Universitat Oberta de Catalunya

An Ontology for the *NMR pulse sequence* concept

Autor: Pau Nolis Fañanas

Tutor: Felipe Geva Urbano

Professor: Ferran Prados Carrasco

Grau Enginyeria Informàtica

Itinerari Enginyeria del programari

Web Semàntica

04/01/2020

FITXA DEL TREBALL FINAL

Títol del treball:	*An Ontology for the NMR pulse sequence concept*
Nom de l'autor:	*Pau Nolis*
Nom del col·laborador/a docent:	*Felipe Geva Urbano*
Nom del PRA:	*Ferran Prados Carrasco*
Data de lliurament (mm/aaaa):	*01/2020*
Titulació o programa:	*Grau Enginyeria Informàtica*
Àrea del Treball Final:	*Web Semàntica*
Idioma del treball:	*Anglès*
Paraules clau	*NMR, pulse sequence, Ontology*

Resum del Treball:

La Ressonància Magnètic Nuclear (RMN) és una tècnica analítica utilitzada àmpliament per l'anàlisi química de mostres de diferent procedència. En aquest treball es presenta una ontologia per al concepte "seqüència de polsos per RMN" en experiments d'estat líquid. L'ontologia es centra en la classificació jeràrquica dels experiments segons les seves dimensions i el(s) nucli(s) d'adquisició. Altre coneixement rellevant que gestiona són les característiques principals dels experiments, la seva referència bibliogràfica, el seu codi font i la llibreria on el podem trobar. Per a la creació de l'ontologia s'ha utilitzat el programari Protégé i s'ha utilitzat una metodologia que ha combinat un procés lineal amb un d'iteratiu.

La finalitat de l'ontologia és que un usuari pugui utilitzar-la i treure'n informació útil. Per a facilitar la interacció amb l'ontologia s'ha dissenyat un prototip d'aplicació que permet la introducció de noves seqüències de polsos, fomentant així el seu creixement. L'aplicació s'ha escrit en codi Java utilitzant les llibreries d'Apache Jena. Per la implementació d'una mínima interfície gràfica s'han utilitzats les llibreries de Window Builder. Finalment, també es proporcionen exemples amb casos d'ús de com un usuari pot extreure informació rellevant de la ontologia mitjançant consultes SPARQL en un servidor local (Fuseki-Jena).

El treball demostra la utilitat de tenir una ontologia que modeli el coneixement del domini "seqüència de polsos de RMN" per tal de facilitar-ne l'extracció d'informació. Cal remarcar que és considera un projecte embrionari amb força potencial de creixement, doncs, només s'ha explorat a mode de *"proof of concept"* una part del domini del concepte.

Abstract:

Nuclear Magnetic Resonance (NMR) is an analytical technique widely used for chemical analysis of samples from different origins. This study presents an ontology for the concept "NMR pulse sequence" in liquid state experiments. The ontology focuses on the hierarchical classification of experiments according to their dimensions and the acquisition nucleus. Other relevant knowledge that manages is the main features of the experiments, their bibliographic reference, their source code or the library where we can find it. For the creation of the ontology protégé software has been used and the design methodology has combined a linear process with an iterative one. The main purpose of the ontology is to extract useful information from it.

To facilitate the user interaction with the ontology, an application prototype has been designed that allows the introduction of new pulse sequences, thus encouraging its growth. The application has been written in Java code using the libraries Apache Jena. For the implementation of a minima graphical user interface a Window Builder libraries are used. Furthermore, examples are also provided with use cases, showing how a user can extract relevant information from the ontology using SPARQL queries on a local server (Fuseki-Jena).

The study demonstrates the usefulness of having an ontology that shapes the knowledge of the domain "NMR pulse sequence" in order to facilitate the extraction of information. It should be noted that it is considered an embryonic project with high growth capabilities, thus only a part of the concept domain has been explored as a "proof of concept".

License

Content

Acronym List

Ordered alphabetically

CHMO	**C**hemical **M**ethods **O**ntology
COSY	**CO**rrealation **S**pectroscop**Y**
DEPT	**D**istorsionless **E**nhanced **P**olarization **T**ransfer
DTD	**D**ocument **T**ype **D**efinition
GUI	**G**raphic **U**ser **I**nterface
FOAF	**F**riend **O**F **A** **F**riend
HSQC	**H**eteronuclear **S**ingle **Q**uantum **C**orrelation
HMBC	**H**eteronuclear **M**ultiple **B**ond **C**orrelation
HTML	**H**yper**T**ext **M**arkup **L**anguage
IFSERF	**I**sotope **F**iltered **S**electiv**E** **R**e**F**ocusing
IUPAC	**I**nternational **U**nion of **P**ure and **A**pplied **C**hemistry
JSON	**J**ava **S**cript **O**bject **N**otation
LOD	**L**inked **O**pen **D**ata
MRI	**M**agnetic **R**esonance **I**maging
NMR	**N**uclear **M**agnetic **R**esonance
nmrCV	**nmr** **C**ontrolled **V**ocabulary
nmrML	**nmr** **M**ark-up **L**anguage
NMR-STAR	**NMR** – **S**elf-defining **T**ext **A**rchival and **R**etrieval
NOESY	**N**ucler **O**verhauser **E**ffect **S**pectroscop**Y**
OBI	**O**ntology for **B**iomedical **I**nvestigations
OWL	**O**ntology **W**eb **L**anguage
RDF	**R**esource **D**escription **F**ramework
RDFS	**RDF**ramework **S**chema
RIF	**R**ule **I**nterchange **F**ormat
SPARQL	**S**PARQL **P**rotocol and **R**DF **Q**uery **L**anguage
SKOS	**S**imple **K**nowledge **O**rganization **S**ystem
ssNMR	**s**olid-**s**tate **N**uclear **M**agnetic **R**esonance

SWRL	**S**emantic **W**eb **R**ule **L**anguage
TFG	**T**reball **F**i **G**rau (catalan language)
TOCSY	**TO**tal **C**orrelation **S**pectroscop**Y**
URI	**U**niform **R**esource **I**dentifier
UTF	**U**nicode **T**ransformation **F**ormat
XDS	**X**ML **D**ata **S**ervice
XML	e**X**tensible **M**arkup **L**anguage
W3C	**W**orld **W**ide **W**eb **C**onsortium
WWW	**W**orld **W**ide **W**eb

Figures index

Table's index

1. Introduction

1.1. NMR data storage, a problem to solve

Since the early days of NMR by I.I.Rabi (Nobel prize in Physics, 1944), Nuclear Magnetic Resonance (NMR) knowledge has not stopped growing, little by little, year per year. Developments have been introduced every year, both, from a technological point of view, with better equipment and from a methodological point of view, with highly robust methods. Nowadays, NMR spectroscopy is considered one of the most powerful analytical techniques in many fields. For chemists, NMR is an indispensable common tool for the structural characterization of small molecules. In medicine, Magnetic Resonance Imaging (MRI) is a key non-invasive technique for diagnosis. For biochemists, NMR it is mainly used for the structure and dynamics characterization of proteins and methods are usually grouped and called bioNMR methods. In material science research field, NMR is used in a specific branch of the technique named solid-state NMR (ssNMR).

Despite its robustness as analytical technique, there is an important question still to be resolved in NMR, which is how all the daily generated data is stored and managed in the context of web semantic era. Today is still common to report NMR data in a traditional way, which is the listing and tabulation of NMR data parameters, mainly, chemical shifts and its correlations, signals multiplicities, signal integration, scalar couplings, residual dipolar couplings, inter atomic distances derived restraints, dihedral angle derived restraints, diffusion coefficients and relaxation rates. In practice, it is encountered that each scientific journal specifies a different format to do so, despite IUPAC publications with formal recommendations [1] [2] [3] [4]. Also problematic is the often encountered lack of details in the description of the experimental section, which difficult the exact reproducibility of the data for non-experimented users.

In fact, data storage problem according to the time knowledge is a common problem. For instance, ~30 years ago the STAR file format (Self-defining Text Archive and Retrieval) appear to handle crystallographic data [5]. The format adapted to NMR data and evolved continuously along years [6] [7] [8] and nowadays is still under development, last update release appeared last year [9]. Another approach is used in nmrML [10] an open XML-based exchange and storage format. Recently, NMReDATA appeared [11], which uses a tag format based approach to store NMR data. It is important to highlight that each of the above mentioned approaches different target research fields. Thus, NMR-STAR is focused on biological systems; nmrML is originally designed to handle metabolomics data and NMReDATA for archiving relevant parameters for the structural determination of small molecules, both synthetic and natural products. For that reason it seems reasonable to think that all of them will evolve in parallel and will achieve enough consistency to be considered formats of reference in NMR world. That will definitely happen when software vendors incorporate such formats and users find an automated and easy way to save and store its data.

From the semantic web point of view, the main problem with the current data storage practice is the non-readability by computers. Another added problem is that the raw data often included in repositories, or stored in local data bases, is completely useless if not linked to the assignment of the chemical structure of molecules. In summary,

key questions regarding long term data storage, persistence, sustainability, usability, accessibility, verification, consistency, experimental reproducibility are still without a clear and unique protocol in NMR community.

1.2. The semantic web approach, NMR ontologies

Nowadays, with the arrival of the semantic web, the development of a NMR ontology, or several ontologies dedicated to targeting different NMR concepts is highly desirable. That would help to handle all the knowledge acquired along all those years of NMR success and even more important, its relations map and linked data. It is easy to imagine that a project on this topic will very be dynamic and in continuous development, so such ontology will be a never ending iterative process, growing in parallel to NMR instrumentation advances, NMR methods improvements or even new NMR concepts, still unimaginable today. To date, three ontologies containing NMR concepts already coexist, CHMO, nmrCV and NMR-star, each with its own particularities and covering different topics. Find below a short summary of its purpose.

CHMO: Chemical Methods Ontology

Created and maintained by Colin Bachelor. Last update 14 Nov, 2019 [12]. That ontology intends to cover all the existent analytical methods and the equipment description that allow collecting chemical information. It is intended to be complementary to the Ontology for Biomedical Investigations (OBI) [13]. The task sounds huge, and nowadays it contains about ~3000 classes. In this context we can find the concept NMR and some related terms.

nmrCV: nuclear magnetic resonance Controlled Vocabulary

This is a specific NMR ontology for describing NMR experiments and acquisitions parameters in the field of metabolomics, which is the study of metabolites in bio-fluid and tissue samples. The nmrCV [14] forms part of the so called nmrML project (nmr Mark-up Language), see Figure 1 below.

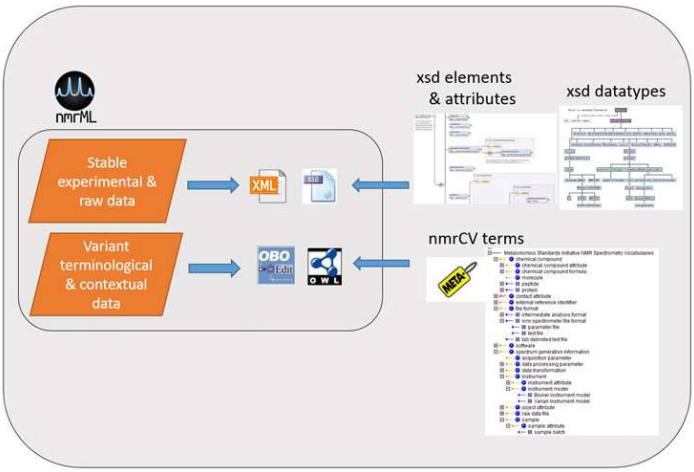


Figure 1: nmrML Project schema showing how nmrCV is related. *Source* https://nmrml.org/schema/

The purpose of nmrML project is:

- Allow 1D and 2D NMR spectra and raw data to be easily shared in a vendor agnostic manner
- Record enough information about an NMR spectrum acquisition to allow for further processing of the raw spectrum without referring to the original vendor files.
- The data format should reference the original files for the sake of posterity and in the case where original vendor specific information is required.
- The data format should be flexible and allow for multiple use cases of NMR experiments.
- The data format should be easy for developers to understand and integrate into software.

This work was included into the European project called COSMOS [15] (Coordination Standard in MetabOlomicS project) financed with €2 million euros. This ontology contains around ~600 classes and was announced through the publication in 2017 [10]. The article concludes:

"Overall the nmrML specification and the expandable nmrCV will allow for a detailed standardized description of NMR workflow functionalities. The use of nmrML in workflow tools and the reuse of containerized workflow components in recombinable app libraries will allow NMR data processing to be more traceable and rerunnable in different (local or cloud) environments".

NMR-STAR: Nuclear Magnetic Resonance – Self defining text archival and retrieval

As stated in its last release publication announcement [9], the NMR-STAR v3.2 ontology provides an extensive controlled vocabulary for the description of NMR spectroscopic studies of biological systems. The ontology includes the description of experiments, the data generated, and the derived results such as molecular structures, dynamics, and functional properties. NMR-STAR is the archival format used by the Biological Nuclear Magnetic Resonance data Bank (BMRB), the international repository of biomolecular NMR data [16] and an archive of the Worldwide Protein Data Bank [17].

Once those three ontologies are presented lets introduce in the next section the NMR pulse sequence concept. This is a term that can be encountered in those ontologies and will be the main focus of this project.

1.3. The NMR pulse sequence concept

It is beyond the scope of this work to describe the theoretical details on how NMR works. Nevertheless, to put this project into a context, it should be explained that a NMR experiment is performed by irradiating with radiofrequency pulses a sample. The precise description along time of how that is performed is called an NMR pulse sequence, which is a code that contains the exact details on radiofrequency pulses, irradiation frequency, field strength, duration, phase, shape and delays between different pulses needed for the experiment. Nowadays the number of NMR pulse sequences is enormous. Most of them are included in libraries at the instrument vendor's software, but many others

are just described in research articles, with several specialized research groups around the world affording each year new contributions. It is important here to highlight that different libraries are encountered depending on the research field of study (biomolecules, solid-state, liquids or imaging). In this project we will focus on NMR pulse sequences for high resolution liquid experiments used for the study of small-to-medium sized molecules [18] [19] [20]. Now, let inspection the NMR pulse sequence class in the previously described ontologies.

CHMO:

The class hierarchy is shown in Figure 2

Thing -> process -> planned-process -> nuclear magnetic resonance pulse sequence

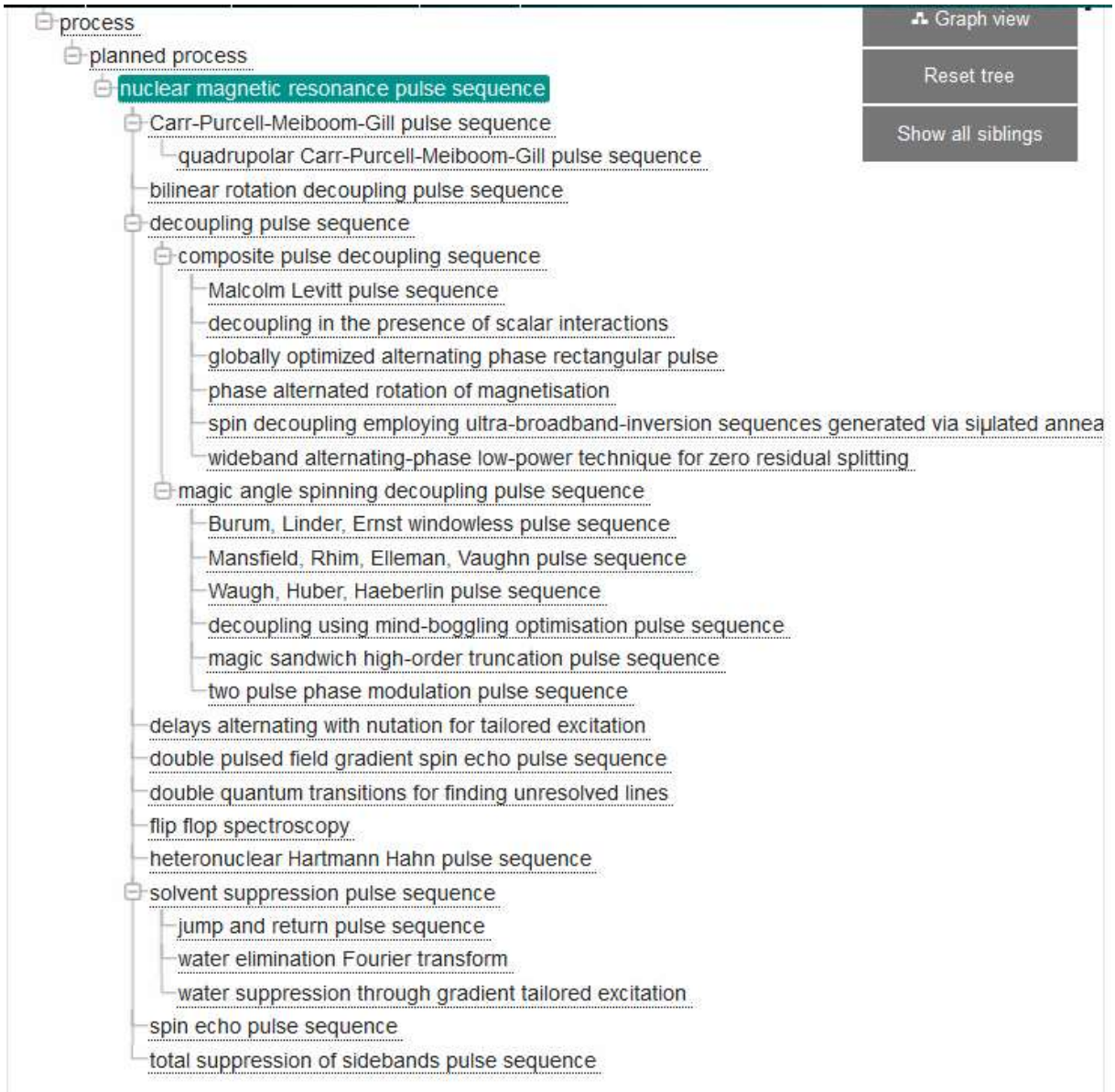


Figure 2: NMR pulse sequence node expansion at CHMO ontology. *Source:* *https://www.ebi.ac.uk/ols/ontologies/chmo*

CHMO ontology contains 25 pulse sequences which it seems not a bad number for a non-specialized NMR ontology; remember that the domain of this ontology is all the analytical techniques a chemist can use to obtain data about its sample. Even I would say 25 is a high number, because standard and daily routine NMR pulse sequences used by chemist can be around 10. However, and surprisingly, are not in the ontology. Thus, concepts like, standard hard pulse (for 1H and 13C), or basic experiments like DEPT, COSY, TOCSY, NOESY, HQSC and HMBC pulse sequences are not covered. Other remarkable strange treat is that some of them are grouped into two categories decoupling sequences and solvent suppression sequences, but the rest are loose. Also, other incoherence is the different applications scope of the NMR pulse sequences encountered. Thus, there are pulse sequences which its application is liquid-state NMR and other which is solid-state NMR, but this is not classified anyway and should be a remarkable treat. In my opinion, this classification should be mandatory, not only because the application is different, also because pulse sequence design and construction building blocks from a technical point of view are different as well.

nmrCV:

The class hierarchy is shown in Figure 3

Thing->entity->continuant->generically dependent continuant-> parameter->pulse parameter->NMR pulse sequence

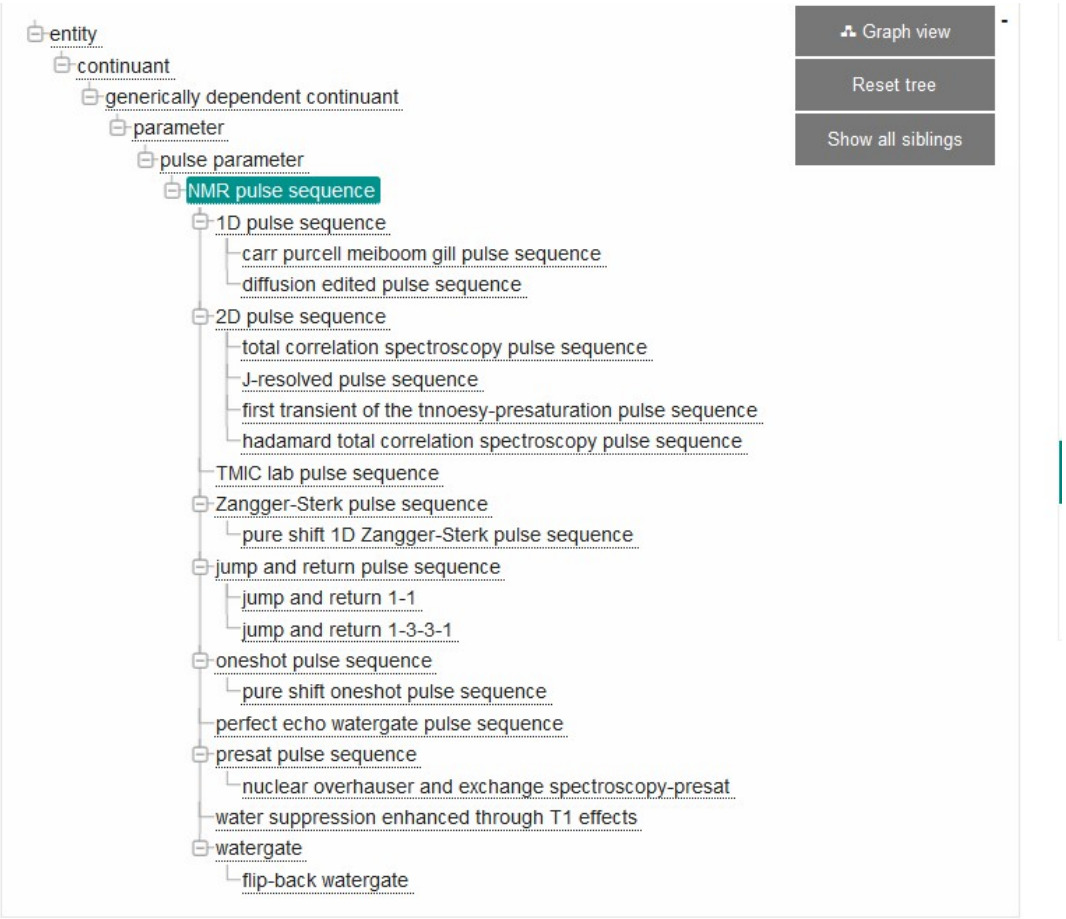


Figure 3: NMR pulse sequence node expansion at nmrCV ontology. Source: https://www.ebi.ac.uk/ols/ontologies/nmrCV

Curiously and considering that nmrCV is an NMR specialized ontology, only 15 pulse sequences are incorporated, so less than encountered in a non-NMR specialized ontology, like CHMO ontology. Nevertheless, such low number agrees with the most used pulse sequences in the NMR metabolomics which is the scope of the ontology. However, in my opinion, the hierarchy tree is not well constructed. It is seen a division between 1D and 2D pulse sequences, which is perfect and in agreement with the main classification encountered in specialized books, but then we can see that only few of them are classified into those categories and the rest are outside, which is a kind of inconsistent design.

NMR-STAR:

The class hierarchy is shown in Figure 4

NMR-STARDictionary -> experimental details -> NMR spectrometer expt

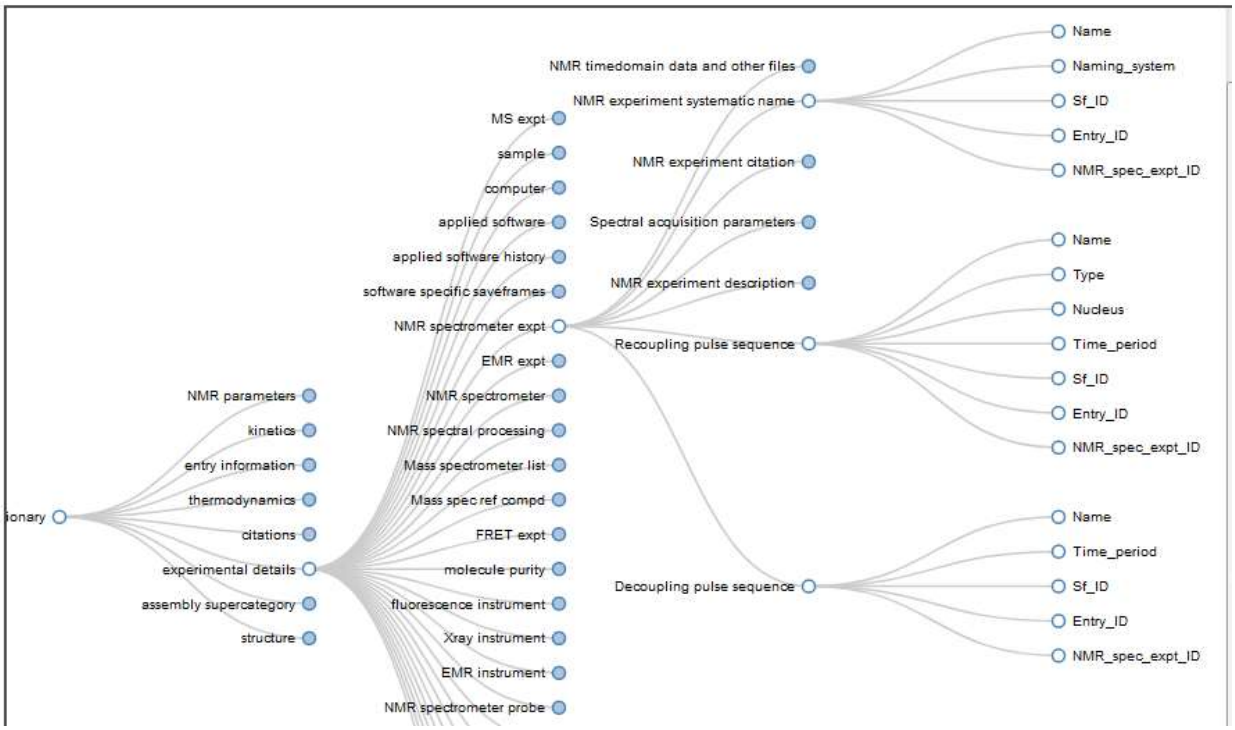


Figure 4: NMR-STAR dictionary expansion. *Source:* http://www.bmrb.wisc.edu/dictionary/interactive/dict_tree.shtml

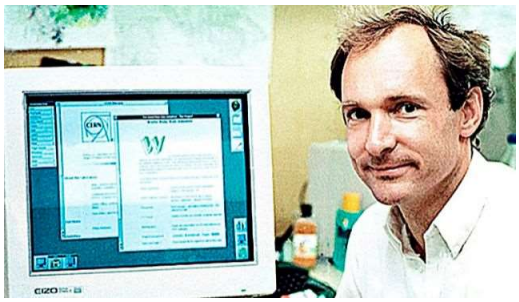
Here is shown how a concrete NMR pulse sequence is described using just two categorizations for Recoupling and Decoupling pulse sequences and another generic for NMR systematic name. So, the ontology is designed with the purpose that the user introduces the pulse sequence by filling some attributes into these categories. This would have the problematic that any user can describe the same pulse sequence in many ways.

Once an overview and inspection of the three available ontologies containing the term NMR pulse sequences is made, it can be stated the necessity to rethink the design of this class to better categorize it. A main treat of the described ontologies is that the scope is very large, so get a deeper insight into all classes would increase exponentially its complexity and has to be done little by little. In this project we will work on that node expansion.

2. State-of-the art

2.1. Semantic Web Concept

The vision or concept of the semantic web was stated by the father of the World Wide Web, Tim Berners-Lee in 1999 [21]:



""I have a dream for the Web [in which computers] become capable of analysing all the data on the Web – the content, links, and transactions between people and computers. A "Semantic Web", which makes this possible, has yet to emerge, but when it does, the day-to-day mechanisms of trade, bureaucracy and our daily lives will be handled by machines talking to machines. The "intelligent agents" people have touted for ages will finally materialize"

Figure 5: Tim Berners-Lee picture

Nowadays, the semantic web is still under construction and constantly in evolution. It is considered as the extension, not a replacement, of the current "web of documents" and is often referred as the "web of data". The main goal that is expected to be afforded by the semantic web compared to the current web is the capability to be fully readable by machines. That treat will be achieved if the web is enriched with significance, but not only for humans like is today, for machines as well. In fact, nowadays, the coexistence of the "two webs" is already happening, with many applications based on semantic web standards appearing every year (See Figure 6). Some of the most known projects using a semantic web approach are:

- DBpedia, a project aiming to extract structured content from the information created in the Wikipedia [22].
- GeoNames, a user editable geographical database [23].
- FOAF (Friend Of A Friend), is a machine-readable ontology describing persons and their relations [24].
- LOD (Linked Open Data), is structured open data which is interlinked with other data [25].

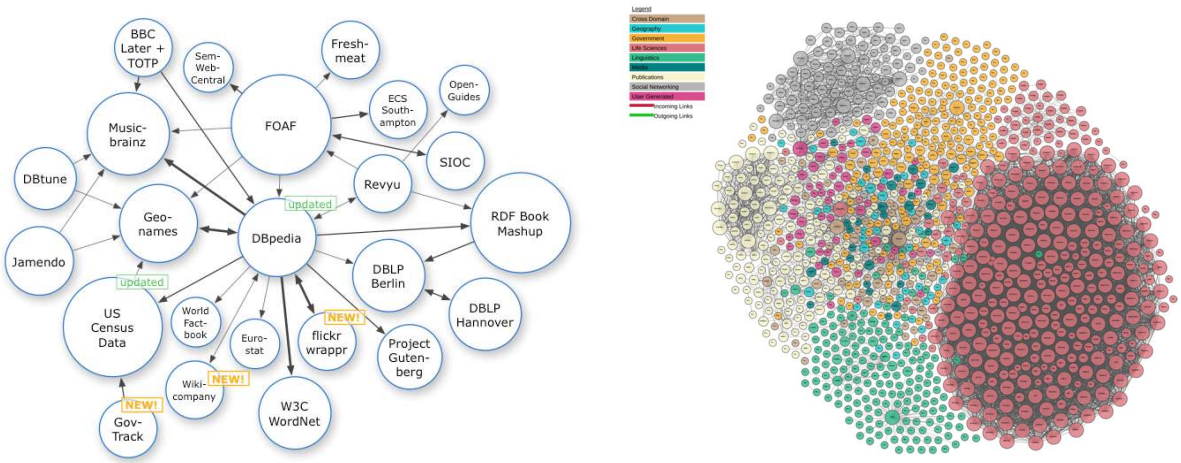


Figure 6: Datasets in the Linking Open Data project, as of September 2007 (left) and 2017 (right)

2.2. Ontologies

Ontologies are nuclear in the semantic web. The reason is that ontologies allow expressing semantic aspects of the information resources. Gruber in 1993 [26] define an ontology as a specification of a representational vocabulary for a shared domain of discourse. Roughly speaking an ontology provides the common vocabulary, meaning and relations in a knowledge area. Ontologies are formalized using classes or concepts, attributes, relations, functions axioms and instances.

Class or concept	The key domain ideas wanted to formalize and are usually organized in taxonomies.
Attributes	Properties that describe the concepts. Can have determined values.
Relations	Interaction between concepts of the same domain (structure building).
Functions	A type of relation that identifies an element using a function of other elements.
Rules	Logical sentences that have to be followed
Instances	Individuals or occurrences in a real world of a concept.
Axioms	True expressions declared upon relations.

Table 1: Ontologies components

One of the most important points of ontologies is the inferred new knowledge, not initially stated, acquired through the heritance of concepts, the defined relations, functions, rules and axioms.

2.3. Semantic web architecture

The World Wide Web Consortium (W3C) [27] has approved standards to build the semantic web architecture (Figure 7).

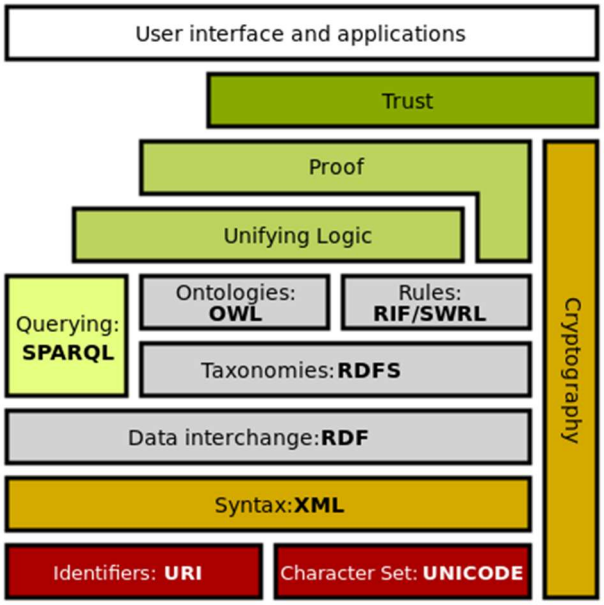


Figure 7: Semantic Web Stack. *Source:* https://en.wikipedia.org/wiki/File:Semantic_web_stack.svg

Notice that while the bottom and middle standard protocols of the architecture are clear and well-defined the upper layers are still in ways of development and are not yet standardized. Next chapter will get insights into each of the semantic web architecture building blocks, usually known as web semantic technologies. The description will go from bottom to top layers.

LAYER	TECHNOLOGY
Codification and Localization	UNICODE and URI
Syntaxis	XML, XML Schema and Namespaces
Description and structure	RDF, RDF Schema
Queries	SPARQL
Logic Integration	OWL, RIF, SWRL
Cryptography	Digital signature and encryption

Table 2: Layers and technologies of the web semantic architecture

2.4. Semantic Web technologies

2.4.1. Character Set: UNICODE

UNICODE [28] is an international standard for the codification, representation and handling of characters and is the adopted by many recent technologies like XML to publish information in the web. Three codification formats are defined depending on the bits used to code a character UTF-8, UTF-16 and UTF-32, where UTF stands for Unicode Transformation Format. It is necessary to define which of the codifications format will be used in the rest of layers, and that is usually indicated in the first line of an XML, for example:

```
<?xml version =”1.0” encoding=”UTF-8”?>
```

2.4.2. Identifiers: URI (IRI, URL, URN)

Unified Resource Identifier URI, as its name indicates, is an identifier of a specific resource, like a page, a book, a document, emails… in fact, anything having identity and it can be abstract or physical resource. This identifier it is very important because it will refer to the resource from which is wanted to obtain or give information. It is important to comment that there are other identifiers which often cause confusion IRI, URL and URN.

- IRI, refers to Internationalized Resource Identifier, which allows UNICODE characters in the identifier and has a defined correspondence to URI
- URL, refers to Uniform Resource Locator and is a subgroup of URIs and contains a localization of the resource. Thus, all URL are URI but not all URI are URL.
- URN, refers to Uniform Resource Name and is a subgroup that allow identify resources but without implying availability.

An URI has a well-defined generic syntax [29]:

```
URI = scheme:[//authority]path[?query][#fragment]
```

Authority, query and fragment components are optional. If exist, authority component divides into three subcomponents:

```
authority = [userinfo@]host[:port]
```

Figure 8, shows a diagram for URI syntax:

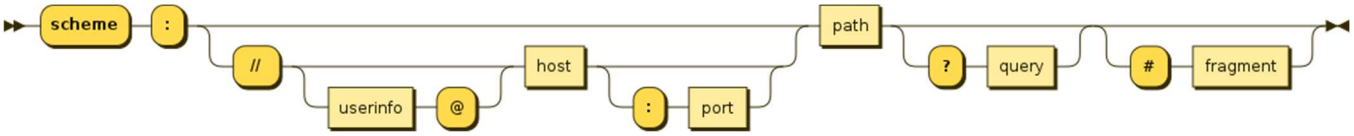


Figure 8: URI syntax diagram. *Source:* https://en.wikipedia.org/wiki/Uniform_Resource_Identifier#cite_note-FOOTNOTERFC_3986,_section_32005-9

Some popular schemes are: http, https, ftp, mailto, filedata and irc. Some URI examples are shown below:

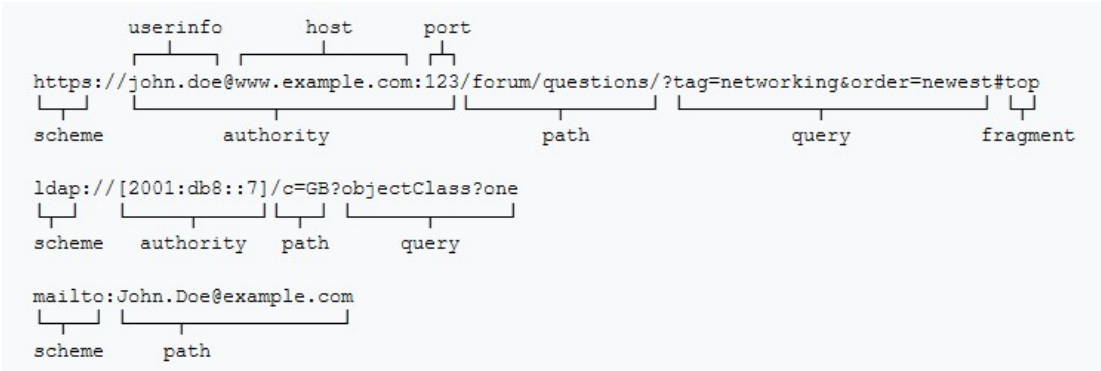


Figure 9: URI syntax examples. *Source:* https://en.wikipedia.org/wiki/Uniform_Resource_Identifier#cite_note-FOOTNOTERFC_3986,_section_32005-9

2.4.3. Syntax: XML

While the current web is based on HyperText MarkUp Language (HTML), the semantic web is based on eXtensible Markup Language (XML). Former focuses on the presentation of the data and is format driven. Second, focuses on transfer of the data and is content driven. In the example below it is clearly seen that while HTML tags refer to the format of the document, the XML tags refer to the content described inside:

HTML example

```
<!DOCTYPE html>
<html>
<head>
<title> Page title </title> </head>
<body>
<hl> First Heading</hl>
<p> First paragraph.</p>
</body></html>
```

XML example

```
<?xml version="1.0>
<contact>
<name>Pau Nolis</name>
<email>pnolis@uoc.edu </email>
<telephone>803522545</telephone>
</contact>
```

XML, is a mark-up language with the main characteristic that is both human and machine-readable. It is defined by a set of rules declared in schemas. The oldest schema language for XML was the Document Type Definition (DTD). A newer schema language, W3C defined is the XML schema. Today are available a large variety of schemas depending on its purposes like maths, finance, music playlists [30]…

2.4.4. **Data interchange: RDF**

Resource Description Framework RDF, lies on the heart of the semantic web because provides the foundation for publishing and linking the data. The important point about RDF is that affords interoperability of the data at a semantic level, enhancing XML capacities which are just a syntactic level.

RDF is based in the expression or sentence of type *subject-predicate-object,* commonly known as triple. The subject is the resource to describe, the predicate is a property or a relation of the resource and the object is the value of such property or relation. A standard way of representation of such triples is using graphs, Figure 10.

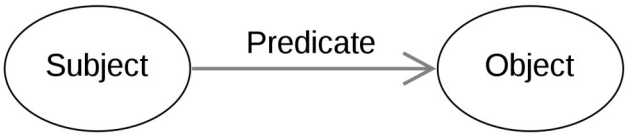


Figure 10: graph representation of RDF triples. *Source:* *https://en.wikipedia.org/wiki/Resource_Description_Framework#/media/File:Basic_RDF_Graph.svg*

It is important to remark that to unequivocally define subjects, predicates and objects RDF uses URIs. Although it is allowed that subjects or objects being blank nodes or also called anonymous resources. The object could also be a Unicode string literal. Table 3 summarize these RDF statements:

subject	URI, blank node
predicate	URI
object	URI, blank node, Unicode string literal

Table 3: RDF entities and possible assignments

See a graph example below extracted from W3C, Figure 11.

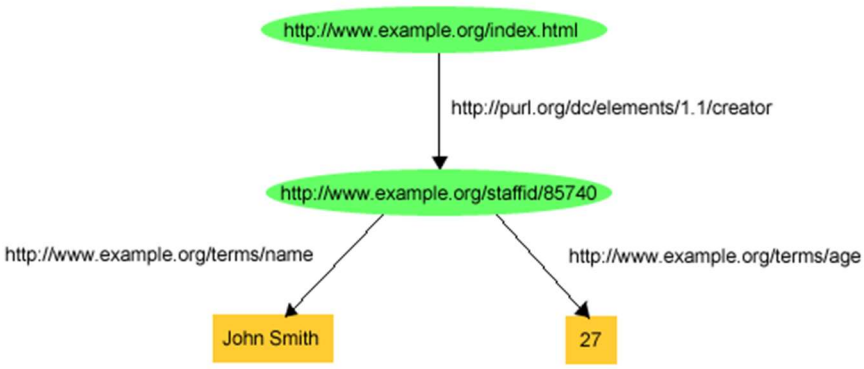


Figure 11: Graph example using URIs for subjects and predicates and string literals for objects. *Source:* *https://www.w3.org/TR/rdf-primer/#typedliterals*

There are some W3C standards to express RDF serialization. Most commons are:

- RDF/XML, based on a XML syntax to define graphs [31].
- N-Triples, which uses simple text lines [32].
- Turtle, an extension of N-triple including syntactic constructions, more compacted and legible definitions [33].
- TriG, an extension of Turtle that allow multiple graph representation in a single file [34].
- N-Quads, an extension of N-Triples that allow a fourth element, a graph name. [35]
- JSON-LD, based on JSON [36].

2.4.5. Taxonomies: RDFS

RDF Schema (RDFS) provides a data-modelling vocabulary for RDF data, being an extension of the basic RDF vocabulary. That is with RDFS can describe properties and classes about RDF resources with a certain semantic that allow establish hierarchies between those properties and classes and define some properties. Common rdfs definitions are shown in Table 4[37]:

Classes	Properties
rdfs:Resource	rdfs:range
rdfs:Class	rdfs:domain
rdfs:Literal	rdfs:type
rdfs:Datatype	rdfs:subClassOf
rdfs:langString	rdfs:subPropertyOf
rdf:HTML	rdfs:label
rdf:XMLLiteral	
rdf:Property	

Table 4: common rdfs classes and properties

See below an example of graph representation using rdf/rdfs vocabulary (Figure 12). Notice that while rdf define the inner properties of triples, rdfs interconnect them affording a richer semantic domain.

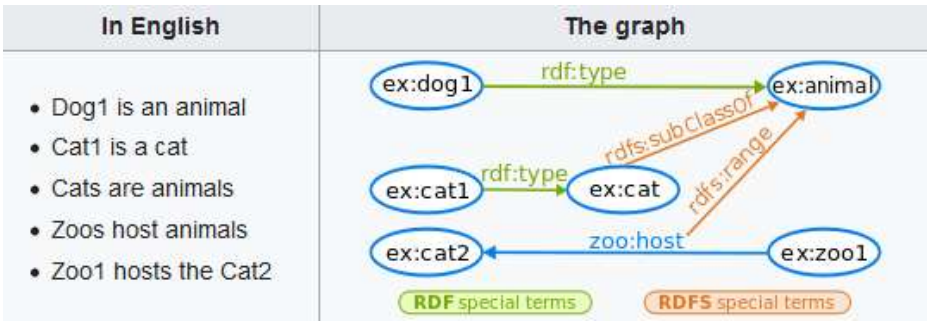


Figure 12: rdf/rdfs graph example. *Source:* https://en.wikipedia.org/wiki/RDF_Schema

2.4.6. Ontologies: OWL

As stated in the web semantic architecture chapter, ontologies are in an upper layer respect RDF/RDFS, is considered the logical layer (Figure 13). Thus, Ontology Web Language (OWL) is an extension of RDFS developed by W3C designed to represent things and its relations affording extra semantics to data. This language can be exploited by computer programs to generate implicit knowledge due to its logic-based nature.

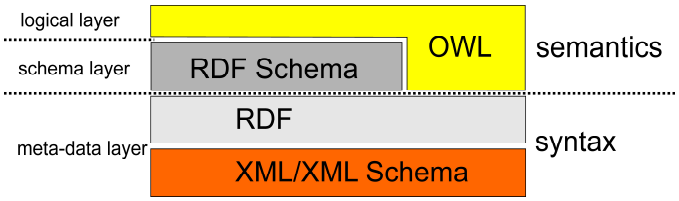


Figure 13: layered scheme for the semantic web. OWL is considered the logical layer at the semantic division.

The current version is OWL2 [38]. Three different levels of OWL language are common, depending on the problem necessities (Table 5).

OWL Lite	Simple restrictions oriented to hierarchical classification.
OWL DL (Description Logic)	Maximal expressivity and guarantee that the results are calculated in a finite time.
OWL Full	Maximal expressivity but without guarantee that the results are calculated in a finite time.

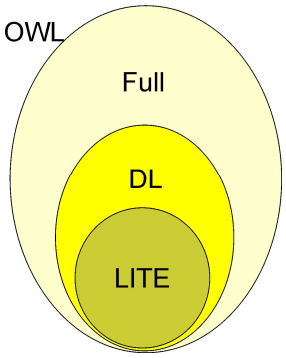


Table 5: owl language levels

OWL can be written in many syntax forms. Most common are OWL2, OWL2/XML, RDF/XML or Manchester syntax. Above find the example for the definition of Tea ontology in each syntax style.

OWL2

```
Ontology(<http://example.com/tea.owl>
  Declaration( Class( :Tea ) ))
```

OWL2/XML

```
<Ontology ontologyIRI="http://example.com/tea.owl" ...>
   <Prefix name="owl"
IRI="http://www.w3.org/2002/07/owl#"/>
   <Declaration>
     <Class IRI="Tea"/>
   </Declaration>
 </Ontology>
```

Manchester

```
Ontology: <http://example.com/tea.owl>
Class: Tea
```

RDF/XML

```
<rdf:RDF ...>
    <owl:Ontology rdf:about=""/>
    <owl:Class rdf:about="#Tea"/>
</rdf:RDF>
```

2.4.7. Rules: RIF/SWRL

Ontologies can implement rules that have to be fulfilled in a domain. There are two common languages; the Rule Interchange (RIF) Format is a W3C recommendation [39] and the Semantic Web Rule Language (SWRL) which is a proposal also from W3C [40]. Most basic rules are based on one of the most logical expression of type: IF/THEN.

2.4.8. Queries: SPARQL

SPARQL is the language to perform queries to the semantic web, using RDF triples as data source. Basically SPARQL can perform four types of queries:

SELECT	Search according with an expression and returns the coincident data.
CONSTRUCT	Generates new RDF triples from recovered data according to the specified pattern.
ASK	Indicates a true/false value depending on the existence in the graph of the specified pattern.
DESCRIBE	Is used to extract sentences from a graph.

A SPARQL query example is shown below extracted from [41]:

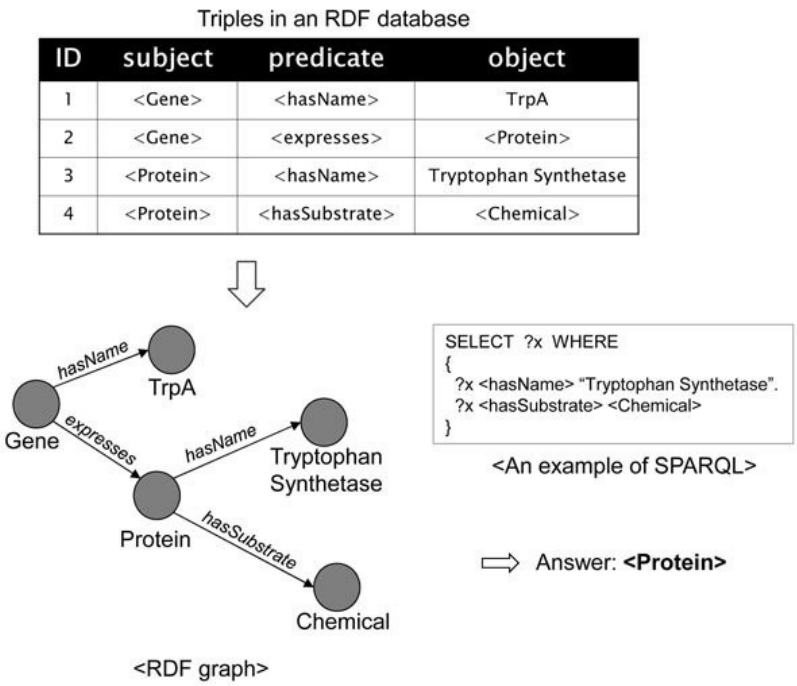


Figure 14: Example of SPARL query. *Source reference [41]*

3. Objectives

3.1. Main objective

Design and implementation of an ontology prototype for the NMR pulse sequence concept. The ontology will classify NMR pulse sequence instances into a class hierarchy and will describe its main features. The used vocabulary will be technical and directed to be useful for NMR users.

3.2. Derivate objectives

From the main objective are derived partial objectives, which in fact are common goals to any ontology design in a specific domain of research [42]:

- To share common understanding of the structure of information among people or software agents
- To enable reuse of domain knowledge
- To make domain assumptions explicit
- To separate domain knowledge from the operational knowledge
- To analyse domain knowledge

3.3. TFG general objectives and competencies

It is worth to mention that this semantic web work forms part of a teaching plan in UOC and the project has the goal to guide the student to acquire the following objectives competencies:

- Study the basic concepts of the semantic web
- Know the structure and representation of some languages of representation of information on the Web
- Know how to structure information through ontologies
- Know the organization and structure of the representation of data linked to the Web
- Knowing how to make queries in SPARQL on semantic repositories

3.4. Personal Interest and Motivation

My personal interest into this project is to get a deeper knowledge of the web semantic and its structure. I would like to fully understand concepts like owl, xml, rdf, rdfs, xds, uri, xms, sparql… For sure, to be focused on a web semantic project will give me the opportunity to read about those concepts and settle them.

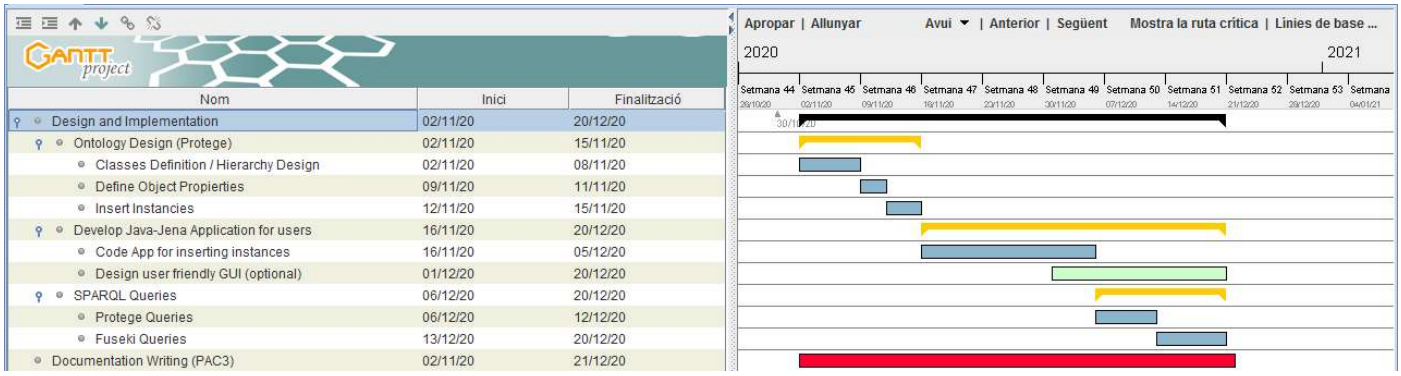
Then I decided to link it to my professional world. Briefly explained, I was graduated on chemistry (2002) and my specialization is on NMR spectroscopy research (PhD 2007), concretely in the development of new NMR pulse sequences for the structural characterization of small molecules. That linkage between the project and my professional career basically fulfils two goals. Former, the needless to spend extra time to get inside a new unknown topic, which is always very time consuming. Second point is that I found very interesting the direct possibility to self-design an ontology based on my professional field and explore thinking on possible applications from this project.

4. Project schedule

The project is strictly performed using the deadlines marked in the project program:

- Delivery 1 project plan (28/9/20)
- Delivery 2 project mandate and plan (20/10/2020)
- Delivery 3 design and implementation (21/12/2020)
- Delivery 4 Memory writing (4/1/2021)
- Delivery 5 Video presentation (10/1/2021)

Figure 15, show Gantt diagram with the scheduled dates of the Design and Implementation part for that project presented in PAC2. Basically, three parts were defined, which are marked yellow: OWL-Ontology design, User-Application, SPARQL-Queries. Those are subdivided in parts, marked blue for mandatory operations and soft green for an optional GUI interface of the application, which in an initial approach will communicate with the user through console typing. Along all those steps a writing documentation is considered, which is marked red and is considered a continuous parallel process.



Nom	Inici	Finalització
Design and Implementation	02/11/20	20/12/20
Ontology Design (Protege)	02/11/20	15/11/20
Classes Definition / Hierarchy Design	02/11/20	08/11/20
Define Object Propierties	09/11/20	11/11/20
Insert Instancies	12/11/20	15/11/20
Develop Java-Jena Application for users	16/11/20	20/12/20
Code App for inserting instances	16/11/20	05/12/20
Design user friendly GUI (optional)	01/12/20	20/12/20
SPARQL Queries	06/12/20	20/12/20
Protege Queries	06/12/20	12/12/20
Fuseki Queries	13/12/20	20/12/20
Documentation Writing (PAC3)	02/11/20	21/12/20

Figure 15: Gantt diagram for the Design and Implementation part of the project

It is important to remark that in practice the timing did not exactly match the planning and some variations occurred, but the important milestones has been successfully reached and developed on time. The more important variations are marked below:

- In the ontology design, data and annotation properties were also added (not only object properties).
- The developed app for users was designed from the beginning using a GUI and not only optional as was initially scheduled.
- The SPARQL queries were tested in Protégé and also using the FUSEKI server. In this memory only screen captures using Fuskei server are reported and Protégé prior test omitted because are considered redundant.
- Some extra days were needed in order to correct some fails detected on the java application from my tutor.

5. Ontology design and implementation

5.1. Introduction

The research methodology is based on a prototype ontology design and further check test to verify its correct operation. The ontology development could follow several strategies; see for instance [42] [43] [44]. Some of the most cited methodologies are Methontology [45], On-To-Knowledge [46], Text2Onto [47], SENSUS-based [48], Grüniger and Fox [49] and Ontology development 101 [42]. The common point among them is the coincidence on several steps to handle.

Step 1: Determine the Domain and Scope (objective)
Step 2: Consider Reusing Existing Ontologies
Step 3: Enumerate Important Terms
Step 4: Define the classes and the class hierarchy
Step 5: Define the properties of Classes-Slots
Step 6: Define the facets of the slots
Step 7: Create instances

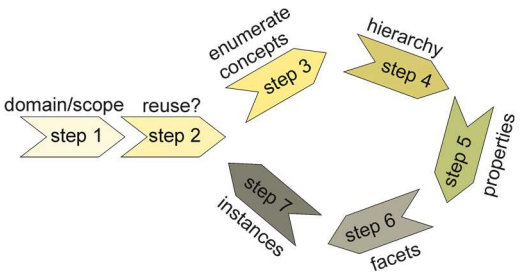


Figure 16: step-by-step ontology design scheme

In the herein project, two former steps have been done linearly. The rest will be implemented using and iterative life-cycle.

5.2. Implementation tool: Protégé

Protégé software [50] (desktop v 4.3.0 build 304) has been used to implement the NMR pulse sequence ontology. As defined on its website [51] Protégé is:

> *"A free, open-source ontology editor and framework for building intelligent systems. Protégé is supported by a strong community of academic, government, and corporate users, who use Protégé to build knowledge-based solutions in areas as diverse as biomedicine, e-commerce, and organizational modeling."*

5.3. Step 1: Determine the Domain and Scope of the ontology

Objective(s) have been presented at Chapter 3 of this document. Now, let's get a deeper insight into domain and scope, therefore let's answer the questions that would define the limits of the scope [42]:

- *What is the domain of the ontology?* Basic NMR pulse sequences used in high-resolution liquid-state of small organic molecules (Figure 16, red square)

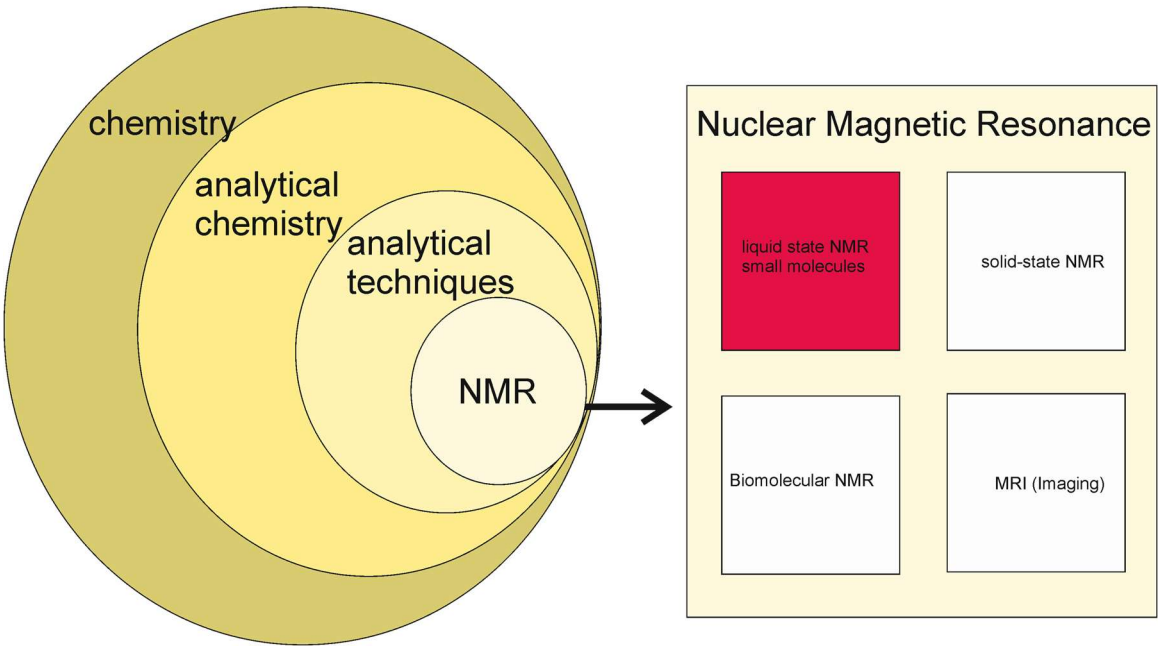


Figure 17: NMR pulse sequence domain (square red). Situation into chemistry parent domain.

- *For what we are going to use the ontology?* NMR pulse sequence classification and main treats assignments
- *For what types of questions the information in the ontology should provide answers?* Mainly the questions that the design will answer will be.
    - Which are the main treats of a given pulse sequence instance?
    - What pulse sequences instances have xxx treats?
    - What pulse sequences instances can I use to perform xxx Type experiments?
    - What pulse sequences instances exist in a given library?
- *Who will use the ontology?* Basically, users dedicated to perform NMR experiments, ranging from basic users to specialized spectroscopist.

5.4. Step 2: Reusing Existing Ontologies

For this project CHMO and nmrCV pulse sequences will be reused, but instead of being treated as classes, they will be treated as instances. It is important to remark that those will be not the only instances of the current designed ontology; also feeding with extra Bruker NMR pulse sequences will be done. It is important to remark that not all Bruker sequences will be introduced, as the project is conceived as a prototype, a representative set will be selected. The purpose of the selection will be fill with some examples most of the classes.

5.5. Step 3: Enumerate Important terms

This part has been handmade on real paper and will be distilled in steps 4 to 7.

5.6. Step 4: Define the classes and hierarchy

5.6.1. Main Classes

Three Main Classes will hang from Thing entity. Find below its description:

NMR pulse sequence: This concept is considered the heart of the ontology. From here a series of subclasses will emerge with the purpose of NMR pulse sequences categorization and classification. Later on, the criteria to do so will be fully explained. An NMR pulse sequences instance will fit into one of those categories. A deep classification in the tree will meant a more detailed classification is achieved, but a pulse sequence could be stated into an upper node and not in a “leave” class if its classification is not clear in it.

NMR pulse sequence feature: This concept is organizing the features (or main treats) of a pulse sequence. It can be related to a pulse sequence design treat or to a chemical analysis goal. Categorization into subclasses will also arise but not as a deeper as encountered into NMR pulse sequence hierarchy. The idea is just to list features into a first level folder.

NMR pulse sequence library: Here the instances of pulse sequence libraries will be listed. Here one could imagine as many as possible. The concept is analog to a book library, but instead of having books, each library contain NMR pulse sequences, so NMR pulse sequences can be repeated between libraries, the same way a book copy is found in many libraries. In this project nmrCV, CHMO and Bruker libraries will be used as examples, but any research group could insert its own library.

5.6.2. NMR pulse sequence class hierarchy

A first separation will be performed taking into account the dimensionality of the experiment achieved using a determined NMR pulse sequence. Basically, one dimension (1D) means one NMR frequency axis describes the spectrum and two Dimension (2D) means a two frequency axis spectrum is obtained. Next categorization is to separate experiments depending on the chemical nuclei under study. In 1D experiments mainly ^{1}H isotope is analyzed, but also a X isotope can be routinely studied (X most commonly being ^{13}C, ^{15}N, ^{29}Si and ^{31}P, although other could be possible). In 2D maps, we can deal with homonuclear correlations (usually ^{1}H vs ^{1}H) or heteronuclear correlations (^{1}H vs X). Those categories are the first two hierarchy levels of the NMR pulse sequence concept (see Figure 18 below).

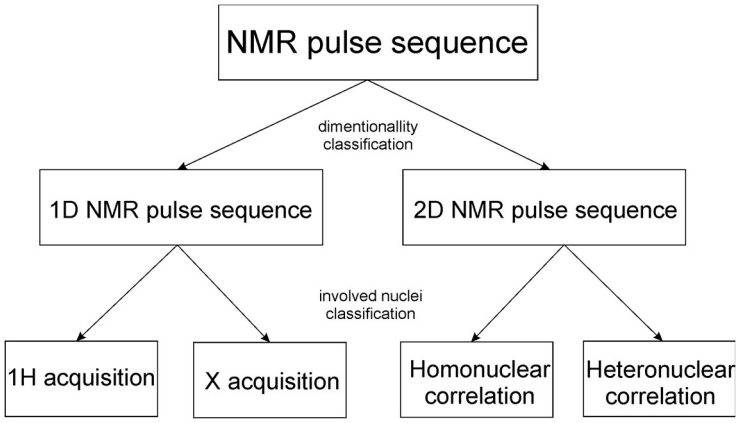


Figure 18: NMR pulse sequence hierarchy

A part from those main classifications, further hierarchy expansion is implemented which adds extra granularity at each node. Figure 19 shows Protégé screen captures showing those expansions:

Figure 19: Fully expanded NMR pulse sequence hierarchy

It is beyond the scope of the work to explain the details of what is the meaning of each class. Nevertheless, it is worth to mention that there are two classes which are considered “special” and need a mention. A “Decoupling pulse sequence” term is introduced which is sibling to 1D and 2D separation. In fact, a decoupling element forms part of a NMR pulse sequence but in the CHMO ontology is considered as being itself, so in order to maintain this term as encountered it is not removed. The other “special” term is “J-resolved” which is introduced sibling to Homonuclear and Heteronuclear experiments and is introduced because it is found nmrCV ontology, and is an experiment which is 2D and correlates 1H frequency vs. J coupling, but is not a Homonuclear neither Heteronuclear experiment and here it is considered a class itself.

5.6.3. NMR pulse sequence feature hierarchy:

This hierarchy is expanded into 15 categories. It is worth to mention that more other categories could be inserted or important for the ontology. For the purpose of this work, the important thing is just list some of the more relevant or characteristic and not generate a full list (Figure 20).

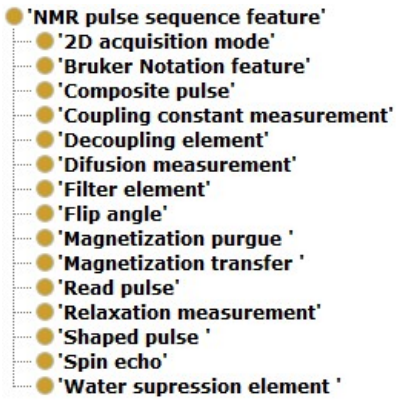


Figure 20: Fully expanded NMR pulse sequence feature hierarchy

5.7. Step 5: Define the properties of classes

5.7.1. Object Properties

The ontology is filled with 4 object properties, see Table 6 below:

Object Property	Domain	Range	Comment
Has feature	NMR pulse sequence	NMR pulse sequence feature	Indicates that a NMR pulse sequence has a concrete feature.
Is found in	NMR pulse sequence	NMR pulse sequence library	Indicates in which NMR pulse sequence library a NMR pulse sequence instance is found
Is analogue to	NMR pulse sequence	NMR pulse sequence	Indicates that two NMR pulse sequences encountered in different libraries are analog (maybe using different labeling)
Has Bruker Notation	NMR pulse sequence feature	Bruker Notation feature	Indicates that a feature instance has a correspondence with a notation in Bruker

Table 6: Object Properties

5.7.2. Data Properties

The ontology is filled with 4 data properties, see Table 7 below:

Data Property	Domain	Comment
Has Acronym	NMR pulse sequence	Used when a pulse sequence is already better known by an acronym
Has BRUKER code	NMR pulse sequence	Shows the BRUKER pulse sequence code for a given sequence
Has BRUKER name	NMR pulse sequence	Indicate the references where the pulse sequence is described (or a relevant element on it)
Has reference(s)	NMR pulse sequence	Indicate the references where the pulse sequence is described (or a relevant element on it)

Table 7: Data Properties

5.7.3. Annotation Properties

Annotation properties used to enhance ontology description are:

Labels: permits the human readable format of entities in the ontology.

Comments: used to describe some relevant feature of an entity.

versionInfo: used to describe the version of a pulse sequence code.

5.8. Step 6: Define the facets of the slots

The facets are the restrictions or role properties. As already set in the tables 1 and 2, the ontology properties use class domain and class range restrictions. Another added restrictions has been disjoints between sibling classes in the hierarchy, in a way that, an instance has a unique classification. Also, a symmetric characteristic is applied for the property “Is analogue to” is also used.

5.9. Step 7: Instancies

Finally, the ontology has been manually filled with 326 individuals distributed filling all of the hierarchy classes and using the entire object, data and annotation properties previously described.

5.10. Ontology metrics

Metrics summary for the ontology can be summarized using a Protégé functionality (Figure 21)

Metrics

Axiom	2641
Logical axiom count	1521
Class count	61
Object property count	5
Data property count	5
Individual count	326
DL expressivity	ALCHI(D)

Figure 21: Metrics summary for the NMR pulse sequence ontology

6. User interaction with the ontology

6.1. Data introduction to the ontology: Java GUI application

As a product, a prototype application that enables a user to introduce own NMR pulse sequences instances into the ontology is delivered (works in windows operating systems). The application is designed using Apache-Jena libraries [52] and Window Builder libraries [53], using Eclipse 2020 platform. Former is a free open source Java framework for building semantic web and linked data applications. Second is is composed of SWT Designer and Swing Designer and is an ideal tool to generate Java GUI applications. The application is delivered compressed into an executable *nmr_pulse_sequence.jar*. The project execution environment is 'JavaSE-1.8' (jdk1.8.0_131). The functionalities added to the prototype are detailed below:

6.1.1. Browsing to the ontology file

When launching the executable .jar file an initial window prompts (Figure 22). This window contains just a Browse button that permits search the *nmr_pulse_sequence.owl* file in your computer. The application launches this from the *main_gui.java* file

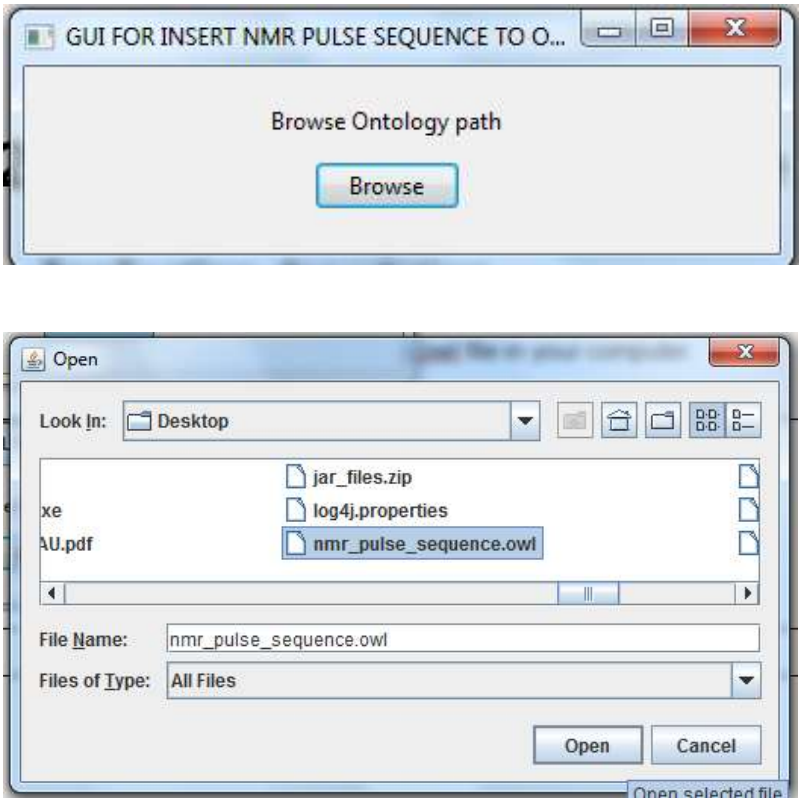


Figure 22: Browse window. Initial GUI window when launching the application. Selection of nmr_pulse_sequence.owl file

6.1.2. GUI main window

After selecting the .owl file a new window will prompt (Figure 23). This window is constructed in the *second_gui.java* file. This will contain several elements that will allow the interaction of the user with the ontology through calling the functions contained in the *ontology.java* file. Concretely, the elements are:

- (1) Text box for typing a pulse sequence name
- (2) Text area for adding a short description of the NMR pulse sequence if necessary
- (3) Text box for typing a new library name
- (4) Text area for adding pulse sequence code (Bruker format)
- (5) Text area for adding references
- (6) ScrollComposite area with an expandable *NMR pulse sequence library* tree
- (7) ScrollComposite area with an expandable *NMR pulse sequence* tree
- (8) ScrollComposite area with an expandable *NMR pulse sequence feature* tree
- (9) Button *add to a library*
- (10) Button *add new library*
- (11) Button *add to a category*
- (12) Button *add feature*
- (13) Button *add code*
- (14) Button *add references*

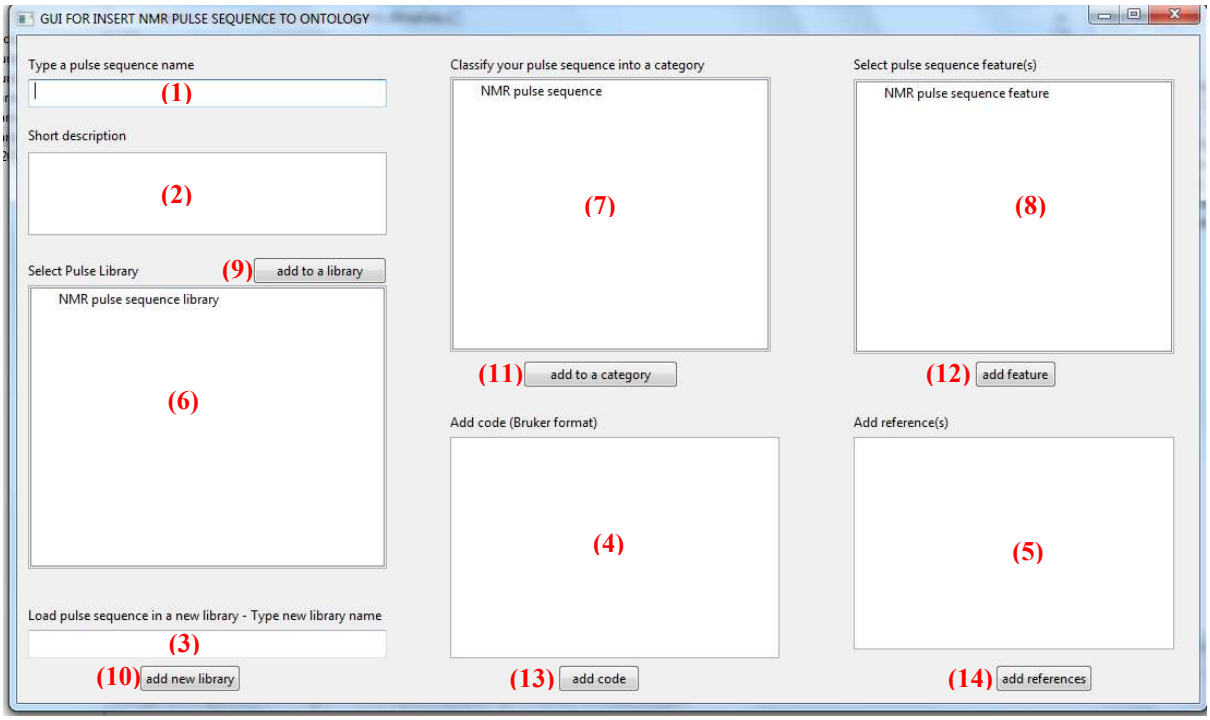


Figure 23: GUI window with elements for the user itneraction with the ontology

Below is a tutorial to demonstrate how the application works.

6.1.3. Step-by-step tutorial

In this step-by-step tutorial a new NMR pulse sequence will be added to the ontology. The pulse sequence is called IFSERF (Isotope Filtered Self Refocusing Element) which is my first published NMR pulse sequence [54]. Briefly, that is a 2D J-resolved experiment with the main goal of allowing JHH coupling constant measurement in symmetric molecules.

Step 1: Inserting a new NMR pulse sequence name and some short description which will be added as an annotation comment. (Figure 24)

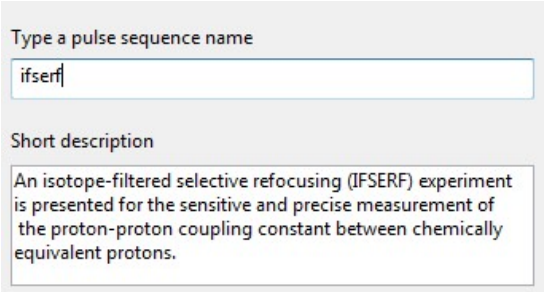


Figure 24: GUI expanded area showing the typing area for the name of a new NMR pulse sequence and its short description

Step 2 (optional): Create a new library. In this case we will create a new library called *nolis* and we will press the *add new library* button. (Figure 25)

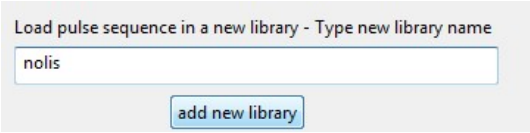


Figure 25: GUI expanded area showing the typing area for the name of a new library and the add new library button

Step 3: Select the library where the pulse sequence will fit. In this case we will select the already newly generated nolis NMR pulse sequence library and then press the *add to a library* button. (Figure 26)

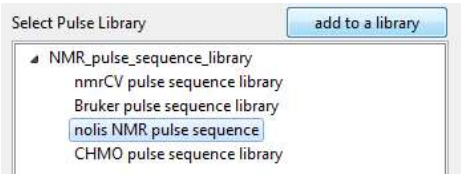


Figure 26: GUI expanded area showing the nolis library selection from the NMR pulse sequence library tree

After clicking on the *add to a library* button the insertion of a pulse sequence to the library will take place. Then a notification window will prompt confirming the process has been successfully completed. (Figure 27)

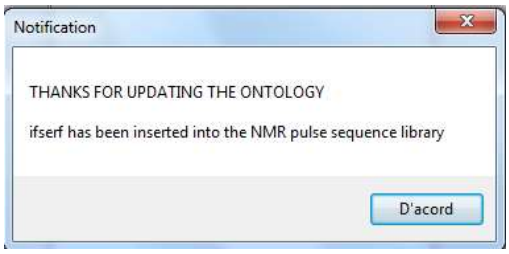


Figure 27: Notification window indicating a successful insertion of a new nmr pulse sequence to a library

Step 4: Classifying the pulse sequence into a category by selecting the appropriate category and pressing the *add to category* button (Figure 28). In this case a J resolved selection is the adequate.

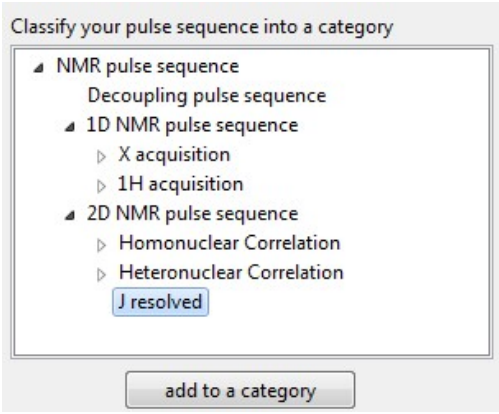


Figure 28: GUI expanded area showing the J resolved category selection NMR pulse sequence tree

Once added a notification prompt will arise notifying the process has been successfully achieved (Figure 29)

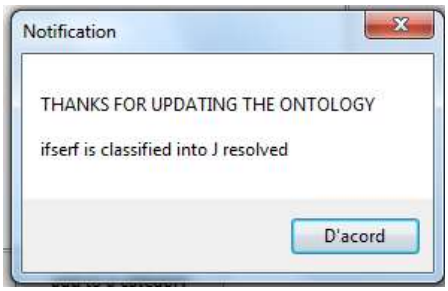


Figure 29: Notification window indicating a successful classification for *ifserf* into J resolved category

Step 5: Adding features to the pulse sequence by selecting the feature from the tree *nmr pulse sequence feature* tree and pressing the button add feature. This step may be repeated as times as necessary in order to select all the necessary features. For this tutorial example just *JHH measurement* which is the main treat for IFSERF is selected (Figure 30).

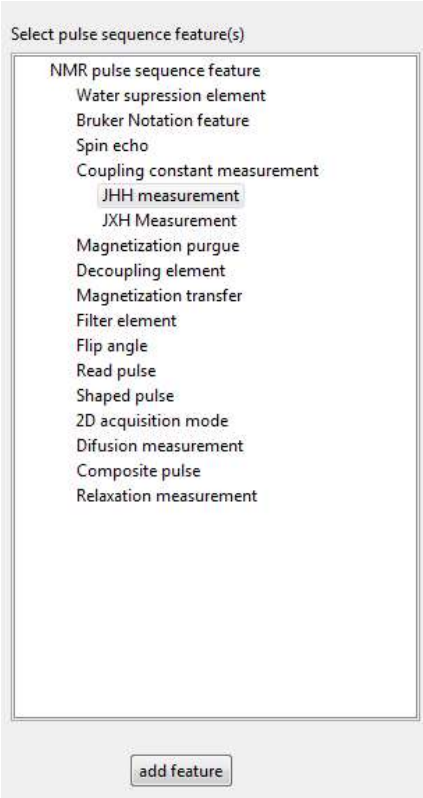


Figure 30: GUI expanded area showing the *JHH measurement* feature selected

After adding the feature a notification prompt will notify that the process has been successfully achieved (Figure 30)

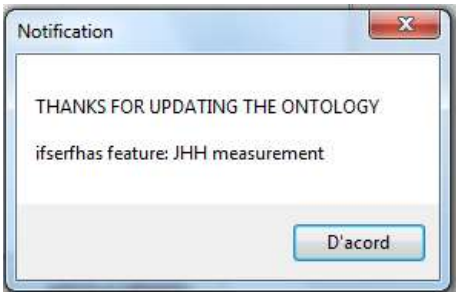


Figure 31: Notification window indicating a successful addition of feature *JHH measurement* has been produced for ifserf

Step 6: Adding a pulse sequence code (in Bruker format). (Figure 32)

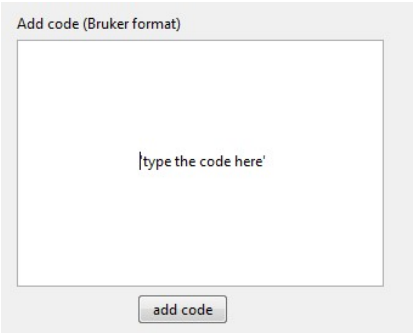


Figure 32: GUI expanded area showing where to insert the pulse sequence Bruker code

After adding the code a notification prompt will notify that the process has been successfully achieved (Figure 33).

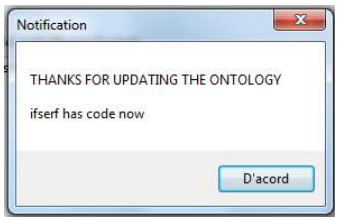


Figure 33: Notification window indicating a successful addition of code for IFSERF

Step 7: Adding reference(s). (Figure 34)

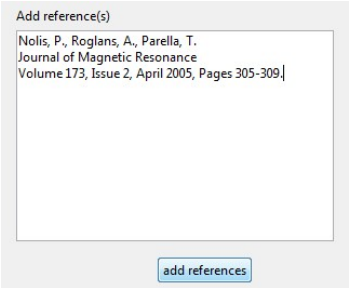


Figure 34: GUI expanded area showing reference insertion for IFSERF NMR pulse sequence

After adding the code a notification prompt will notify that the process has been successfully achieved (Figure 35)

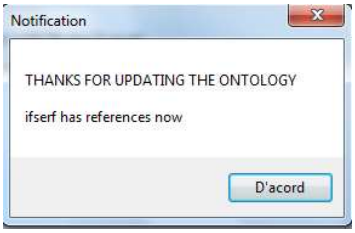


Figure 35: Notification window indicating a successful addition of a reference for IFSERF

6.1.4. Error handling

In order to help the user to follow the step-by-step protocol some notification prompts are added in order to advise if some step or requirement is missing. The events that will generate a notification are listed below:

- Any pulse sequence name or short description is introduced and the *add to a library* button is selected (Figure 36)
- Any library is selected and the *add to a library* button is selected (Figure 37)
- Any category is selected and the *add to a category* button is selected (Figure 38)
- Any feature is selected and the *add feature* button is selected (Figure 39)
- Any code is written and the *add code* button is selected (Figure 40)
- Any reference is written and the *add references* button is selected (Figure 41)
- Try to insert a pulse sequence to a category, adding a feature, adding code or adding reference(s) prior to insert the pulse sequence to a library (Figure 42).

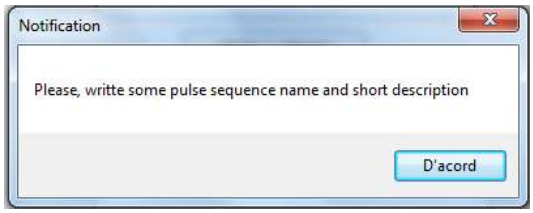


Figure 36: Notification window indicating pulse sequence name and short description must be written

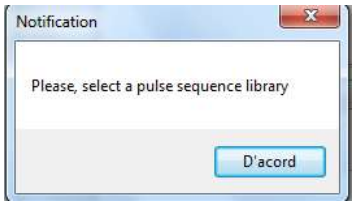


Figure 37: Notification window indicating a pulse sequence library must be selected

Figure 38: Notification window indicating a category should be chosen

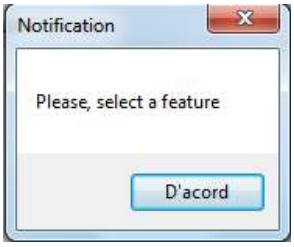


Figure 39: Notification window indicating a feature should be chosen

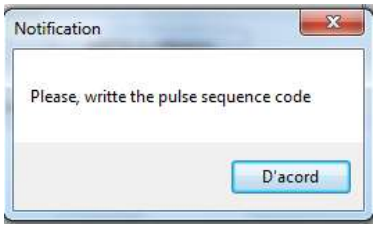


Figure 40: Notification window indicating the pulse sequence code should be written

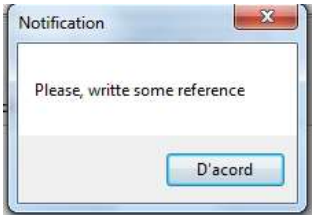


Figure 41: Notification window indicating a reference should be included

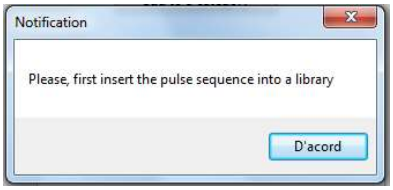


Figure 42: Notification indicating pulse sequence has to be inserted to a library first

6.2. Ontology consulting: SPARQL Queries using Apache Jena Fuseki

The second purpose of the project is to demonstrate the utility of the NMR pulse sequence ontology from a basic user point of view by using a SPARQL end point. To do so, the Apache Jena Fuseki server [55] is used. The demonstration will be performed using a localhost installation which is done following the instructions encountered at https://jena.apache.org/documentation/fuseki2/index.html.

Once installed, the ontology has to be uploaded to the server. A screen capture showing ontology upload at the Apache Jena Fuseki Server is shown below (Figure 43).

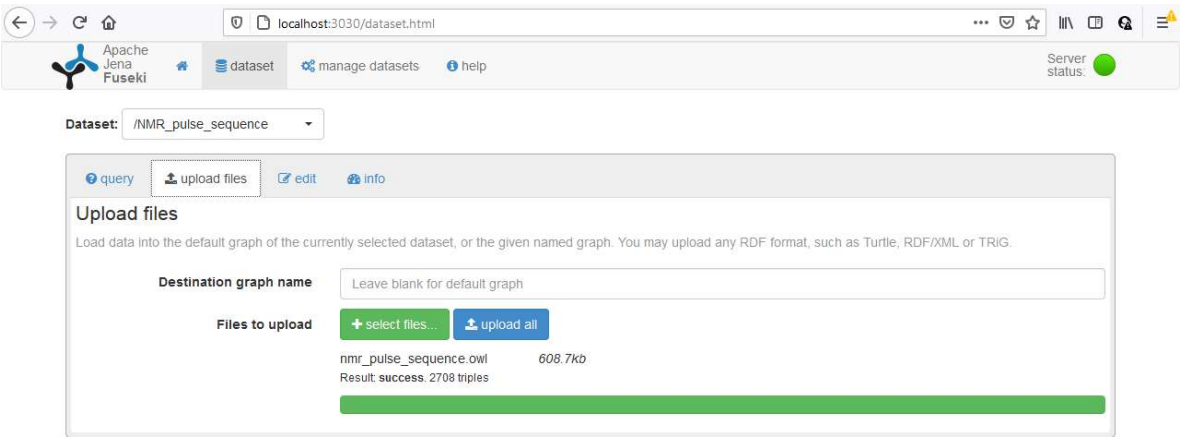


Figure 43: Screen capture of ontology uploading at Apache Jena Fuseki

After uploading the .owl file the system is ready to perform SPARQL queries to the ontology. Some use cases are showed below with screen captures of queries and answers.

6.2.1. Use case 1: List of NMR pulse sequences

Question: What are the pulse sequences contained in the ontology? (Figure 44)

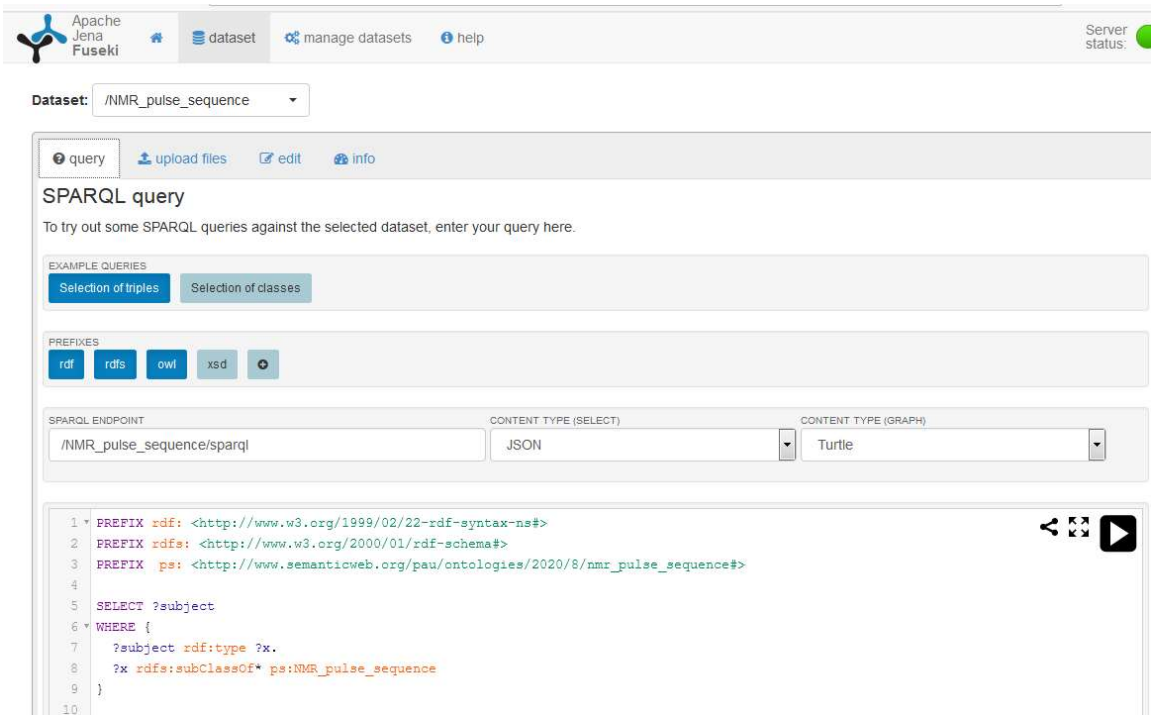


Figure 44: SPARQL query for listing the NMR pulse sequences in the ontology

Answer: the 172 pulse sequences instances uploaded in the ontology (Figure 45)

Figure 45: SPARQL answer for listing the NMR pulse sequences in the ontology

6.2.2. **Use case 2:** Which are the treats of a given NMR pulse sequence?

Example 1: NMR pulse sequence hsqcetgpsi (Figure 46).

```
PREFIX  ps: <http://www.semanticweb.org/pau/ontologies/2020/8/nmr_pulse_sequence#>

SELECT ?object
WHERE {
  ps:hsqcetgpsi ps:hasFeature ?object
}
```

QUERY RESULTS

Table | Raw Response

Showing 1 to 3 of 3 entries

Search: Show 10 entries

	object
1	ps:sensitivity_improvement_using_PEP
2	ps:phase_sensitive_using_echo_antiecho-TPPI_method
3	ps:hsqc_magnetization_transfer

Showing 1 to 3 of 3 entries

Figure 46: SPARQL query/answer for listing the treats of hsqcetgpsi NMR pulse sequence

Example 2: NMR pulse sequence noesygpph (Figure 48)

```
PREFIX  ps: <http://www.semanticweb.org/pau/ontologies/2020/8/nmr_pulse_sequence#>

SELECT ?object
WHERE {
  ps:noesygpph ps:hasFeature ?object
}
```

QUERY RESULTS

Table | Raw Response

Showing 1 to 3 of 3 entries

Search: Show 10 entries

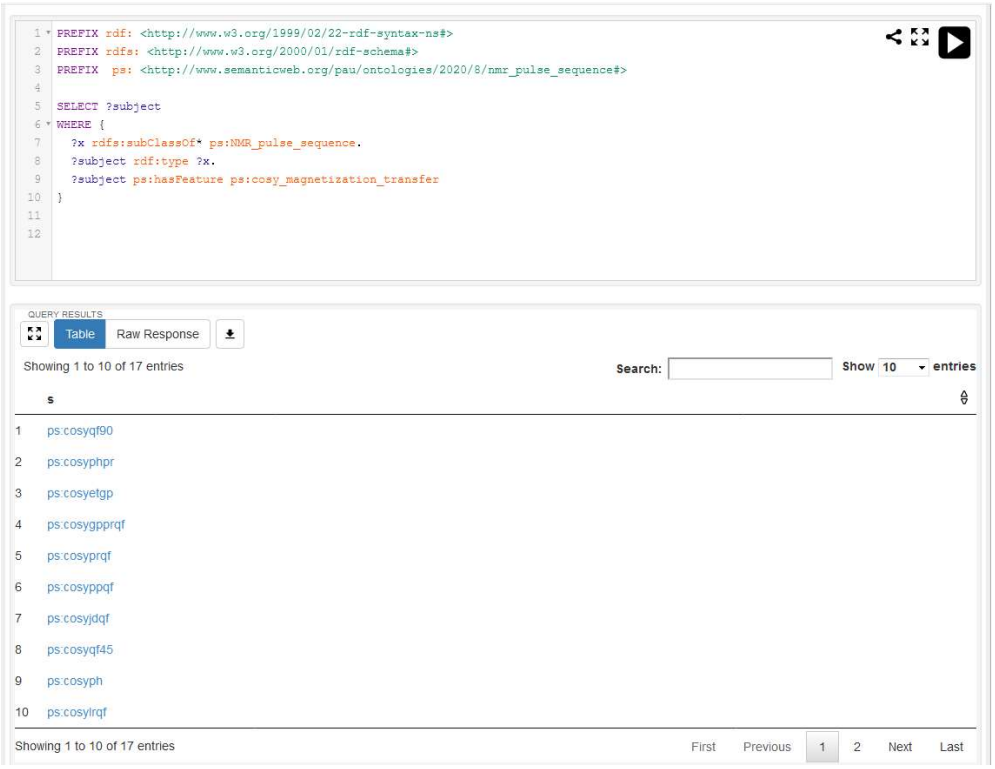
	object
1	ps:coherence_selection_pulsed_field_gradients
2	ps:noesy_magnetization_transfer
3	ps:phase_sensitive_qseq_states_tppi

Showing 1 to 3 of 3 entries

Figure 47: SPARQL query/answer for listing the treats of noesygpph NMR pulse sequence

6.2.3. **Use case 3:** What NMR pulse sequences have *** treat?

Example 1: treat = cosy magnetization transfer (Figure 48)



```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX  ps: <http://www.semanticweb.org/pau/ontologies/2020/8/nmr_pulse_sequence#>

SELECT ?subject
WHERE {
  ?x rdfs:subClassOf* ps:NMR_pulse_sequence.
  ?subject rdf:type ?x.
  ?subject ps:hasFeature ps:cosy_magnetization_transfer
}
```

QUERY RESULTS

Table | Raw Response

Showing 1 to 10 of 17 entries

Search: Show 10 entries

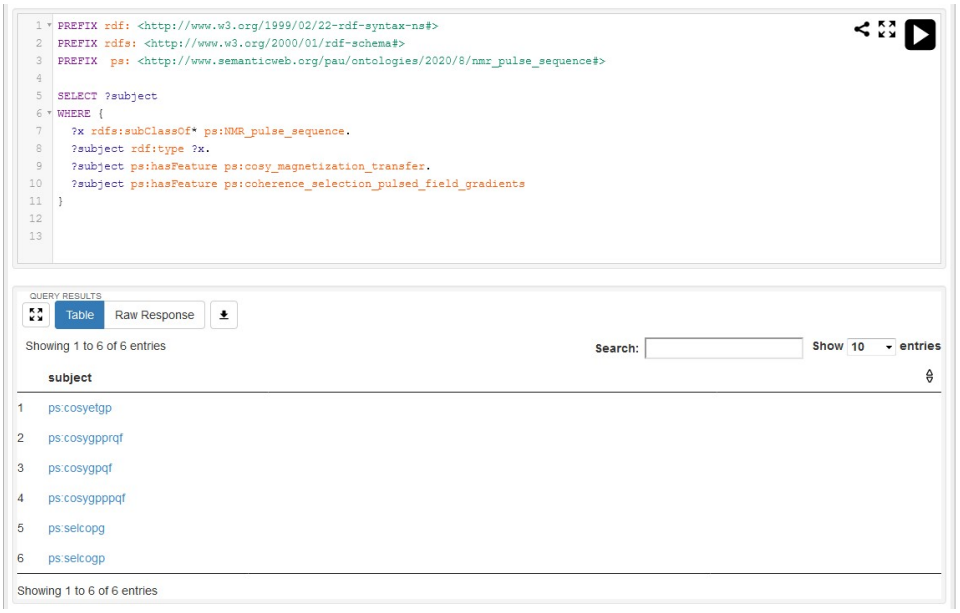
	s
1	ps:cosyqf90
2	ps:cosyphpr
3	ps:cosyetgp
4	ps:cosygpprqf
5	ps:cosyprqf
6	ps:cosyppqf
7	ps:cosyjdqf
8	ps:cosyqf45
9	ps:cosyph
10	ps:cosylrqf

Showing 1 to 10 of 17 entries

First Previous 1 2 Next Last

Figure 48: SPARQL query/answer for listing NMR pulse sequence with treat cosy magnetization transfer

Example 2: treats = cosy magnetization transfer + coherence selection pulsed field gradients (Figure 49)



```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX  ps: <http://www.semanticweb.org/pau/ontologies/2020/8/nmr_pulse_sequence#>

SELECT ?subject
WHERE {
  ?x rdfs:subClassOf* ps:NMR_pulse_sequence.
  ?subject rdf:type ?x.
  ?subject ps:hasFeature ps:cosy_magnetization_transfer.
  ?subject ps:hasFeature ps:coherence_selection_pulsed_field_gradients
}
```

QUERY RESULTS

Table | Raw Response

Showing 1 to 6 of 6 entries

Search: Show 10 entries

	subject
1	ps:cosyetgp
2	ps:cosygpprqf
3	ps:cosygpqf
4	ps:cosygpppqf
5	ps:selcopg
6	ps:selcogp

Showing 1 to 6 of 6 entries

Figure 49: SPARQL query/answer for listing NMR pulse sequence with treats cosy magnetization transfer + coherence selection pulsed field gradients

6.2.4. **Use case 4:** What NMR pulse sequences can one use to perform *** Type experiments?

Example 1: HMQC type experiments (Figure 50)

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX  ps: <http://www.semanticweb.org/pau/ontologies/2020/8/nmr_pulse_sequence#>

SELECT ?subject
WHERE {
  ?x rdfs:subClassOf* ps:HMQC.
  ?subject rdf:type ?x.
}
```

QUERY RESULTS

Table | Raw Response

Showing 1 to 9 of 9 entries

Search: Show 10 entries

	subject
1	ps:hmqcf3gpph19
2	ps:hmqcetgpsi
3	ps:hmqcph
4	ps:hmqcgpqf
5	ps:hmqcphpr
6	ps:hmqcbiqf
7	ps:hmqcbiph
8	ps:hmqcetgp
9	ps:hmqcqf

Showing 1 to 9 of 9 entries

Figure 50: SPARQL query/answer for listing HMQC type NMR pulse sequence

Example 2: TOCSY type experiments (Figure 51)

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX  ps: <http://www.semanticweb.org/pau/ontologies/2020/8/nmr_pulse_sequence#>

SELECT ?subject
WHERE {
  ?x rdfs:subClassOf* ps:TOCSY.
  ?subject rdf:type ?x.
}
```

QUERY RESULTS

Table | Raw Response

Showing 1 to 10 of 11 entries

Search: Show 10 entries

	subject
1	ps:mlevph
2	ps:dispsi2gpph19
3	ps:dipsi2etgp
4	ps:nmrCV_NMR:1400168
5	ps:nmrCV_NMR:1400171
6	ps:mlevgpph19
7	ps:mlevphpp
8	ps:dipsi2etgpsi
9	ps:dipsi2phpr
10	ps:dipsi2ph

Showing 1 to 10 of 11 entries

First Previous 1 2 Next Last

Figure 51: SPARQL query/answer for listing TOCSY type NMR pulse sequence

6.2.5. **Use case 5:** What pulse sequences instances exist in a given library?

Example: list of pulse sequences in nmrCV library (Figure 52)

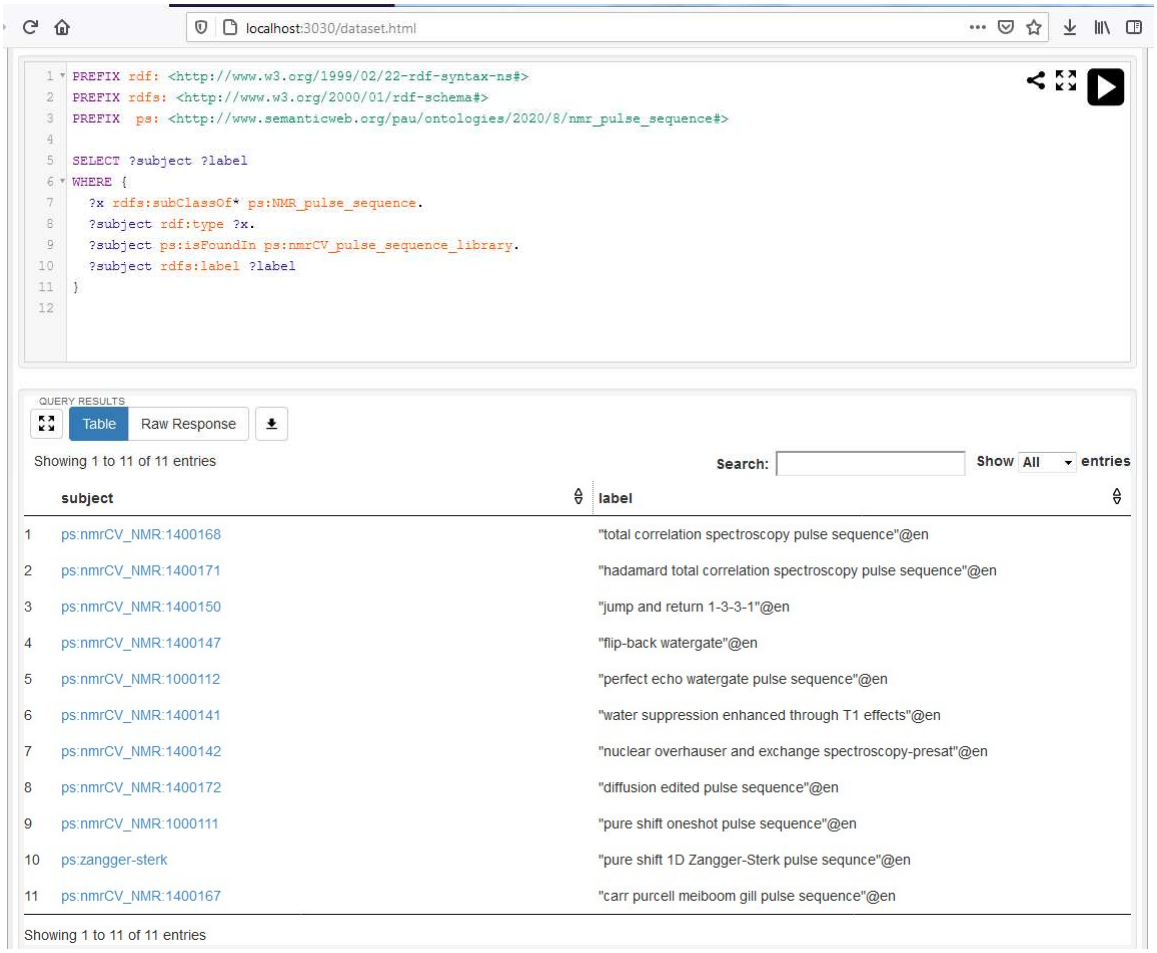


Figure 52: SPARQL query/answer for listing NMR pulse sequence of nmrCV pulse sequence library

7. Summary discussion and final remarks

In this project it is faced the ontology design of an area difficult to capture, the NMR pulse sequences domain. The approach herein presented is focused on the knowledge representation of pulse sequences used in liquid state. It is remarkable that just a small fraction of that domain is collected. Thus, it is considered a pilot ontology and a proof-of-concept project.

The project has ended with three deliverable products which fulfils the objectives of the project. Those are the memory itself, the ontology for the NMR pulse sequence concept and a functional java applet to interact with the ontology. Furthermore, it is shown how a user can extract useful information data using SPARQL queries. However, the project presents an important lack of functionality which is directly related to the inherent prototype nature of the product. That is, the data introduction part (java applet) and consulting part (SPARQL queries from a server) are not integrated, meaning that for consulting the new data a reload of the ontology to the server has to be performed manually.

Lastly, and as project final remarks, it is important to mention the perspective and possible future work of the project, which have many field to grow and expansion capacities.

- Further expansion of the ontology. At that point, coordination, with other NMR experts should be interesting in order to agree the appropriate classes, the hierarchy branches and new properties.

- Improving of the GUI interface which at this point is just functional and very basic.

- Integration of the ontology to a real-time data service provider.

From a personal point of view the project afforded me a deeper knowledge of the web semantic concept. For me the most relevant has been to achieve the understanding of how triples are performed during the ontology implementation and how finally that allows performing SPARQL queries to extract the desired information. In that sense, I have realized how important is prior to start the ontology construction write down the questions one want have the answers. Then one have to perform the ontology design thinking on that the answers to these questions have to be generated.

Also, the project has given me the opportunity of working on the design of a functional applet with a GUI using window builder, which I never used before and I have to say I thought it was much easier; I encountered more problems than expected for understanding and code such a basic interface.

Finally and just to finish I must say I have extract a good profit and knowledge by doing this project related on semantic web topic and more concretely about the ontology concept.

8. References

[1] R. K. Harris, J. Kowalewski and S. C. De Menezes, "Parameters and Symbols for use in Nuclear Magnetic Resonance," *Pure & Appl. Chem,* vol. 69, pp. 2489-2495, 1997.

[2] C. W. Hilbers, R. Daptein, B. D. Sykes, P. E. Wrigh and K. Wütrich, "Recommendations for the Presentation of NMR Structures of Proteins and Nucleic Acids," *Pure & Appl. Chem. 70, 117-142.,* vol. 70, pp. 117-142, 1998.

[3] R. K. Harris, E. D. Becker, S. M. Cabral de Menezes, R. Goodfellow and P. Granger, "NMR nomenclature. Nuclear spin properties and conventions for chemical shifts," *Pure & Appl. Chem.,* vol. 73, pp. 1795-1818, 2001.

[4] R. K. Harris, E. D. Becker, S. M. Cabral de Menezes, P. Granger, R. E. Hoffman and K. W. Slim, "Further Conventions for NMR Shielding and Chemical Shifts," *Pure & Appl. Chem.,* vol. 1, pp. 54-98, 2008.

[5] S. R. Hall, "The STAR file: A new format for electronic data format and storage," *J. Chem. Inf. Comput. Sci.,* vol. 31, no. 2, pp. 323-333, 1991.

[6] S. R. Hall and N. Spadaccini, "The STAR File: detailed specifications," *J. Chem. Inf. Comput. Sci.,* vol. 34, no. 3, pp. 505-508, 1994.

[7] N. Spadaccini and S. R. Hall, "Extensions to the STAR File Syntax," *J. Chem. Inf. Comput. Sci.,* vol. 52, no. 8, pp. 1901-1906, 2012.

[8] S. Hall and A. P. Cook, "STAR Dictionary Definition," *J. Chem. Inf. Comput. Sci.,* vol. 35, pp. 819-825, 1995.

[9] E. L. Ulrich, K. Bascaran, H. Dashti, Y. E. Ionnidis, M. Livny, P. R. Romero, D. Maziuk, J. R. Wedell and H. Yao, "NMR-STAR: comprehensive ontology for representing, archiving and exchanging data from nuclear magnetic resonance spectroscopic experiments," *Journal of Biomolecular NMR,* vol. 73, pp. 5-9, 2019.

[10] D. Schober, D. Jacob, M. Wilson, J. A. Cruz, A. Marcu, J. R. Gran, A. Moing, C. Deborde, L. F. Figueiredo, K. Haugh, P. Rocca-Serra, J. Easton, T. M. Ebels, J. Hao, C. Ludwig, U. L. Günter, A. Rosato, M. S. Klein, I. A. Lewis, C. Luchinat, A. R. Jones, A. Grauslys, M. Larralde, M. Yokochi, N. Kobayachi, A. Porzel, J. L. Griffin, M. R. Viant, D. S. Wishart, C. Steinbeck, R. M. Salek and S. Neumann, "nmrML: A Community Supported Open Data Standard for the Description, Storage, and Exchange of NMR Data," *Analytical Chemistry,* vol. 90, no. 1, pp. 649-656, 2017.

[11] M. Pupier, J.-M. Nuzillard, J. Wist, N. E. Schlörer, S. Kuhn, M. Erdely, C. Steinbeck, A. J. Williams, C. Butts, T. D. Claridge, B. Mikhova, W. Robien, H. Dashti, H. R. Eghbalnia, C. Farés, C. Adam, P. Kessler, F. Moriaud, M. Elyashberg, D. Argiropoulus, P. Giredeau, P. Giraudeuau, R. R. Gil, P. Trevorrow and D. Jeannerat, "NMReDATA, a standard to report the NMR assignment and parameters of organic compounds," *Magn. Reson. Chem.,* vol. 56, no. 8, pp. 703-715, 2018.

[12] C. Bachelor, "CHMO (Chemical Methds Ontology)," [Online]. Available: https://github.com/rsc-ontologies/rsc-cmo. [Accessed 23 Setember 2020].

[13] A. Bandrowski, R. Brinkman, M. Brochhausen, B. H. MattheW, B. Bug, M. C. Chibucos, K. Clancy, M. Courtot, D. Derom, M. Dumontier, L. Fan, J. Fostel, G. Fragoso, F. Gibson, A. Gonzalez-Beltran, M. A. Haendel, Y. He, M. Heiskanen, T. Hernandez-Boussard,, M. Jensen, Y. Lin, A. L. Lister, P. Lord, J. Malone, E. Manduchi, M. McGee, N. Morrison, O. A. James, H. Parkinson, B. Peters, P. Rocca-Serra, A. Ruttenberg, S.-A. Sansone, R. H. Scheuermann, D. Schober, B. Smith, L. N. Soldatova, C. J. Stoeckert Jr., C. . F. Taylor, C. Torniai, J. A. Turner, R. Vita, P. L. Whetzel and J. Zheng, "The Ontology for Biomedical Investigations," *PLOS ONE,* vol. 11, no. 4, 2016.

[14] "nmrCV," [Online]. Available: http://nmrml.org/cv/.

[15] EU FP7 Project COSMOS, "nmrML (Nuclear Magnetic Resonance Mark-up Language)," [Online]. Available: http://nmrml.org/. [Accessed 23 September 2020].

[16] E. L. Ulrich, . H. Akutsu, J. F. Doreleijers, Y. Harano, Y. E. Ioannidis, J. Lin, M. Livny, S. Mading, D. Maziuk, Z. Miller, E. Nakatani, C. F. Schulte, D. E. Tolmie, K. R. Wenger, H. Yao and J. L. Markley, "BioMagResBank," *Nucleic Acids Research,* vol. 36, no. suppl_1, p. D402–D408, 2007.

[17] wwPDB consortium, "Protein Data Bank: the single global archive for 3D macromolecular structure data," *Nucleic Acids Research,* vol. 47, no. D1, pp. D520-D528, 2019.

[18] S. Berger and S. Braun, 200 and More NMR Experiments: A Practical Course, Wiley, 2004.

[19] T. D. Claridge, High-Resolution NMR Techniques in Organic Chemistry, Elsevier, 2016 - Third edition.

[20] T. Parella, "NMR guide," Bruker Biospin, [Online]. Available: https://sites.google.com/site/teodorparella/nmrguide.

[21] T. Berners-Lee and M. Fischetti, Weabing The Web: the original design and ultimate destiny of the World Wide Web by its inventor, San Francisco: HarperSanFrancisco, 1999.

[22] [Online]. Available: https://wiki.dbpedia.org/.

[23] [Online]. Available: http://www.geonames.org/.

[24] [Online]. Available: http://xmlns.com/foaf/spec/.

[25] [Online]. Available: https://www.w3.org/wiki/SweoIG/TaskForces/CommunityProjects/LinkingOpenData#Project_Description.

[26] T. Gruber, “A translation approach to portable ontology specifications,” *Knowledge Acquisition,* vol. 5, no. 2, pp. 199-220, 1993.

[27] W3C, "W3C," [Online]. Available: https://www.w3.org/.

[28] Unicode, Inc., “UNICODE,” [Online]. Available: https://home.unicode.org/.

[29] T. Berners-Lee, R. T. Fielding and L. Masinter, “Uniform Resource Identifiers (URI): Generic Syntax, section 3, Syntax Components.,” 2005. [Online]. Available: https://www.rfc-editor.org/rfc/pdfrfc/rfc3986.txt.pdf.

[30] [Online]. Available: https://en.wikipedia.org/wiki/List_of_types_of_XML_schemas#Music_playlists.

[31] W3C, [Online]. Available: https://www.w3.org/TR/rdf-syntax-grammar/.

[32] W3C, [Online]. Available: https://www.w3.org/TR/2014/REC-n-triples-20140225/.

[33] W3C, [Online]. Available: https://www.w3.org/TR/turtle/.

[34] W3C, [Online]. Available: https://www.w3.org/TR/trig/.

[35] W3C, [Online]. Available: https://www.w3.org/TR/2014/REC-n-quads-20140225/.

[36] W3C, [Online]. Available: https://www.w3.org/TR/json-ld/.

[37] W3C, [Online]. Available: https://www.w3.org/TR/rdf-schema/.

[38] W3C, [Online]. Available: https://www.w3.org/TR/2012/REC-owl2-overview-20121211/.

[39] W3C, [Online]. Available: https://www.w3.org/TR/2013/NOTE-rif-overview-20130205/.

[40] W3C, [Online]. Available: https://www.w3.org/Submission/SWRL/.

[41] J. Lee, M.-D. Pham, J. Lee, W.-S. Han, H. Cho, H. Yu and J.-H. Lee , "Processing SPARQL queries with regular expressions in RDF databases. 12, S6 (2011).," *BMC Bioinformatic,* vol. 12, no. s16, 2011.

[42] N. F. Noy and D. L. McGuiness, “Ontology Development 101: A Guide to Creating Your First Ontology,” *Computer Science,* 2001.

[43] O. Corcho, M. Férnandez-López and A. Gómez-Pérez, “Methodologies, tools and languages for building,” *Data & knowledge engineering,* vol. 46, pp. 41-64, 2003.

[44] M. Cristiani and R. Cruel, “A Survey on Ontology Creation Methodologies,” *International Journal on Semantic Web and Information Systems,* vol. 1, no. 2, p. 21, 2005.

[45] M. Fernández-López, A. Gómez-Pérez, A. Pazos-Sierra y J. Pazos-Sierra, «Building a chemical ontology using METHONTOLOGY and the ontology design environment,» *IEEE Intelligent Systems & their applications,* vol. 4, nº 1, pp. 37-46, 1999.

[46] S. Staab, H. P. Schnur and R. Studer, “Knowledge processes and ontologies,” *IEEE Intelligent Systems,* vol. 16, no. 1, pp. 26-34, 2001.

[47] “Text2Onto,” [Online]. Available: http://neon-toolkit.org/wiki/1.x/Text2Onto.html.

[48] B. Swartout, P. Ramesh, K. Knight y T. Russ, «Toward Distributed Use of Large-Scale Ontologies,» de *AAAI Symposium on Ontological Engineering*, Stanford, 1997.

[49] M. Grüninger y M. S. Fox, «Methodology for the design and evaluation of ontologies,» de *Workshop on Basic*, Montreal, 1995.

[50] M. A. Musen, «The Protégé project: A look back and a look forward,» *AI Matters. Association of Computing Machinery Specific Interest Group in Artificial Intelligence.,* vol. 1, nº 4, 2015.

[51] “Protégé,” [Online]. Available: https://protege.stanford.edu/.

[52] [Online]. Available: https://jena.apache.org/.

[53] [Online]. Available: https://www.eclipse.org/windowbuilder/.

[54] P. Nolis, A. Roglans and T. Parella, “IFSERF, an isotope-filtered SERF experiment for the precise measurement of proton–proton coupling constants between chemically equivalent protons,” *Journal Of Magnetic Resonance,* vol. 173, no. 2, pp. 305-309, 2005.

[55] [Online]. Available: https://jena.apache.org/documentation/fuseki2/index.html.

[57] [Online]. Available: https://www.eclipse.org/.