공학석사 학위논문

RDF Data Management for Vessel and Patent Information

선박 및 특허 정보에 대한 RDF 데이터 관리

지도교수 신 옥 근

2020년 2월

한국해양대학교 대학원

컴퓨터공학과

ZASLYANA BINTI MOZAHKER

본 논문을 ZASLYANA BINTI MOZAHKER 의 공학석사

학위논문으로 인준함

위원장 공학박사 박 휴 찬 (인)

위 원 공학박사 이 장 세 (인)

위 원 공학박사 신 옥 근 (인)

2019년 12월 26일

한국해양대학교 대학원

컴퓨터공학과

Table of Contents

List of Figures

List of Tables

List of Abbreviations

RDF	Resource Description Framework
RDFS	Resource Description Framework Schema
SPARQL	SPARQL Protocol and RDF Query Language
SQL	Structured Query Language
W3C	World Wide Web Consortium
OWL	Web Ontology Language
URI	Uniform Resource Identifiers
IRI	Internationalized Resource Identifier
LOD	Linked Open Data
RIF	Rule Interchange Format
XML	Extensible Markup Language
XHTML	Extensible Hypertext Markup Language
JSON	JavaScript Object Notation
CSV	Comma-Separated Values
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
Turtle	Terse RDF Triple Language
TDB	Triple Database
Port-MIS	Port Management Information System
MMSI	Maritime Mobile Service Identity
MRN	Movement Reference Number
IMO	International Maritime Organization
ISO	International Organization for Standardization

ISA	International Shipping Agency
AIS	Automatic Identification System
KIPRIS	Korea Intellectual Property Rights Information Service
KIPO	Korean Intellectual Property Office
KIPI	Korea Institute of Patent Information
WIPO	World Intellectual Property Organization
PTAB	Patent Trial and Appeal Board
CPC	Cooperative Patent Classification
PCT	Patent Cooperation Treaty
IEEE	Institute of Electrical and Electronics Engineers Incorporated
LOM	Learning Object Metadata

RDF Data Management for Vessel and Patent Information

Zaslyana Binti Mozahker

Department of Computer Engineering,

Graduate School of Korea Maritime and Ocean University

Advised by:

Prof. Ok-Keun Shin

Abstract

The Resource Description Framework (RDF) is widely used to represent information in the Web. Efforts have been made to map RDF data to a relational representation, and this method has been adopted by several systems. RDF is queried using SPARQL, a standard W3C-recommended query language used to query graph and represent data as RDF triples. A set of tools and technologies are implemented and tested using SPARQL and Apache Jena Fuseki as RDF triplestore. The increasing size of RDF data requires the storage and query of an efficient system. These organizations'

framework, in the aspect of designing, analyzing, query optimization, storage and processing is required for efficient retrieval of RDF data. Together, SPARQL and RDF make it easier to merge results from multiple data sources. This thesis provides an overview of the method to manage data based on existing data of two organizations such as Port-MIS and KIPRIS. The RDF model is designed to enable web-based representation, information exchange and yet to suggest a promising direction for future research.

Chapter 1 Introduction

1.1 Background of Research

Semantic Web is an extension of the Web in which data have meaning for humans, and machines in a way that it can be searched, interpreted, shared and reused between applications, organizations and communities. Its technologies enable people to create data stores on the Web, build vocabularies, and write rules for handling data. This is known as the Web of data.

In the Semantic Web framework, an important concept within the Semantic Web as Linked Data. Technologies such as RDF, SPARQL, OWL, and SKOS empower Linked Data. The aim of Linked Data is to relate data using the RDF model to allow machines to browse through the Web [1].

RDF and SPARQL are introduced and various techniques were discussed to improve mappings between RDF and relational data [2]. The implementation with SPARQL, an RDF query language, can join data from various databases, as well as documents, inference engines, or anything that could express its knowledge as a directed labelled graph.

1.2 Research Objectives

This thesis proposes a new query to retrieve specific data from RDF storage. Data retrieval has been made easy by pulling a specific keyword in searching the data where the published data is real and is based on legitimate sites. The consistency of RDF comes at a price where the physical design of the RDF database is complex and there is no consistency on how to interpret the RDF data [3]. Therefore, it is preferable to publish in a highly stable and persistent environment, by choosing a stable namespace for the RDF schema such as W3C. Another good approach is to publish persistent Uniform Resource Identifiers (URI) sets in terms of format and in terms of design, rules and management [4].

The second objective is to propose a design for RDF schema of a vessel information structure and a patent data structure. This implies the existing data does not need to be modified in order to benefit from the advantages of the design pattern, particularly the generation of label graph.

1.3 Organization of Thesis

The organization of the thesis is as follows. An overview of the whole subject of the research and the objectives of the thesis are presented in this chapter. Chapter 2 presents a review on previous papers on the subject of linked open data, along with a detailed explanation of its relation with the semantic web. For example, a semantic link graph. Included in Section 2.1 is the suggested 5-star deployment system along with the benefits in LOD. Section 2.2 explains the RDF schema, storage and the language used which is the SPARQL. Apache Jena Fuseki is used as a server to store RDF Triples and further elaborated in Section 2.3. In chapter 3, the schema and design are explained. While in chapter 4, implementation and queries regarding the vessel and parent information are shown for further classification on how the algorithm of data searching works. Finally, in Chapter 5, conclusion and further works are discussed to ensure the growth of the Web of Data in the upcoming future.

Chapter 2 Literature Review

2.1 Semantic Web

Recently, the Web's Semantic data have increased, and the systems are designed to represent the real world in the data set as accurately as possible. Data symbols are organized linearly and hierarchically to give certain meanings as described later on in this study. Semantic data enables machines to communicate with real information without human interpretation by representing the real world in datasets [5].

RDF provides the framework for publishing, by linking these data since it is widely tested, and scalable technology has been well known for modelling data. The framework was originally a part of the Semantic Web Stack. As shown in Fig. 2.1, it is used in all possible ways where it is necessary to represent a high quality of semantic data. The integrated representation of data in RDF helps the identification, disambiguation and interconnection of information by software agents and various systems to read, analyze and act on [6].

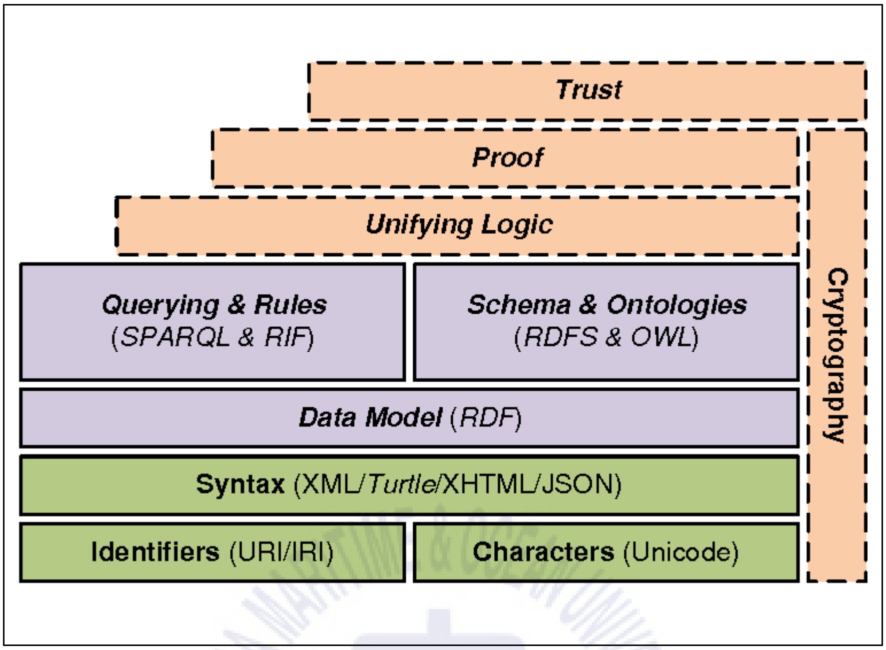


Fig. 2.1 Semantic Web Stack (aka. Semantic Web Layer Cake) [7]

2.2 Linked Open Data

Linked Open Data (LOD) is a robust combination of Linked Data and Open Data where both are linked and open source. DBpedia, for example, is a community-driven initiative for extracting structured data from Wikipedia and making it available on the internet is a remarkable example of a LOD collection providing greater queries, enhancing untapped data exploration and effective results analytics.

The following, Fig 2.2 shows an illustration of Linked Data is DBpedia in converting Wikipedia contents into RDF and link the contents to other databases like GeoNames. Data offered by DBpedia are complete and more accurate by incorporating other data. Since 2008, the LOD Cloud has an average amount of 50 published datasets. As of March 2019, the datasets grow rapidly with the amount of 1,239 datasets with 16,147 links [8, 9, 10].

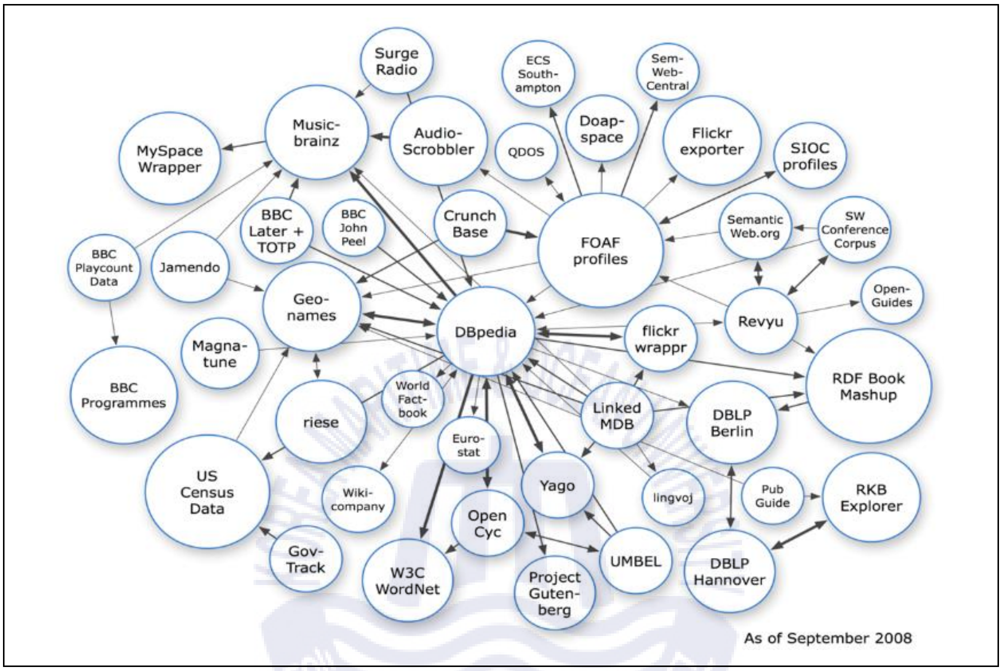


Fig. 2.2 The Linked Open Data Cloud as of 2008 [8]

The inventor of the World Wide Web and the creator and advocate of the Semantic Web and Linked Data, Sir Tim Berners-Lee, suggested a 5-star deployment system [11];

Table 2.1 The 5-star deployment system

★ Data is accessible on the Web (any format) under an open license
★★ Data is accessible as structured data (e.g. Excel rather than an image scan of a table)
★★★ Non-proprietary formats are utilized (e.g. CSV rather than Excel)

★★★★ Utilize URIs to indicate things, so that people can point at individual data
★★★★★ Data is linked to other data to provide context

Linked Data is one of the major elements of the Semantic Web, also known as the Web of Data. The Semantic Web connects between datasets that are understandable not entirely to humans, but also machines. It also gives the best practices for making these links possible. Simply put, Linked Data is a set of design concept to share an interlinked machine-readable data on the Web.

In 2006, for Linked Open Data, Sir Tim Berners-Lee laid down the four design principles of Linked Data [12];

1. Use URIs as names for things;

2. Use HTTP URIs so that these names can be looked up;

3. Provide useful information on lookup of the URIs (esp. RDF, SPARQL);

4. Include links to other URIs, so that more things are discoverable.

In summary, Linked Data supports the extension of the data models and allows easy updates. As a result, data integration and browsing through complex data become easier and more efficient.

2.3 RDF and RDF Schema

RDF stands for the Resource Description Framework. It is a W3C data integration format developed and agreed upon. While there are many standard tools to handle data and more specifically to handle the relationships between data. RDF is the simplest, most efficient standard that has been designed as of now.

The two W3C standards, RDF and RDF Schema (RDFS) are designed to improve the Web with machine-processable semantic data. As a data modelling vocabulary for RDF data, schema data has been introduced. The first public draft of the scheme was published in 1998 [13] and adopted it in 2004 as a recommendation [14].

RDF is based on making statements in the form of subject, predicate and object expressions (i.e. triples) about resources, which in this case is described as the Web. The subject denotes the resource and the predicate expresses the relation between the subject and the object. RDF is a conceptual data-modelling tool in general [15].

RDFS provides a way for the classification and relation between related groups of resources shown in the main classes below. It includes several other properties such as `rfds:label`, `rdfs:comment`, `rdfs:domain`, `rdfs: range`, `rdf: type`, `rdfs: subClassOf` and `rdfs: subPropertyOf`.

The RDFS describes main classes as [16]:

- `rdfs:Resource`: the class of everything

- `rdfs:Class`: the class of all classes
- `rdfs:Literal`: the class of all literal values
- `rdfs:Datatype`: the class of all datatypes
- `rdf:Property`: the class of all properties

In general, RDFS is a semantic extension of RDF depicted in Fig 2.3. A person may have a vehicle and a minivan as an RDF instance, RDFS may specify that MotorVehicle is a class, and MiniVan is a subclass of a MotorVehicle. There are also three other subclasses of MotorVehicle, namely PassengerVehicle, Truck and Van. Then, define a class MiniVan as a subclass of both Van and PassengerVehicle [17].

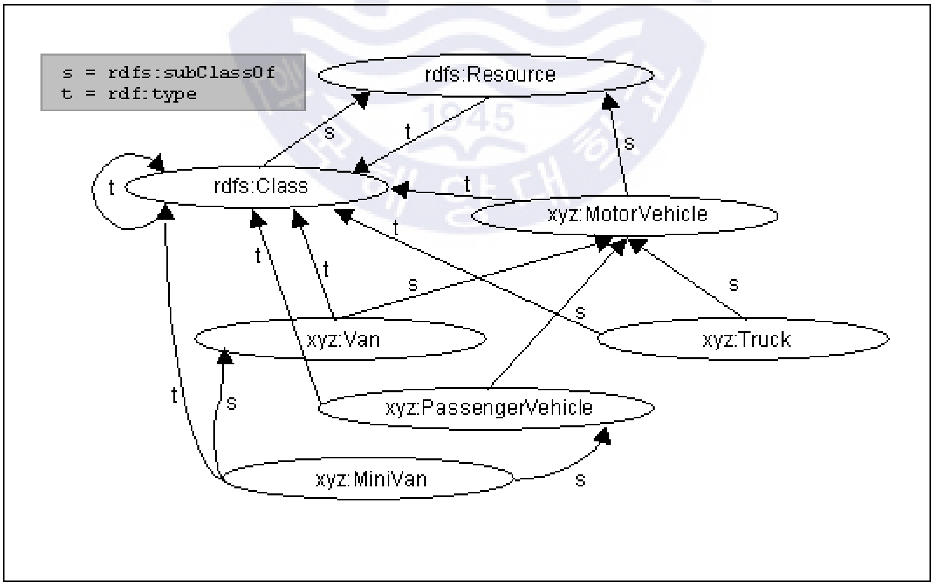


Fig. 2.3 An example of RDF Semantic link graph [17]

As shown in Fig. 2.4 is the sample of an RDF serialization syntax. On a side note, this RDF schema would typically be used in RDF instance data by referencing it with an XML namespace declaration, for example, `xmlns:xyz="http://www.w3.org/2000/03/example/vehicles#"`. This allows the use of abbreviations such as xyz:MotorVehicle to refer directly to the RDF class 'MotorVehicle'.

Query. A sample RDF syntax of a Motor Vehicle

```
<rdf:RDF xml:lang="en"
   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
   xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#">

<rdf:Description ID="MotorVehicle">
 <rdf:type resource="http://www.w3.org/2000/01/rdf-
schema#Class"/>
 <rdfs:subClassOf
  rdf:resource="http://www.w3.org/2000/01/rdf-
schema#Resource"/>
</rdf:Description>

<rdf:Description ID="PassengerVehicle">
 <rdf:type resource="http://www.w3.org/2000/01/rdf-
schema#Class"/>
 <rdfs:subClassOf rdf:resource="#MotorVehicle"/>
</rdf:Description>

<rdf:Description ID="Truck">
 <rdf:type resource="http://www.w3.org/2000/01/rdf-
schema#Class"/>
 <rdfs:subClassOf rdf:resource="#MotorVehicle"/>
</rdf:Description>

<rdf:Description ID="Van">
 <rdf:type resource="http://www.w3.org/2000/01/rdf-
schema#Class"/>
 <rdfs:subClassOf rdf:resource="#MotorVehicle"/>
</rdf:Description>

<rdf:Description ID="MiniVan">
 <rdf:type resource="http://www.w3.org/2000/01/rdf-
schema#Class"/>
 <rdfs:subClassOf rdf:resource="#Van"/>
 <rdfs:subClassOf rdf:resource="#PassengerVehicle"/>
```

```
</rdf:Description>

</rdf:RDF>
```

Fig. 2.4 The sample of RDF serialization syntax [17]

2.4 SPARQL

SPARQL Protocol and RDF Query Language (SPARQL) is originally used to query graph data represented as RDF triples. Together, SPARQL and RDF make it easier to merge results from multiple data sources. It is designed to enable Linked Data for the Semantic Web and to enrich data by linking it to other global semantic resources. Thus, sharing, merging and reusing data are performed in a more meaningful way [18].

According to Cambridge Semantics, the differences between SPARQL and SQL is that both languages give the user access to create, combine, and consume structured data. SQL access tables in relational databases and SPARQL access a web of Linked Data. SPARQL can be used to access relational data as well but was designed to merge disparate sources of data. Moreover, queries can be expressed across a range of datasets such as documents or any relevant information that could be presented as a directed labelled graph. The results of SPARQL queries can be in the form of sets or RDF graphs where the data is stored locally or viewed as RDF by middleware [19].

Relational data made up of rows of data collected into tables, which also called a "relations" in formal relational literature. The rows in a table conform to a set data types and constraints called a schema [19].

Fig. 2.5 describe the structure of a SPARQL query where ① shows the PREFIX definition which will enable a compact URI of the related sites in the query. ② is an example of a SELECT query which will retrieve variables. ③ is the RDF triple pattern of WHERE clause specifying the graph pattern matching with the conditions.

```
prefix xsd:  <http://www.w3.org/2001/XMLSchema#>     ①
prefix vs:  <https://portmis.go.kr/>

SELECT ?ship ?port   ②
WHERE {
   ?x vs:ShipName ?ship .
   FILTER regex(str(?port), 'BUSAN')   ③
   ?x vs:PortName ?port .
}
```

Fig. 2.5 The structure of SPARQL query

2.5 Apache Jena Fuseki as RDF Triplestore

RDF Triplestore is a special database system for data management in RDF, storing an interconnected data and gathering new facts out of the existing ones. Triplestore provides a standard SPARQL query language. By combining the provided primitives, users can define unique patterns and restrictions. A dataset is used to parse and transform SPARQL queries, while Fuseki provides HTTP access to SPARQL.

Apache Jena Fuseki is a component of a SPARQL server for other RDF query and storage systems with the integration of Jena TDB [20, 21]. Fuseki runs in the background as an embedded database program. An embedded database is enabled to provide support for the development and testing of a triplestore. It provides access control at the level on the server, on datasets, on endpoints and also on specific graphs within a dataset. In addition, local HTTPs also protect data in motion. A dataset is secured from corruption, unexpected process terminations and program crashes when accessed using TDB transactions. A TDB dataset must be accessed directly from a single JVM at a time or data corruption may occur [22, 23, 24].

Chapter 3 RDF Schema Design

3.1 RDFS for Vessels

According to FleetMon [25], the world's first terrestrial Automatic Identification System (AIS) data collection was in 2007. Satellite AIS data from 2013 onwards, the data were made available to access historical positions, schedules and port arrivals, trading patterns, static AIS data, port calls and database extracts.

Vessel information may not only consist of the technical specifications and management information, but it may also include the real-time AIS vessel tracking service. In the study of this thesis, two different datasets were analyzed after it was collected from various website across the world to study the RDF data retrieval.

3.1.1 Vessel Information Structure

In order to improve a new concept of searching through the Web of Data, the structure design of datasets focuses on the retrieved data based on the type of information the user intend to look for. However, the vessel structure is divided into few categories.

The general scheme consists of six main categories, which are historical data, machinery, performances, capacities, general, and specifications are shown in Fig. 3.1. In every category, contains every detail describing the vessel technical information.

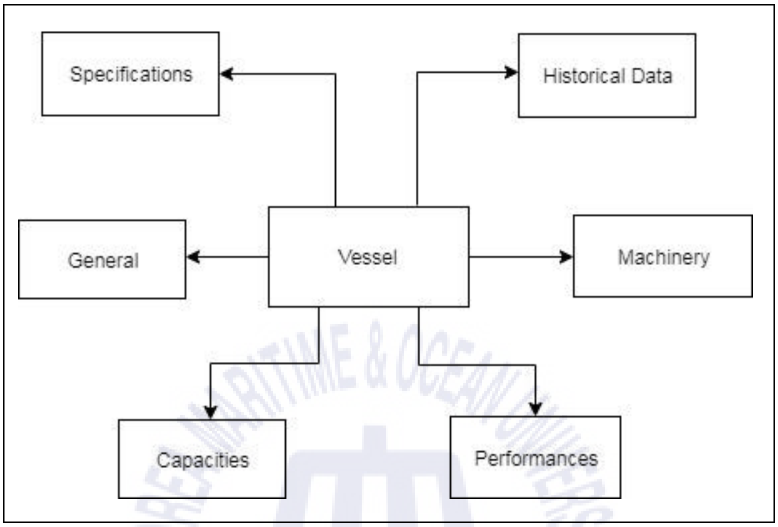


Fig. 3.1 General scheme of the vessel information structure

In the first two category described in Fig. 3.2 the historical data are vessel name, port name, call sign, arrival date, departure date, MMSI, flag which holds the data from the previous location and machinery shows the basic engine information of the vessel such as mains, generators and power of the vessels.

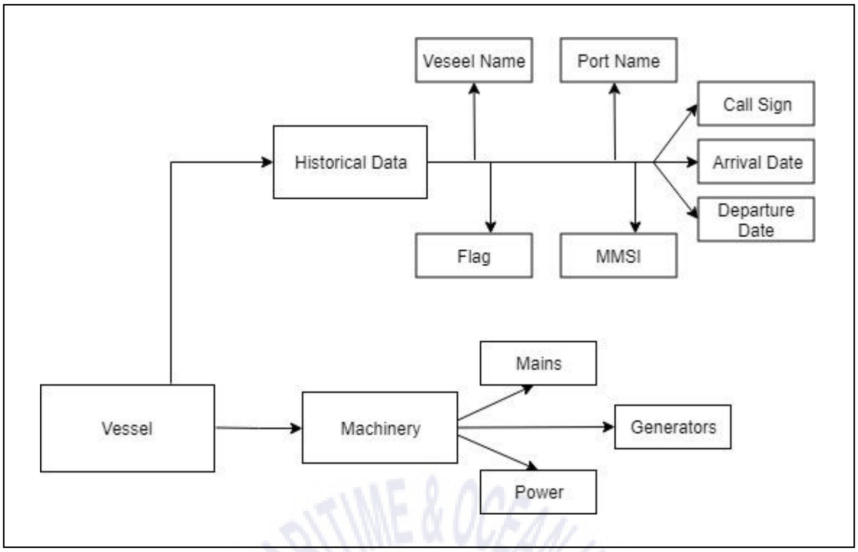


Fig. 3.2 Historical data and machinery

Next, Fig. 3.3 shows the vessel performances and capabilities where details such as fuel consumption, speed, weight and overall volume along with size of the vessel are accessible for data searching and data retrieval. While the final two categories include general information such as the MMSI number, type of vessel, year it was built, name, owner and etc. Whereas, the last category are specifications for instance IMO number, port of registry, class, callsign and others are best described in the Fig. 3.4.

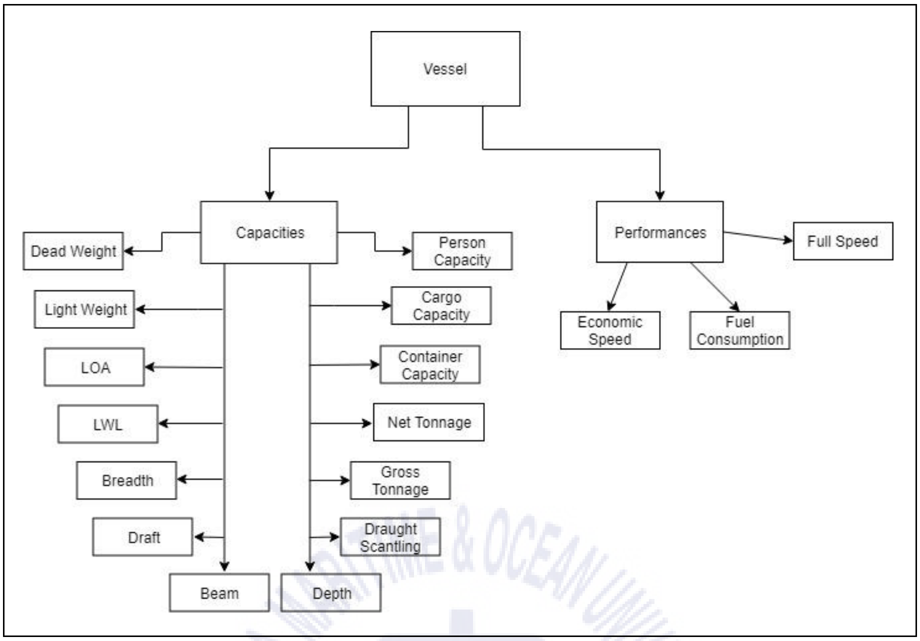


Fig. 3.3 Performances and capacities

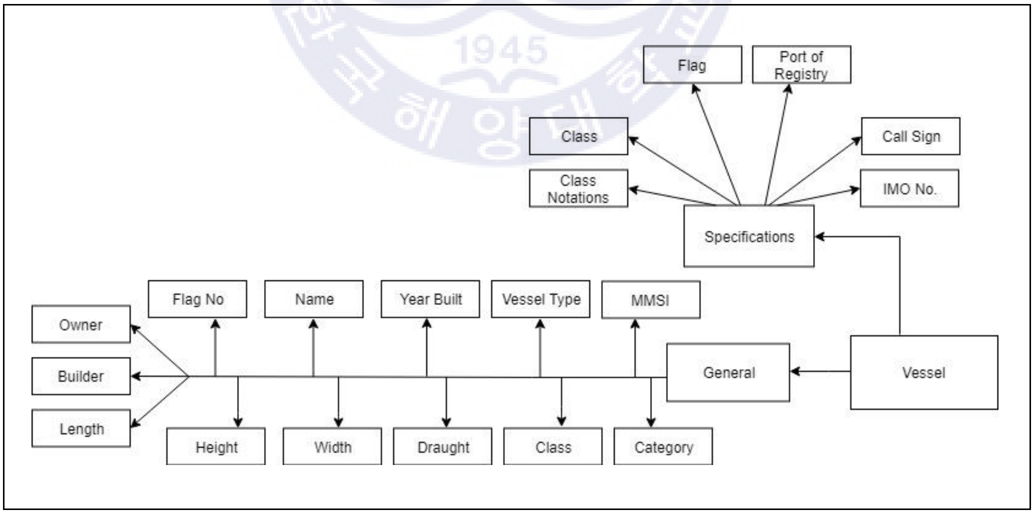


Fig. 3.4 General and specifications

3.1.2 Vessel Information Structure based on Port-MIS

With reference to the marine sector, this thesis studies about the links between datasets and how it should officially document and used at its best. South Korea has been known for dominant hub function in the region of Northeast Asia's logistics hub [26]. Within the marine data management, the availability of such vocabularies are listed as Linked Data and Semantic Web resources through the Port Management Information System (Port-MIS) which is a Korean based site.

Port-MIS is an information system that has processes all administration tasks such as movement, entry and departure of vessels, cargo entering and carrying as well as tax collection that occurs in thirty-one trade ports around the country. The Port-MIS was transformed to a Web-based system. Opening ways to provide a support structure for real-time information, many different forms of civil complaints systems and advanced civil complaint services through wired and wireless internet [27].

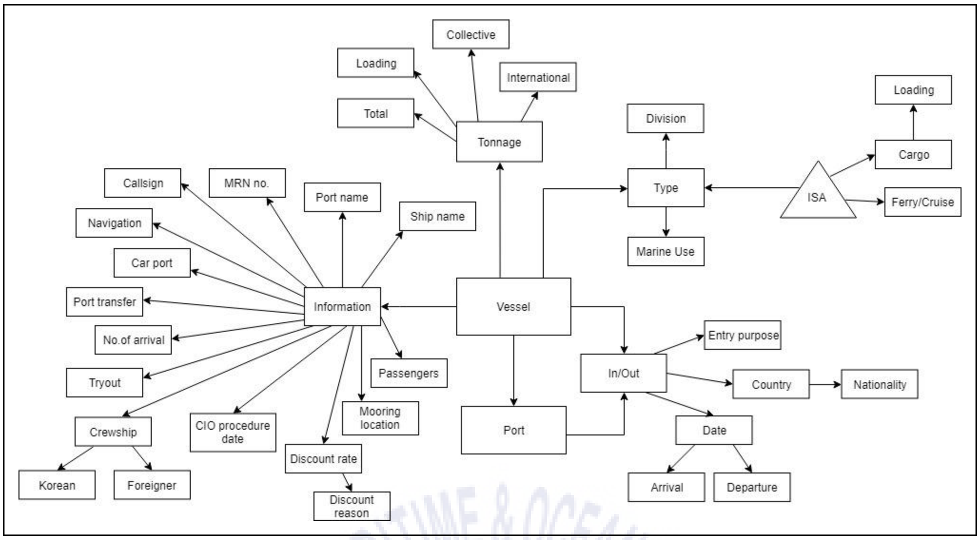


Fig. 3.5　Vessel information structure based on Port-MIS

As shown in Fig. 3.5, the diagram shows how the main subject, vessel is connected to other information related to another instances. For example, its type, tonnage information, related port, traffic flows and other information about the vessel.

The following Fig. 3.6 shows the RDF Schema diagram which shows one major RDF resources; *Vessel*. The `Type` resources are subclasses of the *Vessel* resource. Each instance of these resources has a URI associated with it. Resource related properties are also defined in this schema. The ranges of some properties are resources from other domain themselves. For example, resources *Port* has a property *isConnectedBy*. This property has its domain in the *Port* but its range is in the *In/Out* domain. The schema helps to define the data model making retrieving data more efficient and follow relationship links specified with it [28].

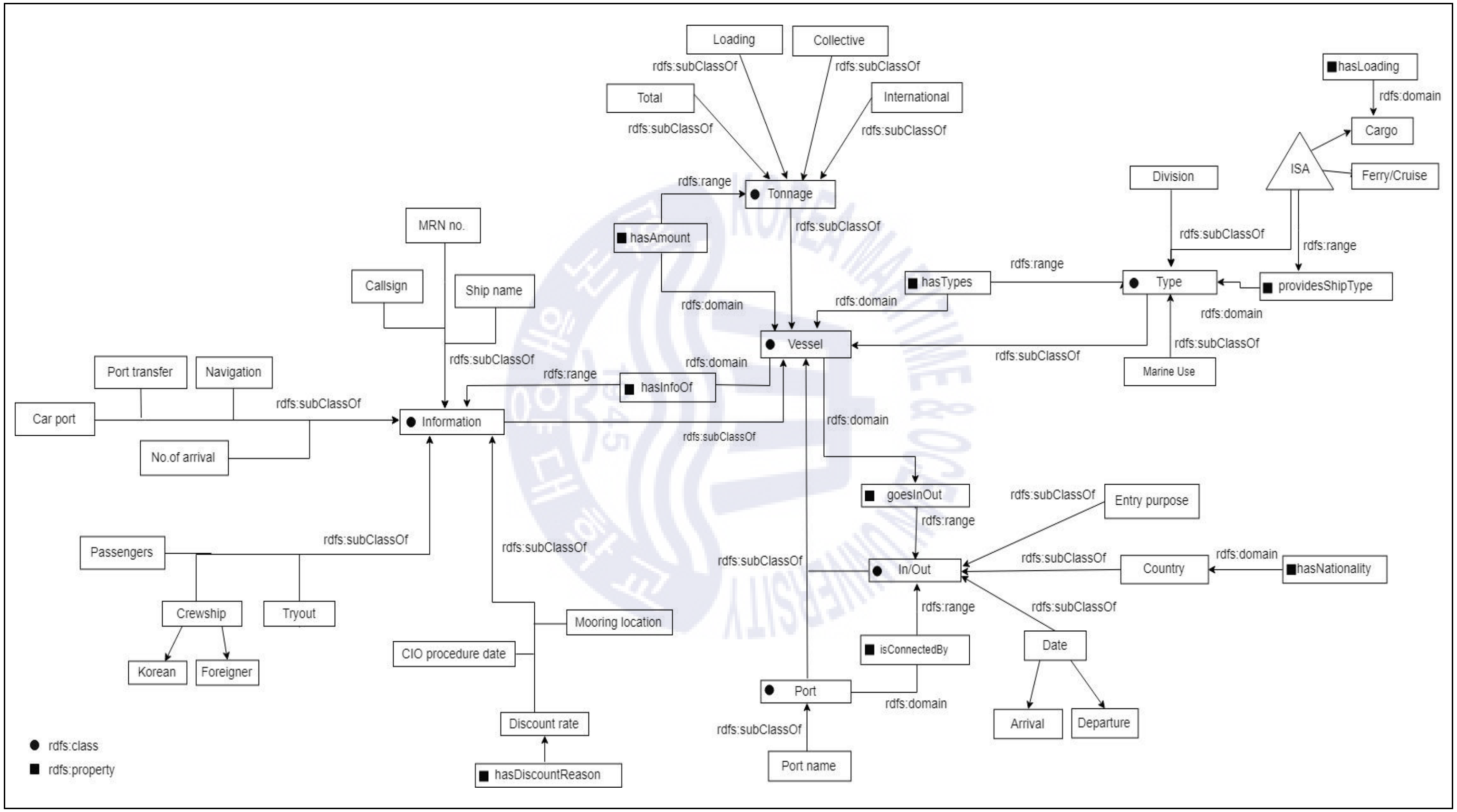


Fig. 3.6 RDF Schema of vessel based on Port-MIS

3.1.3 RDF Sample Syntax of Port-MIS

Fig. 3.7 depicts a part of the RDF schema of vessel based on Port-MIS where a sample was taken to further understand on how it is structured and implemented in the RDF storage system. For example, in Table 3.1, the subject *Vessel* has a predicate *hasTypes* and an object *ISA,* then *ISA* becomes the subject with a predicate *providesShipType* for the object *Cargo* and so forth, it was translated in the form of triples format to make the information readable and understandable through the relationship link graph. A sample of syntax of Port-MIS in Fig 3.8 was constructed to enable the relation of the ship type and its loading information to each other and for easy searching in between the interlinked data.

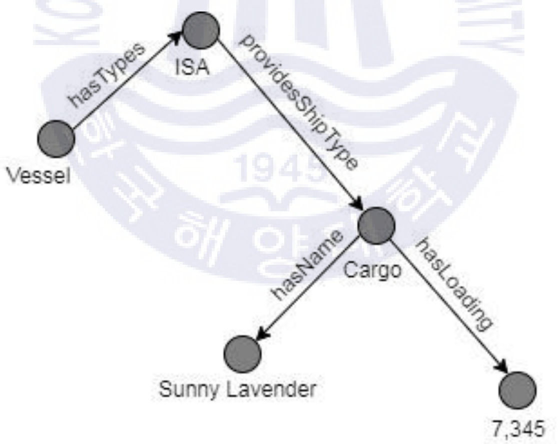


Fig. 3.7 RDF graph of ship type and loading information

Table 3.1 Triples in an RDF database (Port-MIS)

Subject	Predicate	Object
Vessel	hasTypes	ISA
ISA	providesShipType	Cargo
Cargo	hasName	Sunny Lavender
Cargo	hasLoading	7,345

Query. A sample RDF syntax of ship type and loading information

```
<?xml version="1.0"?>
<rdf:RDF xml:lang="en"
   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
   xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
   xmlns:vs="https://"new.portmis.go.kr/port#">

<rdf:Description ID="Vessel">
 <rdf:type resource="http://www.w3.org/2000/01/rdf-
schema#Class"/>
 <rdfs:subClassOf
  rdf:resource="http://www.w3.org/2000/01/rdf-
schema#Resource"/>
</rdf:Description>

<rdf:Description ID="ISA">
 <rdf:type resource="http://www.w3.org/2000/01/rdf-
schema#Class"/>
 <rdfs:subClassOf rdf:resource="#Vessel"/>
</rdf:Description>

<rdf:Description ID="Cargo">
 <rdf:type resource="http://www.w3.org/2000/01/rdf-
schema#Class"/>
 <rdfs:subClassOf rdf:resource="#Vessel"/>
</rdf:Description>

<rdf:Description ID="Sunny Lavender">
 <rdf:type resource="http://www.w3.org/2000/01/rdf-
schema#Class"/>
 <rdfs:subClassOf rdf:resource="#Cargo"/>
</rdf:Description>

<rdf:Description ID="7,345">
```

```
 <rdf:type resource="http://www.w3.org/2000/01/rdf-
schema#Class"/>
 <rdfs:subClassOf rdf:resource="#Cargo"/>
 <rdfs:subClassOf rdf:resource="#Vessel"/>
</rdf:Description>

</rdf:RDF>
```

Fig. 3.8 RDF syntax to show a ship type and loading information

3.2 RDFS for Patent Data

According to Shopify business encyclopedia [29], patents are the right of an inventor to exclude everyone else from making, using or selling their invention for a period of 20 years. Once the patent owner is licensed it shall be the only individual or organization that is permitted to produce, use or offer to sell, process or solve a technological problem that has been developed and registered by it. Anybody else is prohibited unless permission is given.

Additionally, there are a total number of three types of patents which is design patent, utility patent and plant patent. This thesis will further discuss about the utility patents. A utility patent can be a machine, a process, a product or a combination of these three whereas it may also be related to design patent.

In the industry of patent data, a traditional keyword searches in patent datasets might be insufficient to get back all useful information, particularly in the engineering domain.

At first, to form the semantic web the system identifies the marine engineering ontologies. Apart from searching keywords and linking the identified properties with one or more triples of RDF, the system can detect semantic relations between the Web resources. When determining the semantic relation, the outcome of the result is also crucial to ensure the information outcome are not overloaded.

3.2.1 Patent Data Structure

The concept of patent data structure is important due to the number of people trying to find solutions to a certain technical problem. According to World Intellectual Property Organization (WIPO), every year there are approximately 750,000 patents granted all across the world [30]. Therefore, to find out whether some invention has been patented or not, a long time and research needs to be made upon the registered inventions. Thus, the patent data structure has been gathered across the web to ensure the data is not missed.

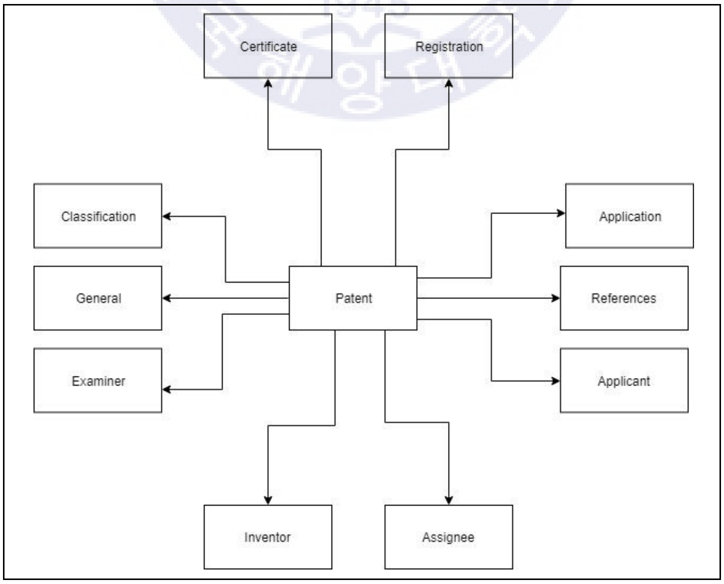


Fig. 3.9 General scheme of the patent data structure

Fig. 3.9 shows the scheme consists of ten main categories, which are certificate, registration, application, references, applicant, assignee, inventor, examiner, general information and classification. In each category, the patent information and its attributes has been assigned accordingly.

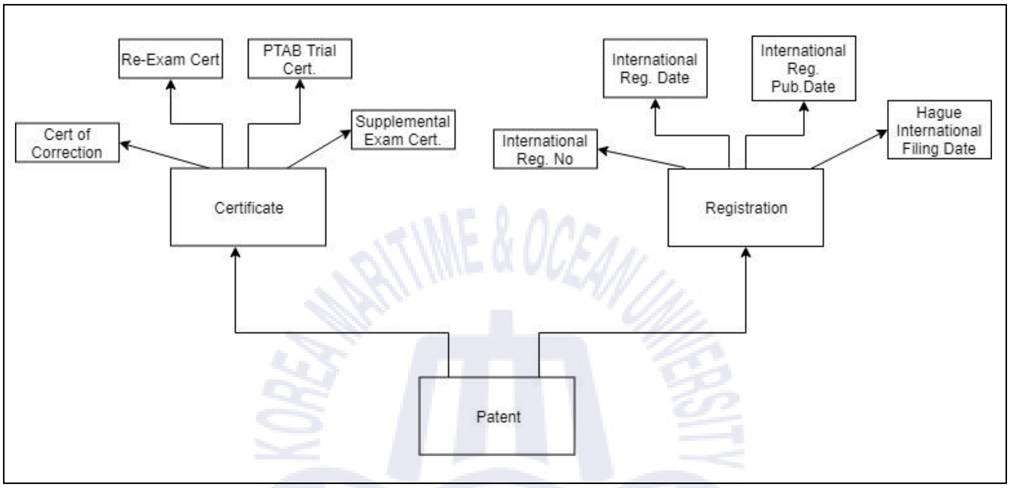


Fig. 3.10 Certificate and registration

The two of ten categories are illustrated in Fig. 3.10, certificate and registration holds information such as certificate of correction, certificate of re-examination, certificate of PTAB trial and certificate of supplemental exam also included the international details of registration number, registration date, publication date and filing date.

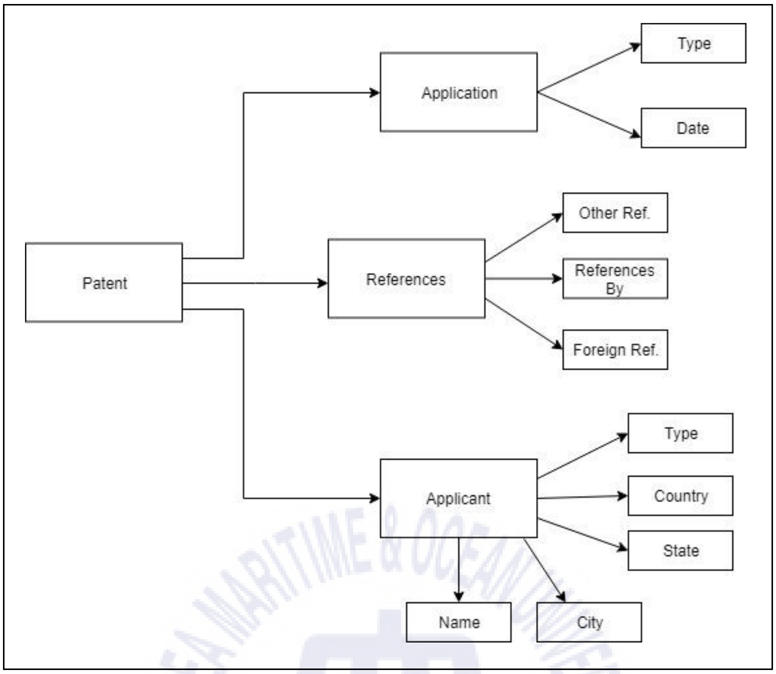


Fig. 3.11 Application, references and applicant

The next three categories shown in Fig. 3.11 may involve the basic information of application details, refences and applicant lists namely application type, application date, other reference, reference by, foreign reference, applicant type, country state, city, and name. This information is especially useful as it could indicate a demand for patent rights in one or more countries.

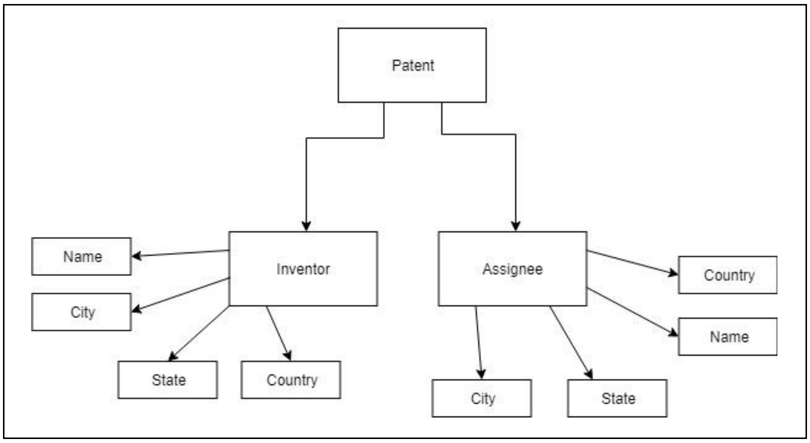


Fig. 3.12 Inventor and assignee

As shown in Fig. 3.12 are two other categories, inventor and assignee that both has the same relatable attributes which is name, country, city, and state respectively. The assignee is the person with the property right to the patent. Whereas, inventor can be the one who has developed a particular invention and can be either one person or more.

Finally, the last three categories may imply the important details in patent data as seen on Fig. 3.13. Illustrated are classification of current CPC, current CPC class and international classification. Included are patent number, issue date, title, claim(s), abstract, description, attorney/agent, prior. filing date, reissue data, reexamined patent app. filing date, prior publication document date, other informational filing and the Patent Cooperation Treaty. Besides, examiner is divided into two, primary examiner and assistant examiner. Each plays an important role. Such role includes approving inventions by reading and understanding every specification as well as searching for existing invention technologies.

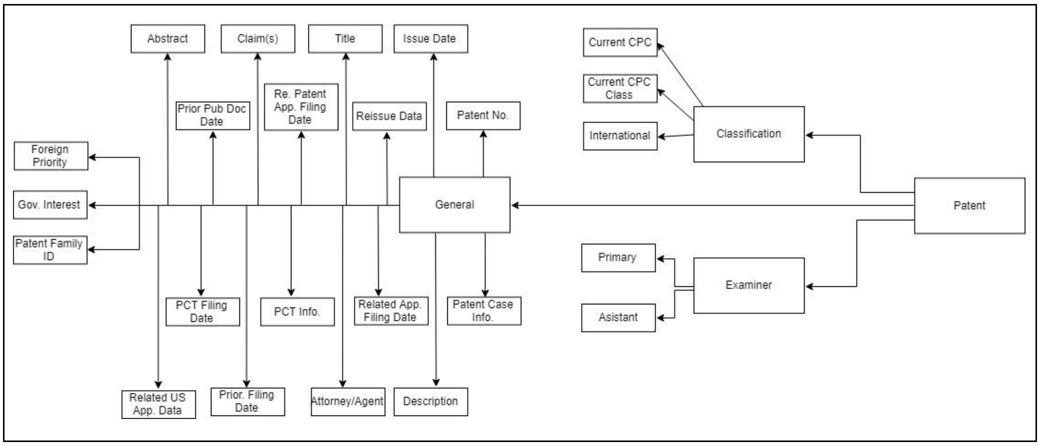


Fig. 3.13 Classification, general and examiner

3.2.2 Patent Data Structure based on KIPRIS

For this thesis, the patent datasets structures were collected from Korea Intellectual Property Rights Information Service (KIPRIS). KIPRIS is Korea's Internet most detailed free search site for intellectual property data. KIPRIS includes every IP information of the Korean intellectual property office (KIPO) in Korea and administered on behalf of KIPO by the Korea Institute of Patent Information (KIPI).

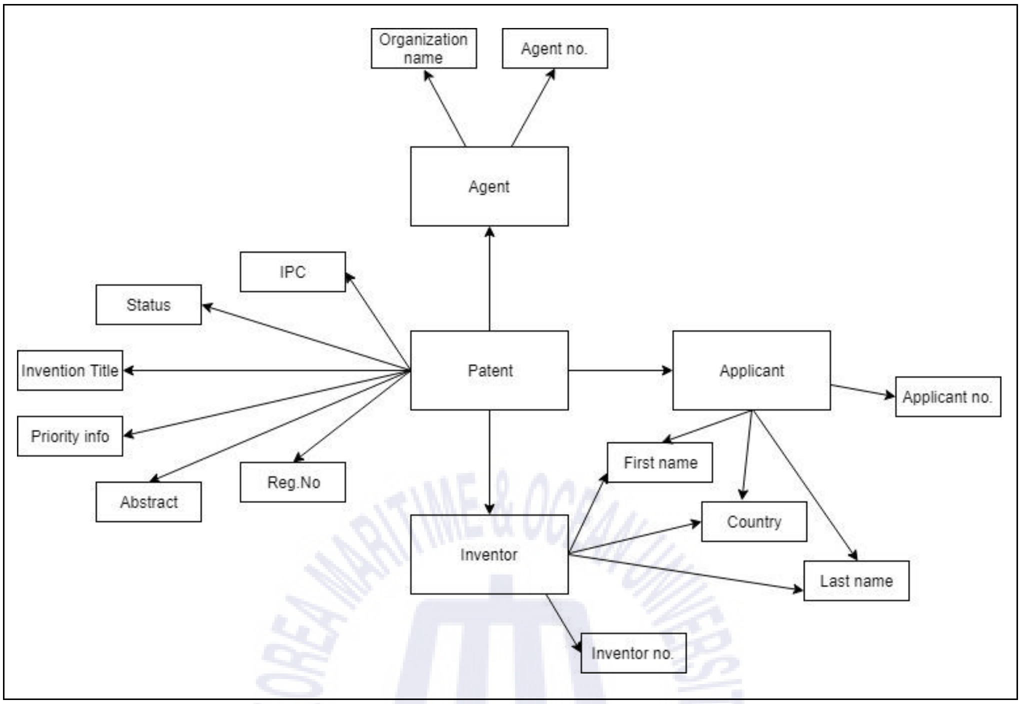


Fig.3.14 Patent data structure based on KIPRIS

Firstly, the components of diagram are shown in Fig. 3.14. The main subject *Patent* has three other subclasses and its data are interlinking with each other. As shown as an example of RDF Schema in Fig. 3.15, the *Agent* resource are a subclass of *Patent* which then has two other two subclasses, *Organization name* and *Agent no.*, a domain and range property *isUndertakenBy* were connected in between the resources *Patent* and *Agent*.

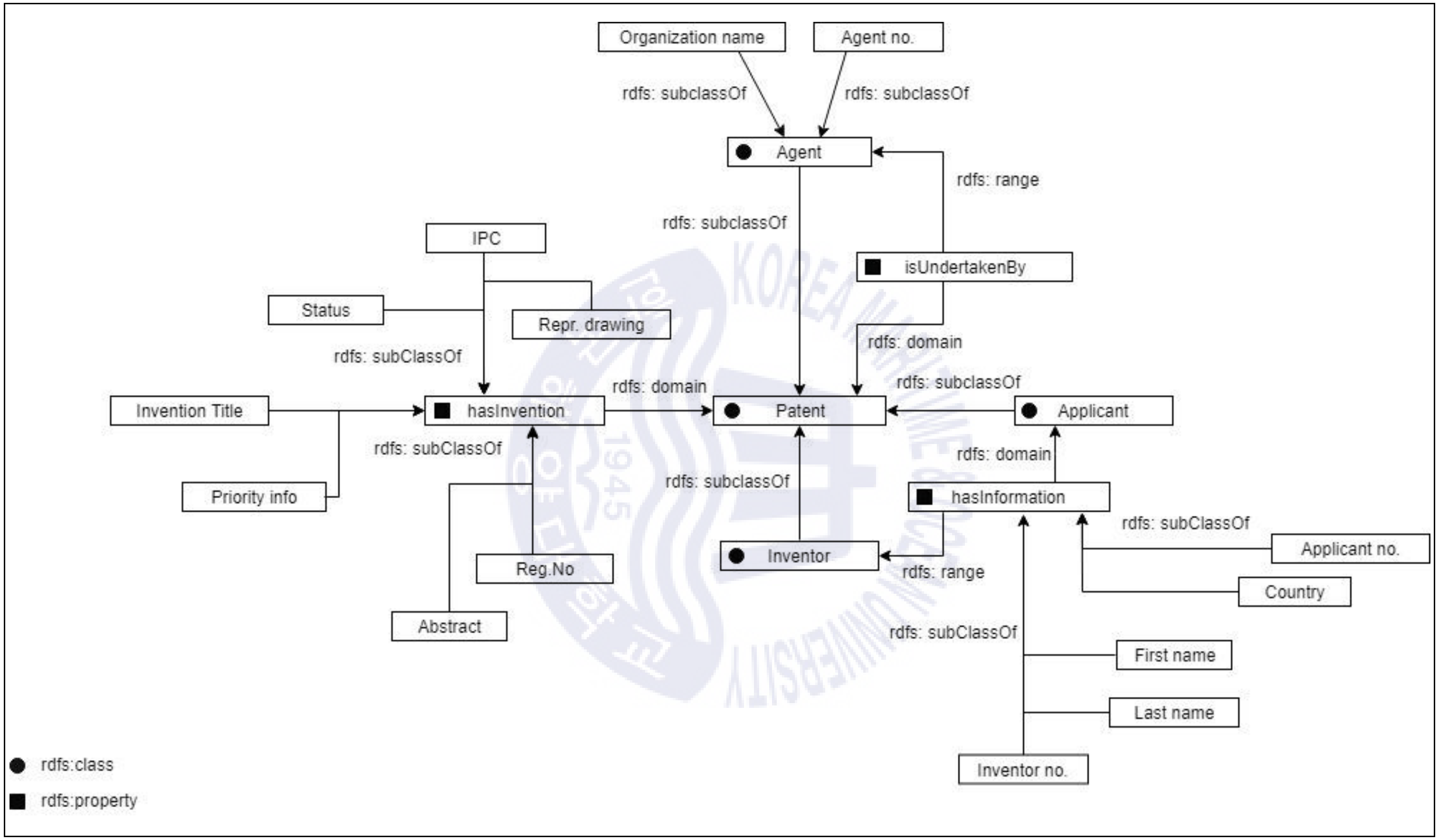


Fig. 3.15 RDF Schema of a patent data based on KIPRIS

3.2.3 RDF Sample Syntax of KIPRIS

Fig. 3.16 depicts an RDF graph of the applicant STX Offshore & Shipbuilding Co. Ltd. in displaying the particular information on the link graph of KIPRIS. Generally, the format for graph, triples and query has a similar characteristic as any other input data respectively. Table 3.2 best explain the graph in triples format where the subject *Applicant* has a predicate *hasApplicantionNo* and an object *1020100060169*. The query in Fig 3.17 was constructed based on a particular applicant's information in a precise manner.

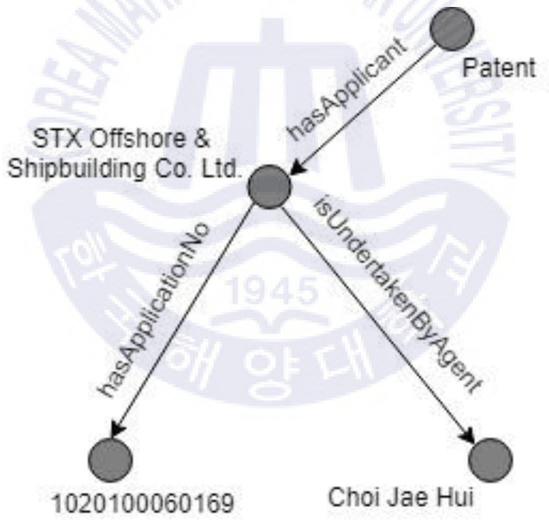


Fig. 3.16 RDF graph of patent information

Table 3.2 Triples in an RDF database (KIPRIS)

Subject	Predicate	Object
Patent	hasApplicant	STX Offshore & Shipbuilding Co. Ltd.
Applicant	isUndertakenByAgent	Choi Jae Hui
Applicant	hasApplicationNo	1020100060169

Query. A sample RDF syntax of applicant's information

```
<?xml version="1.0"?>
<rdf:RDF xml:lang="en"
   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
   xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
   xmlns:pt="http://eng.kipris.or.kr/patent#">

<rdf:Description ID="Patent">
 <rdf:type resource="http://www.w3.org/2000/01/rdf-
schema#Class"/>
 <rdfs:subClassOf
  rdf:resource="http://www.w3.org/2000/01/rdf-
schema#Resource"/>
</rdf:Description>

<rdf:Description ID="Applicant">
 <rdf:type resource="http://www.w3.org/2000/01/rdf-
schema#Class"/>
 <rdfs:subClassOf rdf:resource="#Patent"/>
</rdf:Description>

<rdf:Description ID="STX Offshore & Shipbuilding Co. Ltd.">
 <rdf:type resource="http://www.w3.org/2000/01/rdf-
schema#Class"/>
 <rdfs:subClassOf rdf:resource="#Applicant"/>
</rdf:Description>

<rdf:Description ID="Choi Jae hui">
 <rdf:type resource="http://www.w3.org/2000/01/rdf-
schema#Class"/>
 <rdfs:subClassOf rdf:resource="#Applicant"/>
</rdf:Description>

<rdf:Description ID="Application No">
```

```
 <rdf:type resource="http://www.w3.org/2000/01/rdf-
schema#Class"/>
 <rdfs:subClassOf rdf:resource="#Applicant"/>
 <rdfs:subClassOf rdf:resource="#Agent"/>
</rdf:Description>

</rdf:RDF>
```

Fig. 3.17 RDF syntax to show applicant's information

Chapter 4 Implementations and Testing

4.1 System Architecture

A conceptual architecture for RDF metadata information retrieval system is shown in Fig. 4.1. As illustrated, user constructs query and accesses the SPARQL server to get it to process the conditions while running through linked datasets to collect graph pattern matching with the datasets stored in Apache Jena Fuseki. Then, datasets will be analyzed thoroughly. Lastly, returns a list of resources with matching semantic metadata as result.

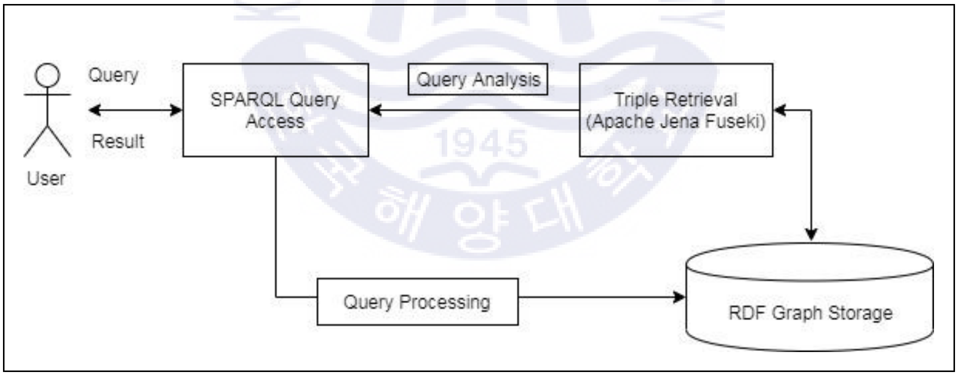


Fig. 4.1 Concept architecture for RDF metadata information retrieval system

4.2 Data Processing Structure

From the year 2007, gathering data from a particular open-source site ensure that not only the Wikipedia information can be linked together to create a new generated Linked Data but also include the educational contents such as the IEEE Learning Object Metadata (IEEE LOM) [31]. Port-MIS and KIPRIS show the purpose of two different sectors being connected in terms of the marine engineering sector in developing a new generation of structured data through the Internet.

The illustration below, Fig. 4.2 describe the whole process of how from a raw uncategorized data being analyze and convert into a directed linked graph format that is readable for both humans and machines.

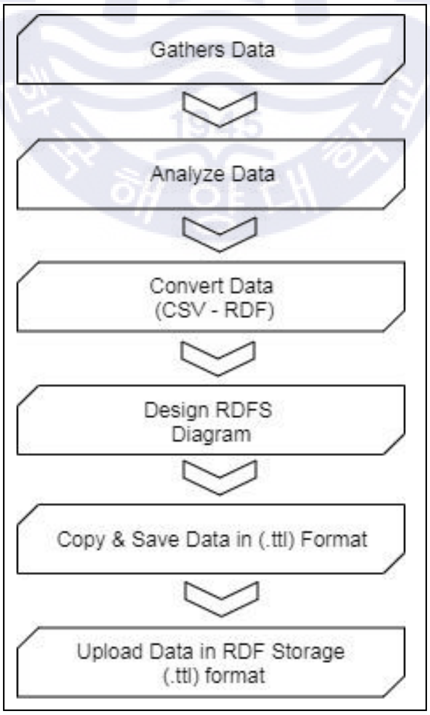


Fig. 4.2 Structural process of data management

Shown in Fig. 4.3, is the structural process of queries management which occur after the process of managing the raw data which was gathered previously. In the query management process, in order to start all the constructions of queries, the server has to run in the background. Constructions of queries are needed in order to obtain data from RDF storage and to populate the ontology and lastly to display correct and reliable results semantically.

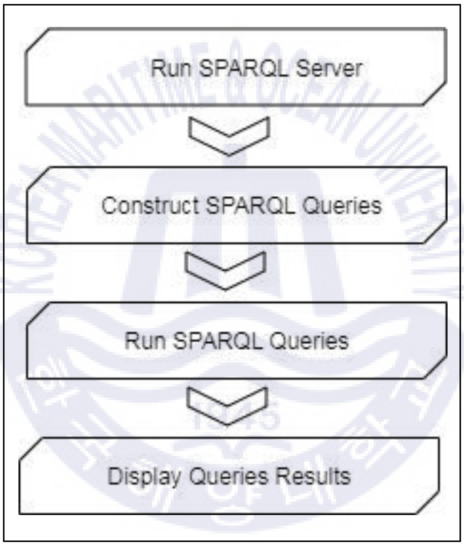


Fig. 4.3 Structural process of queries management

4.3 SPARQL Queries

A linked data RDF schema are designed from a particular open-source site based on Port-MIS and KIPRIS for the purpose of developing a new generation of queries as a whole. As well as develop a searching structured data through semantic queries for it to be presented in the RDF data retrieval system. Fig. 4.4 and Fig. 4.5 shows the queries and result of vessel information based on Port-MIS which performs in a well-structured and efficient manner. Retrieving the information from RDF storage is done within the SPARQL server and the triplestore.

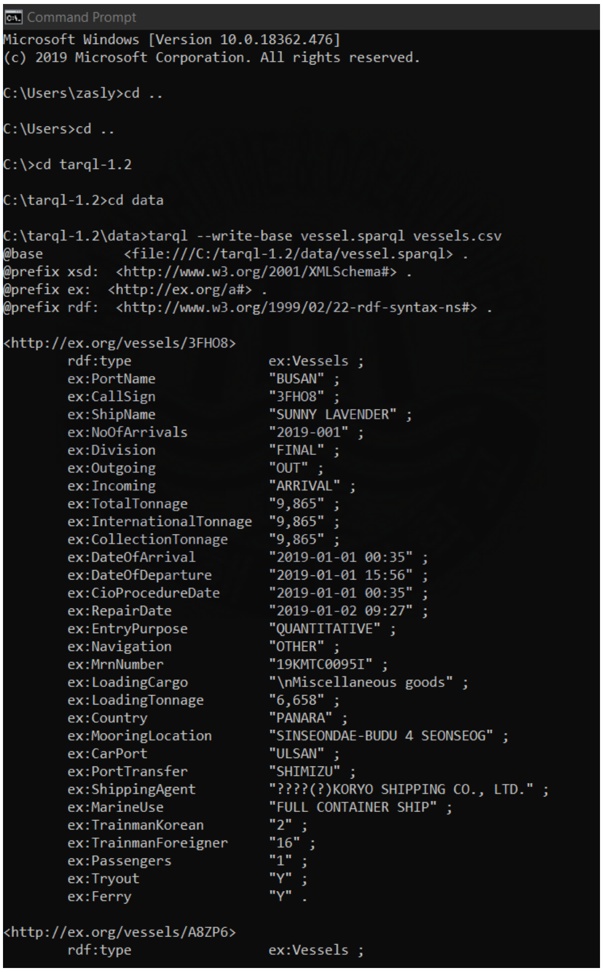
Query. Vessel Information based on Port-MIS

```
PREFIX pt:  <http://eng.kipris.or.kr/enghome/main.jsp>
PREFIX vs:  <https://new.portmis.go.kr/>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>

CONSTRUCT {
  ?URI a vs:Vessels;
    vs:PortName ?PortName;
    vs:CallSign ?CallSign;
    vs:ShipName ?ShipName;
    vs:NoOfArrivals ?NoOfArrivals;
    vs:Division ?Division;
    vs:Outgoing ?Outgoing;
    vs:Incoming ?Incoming;
    vs:TotalTonnage ?TotalTonnage;
    vs:InternationalTonnage ?InternationalTonnage;
    vs:CollectionTonnage ?CollectionTonnage;
    vs:DateOfArrival ?DateOfArrival;
    vs:DateOfDeparture ?DateOfDeparture;
    vs:CioProcedureDate ?CIOProcedureDate;
    vs:RepairDate ?RepairDate;
    vs:EntryPurpose ?EntryPurpose;
    vs:Navigation ?Navigation;
    vs:MrnNumber ?MRNNumber;
    vs:LoadingCargo ?LoadingCargo;
    vs:LoadingTonnage ?LoadingTonnage;
    vs:Country ?Country;
    vs:MooringLocation ?MooringLocation;
    vs:ShipPlaceA ?ShipPlaceA;
    vs:ShipPlaceB ?ShipPlaceB;
    vs:DiscountRate ?DiscountRate;
    vs:DiscountReason ?DiscountReason;
    vs:CarPort ?CarPort;
    vs:PortTransfer ?PortTransfer;
    vs:ShippingAgent ?ShippingAgent;
    vs:MarineUse ?MarineUse;
    vs:TrainmanKorean ?TrainmanKorean;
    vs:TrainmanForeigner ?TrainmanForeigner;
    vs:Passengers ?Passengers;
    vs:Tryout ?Tryout;
```

```
    vs:Ferry ?Ferry;
}
FROM <file:vessels.csv>
WHERE {
  BIND (URI(CONCAT('https://new.portmis.go.kr/', ?CallSign))
AS ?URI)
}
```

Fig. 4.4 The query for vessel information based on Port-MIS



```
Command Prompt
Microsoft Windows [Version 10.0.18362.476]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\zasly>cd ..

C:\Users>cd ..

C:\>cd tarql-1.2

C:\tarql-1.2>cd data

C:\tarql-1.2\data>tarql --write-base vessel.sparql vessels.csv
@base          <file:///C:/tarql-1.2/data/vessel.sparql> .
@prefix xsd:  <http://www.w3.org/2001/XMLSchema#> .
@prefix ex:  <http://ex.org/a#> .
@prefix rdf:  <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .

<http://ex.org/vessels/3FHO8>
        rdf:type                 ex:Vessels ;
        ex:PortName              "BUSAN" ;
        ex:CallSign              "3FHO8" ;
        ex:ShipName              "SUNNY LAVENDER" ;
        ex:NoOfArrivals          "2019-001" ;
        ex:Division              "FINAL" ;
        ex:Outgoing              "OUT" ;
        ex:Incoming              "ARRIVAL" ;
        ex:TotalTonnage          "9,865" ;
        ex:InternationalTonnage  "9,865" ;
        ex:CollectionTonnage     "9,865" ;
        ex:DateOfArrival         "2019-01-01 00:35" ;
        ex:DateOfDeparture       "2019-01-01 15:56" ;
        ex:CioProcedureDate      "2019-01-01 00:35" ;
        ex:RepairDate            "2019-01-02 09:27" ;
        ex:EntryPurpose          "QUANTITATIVE" ;
        ex:Navigation            "OTHER" ;
        ex:MrnNumber             "19KMTC0095I" ;
        ex:LoadingCargo          "\nMiscellaneous goods" ;
        ex:LoadingTonnage        "6,658" ;
        ex:Country               "PANARA" ;
        ex:MooringLocation       "SINSEONDAE-BUDU 4 SEONSEOG" ;
        ex:CarPort               "ULSAN" ;
        ex:PortTransfer          "SHIMIZU" ;
        ex:ShippingAgent         "????(?)KORYO SHIPPING CO., LTD." ;
        ex:MarineUse             "FULL CONTAINER SHIP" ;
        ex:TrainmanKorean        "2" ;
        ex:TrainmanForeigner     "16" ;
        ex:Passengers            "1" ;
        ex:Tryout                "Y" ;
        ex:Ferry                 "Y" .

<http://ex.org/vessels/A8ZP6>
        rdf:type                 ex:Vessels ;
```

Fig. 4.5 SPARQL query result of vessel information based on Port-MIS (in CMD)

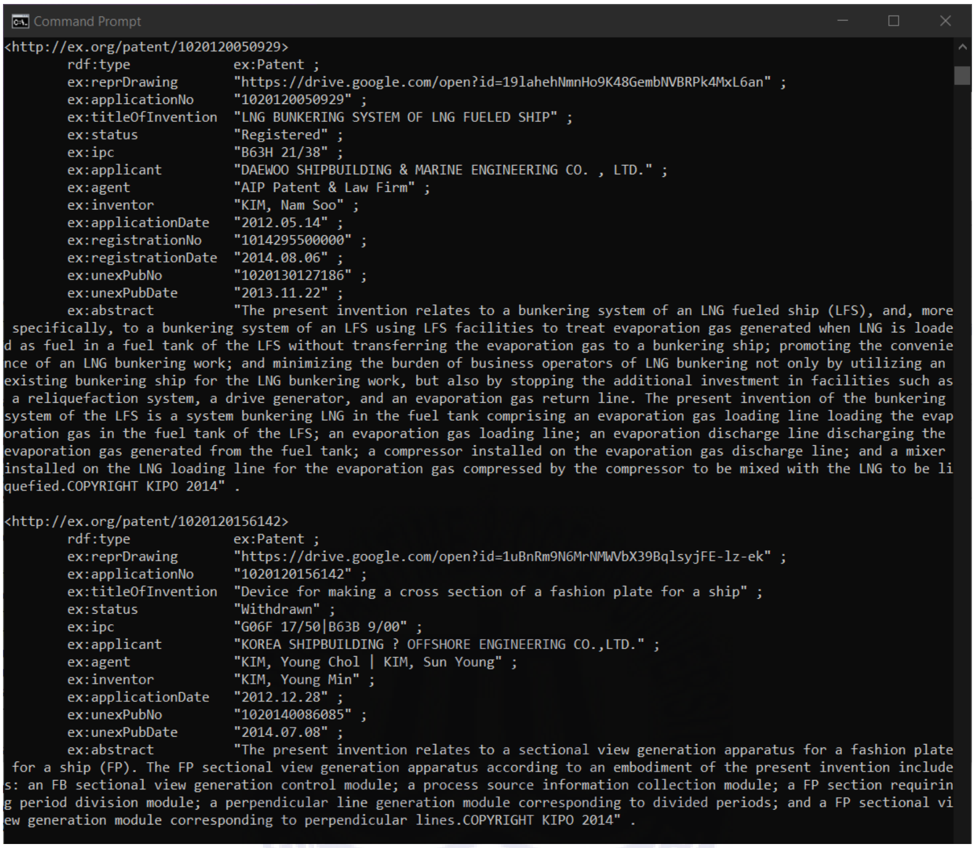
Fig. 4.6 shows the patent data information based on KIPRIS. In the queries below the listed are the values from the data gathered in the sites while Fig 4.7 displays the result of datasets stored in the RDF storage.

Query. Patent Data based on KIPRIS

```
# Define the link of the data as prefix
PREFIX pt: <http://eng.kipris.or.kr/enghome/main.jsp#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>

# Construct query CONSTRUCT define the tables, FROM define the file, WHERE define to
bind each variable
CONSTRUCT {
  ?URI a ex:Patent;
    pt:reprDrawing ?ReprDrawing;
    pt:applicationNo ?ApplicationNo;
    pt:titleOfInvention ?TitleOfInvention;
    pt:status ?Status;
    pt:ipc ?IPC;
    pt:applicant ?Applicant;
    pt:agent ?Agent;
    pt:inventor ?Inventor;
    pt:applicationDate ?ApplicationDate;
    pt:registrationNo ?RegistrationNo;
    pt:registrationDate ?RegistrationDate;
    pt:unexPubNo ?UnexPubNo;
    pt:unexPubDate ?UnexPubDate;
    pt:abstract ?Abstract;
}
FROM <file:patents.csv>
WHERE {
  BIND
URI(CONCAT('http://eng.kipris.or.kr/enghome/main.jsp/', ?Appl
icationNo)) AS ?URI)
  BIND (xsd:integer(?ApplicationNo) AS ?appNo)
  BIND (xsd:integer(?RegistrationNo) AS ?regNo)
  BIND (xsd:integer(?UnexPubNo) AS ?upNo)
}
```

Fig. 4.6 The query for patent data based on KIPRIS



Command Prompt

```
<http://ex.org/patent/1020120050929>
        rdf:type             ex:Patent ;
        ex:reprDrawing       "https://drive.google.com/open?id=19lahehNmnHo9K48GembNVBRPk4MxL6an" ;
        ex:applicationNo     "1020120050929" ;
        ex:titleOfInvention  "LNG BUNKERING SYSTEM OF LNG FUELED SHIP" ;
        ex:status            "Registered" ;
        ex:ipc               "B63H 21/38" ;
        ex:applicant         "DAEWOO SHIPBUILDING & MARINE ENGINEERING CO. , LTD." ;
        ex:agent             "AIP Patent & Law Firm" ;
        ex:inventor          "KIM, Nam Soo" ;
        ex:applicationDate   "2012.05.14" ;
        ex:registrationNo    "1014295500000" ;
        ex:registrationDate  "2014.08.06" ;
        ex:unexPubNo         "1020130127186" ;
        ex:unexPubDate       "2013.11.22" ;
        ex:abstract          "The present invention relates to a bunkering system of an LNG fueled ship (LFS), and, more
 specifically, to a bunkering system of an LFS using LFS facilities to treat evaporation gas generated when LNG is loade
d as fuel in a fuel tank of the LFS without transferring the evaporation gas to a bunkering ship; promoting the convenie
nce of an LNG bunkering work; and minimizing the burden of business operators of LNG bunkering not only by utilizing an
existing bunkering ship for the LNG bunkering work, but also by stopping the additional investment in facilities such as
 a reliquefaction system, a drive generator, and an evaporation gas return line. The present invention of the bunkering
system of the LFS is a system bunkering LNG in the fuel tank comprising an evaporation gas loading line loading the evap
oration gas in the fuel tank of the LFS; an evaporation gas loading line; an evaporation discharge line discharging the
evaporation gas generated from the fuel tank; a compressor installed on the evaporation gas discharge line; and a mixer
installed on the LNG loading line for the evaporation gas compressed by the compressor to be mixed with the LNG to be li
quefied.COPYRIGHT KIPO 2014" .

<http://ex.org/patent/1020120156142>
        rdf:type             ex:Patent ;
        ex:reprDrawing       "https://drive.google.com/open?id=1uBnRm9N6MrNMWVbX39BqlsyjFE-lz-ek" ;
        ex:applicationNo     "1020120156142" ;
        ex:titleOfInvention  "Device for making a cross section of a fashion plate for a ship" ;
        ex:status            "Withdrawn" ;
        ex:ipc               "G06F 17/50|B63B 9/00" ;
        ex:applicant         "KOREA SHIPBUILDING ? OFFSHORE ENGINEERING CO.,LTD." ;
        ex:agent             "KIM, Young Chol | KIM, Sun Young" ;
        ex:inventor          "KIM, Young Min" ;
        ex:applicationDate   "2012.12.28" ;
        ex:unexPubNo         "1020140086085" ;
        ex:unexPubDate       "2014.07.08" ;
        ex:abstract          "The present invention relates to a sectional view generation apparatus for a fashion plate
 for a ship (FP). The FP sectional view generation apparatus according to an embodiment of the present invention include
s: an FB sectional view generation control module; a process source information collection module; a FP section requirin
g period division module; a perpendicular line generation module corresponding to divided periods; and a FP sectional vi
ew generation module corresponding to perpendicular lines.COPYRIGHT KIPO 2014" .
```

Fig. 4.7 SPARQL query result of patent data based on KIPRIS (in CMD)

The followings are the system interfaces which demonstrates the program. Fig. 4.8 presents the dataset page that has uploaded data of patent and vessel information in the Turtle (.ttl) format on RDF triplestore. Next, Fig. 4.9 is where the result of the query is modified and displays the results of the input queries. Lastly, Fig. 4.10 presents the interface of information searching based on a specific keyword and link graph stored in the storage.

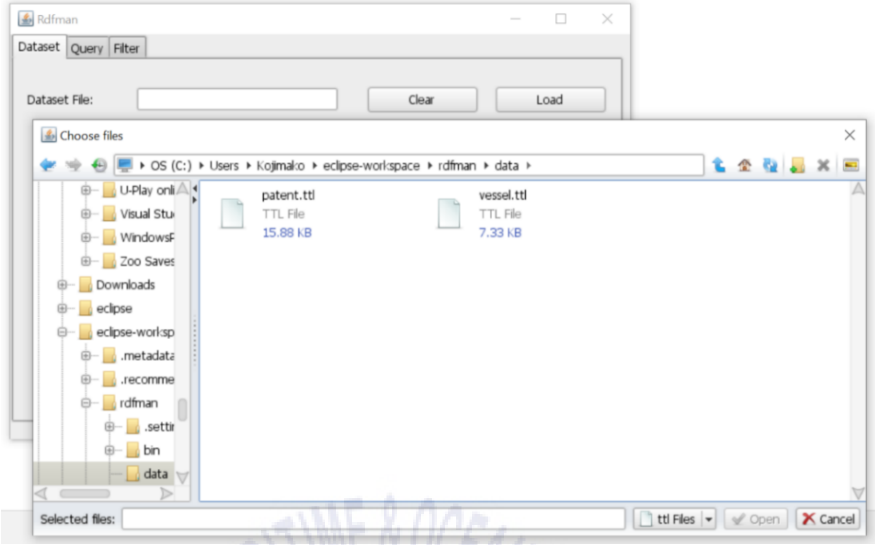


Fig. 4.8 Datasets Page

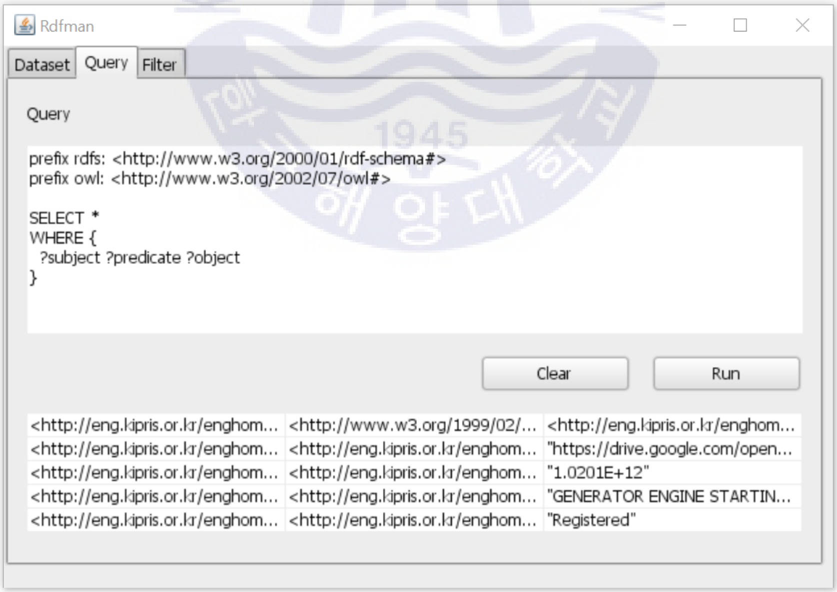


Fig. 4.9 Query Construct Page

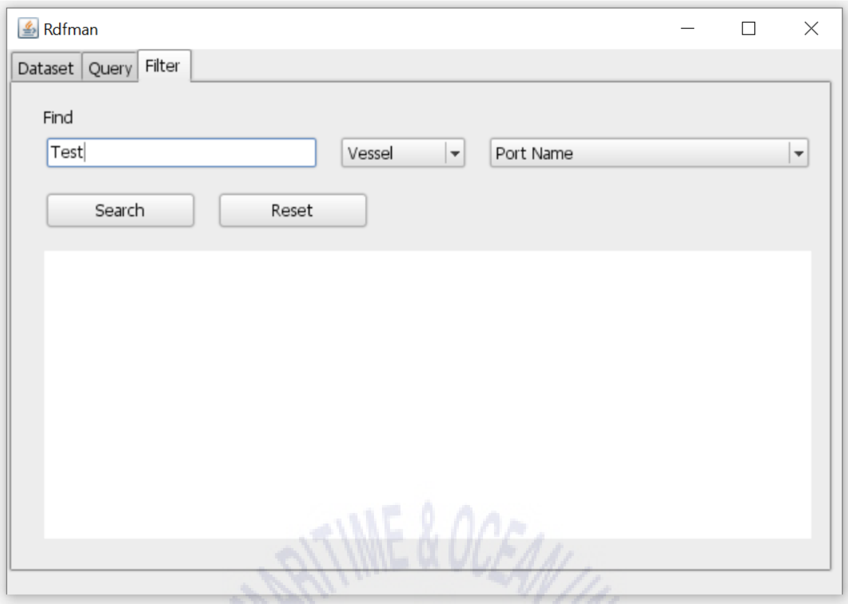


Fig. 4.10　Searching Page

Chapter 5: Conclusion and Further Work

In this paper, an approach was shown in a semantic search to retrieve a precise information related to vessel and patent data. The procedure consists of querying and analyzing the linked graph and the system returned lists of a matching query search. It is expected that this research presents a solution in searching through data to be more connected to the Linked Open Data and the Web.

The challenges arise in completing the proposed objective is that data retrieval does not appear accordingly to the input data high likely were because some query was not updated in the system structure. Then, the graphs were incorrectly designed making the graphs and tables not linked by the input searchable datasets. Therefore, as some of the aspects of future work, may this study bring us closer to the goal of Semantic Web evolution.

References

[1] Lapeyra, L. (2016, October 21). Introduction to the Semantic Web and Linked Data. Retrieved from https://dlis.hypotheses.org/788.

[2] RDF and SPARQL: Using Semantic Web Technology to Integrate the World's Data. (2007). URL https://www.w3.org/2007/03/VLDB/.

[3] Aluc, G., Ozsu, M. T., & Daudjee, K. (2014). Workload Matters: Why RDF Databases Need a New Design. Retrieved from http://www.vldb.org/pvldb/vol7/p837-aluc.pdf.

[4] OPEN DATA SUPPORT: Designing and developing RDF vocabularies . (2012). Retrieved from https://www.europeandataportal.eu/sites/default/files/d2.1.2_training_module_2.4_designing_and_developing_vocabularies_in_rdf_en_edp.pdf.

[5] What is Semantic Data. (2009). Retrieved from http://www.semagix.com/what-is-semantic-data.htm.

[6] Ontotext: What is RDF? Making Data Triple Their Power. (2017). URL https://www.ontotext.com/knowledgehub/fundamentals/what-is-rdf/

[7] Hogan, A. (2013). Linked Data & the Semantic Web Standards. URL http://aidanhogan.com/docs/ldmgmt_semantic_web_linked_data.pdf.

[8] Yin, F. (2013, March). A Parallel Application. Retrieved from https://people.eecs.berkeley.edu/~driscoll/cs267/hw0/html/XiaotingYin/.

[9] Frank, Matthias & Zander, Stefan. (2019). The Linked Data Wiki: Leveraging Organizational Knowledge Bases with Linked Open Data. 10.1007/978-3-030-15640-4_15.

[10] The Linked Open Data Cloud. (2007). Retrieved from https://www.lod-cloud.net/.

[11] Star Open Data. URL https://5stardata.info/en/

[12] Ontotext: What Are Linked Data and Linked Open Data?, (2016). URL https://www.ontotext.com/knowledgehub/fundamentals/linked-data-linked-open-data/

[13] Brickley, R. V. Guha, and A. Layman. Resource Description Rramework (RDF) Schemas. https://www.w3.org/TR/1998/WD-rdf-schema-19980409/, (1998). URL https://www.w3.org/TR/1998/WD-rdf-schema-19980409/.

[14] Dan Brickley and R. V. Guha. RDF Vocabulary Description Language 1.0: RDF Schema. W3C Recommendation, (2004). URL https://www.w3.org/TR/2004/REC-rdf-schema-20040210/

[15] Curé, O., & Blin, G. (Eds.).: RDF Database Systems Triples Storage and SPARQL Query Processing., In: RDF Data Management, pp. 25 (2015)

[16] Jose E. Labra Gayo, Eric Prud'hommeaux, Iovka Boneva, Dimitris Kontokostas (2018) Validating RDF Data, Synthesis Lectures on the Semantic Web: Theory and Technology, Vol. 7, No. 1, 1-328, Morgan & Claypool. URL http://book.validatingrdf.com/

[17] Brickley, D., & Guha, R. V. (2001). Resource Description Framework (RDF) Schema Specification 1.0. URL https://www.w3.org/2001/sw/RDFCore/Schema/20010618/.

[18] RDF and SPARQL: Using Semantic Web Technology to Integrate the World's Data. (2007). URL https://www.w3.org/2007/03/VLDB/.

[19] SPARQL vs SQL. (n.d.). URL https://www.cambridgesemantics.com/blog/semantic-university/learn-sparql/sparql-vs-sql/

[20] Jimenez, E., Goodman, E. L.: Triangle Finding: How Graph Theory Can Help the Semantic Web. In: Joint Workshop on Scalable and High-Performance Semantic Web Systems, pp. 45–58 (2012)

[21] Jose E. Labra Gayo, Eric Prud'hommeaux, Iovka Boneva, Dimitris Kontokostas (2018) Validating RDF Data, Synthesis Lectures on the Semantic Web: Theory and Technology, Vol. 7, No. 1, 1-328, Morgan & Claypool. URL http://book.validatingrdf.com/

[22] Morsey, M., Lehmann, J., Auer, S., Ngomo, A. N.: Usage-Centric Benchmarking of RDF Triple Stores. In: AAAI 2012, pp. 2134–2140 (2012)

[23] Apache Jena Fuseki : Main Server. (2019). URL https://jena.apache.org/documentation/fuseki2/fuseki-main.

[24] Data Access Control for Fuseki. (2019, May). URL https://jena.apache.org/documentation/fuseki2/data-access-control.

[25] Ducruet, C., Lee, S.-W., & Roussin, S. (2009). Local strength and global weakness: A maritime network ... Retrieved from https://www.researchgate.net/publication/41528614_Local_strength_and_global_weakness_A_maritime_network_perspective_on_South_Korea_as_Northeast_Asias_logistics_hub.

[26] Data Request. (2007). Retrieved from https://www.fleetmon.com/services/historical-ais-data/.

[27] Port-MIS: Yes! U-Port Your Future E-Business Safe Voyage (2018) URL https://www.klnet.co.kr/resources/download/02.pdf

[28] Guo, X. S., Chaudhary, M., Dozier, C., Arumainayagam, Y., & Subramanian, V. (2004). A web application using RDF/RDFS for metadata navigation. Proceedings of the Workshop on NLP and XML (NLPXML-2004): RDF/RDFS and OWL in Language Technology on - NLPXML 04.

[29] Patents Definition - What is Patents. (n.d.). URL https://www.shopify.com/encyclopedia/patents.

[30] PARR. (2019). Intellectual Property, Valuation, Exploitation, and Infringement Damages: 2019 cumulative .. supplement. JOHN WILEY & Sons.

[31] Navarrete, R., & Luján-Mora, S. (2015). Use of Linked Data to Enhance Open Educational Resources for 2015 International Conference on Information Technology Based Higher Education and Training (ITHET 2015), pp.1-6. Lisbon, Portugal, 11-13 June 2015. Piscataway, NJ: IEEE.

Acknowledgement

First and foremost, praises and thanks to the God, the Almighty, for His showers of blessings throughout my research work to complete the research successfully.

I would like to express my sincere gratitude to my advisor Prof. Hyu-Chan Park of the Computer Engineering Department at Korea Maritime & Ocean University (KMOU) for the continuous support of my Master's study and research, for his patience, motivation, enthusiasm, and immense knowledge. His guidance helped me in all the time of research and writing of this thesis. I could not have imagined having a better advisor and mentor for my Master's study.

Besides my advisor, I would like to thank the rest of my thesis committee: Prof. Ok-Keun Shin, and Prof. Jang-Se Lee, for their encouragement, insightful comments, and hard questions.

I thank my fellow lab mates in Database Lab: Mohammad Haziq, Jeong-Rae Kim, Nurul Azzahra Putri, and Fachrian Noor for the stimulating discussions, for the sleepless nights we were working together before deadlines, and for all the fun we have had in the last two years.

Last but not the least, I must express my very profound gratitude to my parents, Markhaini, Mozahker and to my partner, Khairul Nizam for providing me with unfailing support and continuous encouragement throughout my years of study and through the process of researching and writing this thesis. This accomplishment would not have been possible without them. Thank you.

January 2020, Zaslyana.