Search for exposomic causality of liver fibrosis using network analysis

Cécile Beust

Bioinformatics Master's Degree 1st year

University of Rennes 1

Internship Report

Cécile BEUST

Search for exposomic causality of liver fibrosis using network analysis

---

Supervised by

Olivier DAMERON and Nathalie THÉRET

Defended on 04/07/2022

ENGAGEMENT DE NON PLAGIAT

Je, soussigné (e) Cécile BEUST……
Etudiant (e) en Master 1 Bioinformatique……

Déclare être pleinement informé (e) que le plagiat de documents ou d'une partie de documents publiés sous toute forme de support (y compris l'internet), constitue une violation des droits d'auteur ainsi qu'une fraude caractérisée.

En conséquence, je m'engage à citer toutes les sources que j'ai utilisées pour la rédaction de ce document.

Signature

INSERM U1242 OSS
Equipe PROSAC

Centre Eugène Marquis
Avenue de la Bataille Flandres Dunkerque
35042 Rennes

Annabelle MONNIER
annabelle.monnier@univ-rennes1.fr

TÉL. 33 (0)2 23 23 61 14

Acknowledgements

I would like to warmly thank my supervisors, Olivier DAMERON and Nathalie THERET, for this internship opportunity, for their kindness and their precious advice as well as their great support throughout this internship.

I would like to thank the DYLISS and Dymec teams members for their warm welcome, their good mood, and their feedbacks on my work. Thanks also to the other IRISA's interns for the good atmosphere in the intern room, as well as to Guillaume COLLET for the batucada lessons on lunck break!

I would also like to thank Sébastien AUBER, who did his internship in 2021 on a similar subject, for his help with the use of some tools. Thanks also to Matéo BOUDET for helping me in the installation of CadBiom and make it work on the Genouest cluster.

I also thank my referent teacher Annabelle MONNIER for her availability and her external support during this internship.

Table of Contents

1 Introduction **1**

1.1 Chronic liver diseases and the exposome concept 1

1.2 Integration of databases thanks to Semantic Web technologies 2

1.3 Goals of the internship 3

2 Materials and methods **4**

2.1 The Comparative Toxicogenomics Database (CTD) as an information source . 4

2.2 Translation of the CTD information into BioPAX format 5

2.3 CadBiom : an application to build dynamic networks 6

3 Results **7**

3.1 Obtaining a BioPAX file describing the content of the CTD 7

3.1.1 Evaluation of the reusability of the 2019 BioPAX file of the CTD . . . 7

3.1.2 Re-use the Java CTD-to-BioPAX converter 8

3.2 Writing a standalone CTD-to-BioPAX converter 8

3.2.1 Convert the tabulated files of the CTD 9

3.2.2 Analysis of the Chemical-Gene interactions file of the CTD 10

3.2.3 Conversion of some simple leaves 12

3.2.4 Extraction of the graph of dependencies between the CTD interactions . 13

3.2.5 Comparaison with the 2019 BioPAX file 13

4 Discussion **14**

4.1 Analysis of the new CTD BioPAX file 14

4.2 Possible improvements for the converter 14

4.3 Next steps of the analysis 15

5 Conclusion **15**

Bibliography **i**

6 Appendixes **iii**

6.1 Presentation of the host structure iii

6.2 Personal assessment of the internship iv

List of abbreviations

BioPAX : Biological Pathway Exchange

CadBiom : Computer Aided Design of BIOlogical Models

CTD : Comparative Toxicogenomics Database

CLD : Chronic Liver Diseases

CSV : Comma-Separated Values

LOD : Linked Open Data

OWL : Ontology Web Language

RDF : Ressource Description Framework

SPARQL : SPARQL Protocol and RDF Query Language

TSV : Tabulation-Separated Values

URI : Uniform Resource Identifier

W3C : World Wide Web

XML : Extensible Markup Language

1 Introduction

1.1 Chronic liver diseases and the exposome concept

Chronic liver diseases (CLD) are long-term diseases that lead to a progressive deterioration of the liver functions. They are characterized by chronic inflammation, destruction of the liver parenchyma and the formation of regeneration nodules, associated with the development of fibrosis and cirrhosis [12]. The causes of these diseases are multiple : viral infection (hepatitis B, C), alcohol overconsumption, metabolic disorders (Non-Alcoholic Fatty Liver Disease), or genetic factors. The diversity of these causal factors makes chronic liver diseases an important public health issue. They have become an epidemiological problem, given the wide range of people affected throughout the world, and the considerable impact of our lifestyle and environment on the development of these diseases [3]. Indeed environmental contaminants contribute to the development of chronic liver diseases [2]. This leads us to the exposome concept, which is defined as the set of environmental exposures to which an individual is exposed through his life from conception to death [14] (Figure 1) . Identifying the links between exposure factors and chronic liver diseases is therefore a key factor in the understanding of these pathologies.

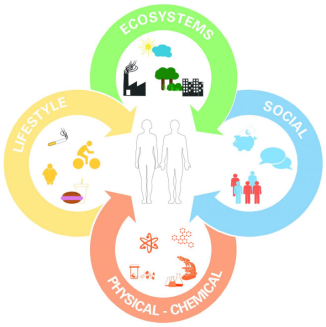

Figure 1. The Exposome is the sum of all the factors our body is exposed to (Escher et al. [9])

In order to study the exposome and how it affects human health, it is important to store and be able to access information about chemicals and any type of environmental exposure, as well as their possible involvements in the onset of diseases or physiological disorders. There are many specialized databases that store information about toxicology like CTD[1], T3DB[2] or Exposome Explorer[3]. Within the framework of this internship, we used the Comparative Toxicogenomics Database (CTD) because it describes environmental exposures at several levels of association

[1] http://ctdbase.org/
[2] http://www.t3db.ca/
[3] http://exposome-explorer.iarc.fr/

(chemical-gene, chemical-disease and gene-disease associations), giving a complete and global vision of the impact of exposures on human health. CTD is the database that provides access to a large amount of cross-referenced information about the impact of chemical exposures on the development of liver diseases.

1.2 Integration of databases thanks to Semantic Web technologies

To analyse the biological interactions described in the CTD, we have to integrate its content in order to structure it and give it meaning. Integrating CTD data would allow to cross-reference its information with other databases. For example, to characterize an entity like a gene, we need to call on the Gene Ontology [4] terms. This need for integration is increasingly felt in the field of life sciences, because of the large expansion of biological data over the last decades [11] [10].

Integrating biological data can be facilitated thanks to the Semantic Web technologies. The Semantic Web is an extension of the World Wide Web in which information is given well-defined and explicitly meaning, better enabling computers and people to work together in cooperation [4]. Thus, the aim of the Semantic Web is to give sense to the Web data, making them interpretable by a machine. This is enabled thanks to standards like RDF (Resource Description Framework)[5], allowing automatic processing and interoperability of data. In RDF, resources are described with identifiers called URIs (Uniform Resource Identifiers) and are represented as triples (subject, predicate, object) as shown in Figure 2. The subject is the resource we are describing, the predicate is the relationship describing the subject, and the object is one of the values of the predicate for the subject.

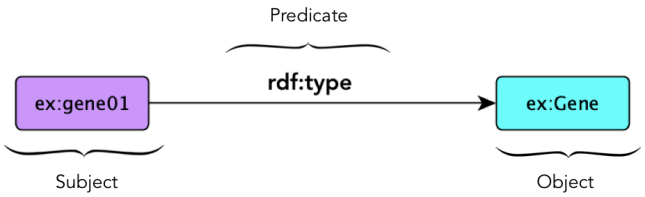

Figure 2. Example of a RDF triple represented as an oriented graph. This triple describes the instantiation relationship between the object `gene01` and its belonging class `Gene`. It describes that the object `gene01` is an instance of the class `Gene` through the predicate `rdf:type`.

[4] `http://geneontology.org/`
[5] `https://www.w3.org/TR/rdf11-concepts/`

A dataset described as RDF triples can be represented as an oriented graph, a web of information about related things ([4]). RDF data can be queried thanks to the SPARQL query language[6].

Semantic web standards also include ontologies, which are computational models explicitly representing the meaning of terms and the relationships between those terms [5]. Ontologies provide controlled vocabularies shared by the community, and allow to describe any type of resource and relationship with the use of hierarchical classes and subclasses and their corresponding properties.

Integration of databases content into a Semantic Web's format is important because most databases like CTD often use their own file formats and are not interoperable with each other. Many life science reference resources are already available in Semantic Web format as shown in Figure 3. Unfortunately, this is not the case for the up-to-date CTD.

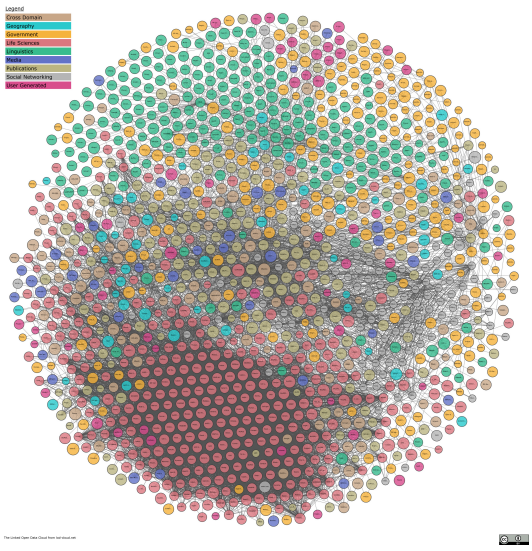

Figure 3. LOD (Linked Open Data) Cloud, representing all RDF datasets linked together in the Web. Life sciences datasets are represented in pink and the density of black lines in this section shows strong interconnection of resources.

1.3 Goals of the internship

The aim of this internship is to use the Comparative Toxicogenomics Database as a source of information to identify environmental factors that contribute to the onset of chronic liver diseases. The first objective was to integrate the CTD data into a Semantic Web format. We chose the BioPAX format, which is adapted to the description of biological pathway data in

[6] `https://www.w3.org/TR/sparql11-overview/`

accordance with Semantic Web standards. Ultimately, the integrated CTD data would allow to build large-scale networks of chemical-gene-disease associations with the CadBiom application. The interrogation of these networks would allow to identify factors triggering the expression genes associated with chronic liver diseases.

2 Materials and methods

2.1 The Comparative Toxicogenomics Database (CTD) as an information source

The Comparative Toxicogenomics Database (`http://ctdbase.org/`) is a publicly available database describing chemical-gene, chemical-disease and gene-disease associations to better understand the impact of environmental exposures on human health. It uses controlled vocabularies about chemicals, genes, diseases, anatomy and pathways as well as ontologies to favor interoperability with other databases and the interconnection of knowledge. The CTD is updated every month and is enriched by inferring relationships through text mining in articles. The information is manually curated by professionals, thus ensuring the accuracy of the stored data [6] (Figure 4). Today the CTD describes more than 45 million toxicogenomic relationships involving 16300 chemicals, 51300 genes, 5500 phenotypes, 7200 diseases and 16300 exposure events for more than 600 species [7]. The CTD is made up of 18 data files freely downloadable in csv, tsv and xml formats via the CTD download tab.

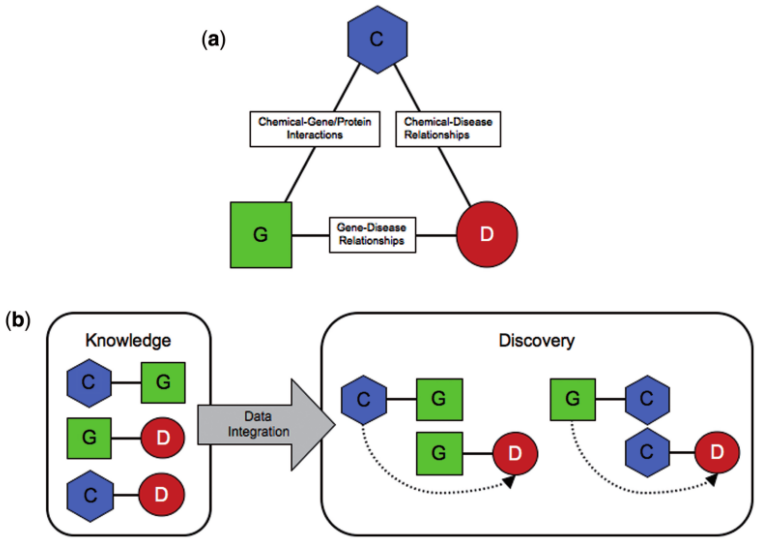

Figure 4. Chemical-Gene-Disease associations described in the CTD (a) and inference process of new associations from literature (b) (Davies et al. [6])

2.2 Translation of the CTD information into BioPAX format

Information on chemical-gene-disease associations from the CTD is defined as biological pathway data since it represents biological regulatory processes. More precisely, a pathway is a set of interactions between physical or genetic components, often describing a causal-and-effect or time-dependant process, that explains an observable biological phenomenon. To answer our biological question we need to integrate CTD information into a Semantic Web's format in order to browse, query, visualize and analyze it. We therefore chose to translate the CTD data in the BioPAX format (Biological Pathway Exchange, `http://www.biopax.org/`) which is an open standard knowledge format for the representation of pathway interactions [8]. It is also a community-developed language that aims to improve the sharing, integration and unification of pathway data knowledge between databases. BioPAX format uses the RDF formalism, and is implemented as an OWL Ontology, which means that the data are represented as hierarchichal structured classes and subclasses having their own properties while inheriting the properties of superclasses. The BioPAX ontology helps to describe pathway data by providing a controlled vocabulary for biological interactions as shown in Figure 5. The level 3 of the BioPAX ontology used for this project was released in 2010 and is available at `http://www.biopax.org/release/biopax-level3.owl`

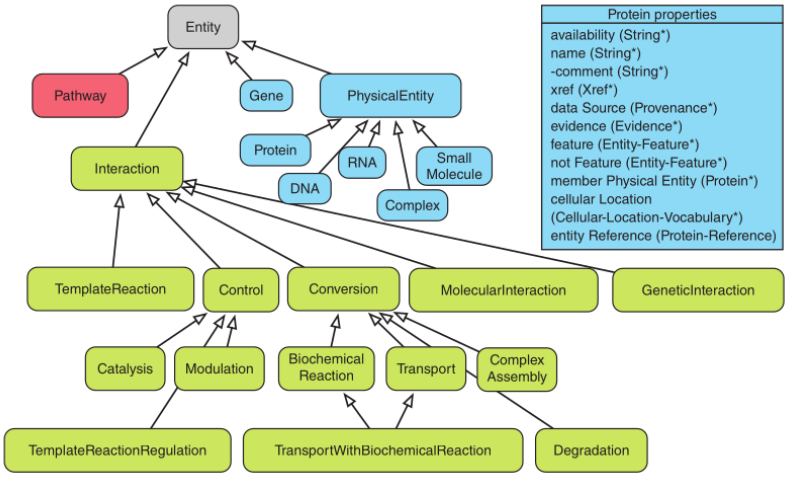

Figure 5. Simplified representation of the hierarchy of the main BioPAX classes. The Entity class is the parent class of all the other BioPAX classes. In the blue box are listed the properties we can apply to the instances of the Protein class (Demir et al., [8])

BioPAX allows the description of several types of pathway data such as metabolic, molecular and signaling pathways and is also suitable for the representation of genetic interactions and gene regulation networks. BioPAX is thus well adapted to represent the pathway data of the CTD.

Moreover, the CadBiom application that we want to use to represent the pathways can take as input BioPAX files to convert it in the BCX format thanks to the program *biopax2cadbiom* (`https://gitlab.inria.fr/DYLISS/biopax2cadbiom`) developed by Pierre Vignet [13].

Some of the data contained in the CTD are available in the BioPAX language on Pathway Commons (`https://www.pathwaycommons.org/`) in the form of an OWL file. Pathway commons is an online database storing integrated pathway data from multiple databases in a Semantic Web's format. This BioPAX file of the CTD was submitted in 2019, and the CTD-to-BioPAX converter used was coded by Igor Rodchenkov and is available on his public GitHub repository (`https://github.com/PathwayCommons/ctd-to-biopax`). We explored the content of this file with SPARQL queries and it revealed that the file was not usable because of its incompletness. We tried to reuse the available converter, but due to system compatibility issues, we were never able to get it to work. So, we chose to recode a CTD-to-BioPAX converter in Python [7] in order to have an up-to-date and well-described BioPAX file of the CTD. The Python converter takes as input a tsv file downloaded from the CTD and gives as output a BioPAX file describing the content of the tabulated file in a RDF formalism. One of the main advantages of representing data in RDF is that it allows automatic querying of datasets with the SPARQL query language. Thanks to SPARQL queries we can analyze the relationships between entities in RDF datasets: class and subclass identification, determination and count of direct or indirect instances of a class, characterization of the relationships linking instances. To perform SPARQL queries we used the Apache Jena Fuseki server software (`https://jena.apache.org/documentation/fuseki2/`).

After getting our BioPAX file of the CTD, we converted some interactions into an oriented graph. To do this we used a bash script *rdf2image*, available on the GitLab repository of Olivier DAMERON (`https://gitlab.com/odameron/rdf2image`).

2.3 CadBiom : an application to build dynamic networks

CadBiom (Computer Aided Design of BIOlogical Model, `http://cadbiom.genouest.org/`) is an application developed by the DYLISS and the Dymec teams, allowing to model the dynamics of biological networks as discrete events [1]. CadBiom models biological interactions

[7] `https://www.python.org/`

on the basis of guarded transition formalism. This discrete modeling approach models system dynamics by taking into account the competition and cooperation events in chains of reactions [13]. A guarded transition is defined with a quadruplet as shown in Figure 6.

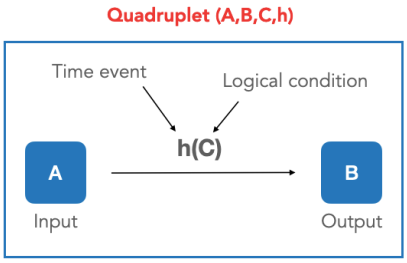

Figure 6. Guarded transition modeled with CadBiom. A guarded transition is defined with a quadruplet (A,B,C,h). A is the input, B is the output, h is the time event and C is the logical condition for the transition to occur. The transition can only occur if A is activated and if C is verified. If the transition occurs, A is inactivated and B is activated. All entities sharing the same time event will occur at the same moment.

A set of transitions forms a trajectory and all trajectories constitute the final biological network. CadBiom also allows to query the networks by "Model Checking" approaches. This is a property of the software allowing to answer a biological question such as *"What leads to the activation of a given gene ?"*, by checking all the state variables and chronology of events leading to the activation of this gene. Model checking enables to select trajectories that can answer the biological question.

3 Results

3.1 Obtaining a BioPAX file describing the content of the CTD

3.1.1 Evaluation of the reusability of the 2019 BioPAX file of the CTD

First, we tried to re-use the available BioPAX file of the CTD on PathwayCommons. This file is from 2019, and was built using 3 files (chemical-gene interactions, chemical vocabulary and gene vocabulary) of the 18 files of the CTD. We explored its content using SPARQL queries in order to qualify and to quantify the contained information. SPARQL queries revealed that there was a considerable loss of information compared to the original data of the CTD. The information was partially described. For example, the chemical `"10074-G5"` (an inhibitor of c-Myc-Max dimerisation) is involved in 15 reactions according to the current CTD taublated files, but only appears in 3 of them according the the 2019 BioPAX version (Figure 7). It could be explained by the fact that the file is not up-to-date, considering that the CTD is updated every

month. But apart from that, the information contained in the file was not clearly described and the way the file was build was unclear.

# ChemicalName	ChemicalID	CasRN	GeneSymbol	GeneID	GeneForms	Organism	OrganismID	Interaction	InteractionActions	PubMedIDs
10074-G5 C534883	AR	367	protein	Homo sapiens	9606			10074-G5 affects the reaction [MYC protein results in increased expression of AR protein]	affects^reaction|increases^expression	32184358
10074-G5 C534883	AR	367	protein	Homo sapiens	9606			10074-G5 inhibits the reaction [EPHB2 protein modified form results in increased expression of AR protein]	decreases^reaction|increases^expression	32184358
10074-G5 C534883	AR	367	protein	Homo sapiens	9606			10074-G5 results in decreased expression of AR protein	decreases^expression	32184358
10074-G5 C534883	AR	367	protein	Homo sapiens	9606			10074-G5 results in decreased expression of AR protein alternative form	decreases^expression	32184358
10074-G5 C534883	EPHB2	2048	protein	Homo sapiens	9606			10074-G5 inhibits the reaction [EPHB2 protein modified form results in increased expression of AR protein]	decreases^reaction|increases^expression	32184358
10074-G5 C534883	EPHB2	2048	protein	Homo sapiens	9606			10074-G5 inhibits the reaction [EPHB2 protein modified form results in increased expression of MYC protein]	decreases^reaction|increases^expression	32184358
10074-G5 C534883	MAX	4149	protein					10074-G5 affects the folding of and results in decreased activity of [MYC protein binds to MAX protein]	affects^binding|affects^folding|decreases^activity	26474287
10074-G5 C534883	MAX	4149	protein					10074-G5 inhibits the reaction [MYC protein binds to MAX protein]	affects^binding|decreases^reaction	26474287
10074-G5 C534883	MYC	4609	protein	Homo sapiens	9606			10074-G5 affects the reaction [MYC protein results in increased expression of AR protein]	affects^reaction|increases^expression	32184358
10074-G5 C534883	MYC	4609	protein	Homo sapiens	9606			10074-G5 analog results in decreased expression of MYC protein	decreases^expression	26036281
10074-G5 C534883	MYC	4609	protein	Homo sapiens	9606			10074-G5 inhibits the reaction [EPHB2 protein modified form results in increased expression of MYC protein]	decreases^reaction|increases^expression	32184358
10074-G5 C534883	MYC	4609	protein	Homo sapiens	9606			10074-G5 results in decreased activity of MYC protein	decreases^activity	25716159
10074-G5 C534883	MYC	4609	protein	Homo sapiens	9606			10074-G5 results in decreased expression of MYC protein	decreases^expression	26036281|32184358
10074-G5 C534883	MYC	4609	protein					10074-G5 affects the folding of and results in decreased activity of [MYC protein binds to MAX protein]	affects^binding|affects^folding|decreases^activity	26474287
10074-G5 C534883	MYC	4609	protein					10074-G5 inhibits the reaction [MYC protein binds to MAX protein]	affects^binding|decreases^reaction	26474287

Figure 7. 15 chemical-gene interactions involving the chemical `10074-G5` (inhibitor of c-Myc-Max dimerisation) stored in the CTD. The 3 interactions described in the BioPAX file of the CTD on PathwayCommons are highlighted in yellow.

3.1.2 Re-use the Java CTD-to-BioPAX converter

Since the 2019 BioPAX file was not usable, we tried to regenerate a 2022 version of the CTD using the CTD-to-BioPAX converter developed by Igor Rodchenkov (`https://github.com/PathwayCommons/ctd-to-biopax`) that was used to build the 2019 file. But I encountered some system incompatibility issues with this converter that I never managed to get it to work, even with the help of Sebastien AUBER who worked with it during his 2021 internship, nor with the help of Matéo BOUDET, an engineer from the Genouest platform. The converter was coded in Java and the code was not easy to understand for people who don't know this language. Moreover, the Java code was structured inside a Maven project [8] and the main problem I encountered was to deal with several versions of Java. Some packages of the Maven project were suitable for an older version of Java and it didn't worked on more recent systems, even with the use of virtualization processes like Docker [9] or using virtual environments on Genouest virtual machines. Altogether, this made future use of the converter not sustainable. Thus, we decided to code a new CTD-to-BioPAX converter using the Python language.

3.2 Writing a standalone CTD-to-BioPAX converter

The code of the CTD-to-BioPAX converter is available on my GitLab repository (`https://gitlab.com/cecilebeust/ctd_to_biopax.git`).

[8] `https://maven.apache.org/`
[9] `https://www.docker.com/`

3.2.1 Convert the tabulated files of the CTD

To convert the content of the CTD into BioPAX, we must first understand how it is represented in the database itself. This is not documented on the CTD website nor in the CTD articles. We chose to focus first on the chemical-gene interactions file, and took an example of such interaction involving the chemical `"10074-G5"` and searched for the corresponding reactions. In the chemical-gene interactions file, one line describes one interaction between one chemical and one gene, and values of each column are separated by tabulations. Table 1 is a table containing the values from the file for a line that describes a relationship involving the chemical `"10074-G5"`. This is the information that will be used to build the RDF triples describing the interaction.

ChemicalName	10074-G5
ChemicalID	C534883
CasRN	
GeneSymbol	AR
GeneID	367
GeneForms	protein
Organism	Homo sapiens
OrganismID	9606
Interaction	10074-G5 results in decreased expression of AR protein
InteractionActions	decreases expression
PubMedIDs	32184358

Table 1. CTD chemical-gene interactions file content example for one line describing one interaction involving the chemical `"10074-G5"`. The CasRN cell is empty because there was no value for the Cas Registry Number for the chemical `"10074-G5"`.

The CTD cross-references information from different databases to build these relationships. It uses the MeSH database [10] to identify the chemical with a `ChemicalID`. When it is available, the chemical also has a Cas Registry Number [11] (`CasRN`, which is not the case here). The CTD also uses NCBI Gene Identifiers [12] to identify the gene involved (`GeneID`), and a NCBI Taxonomy identifier [13] for the organism (`OrganismID`). Each interaction has also an assigned PubMed identifier (`PubMedIDs`) that links it to a literature reference in PubMed [14]. The need to use multiple databases to characterize an entity is justified by the fact that resource descriptions are fragmented on the Web. This is where BioPAX comes in, as it aims to improve the integration and efficient reuse of data between databases [8].

[10] `https://www.ncbi.nlm.nih.gov/mesh/`
[11] `https://www.cas.org/fr/cas-data/cas-registry`
[12] `https://www.ncbi.nlm.nih.gov/gene`
[13] `https://www.ncbi.nlm.nih.gov/taxonomy`
[14] `https://pubmed.ncbi.nlm.nih.gov/`

Taking into account these information, we can build a graph representing the entities involved in this interaction, the relationships between them and the BioPAX class to which they can be connected (Figure 8).

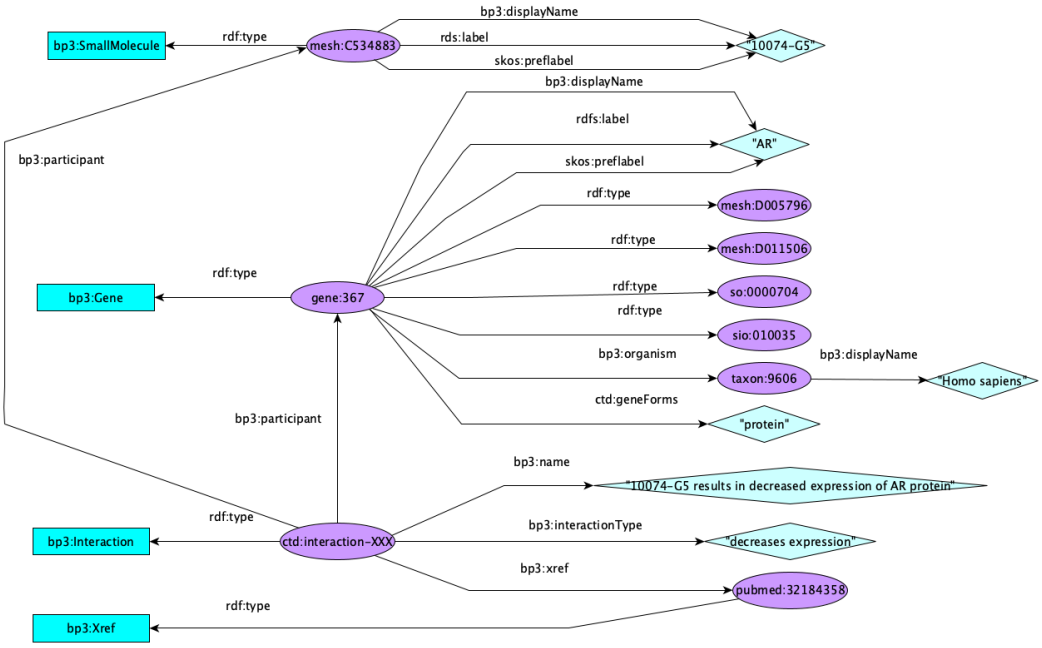

Figure 8. Graph representing an interaction involving the chemical `"10074-G5"`. BioPAX classes are represented by blue boxes and their instances by purple ovals. Each arrow represents a relation linking an instance to its reference class, or linking two instances together. For each relation, the predicate (the nature of the relation) is indicated above the concerned arrow. Strings referring to compound, gene, organism or interaction name are represented by light blue lozenges.

For the description of entities involved in the interactions, we added some external identifiers to those already used in the CTD in order to cross-reference the information about the entities available in other database. We used `SO` [15] and `SKOS` [16] identifiers which refer to other gene ontologies. The `mesh:D005796` instance refers to the Gene section of the MeSH database and the `mesh:D011506` instance refers to the Protein section. After characterizing chemicals and genes, we assigned chemicals to the `bp3:SmallMolecule` BioPAX class and the genes to the `bp3:Gene` class (`bp3` is the prefx meaning BioPAX level 3).

3.2.2 Analysis of the Chemical-Gene interactions file of the CTD

To describe the interactions between chemicals and genes, we used the *InteractionActions* column of the CTD file, which is the one precising the type of the interaction and its effects.

[15] `http://purl.obolibrary.org/obo/SO_`
[16] `http://www.w3.org/2004/02/skos/core#`

In this column, elementary interactions are presented chained to each other by a pipe character, as presented on Figure 7. The column contains 3264 combinations of 134 elementary interactions, that can chain to each other up to 7 times, giving an important depth of description of the interactions. It is therefore difficult to describe one line of the file in BioPAX because it refers to several interactions influencing each other. The python converter has a *DescribeInteractions* function which parses the content of the *InteractionActions* column and decomposes the chain into elementary interactions. These elementary interactions are successively described by the function by attributing one unique identifier to each interaction, and a specific BioPAX class to each interaction type.

```
def DescribeInteractions(InteractionInList, InteractionList, InteractionID, InteractionNames, DicoInteractions, PmidsList, PreviousInteraction="name"):
    # The interaction affects the activity of a protein : we attribute the Control BioPAX class
    if "activity" in InteractionInList:
        if PreviousInteraction != "name": # if there is a previous reaction in the chain, write its name
            dest_file.write("{} bp3:controlled ctd:control-{} .\n".format(PreviousInteraction, InteractionID))
        for ReactionName in InteractionNames:
            if "activity" in ReactionName: # associate the corresponding reaction name
                if ReactionName in DicoInteractions: # check if the current interaction has already been described and stored in the dico
                    InteractionID = DicoInteractions[ReactionName]
                    dest_file.write("ctd:control-{} bp3:name \"{}\" .\n".format(InteractionID, ReactionName))
                else:
                    dest_file.write("ctd:control-{} bp3:name \"{}\" .\n".format(InteractionID, ReactionName))
                    DicoInteractions[ReactionName] = InteractionID
        dest_file.write("ctd:control-{} rdf:type bp3:Control .\n".format(InteractionID)) # Class Control
        dest_file.write("bp3:Control rdfs:subClassOf bp3:Interaction .\n") # Subclass of Interaction
        dest_file.write("ctd:control-{} rdf:resource mesh:{} .\n".format(InteractionID, ChemicalID))
        dest_file.write("ctd:control-{} bp3:controller mesh:{} .\n".format(InteractionID, ChemicalID))
        dest_file.write("ctd:control-{} bp3:controlType \"{}\" .\n".format(InteractionID, InteractionInList))
        dest_file.write("ctd:control-{} bp3:participant mesh:{} .\n".format(InteractionID, ChemicalID))
        dest_file.write("ctd:control-{} bp3:participant gene:{} .\n".format(InteractionID, GeneID))
        dest_file.write("ctd:control-{} bp3:dataSource ctd:provenance-{} .\n".format(InteractionID, ProvenanceID)
        ControlledReactionID = uuid.uuid4() # ID of the controlled reaction
        dest_file.write("ctd:control-{} bp3:controlled ctd:controlled-reaction-{} .\n".format(InteractionID, ControlledReactionID))
        dest_file.write("ctd:controlled-reaction-{} bp3:name 'activity of {} protein' .\n".format(ControlledReactionID, GeneSymbol))
        dest_file.write("ctd:controlled-reaction-{} rdf:type bp3:Control .\n".format(ControlledReactionID))
        if "increases^activity" in InteractionInList:
            dest_file.write("ctd:controlled-reaction-{} bp3:controlType 'ACTIVATION' .\n".format(ControlledReactionID))
        elif "decreases^activity" in InteractionInList:
            dest_file.write("ctd:controlled-reaction-{} bp3:controlType 'INHIBITION' .\n".format(ControlledReactionID))
        dest_file.write("ctd:controlled-reaction-{} bp3:dataSource ctd:provenance-{} .\n".format(ControlledReactionID, ProvenanceID))
        for pmid in PmidsList: # Add the PubMed identifier(s)
            dest_file.write("ctd:control-{} bp3:xref pubmed:{}.\n".format(InteractionID, pmid))
            dest_file.write("pubmed:{} rdf:type bp3:Xref .\n".format(pmid))
            dest_file.write("bp3:Xref rdfs:subClassOf bp3:UtilityClass .\n")
        PreviousInteraction = "ctd:control-{}".format(InteractionID) # this interaction becomes the PreviousInteraction for the next interaction
        if InteractionInList != InteractionList[-1]: # if this interaction is not the last one, we continue
            NextInteractionID = uuid.uuid4()
            InteractionListReduced = InteractionList[1:] # dicard the interaction that we described from the list
            DescribeInteractions(InteractionListReduced[0], InteractionListReduced, NextInteractionID, InteractionNames, \
                DicoInteractions, PmidsList, PreviousInteraction) # Recursivity, the function calls itself
```

Figure 9. Extract of the *DescribeInteractions* function which describes chain interactions. This portion of code describes interactions affecting the activity of a protein.

The converter attributes the `bp3:Control` class *affects reaction*, *increases/decreases activity* and *affects folding* interactions. The `bp3:TemplateReactionRegulation` class is assigned to *increases/decreases expression* interactions and the `bp3:Catalysis` class is assigned to *affects binding* interactions. Other interaction types are described as instances of the `Interaction` class. I chose to first describe these 5 interactions because these are the ones involving the chemical `"10074-G5"` that we took as example. I used this example to build a structure of the converter and make sure that it worked before converting the entire file.

3.2.3 Conversion of some simple leaves

The description of elementary interactions gives a set of RDF triples which can be considered as simple leaves in the final oriented graph of the final BioPAX file. Figure 10 shows one of these leaves.

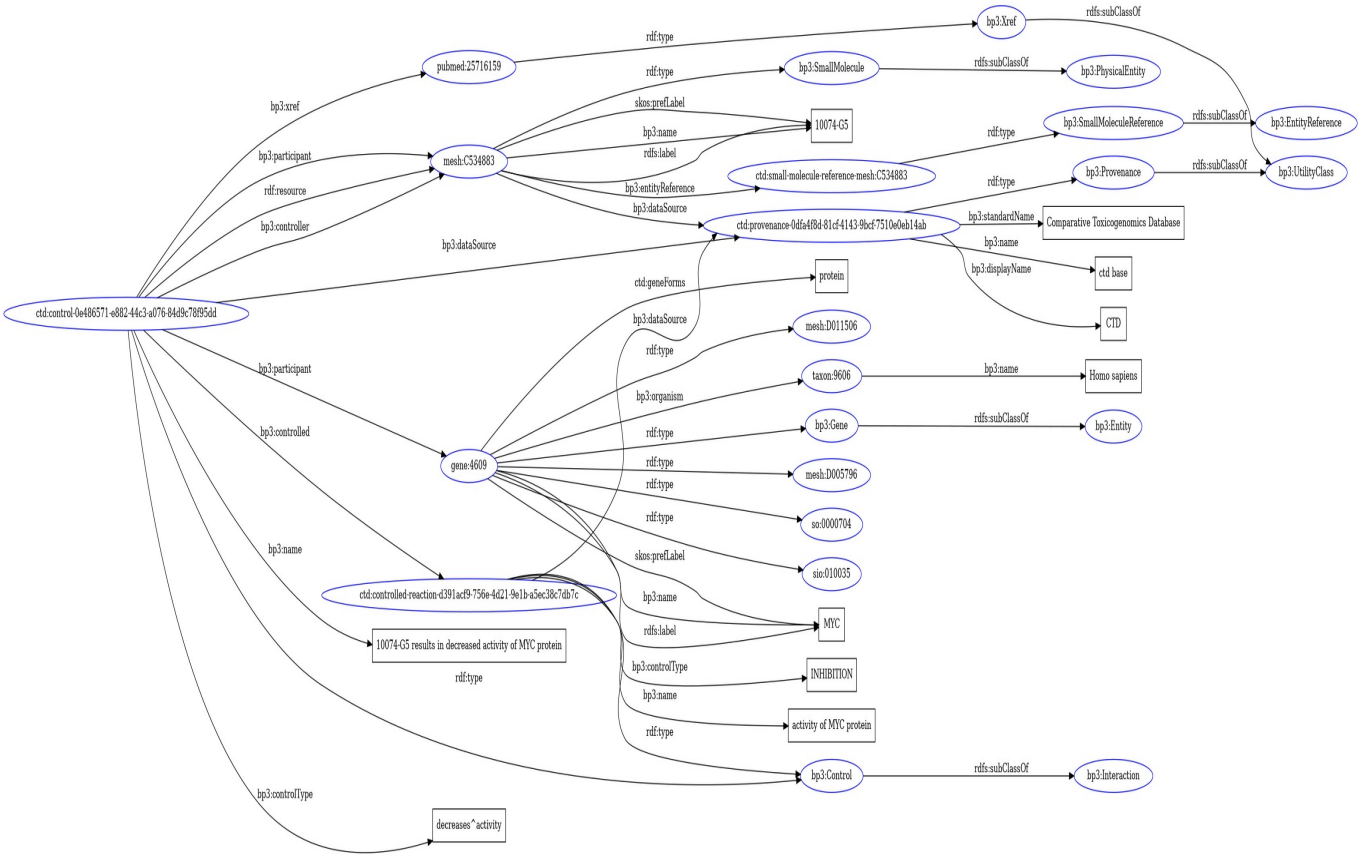

Figure 10. Chemical-gene elementary interaction `"10074-G5 results in decreased activity of MYC protein"` described in BioPAX as RDF triples in an oriented graph. Instances are represented by blue ellipses; classes by boxes containing URIs and strings by boxes containing the value.

This graph represents RDF triples describing an elementary interaction between the chemical `10074-G5` and the gene `MYC`. This interaction influences the activity of the `MYC` protein. The *DescribeInteractions* function attributed it the `bp3:Control` class. Since it decreases the activity of MYC, the `bp3:controlType` of the interaction is `INHIBITION`.

3.2.4 Extraction of the graph of dependencies between the CTD interactions

In addition to describe elementary interactions, the *DescribeInteractions* function successively links chained reactions in the final BioPAX file. Since the interactions are linked by causal relationships and are chained to each other as shown in the *InteractionActions* column of the CTD file in Figure 7, the *DescribeInteractions* function is recursive. It calls itself multiple times while parsing the content of the *InteractionActions* column. The final BioPAX file of chemical-gene interactions of the CTD results in an oriented graph containing simple leaves linked to each other. The final graph of chemical-gene interactions of the CTD is a dependency graph between interactions representing a total of 23, 616, 142 RDF triples.

3.2.5 Comparaison with the 2019 BioPAX file

We compared the new CTD BioPAX file to the one available on PathwayCommons in order to quantify the information gain concerning chemical-gene interactions.

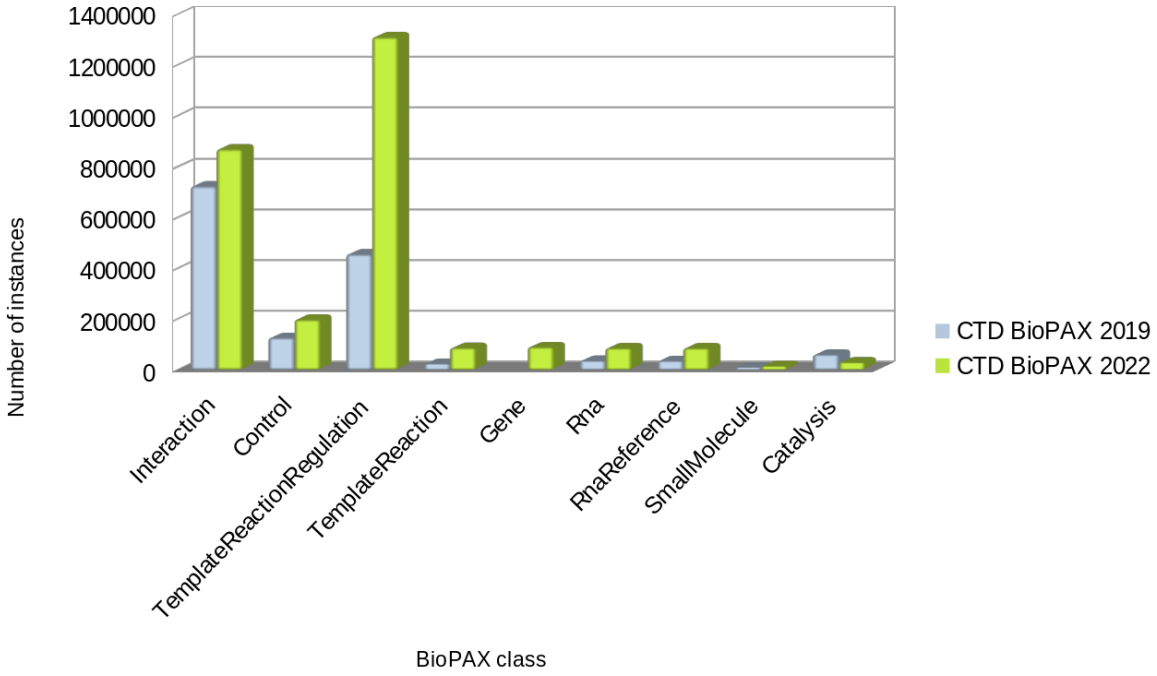

Figure 11. Comparaison of the information in the CTD BioPAX files from 2019 and the newly-built chemical-gene interactions BioPAX file of the CTD (2022). The number of instances of each BioPAX class used to build the file are counted. Each of these instances represents one elementary chemical-gene interaction described using this class.

As shown in Figure 11, the new file is enriched in all classes except the `bp3:Catalysis` one when compared with the 2019 BioPAX file of the CTD. Importantly we noted a very significative increase in instances of the `bp3:TemplateReactionRegulation` class.

4 Discussion

4.1 Analysis of the new CTD BioPAX file

The final BioPAX file of chemical-gene interactions of the CTD that we obtained thanks to the python converter is huge. It contains approximately 130 million of lines for a size of 9,5 GB. It was thus quite difficult to query with SPARQL and I had to subdivize it in two files to make the queries.

The SPARQL queries revealed that the new file contains significantly more information than the file from 2019. Indeed, it contains three times more RDF triples than the 2019 BioPAX file, and thus describes much more chemical-gene interactions. In particular, there is a sharp increase in the number of `TemplateReactionRegulation` interactions described. Since we associated this class to interactions having an impact on gene expression, the new file seems to support the description of three times more of these interactions. This global observation can be explained on one hand by the fact that the CTD had been enriched since 2019 by new chemical-gene interactions. We wanted to compare the two files using the same CTD input data, but for this we needed to get the CTD chemical-gene interaction file from 2019. But this was not possible because the CTD does not store and provide archive files. Also the comparaison of the two files can not be based on the same input data. On the other hand the python converter seems to describe much more interactions. The converter used to build the 2019 BioPAX file seems to not decompose chain reactions into elementary ones and then miss a lot of information about interactions linked by causal relationships.

4.2 Possible improvements for the converter

The python converter works well however it assigns a specific BioPAX class to only 5 types of chemical-gene interactions for the moment. All interactions are described in the file as instances of the Interaction class, but only 5 types are associated to a more precise BioPAX class. By doing this for all the 134 elementary interactions, and by describing these interactions using correct BioPAX properties, it would be possible to precisely describe it all in the final file. By identifying BioPAX subclasses that could describe more precisely the interaction we can further improve the coverage of the converter. Indeed, the more we go down in the BioPAX hierarchy, the more we can describe our interactions with precise terms. In the current file, many

interactions are associated to the `Control` class which is the most global to describe reactions that influence others. We could search for subclasses of `Control` that could describe more precisely the nature of these interactions. Netherless, this would require an important work to study in detail each interaction type in order to determine the most appropriate BioPAX class for each interaction. There are also still some improvements to bring to the description of complex interactions since the most complex ones (high number of chained reactions) are not perfectly described and structured. Finally, the global code of the converter could be optimized in order to be more efficient and less-resource intensive.

4.3 Next steps of the analysis

To analyze CTD data further and answer our biological question, the next step is to build large scale networks of chemical-gene interactions described in BioPAX using the CadBiom application. We plan to do this over the next weeks before the end of the internship. Then it will be possible to query these networks using reachability queries. It will allow to identify the trajectories to activate genes involved in liver fibrosis and next to extract the chemicals involved in these trajectories. Based on this analysis we will identify chemical signatures that have an impact on the expression of genes associated to chronic liver diseases.

5 Conclusion

The CTD-to-BioPAX python converter that I developed allows to build a BioPAX file containing the CTD chemical-gene interactions that is up-to-date and well-described in accordance to Semantic Web standards. Its information is standardized and is computer-readable, allowing it to be queried. In a second part of this project, it will allow to determine exposure factors associated to chronic liver diseases. Considering the wide range of exposures that can be involved, this integrative biology approach makes sense and seems to be the most appropriate to answer this biological question. Ultimately, it could be possible to adapt the converter in order to translate the whole content of the CTD into BioPAX and to be able to query other parameters.

Bibliography

[1] Geoffroy Andrieux, Michel Le Borgne, and Nathalie Théret. “An integrative modeling framework reveals plasticity of TGF-β signaling”. en. In: *BMC Systems Biology* 8.1 (Dec. 2014), p. 30. ISSN: 1752-0509. DOI: 10.1186/1752-0509-8-30. URL: https://bmcsystbiol.biomedcentral.com/articles/10.1186/1752-0509-8-30 (visited on 06/24/2022).

[2] Laura E. Armstrong and Grace L. Guo. “Understanding Environmental Contaminants’ Direct Effects on Non-alcoholic Fatty Liver Disease Progression”. en. In: *Current Environmental Health Reports* 6.3 (Sept. 2019), pp. 95–104. ISSN: 2196-5412. DOI: 10.1007/s40572-019-00231-x. URL: https://doi.org/10.1007/s40572-019-00231-x (visited on 06/14/2022).

[3] Juliane I. Beier and Gavin E. Arteel. “Environmental exposure as a risk-modifying factor in liver diseases: Knowns and unknowns”. en. In: *Acta Pharmaceutica Sinica B* 11.12 (Dec. 2021), pp. 3768–3778. ISSN: 22113835. DOI: 10.1016/j.apsb.2021.09.005. URL: https://linkinghub.elsevier.com/retrieve/pii/S2211383521003476 (visited on 06/14/2022).

[4] Tim Berners-Lee, James Hendler, and Ora Lassila. “The Semantic Web: A New Form of Web Content That is Meaningful to Computers Will Unleash a Revolution of New Possibilities”. In: *ScientificAmerican.com* (May 2001).

[5] H. Chen, T. Yu, and J. Y. Chen. “Semantic Web meets Integrative Biology: a survey”. en. In: *Briefings in Bioinformatics* 14.1 (Jan. 2013), pp. 109–125. ISSN: 1467-5463, 1477-4054. DOI: 10.1093/bib/bbs014. URL: https://academic.oup.com/bib/article-lookup/doi/10.1093/bib/bbs014 (visited on 06/13/2022).

[6] A. P. Davis et al. “Comparative Toxicogenomics Database: a knowledgebase and discovery tool for chemical-gene-disease networks”. en. In: *Nucleic Acids Research* 37.Database (Jan. 2009), pp. D786–D792. ISSN: 0305-1048, 1362-4962. DOI: 10.1093/nar/gkn580. URL: https://academic.oup.com/nar/article-lookup/doi/10.1093/nar/gkn580 (visited on 05/16/2022).

[7] Allan Peter Davis et al. “Comparative Toxicogenomics Database (CTD): update 2021”. en. In: *Nucleic Acids Research* 49.D1 (Jan. 2021), pp. D1138–D1143. ISSN: 0305-1048,

1362-4962. DOI: 10.1093/nar/gkaa891. URL: https://academic.oup.com/nar/article/49/D1/D1138/5929242 (visited on 05/16/2022).

[8] Emek Demir et al. “The BioPAX community standard for pathway data sharing”. en. In: *Nature Biotechnology* 28.9 (Sept. 2010), pp. 935–942. ISSN: 1087-0156, 1546-1696. DOI: 10.1038/nbt.1666. URL: http://www.nature.com/articles/nbt.1666 (visited on 05/16/2022).

[9] Beate I. Escher, Heather M. Stapleton, and Emma L. Schymanski. “Tracking complex mixtures of chemicals in our changing environment”. en. In: *Science* 367.6476 (Jan. 2020), pp. 388–392. ISSN: 0036-8075, 1095-9203. DOI: 10.1126/science.aay6636. URL: https://www.science.org/doi/10.1126/science.aay6636 (visited on 05/17/2022).

[10] Sven Fillinger et al. “Challenges of big data integration in the life sciences”. In: *Analytical and Bioanalytical Chemistry* 411.26 (Oct. 2019), pp. 6791–6800. ISSN: 1618-2642, 1618-2650. DOI: 10.1007/s00216-019-02074-9. URL: http://link.springer.com/10.1007/s00216-019-02074-9 (visited on 06/24/2022).

[11] Sabina Leonelli. “The challenges of big data biology”. en. In: *eLife* 8 (Apr. 2019), e47381. ISSN: 2050-084X. DOI: 10.7554/eLife.47381. URL: https://elifesciences.org/articles/47381 (visited on 06/24/2022).

[12] Ashish Sharma and Shivaraj Nagalli. “Chronic Liver Disease”. eng. In: *StatPearls*. Treasure Island (FL): StatPearls Publishing, 2022. URL: http://www.ncbi.nlm.nih.gov/books/NBK554597/ (visited on 05/16/2022).

[13] Pierre Vignet et al. “Discrete modeling for integration and analysis of large-scale signaling networks”. en. In: *PLOS Computational Biology* 18.6 (June 2022). Ed. by Denis Thieffry, e1010175. ISSN: 1553-7358. DOI: 10.1371/journal.pcbi.1010175. URL: https://dx.plos.org/10.1371/journal.pcbi.1010175 (visited on 06/22/2022).

[14] Qier Wu et al. “Capturing a Comprehensive Picture of Biological Events From Adverse Outcome Pathways in the Drug Exposome”. In: *Frontiers in Public Health* 9 (Dec. 2021), p. 763962. ISSN: 2296-2565. DOI: 10.3389/fpubh.2021.763962. URL: https://www.frontiersin.org/articles/10.3389/fpubh.2021.763962/full (visited on 05/16/2022).

6 Appendixes

6.1 Presentation of the host structure

This first year bioinformatics Master's degree internship is held in collaboration between the DYLISS team (Dynamics, Inference and Logics for BIological Systems and Sequences) of the IRISA (Institute for Research in Computer Science and Randomized Systems) and the Dymec team (Microenvironment Dynamics and Cancer) of the IRSET (Institute for Environmental and Occupational Health Research).

The DYLISS team is a bioinformatics team working on data management and representation in life science using Semantic Web technologies. It is part of the IRISA laboratory, which is a mixed research unit in computer science and new information technologies. The IRISA is attached to the Inria Rennes-Atlantic Brittany center (French national research institute for digital science and technology).

The Dymec team works on the impact of the exposome on the occurence of chronic liver diseases. The team possesses a research axis dedicated to integrative biology and modeling. It is part of the Irset laboratory, a mixed reasearch unit of the Inserm, which focuses on environmental and work-related risks.

6.2 Personal assessment of the internship

This internship allowed me to be part of the DYLISS research team for three and a half months, while having weekly interactions with the Dymec research team at the Irset. I realized the importance of cooperation and communication between researchers from different backgrounds when working at the interface between two disciplines that are computer science and biology.

This internship has also made me more autonomous in my work. It taught me to be well organized in the management of a project over several months. I also had the opportunity to present my work to the members of the DYLISS and Dymec teams. It made me work on my communication skills and taught me to adapt my speech depending on whether I was addressing to biologists or computer scientists.

This project also gave me a good overview of scientific research and its hazards. I learned how to overpass problems and to adapt to the difficulties encountered during this internship, especially computer problems.

Finally, I would also say that this internship made me aware of the importance of producing a quality code that is reusable. This is a good practice that will be useful in all types of projects that I will have to manage in the future.

Abstract

Chronic liver diseases are long-term pathologies affecting a wide range of people nowadays. They can be caused by multiple factors including viral infection, alcohol overconsumption, metabolic disorders or genetic factors to which are added the impact of our lifestyles and environmental exposures. These diseases have become an important public health problem, but the impact of our environment is difficult to determine give the diversity of factors involved and the dynamics of these exposures. This internship project aims to identify environmental exposure factors associated to the occurrence of chronic liver diseases. The Comparative Toxicogenomics Database (CTD) is used as an information source. This database describes exposure events at several levels of associations (chemical-gene, chemical-disease and gene-disease associations). The conversion of the CTD tabulated chemical-gene interactions file into the BioPAX format (Biological Pathway Exchange) thanks to the development of a Python CTD-to-BioPAX converter allows to standardize it in accordance to Semantic Web standards. The obtained BioPAX file can be used as an input of the CadBiom application developed by the DYLISS and Dymec teams in order to build large-scale biological dynamic networks based on guarded transitions. The generated models could be analyzed thanks to reachability queries in order to identify environmental exposures causal signatures associated to the occurrence of chronic liver diseases.

Keywords : chronic liver diseases, exposome, Semantic Web, large-scale networks, BioPAX

Résumé

Les maladies chroniques du foie sont des pathologies de longue durée qui touchent aujourd'hui un grand nombre de personnes. Elles peuvent être causées par de multiples facteurs comme une infection virale, une surconsommation d'alcool, des troubles métaboliques ou des facteurs génétiques, auxquels s'ajoute l'impact de nos modes de vie et des expositions environnementales. Ces maladies sont devenues un problème de santé publique important, mais l'impact de notre environnement est difficile à déterminer étant donné la diversité des facteurs impliqués et la dynamique de ces expositions. Ce projet de stage vise à identifier les facteurs d'exposition environnementaux associés à l'apparition des maladies chroniques du foie. La base de données CTD (Comparative Toxicogenomics Database) est utilisée comme source d'information. Elle décrit les événements d'exposition à plusieurs niveaux d'associations (associations produit chimique-gène, produit chimique-maladie et gène-maladie). La conversion du fichier tabulé d'interactions produit chimique-gène de la CTD au format BioPAX (Biological Pathway Exchange) grâce au développement d'un convertisseur Python CTD vers BioPAX permet de le standardiser selon les standards du Web sémantique. Le fichier BioPAX obtenu peut être utilisé comme entrée de l'application CadBiom développée par les équipes DYLISS et Dymec afin de construire des réseaux biologiques dynamiques à grande échelle basés sur des transitions gardées. Les modèles générés pourront être analysés grâce à des requêtes d'atteignabilité afin d'identifier les signatures causales d'expositions environnementales associées à l'apparition de maladies chroniques du foie.

Mots clés : maladies chroniques hépatiques, exposome, Web Sémantique, réseaux à grande échelle, BioPAX