrence of event/action.

Negation None of the predicate can be linked to a negation.

Temporal constraints

- automaton: no temporal constraints, describe a succession of events.
- timed automata, with clocks: design event occurring before and/or after a certain clock value.
- timed/temporal petri nets: design a succession of events which have to occur at a specific time/in an interval of time.

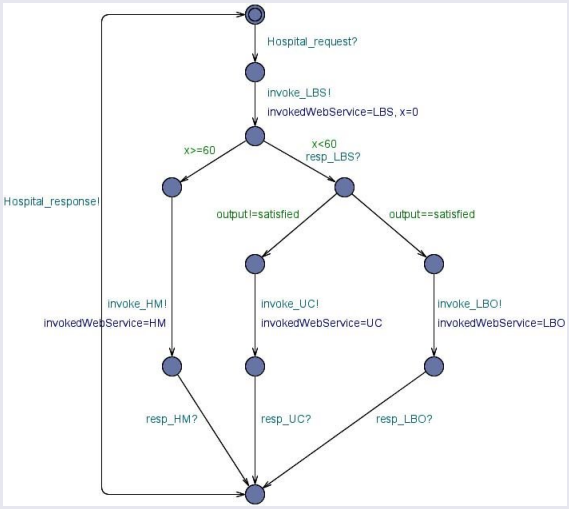
Graphical representation There are several formalisms and representations for timed automaton and timed+temporal Petri nets. Concerning the automaton, vertex represents event/action occurrence and edges represent the state change with the time constraint. The time constraint can also be carried by the vertex. Petri nets have notion of token to represent the system state and the state change is characterized by transition symbol, as automaton the temporal/time constraint can be carried by the transition or the state. As an example, the article [Maâ+13] presents a case study consisting in a hospital ordering blood components for transfusion purposes and they model process with a Timed Automata. The example shows a timed automata describing the process explained as followed by [Maâ+13].

Implementation In the context of Figure 2.5 on the previous page, tools are the same as those described Section 2.2.1 on page 33. For the context of control, as these models are based on finite state machine, they can easily be implemented with classic programming language [BC07]. In the context of health informatics, [JHS06] presents a survey showing that a significant amount of research has been conducted in the area of patient flow and asset allocation. A large number of discrete event simulation attempt to understand the relationship that may exist between various inputs into a health care delivery system.

Discussion DES is an interesting approach to represent temporal model and data with time. The graph representation is an asset, but the initial use leads to data observation. These models follow a "system" approach, they allow an easy description of a system and its changes of states in time. They are thus very useful to observe data. In our context, this type of model may not be easily used, since the data captured by the SNDS has not been thought as an over time evolving system.

Example 3 – Hospital ordering blood components for transfusion purposes [Maâ+13]

We suppose that the required business process composes services of: local blood search (LBS), local blood ordering (LBO), hospital maintenance (HM) and unsatisfactory customer (UC). We also assume that the two first partner services (LBS) and (LBO) are connected to a local blood bank which is situated in the involved hospital. In fact, the blood bank is mainly responsible for rapid response to urgent requests for blood components and for selection of suitable blood component for each clinical condition. Once a hospital unit (such as emergency, surgery, etc.) sends a request to the HBO-TP process, the LBS service is invoked to search for required blood component from the local blood bank. This search is conditioned by awaiting time. Indeed, the process should receive a response from LBS within maximum 60 seconds (for example). Otherwise, the process sends a connection problem report to the (HM) service. In case of receiving a blood search response before reaching 60 seconds, obtained search results are analyzed. If the conditions related to the needed blood component are satisfied, then the (LBO) service is invoked to order blood from the local blood bank. Otherwise, an unsatisfactory customer report is sent to (UC) service for information about unavailable required blood component. Thus, responsibles are put in charge to answer quickly the hospital unit need for blood. Finally, a detailed reply informing about final results is sent to the concerned hospital unit.



The HBO-TP Process modeled in Timed Automata[Maâ+13]

2.2.3 Complex Event Processing

Complex Event Processing (CEP) [Gia+17] aims at processing a stream of event logs with patterns. CEP processes these logs to detect or to locate complex events (or patterns) defined by the user. These models emphasize the effectiveness of processing and the expressivity of pattern. Temporal constraint networks [COP00] or Chronicles [DL07] are temporal models that are interesting for their graphical representation. Some more expressive formalisms, *e.g.* ETALIS [Ani+11b] or logic-based event recognition [Gia+17] propose very expressive representations of complex events, including reasoning techniques and contextual knowledge. It leads to enrich the recorded events.

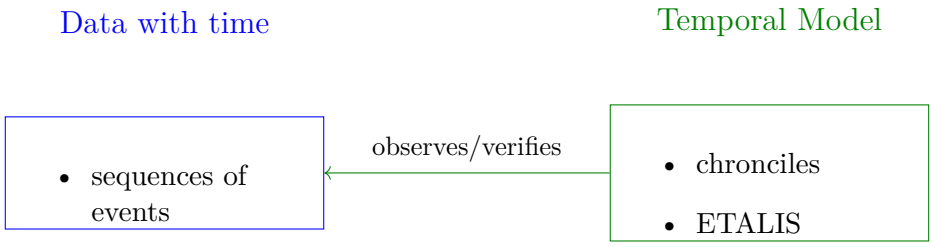


Figure 2.7 – data represented with sed and observed with automaton

Predicat ETALIS distinguishes atomic and complex events, endowed with a time interval when the event started and when it ended. Chronicles uses atomic events.

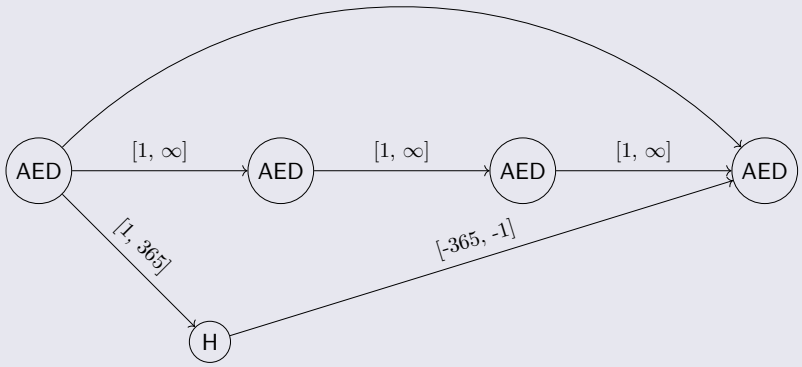
Temporal constraints The complex event descriptions of ETALIS are based on Allen's logic while chronicles [DL07] propose atomic events with more permissive temporal constraints than temporal constraint networks [COP00]. A *Chronicle* is a set of atomic events and a set of temporal constraints between pairs of atomic events [DL07].

Implementation This domain defines formalisms for a better efficiency in order to process streams and expressive to specify patterns. The event stream processing of ETALIS is supported by StreamSQL [SÇZ05],Oracle EPL and in a more general way, can be implemented with constraint programming language such as ASP, datalog or prolog. Effective algorithms to enumerate chronicle occurrences [DL07].Chronicles have been used to discover patterns in biomedical data [Dau+17; SLL18]. To the best of our knowledge, ETALIS has not been applied to medical purposes.

Graphical representation Concerning Chronicles, it can been represented as a constraint network with vertices and temporal constraints with arrows between the vertices.

Example 4 – Chronicle

The article [Pol+15] identifies patients suffering of epilepsy. Cases were identified, using the PMSI database (hospital data from the SNDS), as individuals with a seizure-related hospitalization. As an example, we propose to represent the care pathway of such patients care trajectory with a chronicle:



where *AED* are antiepileptic drugs deliveries and *H* an hospitalization. We can read the chronicle as follows : "An ADE is followed by another AED which is itself followed by another, which is itself followed by another, which is itself followed by another (four successive occurrences of AED). There is at most a year between the first and the last AED. An hospitalization occurs at maximum a year after the first AED and a year before the fourth."

Discussion DES is an interesting approach to represent temporal model as well as data with time. The graph representation already used in medical context is an asset. The absence of negation is a limitation.

2.2.4 Ontology-Mediated Query Answering (OMQA) over Temporal Data

In this section, we are interested in temporal model linked with ontologies. Artale et al. developed a survey which explore the OMQA models with temporal data [Art+17] from which this section is inspired. Ontology-based data access (OBDA) is a successful applica-

tion of the Description Logics (DL) which facilitates access to heterogeneous, distributed and incomplete data. The objective of OBDA is to query data from the knowledge of the ontologies apart from the data schema.
OWL ontology can represent knowledge about static domains, hardly suitable with temporal data. Combining DL with temporal formalism has been investigated by Schmiedel [Sch+90] and Schild [Sch93], where they referred to concept satisfiability and not query answering.
To carry on query answering, Artale et al. [Art+17] distinguish 2 approaches:

- discrete point-based approach: time is discrete, fact comes with a time-point
- interval-based approach: fact are stamped with the interval in which they are true

Figure 2.8 – discrete point-based and interval-based approach OMQA

Predicate In the case of point based temporal ontology-mediated query (OMQ) and domain- or time-centric OMQ, the DL predicates with one domain argument are called concepts and predicates with two domains arguments are called roles. A finite set of timestamped facts is called a temporal Abox where an individual's name defined by a concept can be linked with a timestamp through a role.

In the case of interval-based temporal ontology-mediated query answering, the DL predicates with Abox are extended with Aboxes with intervals [HS91] and Tbox with $\mathcal{HS} - Lite_{horn}^{H}$.

Temporal constraints To query point based temporal ontology-mediated [Art+17] proposes to use 2-FOL query constructed from atoms composed of concepts and roles, variables and temporal variables which can be compared with symbols $<$ and $=$.
To query time-centric ontologies [Art+17] propose to 2-FOL query defined non-temporal properties. Baader *et al.* [BGL12] proposes domain-centric languages, they introduce

$\mathcal{ALC} - \mathcal{LTL}$ and $\mathcal{EL} - \mathcal{LTL}$ [BT15]
To query domain-centric ontologies (ontology based monitoring of dynamic systems) [Baa14] proposes an union of conjunctive queries (UCQ), containing same atoms than before, plus *and, or* and *exists*.
Finally, to query interval-based temporal ontology-mediated query qnswering, 2-FOL formula are expressed with an interval-based view of time with Halpern-Shoham [HS91] and allen's constraints [All83].

Negation Concerning negation, the operator $\neg a$ designs the answer of an atom a as *false* and $\neg\varphi$ which designs the answer of a formula φ as *false*, where the atoms are defined by the $ABox$ of the OMQA system.

Graphical representation There is no graphical representation for the query. However, as seen before, ontology can be represented with OWL and then some tools such as *Protégé* allow to browse these ontologies.

Implementation Semantic Web extensions RDF/S and OWL with validity time [GHV06; Mot12; PUS08] proposes to represent and query data/ontology with time. Some tools like SPARQL, Prolog, datalog, enable to construct Ontology-Mediated query. However, even without a temporal dimension, first-order logic is too expressive for effective ontology-mediated query answering.

Discussion OMQA is an interesting approach to represent temporal model and reason on data with time. The main asset is their implementation with Se mantic Web which proves to be efficient in a medical context as seen before in Section 2.1.3 on page 27. However, the absence of graphical representation to construct query is a limitation.

2.3 Synthesis - general discussion

This state of the art has been organised around four main axes issued from different domains as summarized on Figure 2.9 on the following page. We divided the state of the art in two parts:

- Database representation (on the right, Figure 2.9 on the next page): temporal databases, KRR (Knowledge Representation and Reasonning) and Common Data

Model

- Temporal models (on the left, Figure 2.9 on the following page) : DES (Discrete Event System), Complex Event Processing, Model Checking and the OMQA over temporal data which make the link between database representation and temporal models.

The database domain is focused on data representation and on query systems. We have seen that relational data domains such as T-SQL propose several solutions to query data with time but do not deal with ontological aspects, while the Semantic Web proposes solutions to exploit ontologies, but only a few tools are available concerning temporal queries. In terms of application in the medical field, we observe that the relational model proposed by SQL is very much used in the Common Data Models but it is not very efficient and not accessible to evaluate search tools, while the Semantic Web is the result of a very active community proposing open-source tools. Some epidemiological studies have already been conducted in the medical field, encouraging its usefulness for medical concerns. About the temporal aspects to describe a pattern, we have seen that complex event processing proposes very efficient tools for reasoning about the occurrence of temporal patterns in sequences of events. The notions of sequences and events are central in these approaches, and it is rather difficult to link the predicates proposed by the models with the data representation formalisms. The domain of formal logics proposes highly elaborate semantics and syntaxes models. The predicate descriptions make it suitable to link temporal constraints and data representation. In particularly Description Logic makes possible to create well-formed ontologies and reason about them. The most important limit is the number of available and efficient tools to exploit these formalisms in real applications. The OMQA field has partially addressed this problem, focusing on queries rather than on theoretical results. On the other hand, the results concerning the efficiency evaluation of the tools in the medical area remains an open science that we will partially cover during this thesis.

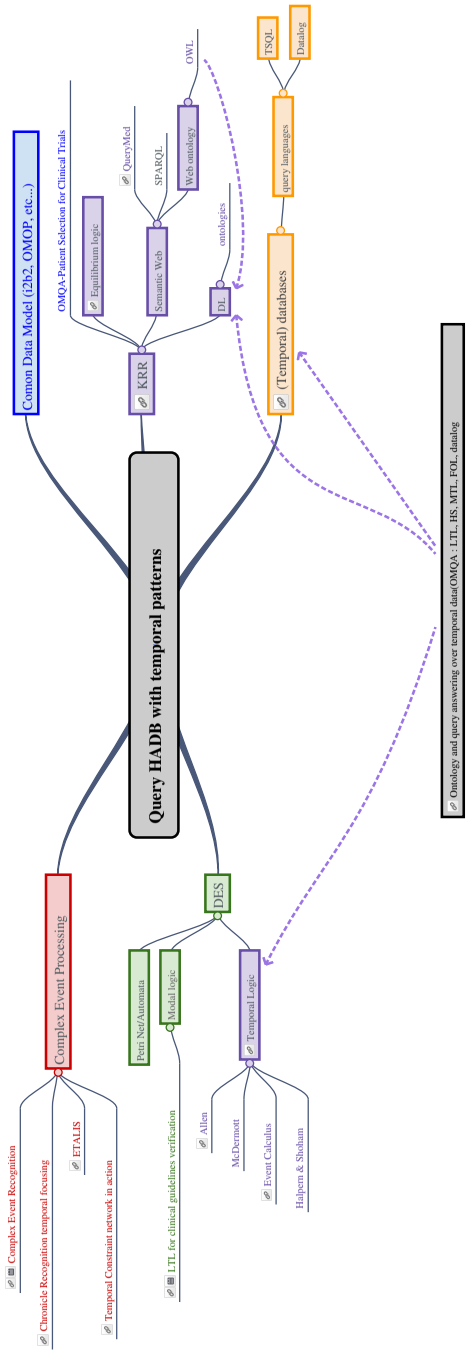


Figure 2.9 – A mindmap to resume the state of the art to query HADB with temporal patterns

Chapter 3

Objective - an overview of the approach

The objective of the thesis is to design a method for combining ontologies and temporal reasoning in order to empower pharmaco-epidemiologists to query a large AHDB and select patients facing a given phenotype. In the problem statement, it was shown that pharmaco- epidemiology studies require selecting sets of patients with a common set of criteria. The selection of such patients in general databases (AHDB in our context) requires a medical phenotype development. We have seen that these criteria depend on the information available in the database and that they can have or be linked by specific temporal constraints.

Thus, we propose to address this problem by a three-level approach:

1. A theoretical level to formalize the database, a phenotype and the task to find individuals from the database verifying a given phenotype. It refers to the green box on Figure 1.2 on page 17.

2. A practical level to create a tool based on the theoretical definition to perform the task. It refers to the brown box on Figure 1.2 on page 17.

3. An applicative level to evaluate the tool on several use cases.

The theoretical level The mathematical model formally defines the task to find patients facing a given phenotype. It includes :

- to formally describe the database (left side in the green box on Figure 1.2 on page 17).

- to define a temporal model used to describe phenotypes (right side in the green box on Figure 1.2 on page 17).

- then, to formally describe whether a patient in the database matches the temporal model.

The theoretical data model must capture information about care sequences of the patients where they are expressed through standardized data. As seen in Chapter 2 on page 19 state-of-the-art, it is advisable to use standardized data to facilitate the manipulation of the data and tools designed to integrate medical data. Using standardized data is improved by the use of medical knowledge such as ontologies (for example, ATC taxonomy or SNOMED CT ontology). So, our approach has to deal with time manipulation in addition to ontology management.

In the following„ we are interested in the temporal model of the Chronicles (refer to the state-of-the-art Section 2.2.3 on page 39 on the *Complex Event Processing*) to describe phenotypes. Then, we propose an OWL-model to describe SNDS data and link them with international ontologies (refer to the state-of-the-art Section 2.1.3 on page 27 on the *Semantic Web*). With this more complete data representation, chronicles can be extended. To do so, we will use 2-FOL formula issued from the OMQA domain (refer to the state-of-the-art Section 2.2.4 on page 41 on the *Ontology-Mediated Query Answering*)

The practical level A tool called HyCOR is developed to find every patient in the database facing a given phenotype. This approach is focused on developing a tool based on the theoretical approach described before while ensuring good efficiency. We evaluate the time needed to complete the task. This tool captures two aspects: the data storage with ontologies and and their querying. We automatically transform the temporal model describing a phenotype into a query. It includes the possibility to represent the data according to the formal model defined by the previous step and query them according to the temporal model defined by the previous step.

In the following, we propose an RDF representation of the data (refer to the state-of-the-art Section 2.1.3 on page 27 on the *Semantic Web*) proposing a set of efficient tools to reason about these data. Then we associate the Semantic Web tools with efficient algorithms from the complex event processing domain (refer to the state-of-the-art Section 2.2.3 on page 39 on the *Complex Event Processing*).

The applicative approach Several use cases have been developed with pharmaco-epidemiologists to evaluate the expressivity and the usefulness of the tool. Use Cases have an application on the French health insurance database (SNDS).

This thesis aims at exploiting formal mathematical models to facilitate the exploitation of large databases for pharmaco-epidemiology purposes. Our approach allows to mathematically formalize the problems related to the patients selection in a large database, linking them to temporal models already developed in the computer science literature (here, we are interested in Chronicles and OMQA). The other advantage is the more efficient use of these models thanks to practical solutions from the Semantic Web and Chronicles algorithms. The solutions proposed are tested on real use cases where data are issued from the SNDS.

In this manuscript, we start in the Chapter 4 on page 49 by presenting the construction of phenotype with chronicles. We detail the theoretical aspects: the notion of events and time in chronicles, then we detail the practical aspects: the automatic construction of a query from a Chronicle that we will compare to an efficient chronicle recognition algorithm linked to efficient queries. The tool allowing this second solution is called HyCOR. We will evaluate the approaches using randomly generated datasets representing care sequences in RDF format.

In the next Chapter 5 on page 69, for the theoretical aspect, we will propose a data schema allowing a formal data representation. In practice, we will use the Protégé [1] tool which enable us to build an ontology of the SNDS linked to international ontologies. We will propose a transformation of a sample of SNDS following this model.

This last data model enable us to extend Chronicles in Chapter 6 on page 83 in order to detail the events of a Chronicle. We will extend the notion of events with types and ontologies. This is the theoretical aspect developed in the Chapter 6 on page 83. In the practical approach, we extend the tool HyCOR to query these data with Chronicles containing events where we can specify the type of the event and concepts linked to this event.

[1] `https://protege.stanford.edu/`

We will end this manuscript with the case study of VTE on real data. We will use the sample created in the Chapter 5 on page 69 to extract patients verifying the VTE phenotypes. We will detail the construction of these phenotypes with the complete chronicles defined in the Chapter 6 on page 83. Then, we will see that it only takes a few seconds for HyCOR to extract patients verifying such a phenotype.

Chapter 4

Express phenotypes with the temporal model of Chronicles

In the previous Chapter 3 on page 45, we have seen that the problem is divided in two sides: a side concerning the formal data representation and the other side concerning the querying of data from a phenotype model. In this chapter we focus on the phenotype side, right side of the Figure 4.1. We propose a formalization of phenotype with chronicles and propose an implementation that extracts the sequences verifying a chronicle.

There are several advantages to use a formal model.

- The first one is to formalize the notion of phenotype: to formalize the temporal constraints and to formalize the entities used in phenotypes (entities from the data model)
- The second is the possibility of proposing efficient algorithms following the model

Thus, we propose to use Chronicles as a model of phenotype representation and to use them as a pattern to query data. We first define the formal notion of sequence of events where events are linked to taxonomies and we introduce the representation of these sequences with RDF. Then, we present the three following contributions:

1. we extend the definition of Chronicles with events belonging to taxonomy
2. we explain how to represent a phenotype with a Chronicle
3. we encode Chronicles as SPARQL queries to enumerate all their occurrences in a sequence of events represented in RDF
4. we propose HyCOR, an hybrid method combining the expressiveness of Semantic Web and the efficiency of a pattern occurrence enumeration algorithm to enumerate efficiently the extended-Chronicle occurrences in a set of sequences.

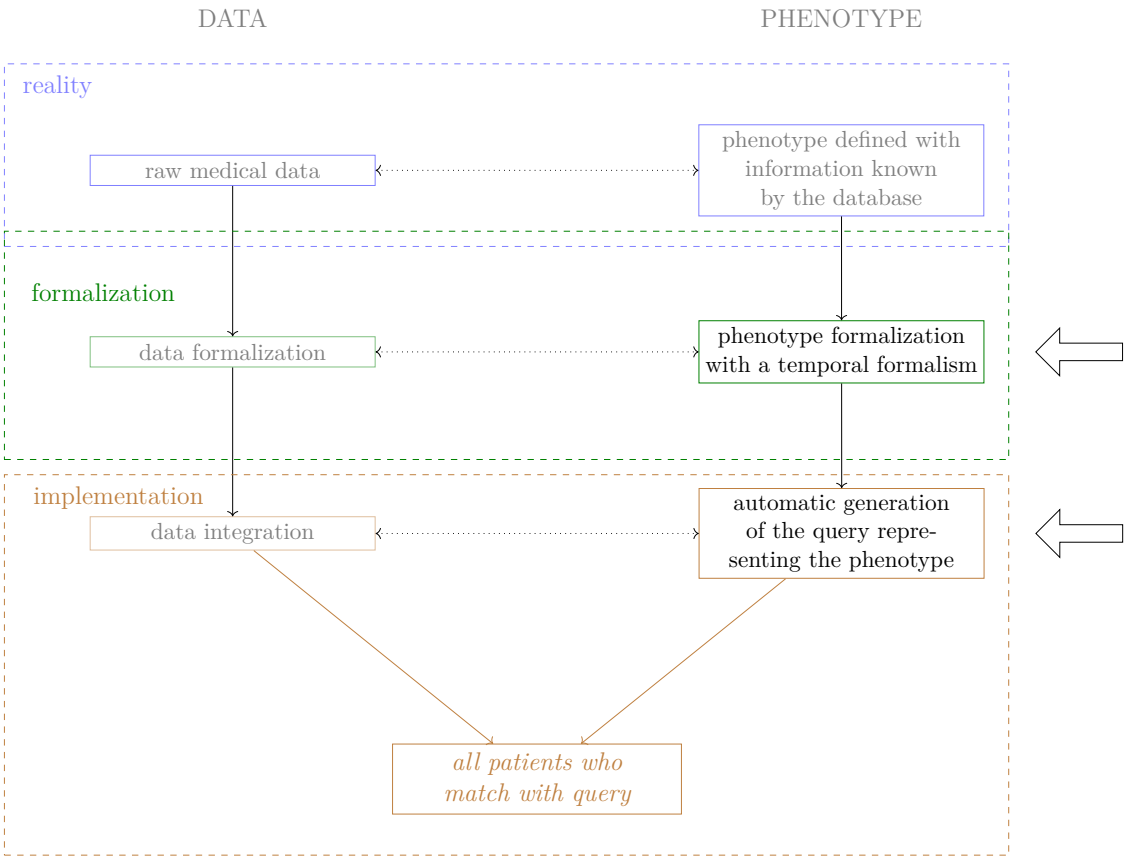


Figure 4.1 – Remind of the problematic, this chapter is interested to phenotype formalization and implementation

4.1 Sequences and Taxonomies

In this work, we adopt a longitudinal view of the SNDS. We assume that each patient has a sequence of timestamped cares, so-called *events*. As a beginning, we formally define a sequence as a set of events where an *event* is a pair (e, t) where e is a code, and $t \in \mathbb{N}$ is a timestamp (in days). We use a simplified view of the events where an event is composed of a code related to a taxonomy and a timestamp – which is the start date. It corresponds to data format proposed by Yann Rivault [Riv19].

We introduce the notation $e \rightsquigarrow c$ to denote that e is a direct or indirect subclass of c. The relation *subClassOf*($\rightsquigarrow$) is a transitive relation defining by RDFS. It is used to state that all the instances of one class are instances of another[1].

Let $(\mathbb{E}, \leq_{\mathbb{E}}, \rightsquigarrow)$ denotes a set of ordered event codes. An event code $e \in \mathbb{E}$ *is a subclass of* $c \in \mathbb{E}$, denoted $e \rightsquigarrow c$, iff e is in the equivalence class of c. By extension, with $c, c' \in \mathbb{E}$, $e \rightsquigarrow c'$ and $c' \rightsquigarrow c \Rightarrow e \rightsquigarrow c$ for all $e \in \mathbb{E}$.

Figure 4.2 on the following page gives an example of a taxonomy: the drug classification taxonomy called ATC. Each ATC code is a class in the ATC taxonomy. For instance, the ATC code B01 is a subclass of B (B01 $\rightsquigarrow$ B) and B01A is a subclass of B01 (B01A $\rightsquigarrow$ B01) then, B01A is a subclass of B (B01A $\rightsquigarrow$ B). These are taxonomies, but in a more general sense, we are interested in any ontology composed of classes linked together by subclass relations.

Let us now introduce a formal definition of a temporal sequence of events where events can belong to taxonomies.

Definition 3 (Sequence) *A* sequence $\boldsymbol{s}$ *is a finite list of events* $\langle (e_1, t_1), (e_2, t_2), \cdots, (e_n, t_n) \rangle$ *where* e_i *is an event code equipped by a taxonomy relation. Events in a sequence are ordered by their timestamps and then their code:* $i \leq j \Leftrightarrow t_i < t_j \vee (t_i = t_j \wedge e_i \leq_{\mathbb{E}} e_j)$, $\forall i, j \in \{1, \ldots, n\}$.

A *dataset of sequences* is a finite unordered set of sequences, $\mathcal{S} = \{\boldsymbol{s_1}, \ldots, \boldsymbol{s_m}\}$. Table. 4.1 illustrates six sequences where each event code is an ATC code at the lowest level of the ATC taxonomy.

[1] `https://www.w3.org/TR/rdf-schema/#ch_subclassof`

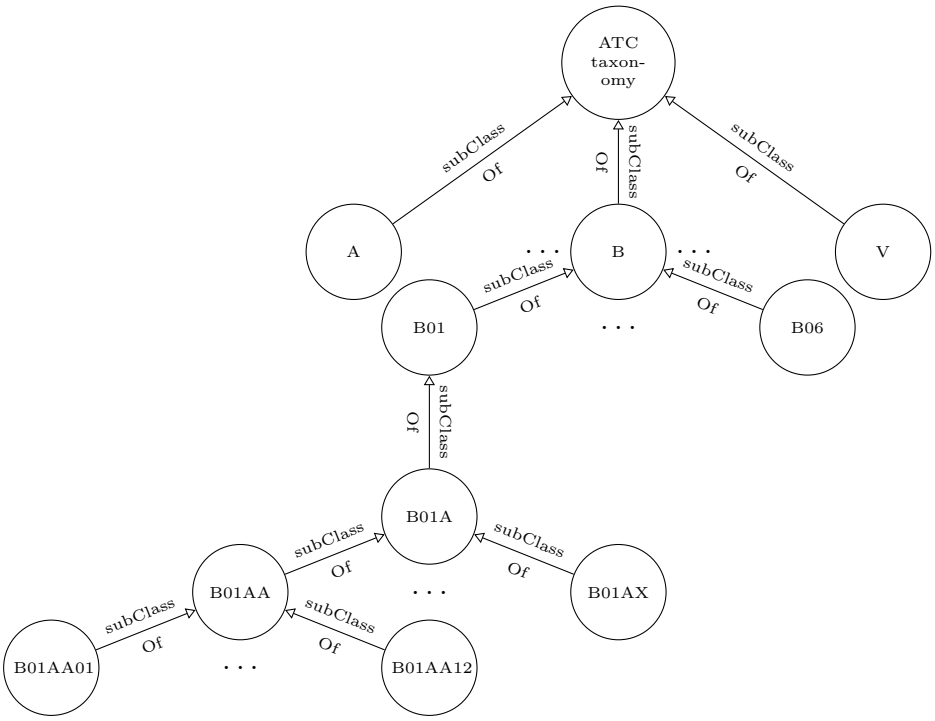


Figure 4.2 – Extract of the representation of the ATC taxonomy in the form of an RDF graph as proposed by the BioPortal platform. Each node is a class/group of drugs linked to each other by relationships of type "subClass Of". This taxonomy is thus hierarchized in 5 levels plus the root.

Table 4.1 – Example of a dataset of six sequences. Each sequence is composed of drug deliveries events (couples of code and timestamp). Codes are ATC codes.

id	Sequence
s_1	(A01AA01, 1), (B01AA01, 3), (A01AB14, 4), (C01AA01, 5), (C02AC01, 6), (D01AA01, 7)
s_2	(B01AA01, 2), (D01AA01, 4), (A01AA01, 5), (C01AA01, 7)
s_3	(A03AA01, 1), (B01AA01, 4), (C01AA01, 5), (B01AA01, 6), (C01AA01, 8), (D01AA01, 9)
s_4	(B01AA01, 4), (A01AB14, 6), (N01AA01, 8), (C01AA01, 9)
s_5	(B01AA01, 1), (A01AA01, 3), (C01AA01, 4)
s_6	(C01AA01, 4), (B01AA01, 5), (A01AA01, 6), (C01AA01, 7), (D01AA01, 10)

4.2 Chronicle Occurrences Enumeration

In this section, we propose an extension of the definition of Chronicles [Dau+17; DL07; SLL18] with items belonging to taxonomy classes. Then, we formally define an extended-Chronicle occurrence in a sequence and the enumeration of all extended-Chronicle occurrences in a sequence.

A *Chronicle* is a set of items and a set of temporal constraints between pairs of events [DL07]. In our applied context, a Chronicle enables us to represent a phenotype. The enumeration of Chronicle occurrences aims at localizing where this medical pattern occurs in a patient care trajectory. We propose a Chronicle extension, named *taxo-Chronicle*, where items may have codes belonging to classes linked by subClass relations.

Definition 4 (taxo-Chronicle) *A **taxo-Chronicle** $\mathscr{C}$ is a pair $(\mathcal{E}, \mathcal{T})$ where*

- *$\mathcal{E}$ is an ordered **set of items** $\{(c_1, 1), \cdots, (c_m, m)\}$, where for all $i \in \{1, \ldots, m\}$, $c_i \in \mathbb{E}$ is an item belonging to a taxonomy. i designates the index of the i-th item index.*
- *$\mathcal{T}$ is a set of **temporal constraints**,* i.e. *expressions of the form $(c_j, j)[t^-, t^+](c_k, k)$ such that*
  - *$(c_j, j), (c_k, k) \in \mathcal{E}$,*
  - *$t^-, t^+ \in \mathbb{R} \cup \{+\infty, -\infty\}$ and*
  - *For all $(c_j, j), (c_k, k) \in \mathcal{E}$, $j < k$,*

$$c_j \rightsquigarrow c_k \implies \exists (c_j, j)[t^-, t^+](c_k, k) \in \mathcal{T} \text{ s.t. } [t^-, t^+] \subseteq]0, +\infty[\tag{4.1}$$

The taxo-Chronicle size is m (number of items).

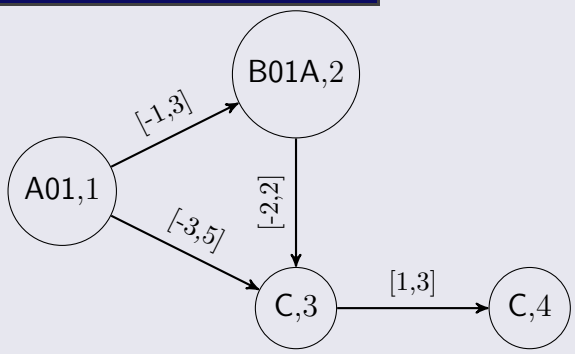
Intuitively, a temporal constraint $(c_j, j)[t^-, t^+](c_k, k)$ enforces an event (c_k, k) to occur with a temporal delay in between t^- and t^+ from an occurrence of (c_j, j). Note that several items can have the same code, hence, eq. 4.1 on the previous page enforces event occurrences to be ordered by the corresponding item index.

The example illustrates graphically the following 4-sized taxo-Chronicle $\mathscr{C} = (\mathcal{E}, \mathcal{T})$:

- $\mathcal{E} = \{(\mathsf{A01}, 1), (\mathsf{B01A}, 2), (\mathsf{C}, 3), (\mathsf{C}, 4)\}$
- $\mathcal{T} = \{(\mathsf{A01}, 1)[-1, 3](\mathsf{B01A}, 2) \quad , \quad (\mathsf{A01}, 1)[-3, 5](\mathsf{C}, 3) \ , \quad (\mathsf{B01A}, 2)[-2, 2](\mathsf{C}, 3) \quad , \quad (\mathsf{C}, 3)[1, 3](\mathsf{C}, 4)\}$

Note that temporal constraints may have negative values. The temporal constraint $(\mathsf{A01}, 1)[-3, 5](\mathsf{C}, 3)$ states that an event with a code in the equivalence class of A01 must occur from 3 days before occurrence of a C to 5 days after this occurrence. Example 5 proposes of a taxo-Chronicle.

Example 5 – A chronicle with 4 items et 4 temporal constraints



taxo-Chronicle example with 4 items (vertices) and 4 temporal constraints (edges with temporal intervals). Vertex codes give the event code (ATC codes).

The chronicle represents an event A01 followed by an event B01A within a delay of $[-1, 3]$ units of time (*ut*). The later is followed by an event C within a delay of $[-2, 2]$ *ut*. In addition the delay between this event C and the event coded A01 must be in $[-3, 5]$ *ut*. Finally, C event is followed by an another event C within a delay of $[1, 3]$ *ut*.

In the following, we introduce the definition of a taxo-Chronicle occurrence in a sequence. Then, one can be interested in two different tasks: enumerating all occurrences of a taxo-Chronicle in a sequence (taxo-Chronicle enumeration), or deciding whether a taxo-Chronicle occurs at least once in the sequence (taxo-Chronicle recognition). In the following, we focus on the taxo-Chronicle enumeration task.

Definition 5 (taxo-Chronicle occurrence) *Let* $\boldsymbol{s} = \langle (e_1, t_1), (e_2, t_2), \ldots, (e_n, t_n) \rangle$ *be a sequence of length* n *and* $\mathscr{C} = (\mathcal{E} = \{(c_1, 1), \ldots, (c_m, m)\}, \mathcal{T})$ *be a taxo-Chronicle of size* m *over a set of event codes* $(\mathbb{E}, \leq_{\mathbb{E}}, \rightsquigarrow)$.

An occurrence *of* $\mathscr{C}$ *in* $\boldsymbol{s}$ *is a subsequence of* $\boldsymbol{s}$ *of length* m, $\tilde{\boldsymbol{s}}$ *such that*

$$\tilde{\boldsymbol{s}} = \langle (e_{\varepsilon_1}, t_{\varepsilon_1}), \ldots, (c_{\varepsilon_m}, t_{\varepsilon_m}) \rangle$$

where $(\varepsilon_i)_{i=1..m}$ *are indices of an event in* $\boldsymbol{s}$ *and s.t.*

1. $e_{\varepsilon_i} \rightsquigarrow c_i$
2. $t_{\varepsilon_j} - t_{\varepsilon_i} \in [t^-, t^+]$ *whenever* $(c_i, i)[t^-, t^+](c_j, j) \in \mathcal{T}$.

$(\varepsilon_i)_{i=1..m}$ describes $\tilde{\boldsymbol{s}}$ a subsequence of $\boldsymbol{s}$. The first condition ensures that the i-th event code of $\tilde{\boldsymbol{s}}$ is a subclass of c_i. The second condition ensures that temporal constraints are satisfied. Note that Eq. 4.1 enforces to have a strict order between events c_k and c_j whenever $c_k \rightsquigarrow c_j$. Thus, all ε_i, $i \in [1, n]$ are distinct. Example 6 illustrates this definition.

Example 6 – A chronicle occurring in sequences

The taxo-Chronicle of the previous example occurs in sequences $\boldsymbol{s_1}$ and $\boldsymbol{s_6}$ of the dataset Table 4.1. $\{(\mathsf{A01AA01}, 1), (\mathsf{B01AA01}, 3), (\mathsf{C01AA01}, 5), \mathsf{C02AC01}, 6)\}$ is an occurrence of $\mathscr{C}$ in $\boldsymbol{s_1}$:

- All the events specified by the taxo-Chronicle items occur: A01AA01 is an undirect subclass of A01 and B01AA01 is an undirect subclass of B01A, and C01AA01 and C01AA01 are undirect subclasses of C (refer to Figure 4.2 which shows a branch of the ATC taxonomy).
- The temporal constraints are respected.

This occurrence is the subsequence of $\boldsymbol{s_1}$ with indices $\langle 1, 2, 4, 5 \rangle$. The taxo-Chronicle does not occur in $\boldsymbol{s_2}$ either in $\boldsymbol{s_4}$ because of unsatisfied temporal constraints. It does not occur in $\boldsymbol{s_5}$ as there is only one event which is a subclass of C and the taxo-Chronicle requires two events. It does not occur in $\boldsymbol{s_3}$ as there are no events which are a subclass of A01.

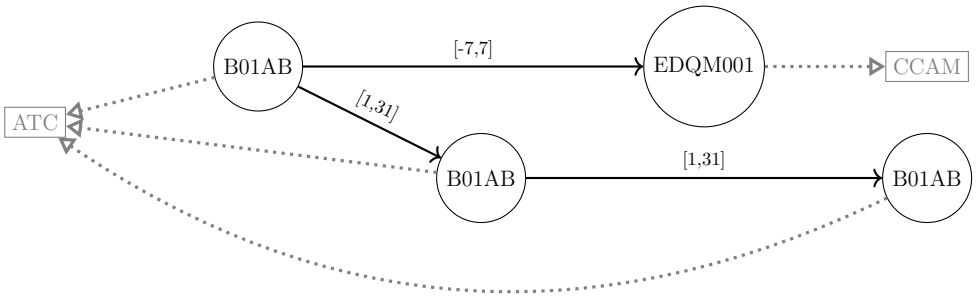


Figure 4.3 – Chronicles for representing VTE phenotype.

4.3 Taxo-Chronicles to represent phenotypes

We use a taxo-Chronicle to represent a phenotype. Following the use case of VTE, we propose the pattern in figure 4.3. This is a 4-sized taxo-Chronicle $\mathcal{C}_{VTE} = (\mathcal{E}, \mathcal{T})$:

- $\mathcal{E} = \{(\mathsf{B01AB}, 1), (\mathsf{EDQM001}, 2), (\mathsf{B01AB}, 3), (\mathsf{B01AB}, 4)\}$
- $\mathcal{T} = \{(\mathsf{B01AB}, 1)[-2, 2](\mathsf{EDQM001}, 2)\ ,\ \ (\mathsf{B01AB}, 1)[-1, 31](\mathsf{B01AB}, 3)\ ,$ $(\mathsf{B01AB}, 3)[1, 31](\mathsf{B01AB}, 4)\}$

The $\mathcal{C}_{VTE}$ taxo-Chronicle specifies:

- An event with a code in the equivalence class of EDQM001 must occur from 2 days before the occurrence of a B01AB to 2 days after this occurrence.
- This same event B01AB is followed by another event with a code in the equivalence class of B01AB from 1 to 31 days after.
- This later occurrence is followed by another event with a code in the equivalence class of B01AB from 1 to 31 days after.

We add to the graphical representation a piece of information: the taxonomy to which the code belongs. So on, we know that B01AB is an ATC code that designates an anticoagulant and EDQM001 is a CCAM code that designates a Doppler code.

In other words, the taxo-Chronicle specifies a pattern containing at least three deliveries of anticoagulants separated by no more than 1 month. The first delivery is 7 days after or before Doppler imaging. Note that a real use case will be presented in Chapter 7.

4.4 Semantic Web for Chronicle Recognition

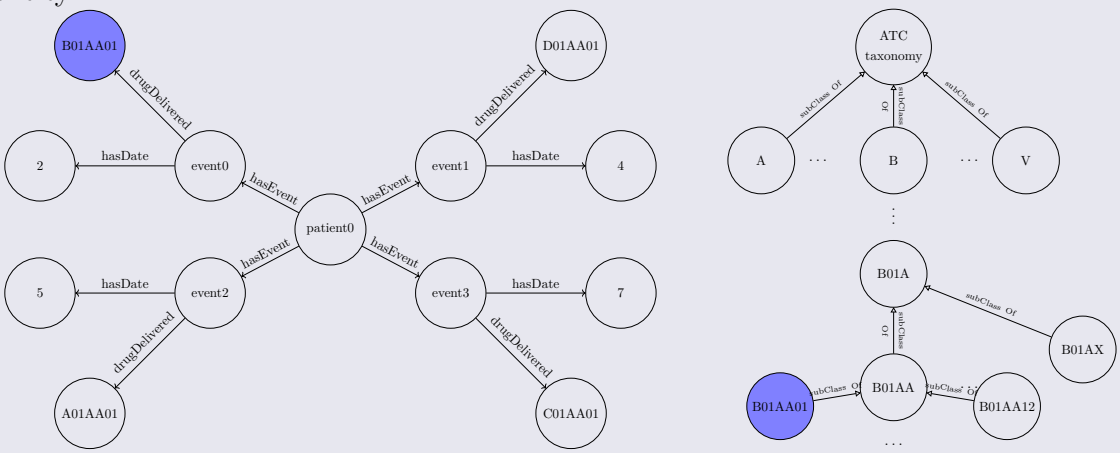
In section 4.4.1, we first describe the database format we are based on. It contains patients care sequences represented in an RDF graph. Then we propose to encode the task of taxo-Chronicle enumeration: the first approach fully uses Semantic Web technologies with the construction of a SPARQL query; a second approach we propose a hybrid tool HyCOR combining SPARQL query and a dedicated algorithm. We keep in mind that our concrete objective is to query a dataset of sequences with temporal patterns.

4.4.1 RDF to represent care sequences

Semantic Web data language RDF is suitable to represent structured data of AHDB [RDL19]. Moreover, linking raw data to standard medical taxonomies is interesting to enrich the cares description with formalized expert knowledge.

Example 7 – RDF graph of a patient with four health events

On the left we present the RDF graph representing the sequence 2 from the Table 4.1 on page 53. It is composed of 12 triples and 13 nodes. It describes a care sequence composed of four drug deliveries events: a drug delivery of B01AA01 on date 2, a drug delivery of D01AA01 on date 4, a drug delivery of A01AA01 on date 5 and a drug delivery of C01AA01 on date 7. On the right, we present a part of the graph of the ATC taxonomy. The code B01AA01 is a class in the ATC taxonomy. So on, we have the information that it is a subclass B01AA which is a subclass of B01A etc... In RDF each node is identified by a unique identifier (URI). The node in the sequence has the same URI as the node in the ATC graph, so they are the same entity.



First of all, we replicate and extend the work of Yann Rivault [RDL19]. It enables to manipulate the SNDS data, not in a tabular format, as provided by the raw data, but in a graph format. In this representation, we will find a set of triples linking each care events to a care sequence. These nodes can also be linked to external resources, such as taxonomies. The example shows the representation of the second sequence of the table 4.1 in a RDF graph.

In the following of the manuscript, we consider a set of sequences where each sequence is composed of events where each event is linked to a code and a date. Codes can be nodes in a taxonomy. Thus the database is composed of the set of sequences and taxonomies.

4.4.2 SPARQL for taxo-Chronicle occurrences enumeration

This section presents the taxo-Chronicle recognition task with SPARQL, the reference query language for RDF data. A SPARQL query enumerates all occurrences of a taxo-Chronicle in a sequence. To do so, we build a five steps query following the notations of the Definition 11 on page 94.

Step 1 specifies a query matching a graph that has a 4-events sequence. Step 2 specifies that each event has a code and a date. Then, steps 3 and 4 specify conditions of code and temporal constraints on dates. This query selects every sequence from the dataset which verifies the phenotype (represented with a taxo-Chronicle) and also selects the events composing the subsequence (the taxo-Chronicle occurrences).

1. A sequence $\boldsymbol{s}$ (`?sequence`) and a taxo-Chronicle $\mathcal{C}$, we are looking for a subsequence $\tilde{\boldsymbol{s}}$ of $\boldsymbol{s}$ composed of as many events (`?evti`) as vertices in the taxo-Chronicle.
   $\forall (e_{\epsilon_i}, t_{\epsilon_i}) \in \tilde{\boldsymbol{s}}$:

```
?sequence :isComposedOf ?evti.
```

2. Each event has a code and a date
   $\forall (e_{\epsilon_i}, t_{\epsilon_i}) \in \tilde{\boldsymbol{s}}$:

```
?evti :hasCode ?cepsi.
?sequence :hasDate ?tepsi.
```

3. The taxo-Chronicle gives information on these events, such that the code e_{ϵ_i} (`?cepsi`) is a direct or undirect class of the code c_i (`B01A`) defined by the Chronicle such that $e_{\epsilon_i} \rightsquigarrow c_i$

   $\forall (e_{\epsilon_i}, t_{\epsilon_i}) \in \tilde{\boldsymbol{s}}$:

   ```
   ?cepsi rdfs:subClassOf* atc:B01A.
   ```

4. The taxo-Chronicle gives temporal constraints on dates between events, such that $t_{\varepsilon_j} - t_{\varepsilon_i} \in [t^-, t^+]$ (`?tepsj-?tepsi` $\in$ [-7,7]) whenever $(c_i, i)[t^-, t^+](c_j, j) \in \mathcal{T}$

   $\forall (c_i, i)[t^-, t^+](c_j, j) \in \mathcal{T}$:

   ```
   FILTER ( (?tepsj - ?tepsi > 7) && (?tepsj - ?tepsi < 7) )
   ```

5. The final query is the conjunction of all these SPARQL lines. It returns all sequences and events matching the query. Figure 4.4 illustrates the SPARQL query build-up according to the previous method for the taxo-Chronicle in figure 4.3.

SPARQL is enough expressive to enumerate taxo-Chronicle occurrences. However, the enumeration of taxo-Chronicle occurrences is a very hard computational task. A SPARQL query cannot compete with dedicated enumeration algorithms as its solver strategy is not optimized for this task (see experiments in section 4.5). Therefore, we propose a hybrid approach to benefit from the best of both fields: efficiency of dedicated approaches and expressiveness of Semantic Web.

4.4.3 HyCOR for taxo-Chronicle recognition

HyCOR (Hybrid-Chronicle Occurrences Recognition) combines SPARQL and a specific algorithm to efficiently enumerate occurrences of a taxo-Chronicle Figure 4.5 illustrates the HyCOR process.

```
SELECT DISTINCT ?sequence, ?evt1, ?evt2, ?evt3, ?evt4
WHERE{
# step 1
  ?sequence :isComposedOf ?evt1;
            :isComposedOf ?evt2;
            :isComposedOf ?evt3;
            :isComposedOf ?evt4.
# step 2
  ?evt1 :hasCode ?c1;
        :hasDate ?t1.
  ?evt2 :hasCode ?c2;
        :hasDate ?t2.
  ?evt3 :hasCode ?c3;
        :hasDate ?t3.
  ?evt4 :hasCode ?c4;
        :hasDate ?t4.
# step 3
  ?c1 rdfs:subClassOf* atc:EDQM001.
  ?c2 rdfs:subClassOf* atc:B01A.
  ?c3 rdfs:subClassOf* atc:B01A.
  ?c4 rdfs:subClassOf* atc:B01A.

# step 4
FILTER ( (?t2 - ?t1 >= -7) && (?t2 - ?t1 <= 7) )
FILTER ( (?t3 - ?t1 >= 1) && (?t3 - ?t1 <= 31) )
FILTER ( (?t4 - ?t3 >= 1) && (?t4 - ?t3 <= 31) )
}
```

Figure 4.4 – Example of a SPARQL query for taxo-Chronicle enumeration

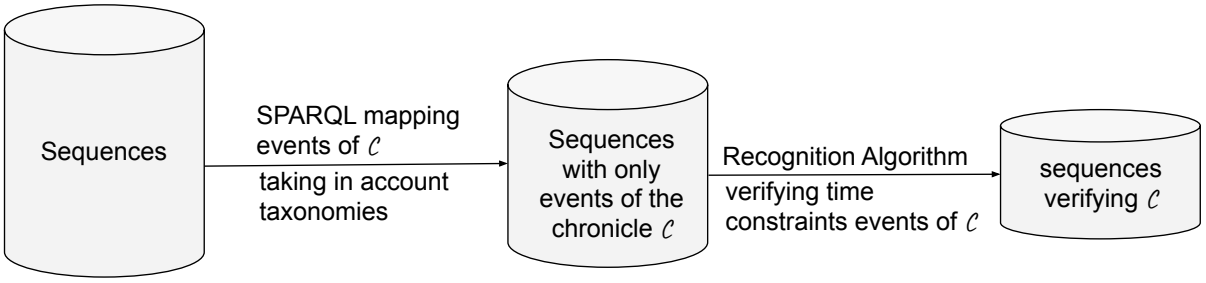


Figure 4.5 – Schema of the HyCOR process to enumerate occurrences of a taxo-Chronicle $\mathscr{C}$

Algorithm 1: Occurrences of a taxo-Chronicle $\mathscr{C}$ in a sequence $\boldsymbol{s}$.

Input: $\mathscr{C} = (\mathcal{E} = \{(c_1, 1), \ldots, (c_m, m)\}, \mathcal{T})$, $\boldsymbol{s} = \langle (e_1, t_1) \ \ldots \ (e_n, t_n) \rangle$
Output: *occs*: a set of occurrences of $\mathscr{C}$ in $\boldsymbol{s}$

1 $occs \leftarrow \emptyset$ // Set of occurrences
2 **foreach** $(e, t) \in \boldsymbol{s}$ **do**
3 **if** $e = c_1$ **then**
  // create a set of admissible positions π
4 $\pi \leftarrow \{[t, t], [-\infty, \infty], \ldots, [-\infty, \infty]\}$;
  // propagate taxo-Chronicle constraints
5 **foreach** $(c_1, 1)[t^-, t^+](c, p) \in \mathcal{T}$ **do**
6 $\pi_p = [max(t_1, t + t^-), min(t + t^+, t_n)]$;
7 $occs \leftarrow occs \cup$ RecEnumerate(π,1,$\mathscr{C}$,$\boldsymbol{s}$) ;
8 **return** *occs*

First, a SPARQL query yields flattened sequences. A flattened sequence contains only events which belong to the equivalence class of at least one event code of $\mathcal{E}$, *i.e.* the taxo-Chronicle events (see definition 4). Such a query for taxo-Chronicle of figure 4.3 is as follows:

```
SELECT ?seq ?date ?code
WHERE{
  VALUES ?code { atc:B01AB ccam:EDQM001 }
  GRAPH snds:kb { ?l rdfs:subClassOf* ?code }
  GRAPH ?seq{
    ?event snds:hasCode ?l.
    ?event snds:hasDate ?date.}
  }
```

With this query, we extract all the sequences containing at least one item of the chronicle. For each sequence, we also retrieve all the events which are subclasses of `atc:B01AB` and the events which are `ccam:EDQM001`.

Secondly, HyCOR applies Algorithm 1 [Guy20] to enumerate taxo-Chronicle occurrences in the flattened sequences. The principle of the algorithm is to progressively refine intervals in which a taxo-Chronicle even $(c_i, i) \in \mathcal{E}$ may occur in $\boldsymbol{s}$. These intervals are called *admissible positions*. The algorithm goes through the set of events $(e_i, t_i) \in \boldsymbol{s}$ and propagates the temporal constraints of the taxo-Chronicle to narrow position intervals until intervals are only a single position. Thus, admissible position designates a subsequence of $\boldsymbol{s}$, *i.e.*, an occurrence of the taxo-Chronicle. Algorithm 1 makes recursive calls to Algorithm 2.

Algorithm 2: RecEnumerate($\pi, k, \mathscr{C}, \boldsymbol{s}$).

Input: π: admissible positions, k: recursion level,
$\mathscr{C} = (\mathcal{E} = \{(c_1, 1), \ldots, (c_m, m)\}, \mathcal{T})$, $\boldsymbol{s} = \langle (e_1, t_1) \ \ldots \ (e_n, t_n) \rangle$

Output: *occs*: a set of occurrences of $\mathscr{C}$ in $\boldsymbol{s}$

```
1  occs ← ∅ // Set of occurrences
2  if k = m + 1 then
       // An occurrence has been found
3      occ ← {(e_{k_i}, t_{k_i}) ∈ s | c_i = e_{k_i}, π_i = t_{k_i}, i = 1..m};
4      return {occ}
5  foreach (e, t) ∈ s s.t. t ∈ π_k do
6      if e = c_k then
           // create a copy of admissible positions π
7          π̃ ← π;
8          π̃_k ← [t, t];
           // propagate taxo-Chronicle constraints
9          satisfiable ← true;
10         foreach (c_k, k)[t⁻, t⁺](c, p) ∈ T do
11             π̃_p ← [max(π̃_p⁻, t + t⁻), min(π̃_p⁺, t + t⁺)];
12             if π̃_p⁻ > π̃_p⁺ then
13                 satisfiable ← false;
14                 break;
15         if satisfiable then
               // Recursive call
16             occs ← occs ∪ RecEnumerate(π̃, k + 1, 𝒞, s) ;
17 return occs
```

The latter assumes that the $k-1$ first events have been located in $\boldsymbol{s}$. This means that the k first intervals of the admissible intervals π are singleton intervals. The recursive call looks for event c_k in the admissible positions of $\boldsymbol{s}$ for k-th event (lines 5-6). If found, it is a candidate for further refinements and temporal constraints of the taxo-Chronicle are propagated. The constraint $(c_k, k)[t^-, t^+](c_p, p)$ is a constraint from (c_k, k) event to the event c at position p. It is used to possibly narrow the admissible positions of event p (line 11). In case the new positions are inconsistent (line 12) then this candidate occurrence cannot satisfy the temporal constraints and is discarded ($satisfiable$ is set to $false$). If all constraints are satisfied, the recursive call attempts to refine further these positions (line 16). Note that only forward constraints are propagated. Indeed, backward constraints (*i.e.* constraint to the event at a lower position than k in the set) have already been taken into account in parent calls.

Let us illustrate the algorithm with a simple example. Let $\boldsymbol{s} = \langle(\mathsf{B},2)\ (\mathsf{C},3)\ (\mathsf{A},5)\ (\mathsf{B},6)\ (\mathsf{C},7)\ (\mathsf{C},9)\ (\mathsf{C},10)\rangle$ and $\mathscr{C} = (\{(\mathsf{A},1),(\mathsf{B},2),\ (\mathsf{C},3)\},\ \{(\mathsf{A},1)[-2,2](\mathsf{B},2),\ (\mathsf{A},1)[-3,5](\mathsf{C},3),\ (\mathsf{B},2)[-1,3](\mathsf{C},3)\})$.

1. Processing of event A
   - generates a single tuple of admissible positions $\pi = ([5,5],[-\infty,\infty],[-\infty,\infty])$
   - constraints propagation:
     - $(\mathsf{A},1)[-2,2](\mathsf{B},2)$: $\pi = ([5,5],[3,7],[-\infty,\infty])$
     - $(\mathsf{A},1)[-3,5](\mathsf{C},3)$: $\pi = ([5,5],[3,7],[2,10])$
2. Processing of event B
   - narrows positions with occurrences: $(\mathsf{B},2)$ is invalid ($2 \notin [3,7]$), but $(\mathsf{B},6)$ satisfies the admissible positions $[3,7]$ so the admissible positions can be updated ($\pi = ([5,5],\ [6,6],\ [2,10])$)
   - constraints propagation:
     - $(\mathsf{B},2)[-1,3](\mathsf{C},3)$: $\pi = ([5,5],[6,6],[2,10] \cap [5,9]) = ([5,5],[6,6],[5,9])$
3. Processing of event C
   - narrows intervals with occurrences: $(\mathsf{C},3)$ and $(\mathsf{C},10)$ are invalid, but $(\mathsf{C},7)$ and $(\mathsf{C},9)$ are valid, then the both subsequences where the taxo-Chronicle occurs are obtained by updating the admissible positions $(([5,5],[6,6],\ [7,7])$ and $([5,5],[6,6],[9,9]))$.

4.5 Experiments

In this section, we compare execution times of SPARQL and HyCOR on synthetic datasets. The HyCOR algorithm is implemented in Python with a personal computer with 16Go RAM and an SSD. The SPARQl queries have been executed with Jena-Fuseki as a SPARQL engine[2]. One of the current problems that the computer industry has to face is the maintenance of tools. The semantic web is not an exception, and the learning

[2] `https://jena.apache.org/documentation/fuseki2/`

of a query engine can be long. However, Fuseki takes up this challenge with an easy installation, quick to learn and with only few bugs found during its use. Moreover, it shows good results, especially thanks to the data format TDB that can be used as a high performance RDF store on a single machin.

4.5.1 Synthetic datasets generation and plan of experiments

Several synthetic datasets have been generated. Each dataset contains a set of sequences where event codes are randomly sampled at the lowest level of ATC taxonomy. The ATC taxonomy contains 1 900 classes. In addition, occurrences of ten 15-sized taxo-Chronicles[3] are embedded in the dataset. For each taxo-Chronicle, a constraint is generated for each pair of events without inconsistency between the temporal constraints.
The synthetic dataset generation process ensures that each taxo-Chronicle occurs in about 20% of sequences. taxo-Chronicles contain event codes from several levels of ATC following this probability: $\frac{1}{15}$ level 1 (ex: `N`), $\frac{2}{15}$ level 2 (ex: `N02`), $\frac{3}{15}$ level 3 (ex: `N02B`), $\frac{3}{15}$ level 4 (ex: `N02BE`), $\frac{6}{15}$ level 5 (ex: `N02BE01`).
We introduce the notation $D_{ns,ne}$ to denote a synthetic dataset with ns sequences and ne care events per sequence (all sequences have the same number of events). For the following experiments, 25 synthetic datasets have been generated where $ns \in \{1\,000, 5\,000, 10\,000, 15\,000, 20\,000\}$ and $ne \in \{100, 200, 300, 400, 500\}$. Each dataset is encoded in RDF. The ATC taxonomy is attached to the dataset.

4.5.2 Experiments and Results

The following experiments evaluate the impact of two main parameters on execution times of SPARQL and HyCOR: the size of the dataset (number of sequences and number of events per sequence) and the chronicle size.
Figure 4.6 compares the execution times of SPARQL and HyCOR with respect to the length of the sequences. It shows that HyCOR is at least one order of magnitude more efficient than pure SPARQL. Figure 4.6 on the facing page shows that SPARQL does not scale up for datasets containing more than 15 000 sequences. The HyCOR SPARQL query language does not have the same limitation. Indeed, the pure SPARQL query uses filters to deal with temporal constraints on each admissible event while SPARQL HyCOR query only uses values to find admissible events. So, the use of filters on a large scale of

[3] A n-sized-taxo-Chronicle denotes a taxo-Chronicle of size n.

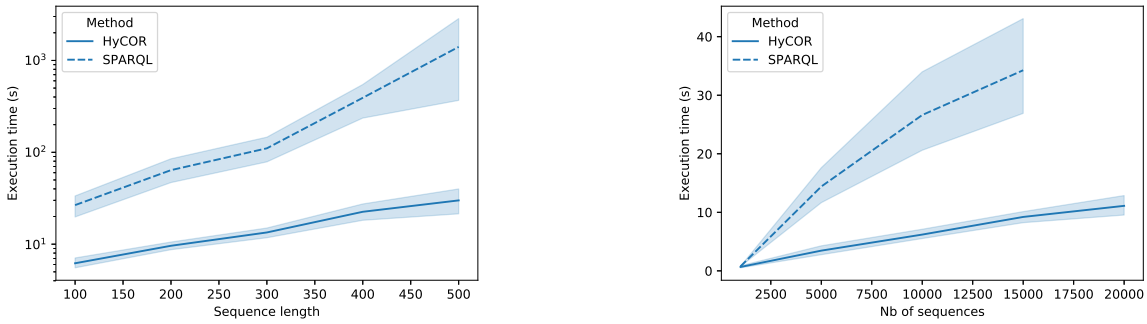


Figure 4.6 – Execution times (in seconds) of SPARQL and HyCOR wrt sequences length on 10 000 sequences (on the left) and wrt number of sequences (with length 100).

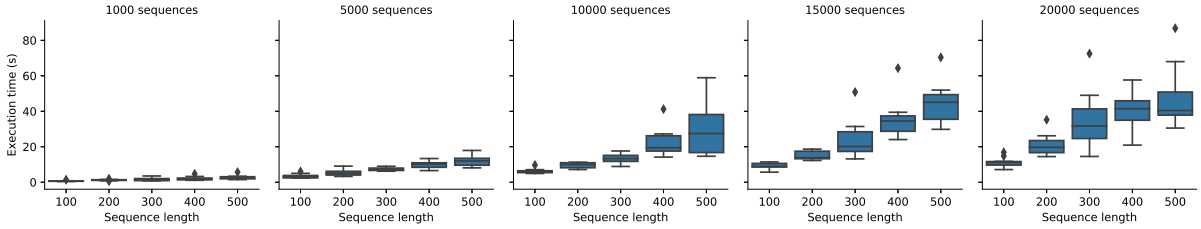


Figure 4.7 – Execution times of taxo-Chronicle occurrences enumeration wrt sequences length.

admissible events seems to be inefficient in SPARQL for this kind of use.
We also evaluate the part of HyCOR execution times spent by the SPARQL mapping and the taxo-Chronicle enumeration algorithm. On average, the SPARQL query execution represents $85\% \pm 3.47$ of the total execution time.

Experiments now focus on the HyCOR evaluation. Figure 4.7 illustrates the impact of the number of events per sequence on the execution times. We observe that time linearly increases with the number of events and of sequences. We can notice in the hardest condition: $D_{20\,000,500}$, enumeration of a taxo-Chronicle with 15 events takes in average less than a minute.
Figure 4.8 presents the execution time of HyCOR wrt chronicle size. HyCOR is executed on a unique dataset ($ns = 10\,000$ and $ne = 100$) and seven sets of 10 taxo-Chronicles with sizes 2 to 14. We observe execution time linearly increases with the chronicle size. HyCOR outperforms pure-SPARQL for the taxo-Chronicle enumeration task. Its execution time increases with the chronicle sizes and the dataset size, but it still offers good results for large datasets.

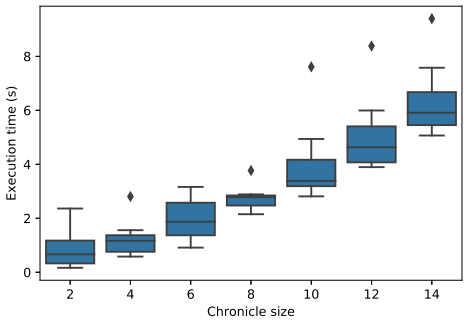


Figure 4.8 – Execution time (in seconds) wrt chronicle size ($ns = 10\,000$ and $ne = 100$)

4.6 Conclusion

We extended the Chronicles model with taxonomies to enumerate complex temporal patterns in sequences. This problem is motivated by the need for phenotyping patients in the SNDS database. Thus we proposed to start from an RDF data model that contains sequences of events dated and linked to taxonomies and we check if a chronicle occurs in these sequences. We proposed HyCOR, an hybrid approach that combines the expressiveness of SPARQL and the efficiency of a dedicated algorithm. The results show that HyCOR is one order of magnitude faster than pure SPARQL queries. It shows the usefulness of using a temporal model: one can describe the desired temporal constraints through an expressive formalism, and the execution of the task is efficient. Moreover, this formal approach should encourage the reusing of phenotypes. A common model should facilitate its reuse for other studies or other similar databases.

This approach achieves four of the objectives set out in the introduction, namely: use taxonomies and metric time constraints in the expression of phenotypes (listed in Table 4.2), and an execution time of a few minutes per phenotype avoiding the preprocessing of data (listed in Table 4.3). HyCOR can be found in the following deposit : `https://gitlab.inria.fr/jbakalar/hycor`

The data format used in this chapter restricts the amount of information that can be retrieved from a database like the SNDS. In the next chapter, we focus a data model that enable to detail care sequences.

Use taxonomies	✓
Expressing metric temporal constraints	✓
Use all types of criteria (diagnosis, medical acts, hospitalization, etc.)	

Table 4.2 – Criteria of evaluation for expressivity of the temporal model

A few minutes per phenotype for a large dataset	✓
No transformation of data at each phenotype	✓

Table 4.3 – Criteria of evaluation for efficacity of the temporal model

Chapter 5

Data model with OWL

The French national health insurance database (so-called SNDS) almost covers all French population (about 66 million inhabitants) [Tup+17; Bez+17]. It contains information about drug deliveries, medical procedures, and hospitalizations. The advantage of this database is to gather information about all reimbursed medical events, from drug deliveries to hospitalizations and specialist consultations. In Chapter 1 on page 11 we explained that the range of medical events recorded in the database makes it suitable to carry out a wide variety of pharmaco-epidemiological (PE) studies [Pal17].
However, the SNDS has been designed for administrative purposes (care reimbursements). It uses a relational data model, composed of over 2 500 tables from several resources and does not follow any data model described in the state-of-the-art. So, the lack of a system to formalize and unify data makes their manipulation for PE purposes difficult [Koz+15].
In this chapter, we propose a data model adapted to facilitate the use of the SNDS for pharmaco-epidemiology purposes. As can be seen on the Figure 5.1 on the following page, we are at the data formalization part of our problematic. In the state of the art Chapter 2 on page 19, we define the notion of data, data schema, data semantics and data format for common data models. We showed that data schemas were enable to organize data. So we were interested in OWL [AH11] to propose a suitable model to organize SNDS data. We previously saw that patients selection requires temporal constraints and ontologies. Thus, the new data model facilitates the management of data with ontologies and temporal criteria. As the Semantic Web has been demonstrated to be well adapted to manage HADB [RDL19], we present an OWL [MV+04] representation to describe data through concepts, relations between concepts including ontologies management. As M.Bienvenu [Bie16] explains *"In information integration, ontologies serve to relate the vocabularies of different data sources and to provide a unified view to the user."*

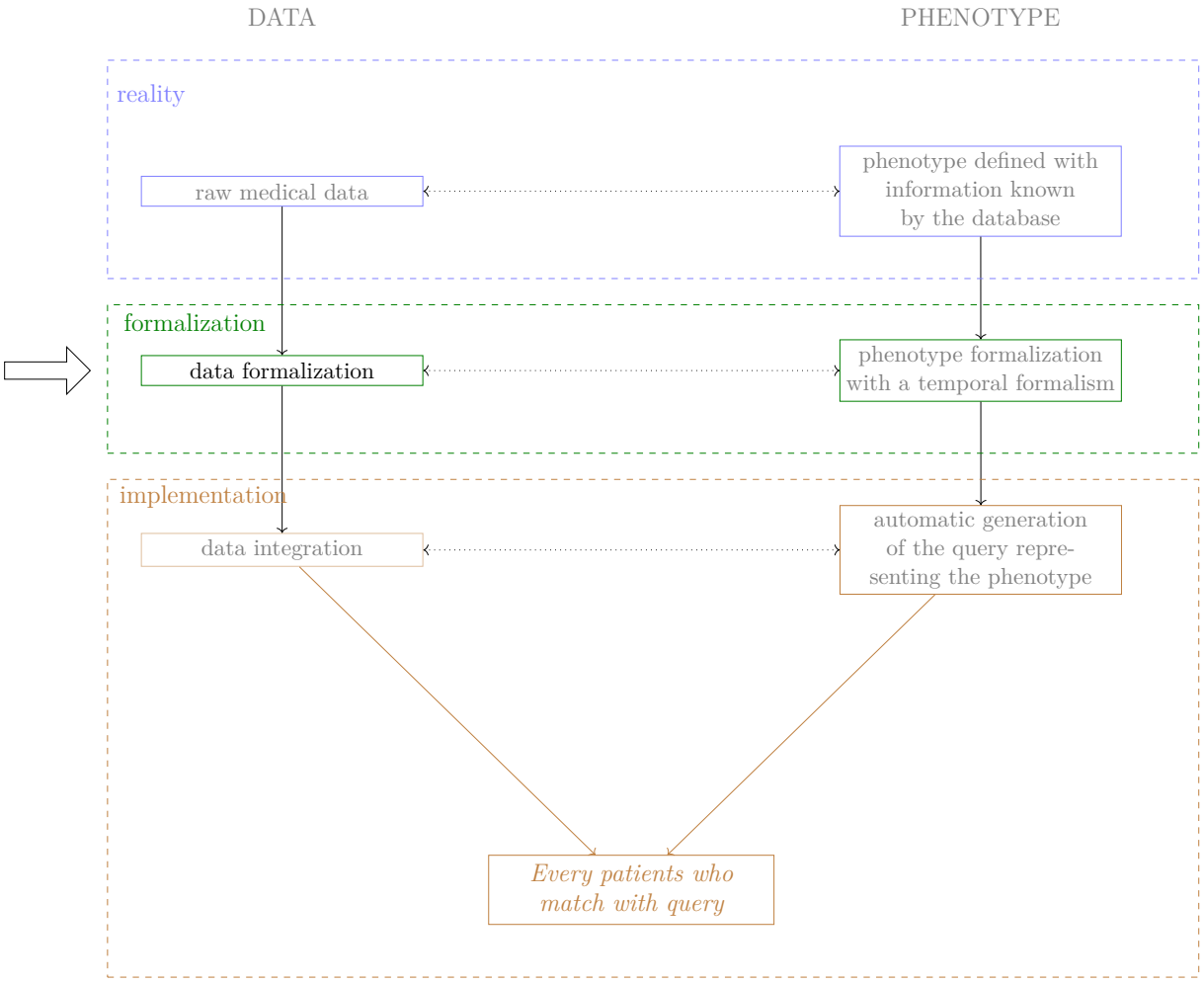


Figure 5.1 – Remind of the problematic, this chapter focuses on data formalization

5.1 Data description and taxonomies

The SNDS contains individual data used for billing and reimbursement of outpatient health care consumption (DCIR: Données de Consommation InterRégimes) as well as private and public hospital data at the individual level, collected in PMSI (Programme de Médicalisation des Systèmes d'Information) by the French Agency for information on hospital care, and provides individual information on causes of death, collected by the CépiDC (Centre d'épidémiologie sur les causes médicales de Décès). is also provides aggregates data collected in DAMIR (Dépenses d'Assurance Maladie Inter-Régimes) which are not suitable for studies at the individual level [Tup+17].
The data are distributed in these different resources. However, similar types of information can be found in several resources. For example, patients who have a Doppler exam are stored in different parts of the database depending on whether it was performed in hospital(PMSI) or out hospital(DCIR).
The manipulation of disparate and unified data makes queries complex. However, the Semantic Web offers efficient solutions to unify disparate data sources together to present a single view. To apply these solutions on SNDS data, we first need to highlight the common concepts that are in the different resources [Thu+19]. To do this, we detail each resource below to detect the concepts they contain. We frame them in this way.

The beneficiaries

- Demographic characteristics of a beneficiary includes gender, age, the rank of birth (twins), geographical code of the town of residence, and health insurance coverage are recorded for each beneficiary.

The DCIR It contains the nature and date of medical services out hospital.

- Drug deliveries: all claims for drugs dispensed by community pharmacies from public or private sector prescribers, including hospital specialists. Drug packages are identified using a French national coding called CIP (Club Inter-Pharmaceutique) corresponding to a bar code. A mapping table by CIP package codes provides their universal Anatomical Therapeutic Chemical (ATC [1]) codes, the number of items in the package (*e.g.* 28 tabs), and dosage of the item (*e.g.* 500 mg).

[1] `https://bioportal.bioontology.org/ontologies/ATC`

- Medical acts: outpatient medical and surgical procedures or imaging examinations are coded according to the French medical classification for clinical procedures (CCAM [2]).

- Medical devices and services related to the functioning of these devices are recorded using the French LPP classification.

- Laboratory exams: the tests performed on an outpatient basis are coded according to NABM [3].

- Performers and Prescribers: general practitioners and specialists, dentists, midwives, physiotherapists, speech therapists, orthoptists, nurses, and podiatrists- chiropodists, in their offices, at the patient's home, in private clinics, or in some health or medical and social welfare centers – also recorded in the SNDS.

- Long-term diseases (LTD): the doctor in charge of a particular patient can request 100% reimbursement of expenditure related to certain long-term diseases (ALD, coded according to the classification of diseases ICD-10 [4]). Eligible diseases are registered on a list of 30 groups of major chronic illnesses. Full reimbursement is also possible for some diseases not included on this list or for multiple disease patients. The SNDS records the different periods during which patient expenditures have been covered by such a device and the corresponding ICD-10 code of the chronic disease.

The PMSI Hospital health care consumption corresponds to hospitalizations in short-stay institutions (medicine, surgery, obstetrics: MCO), aftercare and rehabilitation (SSR), psychiatry (RIM-P), and hospital at home (HAD). The available information concerning hospital health care consumption is derived from anonymous discharge summaries established at the end of each stay:

- Length of stay, day, month and year of discharge, origin and destination before and after the stay,

[2] `http://bioportal.lirmm.fr/ontologies/CCAM`
[3] `http://bioportal.lirmm.fr/ontologies/NABM`
[4] `https://bioportal.bioontology.org/ontologies/ICD10`

- Diagnosis: There are principal diagnoses (DP), and eventually, related diagnoses (DR) and/or eventually associated diagnoses (DA) coded according to ICD-10 bound to a hospital stay.
- Certain medical and surgical medical acts are coded according to CCAM.
- Drugs delivered In hospital: current drugs used during hospital stays are not available in the SNDS. Only costly and innovative drugs dispensing data are specifically recorded for administrative purposes. Those drugs are identified through UCD (Unité Commune de Dispensation) codes, using a national coding scheme distinct from the CIP system used for ambulatory drugs. Hospital pharmacies can also dispense some unlicensed drugs to ambulatory patients under exceptional and temporary conditions, in order to enable patients with no more treatment options to benefit from drugs not yet approved based on the program of Temporary Authorization for Use (TAU) [Pal+16].

The CépiDC It contains the information on the beneficiaries death such as death causes, geographical code of the town of death, date of the death, profession and socio-professional category and several information if pregnancy contributed to the death.

This by resource distribution makes access to the information and their reconciliation tedious for PE studies. In the following section, we generalize the medical notions contained in the database through the different concepts captured there
For more information about the actual relational model of the SNDS, the reader can refer to the web documentation[5].

5.2 Concepts extracted from the SNDS

The definition of concepts simplifies the use of these data. They capture all relevant information for the PE studies, and help for a unified data view.
In the previous descriptions, we extracted ten main concepts listed in column 1 of the Table 5.1 on the following page. Each concept has several properties listed in column 2. Some properties are linked to taxonomies listed in column 3. For example, the concept of Drug Delivery has three properties:

[5] `https://documentation-snds.health-data-hub.fr/`

Concept	**Properties**	**Taxonomy**
Beneficiary	date of birth	
	town of residence	
	gender	
Beneficiary death	date of death	
	causes of death	
Drug delivery	drug code	CIP code, UCD code } ATC corres.
	quantity of boxes delivered	
	date of the delivery	
Long-term disease (LTD)	LTD code	number between 1 and 31
	diagnosis code	ICD-10
Medical device	device code	LPP
	quantity delivered	
	period of the delivery	
Medical act	act code	CCAM
	date	
Laboratory exam	exam code	NABM
	date	
Diagnosis principal diagnosis related diagnosis associated diagnosis	 diagnosis code diagnosis code diagnosis code	ICD-10

Table 5.1 – Concepts extracted from the SNDS

- a drug code which is CIP code or UCD code, both having correspondence with an ATC code
- a number of drug boxes delivered
- a date of the delivery

This example shows we need more information about the drug box delivery performer: is it a drug box delivered by a hospital or a pharmacy? And who prescribed it?
This information is available in the SNDS. To clarify this notion, we add concepts about the Prescribers and the Performers:

- Pharmacy
- Specialist
- Hospital: short stay, aftercare and rehabilitation, psychiatry and home hospitalization

For example, we keep the information whether a Doppler exam is performed within a hospital or by a specialist in a private institute.

The new data model must capture these concepts while maintaining all the information used in PE studies. To ensure the unified view, taxonomies used in data normalization must be included. Note that some types of data such as any information about cost, reimbursement rate, social security company, or any SNDS information essentially administrative/economic are not kept in the following data model as they are useless for pharmaco-epidemiologists.

5.3 A data model for the SNDS adapted for pharmaco-epidemiology purposes

As seen in the state-of-the-art Section 2 on page 19, the Semantic Web is a framework designed to represent, share and manipulate structured data, such as ontologies and relational databases. The keystones of the Semantic Web are :

- formal data representations
- the ability to represent knowledge
- the ability to represent seamlessly data and knowledge, and
- the SPARQL query language.

Semantic Web data language (*e.g.* RDF) is suitable to represent structured data of AHDB [RDL19]. Moreover, linking raw data to standard medical taxonomies is interesting to enrich the cares description with formalized expert knowledge.

5.3.1 SNDS ontology

First of all, we extend the work of Yann Rivault [RDL19] proposing a patient-centered data model for the SNDS based on the Semantic Web. This is the data format used in the previous Chapter (Chapter 4 on page 49). RDF data enable to manipulate the SNDS data, not in a tabular format, as provided by the raw data, but in a graph format. This graph

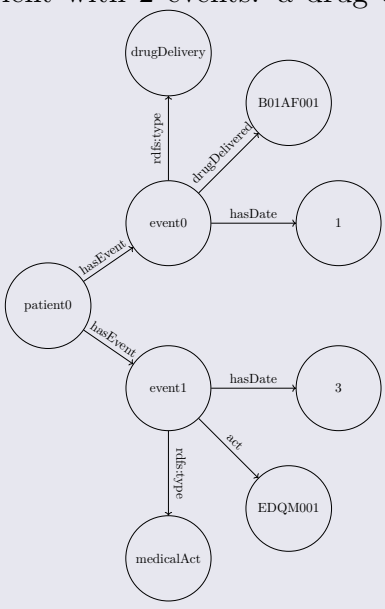
data contains a set of triples linking each patient medical events to the corresponding patient: hence we can specify the "type" of the event. These nodes can also be linked to external resources, such as taxonomies.
In the Section 5.1 on page 71, we framed the following concepts designating the different types of events that exist in the SNDS. These concepts are also linked to a taxonomy. For example, drug dispensing is known to have an ATC or UCD code. We add the following concepts, with their corresponding taxonomies: Drug Delivery, Long-term disease, Medical device, Medical act, Laboratory exam and Diagnosis. The example illustrates the RDF graph of a patient with an anticoagulant delivery and a Doppler imaging procedure:

Example 8 – RDF graph of a patient with two health events

This example represents a RDF graph, composed of 8 triples:

$$G = \langle\ (patient0, hasEvent, event0),\ (patient0, hasEvent, event1),$$
$$(event0, type, DrugDelivered),\ (event1, type, MedicalAct)$$
$$(event0, drugDelivered, B01AF01),\ (event0, hasDate, 1),$$
$$(event1, act, EDQM001),\ (event1, hasDate, 3)\ \rangle$$

This graph describes a patient with 2 events: a drug delivery of B01AF01 and a medical act of EDQM001.



We notice that the RDF data used in Chapter 4 on page 49 does not exploit OWL abilities to represent seamlessly data and knowledge or even to clarify the data model schema. So, we propose on Figure 5.2 on page 79 an OWL model to represent SNDS data schema related to pharmaco-epidemiology. OWL (Web Ontology Language) is the

computational logic-based language of the Semantic Web. It provides a rich and complex vocabulary to link classes and objects together. It also proposes constraints on properties simply specifying the domain and range, and introduces the concepts of restrictions, enumerations, and dataranges. It allows us to define a database schema by defining concepts (as OWL classes), data properties and relations among classes and objects.

To define a patient-centered database schema for the SNDS, we create the concept of **Patient** (*i.e.* beneficiary). On the left of the Figure 5.2 on page 79, we define a **Patient** OWL class having the properties defined in Table 5.1 on page 74.

We also introduce the concept of **Sequence** and care **Events**, where each patient has at least one sequence. A sequence is composed of zero to several care events. An event has 2 data properties (middle of the Figure 5.2 on page 79): a start date and an end date. It can be performed/prescribed by one performer/prescriber which can be: pharmacist, hospital or specialist (middle of the Figure 5.2 on page 79). As seen before in Section 5.2 on page 73, hospitalizations are divided in four categories: short stay, aftercare rehabilitation, psychiatry, and home hospitalization.

The data model presented on Figure 5.2 on page 79 includes a Knowledge Base containing all the taxonomies/ontologies insuring the data normalization. These taxonomies are listed in the column 3 of the Table 5.1 on page 74. The use of classification for SNDS data is a great asset for the unified view of the SNDS, and to enrich the description of cares with formalized expert knowledge.

Following the information previously listed in Table 5.1 on page 74, an Event may be a specific class: a Drug Delivery, a Medical Act, a Laboratory exam, a Medical Device, a Diagnosis or an LTD (Long-Term Illness). These are defined as OWL classes on Figure 5.2 on page 79. Each of these OWL classes is linked to its corresponding nomenclature. As represented at the top of the Figure 5.2 on page 79, these taxonomies compose a Knowledge Base. So on, an event is linked with the relation "hasCode" to a class issued from a taxonomy of the Knowledge Base.

One of the advantages of a Knowledge Base in OWL is the possibility to add other medical knowledge. Even if the SNDS only uses the nomenclatures detailed in the OWL model,

we can add more information. For example, the SNOMED CT ontology can be integrated. The National Library of medicine[6] implemented a mapping from the SNOMED CT ontology to the ICD-10 ontology. As from one class individuals may be members of multiple Class, this mapping can be integrated to the Knowledge Base and enable the user to use SNOMED CT or ICD-10 according to his preferences: the Knowledge Base has the information if a drug code is related to a class from SNOMED CT and ICD-10.
Beyond the existing medical ontologies, a pharmaco-epidemiologist can create his own ontology (for example, with *Protégé*) and integrate it into the Knowledge Base. In the case of our example of venous thromboembolism, we can consider creating an ontology *medical acts for venous thromboembolism* listing all CCAM coding medical acts that can be performed when thrombosis is suspected.

In conclusion, this OWL model formalizes the SNDS proposing a unified patient-centered view. It is composed of a set of classes extracted from the data where we add the concepts of Sequence and Event to allow a patient-centered view. Thanks to this modelling, users can visualize the organization of the database, the concepts included, and the information contained. Moreover, the use of OWL allows us to integrate medical knowledge (*i.e.* Knowledge Base) and thus to manage the data with ontologies. It contributes to developing generic methods to reason on these data. This is the subject of the following chapters.

5.3.2 RDF data following the SNDS ontology

The OWL model allows us both to reason about our data such as checking the consistency of constraints, and to define what we can write with RDF by insuring a valid model of the database. The RDF data will correspond to the instantiation of the classes defined by the model. In example 8, the instance $Patient1$ is an instance of the class Patient, the event $event0$ is an instance of the class Event and is of type "DrugDelivery". Thus, we obtain an SNDS database composed of the SNDS ontology presented on Figure 5.2 on the facing page, of a knowledge base containing taxonomies or other ontologies, and of patients, sequences et eventes which are dated, typed and coded.
Let us take the example of a patient exposed during three months to antithrombotic drugs recorded with a RDF graph validating the previous model.

[6] `https://www.nlm.nih.gov/research/umls/mapping_projects/snomedct_to_icd10cm.html`

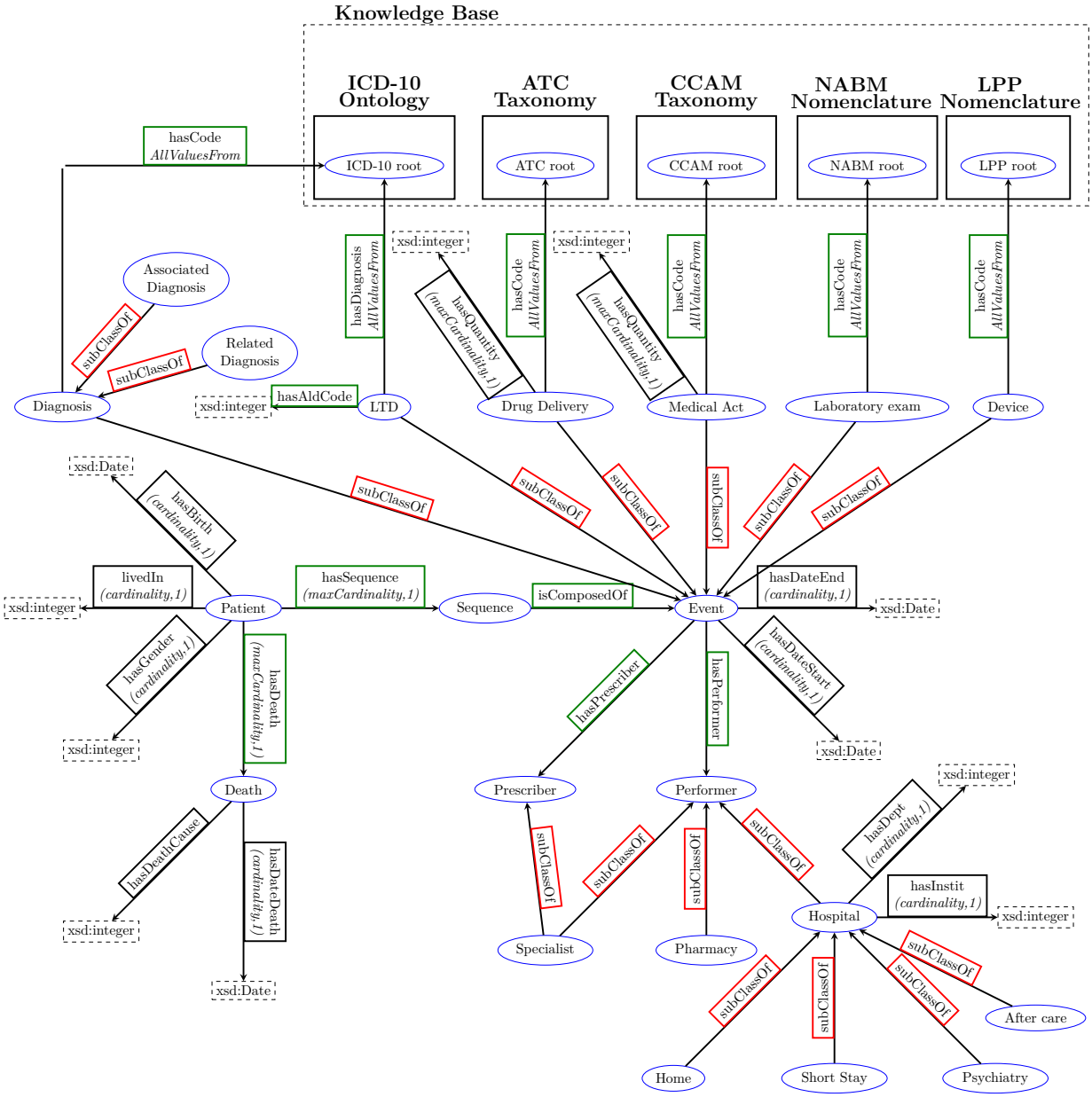
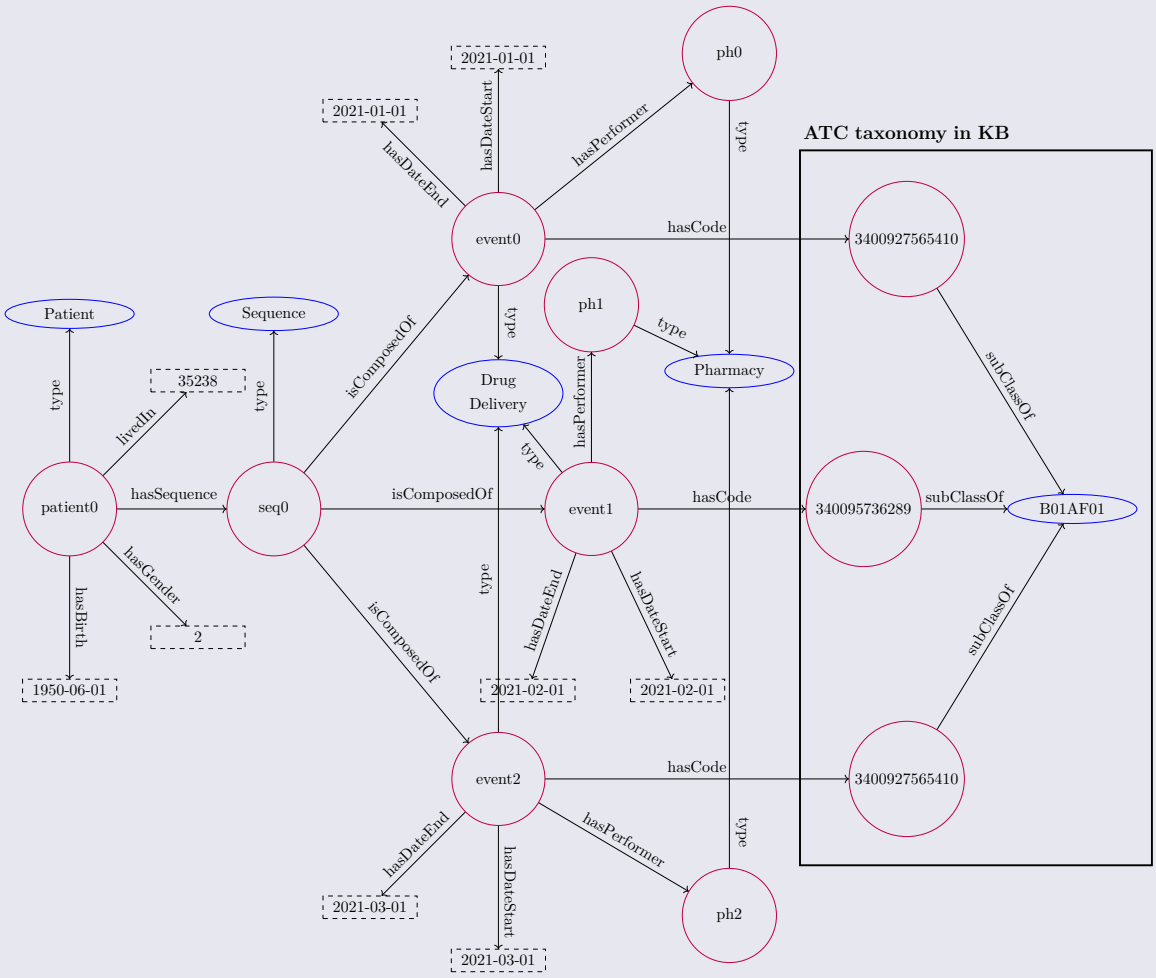


Figure 5.2 – OWL model of the SNDS for pharmaco-epidemiological studies

Example 9 – Patient represented with an RDF-graph compliant with the OWL model of the SNDS

Ellipse blue node represent owl classes, red node represent instances and black dotted node represent data properties



We define the following care sequence of a patient: the *patient0*, an instance of the class Patient, lived in the geographical code *35238*, has birth at date *1950-06-01* and has a gender *2*. He has a *seq0*, an instance of the class Sequence, which is composed of three instances of Drug Delivery: *event0*, *event1* and *event2*. To refer to the OWL model 5.2, Drug Delivery is a subclass of Event. Each event drug delivery has a Perfomer which are instances of Pharmacy: ph0, ph1 and ph2. Each event drug delivery has a code referring to a cip-code where cip-codes are instances from the ATC taxonomy. To refer to the OWL model 5.2, ATC is part of the Knowledge Base. Each event drug delivery has a start date and an end date. *event0* occurs at date "2021-01-01", *event1* occurs at date "2021-02-01" and the third one (*event3*) at date "2021-03-01".

In summary, this example represents a care sequence of a patient having a sequence composed of three events. We conclude this patient is a woman living in Rennes. She is 71 years old, and she has been exposed to antithrombotics during 3 months, as she has one drug delivery of antithrombotics per month during three months.

RDF triples and URI references The RDF-graph, example, represents a collection of RDF triples. Named things (classes, properties and individuals) have unique identifiers which are URIs. Thanks to this, we can refer to things in another ontology. In this thesis, we use the URIs and prefixes defined by the community [7], and we introduce a generic prefix: *snds* for the instances of classes excluding those contained in the Knowledge Base. The prefixes chosen for the ontologies contained in the Knowledge Base are listed below.

```
# Generic prefixes and URIs
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>

# Prefixe referring to the ontology base
PREFIX : <http://www.semanticweb.org/bakalara/ontologies/2021/snds#> .

# Prefixes for the ontologies
PREFIX nabm: <http://bioportal.lirmm.fr/ontologies/NABM>
PREFIX ccam: <http://bioportal.lirmm.fr/ontologies/CCAM>
PREFIX atc: <https://bioportal.bioontology.org/ontologies/ATC>
PREFIX icdten: <https://bioportal.bioontology.org/ontologies/ICD10>
PREFIX lpp: <http://bioportal.lirmm.fr/ontologies/LPP>

# Prefixes for instances of ATC: cip codes and ucd codes
PREFIX cip: <https://bioportal.bioontology.org/ontologies/CIP>
PREFIX ucd: <https://bioportal.bioontology.org/ontologies/CUCD>
```

The implementation of the RDF graph will use the previous prefixes. The classes, in blue, use the base prefix (Colon). The instances of these classes, in purple also use the base prefix (Colon). The B01AF01 class uses the `atc` prefix as it is a class from the ATC taxonomy and its instances use the `cip` prefix as they are cip codes.

All the predicates use the base prefix, except the 'type' predicate which is a `rdf`:property, so it uses the `rdf` prefix. In Chapter 7, you will find the transformation of the raw SNDS data to RDF SNDS data where we detail the information extracted from the SQL tables.

[7] http://prefix.cc/

5.4 Conclusion

In this chapter we have detailed the information contained in the SNDS to extract the main medical concepts useful for using this database in pharmaco-epidemiology. To do this, we used OWL which provides a language for defining structured Web ontologies, and proposed OWL data model, which can also be called an ontology, to unify the SNDS data. In addition to the medical concepts that emerged from the study of the database (Drug Delivery, Laboratory Exam, Prescriber, etc...), we introduced the concepts of patient, sequence and events which gives a patient-centered view. We thus have explicitly the notion of patient, and sequences of events where the events are dated, types, have prescribers and performers. Thus we obtain a database composed of a knowledge base, an ontology as a data model and a set of instantiation of the classes defined by the ontologies. The whole is represented thanks to the OWL and RDF syntax.

In the original data format, acts are stored in several tables depending on which institution performed them. The various tables are not linked to each other (to remind the SNDS raw data follows a star schema), thus, to get access to this information, we need to list all tables containing medical acts and to join them. We quickly arrive at very complex SQL queries, especially if we add temporal constraints to these elements. Therefore, this star schema is not adapted for patient-centered queries and do not enable to include outside resources like ontologies. The RDF format guided by an ontology is an added value. RDF data simplify queries. In the proposed data model, patient care sequences are explecitely represented thanks to the concept of dated event. Hence, using temporal constraint is more easy, as they can be directly applied on the event concept. Furthermore, it is more simple to exploit external resources (graph) with ontologies.

Futhermore, ontologies offer the possibility of grouping objects in the form of concepts (classes) that will simplify data management. We thus keep the advantages of RDF with an additional syntax provided by OWL. Therefore, we can understand the data, unify them by integrating ontologies and reason on the proposed model. This data model is crucial for querying data for sequence extraction purposes. It enables to develop detailed items in our chronicles and to find efficient algortihms to encode them. The Chapter 6 on the next page will propose to extend Chronicles based on the knowledge given by this ontology.

Chapter 6

Chronicles extended with OMQA-Ontology-Mediated Query Answering

In Chapter 4 on page 49, we extended the definition of Chronicles with events related to taxonomies. However, the SNDS ontology proposed in Chapter 5 on page 69 gives us more information on the composition of an event. Indeed, an event is an instance of a class of the SNDS ontology and we have a set of information linked to this event. For example, in the description of the VTE phenotype we are interested in a Doppler imagery performed on an ambulatory basis or in hospital. This detail cannot be specified in the description of the phenotype with a taxo-Chronicle. However, this information is available as the database is based on the SNDS ontology. To be able to specify events in the construction of phenotypes, we need to extend the Chronicle formalism. The goal is to be able to detail an event with more relations than a single "subClassOf". For example, we want to specify an event which has a type "MedicalAct" and which has as performer an hospital.

In the state of the art, we presented the field of Ontology-Mediated Query Answering (OMQA) [Pog+08; Bie16] which rethink the querying of ontology-based data. Artale *et al.* [Art+17] explored the queries over temporal data where First Order Logic (FOL) formulae are a basis to construct ontology-mediated queries.
This chapter proposes to extend the definition of Chronicle by extending the notion of event. We propose to define a Chronicle item as a logical formula, and more precisely a First Order Logic formula (FOL) which will allow us to formalize the notion of event occurrence in a Chronicle. Then we will rely on the work done by Artale *et al.* [Art+17] to transform FOL into an Ontology-Mediated Query over temporal data.

6.1 First order Logic to describe data and OMQ to query them

In this section, we make the link between First Order Logic and OWL ontologies in order to work with FOL formulas. First we define an ontology with FOL formula and then we detail the construction of Ontology-Mediated Query based on ontology defined with FOL definitions. We assume that we have point-based temporal data, which means that data are only linked to a single timestamp. In case of event occuring on a time interval, we are only interested in the start date of this event.

6.1.1 First Order Logic to describe data - semi-temporal ABox and TBox

A OWL-onlotogy is characterized by four aspects [Bie16; DL96] :

- A set of concepts and roles, this set is called an *ABox* :
  - Concepts with one or more domain arguments. It corresponds to the term of class in OWL. For example, *Event* or *DrugDelivery* are concepts.
  - Roles with two domain arguments (for example, *hasCode(e,c)*). They are corresponds to relations in OWL. It exists two types of relations in OWL : *object properties* and *data properties*. *Object properties* connect pairs of individuals and data properties connect individuals with datatypes. Datatypes are entities that refer to sets of data values. So, datatypes are analogous to classes, the main difference being that the former contain data values such as strings and numbers, rather than individuals (literals, date, integers, and so on..) [1].
- A set of concept and roles assertions on individual objects (instance assertions). This set is included in the *ABox*. For example, *event*1 is an assertion/individual of the concept *Event* and the role *hasCode(event1,c)* means the assertion/individual *event*1 has a code *c*.
- A set of universally quantified assertions, called the *TBox*. It defines general poperties on concepts and roles For example, B01AB01 is a subclass of B01AB, or the "subclassOf" is a transitivity property, or even cardinality restrictions.

[1] `https://www.w3.org/TR/owl2-syntax/#Datatypes`

- Inference mechanisms for reasoning on both the $TBox$ and the $ABox$. It consists in checking whether a certain assertion is logically implied by a knowledge base. For example, we can find the statement "if B01AB is a subclass of B01A then B01AB01 is a subclass of B01A ".

An Ontology is thus described through an $ABox$, a $TBox$ and an inference mechanisms. Figure 6.1 on page 87 illustrates the Abox of the SNDS database. concepts are defined in the blue frame, roles in the green frame and data properties in the gray frame, with an example of assertion for each. We can consider a representation of the SNDS database as this ABox with all the assertions and the TBox is finally already defined in the Chapiter 5 on page 69 on the Figure 5.2 on page 79 with relatiosn between concpets ans restrictions, plus the Knowledge Base. Inspired by the definition of A.Artale *et al.* [Art+17], we propose to describe a semi-temporal $ABox$ based on FOL formalism in description 6.

Definition 6 (Semi-temporal ABox) *A semi-temporal ABox, denoted $\mathcal{A}$, consists of assertions of the form:*

$$C_k(c_i, t) \quad D_k(d_i) \quad R_k(r_i, r_j)$$

where C_k and D_k are concept names, R_k is a role name, c_i, d_i, r_i, r_j are individuals names, and $t \in \mathbb{Z}$ a timestamp.

This definition defines concepts of arity 1 (D_k) and concepts of arity 2 which are timed (C_k). For example, in the SNDS, there is the concept of DrugDelivery(d) which is of arity 1, whereas the concept of event Event(e,t) is of arity 2. On the other hand, there are no role that is timed. The roles are always of arity 2. Figure 6.1 on page 87 proposes a visualization of the SNDS ontology in the form of an ABox. To compare with the OWL representation Figure 5.2 on page 79, concepts are the element surrounded in blue and roles are the links framed in green. On the Figure 6.1 on page 87, timestamped concepts are in bold. There are only three of them: events, birth and death. Having only three temporal concepts simplify our problem and will enable us to propose efficient queries.

Then, Artale *et al.* [Art+17] introduce the interpretation domain $\mathcal{I}$ adapted for our temporal $ABox$. In definition 7 on the next page, we introduce the interpretation domain which defines the notion of individuals, *i.e.* the concepts are instantiated and these instances can also be the same instances as those in the roles. As a reminder, the role allows to link by relation two individuals.

Definition 7 (The interpretation domain $\mathcal{I}$) *Let $T \subseteq \mathbb{Z}$ be a interval. A $T-$ interpretation $\mathcal{I}$ is a structure:*

$$\mathcal{I} = ((T, <), \delta^I, R_1^I, R_2^I, \ldots, C_1^I, C_2^I, \ldots, D_1^I, D_2^I, \ldots, c_1^I, c_2^I, \ldots)$$

such that $<$ is the standard linear order on $\mathbb{Z}$ restricted to T, $\delta^I \neq \emptyset$ is the interpretation domain, $R_k^I \subseteq \delta^I \times \delta^I$ and $C_k^I \subseteq \delta^I \times T$, for all k, and $c_i^{\mathcal{I}} \in \delta^{\mathcal{I}}$, for all i. The domain δ^I is time-independant.

For example, the green roles Figure 5.2 on page 79 link two individuals while black roles Figure 5.2 on page 79 link an individual with a date or an integer. In OWL, each of these concept and role has individuals. This set of individuals composed the RDF graph presented in Chapter 5 on page 69. In OWL, the "black roles" are called "data properties". To simplify the problem, we have not included data properties in the possible role type in the temporal $ABox$, but adding this role type does not modify the following results. Concerning the $TBox$, it stays untemporal. Sarker *et al.* [Sar+17] propose a translation of $TBox$ axioms into First-Order predicate logic. We use auxiliary functions π_{x_i}, where the x_i are variables.

Definition 8 (TBox in Description Logic formalism into FO transformation)

$$\begin{aligned}
\pi(C \sqsubseteq D) &\equiv (\forall x_0)(\pi_{x_0}(C) \rightarrow \pi_{x_0}(D)) \\
\pi_{x_i}(A) &\equiv A(x_i) \\
\pi_{x_i}(\neg C) &\equiv \neg \pi_{x_i}(C) \\
\pi_{x_i}(C \sqcap D) &\equiv \pi_{x_i}(C) \wedge \pi_{x_i}(D) \\
\pi_{x_i}(C \sqcup D) &\equiv \pi_{x_i}(C) \vee \pi x_i(D) \\
\pi_{x_i}(\forall R.C) &\equiv (\forall x_{i+1})(R(x_i, x_{i+1}) \rightarrow \pi_{x_{i+1}}(C)) \\
\pi_{x_i}(\exists R.C) &\equiv (\exists x_{i+1})(R(x_i, x_{i+1}) \wedge \pi_{x_{i+1}}(C))
\end{aligned}$$

Thus, beyond the knowledge provided by the $ABox$, we keep the knowledge provided by the $TBox$. The $TBox$ contains all the relations linking classes including all the information provided by the Knowledge Base. For example, the ATC taxonomy being a tree ontology composed only of "subClassOf" relations where "B01 is a subClassOf B" corresponds to the DL notation $B01 \sqsubseteq B$ (line 1 of the definition 8). This informs us that for any instance x_0 of $B01$, x_0 is also an instance of B. Thus, the TBox must be seen as the set of all relations between classes.

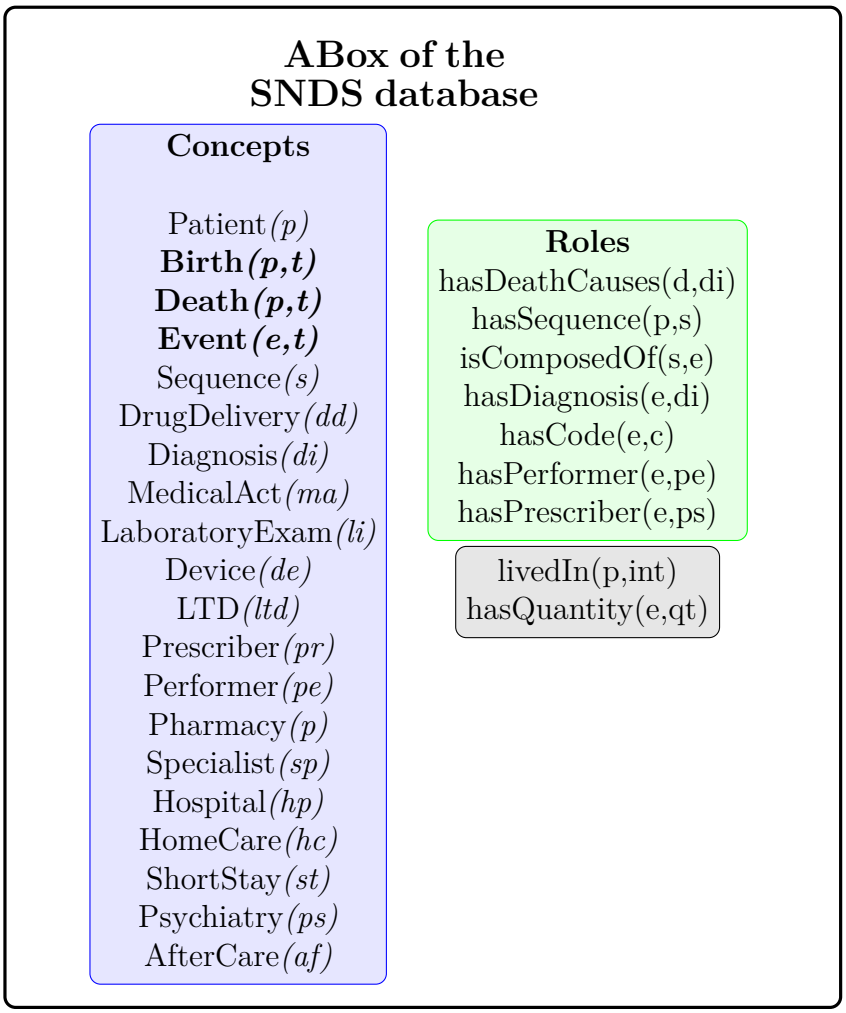


Figure 6.1 – The concepts and roles issued from the SNDS database composing the ABox

In this chapter, we have been able to define through several definitions the notion of semi-temporal ABox and non-temporal TBox. Concepts and roles have individuals defined by the *ABox* and concepts assertions, including the Knowledge Base, are defined by the *TBox* We also shown that these definitions are applicable in the case of the SNDS and we are now interested in querying such data with an approach based on Ontology-Mediated Query Answering.

6.1.2 FOL-formula to construct Ontology-Mediated Query

In the context of logical formulas, we are interested in verifying that a formula is true or false on a domain. Our domain has been defined in the previous section, it is the ontology defined with a *ABox* and *TBox*. In the following we note this domain $\mathcal{I}$. First we will define the truth relation of the FOL formula. This truth-formula defines the operators that can be used. We are interested in a FOL formulae with the assertions defined by the *ABox* in definition 6 on page 85:

$$C_k(c_i, t) \qquad D_k(d_i) \qquad R_k(r_i, r_j) \qquad t_1 < t_2 \text{ and } t_1 = t_2$$

We define the definition of the truth-relation given in the article [Art+17] as follows:
Definition 9 (truth-relation of FOL formula [Art+17]) *For any FOL-formula φ, any T-interpretation $\mathcal{I}$, and any assignments σ of elements of $\Delta^{\mathcal{I}}$ to the domain variables and t of elements of T to the temporal variables, we define the truth-relation $\mathcal{I} \models^{\mathfrak{d},\mathfrak{t}} \varphi$ by induction as follows:*

$$\begin{array}{lll}
\mathcal{I} \models^{\mathfrak{d},\mathfrak{t}} & C_k(x,t) & \textit{iff} \quad (\mathfrak{d}(x), \mathfrak{t}(t)) \in C_k^{\mathcal{I}} \\
\mathcal{I} \models^{\mathfrak{d},\mathfrak{t}} & D_k(x) & \textit{iff} \quad \mathfrak{d}(x) \in D_k^{\mathcal{I}} \\
\mathcal{I} \models^{\mathfrak{d},\mathfrak{t}} & R_k(x,y) & \textit{iff} \quad (\mathfrak{d}(x), \mathfrak{d}(y)) \in R_k^{\mathcal{I}} \\
\mathcal{I} \models^{\mathfrak{d},\mathfrak{t}} & t_1 < t_2 & \textit{iff} \quad \mathfrak{t}(t_1) < \mathfrak{t}(t_2) \\
I \models^{\mathfrak{d},\mathfrak{t}} & t_1 = t_2 & \textit{iff} \quad \mathfrak{t}(t_1) = \mathfrak{t}(t_2) \\
\mathcal{I} \models^{\mathfrak{d},\mathfrak{t}} & \forall x \varphi & \textit{iff} \quad I \models^{\mathfrak{d}',t} \varphi, \textit{ for all } \mathfrak{d}' \textit{ from } \mathfrak{d} \textit{ varying only on } x \\
\mathcal{I} \models^{\mathfrak{d},\mathfrak{t}} & \forall t \varphi & \textit{iff} \quad I \models^{\mathfrak{d},\mathfrak{t}'} \varphi, \textit{ for all } \mathfrak{t}' \textit{ from } \mathfrak{t} \textit{ varying only on } t \\
\mathcal{I} \models^{\mathfrak{d},\mathfrak{t}} & \neg \varphi & \textit{iff} \quad \mathcal{I} \not\models^{\mathfrak{d},\mathfrak{t}} \varphi \\
\mathcal{I} \models^{\mathfrak{d},\mathfrak{t}} & \varphi_1 \wedge \varphi_2 & \textit{iff} \quad \mathcal{I} \models^{\mathfrak{d},\mathfrak{t}} \varphi_1 \textit{ and } \mathcal{I} \models^{\mathfrak{d},\mathfrak{t}} \varphi_2 \\
\mathcal{I} \models^{\mathfrak{d},\mathfrak{t}} & \top & \\
\mathcal{I} \models^{\mathfrak{d},\mathfrak{t}} & \bot &
\end{array} \tag{6.1}$$

First-order connectives and quatifiers $\rightarrow, \leftrightarrow \exists$ are defined in the standard way. We say $\mathcal{I}$ satisfies an Ontology $\mathcal{O}$ or $\mathcal{I}$ is a model of $\mathcal{O}$ if $\mathcal{I} \models \varphi$ for each $\varphi \in \mathcal{O}$. Let us illustrate this definition with an example

Example 10 – FOL-formula to express the fact 'events in the database are related to patients over 18'

we can state a FOL-formula to verify that all the events in the database are related to patients over 17. Assuming the unit of time is one year:

$$
\begin{aligned}
I \models \forall t, \forall e, \forall p, \exists \boldsymbol{s} \\
&(\mathit{Sequence}(\boldsymbol{s}) \ \wedge \ \mathit{Patient}(p) \ \wedge \ \mathit{Event}(e,t) \ \wedge \\
&\ \mathit{hasSequence}(p, \boldsymbol{s}) \ \wedge \ \mathit{isComposedOf}(\boldsymbol{s}, e) \ ) \\
&\rightarrow \exists t_b (t_b + 17 < t) \ \wedge \ \mathit{Birth}(p, t_b)
\end{aligned}
$$

The translation in common language of this formula is:
"For all variables t, e and p, it exists a variable s, (line 1) such as p is an instance of patient, and e an instance of event, and t the timestamp related to the event e and s an instance of sequence (line 2) such as a patient p, has a sequence s and a sequence s has an event e (line 3) implies it exists a timestamps t_b such as the timestamps t is lower than $t_b + 17$ and a patient p is born at date t_b (line 4)".

Finally a FOL formula is composed of entities which are roles or concepts that are linked with the following operators : forall ($\forall$), not($\neg$), and($\wedge$). It is also composed of temporal variables that can be compared with the following operators : forall ($\forall$), less than ($<$) and equal ($=$).
Artale *et al.* [Art+17] recommands the query languages Datalog and SPARQL for ontology-mediated querying of point-based temporal data. D.Toman [Tom03] has shown that most of these languages are fragments of two-sorted first-order language (2-FOL) and more specifically that *two-sorted first-order logic is strictly more expressive than any fixed-dimensional first-order temporal logic with a finite set of temporal connectives.*
However, there is a difference between a truth relation and a query. In a truth-relation we verify that a formula is true, while in the query we return all the instances that verify a formula.

In our context we query temporal data. Artale *et al.* [Art+17] define FOL ontology-mediated queries where a query is a pair $Q(x,t)$ s.t.

$$Q(x,t) = \mathcal{O}, q(x,t)$$

where $\mathcal{O}$ is an ontology and $q(x,t)$ a FOL-formula with free domain variables x and free temporal variables t. We define (a,n) as a certain answer of $Q(x,t)$.
Finally, on one side, we have an ontology $\mathcal{O}$ which is a set of FOL formula ($ABox$ and $TBox$), and on the other side, we query this ontology where a query ($Q(x,t)$) is defined by a FOL-formula.
At this step, we can define any type of FOL ontology-mediated queries. More precisely, we can query any type of ontology if it is an OWL 2 QL[2](subset of OWL 2 which provides many of the main features necessary to express conceptual models such as UML class diagrams and ER diagrams) or an OWL 2 EL ontology[3] (subset of OWL 2 which provides class constructors that are sufficient to express the very large biomedical ontology SNOMED CT).

6.2 Chronicles and FOL-OMQ

FOL ontology-mediated query does not guarantee the efficiency of queries. So, we propose to limit these queries with the Chronicle formalism which will allow to define query formats where the execution is efficient. We propose to reuse the results of Chapter 4 on page 49 to extend the definition of Chronicles and thus combine the efficiency of Chronicles with the expressiveness of FOL ontology-mediated query (FOL-OMQ).

6.2.1 Chronicles with items belonging to FOL query

By following the definition 4 on page 53, a Chronicle is a pair $(\mathcal{E}, \mathcal{T})$ where $\mathcal{E}$ is an ordered set of items where items belong to taxonomy and $\mathcal{T}$ is a set of temporal constraints.
We propose to extend this notion of items to items under the form of a FOL ontology-mediated query (FOL-OMQ) designed to represent an occurrence of an event. In the Chronicle definition, we explained that the item refers to an event occurrence in a sequence. So, we propose to define a FOL-OMQ designing an event occurrence. Events

[2] `https://www.w3.org/TR/owl2-profiles/#OWL_2_QL`
[3] `https://www.w3.org/TR/owl2-profiles/#OWL_2_EL`

have several roles presented on the *ABox* Figure 6.1 on page 87 that we can exploit in the Chronicle. For example, we can specify whether a Doppler imagery takes place in or out hospital.
So, we propose to define an occurrence of an event with a FOL-OMQ. This definition is at first pretty simple:

$$q_{event}(x,t) = \ Event(x,t)$$

This query returns as answers : individuals of event x occurring at a date t.
The interest of using a FOL-OMQ is its versatility. Additional conditions can precise the events to look for. For example:

- An event that is a medical procedure (equation 6.2)
- An event that is the medical procedure EDQM001 (Doppler imaging) (equation 6.3)
- An event that is the medical procedure EDQM001 and is performed by the hospital (equation 6.4)

$$q_{event}(x,t) \ = \ MedicalAct(x,t) \tag{6.2}$$

To remind, MedicalAct is a subclasse of Event

$$\begin{aligned} q_{event}(x,t) \ &= MedicalAct(x,t) \\ &\wedge \ hasCode(x,c) \\ &\wedge \ EDQM001(c) \end{aligned} \tag{6.3}$$

$$\begin{aligned} q_{event}(x,t) \ &= MedicalAct(x,t) \\ &\wedge \ hasCode(x,c) \\ &\wedge \ EDQM001(c) \\ &\wedge \ hasPerformer(x,p) \\ &\wedge \ Hospital(p) \end{aligned} \tag{6.4}$$

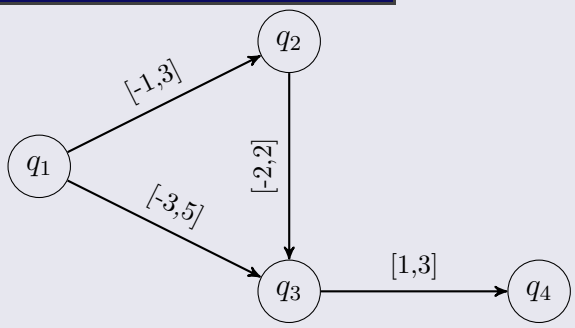
These queries all return as an answer : an individual x of event occurring at date t. They all define the occurrence of an event in a patient's sequence from an ontology $\mathcal{O}$, but they give more details. Equation 6.3 specifies the class from which the procedure code is derived and the Equation 6.4 also specifies that the performer is the hospital.
Let us define an extension of Chronicles, called onto-Chronicles on Definition 10 where items are defined as FOL-OMQ where results are individuals of event(x,t).

Definition 10 (onto-Chronicle) *An* **onto-Chronicle** $\mathscr{C}$ *is a pair* $(\mathcal{E}, \mathcal{T})$ *where*

- $\mathcal{E}$ *is an ordered* ***set of items*** $\{q_1, \cdots, q_m\}$*, where for all* $i \in \{1, \ldots, m\}$*,* q_i *is an item representing a FOL-OMQ where each query has a set of answers* (x, t) *where* (x, t) *is an indivual of the concept Event.*
- $\mathcal{T}$ *is a set of* ***temporal constraints***, i.e. *expressions of the form* $(q_j, j)[t^-, t^+](q_k, k)$ *where* $t^-, t^+ \in \mathbb{R} \cup \{+\infty, -\infty\}$

We give an example of Chronicle.

Example 11 – A Chronicle with 4 items et 4 temporal constraints



Chronicle example with 4 items (vertices) and 4 temporal constraints (edges with temporal intervals). Vertex codes give a query which evaluates the occurence of an event.

$$
\begin{aligned}
q_1(x,t) &= DrugDelivery(x,t) \\
&\wedge\ hasCode(x,c) \\
&\wedge\ A01(c) \\
q_2(x,t) &= DrugDelivery(x,t) \\
&\wedge\ hasCode(x,c) \\
&\wedge\ B01A(c)
\end{aligned}
\qquad
\begin{aligned}
q_3(x,t) &= MedicalAct(x,t) \\
&\wedge\ hasCode(x,c) \\
&\wedge\ C(c) \\
q_4(x,t) &= Event(x,t) \\
&\wedge\ hasCode(x,c) \\
&\wedge\ C(c)
\end{aligned}
$$

The Chronicle represents a drug delivery event A01 followed by a drug delivery event B01A within a delay of $[-1, 3]$ units of time (ut). The later is followed by a medical act event C within a delay of $[-2, 2]$ *ut.* In addition the delay between this event C and the event coded A01 must be in $[-3, 5]$ *ut.* Finally, C event is followed by an another event C within a delay of $[1, 3]$ *ut.*

We are now interested in defining whether a Chronicle occurs in a sequence. A sequence

is a set of assertions: Sequence(s), Event(e,t), isComposedOf(s,e), where each sequence is composed of n events. To simplify the notation and make it suitable with formalism of Chronicle, we design a sequence $\boldsymbol{s}$ as a set of individuals: $\boldsymbol{s} = \langle (e_i, t_i), \ldots, (e_j, t_j) \rangle$.

Definition 11 (Onto-Chronicle occurrence) *Let us define:*

- $\boldsymbol{s} = \langle (e_1, t_1), (e_2, t_2), \ldots, (e_n, t_n) \rangle$ *be a sequence of length n issued from the SNDS ontology* $\mathcal{O}$
- $\mathscr{C} = (\mathcal{E} = \{q_1, \ldots, q_m\}, \mathcal{T})$ *be a Chronicle of size m*
- $\mathcal{I}$ *is a model of* $\mathcal{O}$

$\mathcal{I}, \boldsymbol{s} \models q(\mathbf{e}, \mathbf{t})$ *if it exists at least one instance of* $(\mathbf{e}, \mathbf{t})$ *in* $\mathcal{I}$. *We denote* $(\mathbf{e_i}, \mathbf{t_i})$ *the set of answers of* q_i, *where* $(\mathbf{e_i}, \mathbf{t_i}) \sqsubseteq s$.
An occurrence *of* $\mathscr{C}$ *in* $\boldsymbol{s}$ *is a subsequence of* $\boldsymbol{s}$ *of length m,* $\tilde{\boldsymbol{s}}$ *such that*

$$\tilde{\boldsymbol{s}} = \langle (e_{\varepsilon_1}, t_{\varepsilon_1}), \ldots, (e_{\varepsilon_m}, t_{\varepsilon_m}) \rangle$$

where $(\varepsilon_i)_{i=1..m}$ *are indices of an event in* $\boldsymbol{s}$ *s.t.*

1. $\exists q_i \in \mathcal{E},\ \mathcal{I}, \boldsymbol{s} \models q_i(\mathbf{e_i}, \mathbf{t_i})$
2. $\exists (e_{\varepsilon_i}, t_{\varepsilon_i}) \in (e_i, t_i)$, *and* $\exists (e_{\varepsilon_j}, t_{\varepsilon_j}) \in (e_j, t_j)$ *such that* $t_{\varepsilon_j} - t_{\varepsilon_i} \in [t^-, t^+]$ *whenever* $(q_i, i)[t^-, t^+](q_j, j) \in \mathcal{T}$.

This definition is strictly more expressive than the taxo-Chronicle (definition 4 on page 53) with items related to taxonomy classes. Each item q_i is a FOL-OMQ where the answer is an instance of a timestamped event (e_k, t_k). With this new Chronicle definition, we can give as much information as desired. This formula stays decidable as long as it states to be a conjunctive query (referes to definition 9 on page 88 of the truth-relation). The decidability is not garanteed when using disjunction $\vee$, negation $\neg$ [Art+17]. In practice, avoiding recursion and negation avoids the decidability problems and improve efficiency.

However, the use of negations remains an important topic in epidemiology. Indeed, in the construction of the cohorts of interest, it will be necessary to exclude certain patterns,

such as pregnancy during the observation period, comorbidity that may distort the results, or even specify the duration of a treatment, as in the case of VTE, where the treatment must not exceed 12 months sign of another disease. It is therefore of great interest to be able to integrate negations into a phenotype. Using negation in FOL-OMQ seems to be an inadequate solution to exploit negations. We propose a solution in the following section.

6.2.2 Extension with negations

First of all, negations should be carefully handled. In the context of temporal constraints, a lot of ambiguities can appear [GB20]. To illustrate these ambiguities, we propose an imaginary Chronicle on Figure 6.2 on the next page with an item containing a negation. Intuitively this Chronicle can be read in two different ways:

- a is not followed between 1 and 2 ut by an event b, or
- a is followed between 1 and 2 ut by at least one event which is not b.

In the first case, we define an occurrence of the event a followed by a period between 1 and 2 ut after where there is no occurrences of the event b. In the second case we define an occurrence of the event a followed by a period between 1 and 2 ut where at least one event is not b.

In this section we propose to remove this ambiguity thanks to onto-Chronicle. We will describe each of the two assertions with a FOL-OMQ. To do this, we refer to the Chronicle of the Figure 6.3 on the following page. This onto-Chronicle is composed of two items q_1 and q_2 where :

$$q_1(x,t) \;=\; MedicalAct(x,t) \;\wedge\; hasCode(x,c) \;\wedge\; EDQM001(c)$$

and

$$q_2(x,t) \;=\; \neg\,(Event(x,t) \;\wedge\; hasCode(x,c) \;\wedge\; B01A(c))$$

q_2 represents the fact *"it does not exist any x and t where (x,t) are individuals of an event, or where x does not have a code, or where the code is not an instance of $B01A$."* Then, it has no sense to verify if the event defined by q_1 is followed by q_2.
So, we propose an other type of relation that would add conditions to an existing Chronicle

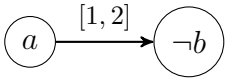


Figure 6.2 – a Chronicle with a negation

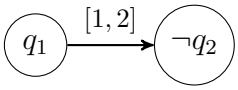


Figure 6.3 – an Onto-Chronicle with a negation

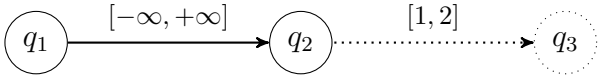


Figure 6.4 – An Onto-Chronicle where the item q_2 is followed by an item with a negation

item. In the example Figure 6.4 on the next page, we define in dotted arrow, a new type of temporal constraint.
This new type of temporal constraint specifies that during the period of time $[1, 2]$ following the event defined by q_2, there is no event defined by q_3. To formally extend the definition of Chronicle, we introduce a new set of "negative" events, $\mathcal{E}_{neg}$ and a new set of temporal constraints, $\mathcal{T}_{neg} = (q_i, i)[t^-, t^+](q_j, j)$ where $q_i \in \mathcal{E}$ and $q_j \in \mathcal{E}_{neg}$. This new temporal constraint only exists between a "basic" item to a negative item.
Definition 12 defines a Onto-neg-Chronicle with a new set of negative items and a new set of temporal constraints between a negative item and a "basic" item.

Definition 12 (Onto-neg-Chronicle) *A **Onto-neg-Chronicle** $\mathscr{C}$ is a quadruplet $(\mathcal{E}_{pos}, \mathcal{E}_{neg}, \mathcal{T}_{pos}, \mathcal{T}_{neg})$ where*

- *$\mathcal{E}_{pos}$ is an ordered **set of items** $\{q_1, \cdots, q_m\}$, where for all $i \in \{1, \ldots, m\}$, q_i is an item representing a FOL-OMQ where each query has a set of answers (x, t) where x is an instance of Event and t the timestamp related to this instance.*

- *$\mathcal{E}_{neg}$ is an ordered **set of items** $\{q_{neg_1}, \cdots, q_{neg_o}\}$, where for all $i \in \{1, \ldots, o\}$, q_i is an item representing a FOL-OMQ where each query has **no answer or** a set of answers (x, t) where x is an instance of Event and t the timestamp related to this instance.*

- *$\mathcal{T}_{pos}$ is a set of **temporal constraints**,* i.e. *expressions of the form $(q_j, j)[t^-, t^+](q_k, k)$ where $t^-, t^+ \in \mathbb{R} \cup \{+\infty, -\infty\}$ and $(q_j, j), (q_k, k) \in \mathcal{E}_{pos}$*

- *$\mathcal{T}_{neg}$ is a set of **during temporal constraints**,* i.e. *expressions of the form $(q_j, j)[t^-, t^+](q_k, k)$ where $t^-, t^+ \in \mathbb{R} \cup \{+\infty, -\infty\}$ and $(q_j, j) \in \mathcal{E}_{pos}$ and $(q_k, k) \in \mathcal{E}_{neg}$*

The Chronicle size is $m + o$ (number of items).

The graphic representation is same as usual. Items belonging to $\mathcal{E}_{neg}$ and arrows belonging to $\mathcal{T}_{neg}$ are dotted.

So, the Chronicle occurrence definition becomes

Definition 13 (Onto-neg-Chronicle occurrence) *Let* $\boldsymbol{s} = \langle (e_1, t_1), (e_2, t_2), \ldots, (e_n, t_n) \rangle$ *be a sequence of length* n *and* $\mathscr{C} = (\mathcal{E} = \{q_1, \ldots, q_m\}, \mathcal{E}_{neg} = \{q_{neg_1}, \ldots, q_{neg_m}\} \mathcal{T}_{pos}, \mathcal{T}_{neg})$ *be a Chronicle of size* m.
$\mathcal{I}, \boldsymbol{s} \models q(\mathbf{e}, \mathbf{t})$ *is true if it exists at least one instance of* $(\mathbf{e}, \mathbf{t})$ *in* $\mathcal{I}$. *We denote* (e_i, t_i) *the set of answers of* q_i.
$\mathcal{I}, \boldsymbol{s} \not\models q(\mathbf{e}, \mathbf{t})$ *is true if it does not exist any instance of* $(\mathbf{e}, \mathbf{t})$ *in* $\mathcal{I}$.
An occurrence *of* $\mathscr{C}$ *in* $\boldsymbol{s}$ *is a subsequence of* $\boldsymbol{s}$ *of length* m, $\tilde{\boldsymbol{s}}$ *such that*

$$\tilde{\boldsymbol{s}} = \langle (e_{\varepsilon_1}, t_{\varepsilon_1}), \ldots, (e_{\varepsilon_m}, t_{\varepsilon_m}) \rangle$$

where $(\varepsilon_i)_{i=1..m}$ *are indices of an event in* $\boldsymbol{s}$ *and s.t.*

1. $\forall q_i \in \mathcal{E}_{pos},\ I, \boldsymbol{s} \models q_i(\mathbf{e}, \mathbf{t})$
2. $\exists (e_{\varepsilon_i}, t_{\varepsilon_i}) \in (e_i, t_i)$, *and* $\exists (e_{\varepsilon_j}, t_{\varepsilon_j}) \in (e_j, t_j)$ *such that* $t_{\varepsilon_j} - t_{\varepsilon_i} \in [t^-, t^+]$ *whenever* $(q_i, i)[t^-, t^+](q_j, j) \in \mathcal{T}$.
3. $\forall q_{neg_l} \in \mathcal{E}_{neg},\ I, \boldsymbol{s} \not\models q_{neg_l}(\mathbf{e}, \mathbf{t})$ *or if* $I, \boldsymbol{s} \models q_{neg_l}(\mathbf{e}, \mathbf{t}),\ \forall (e_{\varepsilon_j}, t_{\varepsilon_j}) \in \tilde{\boldsymbol{s}},\ \nexists (e_{neg_l k}, t_{neg_l k}) \in (e_{neg}, t_{neg})$ *such that* $t_{\varepsilon_j} - t_{neg_k} \in [t^-, t^+]$ *whenever* $(q_j, j)[t^-, t^+](q_{neg_l}, l)$

The only difference between onto-Chronicle and onto-neg-Chronicle is a new type of arrow which finally amounts to checking the non-occurrence of an event on a period. The type of query is the same, for negative query we just make sure that there is no answer or that none of the answers check the temporal conditions. We can even see the definition as a simple supplement : we first check the occurrence of the Onto-Chronicle and then, in a second step, we check the conditions brought by the negation items.

As a conclusion, we have extended the Chronicle model with event occurrences defined by FOL-OMQ. Onto-neg-Chronicles can define conditions related to events and thus fully

exploit the information contained by the ontology of the SNDS. Moreover, this formalism let us describe and introduce the notion of negation in the Chronicles. Even if the proposed negations are in a very specific framework, we can add an option without deteriorating the efficiency.

6.2.3 HyCOR for Chronicle recognition

To apply the Chronicle recognition task, SPARQL is still a candidate to express such queries (cf 6.1.2 on page 88). However, the use of negations in SPARQL has shown to be very inefficient, in addition to the limitations already observed in the Chapter 4 on page 49. So, we propose to reuse the HyCOR tool defined in the section 4.4.3 on page 59. Indeed, the process remains the same, the difference lies in the SPARQL query which is adapted to the OWL model. The recognition algorithm stays the same with an addition of a negation checking. This SPARQL query translates the queries defined by the Chronicle. To remind, the dot in SPARQL is an 'and'($\wedge$) operator.

```
SELECT DISTINCT ?seq ?date1 ?date2 ?date3 ?codeDrug ?codeMed ?codeDiag
WHERE{
  VALUES ?codeDrug { atc:B01AB }
  VALUES ?codeMed { ccam:EDQM001 }
  GRAPH snds:kb {
    ?d rdfs:subClassOf* ?codeDrug.
    ?m rdfs:subClassOf* ?codeMed.
    ?diag rdfs:subClassOf* ?codeDiag.
    }
  GRAPH ?seq{
    #equation q1, q3, q4 and q5
    ?event1 a :DrugDelivery.
    ?event1 :hasCode ?d.
    ?event1 :hasDate ?date1.
    #equation q2
    ?event2 a :MedicalAct.
    ?event2 :hasCode ?m.
    ?event2 :hasDate ?date2.
    ?event2 :hasPerformer ?pe.
    ?pe a :hospital.
    }
  }
```

This query extract all the sequences containing the items of the Chronicle. For each sequence, we also retrieve all the events' individuals which have a code which is an individual of one of the subclasses of atc:B01AB and of ccam:EDQM001. Then HyCOR

uses the result of this query and with the algorithm 1 on page 61 find the sequences verifying the temporal constraints and the negations.

We don't have data sets that allow us to evaluate HyCOR with negations. But the complexity of the algorithm and the SPARQL query being similar,we insure to be on the same order of magnitude as the experiments conducted in Chapter 4 on page 49. On the other hand we can apply it on real data as proposed in next Chapter 7 on page 103.

6.3 Onto-neg Chronicles to represent phenotypes

We use Onto-neg Chronicles to represent a phenotype. Following the use case of VTE, we propose the pattern in Figure 6.5 on the next page. This is a Chronicle $\mathcal{C}_{VTE} = (\mathcal{E}_{pos}, \mathcal{E}_{neg}, \mathcal{T}_{pos}, \mathcal{T}_{neg})$:

- $\mathcal{E}_{pos} = \{q_1, q_2, q_3, q_4\}$
- $\mathcal{E}_{neg} = \{q_5\}$
- $\mathcal{T}_{pos} = \{q_1[-2, 2]q_2\ ,\ \ q_1[-1, 31]q_3\ ,\ \ q_3[1, 31]q_4\}$
- $\mathcal{T}_{neg} = q_1[-365, -1]q_5$

The Onto-neg Chronicle specifies:

- An event with a code which is an instance of the class of EDQM001 must occur from 2 days before the occurrence of a B01AB to 2 days after this occurrence.
- This same event B01AB is followed by another event with a code in the equivalence class of B01AB from 1 to 31 days after.
- This later occurrence is followed by another event with a code in the equivalence class of B01AB from 1 to 31 days after.

$$
\begin{aligned}
q_1(x,t) &= DrugDelivery(x,t) \ \wedge\ hasCode(x,c) \ \wedge\ B01AB(c) \\
q_2(x,t) &= MedicalAct(x,t) \ \wedge\ hasCode(x,c) \ \wedge\ EDQM001(c) \\
q_1(x,t) &= q_3(x,t) = q_4(x,t) \\
q_5(x,t) &= DrugDelivery(x,t) \ \wedge\ hasCode(x,c) \ \wedge\ B01AB(c)
\end{aligned}
$$

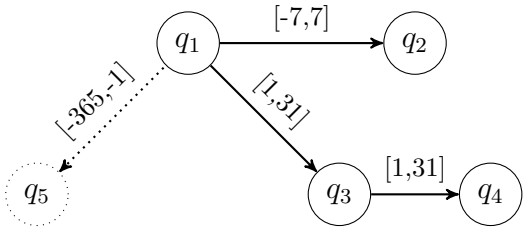


Figure 6.5 – Onto-Neg-Chronicle to represent the use case of VTE where items are FOL-OMQ. q1, q2 and q3 designe deliveries of B01AB, q2 designs a Doppler imagery and q5 a non occurence of cancer.

where B01AB is a class in the ATC taxonomy for antithrombontics, EDQM001 a class in the CCAM taxonomy for Doppler imagery.

6.4 Conclusion

In the phenotypes expressed with taxo-chronicles we were limited in the detail of the events. We could only give a constraint on the code of an event occurrence. In this chapter, we proposed to go beyond this limit by manipulating the concepts linked to the concept of Event itself. Onto-neg-chronicle allows to recognize events thanks to the code, their type and other available information.

In this chapter, we have first presented query based on a first order formula to query a timed ontology. Following these definitions, we were able to extend Chronicles to define two types of items: items representing the occurrence of an event thanks to a query based on a Frist Order Logic formula (FOL-OMQ) and items representing the non-occurrence of an event over a period of time also thanks to a FOL-OMQ. Then, we adapted the hycor tool for the recognition of these onto-neg-Chronicles and showed examples that illustrated that such a formalism was more expressive than taxo-chronicles.
This onto-neg Chronicle is based on the interrogation of an event-centric ontology, *i.e.* an ontology that contains instances/individuals of sequences and events in relation with other concepts. We have shown that this new Chronicle exploit all the information linked to the concept of events and thus offer more possibilities in the description of a phenotype. We also introduced the notion of negation to express the non-occurrence of an event over a period following the occurrence of an event.

Use taxonomies	✓
Expressing metric time constraints between different criteria	✓
Use all types of criteria (diagnosis, medical acts, hospitalization, etc.)	✓

Table 6.1 – Criteria of evaluation for expressivity of the temporal model

This approach achieves three of the objectives set out in the introduction, namely: use taxonomies, several types of events, and metric time constraints in the expression of phenotypes (listed in Table 6.1). However, no synthetic dataset is available to evaluate the tool. In next chapter, we use onto-neg-Chronicles to express VTE in a real use case and to evaluate HyCOR on a real dataset through this use case.

Chapter 7

Application on patients with venous thrombosis in the SNDS

In Chapter 6 on page 83, we previously proposed the onto-neg-Chronicles to describe phenotypes and the HyCOR tool to extract patients verifying such phenotypes from a database. This database follows a data model based on OWL ontologies capturing a maximum of information in a AHDB as presented in Chapter 5 on page 69.

The objective of this chapter is to make a proof of concept of HyCOR based on a realistic case study and real data. To do so, we have access to a geographical-based SNDS subset (the northwestern Brittany, France population) containing 80 000 individuals. We aim at extracting patients with venous thrombosis using phenotypes provided by epidemiologists. To carry out this case study, we first describe the transformation of the SQL relational database into an RDF database following the OWL model given in Chapter 5 on page 69. In a second step, we give examples of onto-neg-chronicle allowing us to describe phenotypes of a deep venous thrombo-embolism (VTE). Then in a third step, we discuss the efficiency and expressiveness of HyCOR based on this proof of concept.

7.1 SNDS transformation – From SQL database to RDF database

In Chapter 5 on page 69 we described the information from the SNDS that is useful for epidemiological purposes and we proposed a suitable OWL data model to represent this information. In the following Section 7.1.1 on the next page we will show concretely which information of the original SQL model we keep, including a description of the SQL tables of the raw database. In the next Section 7.1.2 on page 107, we will show how these same data are stored in RDF format following the OWL data model (Chapter 5 on page 69).

7.1.1 Transformation of the SNDS database

In this section, we first explain how the information is organized in raw data of the SNDS and we select information to keep in the RDF database. As a reminder, the SNDS differentiates care that takes place outside the hospital so-called *city care* (on the left on Figure 7.1 on the facing page) from care that takes place in the hospital so-called *hospital care* services (on the right on Figure 7.1 on the next page). This distinction is important as city care and hospital care are physically stored and managed differently.
Figure 7.1 on the facing page gives a schema of the tables we want to keep by highlighting the tables joined between them either by join keys (logo 🔑) or by chaining (dotted line). This schema highlights sources of data between both the city care system and hospital systems.

City care City cares (on the left, in orange on Figure 7.1 on the next page) are organized with a "star schema", there is a central *service* table (called ER_PRS_F) linked by the same join key to a set of detailed tables :

- information on establishments (ER_ETE_F),
- drug deliveries (ER_PHA_F),
- biological exams (ER_BIO_F),
- and medical procedures (ER_CAM_F).

In the central table service, we find information about the start and end date of the service and a primary key to identify the service.
The details of the patient concerned by the service are found in the Personal Information table (IR_BEN_R), then depending on the type of service, details are found in the associated tables.

For example, if the service is drug delivery, we will make a join with the Drug Delivery table (ER_PHA_F) and we will have access to the quantity and the drug code. To have access to the executing institution or to the prescriber, we join the Drug Delivery table (ER_PHA_F) with the Service table (ER_PRS_F) and with the health institution (ER_ETE_F).

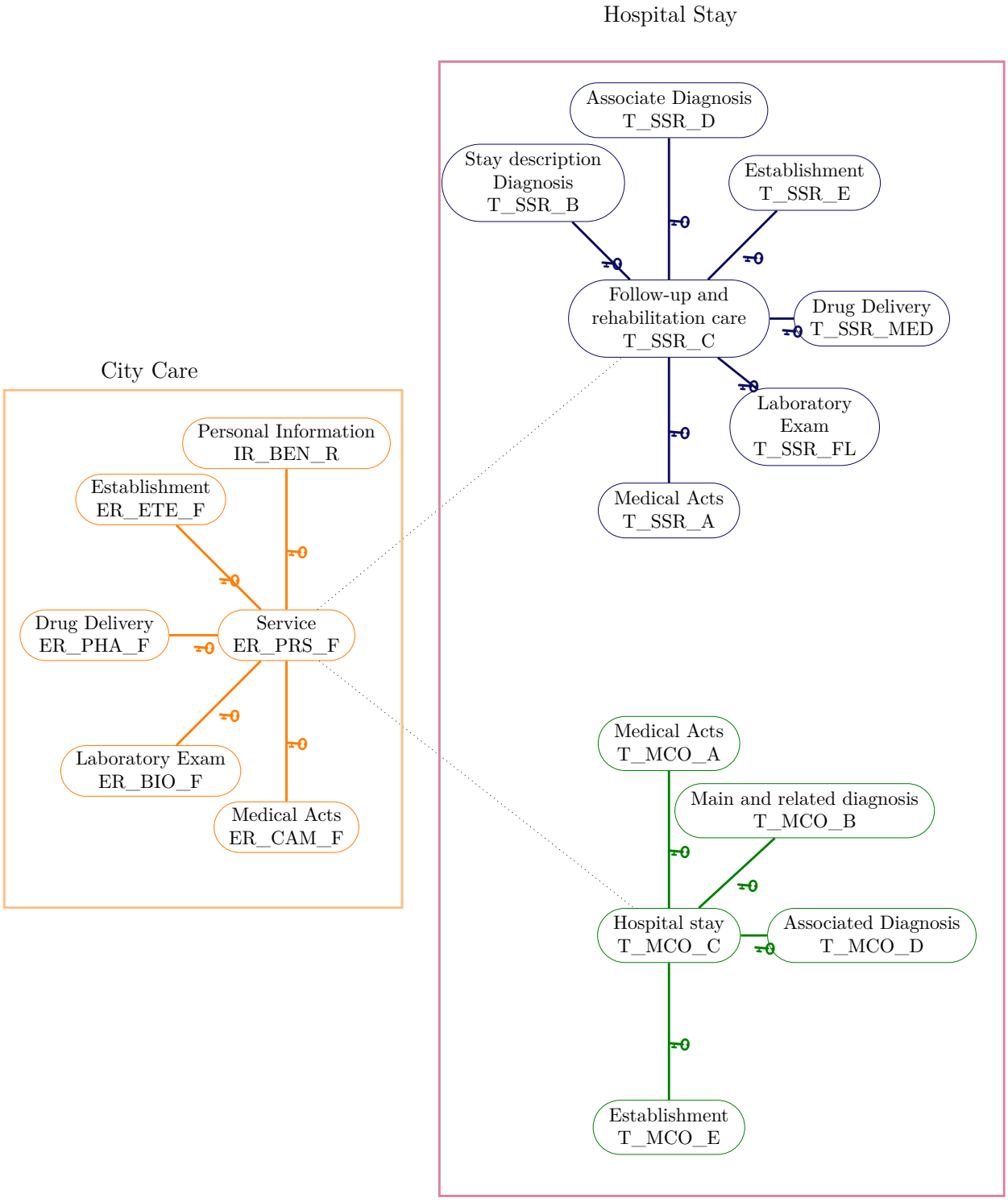


Figure 7.1 – Information extracted from the SNDS for the case study

And so to have the age of the patient, his drug deliveries and the health establishment, we join the three tables IR_BEN_R, ER_ETE_F and ER_PHA_F. To detect particular medical acts, we join the ER_CAM_F table, and so on...
We now sought to extract the same types of information in hospital care.

Hospital care To do this, we looked at three sets of hospital tables: those concerning follow-up care and rehabilitation and those concerning inpatient care in public hospitals [1]. Each of these sets follows a "star schema" : There is a central table to which detailed tables are joined.
The hospital cares (on the right, boxed in purple on Figure 7.1 on the preceding page) are divided into several sets, we are interested in two of them : the short stay hospital care (in green on Figure 7.1 on the previous page) and the Follow-up and Rehabilitation care (in blue on Figure 7.1 on the preceding page). The set of Short Stay Hospital care is composed of a central table containing a main diagnosis and information about the stay. In order to recontextualize these data in French hospital care, these cares refer to reimbursements for short-stay hospitalizations. A hospitalization stay is composed of the main diagnosis for the patient's admission (the first diagnosis, called the principal diagnosis) and also the principal pathology linked to this hospitalization (the second diagnosis, called the relational diagnosis), and then a set of diagnoses associated with the hospitalization are also entered.
Each stay is detailed in the Medical Act table (T_MCO_A) which contains the medical acts carried out during the stay. Stays are detailed with the Main and related Diagnosis table (T_MCO_B), the Associated Diagnosis table (T_MCO_D) and the Establishment table (T_MCO_E) containing information on the health care institution in charge of the stay. These tables are linked with a primary key based on an identification of the stay.
The Rehabilitation care set contains a central table with a main diagnosis and information on the stay. In order to recontextualize these data in the context of French hospital care, this care refers to reimbursements for long-stay hospitalizations. For example, an elderly person treated for a fracture after a fall will initially go to a short stay, but will not be able to go home directly. She will need rehabilitation care and will then be admitted to a rehabilitation stay.
Each rehabilitation hospitalization contains the main diagnosis at the time of the patient's

[1] Note that the tables concerning psychiatric hospital care (T_RIP) and the tables concerning hospitalizations at home (T_HAD) are not presented here, but they follow the same principle as T_SSR and T_MCO.

arrival (first diagnosis), then a reason for hospitalization (second diagnosis), and might contain main pathologies linked to this hospitalization (third diagnosis).
It also contains all the associated co-morbidities (fourth diagnosis) which may influence the management of the patient, in particular, the duration and quantity of care necessary for the patient's recovery.
For example, a patient may be initially entered into hospital for a cardiac decompensation (diagnosis 1) linked to a pulmonary embolism (diagnosis 2) which will justify hospitalization. However, these events may be due to a pathology, for example, lung cancer (diagnosis 3) and require different management depending on whether the patient has comorbidities: obesity, diabetes, asthma, etc (diagnosis 4).
Finally, the plurality of hospital tables brings several difficulties, the first one: there is no central table to link these sets, so to make a query similar to the one described in city care, we must repeat it for the four types of care (short-stay (MCO), long stay (SSR), psychiatric (PSY) and home hospitalization (HAD)).
Moreover, if we want to join it to the patients of the table IR_BEN_R there is no foreign key but a common patient identifier between the hospitalization tables and IR_BEN_R. In addition to not guaranteeing the consistency between identifiers assumed to be identical, it does not allow the database system management to optimize the processing of the joins. To find a patient who has had a short stay hospitalization and then a long stay, we will have to "chain" our two tables on patients' identifiers. We specify that patient identifier is not guaranteed to be unique. As the tables have a primary key identifying a stay, a patient identifier appears as many times as he has stayed. Since the join keys are independent of the patient (they identify a service), they quickly become cumbersome (lots of combinations). The OWL patient-centric data model and the HyCOR tool enable to not have to manage these joins, nor even to know all the tables of the database, but simply the concepts given in the OWL model.

7.1.2 From the SQL relational database to the RDF graph database

In Chapter 5 on page 69 we were interested in transforming the relation database of the SNDS into RDF data following a data model defined with OWL.
The advantage of such a representation is the patient-centric view where each patient has a sequence of care events instead of a service-centric view where each service is detailed

in several tables. Thus, rather than linking services and care as originally proposed in the SNDS, this new representation facilitates the extraction of patients verifying a phenotype. The events are then defined through three important elements: start/end dates, a code linked to a type and a taxonomy, an executor and a prescriber. Thus we can easily identify patients who had a medical act "Doppler imagery". We will talk about an event of type "Medical act" with the code EDQM001 of the CCAM taxonomy and we no longer need to join the different hospital stays with the city care.
However, the difference between city care and hospital care can have clinical importance and enrich the description of a phenotype. Onto-chronicles keep the possibility to differentiate both by detailing the executor of the event: this can be a hospital (including all types of hospitals) or a particular hospitalization (short stay/ long stay). We are approaching the strength of semantic web data: combining data semantics (being interested in the meaning of the data) with efficient query tools.

In this section, we present the tool we developed to transform the data detailed above from an SQL database to an RDF database. This tool follows several steps.
Step 1: We query all the patient identifiers from the Personal Information table (IR_BEN_R) with personal information such as place of residence, date of birth and sex. We do processing to recover the list of personal information for each patient identifier, which we associate with a patient graph. Thus we obtain a set of patient graphs. For example:

```
GRAPH patient1{
    :patient1 :hasBirthDate "2000-01-01"^xsd:integer.
    :patient1 :hasBirthDate "1666-01-01"^^xsd:integer.
    :patient1 :hasLocation "1003".
    :patient :hasGender "2".
}
```

Attention, the primary key of this table is the service, there are many duplicates and some information can appear in double like the dates of birth in this example. Even if absurd in theory, it is normal in practice. Some of this information has been filled in by humans and can lead to inconsistent data. It is not managed in this thesis but it can be managed by the query or at least be aware of it during the analyses.
Step 2: We query the services of city care : medical acts, deliveries of drugs, biological exams and information on the institutions. For each service, we retrieve the start date of the service, the end date, the patient concerned, the service code and the taxonomy

used (ATC for drugs, ICD-10 for diagnosis, CCAM for medical procedures, etc. . .), in the case of drug deliveries, we also retrieve the number of boxes delivered. We thus fill in the information of each patient, service by service:

```
GRAPH patient1{
    ...
    Sequence1 :isComposedOf :patient1evt1.
    :patient1evt1 a owl:NamedIndividual.
    :patient1evt1 a :MedicalAct.
    :patient1evt1 :hasCode ccam:EDQM001.
    :patient1evt1 :hasPrescriber spe:10.

    :isComposedOf :patient1evt2.
    :patient1evt2 a owl:NamedIndividual.
    :patient1evt2 a :DrugDelivery.
    :patient1evt2 :hasCode cip:3400932298617.
    :patient1evt2 :hasPrescriber spe:10
}
```

Note that the bar code of the prescribed boxes, called CIP code, is used for the delivery of drugs. The ATC taxonomy contains the correspondence between barcode and ATC code.

Step 3: We retrieve the same information but from the hospital where we add the notion of associated diagnosis, related diagnosis and associated diagnosis (explained in the previous section)

```
    GRAPH :patient1{
        ...
         Sequence1 :isComposedOf :patient1evt3.
        :patient1evt3 a owl:NamedIndividual.
        :patient1evt3 a :Diagnosis.
        :patient1evt3 :hasCodeRelated icd:I.
        :patient1evt3 :hasCodeAssociated icd:I801 .
        :patient1evt3 :hasCodeAssociated icd:C793 .
        :patient1evt3 :hasCodeAssociated icd:N185 .
        :patient1evt3 :hasPerformer shortstay:1004269.
    }
```

Step 4 : We create/download graphs of the following ontologies:

- The OWL ontology created in Chapter 5 on page 69 brings us the knowledge of the data model and that allows us to make queries more easily (for example find all the hospital stays that are defined as a long stay or short stay in the ontology)
- ATC taxonomy (including bar codes correspondence)

- CCAM taxonomy

- ICD-10 taxonomy

At this step, we have a transformed database in RDF format with as many RDF graphs as patients plus a knowledge base containing several taxonomies.

Step 5 : To deploy this database we have privileged the apache-jena-fuseki[2] tool, which loads the RDF data (in a format called TDB[3]) and we query them with the SPARQL query language.
Thus we have deployed a database containing about 81 636 patients.
This transformation is a one-time task, but it is time-consuming because access to the real data does not allow us to have the right development conditions as on a personal computer. Due to practical storage limitations, this dataset contains a small number of patients. But the results provided in Chapter 4 on page 49 show that the execution time of HyCOR increases linearly with the size of the sample.
This RDF database contains 90 701 172 triples including 10 210 094 events in the set of patients with 125 events per patient in average, 1 in minimum, 3 191 in maximum. The knowledge Base contains 110 609 triples including:

- three taxonomies, ATC (including CIP codes), CCAM and ICD-10,

- the SNDS ontology developed with *Protégé*

- the taxonomies created by the epidemiologists including the list of diagnoses related to deep and the list of medical procedures related to VTE

Thanks to this new data format we are now able to use HyCOR data with examples of phenotypes to find patients with VTE.

[2] `https://jena.apache.org/`
[3] `https://jena.apache.org/documentation/tdb/`

7.2 HyCOR to find patients with deep thrombo-embolism

7.2.1 Express a phenotype with a onto-neg-chronicle

We now apply HyCOR on a real case study. We will first explain the building of a phenotype and then we execute HyCOR to find patients verifying the phenotype. This use case is applied to the real RDF dataset created in the previous section.
Throughout this thesis, we followed the case study of a phenotype designing venous thrombosis and we identified the following two phenotypes to detect patients with VTE from SNDS data:

1. A medical act (for example Doppler or CT scan) prior to anticoagulant deliveries after/before a week and delivery last a minimum of 3 months and a maximum of 12 months. Each delivery is separated by a month.

2. A diagnosis "Pulmonary Embolism" during hospitalization.

In the light of the onto-neg-chronicle developed in Chapter 6 on page 83, we can detail the events of these three phenotypes, two describing the diagnosis established with the medical procedures and one verifying if a diagnosis has been filled.

Phenotype 1 In clinical practice facing suspicion of VTE physicians first prescribe anticoagulants and then confirm or not the diagnosis through specific medical acts. Epidemiologists identified 36 possible CCAM codes for these acts. If the suspicion is confirmed, the act is preceded or followed by anticoagulant initiation within a time window of 7 days and anticoagulant deliveries continue for 3 to 12 months maximum. We keep in mind that PE suspicion leads to hospitalization during which medical acts to confirm the diagnosis are performed and then anticoagulant delivery is observed only after the patient comes back home. Epidemiologists identified the B01A ATC class which is the mother class of 270 anticoagulants' boxes.
We propose the onto-neg-chronicle of Figure 7.2 on the next page to represent this phenotype. This chronicle is composed of three vertices of anticoagulant B01A (vertices 1, 2 and 3). The code given is issued of the ATC taxonomy (in grey). The temporal constraints impose at least 3 successive deliveries of anticoagulants where the first one is separated

by a maximum of one year from the third one. The chronicle is also composed of a medical procedure with the name thrombosis from the CCAM taxonomy. This code has been created by our on, it is the mother class of the 36 CCAM codes identified. Vertice 5 is a negative one. It ensures that there is no treatment beyond one year after the first delivery. Continued treatment beyond a year could be indicative of another disease.

Phenotype 2 In this phenotype, we are interested in patients whose hospitalization continues in rehability care. It should be remembered that there is no monitoring of drug deliveries in the hospital (only of certain expensive drugs), and so it is not possible to detect anticoagulant treatment during a rehabilitation stay. We only have information on drugs delivered in city care.
So, we propose Figure 7.3 on the facing page a chronicle composed of a medical procedure with the name thrombosis from the CCAM taxonomy followed in three months by a rehabilitation stay.
Those empty windows with medical events hidden during hospitalizations in the SNDS have been studied and detailed by A.Palmaro [Pal17]. She evaluates the impact of these "empty windows" on epidemiological studies.

Phenotype 3 These phenotypes only verify if a diagnosis is filled in rather than a medical procedure. Epidemiologists have identified four ICD-10 codes for pulmonary embolism and phlebitis. However, in medical practice this type of diagnosis is rarely recorded in the SNDS.
We will now try to find the patients verifying these phenotypes from the RDF database generated in Section 7.1.2 on page 107.

7.2.2 HyCOR to execute phenotypes

In this section we propose to execute the three phenotypes one after the other using HyCOR. The union of the patients found for each of the phenotypes constitutes the cohort of patients with deep vein thrombosis.
Table 7.1 on the following page sums up the results. In phenotype 1, which is the most general, we find 106 patients in 26,8 seconds. Removing the negation, add 56 patients which might be treated for another reason than a VTE. The execution time is therefore similar. By slightly increasing the interval between the anticoagulant intake and the thrombosis detection procedure from 7 days to 10 days, there are 194 patients *i.e.* 88

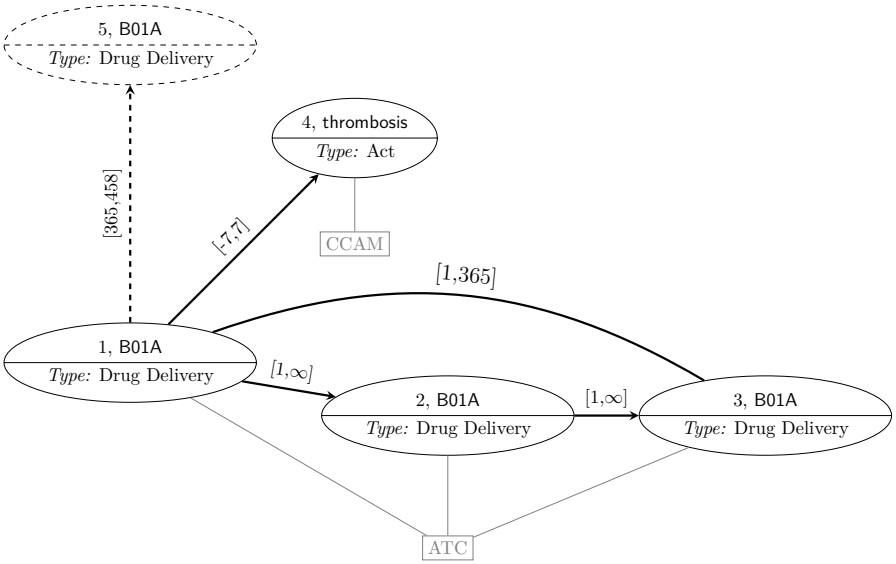


Figure 7.2 – Phenotype 1 : venous thrombosis in the SNDS based on an anticoagulant treatment and a medical procedure

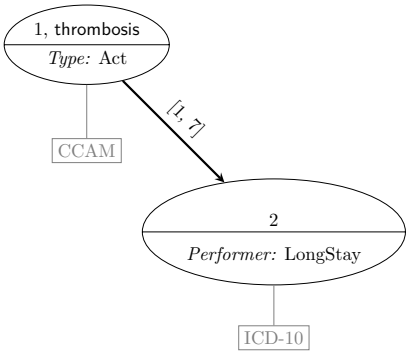


Figure 7.3 – Phenotype 2 : venous thrombosis in the SNDS based on a medical procedure with after-care hospitalization which hides the anticoagulant treatment

	number of patients found	execution time
Phenotype 1		
interval(1,4) = [-7,7]	106	26,8s
interval(1,4) = [-10,10]	194	23,2s
without negation	162	23,3s
Phenotype 2		
interval(1,2) = [1,7]	118	1,5s
interval(1,2) = [1,93]	786	1,5s
Phenotype 3: Diagnosis only		
	534	1s

Table 7.1 – Execution of HyCOR on a subset of SNDS data with 3 phenotypes designing VTE

more patients.
Phenotype 2 extracts 118 patients in a second for patients entering in hospital for a long stay in the week after the medical acts of VTE. Expanding to 3 months, we extract 786 patients, probably not all concerned by VTE.
Finally, phenotype 3 just extracts patients where a diagnosis of VTE has been filled and extracts 534 patients in a second.

These numbers raise questions about the inclusion of long-stay patients and perhaps look for other medical events indicative of VTE. The negations and interval choices seem to be a strong criterion on these phenotypes and may lead to reflection on the choice of patients to include. It is interesting to be able to quickly try different time intervals and to evaluate in real conditions which one is the most adequate. This reflection can be fast thanks to the 30s execution time offered by HyCOR.

7.3 Conclusion

Phenotype expression and execution on real data is a mandatory step for epidemiologists working on Health Administrative Databases. However, the execution of complex phenotypes, including ontological criteria and temporal constraints is a time-consuming task. In this chapter, we have shown that HyCOR facilitates this task.
In the first step, we transform the raw data following a relational schema into RDF data following a graph model. We obtain new data on which we can exploit the tools of the

Semantic Web. Moreover, it allows us to unify the data through the concepts described by the SNDS ontology presented in Chapter 5 on page 69. The developed code is available on the following deposit : `https://gitlab.inria.fr/jbakalar/snds_sqltordf`.
In the second step, we propose three phenotypes to describe VTE. The first phenotype based on the presence of a specific medical procedure and an anti-coagulant treatment shows that 106 patients are found. The second phenotype based on the presence of a specific medical act followed by a long hospital stay finds 118 patients. The last phenotype contains only patients with a diagnosis. Each of these phenotypes is executing in less than 30 seconds, proving its efficiency on real data.
Finally, we can conclude that HyCOR is a tool of interest for the construction of cohorts from phenotypes expressed with onto-neg-chronicles. They are quite expressive, thanks to the possibility to detail the events, and to use ontologies and temporal constraints. Moreover, it enhances epidemiologists to perform a large number of tests thanks to the speed of execution.

Conclusion

Epidemiology aims to study the health status of populations characterized by medical, social or economic criteria. The use of Administrative Health Databases is an asset for these studies, since data are readily available and cover a large population. However, the selection of specific populations in these databases is complex due to the lack of clinical information and diagnoses. Since these information are not available, epidemiologists identify characteristic care pathways - called phenotypes - of the population they wish to study. The description of phenotypes includes specific temporal constraints and knowledge of specific medical procedures and services. The knowledge usable in phenotypes necessarily depends on the information available in the data. As an example, one cannot search for deep venous thrombosis in the French National Database System as the diagnosis is not systematically recorded in data. Instead, patients likely to have deep venous thrombo-embolism are identified by the following phenotype : patients who have been treated with anticoagulants for 3 to 12 months maximum and whose first delivery is followed or preceded within a week by a Doppler or CT scan.
We have shown current tools used in Administrative Health Databases are not suitable for querying these data. To address this problem, this thesis proposes to query health data with a two-sided approach : one side formally representing the data to facilitate the phenotype requirements and the other side querying the data from a phenotype model.

In a first step, this thesis shows that the Semantic Web is adapted to represent health administrative data. It allows to define data models through ontologies and to link them to international ontologies. This thesis defines classes/concepts used in pharmacoepidemiology and existing in international ontologies such as drug codes with the ATC ontology[4] and diagnoses with ICD-10[5] and SNOMED-CT[6].
In addition, this thesis proposes a patient-centered representation with a set of patients where each patient has a sequence containing a set of dated events defined by a type (drug deliveries, medical acts, biologies, etc...), an executor, a prescriber and a code linked to

[4] https://bioportal.bioontology.org/ontologies/ATC
[5] https://bioportal.bioontology.org/ontologies/ICD10
[6] https://bioportal.bioontology.org/ontologies/SNOMEDCT

one or more ontologies (for example, the bar code of a drug box is linked to a code of the international taxonomy ATC).
By choosing the Semantic Web with a graph representation of the data, we allow the intuitive use of the data while integrating medical knowledge (international ontologies or ontologies created by the user). Thanks to OWL ontologies, we can also standardize data while managing data integrity around the notion of concept, links between concepts and properties proposed by OWL. However, the transition from SQL data to RDF data and the creation of a global OWL ontology leads to a complete reconversion of the database. In this thesis, we moved from a SQL format with an economic focus (health insurance reimbursement) to a data format with an epidemiological focus. Even if this implies duplicating information, this thesis has shown that searching information for epidemiology studies is easier with the Semantic Web. In the long term, the use of these data by the pairs will be a source of proposals for improving the SNDS data model proposed in this thesis.

In a second step, this thesis proposes to represent phenotypes with Chronicles. Chronicles define occurrences of event linked by temporal constraints. They are visualized as a constraint graph where events are atomic vertices and temporal constraints are arrows linking vertices. This thesis enhances the expressivity of the Chronicle model by extending the notion of atomic event (binary value : true or false occurrence) into the notion of class linked to ontologies. We call this enrich model "taxo-Chronicles". For example, a taxo-Chronicle defines an event which is an anticoagulant coded B01A in the ATC taxonomy instead of listing all the anticoagulant bar codes. We kept the arrows assigned to temporal intervals (for example, a Doppler procedure is followed/preceded in 7 days by a B01A drug).

In a third step, this thesis proposes an even more expressive model called "Onto-neg-Chronicles", where vertices are queries based on a First Order Formula. These logical formulas represent the occurrence of an event or the absence of an occurrence of an event over a period. The use of a logical formula is necessary to guarantee the efficiency of the queries, in particular when handling negations. These logical formula use predicates issued of the data model.

Last, we developed the HyCOR tool extracting the patients verifying the Onto-neg-Chronicles. This tool is hybrid, at first it queries with SPARQL - the Semantic Web query tool - a database following the proposed model. Using SPARQL enables to exploit all the solutions proposed by the Semantic Web community for the management of ontologies where it is shown to be very efficient. On the other hand, SPARQL shows to not be efficient on negations and temporal constraints defined in Chronicles. Thus, HyCOR retrieves the results of SPARQL queries to check a posteriori the temporal constraints thanks to an efficient algorithm. HyCOR has been evaluated on synthetic data for taxo-Chronicles and shows excellent results : execution time is less than 30 seconds for 20 000 sequences and 200 events per sequence to find 2000 patients verifying a phenotype composed of 15 vertices.
The choice to use Chronicles to select cohorts instead of building a hand-made query for each selection allows the intuitive construction of a phenotype where the concepts used are those of the OWL database model. The Chronicle recognition algorithm is a generic method formalizing the task of recognizing a Chronicle by an efficient algorithm. Chronicles currently allow the succinct use of negation and fixed time events (dates, no periods). We could consider improving the formalism by adding the notion of quantity, or even cumulative quantity (number of boxes of anticoagulants delivered over a year). Another important limitation is the non-integration of algorithmic solutions to manage uncertainties or small deltas in temporal constraints. For example, if we look for two anticoagulants separated by 30 days, a patient who had an anticoagulant on January 1 and a second one on February 1 is not selected. However, HyCOR is a fast and data manager independent solution. Thus, an epidemiologist can manually choose his time intervals between events by trial and error. This was impossible before as a small modification of the query requires a data manager and the cohort selection algorithms can run for several hours.

Finally, this thesis proposes a case study based on a subset of the French Health Administrative Database, called Système National des Données de Santé, containing a general population where an epidemiologist defines phenotypes to find patients with deep venous thrombosis. We transform the original model of the French Administrative Health Database, which is a SQL relational model, into a Semantic Web model (called OWL model) and we generate data in RDF (data format of the Semantic Web). We then evaluate the Onto-neg-Chronicles and obtained excellent results : execution time is less than 30

seconds for 80 000 patients and 50 events on average per patients' sequence. This last evaluation was able to highlight the expressiveness of Onto-neg-Chronicles by implementing the case study of deep venous thrombosis.
By proposing Chronicles to represent phenotypes we offer a formalization of the notion of phenotype. Instead of using phrasal descriptions or decisional schema, this thesis proposes a formal model where each concept used in the model is an existing concept in the database. In the literature, we talk about phenotypes and in particular about phenotype shares (redSiam[7]) but little about the computational difficulties linked to the use of temporal criteria in these phenotypes. This thesis offers a very expressive format and a way to unify phenotypes across different databases around the world. There is a practical limit where the use of HyCOR by the largest number of people today requires an industrialization phase, i.e. developing an accessible API, easy to install and an adapted visualization. One could also choose to use HyCOR as a back-end to an already existing visualization. Such an application will require scaling up to even larger datasets, for which the results of the experiments performed in this manuscript are encouraging. However, the code allowing the transformation of SQL data from SNDS to RDF data and the HyCOR tool are available on GitHub (applicable on any OWL-RDF database) and are ready to be implemented on secured servers as it was done during the thesis.

To conclude, the results of this thesis are satisfactory both at the expressiveness level on the expression of phenotypes and at the efficiency level with HyCOR. This proposal is applied to real data from the SNDS with an epidemiological problem concerning deep venous thromboembolism.

The first perspective for this project is its use on several databases with several use cases. There is little collaboration between European countries, while Germany and England have also large-scale Administrative Health Databases. Some unification solutions are emerging, such as the proposal of a universal data model OMOP, or the implementation of international ontologies like ATC, ICD-10 and SNOMED-CT .
I think the Semantic Web approach with the provision of ontologies, preferably free of access, is more adapted to health data concentrating a large variety of concepts, rather than rigid relational models. Efforts should continue to focus on the development of free common models that will allow comparison and evaluation of phenotypes developed by

[7] `https://www.redsiam.fr/`

epidemiologists across multiple data sets.
In a longer term perspective it is interesting to manage the presence of non-observable periods in the databases. In the case of the French National Health Database Systems, patients entering in specialized institutes do not have any details on their medical acts for more or less long periods. We would like to be able to consider these unobservable periods in HyCOR. This phenomenon is studied in the thesis of A. Palmaro [Pal17], who shows the impact of these empty periods by comparing the same studies conducted on the English and French databases.

BIBLIOGRAPHY

[AH11] Dean Allemang and James Hendler, *Semantic web for the working ontologist: effective modeling in RDFS and OWL*, Elsevier, 2011.

[Ahl20] Anders Ahlbom, "Epidemiology is about disease in populations", *in*: *European Journal of Epidemiology* 35.*12* (2020), pp. 1111–1113.

[All83] James F Allen, "Maintaining knowledge about temporal intervals", *in*: *Communications of the ACM* 26.*11* (1983), pp. 832–843.

[Ani+11a] Darko Anicic et al., "EP-SPARQL: a unified language for event processing and stream reasoning", *in*: *Proc. of Int. Conf. on World Wide Web (WWW)*, 2011, pp. 635–644.

[Ani+11b] Darko Anicic et al., "ETALIS: Rule-based reasoning in event processing", *in*: *Proc. of Reasoning in event-based distributed systems*, 2011, pp. 99–124.

[Art+17] Alessandro Artale et al., "Ontology-mediated query answering over temporal data: A survey", *in*: (2017).

[Baa+03] Franz Baader et al., *The description logic handbook: Theory, implementation and applications*, Cambridge university press, 2003.

[Baa+17] Franz Baader et al., *Introduction to description logic*, Cambridge University Press, 2017.

[Baa14] Franz Baader, "Ontology-Based Monitoring of Dynamic Systems.", *in*: *KR*, 2014.

[BC07] Francesco Basile and Pasquale Chiacchio, "On the implementation of supervised control of discrete event systems", *in*: *IEEE Transactions on Control Systems Technology* 15.*4* (2007), pp. 725–739.

[BCD09] Francesco Basile, Pasquale Chiacchio, and Gianmaria De Tommasi, "An efficient approach for online diagnosis of discrete event systems", *in*: *IEEE Transactions on Automatic Control* 54.*4* (2009), pp. 748–759.

[BCM05] Patricia Bouyer, Fabrice Chevalier, and Nicolas Markey, "On the expressiveness of TPTL and MTL", *in*: *International Conference on Foundations of Software Technology and Theoretical Computer Science*, Springer, 2005, pp. 432–443.

[Bez+17] Julien Bezin et al., "The national healthcare system claims databases in France, SNIIRAM and EGB: Powerful tools for pharmacoepidemiology", *in*: *Pharmacoepidemiology and drug safety* 26.*8* (2017), pp. 954–962.

[BGL12] Franz Baader, Silvio Ghilardi, and Carsten Lutz, "LTL over description logic axioms", *in*: *ACM Transactions on Computational Logic (TOCL)* 13.*3* (2012), pp. 1–32.

[Bie16] Meghyn Bienvenu, "Ontology-mediated query answering: harnessing knowledge to get more from data", *in*: *Proc. of Int. Join Conf. on Artificial Intelligence (IJCAI)*, 2016, pp. 4058–4061.

[BK08] Christel Baier and Joost-Pieter Katoen, *Principles of model checking*, MIT press, 2008.

[BN18] Pierfrancesco Bellini and Paolo Nesi, "Performance assessment of RDF graph databases for smart city services", *in*: *Journal of Visual Languages & Computing* 45 (2018), pp. 24–38.

[Böh+17] Michael H Böhlen et al., "Temporal data management–an overview", *in*: *European Business Intelligence and Big Data Summer School*, Springer, 2017, pp. 51–83.

[Bra+18] Freddie Bray et al., "Global cancer statistics 2018: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries", *in*: *CA: a cancer journal for clinicians* 68.*6* (2018), pp. 394–424.

[BRV04] Bernard Berthomieu*, P-O Ribet, and François Vernadat, "The tool TINA–construction of abstract state spaces for Petri nets and time Petri nets", *in*: *International journal of production research* 42.*14* (2004), pp. 2741–2756.

[BS12] Rene Boel and Geert Stremersch, *Discrete event systems: analysis and control*, vol. 569, Springer Science & Business Media, 2012.

[BT15] Stefan Borgwardt and Veronika Thost, "Temporal query answering in the description logic EL", *in*: *Twenty-Fourth International Joint Conference on Artificial Intelligence*, 2015.

[BZ12] Asma Ben Abacha and Pierre Zweigenbaum, "Medical question answering: translating medical questions into sparql queries", *in*: *Proceedings of the 2nd ACM SIGHIT international health informatics symposium*, 2012, pp. 41–50.

[Cal+05] Diego Calvanese et al., "DL-Lite: Tractable description logics for ontologies", *in*: *AAAI*, vol. 5, 2005, pp. 602–607.

[CB17] Werner Ceusters and Jonathan Blaisure, "A Realism-Based View on Counts in OMOP's Common Data Model.", *in*: *pHealth*, 2017, pp. 55–62.

[Cla97] Edmund M Clarke, "Model checking", *in*: *International Conference on Foundations of Software Technology and Theoretical Computer Science*, Springer, 1997, pp. 54–56.

[Con] cohort Constances, *scientific productions*, URL: `https://www.constances.fr/productions-scientifiques` (visited on 02/01/2020).

[COP00] Pedro Cabalar, Ramón P Otero, and Silvia Gómez Pose, "Temporal constraint networks in action", *in*: *Proc. of European Conf. on Artificial Intelligence (ECAI)*, 2000, pp. 543–547.

[Cou+11] E Couto et al., "Mediterranean dietary pattern and cancer risk in the EPIC cohort", *in*: *British journal of cancer* 104.*9* (2011), pp. 1493–1499.

[CSK11] Pinaki Chakraborty, Prem Chandra Saxena, and Chittaranjan Padmanabha Katti, "Fifty years of automata simulation: a review", *in*: *acm inroads* 2.*4* (2011), pp. 59–70.

[CYC13] Huajun Chen, Tong Yu, and Jake Y Chen, "Semantic web meets integrative biology: a survey", *in*: *Briefings in bioinformatics* 14.*1* (2013), pp. 109–125.

[Dan+19] Mark D Danese et al., "The generalized data model for clinical research", *in*: *BMC medical informatics and decision making* 19.*1* (2019), pp. 1–13.

[Dau+17] Yann Dauxais et al., "Discriminant chronicles mining", *in*: *Proc. of Conf. on Artificial Intelligence in Medicine in Europe (AIME)*, 2017, pp. 234–244.

[De 96] Bart De Schutter, *Max-algebraic system theory for discrete event systems*, PhD thesis, Faculty of Applied Sciences, KU Leuven, Leuven, Belgium, 1996.

[DEL+15] Denis DELAMARREabc et al., "Semantic integration of medication data into the EHOP clinical data warehouse", *in*: (2015).

[DL07] Christophe Dousson and Pierre Le Maigat, “Chronicle Recognition Improvement Using Temporal Focusing and Hierarchization.”, *in*: *Proc. of Int. Join Conf. on Artificial Intelligence (IJCAI)*, 2007, pp. 324–329.

[DL96] Giuseppe De Giacomo and Maurizio Lenzerini, “TBox and ABox reasoning in expressive description logics.”, *in*: *KR* 96.*316-327* (1996), p. 10.

[DP04] Alexandre Duret-Lutz and Denis Poitrenaud, “SPOT: an extensible model checking library using transition-based generalized Bu/spl uml/chi automata”, *in*: *The IEEE Computer Society's 12th Annual International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunications Systems, 2004.(MASCOTS 2004). Proceedings.* IEEE, 2004, pp. 76–83.

[Dru+19] Brett Drury et al., “A survey of semantic web technology for agriculture”, *in*: *Information Processing in Agriculture* 6.*4* (2019), pp. 487–501.

[ElS+18] Shaker El-Sappagh et al., “SNOMED CT standard ontology based on the ontology for general medical science”, *in*: *BMC medical informatics and decision making* 18.*1* (2018), pp. 1–19.

[ENC] ENCePP, *ENCePP Resource Database - Inventory of Patient Registries*, URL: `https://www.ema.europa.eu/en/documents/regulatory-procedural-guideline/encepp-resource-database-inventory-patient-registries_en.pdf` (visited on 02/01/2020).

[GB20] Thomas Guyet and Philippe Besnard, “Semantics of negative sequential patterns”, *in*: *arXiv preprint arXiv:2002.06920* (2020).

[GDL+14] Richard E Gliklich, Nancy A Dreyer, Michelle B Leavy, et al., “Registries for evaluating patient outcomes: a user's guide”, *in*: (2014).

[GHV06] Claudio Gutierrez, Carlos A Hurtado, and Alejandro Vaisman, “Introducing time into RDF”, *in*: *IEEE Transactions on Knowledge and Data Engineering* 19.*2* (2006), pp. 207–218.

[Gia+17] Nikos Giatrakos et al., “Complex Event Recognition in the Big Data Era”, *in*: *Proc. VLDB Endow.* Vol. 10, 2017, pp. 1996–1999.

[Gua] The Guardian, *Trial over weight-loss pill behind 'up to 2,000 deaths' opens in France*, URL: `https://www.theguardian.com/world/2019/sep/23/trial-over-weight-loss-pill-behind-up-to-2000-deaths-to-start-in-france` (visited on 09/23/2019).

[Guy20] Thomas Guyet, "Enhancing sequential pattern mining with time and reasoning", PhD thesis, Université de Rennes 1, 2020.

[Her+15] Emily Herrett et al., "Data resource profile: clinical practice research datalink (CPRD)", *in*: *International journal of epidemiology* 44.*3* (2015), pp. 827–836.

[HKR09] Pascal Hitzler, Markus Krotzsch, and Sebastian Rudolph, *Foundations of semantic web technologies*, CRC press, 2009.

[Hor+04] Matthew Horridge et al., "A practical guide to building OWL ontologies using the Protégé-OWL plugin and CO-ODE tools edition 1.0", *in*: *University of Manchester* (2004).

[HP03] Ian Horrocks and Peter F Patel-Schneider, "Reducing OWL entailment to description logic satisfiability", *in*: *International semantic web conference*, Springer, 2003, pp. 17–29.

[Hri+15] George Hripcsak et al., "Observational Health Data Sciences and Informatics (OHDSI): opportunities for observational researchers", *in*: *Studies in health technology and informatics* 216 (2015), p. 574.

[HRS98] Thomas A Henzinger, J-F Raskin, and P-Y Schobbens, "The regular real-time languages", *in*: *International Colloquium on Automata, Languages, and Programming*, Springer, 1998, pp. 580–591.

[HS91] Joseph Y Halpern and Yoav Shoham, "A propositional modal logic of time intervals", *in*: *Journal of the ACM (JACM)* 38.*4* (1991), pp. 935–962.

[JHS06] Sheldon H Jacobson, Shane N Hall, and James R Swisher, "Discrete-event simulation of health care systems", *in*: *Patient flow: Reducing delay in healthcare delivery*, Springer, 2006, pp. 211–252.

[Joh+16] Alistair EW Johnson et al., "MIMIC-III, a freely accessible critical care database", *in*: *Scientific data* 3.*1* (2016), pp. 1–9.

[JS99] Christian S Jensen and Richard T Snodgrass, "Temporal data management", *in*: *IEEE Transactions on knowledge and data engineering* 11.*1* (1999), pp. 36–44.

[JSS94] Christian S Jensen, Michael D Soo, and Richard T Snodgrass, "Unifying temporal data models via a conceptual model", *in*: *Information systems* 19.*7* (1994), pp. 513–547.

[Kal+19] Elem Güzel Kalayci et al., "Ontology-based access to temporal data with Ontop: A framework proposal", *in*: *Int. J. of Applied Mathematics and Computer Science* 29.*1* (2019), pp. 17–30.

[Kir+16] Jacqueline C Kirby et al., "PheKB: a catalog and workflow for creating electronic phenotype algorithms for transportability", *in*: *Journal of the American Medical Informatics Association* 23.*6* (2016), pp. 1046–1052.

[KK15] Parminder Kaur and Aditya Khamparia, "Diagnosis of liver cancer ontology using SPARQL", *in*: *international journal of applied engineering research* 10.*69* (2015), pp. 15–18.

[Koz+15] Stanisław Kozielski et al., "Beyond databases, architectures and structures", *in*: *Zbornik radova* 11 (2015).

[Kru+18] Clemens Scott Kruse et al., "The use of electronic health records to support population health: a systematic review of the literature", *in*: *Journal of medical systems* 42.*11* (2018), pp. 1–16.

[Lev86] Hector J Levesque, "Knowledge representation and reasoning", *in*: *Annual review of computer science* 1.*1* (1986), pp. 255–287.

[LM97] Nikos A. Lorentzos and Yannis G. Mitsopoulos, "SQL extension for interval data", *in*: *IEEE Transactions on knowledge and Data Engineering* 9.*3* (1997), pp. 480–499.

[Maâ+13] Afef Jmal Maâlej et al., "Automated significant load testing for WS-BPEL compositions", *in*: *2013 IEEE Sixth International Conference on Software Testing, Verification and Validation Workshops*, IEEE, 2013, pp. 144–153.

[MAK12] Robert N Moll, Michael A Arbib, and Assaf J Kfoury, *An introduction to formal language theory*, Springer Science & Business Media, 2012.

[Moo+19] Jason H Moore et al., "Preparing next-generation scientists for biomedical big data: artificial intelligence approaches", *in*: *Personalized medicine* 16.*3* (2019), pp. 247–257.

[Mot12] Boris Motik, "Representing and querying validity time in RDF and OWL: A logic-based approach", *in*: *Journal of Web Semantics* 12 (2012), pp. 3–21.

[Mus15] Mark A Musen, "The protégé project: a look back and a look forward", *in*: *AI matters* 1.*4* (2015), pp. 4–12.

[MV+04] Deborah L McGuinness, Frank Van Harmelen, et al., "OWL web ontology language overview", *in*: *W3C recommendation* 10.*10* (2004), p. 2004.

[Nak+21] Eiji Nakatani et al., "Data resource profile of Shizuoka Kokuho Database (SKDB) using integrated health-and care-insurance claims and health check-ups: the Shizuoka Study", *in*: *Journal of Epidemiology* (2021), JE20200480.

[Noh04] Seo-Young Noh, "Literature Review on Temporal, Spatial, and Spatiotermpoal Data Models", *in*: (2004).

[OCo+09] Martin J O'Connor et al., "Knowledge-data integration for temporal reasoning in a clinical trial system", *in*: *Int. J. of Medical Informatics* 78 (2009), pp. 77–85.

[Og+00] Emmanuel Oger, EPI-GETBO study group, et al., "Incidence of venous thromboembolism: a community-based study in Western France", *in*: *Thrombosis and haemostasis* 83.*05* (2000), pp. 657–660.

[OS95] Gultekin Ozsoyoglu and Richard T Snodgrass, "Temporal and real-time databases: A survey", *in*: *IEEE Transactions on Knowledge and Data Engineering* 7.*4* (1995), pp. 513–532.

[OW06] Joël Ouaknine and James Worrell, "On metric temporal logic and faulty Turing machines", *in*: *International Conference on Foundations of Software Science and Computation Structures*, Springer, 2006, pp. 217–230.

[PA08] Iris Pigeot and Wolfgang Ahrens, "Establishment of a pharmacoepidemiological database in Germany: methodological potential, scientific value and practical limitations", *in*: *Pharmacoepidemiology and drug safety* 17.*3* (2008), pp. 215–223.

[Pac+18] Anil Pacaci et al., "A semantic transformation methodology for the secondary use of observational healthcare data in postmarketing safety studies", *in*: *Frontiers in pharmacology* 9 (2018), p. 435.

[Pal+16] Aurore Palmaro et al., "Overview of drug data within French health insurance databases and implications for pharmacoepidemiological studies", *in*: *Fundamental & clinical pharmacology* 30.*6* (2016), pp. 616–624.

[Pal17] Aurore Palmaro, “Measurement of discontinuous drug exposure in large healthcare databases”, PhD thesis, Université Toulouse 3 Paul Sabatier, 2017.

[Par15] Antoine Pariente, “La pharmaco-épidémiologie: une autre surveillance du médicament”, *in*: *Bulletin de l’Académie Nationale de Médecine* 199.*2-3* (2015), pp. 275–279.

[PD06] Pavithra Prabhakar and Deepak D’Souza, “On the expressiveness of MTL with past operators”, *in*: *International Conference on Formal Modeling and Analysis of Timed Systems*, Springer, 2006, pp. 322–336.

[PMV03] Josien PW Pluim, JB Antoine Maintz, and Max A Viergever, “Mutual-information-based registration of medical images: a survey”, *in*: *IEEE transactions on medical imaging* 22.*8* (2003), pp. 986–1004.

[Pog+08] Antonella Poggi et al., “Linking data to ontologies”, *in*: *Journal on data semantics X*, Springer, 2008, pp. 133–173.

[Pol+15] Elisabeth Polard et al., “Brand name to generic substitution of antiepileptic drugs does not lead to seizure-related hospitalization: a population-based case-crossover study”, *in*: *Pharmacoepidemiology and drug safety* 24.*11* (2015), pp. 1161–1169.

[Por14] Miquel Porta, *A dictionary of epidemiology*, Oxford university press, 2014.

[Pos+19] Andrew Post et al., “A method for EHR phenotype management in an i2b2 data warehouse”, *in*: *AMIA Summits on Translational Science Proceedings* 2019 (2019), p. 92.

[PUS08] Andrea Pugliese, Octavian Udrea, and VS Subrahmanian, “Scaling RDF with time”, *in*: *Proceedings of the 17th international conference on World Wide Web*, 2008, pp. 605–614.

[RDL19] Yann Rivault, Olivier Dameron, and Nolwenn Le Meur, “queryMed: Semantic Web functions for linking pharmacological and medical knowledge to data”, *in*: *Bioinformatics* (2019).

[Rei12] Wolfgang Reisig, *Petri nets: an introduction*, vol. 4, Springer Science & Business Media, 2012.

[Ric+20] R Richesson et al., *Rethinking Clinical Trials: A Living Textbook of Pragmatic Clinical Trials*, ed. by MD: NIH Health Care Systems Research Collaboratory, https://rethinkingclinicaltrials.org/chapters/conduct/electronic-health-records-based-phenotyping/definitions/, 2020.

[Riv19] Yann Rivault, "Analyse de trajectoires de soins à partir de bases de données médico-administratives : apport d'un enrichissement par des connaissances biomédicales issues du Web des données", PhD thesis, 2019.

[RW89] Peter JG Ramadge and W Murray Wonham, "The control of discrete event systems", *in*: *Proceedings of the IEEE* 77.*1* (1989), pp. 81–98.

[Ryb87] Henryk Rybiński, "On first-order-logic databases", *in*: *ACM Transactions on Database Systems (TODS)* 12.*3* (1987), pp. 325–349.

[Sah+18] Alexandre Sahuguède et al., "Mapping Chronicles to a k-dimensional Euclidean Space via Random Projections", *in*: *Proc. of Int. Conf. on Automation Science and Engineering (CASE)*, 2018, pp. 1177–1182.

[Sal+12] Manuel Salvadores et al., "Using sparql to query bioportal ontologies and metadata", *in*: *International semantic web conference*, Springer, 2012, pp. 180–195.

[Sar+17] Md Kamruzzaman Sarker et al., "Explaining trained neural networks with semantic web technologies: First steps", *in*: *arXiv preprint arXiv:1710.04324* (2017).

[Sca+19] Lucie-Marie Scailteux et al., "French administrative health care database (SNDS): the value of its enrichment", *in*: *Therapies* 74.*2* (2019), pp. 215–223.

[Sch+90] Albrecht Schmiedel et al., *A temporal terminological logic*, Techn. Univ., Projektgruppe KIT, 1990.

[Sch93] Klaus Schild, "Combining terminological logics with tense logic", *in*: *Portuguese Conference on Artificial Intelligence*, Springer, 1993, pp. 105–120.

[SÇZ05] Michael Stonebraker, Uğur Çetintemel, and Stan Zdonik, "The 8 requirements of real-time stream processing", *in*: *ACM Sigmod Record* 34.*4* (2005), pp. 42–47.

[SLL18] Alexandre Sahuguède, Euriell Le Corronc, and Marie-Véronique Le Lann, "An Ordered Chronicle Discovery Algorithm", *in*: *3nd ECML/PKDD Workshop on Advanced Analytics and Learning on Temporal Data, AALTD'18*, 2018.

[SLT98] Meera Sampath, Stéphane Lafortune, and Demosthenis Teneketzis, "Active diagnosis of discrete-event systems", *in*: *IEEE transactions on automatic control* 43.*7* (1998), pp. 908–929.

[Smi+07] Barry Smith et al., "The OBO Foundry: coordinated evolution of ontologies to support biomedical data integration", *in*: *Nature biotechnology* 25.*11* (2007), pp. 1251–1255.

[Sno+86] Richard Snodgrass et al., "Temporal databases", *in*: *Computer* 19.*09* (1986), pp. 35–42.

[Sno12] Richard T Snodgrass, *The TSQL2 temporal query language*, vol. 330, Springer Science & Business Media, 2012.

[Sno99] Richard T Snodgrass, *Developing time-oriented database applications in SQL*, timesql, 1999.

[SS06] Vijayan Sugumaran and Veda C Storey, "The role of domain ontologies in database design: An ontology management and conceptual modeling environment", *in*: *ACM Transactions on Database Systems (TODS)* 31.*3* (2006), pp. 1064–1094.

[SST18] Chayma Sellami, Ahmed Samet, and Mohamed Anis Bach Tobji, "Frequent chronicle mining: Application on predictive maintenance", *in*: *Proc. of Int. Conf. on Machine Learning and Applications (ICMLA)*, 2018, pp. 1388–1393.

[Sut+20] Reed T Sutton et al., "An overview of clinical decision support systems: benefits, risks, and strategies for success", *in*: *NPJ digital medicine* 3.*1* (2020), pp. 1–10.

[Tan+93] Abdullah Uz Tansel et al., *Temporal databases: theory, design, and implementation*, Benjamin-Cummings Publishing Co., Inc., 1993.

[Thu+19] N Thurin et al., "Standardisation de l'utilisation des données du Système national des données de santé à des fins de recherche médicale: présentation d'un modèle de données optimisé centré sur le patient", *in*: *Revue d'Épidémiologie et de Santé Publique* 67 (2019), S76.

[Tom03] David Toman, "On completeness of multi-dimensional first-order temporal logics", *in*: *10th International Symposium on Temporal Representation and Reasoning, 2003 and Fourth International Conference on Temporal Logic. Proceedings.* IEEE, 2003, pp. 99–106.

[Tom97] David Toman, "A point-based temporal extension of sql", *in*: *International Conference on Deductive and Object-Oriented Databases*, Springer, 1997, pp. 103–121.

[Tup+17] P Tuppin et al., "Value of a national administrative database to guide public decisions: From the système national d'information interrégimes de l'Assurance Maladie (SNIIRAM) to the système national des données de santé (SNDS) in France", *in*: *Revue d'epidemiologie et de sante publique* 65 (2017), S149–S167.

[TV12] M Thenmozhi and K Vivekanandan, "An ontology based hybrid approach to derive multidimensional schema for data warehouse", *in*: *International Journal of Computer Applications* 54.*8* (2012), pp. 36–42.

[VBQ+13] Alain Venot, Anita Burgun, Catherine Quantin, et al., *Informatique médicale, e-santé: fondements et applications*, Springer, 2013.

[Wan+10] Yafang Wang et al., "Timely YAGO: harvesting, querying, and visualizing temporal knowledge from wikipedia", *in*: *Proc. of Int. Conf. on Extending Database Technology (EDBT)*, 2010, pp. 697–700.

[Wei+10] Alain Weill et al., "Benfluorex and valvular heart disease: a cohort study of a million people with diabetes mellitus", *in*: *Pharmacoepidemiology and drug safety* 19.*12* (2010), pp. 1256–1262.

[Whe+11] Patricia L Whetzel et al., "BioPortal: enhanced functionality via new Web services from the National Center for Biomedical Ontology to access and use ontologies in software applications", *in*: *Nucleic acids research* 39.*suppl_2* (2011), W541–W545.

[WP19] John Weeks and Roy Pardee, "Learning to share health care data: a brief timeline of influential common data models and distributed health data networks in US health care research", *in*: *eGEMs* 7.*1* (2019).

[Zen+15] Xhemal Zenuni et al., "State of the art of semantic web for healthcare", *in*: *Procedia-Social and Behavioral Sciences* 195 (2015), pp. 1990–1998.

[ZG+15] Marie Zins, Marcel Goldberg, et al., “The French CONSTANCES population-based cohort: design, inclusion and follow-up”, *in*: *European journal of epidemiology* 30.*12* (2015), pp. 1317–1328.

[Zha+19] Fu Zhang et al., “Temporal Data Representation and Querying Based on RDF”, *in*: *IEEE Access* 7 (2019), pp. 85000–85023.

[ZL13] Janan Zaytoon and Stéphane Lafortune, “Overview of fault diagnosis methods for discrete event systems”, *in*: *Annual Reviews in Control* 37.*2* (2013), pp. 308–320.

Modèles Temporels pour explorer les bases de données administratives de santé

L'expansion rapide des données de santé et les millions de patients qu'elles contiennent, représentent une opportunité sans précédent pour améliorer la santé publique. Les quantités massives de données fournissent beaucoup d'informations et offrent un large éventail de possibilités pour les études épidémiologiques qui visent à améliorer la santé publique. Par exemple, en France, depuis 2008, l'assurance maladie enregistre les remboursements de soins de santé et les conserve dans une base de données. Ces données sont très vastes puisqu'elles contiennent près de 99 % de la population française avec des informations sur les remboursements de soins de santé: délivrances de médicaments, actes médicaux ou visites médicales et hospitalisations ainsi que des informations démographiques telles que âge, sexe et lieu de vie .

Même si l'utilisation initiale de cette base de données est le remboursement des produits de santé, les épidémiologistes ont tenté d'utiliser ces données pour des études de santé publique. La première étude à grande échelle [6, 2] à fort impact a débuté avec la suspension du benfluorex (*i.e.* médiator). Ce médicament controversé a été prgressivement retiré du marché en Espagne et en Italie, et autorisé en France jusqu'en 2009, date à laquelle il a été retiré du marché suite aux résultats de l'étude. Cette étude a comparé des patients diabétiques français exposés au benfluorex avec des patients diabétiques français non exposés au benfluorex. Elle conclut à un risque significativement accru de valvulopathie cardiaque pouvant conduire au décès pour la population traitée au benfluorex. Il s'agit de la première étude publique sans conflit d'intérêt, basée sur une large population, qui conclut sur le danger des traitements au benfluorex. L'exemple du benfluorex n'est qu'un petit exemple parmi les études menées par les épidémiologistes, mais il est le précurseur des études épidémiologiques menées grâce à la *base de données administratives de santé* française, appelé *SNDS* (Système National des Données de Santé) [5].

L'épidémiologie vise à étudier l'état de la santé publique, comme l'identification des facteurs de risque d'une certaine maladie (par exemple, l'utilisation du médiator pour les maladies cardiovasculaires) ou l'étude d'une maladie (par exemple, l'étude du nombre de trombo-embolies veineuses en Bretagne) ou l'enquête sur une épidémie (par exemple, l'étude de l'évolution du covid-19). Pour réaliser des études, le premier défi des épidémiologistes est de trouver des données de santé appropriées. Ces données peuvent être collectées pour l'étude (données cliniques) ou sont déjà disponibles grâce à l'existence de registres (*e.g.* R.E.I.N[1] en France), des cohortes (*e.g.* Constances) ou des bases de données administratives de santé (*e.g.* SNDS en France ou GePaRD en Allemagne).

Les données cliniques semblent idéales car elles contiennent beaucoup de détails tels que les procédures médicales, les rapports médicaux, des données quantitatives précises, et même des données sociales sur le mode de vie des patients (régime alimentaire, tabagisme, etc.). Cependant, aucun système d'information n'acquiert automatiquement des données cliniques. La collecte de telles informations pour répondre à une question épidémiologique précise demande beaucoup de temps et est très coûteuse alors que les base de données administratives de santé existantes contiennent déjà une large couverture de la population. L'existence d'une population exhaustive issue d'une base de données administratives de santé est un atout fort pour la fiabilité d'une étude épidémiologique. Pour le benfluorex, les données étaient déjà disponibles à grande échelle alors que la collecte de données cliniques aurait nécessité un montage expérimental long et n'aurait pu concerner qu'un petit nombre de patients.

[1] Le Réseau d'épidémiologie et d'information en néphrologie (REIN) est un système d'information concernant les traitements substitutifs de l'insuffisance rénale chronique dans le domaine de la santé publique `https://www.agence-biomedecine.fr/R-E-I-N-Reseau-Epidemiologique-et-Information-en-Nephrologie`

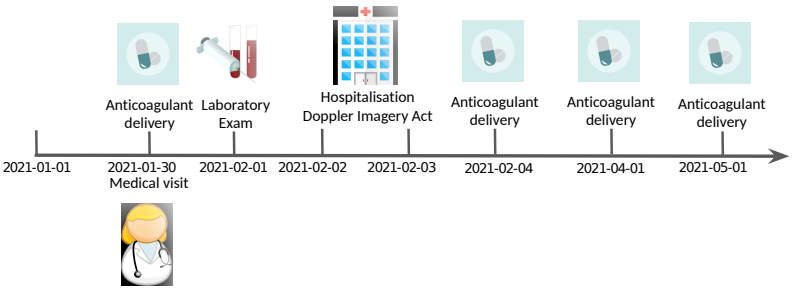


Figure 1: Illustration d'un séquence de soin

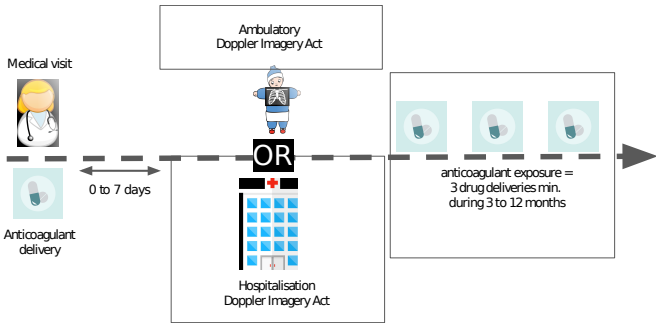


Figure 2: Illustration d'un parcours de soin aussi appelé phénotype poue trouver les patients atteints de TEV

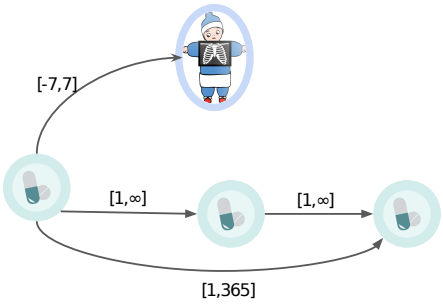


Figure 3: Illustration d'une Chronique représentant un phénotype de VTE

Pourtant, l'extraction d'une population appropriée pour une analyse épidémiologique à partir d'un grand ensemble de données de patients est une tâche informatique complexe. En pratique, les épidémiologistes décrivent un *parcours de soins* définissant la population à extraire, en prenant soin d'utiliser les informations fournies par la base de données. Le jeu de données des patients est un jeu de données d'individus où chaque individu a un ensemble d'événements médicaux appelé *séquence de soins*. Ensuite, la description doit être traduite par un informaticien spécialisé dans les bases de données pour retrouver les patients qui ont suivi ce parcours de soins. Illustrons une séquence de soins sur la figure 1 et un parcours de soins sur la figure 2. La figure 1 illustre un patient dont le parcours de soins est composé des événements suivants: une délivrance de médicaments à la date du "2021-01-30" ainsi qu'une visite chez un médecin généraliste à la même date puis un examen de laboratoire le lendemain, puis une hospitalisation de deux jours du "2021-02-02" au "2021-02-03" et enfin trois délivrances de médicaments aux dates du "2021-02-04", du "2021-03-01" et du "2021-04-01".

Imaginons maintenant un épidémiologiste qui extrait une population souffrant de thrombo-embolie veineuse (TEV). Il cherche à décrire un parcours de soins typique pour les patients atteints de TEV tel qu'illustré dans la Figure 2: Un acte médical (imagerie Doppler) avant ou après la délivrance d'anticoagulants pendant 0 à 7 jours et la délivrance dure au minimum 3 mois et maximum 12 mois. L'imagerie Doppler peut être réalisée ou non à l'hôpital.

La séquence de soins présentée Figure 1 vérifie le parcours de soins présenté Figure 2. En effet, la séquence de soins contient une délivrance d'anticoagulant suivie de l'imagerie Doppler à l'hôpital, puis de trois délivrances d'anticoagulant. Ainsi, l'épidémiologiste devrait retrouver ce patient dans son extraction.

Cette thèse vise à interroger des séquences de soins pour n'extraire que celles vérifiant un parcours de soins donné en proposant une approche formelle combinant expressivité et efficacité. La notion de parcours de soins comme critère de sélection d'une population est appelée *phénotype informatique* par Richesson *et al.* [4]. En épidémiologie, des diagrammes de décision sont usuellement utilisés pour formaliser les phénotypes. En informatique, ce formalisme doit être vu comme une requête. Pour détailler la notion de phénotype, nous réutilisons l'exemple de TEV Figure 2. Ci-dessous, les médecins proposent des descriptions de phénotypes permettant d'identifier les patients souffrant de TEV [3].

Dans la pratique clinique, face à une suspicion de TEV, les médecins prescrivent d'abord des anticuagulants, puis confirment ou non le diagnostic par des procédures médicales spécifiques: une échographie Doppler ou un scanner. Les patients suspectés d'embolie pulmonaire sont hospitalisés alors que les patients suspectés de thrombose veineuse profonde (TVP) sont pris en charge en ambulatoire. D'un côté, si la suspicion de TVP est confirmée, les traitements d'anticoagulant se poursuivent pendant 3 à 12 mois (une fois par mois). Ainsi, le diagnostic (par les mêmes procédures médicales que ci-dessus) est précédé ou suivi de l'initiation d'un traitement anticoagulant dans une fenêtre temporelle d'au plus 7 jours. D'autre part, la suspicion d'embolie pulmonaire entraîne une hospitalisation au cours de laquelle des procédures médicales sont effectuées pour confirmer le diagnostic, puis l'administration d'un traitement anticoagulant n'est observée qu'après le retour du patient à son domicile.

Il y a 2 dimensions importantes dans ces descriptions:

- l'utilisation de concepts ontologiques (acte d'imagerie Doppler / anticoagulant / spécialiste): le code de l'acte médical fourni à un patient est donné, mais il est plus précis qu'un critère "anticoagulant" et une connaissance symbolique du domaine est nécessaire pour concilier les deux. Ici, "anticoagulant" fait référence à une classe de médicaments décrite dans la taxonomie ATC, une classification internationale.[2]
- l'utilisation de contraintes temporelles entre les événements ("pendant ou avant 1 à 2 mois", "dans une fenêtre temporelle de 1 à 2 jours"): l'ordre temporel des soins et les durées/intervalles numériques précisent l'organisation temporelle des événements.

Cette thèse propose ainsi un formalisme pour exprimer les phénotypes avec ces deux dimensions tout en s'assurant que les requêtes résultant de ces formalismes sont efficaces. Dans notre contexte, une donnée n'est pas simplement une information binaire, textuelle ou un nombre, c'est une entité qui peut faire partie de plusieurs groupes nommés également traités comme des entités. L'un des défis majeurs dans le domaine médical est l'utilisation sémantiquement riche des données. Les données ont plusieurs niveaux de spécificités, elles sont typées, classées, et font parfois référence à des connaissances médicales. Un des exemples les plus parlants que nous allons considérer dans cette thèse est la classification d'un médicament fournie par la taxonomie ATC: l'élément "boîte de Calciparine" est une entité appartenant à un groupe: le groupe "Héparine", lui-même appartenant au groupe "Agents antithrombotiques", lui-même appartenant au groupe "Système nerveux". De nombreuses ontologies médicales internationales classent des concepts médicaux. Si on utilise pas d'ontologies disponibles pour sélectionner les individus qui se sont vus délivrer un anticoagulant, nous devons rechercher manuellement (faire une liste) de chaque boîte vendue sur le marché qui est un anticoagulant. Par exemple, au lieu d'utiliser le groupe "Agents antithrombotiques", nous dressons la liste des boîtes d'anticoagulants: previscan 20mg cpr 30, indione 50mg cpr 20, coumadin 10mg cpr 25, coumadin 10mg cpr 30, coumadin 5mg cpr 30, etc... Dans l'étude TEV, il existe une liste de près de 130 boîtes d'anticoagulants.

Dans le chapitre 5, nous développons une ontologie de la base de données du SNDS. Cette ontologie définit des concepts liés aux données de santé, tels que le concept de délivrance de médicaments ou d'hospitalisation. Ces concepts nous permettent de lier cette ontologie à des ontologies internationales et ainsi d'exploiter les connaissances médicales existantes. De plus, cette représentation propose d'intégrer explicitement les concepts de séquence de soins et d'événement de soins. Ceci permet de répondre au défi suivant visant à interroger les parcours de soins sur les séquences de soins.

Le deuxième objectif de cette thèse est l'expression des phénotypes, expliquée dans les chapitres 4 et 6. Comme vu précédemment, un phénotype doit pouvoir exprimer des événements médicaux liés à des ontologies mais aussi liés par des contraintes temporelles. Dans notre contexte, les contraintes temporelles sont des intervalles temporels qui désignent l'intervalle de temps entre

[2]ATC: Anatomical Therapeutic Chemical Classification System.

deux événements. Par exemple, pour la TEV, nous nous intéressons à l'imagerie Doppler suivie dans la semaine d'une délivrance d'anticoagulant. Cette contrainte est essentielle, une délivrance d'anticoagulant douze mois plus tard peut renvoyer à une autre pathologie que la TEV. En pratique, il est facile de rechercher une délivrance d'anticoagulant à la date du "2021-01-30", mais il est plus complexe de trouver une délivrance qui a eu lieu après une hospitalisation et que cette même hospitalisation a été précédée d'une visite chez un spécialiste.

Avec les outils d'interrogation utilisés actuellement (détaillés dans le chapitre sur l'état de l'art), il n'existe pas de modèles temporels génériques pour formaliser la structure d'un phénotype. Ce sont les informaticiens qui travaillent avec les épidémiologistes qui construisent une requête ou un ensemble de requêtes traduisant la description d'un phénotype. Or, la construction d'une requête est longue et nécessite une bonne connaissance de la base de données. Pour éviter ce problème nous proposons dans le Chapitre 4 le modèle des Chroniques de Dousson [1] pour formaliser la notion de phénotypes. Ce modèle définit des occurrences d'événements liées par des contraintes temporelles. L'algorithme existant des Chroniques permet de vérifier des séquences vérifiant les contraintes temporelles et les événements. Nous étendons la notion d'événement avec des concepts de taxonomie et ainsi pouvoir exploiter les informations contenues dans les séquences de soins et les ontologies internationales. Ce formalisme est d'autant plus intéressant qu'il peut être représenté par un graphe de contraintes qui constitue un visuel. Un exemple simple de Chronique est présenté dans la Figure 3. La Chronique représente le phénotype de la Figure 2 où elle désigne qu'une délivrance d'anticoagulant est précédée/suivie entre 7 jours avant et 7 jours après par une imagerie Doppler et que ce même anticoagulant est suivi de deux autres anticoagulants dans l'année (1 à 365 jours).

Enfin, nous avons créé l'outil HyCOR pour réaliser l'extraction de séquences de soins vérifiant une Chronique. Il s'agit d'un outil hybride combinant la gestion des ontologies du Web sémantique et l'efficacité des algorithmes des Chroniques. HyCOR est très puissant, il trouve des ensembles de patients vérifiant un phénotype complexe, en utilisant l'ontologie et les contraintes temporelles, en quelques secondes. L'outil développé a été testé sur des données synthétiques montrant qu'une chronique à 15 événements dans un dataset contenant 20000 patients avec 500 événements par patient prenait moins d'une minute pour extraire 4000 patients. L'application sur données réelles a permis d'extraire 1184 patients souffrant de TEV depuis un sous-ensemble des données du SNDS (env 377 000 individus) en moins de 30 secondes.

References

[1] Christophe Dousson and Pierre Le Maigat. Chronicle recognition improvement using temporal focusing and hierarchization. In *Proc. of Int. Join Conf. on Artificial Intelligence (IJCAI)*, pages 324–329, 2007.

[2] The Guardian. Trial over weight-loss pill behind 'up to 2,000 deaths' opens in france.

[3] Emmanuel Oger, EPI-GETBO study group, et al. Incidence of venous thromboembolism: a community-based study in western france. *Thrombosis and haemostasis*, 83(05):657–660, 2000.

[4] R Richesson, LK Wiley, S Gold, and L Rasmussen. *Rethinking Clinical Trials: A Living Textbook of Pragmatic Clinical Trials.* https://rethinkingclinicaltrials.org/chapters/conduct/electronic-health-records-based-phenotyping/definitions/, 2020.

[5] P Tuppin, J Rudant, P Constantinou, C Gastaldi-Ménager, A Rachas, L De Roquefeuil, G Maura, H Caillol, A Tajahmady, J Coste, et al. Value of a national administrative database to guide public decisions: From the système national d'information interrégimes de l'assurance maladie (sniiram) to the système national des données de santé (snds) in france. *Revue d'epidemiologie et de sante publique*, 65:S149–S167, 2017.

[6] Alain Weill, Michel Païta, Philippe Tuppin, Jean-Paul Fagot, Anke Neumann, Dominique Simon, Philippe Ricordeau, Jean-Louis Montastruc, and Hubert Allemand. Benfluorex and valvular heart disease: a cohort study of a million people with diabetes mellitus. *Pharmacoepidemiology and drug safety*, 19(12):1256–1262, 2010.

UNIVERSITÉ DE RENNES 1

Titre : Modèles Temporels pour l'exploration des bases de données de santé administratives

Mot clés : Modèle Temporel, Chroniques, Phénotype, Web sémantique, OWL, SNDS

Résumé : La pharmaco-épidémiologie étudie les avantages et les risques de l'utilisation des médicaments sur la population dans des situations réelles où ces populations sont issues de bases de données médicales. Cette thèse porte sur les bases de données de santé administratives avec une application sur la base française appelée SNDS (Système National des Données de Santé). Le SNDS contient des informations administratives de type remboursements de soins : délivrances de médicaments, actes médicaux et hospitalisations mais ne contient pas de rapports médicaux, ni de diagnostic. L'absence de diagnostic rend la sélection de population d'intérêt complexe pour les pharmaco-épidémiologistes. L'objectif de cette thèse est de faciliter la sélection de populations d'intérêts dans les bases de données de santé administratives avec un outil intuitif et efficace. Cette thèse propose une approche formelle basée sur les Chroniques permettant de caractériser des phénotypes temporels, c'est-à-dire une description d'évènements médicaux témoignant d'une maladie, et une autre approche basée sur le Web Sémantique proposant une représentation des données avec OWL. Cette thèse propose l'outil efficace HyCOR, utilisant le Web Sémantique combiné au modèle temporel des Chroniques afin de trouver tous les patients exprimant un phénotype temporel donné. Cet outil a été testé sur un cas d'étude réel visant à sélectionner les patient atteints de thrombose veineuse dans le SNDS.

Title: Temporal Model to explore Administrative Healthcare Databases

Keywords: Temporal Model, Chronicles, Phenotype, Semantic Web, OWL, SNDS

Abstract: Pharmacoepidemiology studies the benefits and risks of drug use on the population in real situations where these populations are issued from medical databases. This thesis focuses on administrative healthcare databases with an application on the French database called SNDS (Système National des Données de Santé). The SNDS contains administrative information such as reimbursements of care: deliveries of drugs, medical acts and hospitalizations but does not contain medical reports or diagnoses. The absence of diagnosis makes the selection of the population of interest complex for pharmacoepidemiologists.The objective of this thesis is to facilitate the selection of populations of interest in administrative healthcare databases with an intuitive and efficient tool. This thesis proposes a formal approach based on Chronicles to characterize temporal phenotypes, i.e. a description of medical events testifying of a disease, and another approach based on the Semantic Web proposing a data representation with OWL. This thesis proposes the efficient HyCOR tool, combining Semantic Web tools with the Chronicles temporal model to find all patients expressing a given temporal phenotype. This tool has been evaluated on a real use case aiming at selecting patients suffering of venous thrombosis in the SNDS.