NETON: A NEW TOOL FOR DISCOVERING THE SEMANTIC POTENTIAL OF BIOMEDICAL DATA IN UMLS SEMANTIC NETWORK

A THESIS SUBMITTED TO

THE GRADUATE SCHOOL OF INFORMATICS

OF

THE MIDDLE EAST TECHNICAL UNIVERSITY

BY

BİRSEN GÜLDEN ÖZDEMİR

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

IN

THE DEPARTMENT OF INFORMATION SYSTEMS

MARCH 2010

Approval of the Graduate School of Informatics

____
Prof. Dr. Nazife Baykal
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Doctor of Philosophy.

____
Assist. Prof. Dr. Tuğba T. Temizel
Head of Department

This is to certify that we have read this thesis and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Doctor of Philosophy.

____
Prof. Dr. Nazife Baykal
Supervisor

Examining Committee Members:

Assist. Prof. Dr. Tuğba T. Temizel (METU, II) _________________________

Prof. Dr. Nazife Baykal (METU, II) _________________________

Assoc. Prof. Dr. İlyas Çiçekli (BİLKENT, CS) _________________________

Assist. Prof. Dr. Sevgi Özkan (METU, II) _________________________

Assist. Prof. Dr. Pınar Şenkul (METU, CENG) _________________________

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name : Birsen Gülden Özdemir

Signature :

ABSTRACT

NETON: A NEW TOOL FOR DISCOVERING THE SEMANTIC POTENTIAL OF
BIOMEDICAL DATA IN UMLS SEMANTIC NETWORK

Gülden Özdemir, Birsen

Ph.D., Department of Information Systems

Supervisor: Prof. Dr. Nazife Baykal

March 2010, 196 pages

The Unified Medical Language System Semantic Network (UMLS SN) being an upper-level abstraction of the biomedical domain has a complex structure due to many relationships, making it difficult for human orientation. Therefore, while the SN is a valuable source for modeling contents of the biomedical domain its usage is limited.

NetON was designed and built for the automatic transformation of UMLS SN to OWL sublanguages to support semantic operations between biomedical systems. NetON uses advances in the Semantic Web, a candidate technology for sustaining knowledge intensive tasks. Ontology Web Language (OWL) sublanguage rules are used to represent information in UMLS SN. The major contribution of NetON is the opportunity of automatic transformation of UMLS SN to OWL sublanguages named as OWL Basic Species. The aim of NetON is maximum possible information transformation from UMLS SN. The only information that is not able to be transformed to any OWL Basic Species due to the lack of appropriate constructors in OWL standard is inheritance blockings in UMLS SN.

In UMLS SN, there are unseen assertions that can be inferred by using inference rules on explicitly specified assertions which are not essentially valid for all the descendants. Deduction outcomes of any OWL reasoners on NetON OWL Basic Species will also include false positives due to the lack of inheritance blocking information. The algorithms of the second dimension consider the inheritance blocking information while executing inference rules. As this cannot be done by any OWL reasoner, the second dimension offers a solution for application developers.

Keywords: UMLS, Semantic Network, OWL, Ontology, Semantic Principles

ÖZ

NETON: BİRLEŞTİRİLMİŞ MEDİKAL DİLİ SİSTEMİ SEMANTİK AĞINDA BULUNAN BİYOMEDİKAL VERİNİN POTANSİYELİNİ KEŞFETMEK İÇİN YENİ BİR ARAÇ

Gülden Özdemir, Birsen

Doktora, Bilişim Sistemleri

Danışman: Prof. Dr. Nazife Baykal

Mart 2010, 196 sayfa

Biomedikal alanda üst seviyede model oluşturmak amacıyla tasarlanan Birleştirilmiş Medikal Dili Sistemi (BMDS)'nde bulunan Semantik Ağ (SA) çok sayıda barındırdığı ilişkiler nedeniyle insan yönetimi ve kullanımı açısından karmaşık bir yapıya sahiptir. Bu nedenle, biomedikal alandaki içeriği modellemede etkili ve değerli bir kaynak olmasına rağmen SA kullanımı sınırlı kalmaktadır.

NetON biyomedikal sistemler arasında anlamsal operasyonları desteklemek amacıyla BMDS SA'nın OWL alt dillerine otomatik dönüşümü için tasarlanmış ve oluşturulmuştur. NetON bilgi odaklı çalışmaları desteklemeye aday bir teknoloji olan Semantik Web'teki ilerlemeleri kullanır. SA'da bulunan bilgileri temsil etmek için OWL alt dillerinin kuralları uygulanır. NetON'un en önemli katkısı BMDS SA'yı 'OWL Temel Türleri' olarak adlandırılan OWL alt dillerine otomatik dönüşümü için imkan sağlamasıdır. NetON'un amacı SA'dan OWL formatına maksimum bilgi dönüşümünü sağlamaktır. BMDS SA içinde bulunup OWL standardında geçerli olan uygun bir yapı taşı bulunmaması nedeniyle, herhangi OWL Temel Türlerinden birine dönüştürülemeyen tek bilgi ilişkilerdeki miras aktarımını belirleyen notasyon olmuştur.

Çıkarım kurallarını kullanarak BMDS SA'da açıkça belirtilen semantik tipler arasındaki ilişkilerden açıkça görünmeyen diğer ilişkilere ulaşmak mümkündür. Ancak, bu ilişkilerin tüm torunlar için geçerli olma zorunluluğunun bulunmadığı unutulmamalıdır. OWL formatında çıkarım yapan herhangi bir yazılımın ya da aracın NetON OWL Temel Türlerini kullanarak yapacağı çıkarımlar miras aktarımını belirleyen notasyon eksikliği nedeniyle yanlış pozitif sonuçları da içinde barındıracaktır. NetON'un ikinci boyutu için geliştirilen algoritmalar çıkarımsal kuralları uygularken miras aktarımını belirleyen notasyon bilgisini de kullanırlar. Bu OWL formatında çıkarım yapan herhangi bir yazılım ya da araç tarafından yapılamayacağından, NetON'un ikinci boyutu ilgili uygulama geliştiriciler için bir çözüm sunar.

Anahtar Kelimeler: Birleştirilmiş Medikal Dili Sistemi, Semantik Ağ, OWL, Ontoloji, Semantik İlkeleri

To

my beloved husband

and

my family

ACKNOWLEDGMENTS

I would like to express my profound gratitude to my thesis advisor Prof. Dr. Nazife Baykal for his guidance throughout the study. I would also like to thank my thesis committee members for their suggestions and comments. Thanks to Sibel Gülnar for solving every administrative problem.

I appreciate Georgy Dimirovski and Mitat Konar for their valuable contributions. Thanks are extended to Deniz Şerifoğlu, Kıvanç Onan, and Zahid Gürbüz for their help on development process of NetON.

I would also like to thank my family for their understanding and support. I also want to thank Mehtap Özdemir for her special support.

I would especially thank my husband, Önder Özdemir for his encouragement, patience, and support throughout this doctorate study. Since he came into my life I felt his love and support any time which made me more powerful and energetic all the time.

TABLE OF CONTENTS

CHAPTER

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

AUI	: Atom Unique Identifiers
ASCII	: American Standard Code for Information Interchange
BioTop	: Abstract model of biomedical domain
CDAO	: Comparative Data Analysis Ontology
CHI	: Consolidated Health Informatics
CPT	: Current Procedural Terminology
CUI	: Concept Unique Identifier
DAML+OIL	: DARPA Markup Language
DARPA	: Defense Advanced Research Projects Agency
DL	: Description Logic
FMA	: Foundational Model of Anatomy
GO	: Gene Ontology
HIPAA	: Health Insurance Portability and Accountability Act
HTML	: Hyper Text Markup Language
ICD-10	: International Classification of Diseases and Related Health Problems
IS	: Information System
LOINC	: Logical Observation Identifier Names and Codes
LUI	: Lexical Unique Identifiers
MedDRA	: Medical Dictionary for Regulatory Activities
MEDLINE	: Medical Literature Analysis and Retrieval System
MeSH	: Medical Subject Headings
MLP	: Medical Language Processing

NCBO	: National Center for Biomedical Ontology
NCI	: National Cancer Institute
NLM	: National Library of Medicine
NLP	: Natural Language Processing
OBO	: Open Biomedical Ontologies
ORF	: Original Release Format
OWL	: Ontology Web Language
RDF	: Resource Description Framework
RDFS	: Resource Description Framework Schema
RL	: Relation between semantic types
RRF	: Rich Release Format
RSN	: Refined Semantic Network
UMLS	: Unified Medical Language System
UMLS SN	: Unified Medical Language System Semantic Network
SN	: Semantic Network
SNOMED	: Systematized Nomenclature of Medicine
SRDEF	: File consisting of basic information about the STYs and RLs in UMLS SN
SRFIL	: File containing descriptions of UMLS SN files
SRFLD	: File including fields descriptions inside UMLS SN files
SRSTR	: File involving structure of the network for UMLS SN
SRSTRE1	: File including fully inherited set of RLs in UMLS SN with unique identifiers
SRSTRE2	: File including fully inherited set of RLs in UMLS SN with names
STY	: Semantic Type
SUI	: String Unique Identifiers
TS	: Terminological System
WOD	: Without Descriptive Information of STYs or RLs
XML	: Extensible Markup Language

CHAPTER 1

INTRODUCTION

Biomedical vocabularies have always played a critical role in the context of biology and medicine information systems. Even though a variety of related knowledge sources containing vast amounts of valuable domain knowledge have been developed so far (e.g. Systematized Nomenclature of Medicine (SNOMED) and Unified Medical Language System), they cannot be directly integrated into real-world biomedical information systems. In addition to being too comprehensive, most of them are also not formalized in a suitable representation language to be reused and shared.

The Unified Medical Language System (UMLS) [58, 66], a comprehensive resource of biomedical terminology, consists of two knowledge sources: Metathesaurus and Semantic Network (SN). Metathesaurus [64] includes biomedical concepts from various source terminologies. The SN [59, 60, 61, 65] is an upper-level abstraction of the biomedical domain. The two knowledge sources are linked through assignments of one or more semantic types (STYs) to each concept of the Metathesaurus. These STY assignment(s) capture the semantics of a concept to identify its nature. The UMLS SN has a complex structure, due to many relationships, making it difficult for human orientation. Therefore, while the SN is a valuable source for modeling contents of the biomedical domain its usage is limited. The list of biomedical concepts can never be complete as new information, relationships, and details are always being discovered in the biomedical domain. This necessitates that UMLS knowledge sources continue to evolve to adapt to biomedical requirements.

Biomedical information is usually spread among several semantically or syntactically incompatible heterogeneous and independent systems. For a realistic contemporary approach, there is a need for a middle layer that uses common standards to exchange information between systems. We propose the use of a semantic layer to make the biomedical systems technology and data format independent. The main advantage of having a semantic middle layer is that the current integration technologies and applications will not be required to change; rather, a new layer making use of the already existing infrastructure will be added. (See section 4.1)

Taking into consideration the above situation, offering a representation of UMLS SN information appropriate for carrying out semantic operations is a significant problem. NetON (Unified Medical Language System Semantic **Net**work OWL **ON**tology Builder) was designed and built for the transformation of UMLS Semantic Network to OWL sublanguages (Lite, DL, Full) to support semantic operations between biomedical systems using such a middle semantic layer. NetON uses advances in the Semantic Web, a candidate technology for sustaining knowledge intensive tasks. In our tool, Ontology Web Language (OWL) sublanguage rules are used to represent information in UMLS SN.

1.1 Motivation of the Thesis

Biomedical environment gets help from IS (Information System) technologies to solve problems due to its complex nature. Biomedical ontologies and vocabularies play a vital role in the context of biomedical ISs. Being an abstract model, the UMLS SN is a valuable source for modeling contents of the biomedical domain. However, its usage is limited due to its complex structure and formalization that is not in a suitable representation language to be reused and shared. The need for knowledge-intensive activities in biomedical applications makes it necessary to transform UMLS SN current formalism to a proper format that uses a knowledge-oriented language.

The benefits of OWL (See section 3.1.2) indicates that Ontology Web Language is a suitable candidate standard for representation of UMLS SN. It is seen that major restrictions of UMLS SN content can be described by OWL constructors, with detailed inspection of OWL and UMLS SN.

The discovery of new information, relationships, and details necessitates the continuous evolvement of UMLS SN to adapt to biomedical requirements. Therefore, it is seen that any transformation effort on the formalism of UMLS SN should be done automatically to immediately reflect the change to the new formalism.

It is seen by the literature survey that there is no study on automatically transforming the maximum amount of UMLS SN information into OWL Sublanguages without changing the biomedical abstract model semantics. There was not also any study on transforming whole information included in UMLS SN to OWL without changing the owned semantics.

1.2 Contribution of the Thesis

NetON transforms UMLS SN to OWL in order to achieve a semantically convenient representation of upper-level abstraction of the biomedical domain. NetON was designed and built for the **automatic transformation** of UMLS Semantic Network to OWL sublanguages to support semantic operations between biomedical systems. The tool generates accessible OWL contents for both humans and software applications so that, respectively, developers can easily understand the generated OWL documents and biomedical applications can efficiently interoperate semantically.

The major contribution of NetON is the automatic transformation of UMLS SN to OWL sublanguages that use the first dimension algorithms in NetON. These algorithms were developed with the aim of maximum possible information transformation from UMLS SN to OWL sublanguages without changing the biomedical abstract model semantics. The only information that is not able to be transformed to any OWL Basic Species due to the lack of appropriate constructors in OWL standard is inheritance blockings in UMLS SN.

In UMLS SN, there are unseen assertions that can be inferred by using inference rules on explicitly specified assertions in SRSTR file that includes the structure of UMLS SN. However, explicitly declared relations between semantic types are not essentially valid for all the descendants. Deduction outcomes of any OWL reasoners on NetON OWL Basic Species will also include false positives due to the lack of inheritance blocking information. The algorithms of the second (Extended) dimension consider inheritance blocking information while executing inference rules. As this cannot be done by any OWL reasoner, the second dimension offers a solution for application developers. The whole information included in UMLS SN is transformed to OWL without changing the biomedical abstract model semantics by using developed second dimension algorithms.

OWL ontology developers must decide on the most appropriate OWL Sublanguage Species according to their requirements. Requirements of developers or applications in reasoning support and articulation from UMLS SN will determine the selection between OWL Lite, OWL DL and OWL Full Species.

The requirements of OWL ontology developers determine the most appropriate OWL Species from first and second dimension documents for them. OWL Sublanguage (First Dimension) documents are chosen by applications or developers when they only need explicitly defined information in UMLS SN. OWL Extended (Second Dimension) documents are chosen by applications or developers when they also need inherited information for descendants of STYs and RLs in addition to explicitly defined information in UMLS SN.

Proposed UMLS SN transformed by NetON promises reliable, correct, comprehensive, and significant representation of relations and classes. Consequently, dealings different biomedical activities can be carried out in a more efficient, generic and easier way over OWL content of UMLS SN.

Any generated OWL Species can also be altered by the customization application embedded in NetON. (See section 3.3.5) These altered species can use OWL relaxed primitives for constraint representation. Even such kind of alteration will not compute specific assertions concerning complexity and consistency; they might provide collective operability of databases, reasoning or non-reasoning applications using RDF Schema representation and OWL systems.

NetON is designed to be a bridge between the current presentation of UMLS Semantic Network and future technology where information will have exact meaning and can be understood and processed by computers. NetON enables the implementation of semantically manageable and interoperable biomedical systems by generating the OWL Species of UMLS Semantic Network as a component of the proposed semantic layer. OWL Species separates data from formatting and simplifies data sharing, data transport and platform changes. They provide machine-processable semantic models for UMLS SN and can be interpreted by biomedical applications. They are also shareable and reusable in different biomedical applications. Our tool targets accessible semantic content to allow efficient semantic interoperation and integration among these applications. (See section 3.3.8 for further information)

1.3 Structure of the Thesis

The following chapters describe the structure of this work. Chapter 2 contains the literature survey. Chapter 3 describes NetON approach, including the nature of OWL documents generated by our tool and an explanation for representation analysis of UMLS SN in OWL sublanguages. Chapter 4 includes discussion and conclusion.

CHAPTER 2

LITERATURE SURVEY

2.1 The Unified Medical Language System

The Unified Medical Language System (UMLS), a comprehensive resource of biomedical terminology, is started in 1986 by National Library of Medicine as a long-term research and development project. UMLS mainly provides biomedical and health data for system developers, informatics researchers, librarians, and other information professionals. The UMLS knowledge sources and related software tools are created and spread by National Library of Medicine (NLM). Systems to process, create, integrate and retrieve biomedical data and information can be built by developers by using the knowledge sources and tools. Those systems can perform varied functions involving information such as scientific literature, patient records, public health data, and guidelines. The three UMLS knowledge sources are Metathesaurus, Semantic Network and SPECIALIST Lexicon & Lexical Tools. Those knowledge sources can be distributed with the UMLS install and customization program (MetamorphoSys) or lexical tools. **Metathesaurus** includes more than 2.1 million concepts and 8 million unique concept names from over 140 source vocabularies (e.g. ICD10, MeSH, SNOMED …) in the current release (2009AA) of UMLS. [3] **Semantic Network** defines one-hundred thirty-five extensive categories and fifty-four relationships between categories for labeling the biomedical domain.

SPECIALIST Lexicon & Lexical Tools provide lexical information and programs for language processing. (Figure 1) In order to customize or use the UMLS knowledge sources for particular purposes, developers can also be aided by the related software tools. [1] Perl Y. and Geller J. questionnaire [67] distributed among researchers, programmers, physicians, professors and others showed that concepts and categories are most used subject areas of the UMLS.

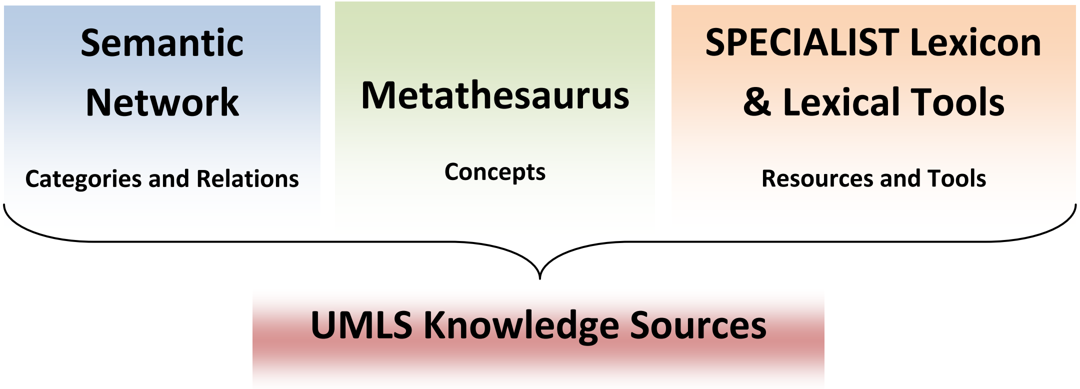


Figure 1 UMLS knowledge sources

2.1.1 Metathesaurus

The *Metathesaurus* being a multi-lingual and multi-purpose biomedical vocabulary knowledge source extensively progresses. The *Metathesaurus* includes the different names of concepts associated with health and biomedicine and relationships between them. Varied vocabularies, *code sets*, and *thesauri*, or "*source vocabularies*" are brought together to create the *Metathesaurus*. Sets of terms representing the same meaning are normally referred to as concepts. Terms from each source vocabulary are organized by meaning and assigned a concept unique identifier (CUI). [3]

The *Metathesaurus* contains over five million terms, or names, organized by meaning into concepts and assigned a unique identifier. Metathesaurus data is stored in a series of relational tables and files. Metathesaurus data can be installed locally using MetamorphoSys, a free software tool distributed with the UMLS. [2]

MetamorphoSys is used to install the UMLS Knowledge Sources, the UMLS resources, and customize the *Metathesaurus*. [1]

The *Metathesaurus* is not a vocabulary. It contains many vocabularies that are standards and helps to create mappings between these vocabularies but it was never intended to replace them. The *Metathesaurus* is linked to other UMLS knowledge sources. Generally, if a concept does not appear in any of the source vocabularies, it will also not appear in the Metathesaurus. [3] The UMLS Metathesaurus organizes all of the original data from the source vocabulary including unique identifiers, definitions, or term spelling variants into a common format.

Sixty-two percent of the Metathesaurus source vocabularies are in English. However, the Metathesaurus also contains terms from seventeen other languages such as Spanish, Italian, Dutch, French, Japanese, and Portuguese. [2]

The table below shows some of the *terms* which are part of the *concept* "Atrial Fibrillation." The *source vocabulary* that contributed each term is listed after it. Often a source vocabulary will contribute more than one term to a concept.

Table 1 Some terms that are part of the concept 'Atrial Fibrillation' [3]

Atrial fibrillation	ICD-9-CM
AF	NCI Thesaurus
Afib	MedDRA
Atrial fibrillation (disorder)	SNOMED Clinical Terms
atrium; fibrillation	ICPC2-ICD10 Thesaurus

In the current release (2009AA) of UMLS there are more than 2.1 million concepts and 8 million unique concept names from over 140 source vocabularies (e.g. SNOMED, MeSH, ICD10…) [71]. Each Metathesaurus concept is assigned to one or more semantic types in the UMLS SN by the categorization link established manually by the Metathesaurus editors independent of its hierarchical position in the original source vocabularies [70]. The categorization of Metathesaurus concepts allows users to subsetting and conceptualizing the domain [62]

2.1.1.1 Source Vocabularies

There is a concept in the Metathesaurus for each source vocabulary itself, which is assigned the Semantic Type "Intellectual Product". A special file (MRSAB.RRF and MRSAB in ORF) stores the version of each source vocabulary present in a particular edition of the Metathesaurus. All other Metathesaurus files that reference source vocabularies use "root" or versionless abbreviations, e.g., ICD9CM, not ICD9CM2003, thus avoiding routine wholesale updates to reflect the new versions. If one prefers versioned vocabulary source abbreviations in custom Metathesaurus subset files, MetamorphoSys offers this option. [3]

Although Metathesaurus preserves all the meanings and content in its source vocabularies, the Metathesaurus stores this information in a single common format. The native format of each vocabulary is carefully studied and then "inverted" into the common Metathesaurus format. For some vocabularies, this involves representing implied information in a more explicit format. For example, if a source vocabulary stores its preferred concept name as the first occurrence in a list of alternative concept names, that first name is explicitly tagged as the preferred name for that source in the Metathesaurus. [1]

Many different types of biomedical vocabularies are included in the Metathesaurus. There are many different ways to categorize them. Some vocabularies fall into more than one category. Major categories include [2]:

- Diagnosis
    - *LOINC*-Logical Observation Identifier Names and Codes
- Procedures & Supplies
    - *CPT*-Current Procedural Terminology

- Comprehensive Vocabularies/Thesauri
    - *SNOMED CT*-Systematized Nomenclature of Medicine-Clinical Terms
- Diseases
    - *ICD-10*-International Classification of Diseases and Related Health Problems

Other categories are anatomy, drugs, genetics, nursing and miscellaneous.

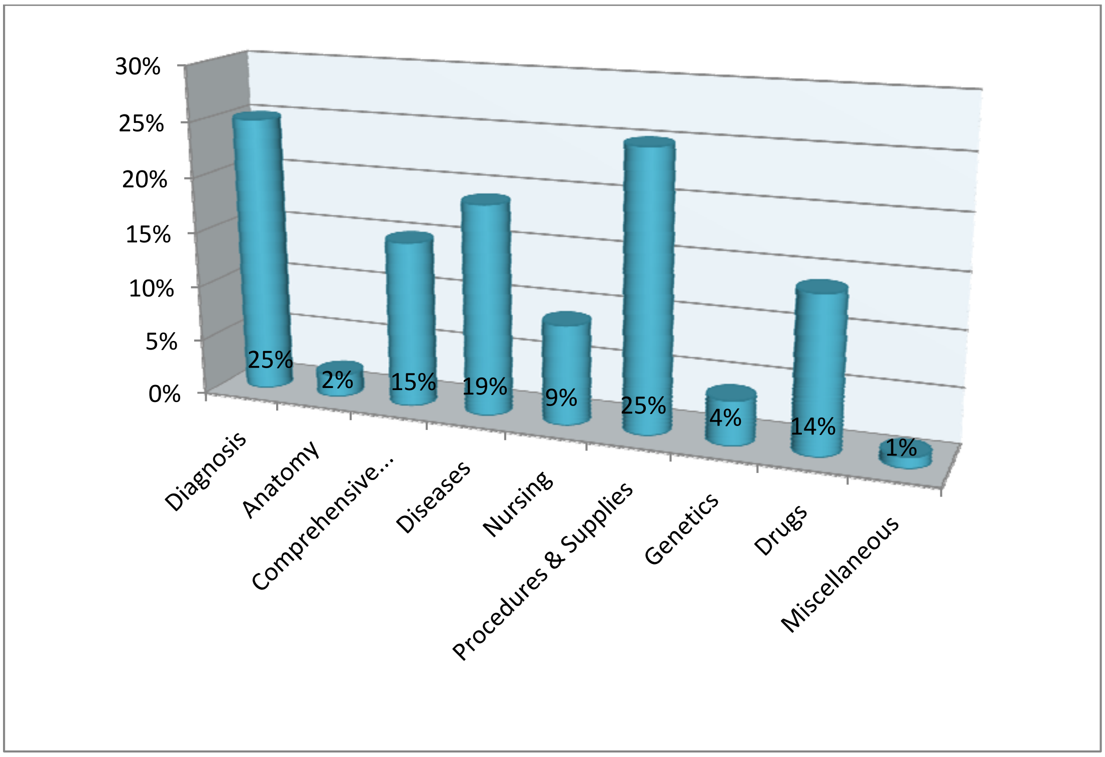


Figure 2 Different vocabulary categories represented in the Metathesaurus [2]

In Figure 2, the graph shows the percentage of different vocabulary categories represented in the Metathesaurus.

Inclusion of U.S. Standard Code Sets and Terminologies:

The Metathesaurus includes the code sets mandated for use in electronic administrative transactions in the U.S. under the influence of the Health Insurance Portability and Accountability Act (HIPAA)., the Metathesaurus includes all concepts and terms from these code sets, except the National Drug Codes (NDC). NDC codes available from the Food and Drug Administration are included as attributes of clinical drug concepts present in the FDA National Drug Code Directory (MTHFDA), which is a source vocabulary. [3]

NLM intends to incorporate all clinical terminologies designated as target U.S. standards by CHI (Consolidated Health Informatics) program and recommended as U.S. standards. Several of these (e.g., LOINC, SNOMED CT, RxNorm) are already present in the Metathesaurus. A vocabulary has been designated as a HIPAA or CHI standard. [2]

Inclusion of Languages Other Than English:

Many different languages are present in the Metathesaurus. The Metathesaurus includes many translations of some source vocabularies and in many cases, only the English version. As previously explained, MetamorphoSys makes it easy to create a subset of the Metathesaurus that excludes the languages that are not relevant in a particular application. [1]

2.1.1.2 Subsets

The Metathesaurus contains over 2.1 million concepts. Users create a useful subset, or smaller grouping of concepts, by choosing source vocabularies or applying a filter in MetamorphoSys, the free UMLS installation. Examples of subsets include [3]:

- Source vocabularies in a language (all Spanish vocabularies)
- All terms that are free for use within the United States
- CPT codes to be used for billing purposes
- Terms with the semantic type 'Clinical Drug'

There are two relational formats for Metathesaurus subsets [1]:

- Rich Release Format (RRF)
- Original Release Format (ORF)

Developers are encouraged to use the RRF which better represents the detailed semantics of each source vocabulary and provides better representations of concept name, source, and hierarchical information. [2]

2.1.1.3 Preferred Terms

A term from among the various names within a concept is identified as the *default* preferred term. Preferred terms are computed from a list of ranked source vocabularies. A default ranking of vocabularies is provided within the MetamorphoSys subset tool. Users can select the vocabulary for preferred terms when customizing their own subset. Below in the left column is a listing of the terms that could identify the concept Hodgkin's Disease. From this listing, one term, Hodgkin Disease is selected to be the preferred term and represent the concept [2]:

Collection of terms in the concept	**Preferred term**
Hodgkins Disease	Hodgkin Disease
Hodgkin disease	
Hodgkin's disease, unspecified	
Hodgkin's disease, unspecified type	
Hodgkin's disease (clinical)	
Hodgkin's disease NOS, unspecified site	
Hodgkin's disease NOS (disorder)	
Hodgkin's sarcoma (clinical)	
…	

2.1.1.4 Unique Identifiers in the Metathesaurus

In Metathesaurus, when a concept is added, it receives a unique identifier and is placed in the Metathesaurus structure. This structure has four levels of specification [2]:

- **Concept Unique Identifiers (CUI):** The diverse names in different source vocabularies can have the same meaning. In Metathesaurus, names having the same meaning (synonyms) are constructed as a concept having unique identifier. CUIs link all Metathesaurus information related to particular concepts. In other words, a CUI can be utilized to find all the concept attributes, relationships, and names for a specific concept in a Metathesaurus file. CUIs also serve as permanent, publicly available identifiers for biomedical concepts or meanings to which many individual source vocabularies are linked. CUI contain the letter C followed by seven numbers. In Figure 3 below the CUI is C0018681.
    - **Concept Names Added During Metathesaurus Construction:** Although the majority of concept names present in the Metathesaurus are from source vocabularies, some names of concepts are produced during construction of Metathesaurus. This occurs in the following circumstances:
        - ✓ A unique name is created for a string with multiple meanings
        - ✓ A more explicit name is created when none of the source vocabulary names for a concept conveys its meaning adequately
        - ✓ An American English variant is generated for a British spelling
        - ✓ An equivalent basic Latin ASCII character set string is generated for a string in an extended character set, such as Unicode

      Like all other concept names in the Metathesaurus, names created during Metathesaurus construction are labeled to indicate their source.

    - **Strings with Multiple Meaning:** Strings having multiple meaning can appear in more than one concept in the Metathesaurus.

- **Lexical (term) Unique Identifiers (LUI):** LUI links strings that are lexical variants [9]. LUI contain the letter L followed by seven numbers. In Figure 3 below there are three lexical variants, each given a separate LUI.

- **String Unique Identifiers (SUI):** A punctuation difference or lower upper case, i.e. any dissimilarity in character set, is stated as a split string having unique string identifier. SUI contain the letter S followed by numbers. In Figure 3 below there are four strings with four different SUI.

- **Atom Unique Identifiers (AUI):** Each incidence of a string from each of the source vocabulary is allocated with AUI in Metathesaurus. AUI contain the letter A followed by seven numbers. In Figure 3 below there are five strings from five sources with five different AUI. The abbreviation for the source that contributed each string is noted in parentheses after the string.

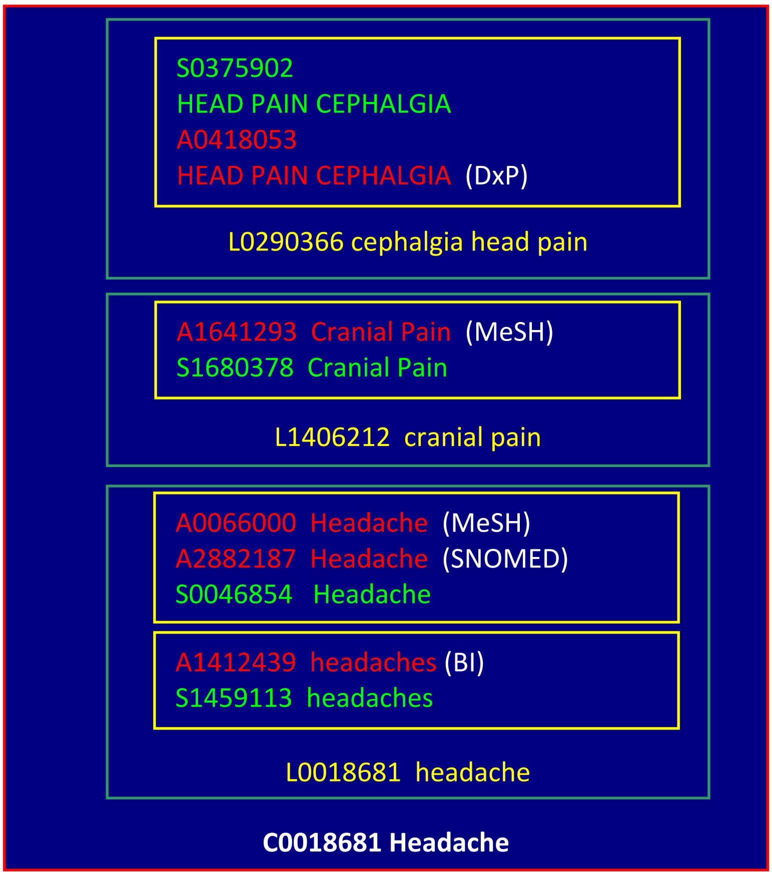


Figure 3 Unique identifiers in the Metathesaurus [2]

2.1.1.5 Representation of Ambiguity in the Metathesaurus

Separate Metathesaurus files (AMBIGLUI.RRF and AMBIGSUI.RRF (AMBIG.LUI and AMBIG.SUI in ORF)) contain the LUIs and SUIs of all ambiguous terms and strings known to the Metathesaurus. [1]

2.1.1.6 Metathesaurus Data Files

The Metathesaurus consists of *forty* data, metadata, and index *files*. The data files listed below table contain information obtained from the source vocabularies. *Concept Unique Identifiers (CUI)* links concept data across files. The table below illustrates what information populates each data file.

Table 2 Metathesaurus data files [2]

Data File Name	**Contents**
MRCONSO.RRF	Names, Synonyms, Terms, Term Types, Codes
MRREL.RRF	Relationships
MRHIER.RRF	Hierarchies
MRSAT.RRF	Attributes
MRDEF.RRF	Definitions
MRMAP.RRF	Mappings
MRSMAP.RRF	Simplified Mappings
MRSTY.RRF	Semantic Types

In addition to the data files two other types of files are produced with each Metathesaurus release. Index files are produced to help developers build applications that search for specific words or groups of words. For example, the index file MRXNW_ENG.RRF connects words to all related strings, terms, and concept identifiers. Metadata files contain information about each specific release of the Metathesaurus including its sources and files. For example, MRFILES.RRF contains a listing of all files in a Metathesaurus subset with a brief description and a listing of all rows and columns. Subsets may include files with a size of 0 bytes. Some files may not be included with a subset. For example, MRXW_DUT.RRF will only be included in a subset that contains Dutch language terms. [2]

2.1.1.7 Metathesaurus Release Formats

There are two relational formats that are available as output options of MetamorphoSys [1]:

- **The Rich Release Format (RRF):** All Rich Release Format file names have an extension (.RRF). The Rich Release Format has a number of advantages and is the preferred format for new users of the Metathesaurus and for most data creation applications.
- **The Original Release Format (ORF):** Original Release Format files have no extension.

2.1.1.8 Customization of the Metathesaurus

The Metathesaurus is designed to facilitate customization. The Metathesaurus information is labeled to its source(s), so it is possible to determine which concept names, attributes, and relationships come from which source vocabularies and which attributes and relationships were added during Metathesaurus construction. The labels allow to subset the Metathesaurus by excluding information from specific source vocabularies, including those for which one has necessary licenses or permissions. It is also easy to exclude all source vocabularies that have particular restriction levels or all information in particular languages. In addition to identifying the source(s), restriction levels, and language of the information it contains, the Metathesaurus includes various more specific concept name flags and relationship labels that can help to exclude content that is not relevant or helpful for particular applications. [2]

MetamorphoSys, the installation and customization program can be used to change the default preferred names of concepts; to alter the defaulting character set (from 7-bit ASCII to Unicode UTF8); and to include versioned vocabulary source abbreviations in every Metathesaurus file. MetamorphoSys also generates special subsets referred to as Content Views. A content view may specify any pre-defined subset of the Metathesaurus that is useful for some specific purpose. The actual definition of a content view can take a variety of different forms [1].

- A complex query that identifies particular sets of data;
- A list of sources that participate in the view;
- An actual list of Metathesaurus UIs maintained over time.

A Content View Flag (CVF) consists of an arbitrary bit field, with each bit representing membership in a particular Content View; each Content View is documented in MRDOC.RRF.

2.1.2 The Semantic Network

The **Semantic Network** was created in an attempt to offer a semantic framework for the UMLS vocabularies [62]. The biomedical knowledge in the SN identifies semantic relations between the semantic types. The SN consists of categories called semantic types (STYs), which are organized in two single-inheritance hierarchies: Event and Entity. In addition to is-a, different relationships are also defined in the UMLS SN which are used to represent hierarchical and associative relations among semantic types. Even UMLS SN is a valuable source for modeling contents of biomedical reality its use is limited due to its complexity (See section 3.1.1).

The Semantic Network provides a reliable classification of all concepts presented in the *Metathesaurus* and provides practical relationships among them. Categories are called as Semantic Types in SN. The STYs are the nodes in the UMLS SN, and the RLs between them are the links. The SN is an authority for the STYs which are allocated to Metathesaurus concepts. The STYs hierarchies and textual descriptions describe the SN that can be used to categorize any medical vocabulary too. The SN can assist biomedical applications to interpret meaning. The SN consists of [2]:

- ***High level categories*** **(STYs):** In SN, there are 135 STYs which are broad subject categories. Every concept in Metathesaurus is assigned at least one STY; very few terms are assigned as many as five semantic types. Semantic types are listed in the Metathesaurus file MRSTY.RRF. Some semantic types is shown below figure.

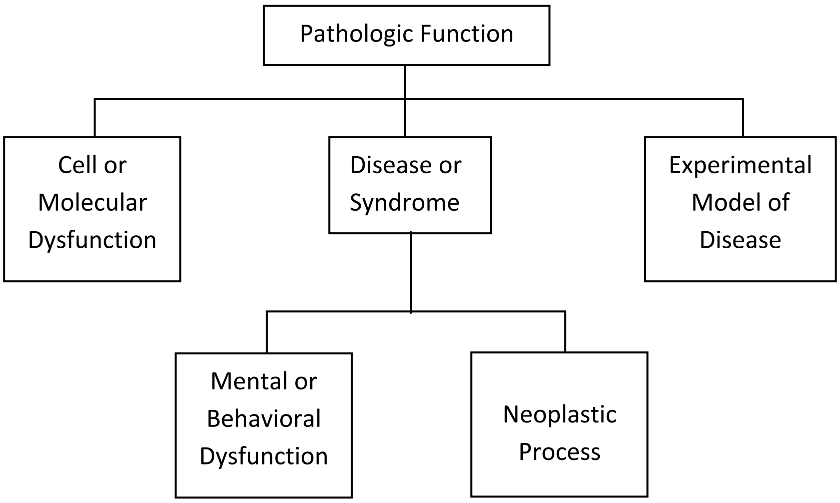

Pathologic Function
Cell or Molecular Dysfunction
Disease or Syndrome
Experimental Model of Disease
Mental or Behavioral Dysfunction
Neoplastic Process

Figure 4 Semantic types (high level categories)

- ***Relationships between semantic types*** **(RLs):** In SN, there are 54 relationships. Semantic relationships are useful relationships that exist between semantic types. For example: Clinical Drug treats Disease or Syndrome. The below figure and table illustrates two semantic relationships. (Figure 5) (Table 3)

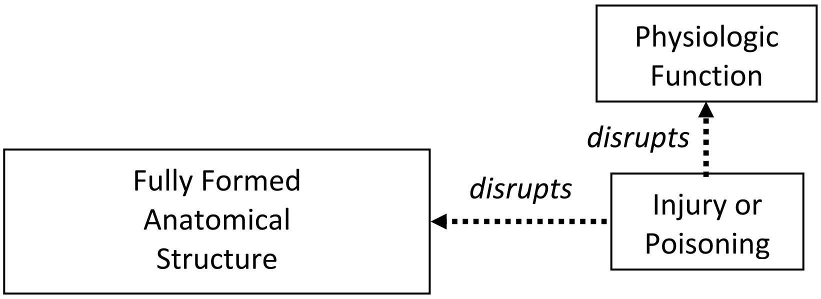


Figure 5 Semantic relationships (relationships between semantic types) [2]

Table 3 Semantic relationships (relationships between semantic types)

Semantic Type	**Semantic Relationship**	**Semantic Type**
Injury or Poisoning	disrupts	Fully Formed Anatomical Structure
Injury or Poisoning	disrupts	Physiologic Function

Semantic types and semantic relationships create a network that represents the *biomedical domain*. Semantic types and relationships help with interpreting the meaning of concepts in Metathesaurus.

2.1.2.1 Identifiers in the Semantic Network

The STYs are described with the following information [1]:

- Definition
- Hierarchy position - tree number
- Unique identifier
- Direct children and parent

The RLs are described with the following information [1]:

- Definition
- Examples
- Hierarchy position - tree number
- Unique identifier

2.1.2.2 Semantic Types

Each Metathesaurus concept is assigned at least one semantic type. The STYs exist in differing levels of granularity or specificity. Therefore, the most precise STY existing in the hierarchy is allocated to the concept. The concept “trout”, for instance, would receive the STY 'fish', not the semantic type 'animal' because fish is more specific. The hierarchy of STYs is organized into two main categories, **Entity** and **Event**. Current semantic types list can be seen from Appendices: Entity (Appendix B) & Event (Appendix C)

2.1.2.3 Semantic Relationships

The main link among most STYs is the hierarchical relationship. In addition, a set of non-hierarchical RLs between the STYs has been defined. Some examples of the 'isa' relationship [1]:

- Animal isa Entity
- Carbohydrate isa Chemical
- Human isa Mammal

There are five major, non-hierarchical relationships: Conceptually related to, functionally related to, temporally related to, spatially related to, and physically related to. There is also hierarchy among RLs in UMLS SN. (Figure 6)

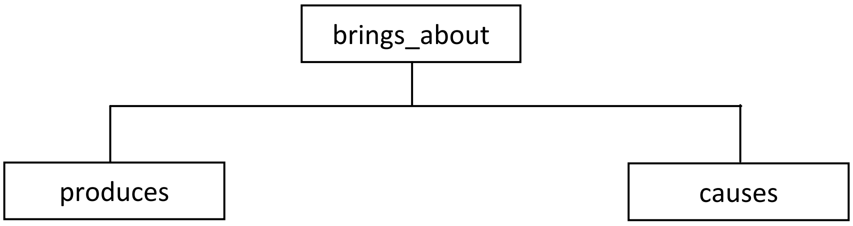


Figure 6 A part of the hierarchy for relationships in SN

Offered semantic relationships in UMLS SN can be seen in Appendix A. The RLs among high-level STYs are usually inherited by the children of these STYs by means of the "isa" link. When there is a disagreement among the STY assignment in UMLS SN and the RL to be inherited, however, the relation is clearly blocked. When the RL may not be inherited by the children of related STYs, this link is only blocked for every child of related STYs, as well. [2]

The RLs among STYs are also not essentially valid for all instances of Metathesaurus concepts assigned to these STYs. That is to say, the RL may not or may be applicable among any specific pair of concepts.

2.1.2.3.1 Parent - Child (Broader/Narrower) Relationships

One of the more important relationships within the Semantic Network is the **Parent-Child**, or Broader-Narrower, relationship. This relationship illustrates the hierarchies that exist between biomedical concepts. Child (narrower) relationships can be seen as a 'subtype'. For example, the STY 'Biologic Function' is the parent of, or broader than, the STY 'Physiologic Function'. Some examples of the Parent-Child relationship can be seen at Table 4 [1]:

Table 4 Some examples of the parent-child relationship [1]

Parent (broader) type	**Child (narrower) type**
Physiologic Function	Organism Function
Affects	Disrupts
Finding	Sign or Symptom

2.1.2.3.2 Semantic Relationships at the Concept Level

Semantic relationships cannot or can hold at the concept level. For example, the relationship "*Clinical Drug causes Disease or Syndrome*" does not hold at the concept level for Aspirin and Cancer. Aspirin does not cause cancer. [1] Not all relationships that apply at the concept level are indicated in the Semantic Network. The bellow relationship ‘evaluation_of’ does not hold between the concepts ‘fever’ and ‘body weight’, for instance. (Table 5)

Table 5 Semantic relationships at the concept level

Type	Relationship	Type
Sign or Symptom	evaluation_of	Organism Attribute

Figure 7 shows the graphical representation of a part of the UMLS SN [2]:

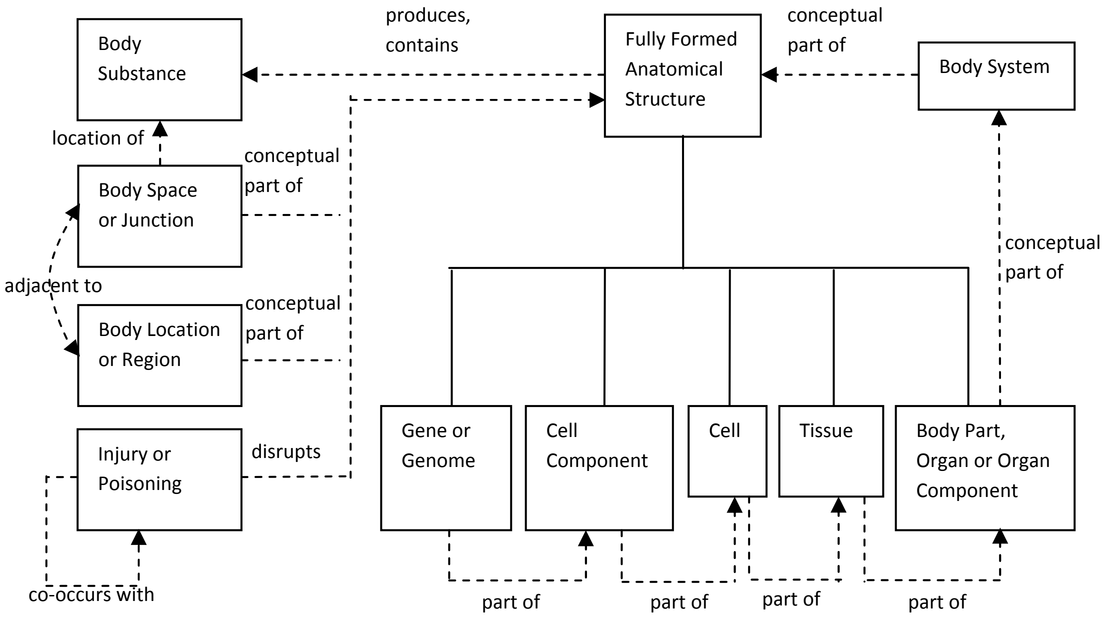


Figure 7 Graphical representation of a portion of the UMLS SN

2.1.2.4 Semantic Network Formats

There are two formats for the UMLS SN [2]: a unit record format and a relational table format.

- **SN ASCII Relational Format:** UMLS SN semantics is distributed to six different relational format files [44]. SRFIL file contains descriptions of those files. SRFLD file include fields descriptions inside those files. SRDEF file consist of basic information about the semantic types and relations. SRSTR file involves structure of the network. SRSTRE1 and SRSTRE2 files include fully inherited set of relations with unique identifiers and with names, respectively. Specific descriptions of ASCII relational format tables can be seen in Appendix D. Sample relational records can be seen in Appendix E. In NetON, ASCII Relational Format files of UMLS SN were used for the transformation into OWL.
- **SN ASCII Unit Record Format:** Individual records for both RLs and STYs are included in the 'SU' file which represents information different than files in ASCII Relational Format.

2.1.3 Lexical Tools and SPECIALIST Lexicon

The SPECIALIST Lexicon is an English dictionary containing many words from the biomedical domain. Words are chosen for lexical coding from diverse sources including MEDLINE abstracts, *Dorland's Illustrated Medical Dictionary* and the general English vocabulary. The majority of the words are nouns. The lexicon consists of a set of lexical entries. Each entry represents a word (lexical item). The entry covers one or more spellings in a particular part of speech and describes the morphologic, orthographic and syntactic properties of a word.

The lexical tools are a collection of java programs that process natural language words and terms. The lexical tools include a normalize (Norm), a word index generator (Wordind), and a lexical variant generator (LVG). The lexical tools are utilized to produce the normalized string indexes or words to the Metathesaurus. Together the SPECIALIST Lexicon and lexical tools allow users to develop Natural Language Processing programs. [1]

2.1.4 Recent Studies in Biomedical Domain Using UMLS

Denecke, K. [45] worked on a method using MLP (medical language processing) techniques. He mapped noun expressions in a medical text to terminological concepts. During the determination of the semantic relations, he used semantics offered by terminologies (e.g. UMLS, etc.) Denecke, K. indicated that UMLS was used by many MLP researches. He also examined to adaptability of his method to terminologies in medicine by utilizing the UMLS and related algorithms for mapping. His method mapped syntax into semantics by using semantic conversion regulations and language engineering techniques together to transform related part of medical text into semantic format. This semantic format of medical text (i.e. procedures or diagnoses documents) was investigated to find particular terms and their semantic characters. After finding 80% correctness in validation test on chest X-ray documents, Denecke, K. concluded that the extraction data from medical documents could be done with the proposed method. The applicability of the method to the UMLS was revealed. Denecke, K.used UMLS as domain knowledge to observe the suitability of the method to various medical subdomains. As a future work Denecke, K. recommended to recover the method regarding composite syntax structure processing. Denecke, K. stated that semantic format transformed by the method would be also utilized in medical images annotation and retrieval.

Abu-Hanna, A., et al. [101] stated that health care depended on procedures based on information. Treatments, diagnoses and predictions were documented by medical specialists and shared with researchers, managers and other specialists. They indicated that being basically an ontology, TS (Terminological Systems) in medicine contained terms, term properties and relations. TSs were commonly based on frames. Abu-Hanna, A., et al. indicated that TS was constructed with unplanned manner and because of unsystematic semantics of formalisms the related applications interoperability was reduced. Being very popular in latest time, Protégé [26] allowed accumulating knowledge and combining ontologies with applications using those ontologies. Capability to identify slots and meta classes in Protégé offered unambiguous separation of knowledge levels and therefore, permitted improved modeling flexibleness. Being powerful applicant for construction of TSs, Protégé was used to identify a terminological system in the field of intensive care. Abu-Hanna, A., et al. analyzed terminological system by the help of cognitive structure and mapped its elements to Protégé constructors. The outcome of this approach was the identification of knowledge elements for construction of TSs. Abu-Hanna, A., et al. indicated that this approach made possible the reuse them, guided for a systematic procedure for construction of TSs in medical domain. They concluded that supposing that Protégé could describe large variety of medical terminological systems (e.g. UMLS) and then verifying the correctness of them was reasonable.

The aim of the WRAPIN project [47] was to validate health information in Internet that could be used as precious knowledge in medical domain. This project also aimed the efficient information search in Internet. Gaudinata, A., et al. [46]that were included in the WRAPIN project stated that knowledge extraction from text documents in Internet was a challenging task. In order to improve the retrieval and management procedure of knowledge, they suggested using UMLS sources. As UMLS Semantic Network and MeSh terms had similar nature, Gaudinata, A., et al. especially used UMLS SN to find MeSH terms (UMLS SN categories) in examined files to detect and match the main conception of user queries. They used the original weight calculation among UMLS concepts to determine the significance of words and expressions in documents and to improve the MeSH terms (UMLS SN categories) extraction procedure. They indicated that the annotation of document based on manual work was the main limitation in preventing to extract terms with higher accuracy. They mainly tested their extraction method in querying the medical databases. Having optimistic result on evaluation their approach, Gaudinata, A., et al. decided to generalize the method for whole medical domain as a future work.

Chen, Y., et al. [48] stated that users of UMLS SN had difficulties in inclusive understanding of the SN having approximately 7000 RLs among semantic types. Therefore, there were studies on abstraction of UMLS SN to up hold the Semantic

Network understanding and visual representation [49, 76, 77]. McCray, A. T., et al. [76], for instance, grouped STYs in fifteen clusters according to usefulness, wholeness, uniqueness, and semantic strength. Kumar, A. [49] utilized the predicate logic representation to diminish the difficulty of UMLS Semantic Network.

A metaschema being an abstraction of UMLS Semantic Network could be defined by a partioning method that separated linked clusters of STYs which were symbolized with unique meta-semantic type. Zhang, L., et al. [50] engaged UMLS experts to develop personal metaschema and then combine all of them in agreed metaschema by examining the Semantic Network from lower STYs to upper ones which they called this study as top-down method. Chen, Y., et al. [48] employed UMLS experts to examine the Semantic Network trees from upper STYs to the lower ones which they called this study as bottom-up method. Chen, Y., et al. [48], in addition, introduced an evaluation and comparison method that applied to top-down and bottom-up metaschemas. They stated that the bottom-up metaschema was more reliable than top-down correspondent but both metaschema was actually had positive and negative parts. After evaluation of top-down and bottom-up metaschemas, they merged them by using the best and keeping away the worst parts of metaschemas. Chen, Y., et al. [48] concluded that merged metaschema had improved structural properties and offered finer assistance for visulazing and orienting the UMLS Semantic Network.

Chen, L., et al. [54] indicated that STYs about chemical categories in the UMLS SN had unclear explanation. The chemicals being a production of a reaction and plain mixture of chemicals were called as conjugates and complexes. They altered the RSN (Refined Semantic Network) [55], an improved abstraction of UMLS, to appropriately represent multiple STY assignments of chemicals. They also added new RLs. In this study [54], Chen, L., et al. called the combination of multi-typed chemicals as complex types or conjugate types. They stated that RSN was appropriate to the nature of chemicals as various chemicals could be combined to get new ones. Chen, L., et al. concluded that altered RSN simplifying STY assignment offered an improved modeling for chemical information with the new complex and conjugate types and added RLs. This kind of classification would allow a user directly to notice the nature of chemicals.

Perl, Y., et al. [63] signified that the UMLS Semantic Network was hard for understanding and visualizing due to having huge and complicated nature. In order to propose a solution, they divided the Semantic Network to consistent groups of STYs according to dispersion of RLs in UMLS Semantic Network. The outcome of this division was unified, individually rooted groups of STYs which were named according to the mutual character of sets and called meta-STYs. The combination of meta-STYs with taxonomic and associative RLs formed an abstract meta-schema of UMLS Semantic Network. Perl, Y., et al. used this meta-schema for auditing the classification of UMLS. They compared their meta-schema with meta-schemas constructed by UMLS experts. Perl, Y., et al. concluded that it was hard to understand UMLS SN due to its volume and complicated nature even it was the abstraction of UMLS. They stated that their meta-schema being semantically parallel to UMLS SN had the advantage of easier comprehension and visualization.

2.2 Semantics in Computer Science

Computer science intends to represent knowledge in a computationally amenable form. There is a need to capture and interpret knowledge unambiguously for both human and the computer. This is made possible by using a representation language with well defined semantics. Semantics generally stands for explicit meaning in computer science. A computer must understand what each of the statements in the depiction of some knowledge stands for [86].

The foundation technology for the achievement of semantics in computer science is the ontology. There are multiple definitions of ontology in the literature [92, 93]. A semantic model including concepts, inter-conceptual relations, their axioms and properties can be viewed as an ontology that corresponds to a reusable, shareable and common observation of a particular application domain. Ontologies provide meaning with the aid of knowledge representation languages [94]. The precise semantics of those languages tell how a human or a computer application ought to understand the statements in an unambiguous way [87].

2.3 The Web Vision

A future Web vision is the Semantic Web [83] in which information is specified with precise meaning in order to make machines automatically integrate and process it. Semantic interoperability between systems and management of knowledge is made possible by Semantic Web technologies, within which the keystone technology is the ontology [82]. An ontology capturing domain knowledge is a file or document that includes inference rules and taxonomy to formally describe the relations among concepts [85].

In the biomedical domain, ontologies have been extensively used with different purposes over the last several years. Ontologies are used for the formalization of medical concepts [88, 89, 90]. They have also been used in the biomedical domain for interoperability and integration [95, 97, 98, 99]. In addition, the number of bio-ontologies and related projects is rising [91]. Semantic Web technologies have recently been used in the Semantic Health European project [96] for achieving interoperability and representation of clinical knowledge. The National Center for Biomedical Ontology (NCBO) uses biomedical ontologies to support the analysis and management of biomedical data [4]. The significance of ontologies, as exhibited in the applications above, motivated us to work on ontology representation related to UMLS Semantic Network.

2.3.1 Semantic Web

Upward partial understanding and downward compatibility are followed by Semantic Web layers that are firstly proposed by Tim Berners-Lee [6]. Berners-Lee, et al. [7, 8] and Christopher, D.W. [5] states that the construction of the Semantic Web will happen in a piecemeal manner, and will be driven by the applications. Following figure shows Semantic Web layers (Figure 8) [6]:

XML allows syntactic interoperability between computer systems by utilizing user-defined vocabularies. Being a basic model for data, **RDF** is the basic Semantic Web language. having a syntax based on XML. Statements are triples composed of a binary predicate linking as logical facts *P(x, y)* [5].

RDF Schema provides a simple language for writing ontologies. Objects sharing similar characteristics are put together to form classes. Binary properties (such as works for) are used to establish connections between classes. The application of predicates is restricted by range and domain restrictions. For example, the property (i.e. predicate) works for can be restricted to apply only to employees (domain restriction), and to have as value only companies (range restriction). Classes can be put together in hierarchies through the subclass relationship. For example, the class of island destinations is a subclass of all destinations. The hierarchical organization of classes is important due to the notion of inheritance: once a class C has been declared a subclass of D, every known individuals of C are also automatically classified as individuals of D. This has far-reaching implications for matching customer preferences to service offerings. For example, a customer may wish to make holidays on an Indonesian island. On the other hand, the hotel Beach advertises its location to be Bali. It is not necessary (nor is it realistic) for the hotel to add information that it is located in Indonesia and on an island; instead, this information is inferred by the ontology automatically. However, RDF and RDFS are not adequate for describing more complex information. Therefore, **OWL**, an ontology language, is constructed on the top of RDFS and RDF. The next generation layers of Semantic Web are named as Logic, Proof and Trust. The Logical Layer states logical principles to additionally improve the ontology language and to permit the declaration of application. The Proof Layer declares the real procedure and validation of proof. The Trust Layer generally uses digital signatures. [6]

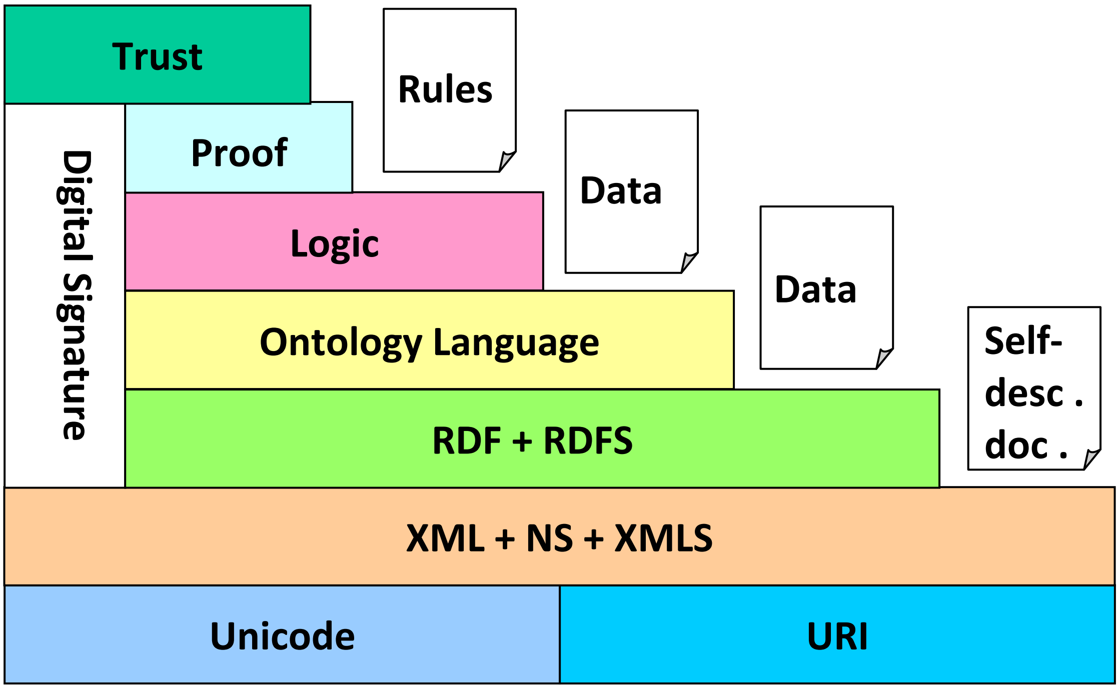


Figure 8 Semantic Web layered approach [6]

2.3.1.1 Extensible Markup Language

Extensible Markup Language (XML) development was driven by short comings of HTML. An XML document contains XML elements (root element and its children elements) [9]. The start tag of elements uses attributes while describing information that is not part of data. XML Namespaces are used to keep away from name conflicts. XML Schema includes the structure of an XML document and is used to validate this document [11].

In summary, XML separates data from formatting. It simplifies data sharing, data transport and platform changes. XML Schema contains only the structure of XML documents. However, XML and XML Schema do not provide semantics (meanings) of data.

2.3.1.2 Resource Description Framework

Being a W3C recommendation, Resource Description Framework (RDF) provides a model for data and is metadata (data about data) and is written with XML syntax. It describes subjects/classes with predicates/attributes and objects [12]. This basic building blog is called statement. RDF identifies things with Uniform Resource Identifier (URI) that allows a global, worldwide, unique naming schema [13]. There are two statements in the below example:

```
<RDF>
  <Description about=" Doğuş University ">
    <domain> education </domain>
    <telephone> 0(216)5445544 </telephone>
  </Description>
</RDF>
```

Statement 1: "The domain of Doğuş University is education".

Statement 2: "The telephone of Doğuş University is 0(216)5445544".

A triple, a graph, and XML code are three views of an RDF statement [39]. For disambiguation, a namespace is utilized. The graph and triples of data model of below RDF/XML code can be seen in Figure 9 and Table 6, correspondingly:

```
<?xml version="1.0"?>
<rdf:RDF
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:a="www.dogus.edu.tr/">
 <rdf:Description rdf:about="Doğuş University">
    <a:domain>education</a:domain>
    <a:telephone>0(216)5445544</a: telephone >
 </rdf:Description>
</rdf:RDF>
```

Table 6 Triples of the data model

Subject	Predicate	Object
Doğuş University	www.dogus.edu.tr/domain	Education
Doğuş University	www.dogus.edu.tr/telephone	0(216)5445544

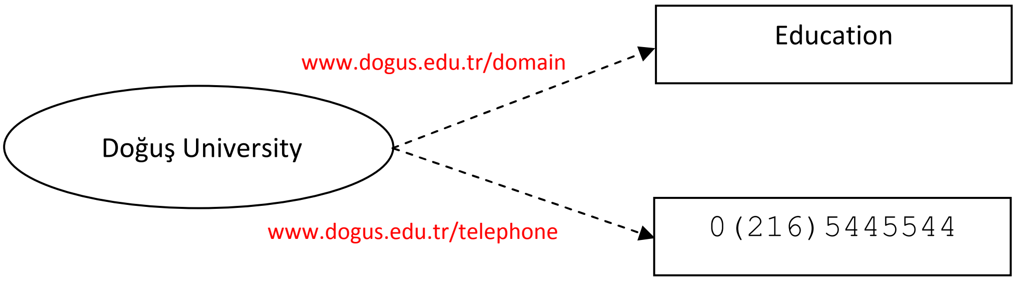


Figure 9 Data model graph

In RDF, resources can be described without a name to create anonymous resource, with an about attribute to reference a resource, or with an ID attribute [39]. Additionally, using the rdf:resource attribute for two entities is indicated as they are the same resources[13].

In RDF, a container may be described by using rdf:parseType = "Collection" (the “Collection” value for rdf:parseType attribute) and cannot be blocked to forbid including any other elements than described ones [12].

2.3.1.3 Resource Description Framework Schema

RDF describes resources with subjects, properties, and values. The semantics of a specific application domain cannot be defined by RDF. However, RDF Schema (RDFS) that is an extension of RDF can do this. RDFS offers the structure for description of application-based properties and classes, but not offers individuals of those application-based properties and classes. Like classes, hierarchical relationships between properties can also be defined. Classes can be organized in hierarchies by subclass relationships. Any subclass inherits the abilities from its super class [14]:

```
<rdf:RDF
 xmlns:rdf= "http://www.w3.org/1999/02/22-rdf-syntax-ns#"
 xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
 xml:base=  "http://www.hospital.tr/hospitals#">
 <rdfs:Class rdf:ID="person" />
 <rdfs:Class rdf:ID="doctor">
   <rdfs:subClassOf rdf:resource="#person"/>
 </rdfs:Class>
</rdf:RDF>
```

2.3.1.4 RDF versus RDF Schema

RDF Schema (RDFS), in brief, provides semantic model for particular domain by indicating the relationships between classes, and describing which properties apply to which kinds of classes. RDF and RDFS primitives have standardized meanings.

In RDFS, the main classes are rdf:Statement, rdf:Property, rdfs:Literal, rdfs:Class, and rdfs:Resource. The main properties are rdfs:subPropertyOf, rdfs:subClassOf, and rdf:type. The properties rdfs:range and rdfs:domain restrict the range(object class) and domain (subject class) of a property. The utility properties rdfs:label, rfds:comment, rdfs:isDefinedBy, and rdfs:seeAlso describe resources [14].

There are different layers involved in RDF and RDFS. Considering the RDF statement (*Birsen Gülden* is treated by *Mehtap Özdemir*), the RDFS and RDF layers can be seen in the following figure [13]:

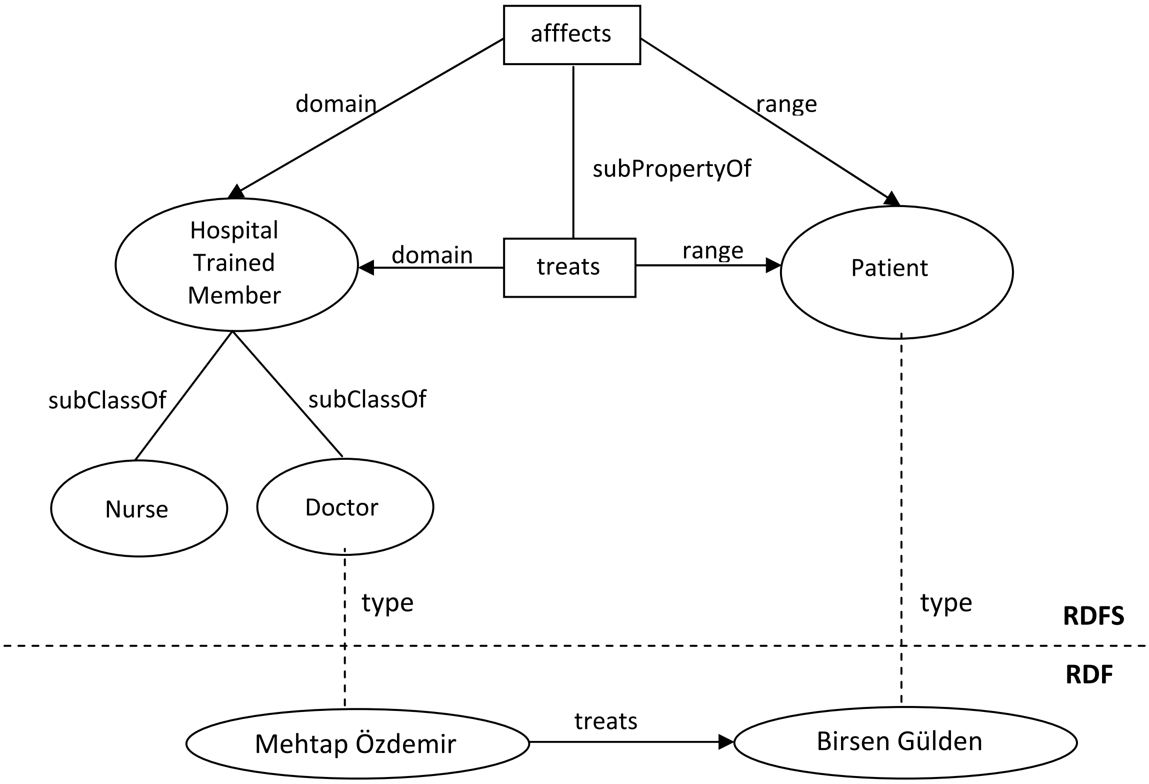


Figure 10 RDF and RDFS layers

2.3.1.5 Web Ontology Language

The precise description of things and their relationships are called as ontology. The major requirements of ontology are a well-defined syntax, a formal semantics, and efficient reasoning support. Semantics is prerequisite for reasoning support [40]. Being a W3C standard and being written in XML, Web Ontology Language (OWL) is an ontology language. OWL is for information processing and its primitives have standardized meanings [17].

2.3.1.5.1 OWL Sublanguages

The complete set of ontology language necessities appears underivable by joining of RDFS with entire logic. Therefore, three diverse sublanguages were developed by W3C'sWeb Ontology Working Group. The sublanguages leaned in satisfying varied characteristics of requirements [6]:

- **OWL Lite**: Excluding statements like class enumeration, casual cardinality and disjointedness, OWL Lite restricts a few OWL DL constructs. Facility on comprehension and implementation by application developers is the benefit of this sublanguage. Its drawback is limited interpretation.
- **OWL DL** (includes OWL Lite): Being based on description logic, this sublanguage prohibits the usage of OWL constructs to one another. Well-organized support on reasoning is the benefit of OWL DL. RDF restriction in one respect and extension in another, however, permits the lost of entire compatibility of legal OWL DL with RDF.
- **OWL Full** (includes OWL DL)**:** This sublanguage permits joining of entire language primitives of OWL in arbitrary manner with RDF or RDFS. Performing the language constructs to one another brings also about the probability of changing the inference of pre-specified OWL or RDF constructs. Semantic and syntactic upward compatibility with RDF exhibits the advantage of OWL. There is no resourceful support on reasoning due to the drawback of this powerful undecidable sublanguage.

2.3.1.5.2 OWL Syntax

OWL can be written with an RDF/XML-based syntax, an alternative XML-based syntax, an abstract syntax, and a graphic syntax. While the first one is the primarily used syntax, the last one is more understandable than others [16].

The owl:Class is used to describe classes. The owl:ObjectProperty and owl:DatatypeProperty constructors are used to describe object and data type properties [41]. The owl:allValuesFrom, owl:someValuesFrom, owl:hasValue, owl:maximumCardinality, owl:minimumCardinality, and owl:cardinality constructors are used to restrict the properties for a specific class decleration (local restriction). The range(object class) and domain (subject class) of an object property can be restricted by owl:range and owl:domain, respectively. This kind of restriction of an object property should be followed by all classes in an OWL document (global restriction). The owl:complementOf, owl:unionOf, owl:intersectionOf constructors are used for boolean combinations of class descriptions. In addition, the owl:InverseFunctionalProperty, owl:FunctionalProperty, owl:SymmetricProperty, and owl:TransitiveProperty are some properties of property elements [15].

2.3.1.5.3 Recent Studies in Biomedical Domain Using OWL

Goodwin J. [87] stated that capability of electronic information trading generated chances of using topographic data in different manners. The novel view of commerce could be shaped by Semantic Web for information providers such as Ordnance Survey. Mobile users, for example, could contact with Ordnance Survey data by using suitable software. Semantic exchange of information among sources could decrease service prices and time and improve the interaction between industry and government. With the mentioned prediction of benefits in mind, Ordnance Survey studied on a topographic ontology development in OWL DL to strengthen their data and to interoperate with different services and sources based on information. This ontology could also be used in biomedical domain. They concluded that even having many benefits of information described in OWL format, modeling knowledge in OWL was difficult task.

Sattanathan, S., et al. [37] indicated that synthesis of Web services created by varied suppliers had a problem of content representation diversity that complicates the combination of them, which commonly needed to be combined to solve users' requests. They proposed an approach based on ontology in OWL for mediation and representation of Web Services to emphasize the formal representation significance. They concluded that Web services described with formal semantics with OWL would permit improved service finding, smoother combination and interoperability of services. Formally described biomedical Web services in OWL could also allow superior service finding, smoother combination and interoperability among them.

Sánchez, D. & Moreno A. [51] indicated that hierarchical structure was the main focus of ontology learning procedures; even the non-hierarchical structure had the basic for constructing domain knowledge. Sánchez, D. & Moreno A. focused on non-hierarchical learning procedures in construction of domain ontology. Their method was capable of determining domain specific verb (to build domain patterns), finding non-hierarchically linked terms (by utilizing the built domain patterns) and tagging these links, utilizing the source of Web. After identifying and removing redundancy and determining implied links, the constructed domain ontology is represented with OWL to allow further processing by computers. They considered domain specific verb expressions as non-hierarchical tags. Then, these tags were used to find associated terms utilizing the entire Web as data source. There was no supervision throughout Web learning and analysis procedure. When a novel information was obtained, evaluation of the corpus was done dynamically. They combined statistical evaluation and linguistic investigation based on patterns to adapt these techniques to the environment of Web to introduce a new method for finding tagged non-hierarchical links. They validated the links found by their method with WordNet [52], an English lexical database. Sánchez, D. & Moreno A. indicated that the method is not dependant on a domain. Their method could be tried for a challenging and complex domain (e.g. biomedical domain) too.

Rajapakse, M., et al. [53] stated that unconstrained entrées to the information in literature databases and the Web reached to the huge amount being hard to manage. They concentrated to this problem and proposed ontology-based structure including a domain-based OWL DL ontology, an ontology sampling distributer for text miningly derived sentences, and an information accumulation engine. Rajapakse, M., et al. introduced the way of practical and effective questioning of the domain ontology by health practitioners or investigators that could not be obtained by existing search engines. They also used visual query to combine information and to exercise reasoning over ontology. Their basic structure was organized to assist integration and navigation of information in the field of dengue, a common contagious disease. Rajapakse, M., et al. used their structure to search and discover for dengue disease knowledge taken from different sources and combined with respect to dengue ontology. They also performed data mining to the ontology to clarify tendency of referencing the dengue serotypes in paper abstracts from 1974 to 2008. Rajapakse, M., et al. concluded that their structure for combining formless information could also be used by different domains too by providing domain-based ontology and with several customizations on text mining channel.

Noy, N. F., & Rubin, D. L. [100] stated that FMA (Foundational Model of Anatomy) representing a model for human body was a remarkable resource to assist distribution of knowledge between applications utilizing anatomical data. FMA was represented with frame language in Protégé. Noy, N. F., & Rubin, D. L. tried to translate it to OWL by focusing to represent unambiguously stated and complete information in FMA. Their translation represented with OWL DL and OWL Full. OWL Full notation imported the OWL DL elements and included FMS terms that could not be stated in OWL DL.

2.3.1.6 A Discussion of Semantic Web Standards

The Semantic Web development advances in layers. A domain knowledge needs to be expressed by enriching data with meaning [83]. In this structure, Ontology Web Language (OWL) [15] and Resource Description Framework (RDF) [12] are used as meta-data for achievement of modeling and enrichment of data represented by HMTL/XML [84]. Semantic Web languages use XML syntax and have varying degrees of expressivity. The underlying semantics in these languages is provided by graph theory and description logics.

Extensible Markup Language (XML) [10] separates data from Hyper Text Markup Language (HTML) formatting and simplifies data sharing, and data transport, platform changes. It is a universally accepted way of structuring data (syntax). The structure of XML documents is contained by XML Schema [11]. However, XML and XML Schema do not provide semantics of the data.

RDF provides syntax to model data. A subject, a predicate, and an object are encoded by triple based language RDF which is approximately restricted to binary ground predicates. RDF Schema (RDFS) [14] takes the basic RDF model and extends it to provide a minimal ontology representation language by creating application specific vocabularies in order to mainly specify relationships between RDF items [84]. Since subclass and property hierarchies, with range and domain specifications of properties, broadly limits RDFS [73].

OWL, which is based on DAML+OIL [72], comes with stronger syntax, greater machine interpretability and a larger vocabulary than RDF and is being used to encode knowledge processing and enable interoperability in distributed computer systems. In this work, ontology in the Web Ontology Language [15] is used to represent the UMLS Semantic Network.

2.4 Related Work

2.4.1 Biomedical Information Representation Formats and OWL

Biomedical ontologies had been presenting crucial domain information mainly offering annotation, consistency and decision support actions. Their commitment and basics in OWL format had been focused by different studies.

BioPortal [27]funded by National Institutes of Health (NIH) , for example, offered right of entry to ontologies repository built in Protégé frames, Open Biomedical Ontologies (OBO) format, RDF, and OWL by Web browsers and Web services in order to view or search ontologies. BioPortal assisted users on ontology development by attaching notes to terms or by mapping among terms.

Harris, M. A., et al. [38] stated that formal terminologies about cellular and molecular biology was offered with GO (Gene Ontology) project that was used by numerous genome and organism team and databases that also used in enlargement and revision of GO. Aranguren ME, et al. [86] stated the approach of representations of Gene Ontology's Directed Acyclic Graph into OWL together with OBO format in order to make scientific viewpoint clearly understood by users in biomedical domain. After indicating the advantages of semantic language representation, they inspected the semantics of Gene Ontology's Directed Acyclic Graph with the intention of comparison with OWL DL.

Basic ideas of a finely interpreted ontology, Comparative Data Analysis Ontology (CDAO), were identified Prosdocimi F, et al. [28] by utilizing OWL. CDAO was interpreted in order to indicate the capacity of data representation and reasoning support. Evolutionary changes, character-state data attribute representation and phylogenetic trees of compared things were presented in comparative analysis. The authors selected OWL representation for their aim which was to develop a reasoning framework for deduction on varied data types.

Moreira DA, and Musen MA [29] worked on a Protégé tab plug-in converting OBO files into OWL format (or vice versa) with the aim of providing computational representation to OBO format to simplify ontology improvement and reasoning. They utilized OBO to OWL conversion rules offered by OWL and OBO specialists in NCBO consortium funded by the NIH [31].

Aitken S, et al. [32] offered COBrA-CT ontology version manager tools containing a version administrator requester tool and an ontology server database utilizing OGSA-DAI Grid technology so as to represent OBO in OWL. COBrA-CT reserved ontologies to share with users by means of connection to the server for arranging ontology improvement between numerous developers.

Aitken S., et al. [30] presented a tool using the Protégé [26] and OBO to OWL Converter [29] for translation of OBO into OWL format. They planned the tool as utilizing an ontology administration server [32] for maintaining version administration.

Golbreich C., et al. [56] transformed the Foundational Model of Anatomy (FMA) from Protégé frame representation to OWL DL by using several enrichment and translation rules. They also used Racer [43] for reasoning with some smaller parts of the transformed OWL DL document. They concluded that using OWL DL is advantageous than using frames for representation FMA for the future of life sciences.

Van Assem, M., et al. [33] worked on conversion process of thesauri format of WordNet and MeSH to RDFS and OWL. They stated that transforming of sources to RDF model was difficult due to existence of many alternatives. They concluded that RDFS and OWL semantics could pick up the quality of thesaurus, which generally use imprecise semantics for relations and other terms.

Supekar, K., et al. [34] stated the representation of existing biomedical terminology in a suitable formal format was a challenging task. They worked on translation and representation of structured and multilingual lexical components of terminology in OWL. They advised to utilize a logical approach using Object Annotation among other recommendations. They concluded that their solution would aid researchers desiring to construct consistent and formal biomedical terminologies.

Soualmia, L., et al. [35] indicated that the resources in Catalogue and Index of French Medical Sites (CISMeF) were defined with French version of MeSH thesaurus. They worked on transforming CISMeF terminology to OWL DL founded on syntactic representation of MeSH hierarchy. They are concluded that the transformation should also consider the semantic principles of the resources besides syntactical principles. They also denoted that use of the UMLS Semantic Network relations could improve the transformation process. After improving the representation in OWL they recommended to practice it in information retrieval as a future work. They stated that the formal representation of MeSH would be helpful for especially bibliographic databases (e.g. Medline).

Sidhu, A.S., et al. [36] denoted that existing organization of proteins are differing and inadequate formal representation of protein information prevented the biological databases interoperability. Therefore, they suggested a Protein Ontology in OWL format to deal with the mentioned difficulty. They concluded that their OWL ontology including model of protein information mainly structured the related protein attributes that denoted as text before. They decided to add more instances to the OWL ontology and to work on automatic transformation of protein databases to protein ontology databases as a future work.

2.4.2 UMLS Semantic Network and OWL

Bontas EP, et.al. stated that ontology-based natural language processing and domain ontologies were used in order to develop a Semantic Web-driven retrieval system for the 'lung pathology' domain in the research project of 'A Semantic Web for Pathology'. UMLS (mainly Metathesaurus) was used as an input for the domain ontology. [19] In another paper [18], Bontas EP et.al. declared an information system for pathology combining NLP techniques and Semantic Web for Semantic Web driven ontology maintenance and generation like in [19] to carry out a content-driven retrieval and storage of medical images and reports. [18] Bontas EP, et.al. stated in both papers [18, 19] that the UMLS Semantic Network was translated to OWL in order to create a taxonomy of semantic relations and types as properties and classes, respectively. In both papers [18, 19] the authors did not give any detail or extra information about this translation to OWL.

Jimenez-Ruiz E. [20] indicated the creation of an OWL version of the UMLS SN by using a customized OWL constructor [21]. Jimenez-Ruiz E. [21] stated development of a tool extending Protégé-OWL and managing a collection of related ontologies. Nevertheless, in both references [20, 21] the authors did not give any detail or extra information about this creation of OWL version. Only there was a link for created OWL version of the UMLS SN (umls_semantic_network.owl) [20]. As it was not again specifically defined as OWL DL or OWL Lite, the mentioned translation to OWL outcome could be realized as OWL Full. A further inspection on the created OWL file [20] revealed that some descriptive information about the STYs was also identified as attributes of classes. Since individuals and classes are disjoint domains in OWL Lite and DL, when the attributes represented with above mentioned manner the interested class would be both a class and an individual at the same time. Therefore, the stated OWL [20] description could not be either OWL Lite or OWL DL but only OWL Full.

BioTop [23] including taxonomic hierarchies of classes and relation with range and domain restrictions derived from an improvement of the GENIA ontology. Like the UMLS SN, its backbone was composed by a taxonomic hierarchies of classes and relations hierarchy populated with sixty relations with domain and range constraints. The major dissimilarity between UMLS Semantic Network and BioTop was representation format. Schulz S., et.al. [22] generated mapping between the BioTop [23] and UMLS Semantic Network [60] and evaluated the relation interpretation. They used middle mapping file, which brought in both Biotop and UMLS Semantic Network. They sought for representation of UMLS Semantic Network covering semantic type hierarchies in addition to the range and domain limitations of the semantic relations by using BioTop. Since BioTop was in OWL DL, first the authors performed manually the representation of UMLS SN in OWL DL formalism. As the main aim of the authors was mapping process, relations between semantic types in UMLS SN were represented by using domain and range constructors in related property descriptions like in BioTop. Further detailed assessment on OWL version of UMLS SN developed by Schulz S. et.al. [22] can be seen at section 3.3.7.

Kashyap V., et.al. [24] also targeted UMLS Semantic Network representation in OWL. They recommended OWL constructors for representation of the semantic types and relations. They also discussed diverse potential interpretations of semantic relations between semantic types. Kashyap V., et.al. [24] tied up that the coherent understanding of the RLs in the UMLS Semantic Network ought to rely on the application. Further detailed assessment on recommendations by Kashyap V., et al. [24] can be seen at section 3.3.7.

CHAPTER 3

A NEW TOOL FOR UMLS SEMANTIC NETWORK

3.1 The Need of Tools for Supporting Knowledge-intensive Biomedical Actions

The need for knowledge-intensive activities in biomedical applications makes it essential to investigate whether the current formalism of UMLS SN can be used or if a knowledge-oriented language, such as OWL, is more suitable. The benefits of OWL for sustaining semantic activities and the limitations of UMLS SN are discussed in this section.

3.1.1 UMLS SN Limitations

The UMLS SN has been essentially stable over the last ten years. However, some SN users have suggested changes concerning coverage of additional semantic types and relations, updates to reflect current biomedical science content, and support for multiple inheritances organization and compatibility with ontological principles. Changes in formalism (e.g., description logics) have also been suggested [62]. Contradictions in the SN classification, even though semantic classification is vital to many applications, have been reported by Cimino et al. [74]. The SN types are recommended to be further simplified to shape basic-level semantic categories by Burgun et al. [75] for an ontological perspective. There has been research on auditing [57, 79, 80, 81] and simplifying [76, 77, 78] the UMLS semantic classification. The National Library of Medicine (NLM) tries to identify the current potential of the SN and its impact on applications and maintenance of the network and to analyze and discuss suggestions for changing and improving the SN.

The UMLS SN offers a high-level abstraction of the large Metathesaurus knowledge source to facilitate interaction with embedded UMLS knowledge. There are 53 different associative relationships in the UMLS SN which are used to specify 612 assertions in the SRSTR file [44] (e.g., Biologic Function, process_of, Organism) from which other assertions can be deduced. There are in entirety about 7000 assertions [48, 22]. Accordingly it is not easy for a user to fully comprehend the UMLS SN. The motivation for reducing the complexity of the SN is to make it easier for human comprehension and for system integration [74, 81]. However, humans in general have difficulty with formal notions.

The complexity of the SN, difficulty in understanding the formal notions by human, and UMLS SN content change management reveal that there is a need for intermediary tools that allow semantic representation of UMLS SN that can be understood and used by biomedical computer applications or systems. NetON proposes a solution to the above mentioned main problems by using several original algorithms to transform the formalism and semantics of UMLS SN to OWL sublanguages.

3.1.2 Why OWL?

There is knowledge inconsistencies on descriptions of concepts/categories used in diverse biomedical organizations, even though concepts and categories are used to guide biomedical domain activities [56]. Therefore, there is an obvious need for supervision mechanisms.

Semantic Web approaches for data management might aid to discover the suitable biomedical concept/category and description formalism for a particular situation.

OWL is the de facto language for the Semantic Web and is a standard for expressing ontologies. It inherits aspects of frames [68] to provide modeling constructs for formal representation of semantics knowledge [69]. Progress in diverse languages and technologies for querying and exploiting content is another benefit of OWL. Numerous software packages and applications can process knowledge represented in OWL, which supports the creation of reusable libraries [27, 26]. Furthermore, using a widespread formalism for representation of biomedical information (e.g., UMLS SN) makes possible improved knowledge supervision. The above mentioned items are the main reasons that NetON was designed to allow UMLS SN being encoded in OWL.

3.2 Motivation of NetON

Biomedical environment gets help from IS (information system) technologies to solve problems due to its complex nature. Biomedical ontologies and vocabularies play a vital role in the context of biomedical ISs. Being an abstract model, the UMLS SN is a valuable source for modeling contents of the biomedical domain. However, its use is limited due to its complex structure and formalization that is not in a suitable representation language to be reused and shared. The need for knowledge-intensive activities in biomedical applications makes it necessary to transform UMLS SN current formalism to a proper format that uses a knowledge-oriented language.

The limitations of UMLS SN (See section 3.1.1) and benefits of OWL (See section 3.1.2) indicates that Ontology Web Language is a suitable candidate standard for representation of UMLS SN. It is seen that major restrictions of UMLS SN content can be described by OWL constructors, with detailed inspection of OWL and UMLS SN.

The discovery of new information, relationships, and details necessitates the continuous evolvement of UMLS SN to adapt to biomedical requirements. Therefore, it is seen that any transformation effort on the formalism of UMLS SN should be done automatically to immediately reflect the change to the new formalism.

It is seen by the literature survey that there is no study on automatically transforming the maximum amount of UMLS SN information in OWL Sublanguages without changing the biomedical abstract model semantics. There was not also any study on transforming whole information included in UMLS SN to OWL without changing the owned semantics.

3.3 NetON Approach

The UMLS SN and OWL were investigated in detail to decide and construct an approach for transformation of UMLS SN to OWL sublanguages. As our main aim was to conserve the meaning of information during the transformation, only the syntactic conversion would not be enough. Therefore, we unified the syntactic conversion with the semantic one. During the building the related algorithms, we utilized standard way to define the transformation rules including the syntactic and semantic conversion regulations jointly. In this section, NetON approach for the process of transforming the UMLS SN to semantic expression in OWL sublanguages is described.

3.3.1 OWL Sublanguage Species of NetON

Being a NetON developer, I first understood the semantics of SRFIL (File containing descriptions of UMLS SN files) and SRFLD (File including fields descriptions inside UMLS SN files) files to use their semantics in first dimension algorithms during the development process of NetON. The user of NetON needs to download UMLS SN files [44] to his/her computer. The downloaded SRDEF (File consisting of basic information about the STYs and RLs) and SRSTR (File involving basic 612 assertions in UMLS SN) files (See Appendix E) are processed by NetON wherein the transformation processes to OWL Sublanguage Species take in place using first dimension algorithms. The developed first dimension algorithms of NetON were deployed as transformation bridges between the UMLS SN and OWL Sublanguage Species. (Figure 11)

OWL is defined as three different sublanguages by W3C's Web Ontology Working Group. Each sublanguage fulfills diverse phases of requirements [73]: OWL Lite, OWL DL and OWL Full. OWL Full contains OWL DL, including OWL Lite. (Figure 12) NetON used the rules of these sublanguages to generate OWL Species. Relevant menu options invoking the appropriate algorithms can be chosen to generate the OWL Species describing the machine accessible semantics of UMLS SN.

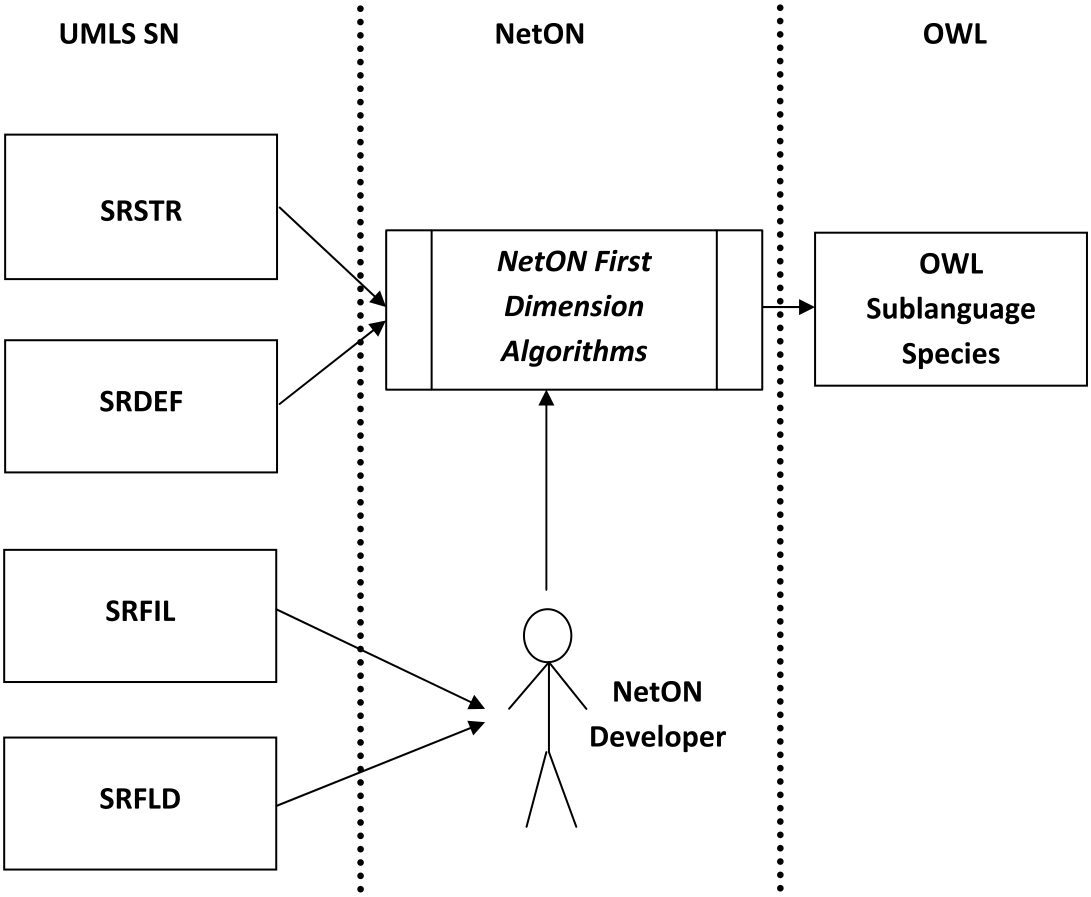


Figure 11 Transformation of the UMLS SN to OWL Sublanguage Species

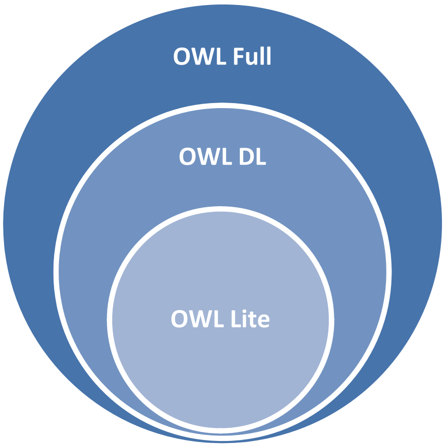


Figure 12 OWL sublanguages

In the NetON approach, according to above mentioned sublanguages, the following rules are applied to each OWL Species. Being entirely upward compatible with RDF, both semantically and syntactically, a generated OWL Full document utilizes a combination of OWL languages primitives. However, there is no complete or efficient reasoning support for OWL Full. For that reason, NetON also allows the generation of an OWL DL document in which of OWL constructors application to each other is prohibited to keep up a correspondence with well studied description logic. To make use of computational tractability of description logics, the vocabulary partitioning restriction is obeyed. The advantage of this is that a generated OWL DL document permits resourceful reasoning support. However, the entire compatibility with RDF is lost. NetON also includes an algorithm that generates an OWL Lite document, including the use of a further restricted constructor subset. Having restricted expressivity, an OWL Lite document allows easier implementation for tool builders. (Table 7)

One of the weaknesses of OWL documents is that even if some part of the document needs to be used by computer applications, the whole document must be handled. For applications or developers that may need only relations between semantic types, NetON has an option for excluding detailed information about

STYs/RLs inside the document. If users choose not to include detailed information the related algorithms exclude them form generated OWL Sublanguage Species (Table 8). The generated OWL documents without descriptive information for STYs and RLs are named by adding WOD (Without Descriptive Information) to the end of the related OWL Species name.

Table 7 OWL Sublanguage Species offered by NetON

Sublanguage	**With Hierarchical Relations of STYs & RLs**	**With Associative Relations Among STYs**	**With Detailed Information about STYs**	**With Detailed Information about RLs**
OWL Lite	√	-	-	-
OWLDL	√	√	√	-
OWL Full	√	√	√	√

Table 8 OWL Sublanguage Species offered by NetON without detailed information about STYs/RLs

Sublanguage	**With Hierarchical Relations of STYs & RLs**	**With Associative Relations Among STYs**
OWL Lite	√	-
OWLDL	√	√
OWL Full	√	√

3.3.1.1 Representation Analysis of OWL Sublanguage Species

UMLS SN is a well established, semi-formal ontology. The need for knowledge-intensive activities in biomedical applications makes it necessary to transform UMLS

SN current formalism to a proper format using a knowledge-oriented language. As stated in literature survey part of the dissertation document, OWL is a good candidate for this purpose.

The transformation of UMLS SN in OWL sublanguages requires the semantic interpretation of both UMLS SN current description and OWL standard. For this intend the UMLS SN files [44] and OWL standard [15] were examined to make understanding of their information. The semantic explanations of the UMLS SN and OWL are explained in literature survey part of the dissertation report. The transformation outcomes, which are OWL documents, are briefly described in Section 3.3.1.1.

In order to represent the UMLS SN semantically, Web Ontology Language offers a standard language for expressing ontologies that offers constructs for modeling. Since upper-level biomedical knowledge in UMLS Semantic Network is basically presented as the semantic relations and classification, OWL suits as a representation language to the current structure of UMLS SN by inheriting characteristics of frames and having basic constructs for defining relations and classes.

As OWL is written with XML syntax, every OWL Species generated by NetON mentioned in section 3.3.1.1 starts with following line:

```
<?xml version="1.0"?>
```

Generated OWL Species begin with owl:Ontology element, which contains comment:

```
<owl:Ontology>
     <rdfs:comment xml:lang="en">
        OWL Representation of the UMLS Semantic Network
     </rdfs:comment>
</owl:Ontology>
```

In OWL, the class extension means a set of individuals (instances) related with corresponding OWL class. Classes can be described by named or unnamed anonymous classes. A class identifier is used for a named class description. An anonymous class is described by introducing restrictions on the class extension. In NetON, besides named classes, two different anonymous class descriptions are used: Union combination of classes or a class that points up instances that assures a particular property restriction, respectively.

Supplementary features of classes can be defined by class axioms. In NetON generated OWL Species, rdfs:subClassOf constructor is used for merging class descriptions. A subset extension of a class is defined by using this constructor. (See Table 9)

There are two main property types in owl: Datatype and object. Individuals can be linked to individuals with 'object properties' and can be linked to 'datatype properties', respectively. In NetON, these two types of properties are used while transforming UMLS SN to OWL Species.

Extra features of properties can be defined by property axioms. In NetON generated OWL Species, the constructors rdfs:subPropertyOf and owl:inverseOf are used to represent property hierarchy and relations with other properties, respectively. Main used axioms during the transformation of UMLS SN to OWL Species can be seen in Table 9.

Table 9 Main axioms used during the transformation of UMLS SN to OWL Sublanguage Species

Transformation Axioms		**NetON OWL Sublanguage Species**	
		Class Axiom	**Property Axiom**
UMLS SN	**STY Hierarchy**	`rdfs:subClassOf`	
	RL Hierarchy		`rdfs:subPropertyOf`
	Inverse of RL		`rdf:inverseOf`

The algorithms in NetON are developed with the aim of maximum possible information transformation from UMLS SN to OWL sublanguages. The taxonomic hierarchy of STYs and RLs are only information that can be represented in OWL Lite Species due to the narrow set of valid constructors. Encompassing the representation in OWL Lite Species, assertions and STYs detailed information are also transformed into OWL DL Species. OWL Full Species comprises the OWL DL Species representation with different notations for STYs detailed information and also includes the RLs detailed information. Inheritance blockings are the only information that is not able to be transformed into any OWL sublanguage due to the lack of appropriate constructors in OWL standard. Primary constructors used for the transformation of UMLS SN to OWL sublanguages can be seen in Table 10.

Table 10 Primary constructors used during the transformation of UMLS SN to OWL sublanguages

Transformation Constructors		**NetON OWL Sublanguage Species**		
		OWL Lite	**OWL DL**	**OWL Full**
UMLS SN	**STY**	owl:Class	owl:Class	owl:Class
	RL	owl:objectProperty	owl:objectProperty	owl:objectProperty
	Assertion	-	owl:Restriction, owl:onProperty, owl:allValues, (owl:unionOf)	owl:Restriction, owl:onProperty, owl:allValues, (owl:unionOf)
	STYs Detailed Information	-	owl:Restriction, owl:onProperty, owl:hasValue owl:DatatypeProperty	Tags defined by NetON
	RLs Detailed Information	-	-	Tags defined by NetON
	Inheritance Blocking	-	-	-

3.3.1.1.1 OWL Lite Species

The OWL standard, as stated at literature survey part, extensively uses RDFS and RDF primitives. Web Ontology Language constructors are specializations of their RDF and RDFS counterparts. OWL documents are usually includes OWL ontologies. Since an rdf:RDF element is the root element of an OWL ontology, and it identifies numerous namespaces, the root element and related namespaces in each generated OWL Lite Species are characterized as following:

```
<rdf:RDF

     xmlns:owl="http://www.w3.org/2002/07/owl#"
     xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
     xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
     xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
     xml:base="http://birsengozdemir.com/SN_OWL_Full.xml#"
     xmlns="http://birsengozdemir.com/SN_OWL_Full.xml">

     ...

</rdf:RDF>
```

3.3.1.1.1.1 Semantic Type and Relation Transformation

Two main entities can be revealed in UMLS SN: semantic types (STYs) and semantic relations (RLs). Each STY represents a biomedical category (i.e., Organism).The UMLS SN describes core biomedical knowledge consisting of these entities.

In NetON, a category is represented in OWL Lite Species as a named class using owl:Class tag, which is a valid constructor in OWL Lite. In UMLS Semantic Network current representation [44], STYs aren’t stated as being mutually exclusive. Therefore, any statement with owl:disjointWith axiom is not used for described classes in NetON Lite Species.

A specialization of the generic class can be considered as the semantic type that was subclass of another one. In NetON, the STY hierarchy in the UMLS SN is represented as a subclass axiom (`rdfs:subClassOf)`, which is valid in OWL Lite. Accordingly, the taxonomic hierarchy of the UMLS Semantic Network STYs is transformed into OWL Lite. Let's give an example:

In SRDEF file [44]:

It is understood that 'Organism' is a semantic type (STY) from the following definition in SRDEF file:

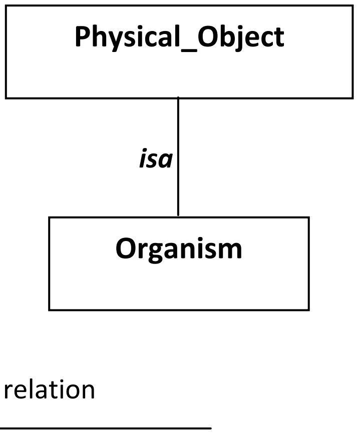
STY|T001|Organism|A1.1|Generally, a living individual, including all plants and animals.|Homozygote; Radiation Chimera; Transgenic Organisms|||orgm||

In SRSTR file [44]:

The following assertion presents that 'Organism' is subclass of 'Physical Object':

Organism|isa|Physical Object|D|

Graphical Representation of the assertion:



In a NetON OWL Lite document:

Class description in OWL:

```
<owl:Class rdf:ID="Organism" >
      <rdfs:subClassOf >
            <owl:Class rdf:about="#Physical_Object" />
      </rdfs:subClassOf>
</owl:Class>
```

Each RL represents a relationship between STYs in UMLS SN. The relationships used in the semantic network are also organized in an is-a hierarchy, with five roots: conceptually related to, functionally related to, temporally related to, spatially related to, physically related to. For instance, the relationship root 'functionally relates to' had several is-a children, including 'brings_about' which in turn has two children, including 'produces' and 'causes'. In UMLS SN documentation, the relationships in the semantic network are binary. In NetON, an UMLS SN relation is translated to OWL object property (owl:ObjectProperty).

A specialization of the generic property can be considered as the semantic relation being sub-property of another one. In NetON, the RL hierarchy in the UMLS SN is represented as a subproperty axiom (`rdfs:subPropertyOf`), which is valid in OWL Lite. Accordingly, the taxonomic hierarchy of the UMLS Semantic Network RLs is transformed into OWL Lite. Let's give an example:

In SRDEF file [44]:

It is understood that 'causes' is a relation type (RL) from the following definition in SRDEF file:

RL|T147|causes|R3.2.2|Brings about a condition or an effect. Implied here is that an agent, such as for example, a pharmacologic substance or an organism, has brought about the effect. This includes induces, effects, evokes, and etiology.||||CA|caused_by|

In SRSTR file [44]:

The following information presents that 'causes' is subclass of 'brings_about':

causes|isa|brings_about|D|

In a NetON OWL Lite document:

Property descriptions in OWL:

```
<owl:ObjectProperty rdf:ID="causes">
    <rdfs:subPropertyOf rdf:resource="#brings_about"/>
    ...
</owl:ObjectProperty>
```

Supplementary features of properties can be defined by property axioms. The relationships in the UMLS SN have inverses, which can be specified through OWL axioms involving the inverseOf role constructor. In NetON, the applicable constructor `owl:inverseOf` is used in OWL Lite Species. Thus, the inverses of the UMLS Semantic Network RLs are transformed into OWL Lite. Let's give an example:

In SRDEF file [44]:

It is understood that 'caused_by' is inverse of 'causes' from the following definition in SRDEF file:

RL|T147|causes|R3.2.2|Brings about a condition or an effect. Implied here is that an agent, such as for example, a pharmacologic substance or an organism, has brought about the effect. This includes induces, effects, evokes, and etiology.||||CA|caused_by|

In a NetON OWL Lite document:

Property descriptions in OWL:

```
<owl:ObjectProperty rdf:ID="causes">
    ...
    <owl:inverseOf rdf:resource="#caused_by"/>
</owl:ObjectProperty>
```

Since, some relationships in UMLS SN are symmetric properties; this is taken into account by the developed algorithms in NetON. Let's see the following example on interect_with property description:

```
<owl:ObjectProperty rdf:ID="interacts_with"  >
 <rdfs:subPropertyOf rdf:resource="#affects"  />
  <owl:inverseOf rdf:resource="#interacts_with" />
  <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#SymmetricProperty"/>
</owl:ObjectProperty>
```

3.3.1.1.1.2 Assertion Transformation

Relations between STYs can be called as assertions (e.g. *Animal* exhibits *Behavior*). One way for transformation of the assertions represented in the SRSTR file into the OWL Sublanguage Species can be using property axioms including constructors rdfs:range and rdfs:domain. The subject of the property can be declared as to be a member of the class description indicated by domain axiom. The object of the property indicated by range axiom can be affirmed as to be either a member of the class description or a data value in the specified data range. Multiple domain and range axioms can also be defined for any property. Let's give an example:

In SRDEF file [44]:

It is understood that 'exhibits' is a relation/property:

RL|T145|exhibits|R3.3.2|Shows or demonstrates.||||EX|exhibited_by|

In SRSTR file [44]:

The following information presents two assertions including 'exhibits':

Animal|exhibits|Behavior|D|

Group|exhibits|Behavior|D|

In OWL:

Property descriptions in OWL:

```
<owl:ObjectProperty rdf:ID="exhibits">
    ...
    <rdfs:domain rdf:resource="#Animal"/>
    <rdfs:domain rdf:resource="#Group"/>
    <rdfs:range rdf:resource="#Behavior"/>
</owl:ObjectProperty>
```

Multiple domain or range axioms for any property ought to be understood as intersection of all domains or range values. Therefore, the above property description in OWL limits the domain of ‘exhibit’ property to instances that are members of the conjunction of ‘Animal’ and ‘Group’ classes. However, exhibits (Animal, Behavior) and exhibits (Group, Behavior) assertions in SRSTR file imply that domain of the ‘exhibits’ property can be either an Animal or a Group. The owl:unionOf constructor can be used for an anonymous class description to declare that multiple classes can act as the domain of the ‘exhibits’ property:

```
<owl:ObjectProperty rdf:ID="exhibits">
    ...
    <rdfs:domain>
      <owl:Class>
         <owl:unionOf rdf:parseType="Collection">
            <owl:Class rdf:about="#Animal"/>
            <owl:Class rdf:about="#Group"/>
          </owl:unionOf>
       </owl:Class>
    </rdfs:domain>
    <rdfs:range rdf:resource="#Behavior"/>
</owl:ObjectProperty>
```

The constructor owl:unionOf cannot be used in OWL Lite ontologies. Therefore, relations between STYs which are assertions (e.g. *Animal* exhibits *Behavior*) are not able to be transformed into OWL Lite Species using property axioms including constructors rdfs:range and rdfs:domain due to the limitation of OWL Lite standard.

The basic assertions represented in SRSTR file can also be transformed as axioms to OWL Species, as stated above. Since axioms originate from inheritance hierarchies of the various semantic types and relationships, the associative relations among STYs seem to be denoted through axioms involving the rdfs:subClassOf, owl:Restriction, owl:onProperty, owl:someValuesFrom, constructors. For example, property_of (Amino_Acid_Sequence, Amino_Acid_Peptide_or_Protein) and property_of (Amino_Acid_Sequence, Gene_or_Genome) assertions that can be stated as:

In SRSTR file [44]:

Amino Acid Sequence|property_of|Amino Acid, Peptide, or Protein|D|

Amino Acid Sequence|property_of|Gene or Genome|D|

In OWL:

```
<owl:Class rdf:ID=" Amino_Acid_Sequence " >
    ...
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty rdf:resource="#property_of" />
        <owl:someValuesFrom>
          <owl:Class rdf:about="# Gene_or_Genome" />
        </owl:someValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty rdf:resource="#property_of" />
        <owl:someValuesFrom>
          <owl:Class rdf:about="#Amino_Acid_Peptide_or_Protein"/>
        </owl:someValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
</owl:Class>
```

The above representation of property_of (Amino_Acid_Sequence, Gene_or_Genome) and property_of (Amino_Acid_Sequence, Amino_Acid_Peptide_or_Protein) assertions is suitable for OWL Lite rules. However, deeper examination revealed that semantic meaning of the above representation and the UMLS SN intended meaning were different. According to the above representation with owl:someValuesFrom constructor, an individual of 'Amino_Acid_Sequence' class should have at least one value of 'property_of' relation to be an individual of 'Gene_or_Genome' and should also have at least one value of 'property_of' relation to be an individual of 'Amino_Acid_Peptide_or_Protein', simultaneously. This represented semantic is not actually intended by UMLS SN knowledge. Therefore, this representation is not used to transform assertions in UMLS SN into OWL Lite Species.

The associative relations among STYs appear also to be denoted through axioms involving the rdfs:subClassOf, owl:Restriction, owl:onProperty, owl:someValuesFrom, and owl:unionOf constructors. For example, property_of (Amino_Acid_Sequence, Amino_Acid_Peptide_or_Protein) and property_of (Amino_Acid_Sequence, Gene_or_Genome) assertions that can be stated as:

```
<owl:Class rdf:ID=" Amino_Acid_Sequence " >
    ...
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty rdf:resource="#property_of" />
        <owl:someValuesFrom>
         <owl:Class>
           <owl:unionOf rdf:parseType="Collection">
            <owl:Class rdf:about="#Gene_or_Genome"/>
            <owl:Class rdf:about="#Amino_Acid_Peptide_or_Protein"/>
          </owl:unionOf>
         </owl:Class>
        </owl:someValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
</owl:Class>
```

According to the above representation with owl:unionOf constructor, an individual of Amino_Acid_Sequence class should have at least one value of property_of relation to be an individual of conjunction of Gene_or_Genome and Amino_Acid_Peptide_or_Protein classes. Even this represented semantic can specify UMLS SN knowledge, it doesn't exclude existence of other individuals for property_of (Amino_Acid_Sequence, c) assertion where c doesn't belong to either Gene_or_Genome or Amino_Acid_Peptide_or_Protein classes. Additionally, the owl:unionOf constructor is not allowed to be used in OWL Lite standard. Therefore, this representation is not able to be used to transform assertions in UMLS SN into OWL Lite Species.

The associative relations among STYs seem also to be denoted through most appropriate axioms involving the rdfs:subClassOf, owl:Restriction, owl:onProperty, owl:allValuesFrom, constructors. For example, property_of (Amino_Acid_Sequence, Amino_Acid_Peptide_or_Protein) and property_of (Amino_Acid_Sequence, Gene_or_Genome) assertions that can be stated as:

In SRSTR file [44]:

Amino Acid Sequence|property_of|Amino Acid, Peptide, or Protein|D|

Amino Acid Sequence|property_of|Gene or Genome|D|

In OWL:

```
<owl:Class rdf:ID=" Amino_Acid_Sequence " >
    ...
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty rdf:resource="#property_of" />
        <owl:allValuesFrom>
          <owl:Class rdf:about="# Gene_or_Genome" />
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty rdf:resource="#property_of" />
        <owl:allValuesFrom>
          <owl:Class rdf:about="#Amino_Acid_Peptide_or_Protein"/>
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
</owl:Class>
```

The above representation of property_of (Amino_Acid_Sequence, Gene_or_Genome) and property_of (Amino_Acid_Sequence, Amino_Acid_Peptide_or_Protein) assertions is suitable for OWL Lite rules. However, further examination revealed that the above representation semantic meaning comprise inconsistency. According to the above representation with owl:allValuesFrom constructor, an individual of Amino_Acid_Sequence class should have all values of property_of relation to be an individual of Gene_or_Genome and should also have all values of property_of relation to be an individual of Amino_Acid_Peptide_or_Protein, simultaneously. Since Gene_or_Genome and Amino_Acid_Peptide_or_Protein classes are different categories/classes, the above represented semantic is actually inconsistent. Therefore, this representation cannot be used to transform assertions in UMLS SN into OWL Lite Species.

The associative relations among STYs appears also to be denoted through most appropriate axioms involving the rdfs:subClassOf, owl:Restriction, owl:onProperty, owl:allValuesFrom, and owl:unionOf constructors. For example, property_of (Amino_Acid_Sequence, Amino_Acid_Peptide_or_Protein) and property_of (Amino_Acid_Sequence, Gene_or_Genome) assertions that can be stated as:

```
<owl:Class rdf:ID=" Amino_Acid_Sequence " >
    ...
    <rdfs:subClassOf>
```

```
      <owl:Restriction>
        <owl:onProperty rdf:resource="#property_of" />
        <owl:allValuesFrom>
         <owl:Class>
           <owl:unionOf rdf:parseType="Collection">
             <owl:Class rdf:about="#Gene_or_Genome"/>
             <owl:Class rdf:about="#Amino_Acid_Peptide_or_Protein"/>
           </owl:unionOf>
         </owl:Class>
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
</owl:Class>
```

According to the above representation with owl:unionOf constructor, an individual of Amino_Acid_Sequence class should have all values of property_of relation to be an individual of conjunction of Amino_Acid_Peptide_or_Protein and Gene_or_Genome classes. This represented semantic seems to be the best one for specification of assertions, it leaves out existence of other individuals for property_of (Amino_Acid_Sequence, c) assertion where c doesn't belong to either Gene_or_Genome or Amino_Acid_Peptide_or_Protein classes. However, the owl:unionOf constructor cannot be used in OWL Lite ontologies. Therefore, this representation is not used to transform assertions in UMLS SN into OWL Lite Species. As a result of above mentioned analysis, the assertions in UMLS SN are not able to be transformed into the OWL Lite Species due to the limitations of OWL Lite standard.

3.3.1.1.1.3 Descriptive Information Transformation

The descriptive information about the STYs is specified in the SRDEF file (e.g. unique identifier, definition, examples, etc.). They can be identified as attributes of each category/class in OWL. The attributes of a class are commonly represented as following:

In SRDEF file [44]:

It is understood that 'Organism' has some descriptive information (e.g. Type, Name, Unique Identifier, etc.) from the following definition:

STY|T001|Organism|A1.1|Generally, a living individual, including all plants and animals.|Homozygote; Radiation Chimera; Transgenic Organisms|||orgm||

In OWL:

```
<owl:Class rdf:ID="Organism">
   ...
    <RecordType>STY</RecordType>
    <Label>Organism</Label>
    <UniqueIdentifier>T001</ UniqueIdentifier>
   ...
</owl:Class>
```

In OWL, as stated above, the class extension means a set of individuals (instances) related with corresponding OWL class. Individuals and classes are disjoint domains in OWL Lite. When the attributes are represented like in above example the class (`Organism)` will be both a class and an individual at the same time. Therefore, the mentioned descriptive information is not able to be transformed into OWL Lite Species in this format.

The descriptive information about the STYs can also be identified by using value restriction (`owl:hasValue`) on a data type property. Let's illustrate the above example with the `owl:hasValue` constuctor usage:

In OWL:

```
<?xml version="1.0"?>
<rdf:RDF >
   ...
   <owl:Class rdf:ID="Organism">
   <rdfs:subClassOf  >
     <owl:Restriction>
        <owl:onProperty rdf:resource="#SNRecordType"/>
        <owl:hasValue rdf:datatype="xsd:string">STY</owl:hasValue>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf  >
      <owl:Restriction>
        <owl:onProperty rdf:resource="#SNRecordName" />
        <owl:hasValue rdf:datatype="xsd:string">Organism</owl:hasValue>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf  >
      <owl:Restriction>
        <owl:onProperty rdf:resource="#SNUniqueIdentifier"/>
        <owl:hasValue rdf:datatype="xsd:string">T001</owl:hasValue>
      </owl:Restriction>
    </rdfs:subClassOf>
    ...
  </owl:Class>
  ...
  <owl:DatatypeProperty rdf:ID="SNRecordType"/>
  <owl:DatatypeProperty rdf:ID=" SNRecordName"/>
  <owl:DatatypeProperty rdf:ID="SNUniqueIdentifier"/>
</rdf:RDF >
```

However, the `owl:hasValue` is not allowed to be used in OWL Lite ontologies. Therefore, the mentioned descriptive information is not able to be transformed into OWL Lite Species in this format due to the limitations of OWL Lite standard.

The descriptive information about the RLs is also specified in the SRDEF file (e.g. unique identifier, definition, abbreviation, etc.). They can be identified as attributes of each relation/property in OWL. The attributes of an object property are commonly represented as following:

In SRDEF file [44]:

It is understood that 'treats' has some descriptive information (e.g. Type, Name, Unique Identifier, etc.) from the following definition:

RL|T154|treats|R3.1.2|Applies a remedy with the object of affecting a cure or managing a condition.||||TS|treated_by|

In OWL:

```
<owl:ObjectProperty rdf:ID="treats">
    ...
    <RecordType>RL</RecordType>
    <Label>treats</Label>
    <UniqueIdentifier>T154</ UniqueIdentifier>
    ...
</owl:ObjectProperty>
```

Individuals and properties should be defined as disjoint domains in OWL Lite. When the attributes are represented like in above example the property will be used as both a property and an individual at the same time. Therefore, the mentioned descriptive information is not able to be transformed into OWL Lite Species in this format too due to the limitations of OWL Lite standard.

3.3.1.1.2 OWL DL Species

The root element and related namespaces in each generated OWL DL Species are characterized as following:

```
<rdf:RDF

        xmlns:owl="http://www.w3.org/2002/07/owl#"
        xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
        xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
        xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
        xml:base="http://birsengozdemir.com/SN_OWL_DL.xml#"
        xmlns="http://birsengozdemir.com/SN_OWL_DL.xml">

        ...

</rdf:RDF>
```

3.3.1.1.2.1 Semantic Type and Relation Transformation

There are two main entities in UMLS SN, as stated before: semantic types (STYs) and semantic relations (RLs). The representations of semantic types (STYs) and semantic relations (RLs) in OWL DL Species are completed by generated algorithms using the same principles with OWL Lite transformation which are explained above. (See section 3.3.1.1.1)

In NetON, a category is also represented in OWL DL Species as a named class using owl:Class tag, which is a valid constructor in OWL DL. In UMLS Semantic Network current representation [44], STYs doesn't state as being mutually exclusive. Therefore, any statement with owl:disjointWith axiom is not used for described classes in NetON DL Species too.

In NetON, the STY hierarchy in the UMLS SN is represented as a subclass axiom (`rdfs:subClassOf`), which is valid in OWL DL standard. Accordingly, the taxonomic hierarchy of the UMLS Semantic Network STYs is transformed into OWL DL. Let's give an example:

In SRDEF file [44]:

It is understood that 'Virus' is a semantic type (STY) from the following definition in SRDEF file:

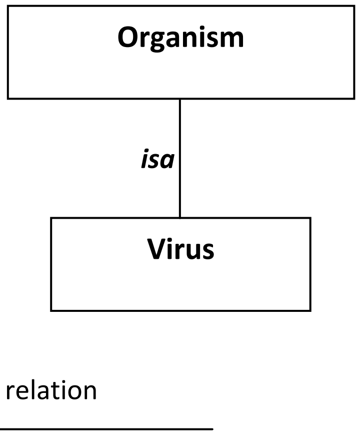
STY|T005|Virus|A1.1.3|An organism consisting of a core of a single nucleic acid enclosed in a protective coat of protein. A virus may replicate only inside a host living cell. A virus exhibits some but not all of the usual characteristics of living things.|Coliphages; Echovirus 6, Human; Parvoviridae; Fort Morgan virus|||virs||

In SRSTR file [44]:

The following assertion presents that 'Virus' is subclass of 'Organism':

Virus|isa|Organism|D|

Graphical Representation of the assertion:



In a NetON OWL DL document:

Class description in OWL DL:

```
<owl:Class rdf:ID="Virus" >
      <rdfs:subClassOf >
            <owl:Class rdf:about="#Organism" />
      </rdfs:subClassOf>
</owl:Class>
```

In NetON OWL DL Species, an UMLS SN relation is translated to OWL object property (owl:ObjectProperty).

In NetON, the RL hierarchy in the UMLS SN is represented as a subproperty axiom (`rdfs:subPropertyOf`), which is valid in OWL DL standard. Accordingly, the taxonomic hierarchy of the UMLS Semantic Network RLs is transformed into OWL DL Species. Let’s give an example:

In SRDEF file [44]:

It is understood that ‘part_of’ is a relation type (RL) from the following definition in SRDEF file:

RL|T133|part_of|R1.1|Composes, with one or more other physical units, some larger whole. This includes component of, division of, portion of, fragment of, section of, and layer of.||||PT|has_part|

In SRSTR file [44]:

The following information presents that ‘part_of’ is subclass of ‘brings_about’:

part_of|isa|physically_related_to|D|

In a NetON OWL DL document:

Property descriptions in OWL DL:

```
<owl:ObjectProperty rdf:ID="part_of">
   <rdfs:subPropertyOf rdf:resource="#physically_related_to"/>
   ...
</owl:ObjectProperty>
```

Since the relationships in the UMLS SN had inverses, the applicable constructor `owl:inverseOf` is used in OWL DL Species. Thus, the inverses of the UMLS Semantic Network RLs are transformed into OWL DL Species. Let's give an example:

In SRDEF file [44]:

It is understood that 'has_part' is inverse of 'part_of' from the following definition in SRDEF file:

RL|T133|part_of|R1.1|Composes, with one or more other physical units, some larger whole. This includes component of, division of, portion of, fragment of, section of, and layer of.||||PT|has_part|

In a NetON OWL DL document:

Property descriptions in OWL DL:

```
<owl:ObjectProperty rdf:ID="part_of">
    ...
    <owl:inverseOf rdf:resource="#has_part"/>
</owl:ObjectProperty>
```

3.3.1.1.2.2 Assertion Transformation

One way for transformation of the assertions (e.g. *Virus* causes *Pathologic Function*) represented in the SRSTR file into the OWL Species is using property axioms including constructors rdfs:range and rdfs:domain. Multiple domain and range axioms can also be defined for any property, as it is stated before. Let's give an example:

In SRDEF file [44]:

It is understood that 'causes' is a relation/property:

RL|T147|causes|R3.2.2|Brings about a condition or an effect. Implied here is that an agent, such as for example, a pharmacologic substance or an organism, has brought about the effect. This includes induces, effects, evokes, and etiology.||||CA|caused_by|

In SRSTR file [44]:

The following information presents all assertions including 'causes':

Bacterium|causes|Pathologic Function|D|

Fungus|causes|Pathologic Function|D|

Virus|causes|Pathologic Function|D|

Invertebrate|causes|Pathologic Function|D|

Manufactured Object|causes|Anatomical Abnormality|D|

Manufactured Object|causes|Injury or Poisoning|D|

Manufactured Object|causes|Pathologic Function|D|

Rickettsia or Chlamydia|causes|Pathologic Function|D|

Substance|causes|Anatomical Abnormality|D|

Substance|causes|Injury or Poisoning|D|

Substance|causes|Pathologic Function|D|

In OWL:

Property descriptions in OWL:

```
<owl:ObjectProperty rdf:ID="causes">
    ...
    <rdfs:domain rdf:resource="#Bacterium"/>
    <rdfs:domain rdf:resource="#Fungus"/>
    <rdfs:domain rdf:resource="#Invertebrate"/>
    <rdfs:domain rdf:resource="#Manufactured Object"/>
    <rdfs:domain rdf:resource="#Rickettsia or Chlamydia"/>
    <rdfs:domain rdf:resource="#Substance"/>
    <rdfs:domain rdf:resource="#Virus"/>
    <rdfs:range rdf:resource="#Pathologic Function"/>
    <rdfs:range rdf:resource="#Anatomical Abnormality"/>
    <rdfs:range rdf:resource="#Injury or Poisoning"/>
</owl:ObjectProperty>
```

Multiple domain or range axioms for any property ought to be understood as intersection of all domains or range values. Therefore, the above property description in OWL limits the range of 'causes' property to instances that are members of the conjunction of 'Pathologic Function', 'Anatomical Abnormality' and 'Injury or Poisoning' classes. Nevertheless, causes (Manufactured Object, Anatomical Abnormality), causes (Fungus, Pathologic Function) and causes (Substance, Injury or Poisoning) assertions, for example, in SRSTR file imply that range of the 'causes' property can be either an 'Pathologic Function', an 'Anatomical Abnormality' or an 'Injury or Poisoning'.

In order to produce solution, the owl:unionOf constructor can also be used for an anonymous class description in order to declare that multiple classes can act as the domain or range of the 'causes' property:

```
<owl:ObjectProperty rdf:ID="causes">
    ...
    <rdfs:domain>
      <owl:Class>
         <owl:unionOf rdf:parseType="Collection">
             <owl:Class rdf:about="#Bacterium"/>
             <owl:Class rdf:about="#Fungus"/>
             <owl:Class rdf:about="#Invertebrate"/>
             <owl:Class rdf:about="#Manufactured Object"/>
             <owl:Class rdf:about="#Rickettsia or Chlamydia"/>
             <owl:Class rdf:about="#Anatomical Abnormality"/>
             <owl:Class rdf:about="#Injury or Poisoning"/>
          </owl:unionOf>
       </owl:Class>
    </rdfs:domain>
    <rdfs:range>
      <owl:Class>
         <owl:unionOf rdf:parseType="Collection">
            <owl:Class rdf:about="#Pathologic Function"/>
            <owl:Class rdf:about="#Substance"/>
            <owl:Class rdf:about="#Virus"/>
          </owl:unionOf>
       </owl:Class>
    </rdfs:range>
</owl:ObjectProperty>
```

The constructor owl:unionOf can be used in OWL DL ontologies. Therefore, relations between STYs which are assertions (e.g. *Fungus* causes *Pathologic Function*) can be transformed into OWL DL Species using property axioms including constructors rdfs:range, rdfs:domain and owl:unionOf.

In UMLS SN documents, on the other hand, it is understood that particular range constraint(s) are only valid with particular domain constraint(s) as it was also stated by Schulz S. et.al. [22]. In UMLS SN, for example, the range of the relation ‘causes’ is limited to ‘Pathologic Function’, ‘Anatomical Abnormality’ and ‘Injury or Poisoning’ and its domain to ‘Bacterium’, ‘Fungus’, ‘Virus’, ‘Invertebrate’, ‘Rickettsia or Chlamydia’, ‘Manufactured Object’ and ‘Substance’.

It can be seen that some assertions like causes(Bacterium, Injury or Poisoning) or causes(Fungus, Anatomical Abnormality) are not allowed in UMLS SN by looking all the assertions in SRSTR file that includes ‘causes’ relation stated above. The following table shows the domain and range of ‘causes’ relation and valid assertions between them. (Table11)

Table 11 The range and domain of ‘causes’ relation and valid assertions between them

‘causes’ relation		**RANGE**		
		Pathologic Function	**Injury or Poisoning**	**Anatomical Abnormality**
DOMAIN	**Bacterium**	√	-	-
	Fungus	√	-	-
	Virus	√	-	-
	Invertebrate	√	-	-
	Rickettsia or Chlamydia	√	-	-
	Manufactured Object	√	√	√
	Substance	√	√	√

From the above explanation, it is understood that even the owl:unionOf constructor is used to declare that multiple classes can act as the domain or range of the ‘causes’ property, it will not sufficiently represent the assertions in UMLS SN having meaning that particular range constraints are only valid with particular domain constraints. Ignoring this and using simply the union of categories/classes in domain and range constraints will lead to misinterpreted models.

The associative relations among STYs seem to be denoted through axioms involving the rdfs:subClassOf, owl:Restriction, owl:onProperty, owl:someValuesFrom, constructors. For example, causes (Substance, Anatomical Abnormality), causes (Substance, Injury or Poisoning) and causes (Substance, Pathologic Function) assertions that can be stated as:

In SRSTR file:

Substance|causes|Anatomical Abnormality|D|

Substance|causes|Injury or Poisoning|D|

Substance|causes|Pathologic Function|D|

In OWL:

```
<owl:Class rdf:ID="Substance" >
    ...
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty rdf:resource="#causes" />
        <owl:someValuesFrom>
          <owl:Class rdf:about="#Anatomical Abnormality" />
        </owl:someValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty rdf:resource="#causes" />
        <owl:someValuesFrom>
          <owl:Class rdf:about="#Injury or Poisoning"/>
        </owl:someValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf>
      <owl:Restriction>
```

```
        <owl:onProperty rdf:resource="#causes" />
        <owl:someValuesFrom>
          <owl:Class rdf:about="#Pathologic Function"/>
        </owl:someValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
</owl:Class>
```

The above representation of causes (Substance, Pathologic Function), causes (Substance, Injury or Poisoning) and causes (Substance, Anatomical Abnormality) assertions is fitting with OWL DL rules. Nevertheless, detailed inspection exposed that the UMLS SN intended meaning and the semantics of above representation were dissimilar. According to the above representation with owl:someValuesFrom constructor, an individual of 'Substance' class should have at least one value of 'causes' relation to be an individual of 'Anatomical Abnormality' and should also have at least one value of 'causes' relation to be an individual of 'Injury or Poisoning' and should moreover have at least one value of 'causes' relation to be an individual of 'Pathologic Function', simultaneously. This represented semantic is not actually intended by UMLS SN knowledge. Therefore, this representation is not used to transform assertions in UMLS SN into OWL DL Species.

The associative relations among STYs appear also to be denoted through axioms involving the rdfs:subClassOf, owl:Restriction, owl:onProperty, owl:someValuesFrom, and owl:unionOf constructors. For example, causes (Substance, Anatomical Abnormality), causes (Substance, Injury or Poisoning) and causes (Substance, Pathologic Function) assertions that can be stated as:

```
<owl:Class rdf:ID="Substance" >
    ...
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty rdf:resource="#causes" />
        <owl:someValuesFrom>
         <owl:Class>
           <owl:unionOf rdf:parseType="Collection">
             <owl:Class rdf:about="#Anatomical Abnormality"/>
             <owl:Class rdf:about="#Injury or Poisoning"/>
             <owl:Class rdf:about="#Pathologic Function"/>
            </owl:unionOf>
         </owl:Class>
        </owl:someValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
</owl:Class>
```

According to the above representation with owl:unionOf constructor, an individual of 'Substance' class should have at least one value of 'causes' relation to be an individual of conjunction of 'Anatomical Abnormality', 'Injury or Poisoning' and 'Pathologic Function' classes. Even this represented semantic can specify UMLS SN knowledge, it doesn't exclude existence of other individuals for causes (Substance, *c*) assertion where *c* doesn't belong to either 'Anatomical Abnormality', 'Injury or Poisoning' or 'Pathologic Function' classes. The owl:unionOf constructor can be used in OWL DL ontologies. However, this representation is not able to be used to transform assertions in UMLS SN into OWL DL Species, due to the above mentioned explanations.

The associative relations among STYs seem also to be denoted through most appropriate axioms involving the rdfs:subClassOf, owl:Restriction, owl:onProperty, owl:allValuesFrom, constructors. For example, causes (Substance, Anatomical Abnormality), causes (Substance, Injury or Poisoning) and causes (Substance, Pathologic Function) assertions that can be stated as:

In OWL:

```
<owl:Class rdf:ID="Substance" >
```

```
    ...
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty rdf:resource="#causes" />
        <owl:allValuesFrom>
          <owl:Class rdf:about="#Anatomical Abnormality" />
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty rdf:resource="#causes" />
        <owl:allValuesFrom>
          <owl:Class rdf:about="#Injury or Poisoning"/>
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty rdf:resource="#causes" />
        <owl:allValuesFrom>
          <owl:Class rdf:about="#Pathologic Function"/>
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
</owl:Class>
```

The above representation of causes (Substance, Anatomical Abnormality), causes (Substance, Injury or Poisoning) and causes (Substance, Pathologic Function) assertions is suitable for OWL DL rules. However, further examination revealed that the above representation semantic meaning comprise inconsistency. According to the above representation with owl:allValuesFrom constructor, an individual of 'Substance' class should have all values of 'causes' relation to be an individual of 'Anatomical Abnormality' and should also have all values of 'causes' relation to be an individual of 'Injury or Poisoning' and should moreover have all values of 'causes' relation to be an individual of ‘Pathologic Function’, simultaneously. Since ‘Anatomical Abnormality’, ‘Injury or Poisoning’ and ‘Pathologic Function’ classes are different categories/classes, the above represented semantic is actually inconsistent. Therefore, this representation is not able to be used to transform assertions in UMLS SN into OWL DL Species.

The associative relations among STYs appear also to be denoted through most appropriate axioms involving the rdfs:subClassOf, owl:Restriction, owl:onProperty, owl:allValuesFrom, and owl:unionOf constructors. For example, causes (Substance, Anatomical Abnormality), causes (Substance, Injury or Poisoning) and causes (Substance, Pathologic Function) assertions that can be stated as:

```
<owl:Class rdf:ID="Substance" >
    ...
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty rdf:resource="#causes" />
        <owl:allValuesFrom>
         <owl:Class>
           <owl:unionOf rdf:parseType="Collection">
            <owl:Class rdf:about="#Anatomical Abnormality"/>
            <owl:Class rdf:about="#Injury or Poisoning"/>
            <owl:Class rdf:about="#Pathologic Function"/>
           </owl:unionOf>
         </owl:Class>
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
</owl:Class>
```

According to the above representation with owl:unionOf constructor, an individual of ‘Substance’ class should have all values of ‘causes’ relation to be an individual of conjunction of ‘Anatomical Abnormality’, ‘Injury or Poisoning’ and ‘Pathologic

Function’ classes. This represented semantic seems to be the best one for specification of assertions, it leaves out existence of other individuals for causes (Substance, *c*) assertion where *c* doesn’t belong to either ‘Anatomical Abnormality’, ‘Injury or Poisoning’ and ‘Pathologic Function’ classes. The owl:unionOf constructor can be used in OWL DL ontologies. As a result of above mentioned analysis, the last representation is used to transform assertions in UMLS SN into OWL DL Species.

3.3.1.1.2.3 Descriptive Information Transformation

The descriptive information about the STYs is frequently identified as attributes of each category/class in OWL. The attributes of a class are commonly represented as following:

In SRDEF file [44]:

It is understood that ‘Fungus’ has some descriptive information (e.g. TreeNumber, Examples, Abbreviation, etc.) from the following definition:

STY|T004|Fungus|A1.1.2|A eukaryotic organism characterized by the absence of chlorophyll and the presence of a rigid cell wall. Included here are both slime molds and true fungi such as yeasts, molds, mildews, and mushrooms.|Aspergillus clavatus; Blastomyces; Helminthosporium; Neurospora|||fngs||

In OWL:

```
<owl:Class rdf:ID="Fungus">
    ...
   <TreeNumber> A1.1.2</TreeNumber>
   <Examples>
    Aspergillus clavatus; Blastomyces; Helminthosporium; Neurospora
   </Examples>
   <Abbreviation>fngs</Abbreviation>
```

```
    ...
</owl:Class>
```

Individuals and classes are disjoint domains in OWL DL standard. When the attributes represented like in above example the class (`Fungus`) will be both a class and an individual at the same time. Therefore, the mentioned descriptive information is not transformed into OWL DL Species in this format.

The descriptive information about the STYs can also be identified by using value restriction (`owl:hasValue`) on a data type property. Let's illustrate the above example with the `owl:hasValue` constructor usage:

In OWL:

```
<?xml version="1.0"?>
<rdf:RDF >
   ...
   <owl:Class rdf:ID="Fungus">
   <rdfs:subClassOf  >
     <owl:Restriction>
        <owl:onProperty rdf:resource="#SNTreeNumber"/>
        <owl:hasValue rdf:datatype="xsd:string">A1.1.2</owl:hasValue>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf  >
      <owl:Restriction>
        <owl:onProperty rdf:resource="#SNExamples"  />
        <owl:hasValue rdf:datatype="xsd:string">
          Aspergillus clavatus; Blastomyces; Helminthosporium; Neurospora
        </owl:hasValue>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf  >
      <owl:Restriction>
        <owl:onProperty rdf:resource="#SNAbbreviation"/>
        <owl:hasValue rdf:datatype="xsd:string">fngs</owl:hasValue>
      </owl:Restriction>
    </rdfs:subClassOf>
    ...
  </owl:Class>
  ...
  <owl:DatatypeProperty rdf:ID="SNTreeNumber"/>
  <owl:DatatypeProperty rdf:ID="SNExamples"/>
  <owl:DatatypeProperty rdf:ID="SNAbbreviation"/>
</rdf:RDF >
```

Data type properties relate objects to datatype values. Data type properties SNRecordType, SNRecordName, SNRecordDefinition, SNUniqueIdentifier, SNTreeNumber, SNExamples, SNAbbreviation, SNUsageNotes, SNNonHumanFlag and SNMainCategory are used in generated OWL DL Species of NetON.

Since the `owl:hasValue` is allowed to be used in OWL DL, the mentioned descriptive information is transformed into OWL DL Species in this format.

Given that semantic types in UMLS SN are either Event or Entity, this is taken into account by the developed algorithms in NetON. Let’s see the following examples:

```
<owl:Class rdf:ID="Fungus">
   <rdfs:subClassOf  >
     <owl:Restriction>
        <owl:onProperty rdf:resource="#SNMainCategory"/>
        <owl:hasValue rdf:datatype="xsd:string">Entity</owl:hasValue>
      </owl:Restriction>
    </rdfs:subClassOf>
    ...
</owl:Class>

<owl:Class rdf:ID=" Pathologic_Function">
   <rdfs:subClassOf  >
     <owl:Restriction>
        <owl:onProperty rdf:resource="#SNMainCategory"/>
```

```
          <owl:hasValue rdf:datatype="xsd:string">Event</owl:hasValue>
        </owl:Restriction>
    </rdfs:subClassOf>
    ...
</owl:Class>
```

The descriptive information about the RLs is usually identified as attributes of each relation/property in OWL. The attributes of an object property are commonly represented as following:

In SRDEF file [44]:

It is understood that 'causes' has some descriptive information (e.g. TreeNumber, Abbreviation, etc.) from the following definition:

RL|T147|causes|R3.2.2|Brings about a condition or an effect. Implied here is that an agent, such as for example, a pharmacologic substance or an organism, has brought about the effect. This includes induces, effects, evokes, and etiology.||||CA|caused_by|

In OWL:

```
<owl:ObjectProperty rdf:ID="causes">
     ...
    <TreeNumber>R3.2.2</TreeNumber>
    <Abbreviation>CA</Abbreviation>
     ...
</owl:ObjectProperty>
```

As stated above, individuals and properties are disjoint domains in OWL DL. When the attributes represented like in above example the property will be used as both a property and an individual at the same time. Therefore, the mentioned descriptive information is not able to be transformed into OWL DL Species due to limitations of the OWL DL standard.

3.3.1.1.3 OWL Full Species

The root element and related namespaces in each generated OWL Full Species are characterized as following:

```
<rdf:RDF

     xmlns:owl="http://www.w3.org/2002/07/owl#"
     xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
     xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
     xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
     xml:base="http://birsengozdemir.com/SN_OWL_Full.xml#"
     xmlns="http://birsengozdemir.com/SN_OWL_Full.xml">
     xmlns:rsfld="http://semanticnetwork.nlm.nih.gov/Download
     /RelationalFiles/SRFLD#">

     ...

</rdf:RDF>
```

3.3.1.1.3.1 Semantic Type and Relation Transformation

Semantic types (STYs) and semantic relations (RLs) in UMLS SN are represented in OWL Full Species by generated algorithms using the same principles with OWL Lite transformation which are explained above. (See section 3.3.1.1.1)

In NetON, biomedical categories (STYs) and their hierarchy are also represented as named classes (`owl:Class`) and subclass axioms (`rdfs:subClassOf)` in OWL Full Species, respectively. In UMLS SN files [44], STYs are not stated as being mutually exclusive. Therefore, any statement with owl:disjointWith axiom is not used for described classes in OWL Full Species as well.

In NetON, an UMLS SN relation is transformed to OWL object property (owl:ObjectProperty) and the RL hierarchy is represented with subproperty axiom (`rdfs:subPropertyOf`) in OWL Full Species.

In NetON, the applicable constructor `owl:inverseOf` is used in OWL Full Species by the algorithms to transform inverse relations in UMLS SN. Let's give an example:

In SRDEF file [44]:

STY|T195|Antibiotic|A1.4.1.1.1.1|A pharmacologically active compound produced by growing microorganisms which kill or inhibit growth of other microorganisms.|Antibiotics; Cephalosporins; Methicillin|||antb||

RL|T151|affects|R3.1|Produces a direct effect on. Implied here is the altering or influencing of an existing condition, state, situation, or entity. This includes has a role in, alters, influences, predisposes, catalyzes, stimulates, regulates, depresses, impedes, enhances, contributes to, leads to, and modifies.||||AF|affected_by|

In SRSTR file [44]:

Antibiotic|isa|Pharmacologic Substance|D|

affects|isa|functionally_related_to|D|

In an NetON OWL Full document:

```
<owl:Class rdf:ID="Antibiotic"  >
    <rdfs:subClassOf >
      <owl:Class rdf:about="#Pharmacologic_Substance" />
    </rdfs:subClassOf>
</owl:Class>

<owl:ObjectProperty rdf:ID="affects">
   <rdfs:subPropertyOf rdf:resource="#functionally_related_to"/>
    <owl:inverseOf rdf:resource="#affected_by" />
  </owl:ObjectProperty>
```

3.3.1.1.3.2 Assertion Transformation

Relations between STYs (i.e. assertions) in UMLS SN can be represented as property axioms including rdfs:range and rdfs:domain constructors. Multiple domain or range axioms for any property ought to be understood as intersection of all domain or range values. The below property description in OWL limits the domain of ‘evaluation_of’ property to instances that are members of the conjunction of ‘Finding’ and ‘Qualitative Concept’ classes. Nevertheless, evaluation_of (Finding, Organism Attribute), evaluation_of (Qualitative Concept, Activity) assertions in UMLS SN, for example, implies that domain of the ‘evaluation_of’ property can be either a ‘Finding’ or a ‘Qualitative Concept’.

In SRDEF file [44]:

RL|T161|evaluation_of|R5.1|Judgment of the value or degree of some attribute or process.||||EV|has_evaluation|

In SRSTR file (all assertions including ‘evaluation_of’):

Finding|evaluation_of|Biologic Function|D|

Finding|evaluation_of|Organism Attribute|D|

Qualitative Concept|evaluation_of|Activity|D|

In OWL:

```
<owl:ObjectProperty rdf:ID="evaluation_of">
    ...
    <rdfs:domain rdf:resource="#Finding"/>
    <rdfs:domain rdf:resource="#Qualitative Concept"/>
    <rdfs:range rdf:resource="#Biologic Function"/>
    <rdfs:range rdf:resource="#Organism Attribute"/>
    <rdfs:range rdf:resource="#Activity"/>
</owl:ObjectProperty>
```

Multiple classes acting as the domain or range of a property can be described by the help of owl:unionOf constructor. This constructor is valid in OWL Full ontologies:

```
<owl:ObjectProperty rdf:ID="evaluation_of">
    ...
    <rdfs:domain>
     <owl:Class>
         <owl:unionOf rdf:parseType="Collection">
             <owl:Class rdf:about="#Finding"/>
             <owl:Class rdf:about="#Qualitative Concept"/>
          </owl:unionOf>
       </owl:Class>
    <rdfs:domain>
    <rdfs:range>
       <owl:Class>
         <owl:unionOf rdf:parseType="Collection">
             <owl:Class rdf:about="#Biologic Function"/>
             <owl:Class rdf:about="#Organism Attribute"/>
             <owl:Class rdf:about="#Activity"/>
          </owl:unionOf>
       </owl:Class>
    <rdfs:range>
</owl:ObjectProperty>
```

Even the owl:unionOf constructor is used in order to declare that multiple classes can act as the domain or range of the 'evaluation_of' property, it will not sufficiently represent the assertions in UMLS SN having meaning that particular range constraints are only valid with particular domain constraints. Ignoring this and using simply the union of categories/classes in domain and range constraints will lead to misrepresented models. The following table shows the domain and range of 'evaluation_of' relation and valid and invalid assertions between them. (Table 12)

Table 12 The range and domain of ‘evaluation_of’ relation and valid assertions between them

‘evaluation_of’ relation		**DOMAIN**	
		Finding	**Qualitative Concept**
RANGE	**Biologic Function**	√	-
	Organism Attribute	√	-
	Activity	-	√

The associative relations among STYs seem also to be denoted through axioms involving the rdfs:subClassOf, owl:Restriction, owl:onProperty, owl:someValuesFrom, constructors. A particular property can be linked to owl:Restriction class by using the owl:onProperty property. A particular type of anonymous class description with property constraints can be used as value restrictions on the range of the property. [15] For example, evaluation_of (Finding, Organism Attribute) and evaluation_of (Finding, Biologic Function) assertions can be stated as:

```
<owl:Class rdf:ID="Finding" >
    ...
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty rdf:resource="#evaluation_of" />
        <owl:someValuesFrom>
          <owl:Class rdf:about="#Organism Attribute" />
        </owl:someValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty rdf:resource="#evaluation_of" />
        <owl:someValuesFrom>
          <owl:Class rdf:about="#Biologic Function"/>
        </owl:someValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
</owl:Class>
```

However, deeper examination revealed that the semantic meaning of above representation and the UMLS SN intended meaning were different. According to the above representation with owl:someValuesFrom constructor, an individual of ‘Finding’ class should have at least one value of ‘evaluation_of’ relation to be an individual of ‘Organism Attribute’ and should also have at least one value of ‘evaluation_of’ relation to be an individual of ‘Biologic Function’, simultaneously. This represented semantic is not actually intended by UMLS SN knowledge. The owl:unionOf can be added to the above mentioned constructors in order to represent assertions. In that occasion, an individual of ‘Finding’ class should have at least one value of ‘evaluation_of’ relation to be an individual of conjunction of ‘Organism Attribute’ and ‘Biologic Function’ classes. Even this represented semantic can specify UMLS SN knowledge, it doesn’t exclude existence of other individuals for evaluation_of (Finding, *c*) assertion where *c* doesn’t belong to either ‘Organism Attribute’, ‘Biologic Function’ or ‘Activity’ classes. Therefore, above declared representations are not used to transform assertions in UMLS SN into OWL Full Species.

The assertions seem also to be denoted through most appropriate axioms involving the rdfs:subClassOf, owl:Restriction, owl:onProperty, owl:allValuesFrom constructors. Nonetheless, additional inspection exposed that the UMLS SN intended meaning was different. According to the depiction with owl:allValuesFrom constructor, an individual of ‘Finding’ class should have all values of ‘evaluation_of’ relation to be an individual of intersection of ‘Organism Attribute’ , ‘Biologic Function’ and ‘Activity’ classes. Since ‘Organism Attribute’, ‘Biologic Function’ and

‘Activity’ classes are semantically different categories/classes, the mentioned depiction is actually inconsistent.

The owl:unionOf can be added to the above mentioned constructors in order to represent assertions. In that occasion, an individual of ‘Finding’ class should have all values of ‘evaluation_of’ relation to be an individual of conjunction of ‘Organism Attribute’ ‘Biologic Function’ and ‘Activity’ classes. This representation seem to be the best one for specification of assertions, it leaves out existence of other individuals for evaluation_of (Finding, *c*) assertion where *c* doesn’t belong to either ‘Organism Attribute’, or ‘Biologic Function’ and ‘Activity’ classes. As a result of above mentioned analysis, the last representation is used to transform assertions in UMLS SN into OWL Full Species:

```
<owl:Class rdf:ID="Finding" >
    ...
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty rdf:resource="#evaluation_of" />
        <owl:allValuesFrom>
         <owl:Class>
           <owl:unionOf rdf:parseType="Collection">
             <owl:Class rdf:about="#Organism Attribute"/>
             <owl:Class rdf:about="#Biologic Function"/>
             <owl:Class rdf:about="#Activity"/>
            </owl:unionOf>
         </owl:Class>
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
</owl:Class>
```

3.3.1.1.3.3 Descriptive Information Transformation

The descriptive information about the STYs in UMLS SN (Table 13) can be identified as attributes of each category/class in OWL Full. In this representation the concerned class will be both a class and an individual at the same time. Since, individuals and classes are not disjoint domains in OWL Full, the descriptive information is transformed into OWL Full Species in this format by developed algorithms in NetON. Given that semantic types in UMLS SN are either Event or Entity, this is also taken into account by NetON:

```
<owl:Class rdf:ID="Antibiotic">
    ...
   <TreeNumber>A1.4.1.1.1.1</TreeNumber>
   <Examples>Antibiotics; Cephalosporins; Methicillin</Examples>
   <Abbreviation>antb</Abbreviation>
   <MainCategory>Entity</MainCategory>
    ...
</owl:Class>
```

Table 13 Descriptive information of STYs in UMLS SN

Descriptive information of STYs in UMLS SN	
Abbreviation	**Description**
RT	record type
UI	unique identifier
STY	semantic type name
STN	semantic type tree number
DEF	definition
EX	examples
UN	usage note
NH	non-human flag
ABR	abbreviation

The descriptive information about the RLs in UMLS SN (Table 14) can also be identified as attributes of each relation/property in OWL Full. In this representation the interested property will be used as both a property and an individual at the same time. In view of the fact that individuals and properties are not disjoint domains in OWL Full, the mentioned descriptive information is transformed into OWL Full Species in this format by NetON:

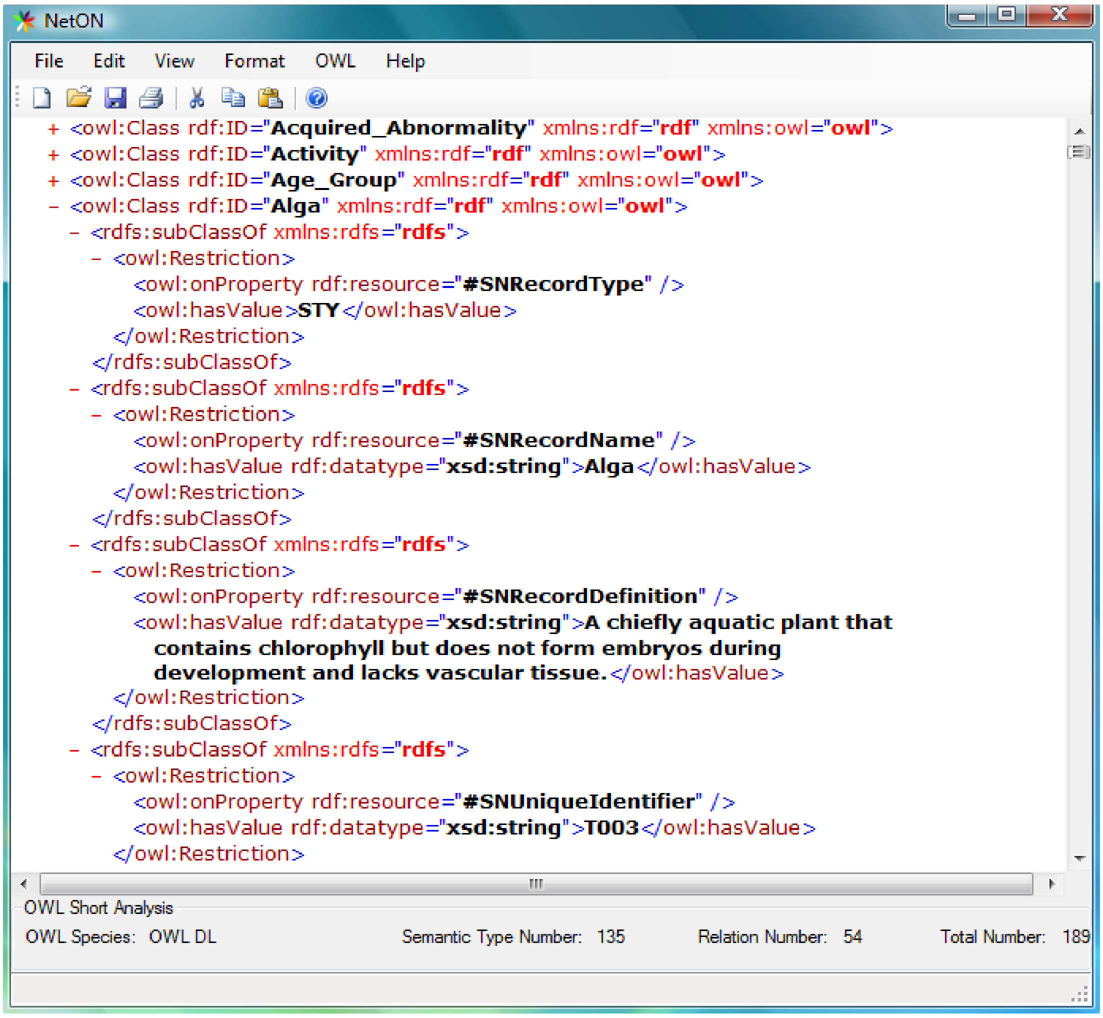
```
<owl:ObjectProperty rdf:ID="affects">
     ...
    <TreeNumber>R3.1</TreeNumber>
    <Abbreviation>AF</Abbreviation>
     ...
</owl:ObjectProperty>
```

Table 14 Descriptive information of RLs in UMLS SN

Descriptive information of RLs in UMLS SN	
Abbreviation	**Description**
RT	record type
UI	unique identifier
RL	relation name
RTN	relation tree number
DEF	definition
EX	examples
UN	usage note
NH	non-human flag
ABR	abbreviation
RIN	relation inverse

3.3.1.2 Validation of OWL Sublanguage Species

The OWL Short Analysis box of NetON is used in order to validate the transformation of OWL Full Species. The name of generated OWL Sublanguage Species appears at the left side of this box. While transforming the UMLS SN files [44] to OWL Full Species the developed algorithms check correspondingly utilized files for the semantic types (STYs) and relations (RLs) numbers. Those numbers with the total number emerged in the OWL Short Analysis box allowing users to control whether the used files are the correct ones (original files) including expected number of STYs and RLs (Figure 13). The short analysis information in the box appears after selection of a related menu option to generate OWL Sublanguage documents:



```
+ <owl:Class rdf:ID="Acquired_Abnormality" xmlns:rdf="rdf" xmlns:owl="owl">
+ <owl:Class rdf:ID="Activity" xmlns:rdf="rdf" xmlns:owl="owl">
+ <owl:Class rdf:ID="Age_Group" xmlns:rdf="rdf" xmlns:owl="owl">
- <owl:Class rdf:ID="Alga" xmlns:rdf="rdf" xmlns:owl="owl">
  - <rdfs:subClassOf xmlns:rdfs="rdfs">
    - <owl:Restriction>
        <owl:onProperty rdf:resource="#SNRecordType" />
        <owl:hasValue>STY</owl:hasValue>
      </owl:Restriction>
    </rdfs:subClassOf>
  - <rdfs:subClassOf xmlns:rdfs="rdfs">
    - <owl:Restriction>
        <owl:onProperty rdf:resource="#SNRecordName" />
        <owl:hasValue rdf:datatype="xsd:string">Alga</owl:hasValue>
      </owl:Restriction>
    </rdfs:subClassOf>
  - <rdfs:subClassOf xmlns:rdfs="rdfs">
    - <owl:Restriction>
        <owl:onProperty rdf:resource="#SNRecordDefinition" />
        <owl:hasValue rdf:datatype="xsd:string">A chiefly aquatic plant that
          contains chlorophyll but does not form embryos during
          development and lacks vascular tissue.</owl:hasValue>
      </owl:Restriction>
    </rdfs:subClassOf>
  - <rdfs:subClassOf xmlns:rdfs="rdfs">
    - <owl:Restriction>
        <owl:onProperty rdf:resource="#SNUniqueIdentifier" />
        <owl:hasValue rdf:datatype="xsd:string">T003</owl:hasValue>
      </owl:Restriction>
```

OWL Short Analysis

OWL Species: OWL DL   Semantic Type Number: 135   Relation Number: 54   Total Number: 189

Figure 13 Web browser view of OWL DL Species

The OWL Sublanguage Species documents generated by NetON were controlled with WonderWeb [25], which is a European Union Project. WonderWeb includes algorithms for OWL ontology validation. It checks usage of constructors in an OWL ontology with rules of OWL sublanguage (Full, DL, Lite) standards specified by W3C'sWeb Ontology Working Group [15]. Besides returning a depiction of individuals, classes and properties in controlled ontology in terms of the OWL Abstract Syntax, any constructs being connected with specific sublanguages of OWL are also reported by WonderWeb. The validation of the OWL Full, OWL DL and OWL Lite Species generated by NetON with WonderWeb showed that the controlled OWL Sublanguage Species complied with the rules of OWL sublanguage standards. It was seen that OWL Full Species permitted joining of entire language constructors of OWL in arbitrary manner with RDF or RDFS; OWL DL Species prohibited the usage of OWL constructs to one another; and OWL Lite Species further restricted the usage of OWL constructors. (Figure 14)

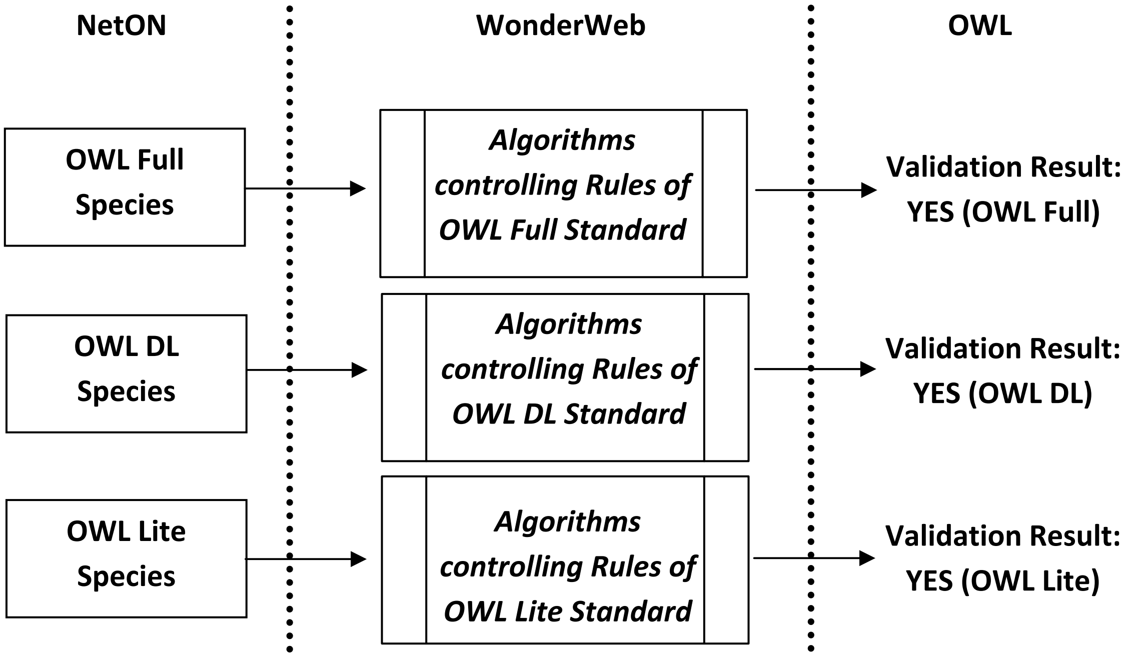
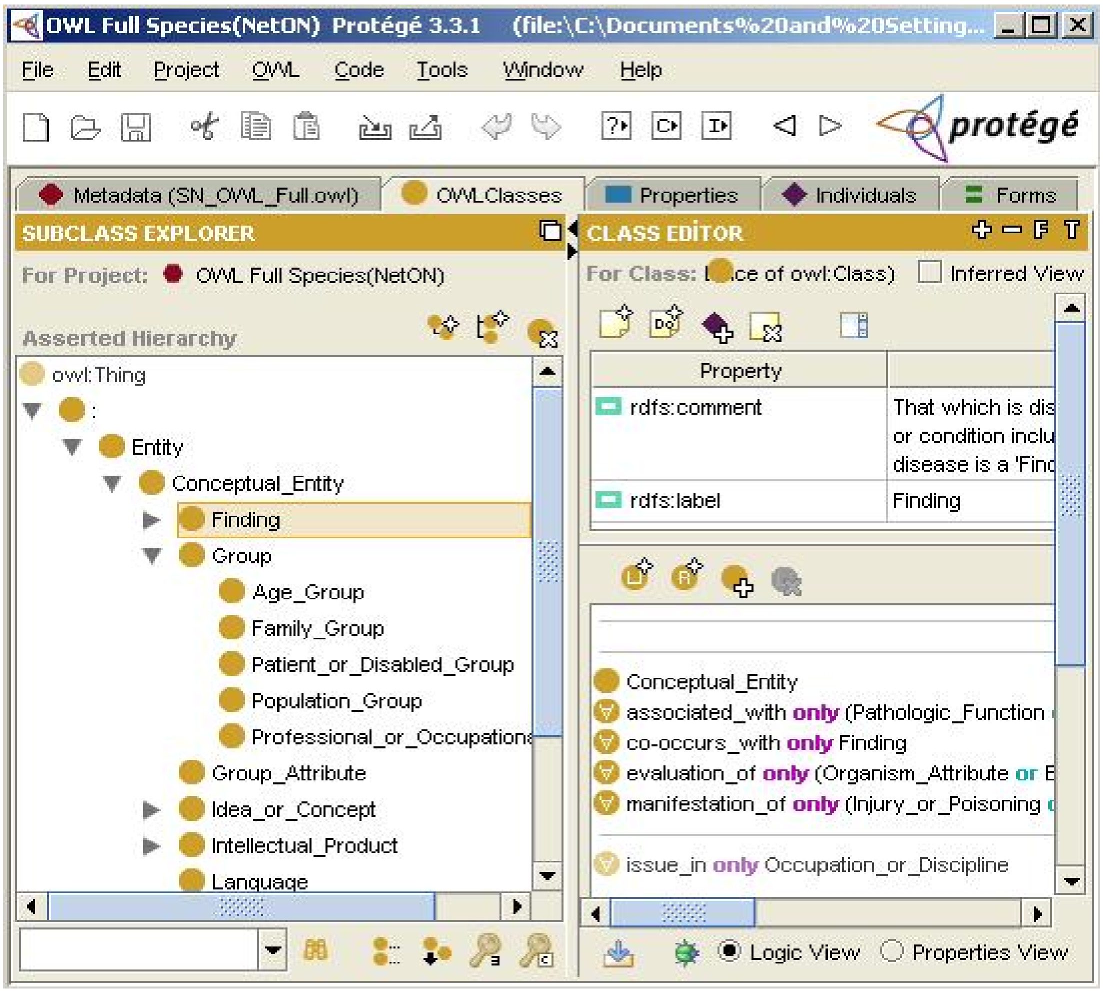

NetON
WonderWeb
OWL
OWL Full Species
Algorithms controlling Rules of OWL Full Standard
Validation Result: YES (OWL Full)
OWL DL Species
Algorithms controlling Rules of OWL DL Standard
Validation Result: YES (OWL DL)
OWL Lite Species
Algorithms controlling Rules of OWL Lite Standard
Validation Result: YES (OWL Lite)

Figure 14 The validation of OWL Sublanguage Species with WonderWeb

The OWL Sublanguage Species generated by NetON was also loaded to Protégé [26], an open source ontology editor. This was done with the intention of validation for the usage opportunity of generated OWL Sublanguage Species in Protégé OWL Editor. Figure 15 shows the imported OWL Full Species in Protégé:



Property	
rdfs:comment	That which is dis or condition inclu disease is a 'Finc
rdfs:label	Finding

Conceptual_Entity
associated_with only (Pathologic_Function
co-occurs_with only Finding
evaluation_of only (Organism_Attribute or E
manifestation_of only (Injury_or_Poisoning
issue_in only Occupation_or_Discipline

● Logic View ○ Properties View

Figure 15 Protégé OWL Full Species import

3.3.1.3 Inheritance Blocking

In UMLS SN, there are unseen assertions that can be inferred by using inference rules on explicitly specified assertions. However, explicitly declared relations between semantic types are not essentially valid for all the descendants. Deduction outcomes of any OWL reasoners (such as Pellet or Racer [42, 43]) on NetON OWL Sublanguage Species (except OWL Lite) will also include false positives due to the lack of inheritance blocking information on assertions in NetON OWL DL and OWL Full Species. Knowing that 'Diagnostic Procedure' has no child 'Intellectual Product' has two children and conceptual_part_of (Intellectual Product, Diagnostic Procedure) assertion stated as Descendants Not Included (DNI) in the SRSTR file, the following example can be seen:

In SRDEF file:

RL|T160|conceptual_part_of|R5.11|Conceptually a portion, division, or component of some larger whole.||||CP|has_conceptual_part|

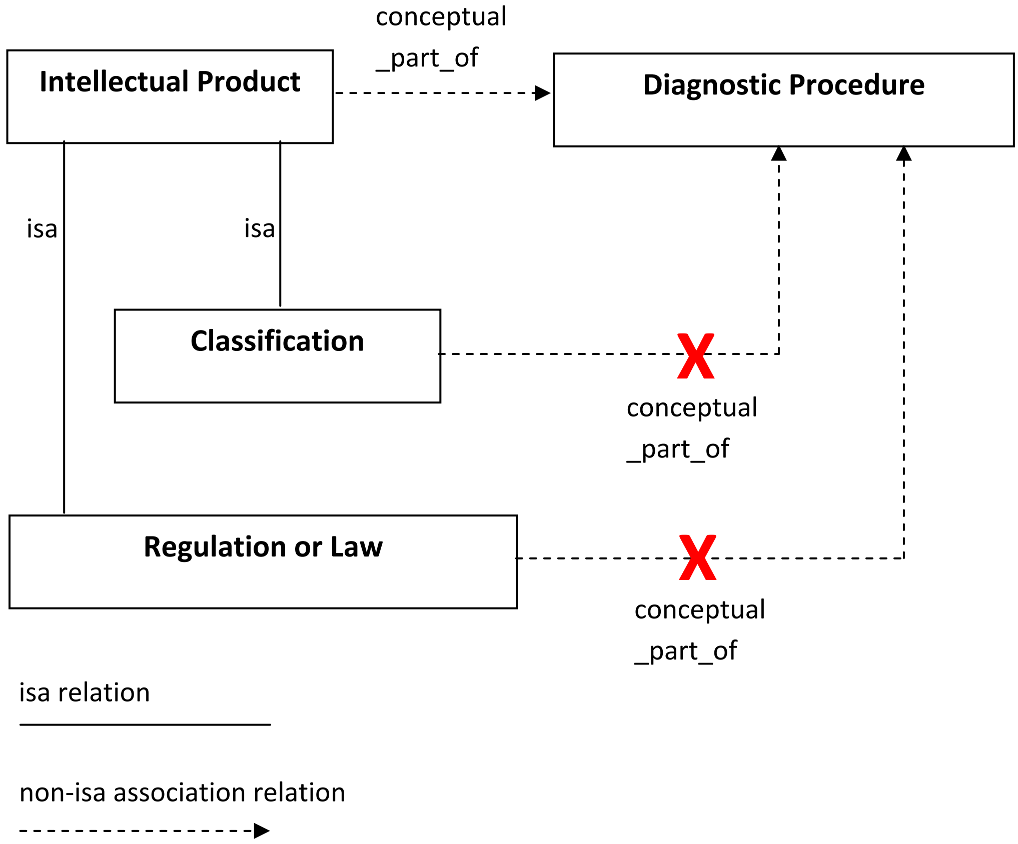
In SRSTR file:

Intellectual Product|conceptual_part_of|Diagnostic Procedure|DNI|

Regulation or Law|isa|Intellectual Product|D|

Classification|isa|Intellectual Product|D|

Graphical Representation of the assertion with subject descendants:



The Descendants Not Included (DNI) notation (like other inheritance blocking notations) is not able to be represented in OWL Sublanguage Species due to the limitations of OWL Sublanguage standards. When an OWL reasoner [42, 43] reads the conceptual_part_of (Intellectual Product, Diagnostic Procedure) assertion from OWL DL Species, for example, it will apply the basic inference rules (inheritance) to the assertion in order to find the hidden conceptual_part_of (Classification, Diagnostic Procedure) and conceptual_part_of (Regulation or Law, Diagnostic Procedure) assertions. The OWL reasoner would state that those two inferred assertions are valid ones. However, conceptual_part_of (Classification, Diagnostic Procedure) and conceptual_part_of (Regulation or Law, Diagnostic Procedure) assertions are not valid due to the declaration of conceptual_part_of (Intellectual Product, Diagnostic Procedure) assertion with Descendants Not Included (DNI) notation, which has the meaning that descendants of ‘Intellectual Product’ and ‘Diagnostic Procedure’ should not inherit the ‘conceptual_part_of’ relation. Therefore, conceptual_part_of (Classification, Diagnostic Procedure) and conceptual_part_of (Regulation or Law, Diagnostic Procedure) assertions which can be found by any OWL reasoner are **false positive** assertions. (See section 3.3.2.1 for Blocked (B) example).

There are 10 Blocked (B) and 27 Descendant Not Included (DNI) assertions defined in SRSTR file, which includes information on the structure of UMLS SN. Table 15 shows the possible number for **false positive** results that comes from Blocked (B) and Descendant Not Included (DNI) assertions, respectively.

Table 15 Possible number for false positive results that comes from Blocked(B) and Descendant Not Included (DNI) assertions

False Positive	**Top Assertion Number**	**Descendant Assertion Number**
Blocked(B)	10	2
Descendant Not Included (DNI)	-	78

The algorithms of the second dimension consider the inheritance blocking information Blocked (B), Descendant Not Included (DNI) and Defined (D) while executing inference rules. As this cannot be done by any OWL reasoner, the second dimension offers a solution for application developers.

3.3.2 Second Dimension for OWL Species

The relations between STYs can be called as assertions as stated before. In UMLS Semantic Network, there are concealed assertions that can be revealed by using inference rules on explicitly specified assertions. For example, declared relations between STYs are usually inherited by all the children of those STYs. However, some of them are not entirely valid for all the descendants of those STYs. This information is stated in the SRSTR file including the structure of UMLS SN as inheritance blocking. UMLS SN specifies three different inheritance blockings: B (Blocked), DNI (Descendants not included) and D (Descendants included).

One of the limitations of NetON OWL Sublanguage Species described in the previous section is that they are not able to define the inheritance blocking B, DNI and D due to the expressiveness limitations of OWL sublanguage standards. In order to overcome the limitations and help biomedical application developers the NetON offers OWL Species that have also second dimension. Thinking OWL Sublanguage Species as first dimension, ‘Extended’ is notified as second dimension in generation of OWL documents. Second dimension is applicable for OWL Lite, OWL DL, and OWL Full Species. Since assertions are not able to be transformed to OWL Lite Species due to the limitations of OWL Lite sublanguage standard, main contribution of Second dimension can be seen with especially OWL DL and OWL Full Species. (Table 16) The relevant menu options that run the respectively developed algorithms may be chosen to generate the corresponding OWL Extended Species.

Deduction outcomes of any OWL reasoners (such as Pellet or Racer [42, 43]) on NetON OWL Sublanguage Species will also include false positives (See section 3.3.1.3) due to the lack of inheritance blocking information in NetON OWL

Sublanguage Species. With the second dimension, algorithms including inference rules are executed on top of algorithms for generation of OWL Sublanguage Species. OWL Full-Extended Species, for instance, may be generated after carrying out the related algorithms including OWL Full and Extended restrictions, respectively. The algorithms of the second dimension consider the inheritance blocking information B, DNI and D while executing inference rules. As this cannot be done by any OWL reasoner, the second dimension offers a solution for application developers who need machine processable form of UMLS SN information including whole concealed and valid information. (Figure 16)

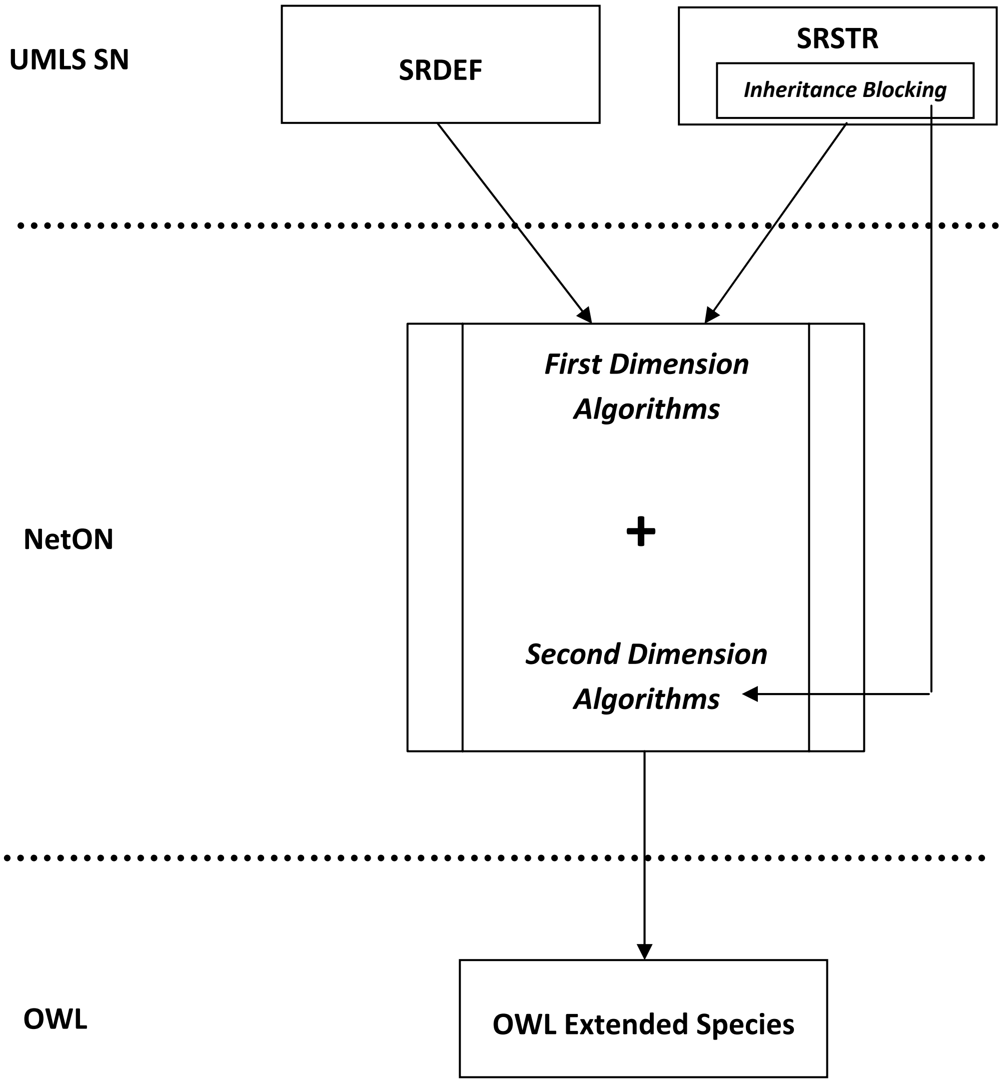


Figure 16 Transformation of the UMLS SN to OWL Extended Species

Table 16 Second dimension offered by NetON

Second Dimension	**With Inherited Assertions by Descendant(s) of STYs**	**With Inherited Assertions by Descendant(s) of RLs**
Extended	√	√

An OWL Extended document is generated by a corresponding algorithm to include concealed associative relations among STYs by considering the inheritance blocking information B, DNI and D while executing inference rules. (Figure 17) The following sections explain details of second dimension algorithms.

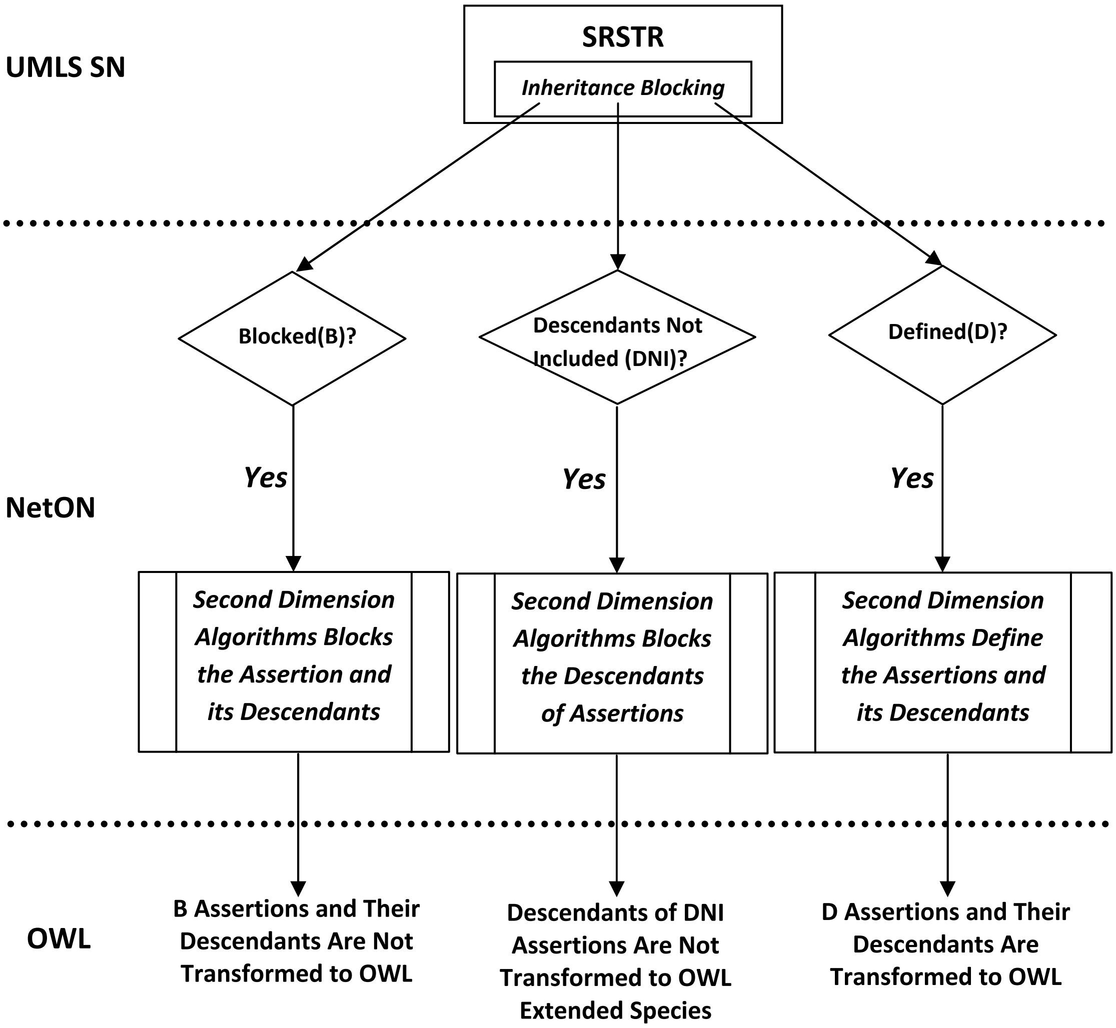

UMLS SN
SRSTR
Inheritance Blocking
Blocked(B)?
Descendants Not Included (DNI)?
Defined(D)?
Yes
Yes
Yes
NetON
Second Dimension Algorithms Blocks the Assertion and its Descendants
Second Dimension Algorithms Blocks the Descendants of Assertions
Second Dimension Algorithms Define the Assertions and its Descendants
OWL
B Assertions and Their Descendants Are Not Transformed to OWL
Descendants of DNI Assertions Are Not Transformed to OWL Extended Species
D Assertions and Their Descendants Are Transformed to OWL

Figure 17 Inheritance blocking control by NetON second dimension algorithms

3.3.2.1 Blocked (B) Control in Second Dimension

The algorithms that generate an Extended Type documents consider the blocked notation (B) in UMLS SN. This information is stated as B in the SRSTR file for explicit relations between STYs. B means "Blocked" and indicates relations that are not valid between specific STYs. Knowing that 'Alga' is a 'Plant' and process_of (Mental Process, Plant) assertion is stated as B in the SRSTR file, the following example can be seen:

In SRDEF file:

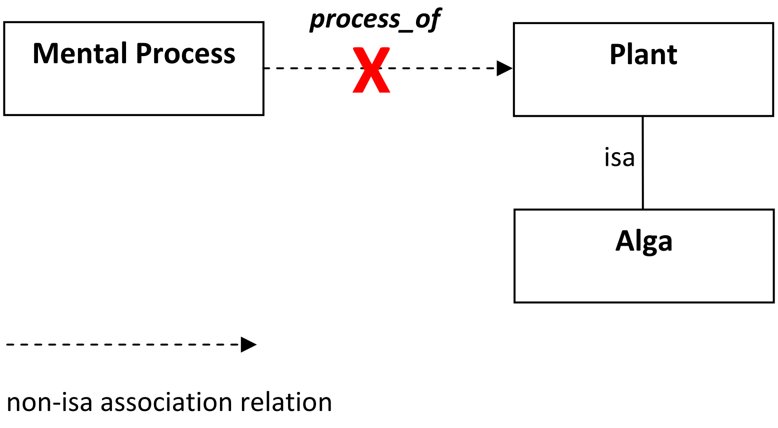
RL|T140|**process_of**|R3.4.1|Action, function, or state of.||||PO|has_process|

In SRSTR file:

Mental Process|**process_of**|Plant|**B**|

Alga|isa|Plant|D|

Graphical Representation of the assertion:



In a NetON OWL Extended document:

As process_of (Mental Process, Plant) assertion is stated as "B" in the SRSTR file, it is not transformed into OWL Extended Type documents generated by NetON. The following example shows the description of "Mental Process" class in both OWL Full-Extended (WOD) and OWL DL- Extended (WOD) documents. It can be seen that process_of (Mental Process, Plant) assertion was not included in Mental_Process class declaration. NETON related algorithms find the subclass(es) of 'Plant' not to include them with object part of process_of relation. The algorithms also find the subclass(es) of 'Mental Process' not to include them with subject part of process_of relation. As 'Mental Process' has no child and 'Alga' is the subclass of 'Plant', process_of (Mental Process, Alga) assertion was not included in Mental_Process class declaration (The entire description of 'Mental Process' semantic type can be seen at Appendix G):

Mental_Process class definition:

```
<owl:Class rdf:ID="Mental_Process" >
     ...
  <rdfs:subClassOf >
    <owl:Restriction>
      <owl:onProperty rdf:resource="#process_of" />
      <owl:allValuesFrom>
        <owl:Class>
          <owl:unionOf rdf:parseType="Collection">
            <owl:Class rdf:about="#Mental_Process" />
            <owl:Class rdf:about="#Organism_Function" />
            <owl:Class rdf:about="#Organ_or_Tissue_Function"/>
            <owl:Class rdf:about="#Genetic_Function" />
            <owl:Class rdf:about="#Molecular_Function" />
            <owl:Class rdf:about="#Cell_Function" />
            <owl:Class rdf:about="#Physiologic_Function" />
            <owl:Class rdf:about="#Experimental_Model_of_Disease"/>
            <owl:Class rdf:about="#Neoplastic_Process"/>
            <owl:Class rdf:about="#Mental_or_Behavioral_Dysfunction"/>
            <owl:Class rdf:about="#Disease_or_Syndrome"/>
            <owl:Class rdf:about="#Cell_or_Molecular_Dysfunction"/>
            <owl:Class rdf:about="#Pathologic_Function" />
            <owl:Class rdf:about="#Biologic_Function" />
            <owl:Class rdf:about="#Natural_Phenomenon_or_Process"/>
            <owl:Class rdf:about="#Archaeon" />
            <owl:Class rdf:about="#Reptile" />
            <owl:Class rdf:about="#Human" />
            <owl:Class rdf:about="#Mammal" />
            <owl:Class rdf:about="#Fish" />
            <owl:Class rdf:about="#Bird" />
            <owl:Class rdf:about="#Amphibian" />
            <owl:Class rdf:about="#Vertebrate" />
            <owl:Class rdf:about="#Invertebrate" />
            <owl:Class rdf:about="#Animal" />
            <owl:Class rdf:about="#Organism" />
          </owl:unionOf>
        </owl:Class>
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
      ...
</owl:Class>
```

3.3.2.2 Descendants Not Included (DNI) Control in Second Dimension

Even though explicitly declared relations between STYs (assertions in the SRSTR file) are usually inherited by all the children of those STYs, some of them are not entirely valid for all the descendants. This information is stated in the SRSTR file as inheritance blocking DNI which means defined but not inherited by the children of the STYs. The responsible algorithms take this information into consideration while generating OWL Extended Type documents. Knowing that ‘Disease or Syndrome’ has two children and conceptually_related_to (Disease or Syndrome, Experimental Model of Disease) assertion stated as DNI in the SRSTR file, the following example can be seen:

In SRDEF file:

RL|T158|conceptually_related_to|R5|Related by some abstract concept, thought, or idea.||||CR|conceptually_related_to|

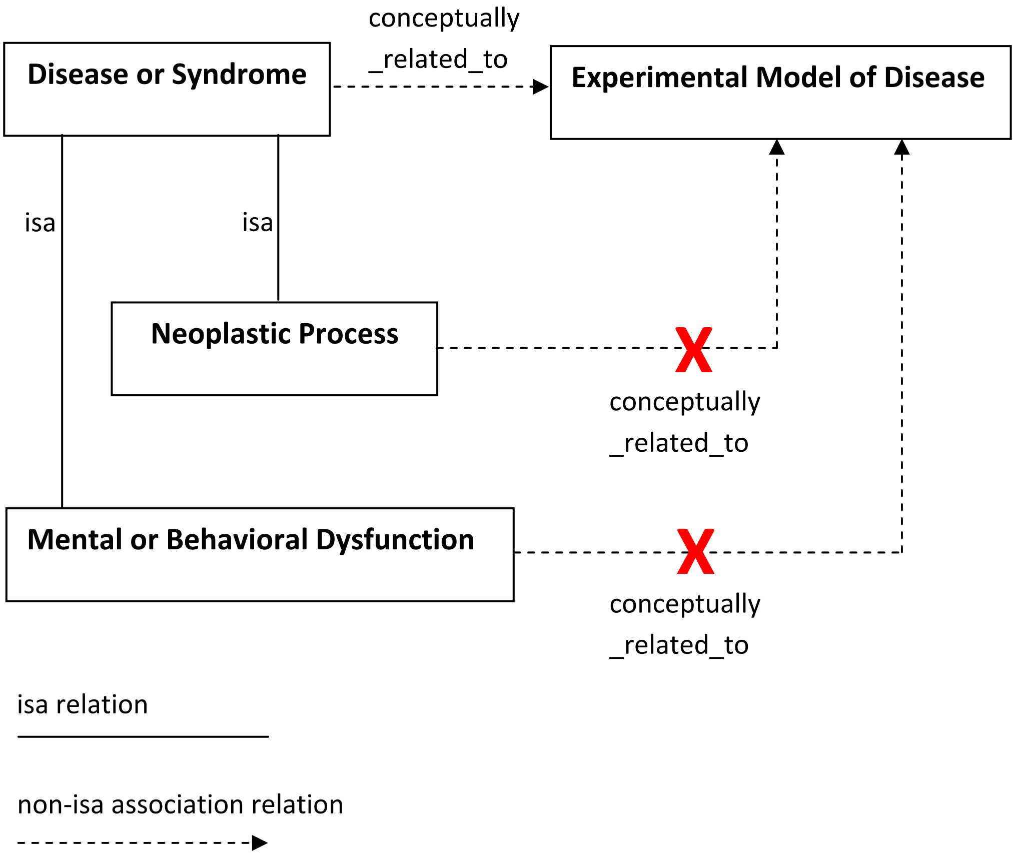
In SRSTR file:

Neoplastic Process|isa|Disease or Syndrome|D|

Mental or Behavioral Dysfunction|isa|Disease or Syndrome|D|

Disease or Syndrome|conceptually_related_to|Experimental Model of Disease|DNI|

Graphical Representation of the assertion with subject descendants:



In a NetON OWL Extended document:

As conceptually_related_to (Disease or Syndrome, Experimental Model of Disease) assertion is stated as "DNI" in the SRSTR file, the children (if there is any) of STYs are controlled by the related algorithms and the relation for those children are not transformed into OWL Extended Type documents generated by NetON. The algorithms find the subclass(es) of 'Disease or Syndrome' not to include them with subject part of process_of relation. The algorithms also find the subclass(es) of 'Experimental Model of Disease' not to include them with object part of process_of relation. 'Experimental Model of Disease' has no child and 'Disease or Syndrome' has two children.

The following example shows the descriptions of "Disease or Syndrome", "Neoplastic Process" and "Mental or Behavioral Dysfunction" classes in both OWL Full-Extended (WOD) and OWL DL-Extended (WOD) documents. It can be seen that conceptually_related_to (Disease or Syndrome, Experimental Model of Disease) assertion is included in 'Disease or Syndrome' class declaration, however, conceptually_related_to (Neoplastic Process, Experimental Model of Disease) and conceptually_related_to (Mental or Behavioral Dysfunction, Experimental Model of Disease) assertions are not taken in 'Neoplastic Process and Mental' or 'Behavioral Dysfunction' class declarations (The entire descriptions of semantic types can be seen at Appendix G):

Disease_or_Syndrome class definition:

```
<owl:Class rdf:ID="Disease_or_Syndrome"  >
     ...
    <rdfs:subClassOf >
      <owl:Restriction>
        <owl:onProperty rdf:resource="#conceptually_related_to"/>
        <owl:allValuesFrom>
          <owl:Class rdf:about="#Experimental_Model_of_Disease"/>
        </owl:allValuesFrom>
      </owl:Restriction>
     </rdfs:subClassOf>
       ...
</owl:Class>
```

Neoplastic_Process class definition:

```
<owl:Class rdf:ID="Neoplastic_Process"    >
       ...
       //There is no 'conceptually_related_to' property in the description.
       // The entire description of the STY can be seen at Appendix G.
       ...
</owl:Class>
```

Mental_or_Behavioral_Dysfunction class definition:

```
<owl:Class rdf:ID="Mental_or_Behavioral_Dysfunction"    >
      ...
       //There is no 'conceptually_related_to' property in the description.
       // The entire description of the STY can be seen at Appendix G.
       ...
 </owl:Class>
```

Knowing that 'Organism Function' has one child, 'Temporal Concept' has no child, and conceptual_part_of (Temporal Concept, Organism Function) assertion stated as DNI in the SRSTR file, the following example can be seen:

In SRDEF file:

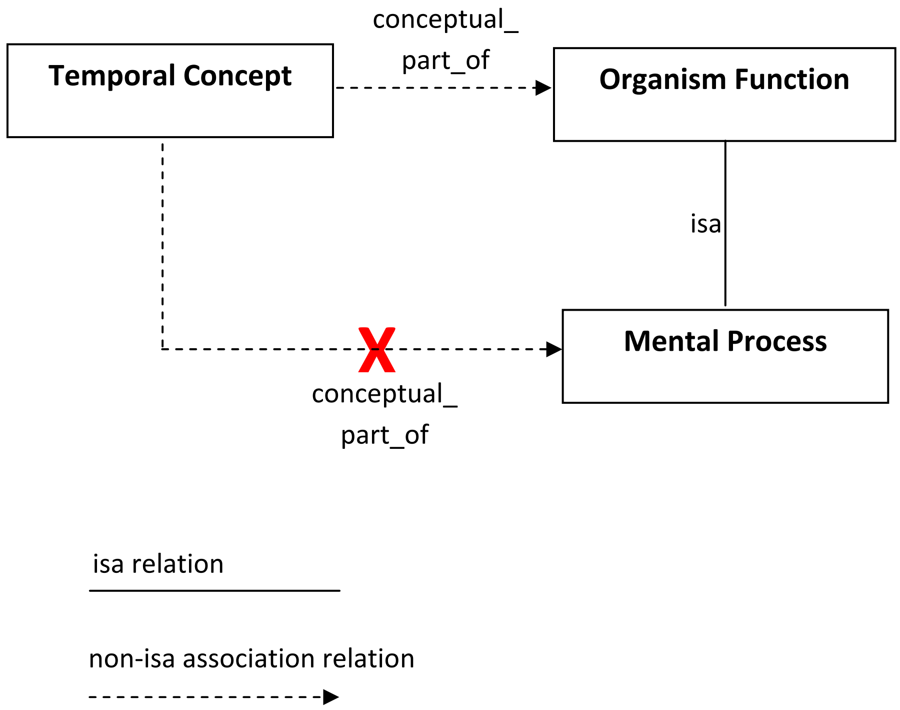
RL|T160|conceptual_part_of|R5.11|Conceptually a portion, division, or component of some larger whole.||||CP|has_conceptual_part|

In SRSTR file:

Mental Process|isa|Organism Function|D|

Temporal Concept|conceptual_part_of|Organism Function|DNI|

<u>Graphical Representation of the assertion with object descendants</u>:



<u>In a NetON OWL Extended document</u>:

As conceptual_part_of (Temporal Concept, Organism Function) assertion is stated as "DNI" in the SRSTR file, the children (if there is any) of STYs are controlled by the related algorithms and the relation for those children are not transformed into OWL Extended Type documents generated by NetON. The algorithms find the subclass(es) of 'Temporal Concept' not to include them with subject of 'conceptual_part_of' relation. The algorithms also find the subclass(es) of 'Organism Function' not to include them with object of 'conceptual_part_of' relation. 'Temporal Concept' has no child and 'Organism Function' has one child.

The following example shows the descriptions of "Temporal Concept" class in both OWL Full-Extended (WOD) and OWL DL-Extended (WOD) documents. It can be seen that conceptual_part_of (Temporal Concept, Organism Function) assertion is included in ‘Temporal Concept’ class declaration, however, conceptual_part_of (Temporal Concept, Mental Process) assertion is not taken in ‘Temporal Concept’ class declaration (The entire description of semantic type can be seen at Appendix G):

Temporal_Concept class definition:

```
<owl:Class rdf:ID="Temporal_Concept" >
    ...
    <rdfs:subClassOf >
      <owl:Restriction>
        <owl:onProperty rdf:resource="#conceptual_part_of" />
        <owl:allValuesFrom>
          <owl:Class>
           <owl:unionOf rdf:parseType="Collection">
            <owl:Class rdf:about="#Temporal_Concept" />
            <owl:Class rdf:about="#Organism_Function" />
            <owl:Class rdf:about="#Organ_or_Tissue_Function"/>
            <owl:Class rdf:about="#Cell_Function" />
           </owl:unionOf>
          </owl:Class>
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
    ...
  </owl:Class>
```

3.3.2.3 Defined (D) Control in Second Dimension

The NetON includes the algorithms that generates an OWL Extended Type document obeying the rules stated in above two sections (Blocked and Descendants

Not Included) and moreover includes relations between descendant(s) of STYs while considering inheritance blocking D in UMLS SN.

The algorithms that generate an Extended Type document consider the inheritance blocking defined (D) for explicitly stated assertions in UMLS SN. D means "Defined for the STYs and its children". This information is stated in the SRSTR file as D. It infers that D stated relations are valid between declared STYs and also this relation inherited by their children. The following example can be seen, knowing that 'Pharmacologic Substance' has one child, 'Pathologic Function' has five descendants and diagnoses (Pharmacologic Substance, Pathologic Function) assertion is stated as D in the SRSTR file:

In SRDEF file:

RL|T163|**diagnoses**|R5.6|Distinguishes or identifies the nature or characteristics of.||||DI|diagnosed_by|

In SRSTR file:

Disease or Syndrome|isa|Pathologic Function|D|

Cell or Molecular Dysfunction|isa|Pathologic Function|D|

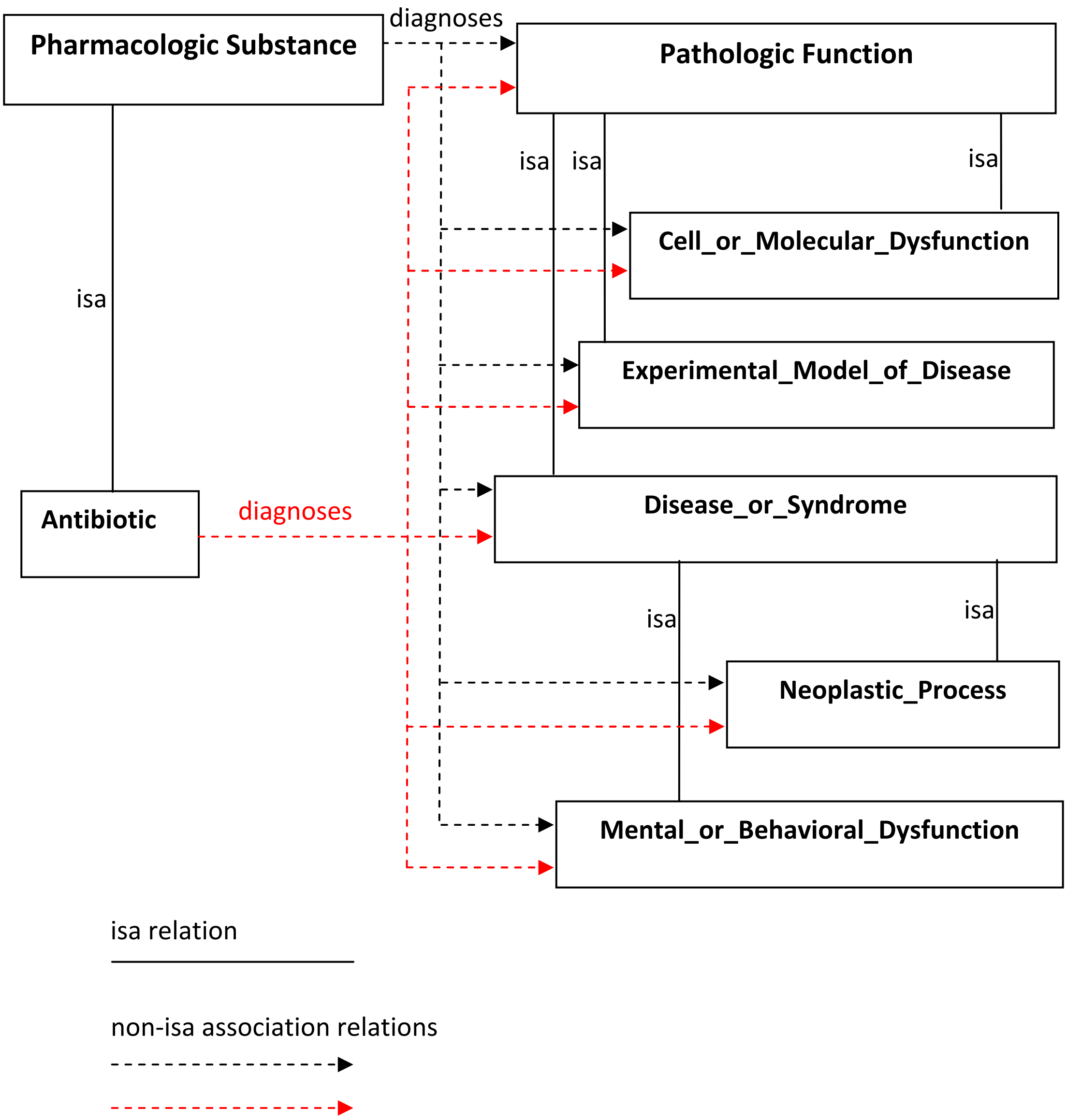
Experimental Model of Disease|isa|Pathologic Function|D|

Neoplastic Process|isa|Disease or Syndrome|D|

Mental or Behavioral Dysfunction|isa|Disease or Syndrome|D|

Pharmacologic Substance|diagnoses|Pathologic Function|D|

Graphical Representation of the assertion with subject descendants:



In a NetON OWL Extended document:

As diagnoses (Pharmacologic Substance, Pathologic Function) assertion is stated as "D" in the SRSTR file, this assertion is also valid for both the children of "Pharmacologic Substance" and the children of "Pathologic Function". The following example shows the descriptions of "Pharmacologic Substance" and "Antibiotic" classes in both OWL Full-Extended (WOD) and OWL DL-Extended (WOD) documents. It can be seen that diagnoses (Pharmacologic Substance, Pathologic

Function), diagnoses (Pharmacologic Substance, Cell_or_Molecular_Dysfunction), diagnoses (Pharmacologic Substance, Experimental_Model_of_Disease), diagnoses (Pharmacologic Substance, Disease_or_Syndrome), diagnoses (Pharmacologic Substance, Neoplastic_Process), and diagnoses (Pharmacologic Substance, Mental_or_Behavioral_Dysfunction) assertions are included in 'Pharmacologic Substance' class declaration. Besides, diagnoses (Antibiotic, Pathologic Function), diagnoses (Antibiotic, Cell_or_Molecular_Dysfunction), diagnoses (Antibiotic, Experimental_Model_of_Disease), diagnoses (Antibiotic, Disease_or_Syndrome), diagnoses (Antibiotic, Neoplastic_Process), and diagnoses (Antibiotic, Mental_or_Behavioral_Dysfunction) assertions are included in 'Antibiotic' class declaration. (The entire descriptions of semantic types can be seen at Appendix G):

***Pharmacologic_Substance class definition*:**

```
<owl:Class rdf:ID="Pharmacologic_Substance"  >
     ...
 <rdfs:subClassOf >
   <owl:Restriction>
    <owl:onProperty rdf:resource="#diagnoses" />
    <owl:allValuesFrom>
      <owl:Class>
        <owl:unionOf rdf:parseType="Collection">
         <owl:Class rdf:about="#Experimental_Model_of_Disease"/>
         <owl:Class rdf:about="#Neoplastic_Process"/>
         <owl:Class rdf:about="#Mental_or_Behavioral_Dysfunction"/>
         <owl:Class rdf:about="#Disease_or_Syndrome"/>
         <owl:Class rdf:about="#Cell_or_Molecular_Dysfunction"/>
         <owl:Class rdf:about="#Pathologic_Function"/>
        </owl:unionOf>
      </owl:Class>
    </owl:allValuesFrom>
   </owl:Restriction>
  </rdfs:subClassOf>
```

```
     ...
</owl:Class>
```

Antibiotic class definition:

```
<owl:Class rdf:ID="Antibiotic">
    ...
  <rdfs:subClassOf >
    <owl:Restriction>
     <owl:onProperty rdf:resource="#diagnoses" />
     <owl:allValuesFrom>
       <owl:Class>
        <owl:unionOf rdf:parseType="Collection">
         <owl:Class rdf:about="#Experimental_Model_of_Disease"/>
         <owl:Class rdf:about="#Neoplastic_Process"/>
         <owl:Class rdf:about="#Mental_or_Behavioral_Dysfunction"/>
         <owl:Class rdf:about="#Disease_or_Syndrome"/>
         <owl:Class rdf:about="#Cell_or_Molecular_Dysfunction"/>
         <owl:Class rdf:about="#Pathologic_Function"/>
        </owl:unionOf>
       </owl:Class>
      </owl:allValuesFrom>
    </owl:Restriction>
  </rdfs:subClassOf>
     ...
</owl:Class>
```

Hierarchical relations of STYs are stated in UMLS SN with inheritance blocking D. NetON algorithms including reasoning rules for generation of OWL Extended Type documents use this hierarchical information to describe entire super classes a STY . The following example shows the super classes of ‘Temporal_Concept’ STY (The entire description of the STY can be seen at Appendix G):

In SRDEF file:

RL|T186|isa|H|The basic hierarchical link in the Network. If one item "isa" another item then the first item is more specific in meaning than the second item.||||IS|inverse_isa|

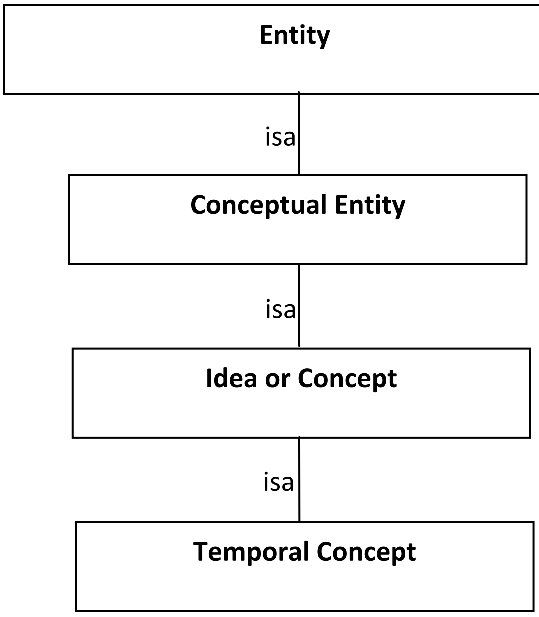
In SRSTR file:

Temporal Concept|isa|Idea or Concept|D|

Idea or Concept|isa|Conceptual Entity|D|

Conceptual Entity|isa|Entity|D|

Graphical Representation of the STYs Hierarchy:



In a NetON OWL Extended document:

As 'isa' relations are stated as D in the SRSTR file, this relation is also valid for descendant semantic types. The following example shows the description of 'Temporal Concept' class in both OWL Full-Extended (WOD) and OWL DL-Extended

(WOD) documents. It can be seen that ‘Temporal Concept’ is an ‘Idea or Concept’. Since ‘Idea or Concept’ is a ‘Conceptual Entity’, it implies that ‘Temporal Concept’ is also a ‘Conceptual Entity’. As ‘Conceptual Entity’ is a ‘Entity’, it implies that ‘Temporal Concept’ is also an ‘Entity’ too (The entire description of the STY can be seen at Appendix G):

***Temporal Concept class definition*:**

```
<owl:Class rdf:ID="Temporal_Concept" >
    <rdfs:subClassOf  >
      <owl:Class rdf:about="#Idea_or_Concept" />
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Class rdf:about="#Conceptual_Entity" />
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Class rdf:about="#Entity" />
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      ...
</owl:Class>
```

Hierarchical relations of RLs are also stated in UMLS SN with inheritance blocking D. NetON algorithms including reasoning rules for generation of OWL Extended Type documents use this hierarchical information to describe entire super relations of a RL. The following example shows the super relations of ‘part_of’ RL:

***In SRDEF file*:**

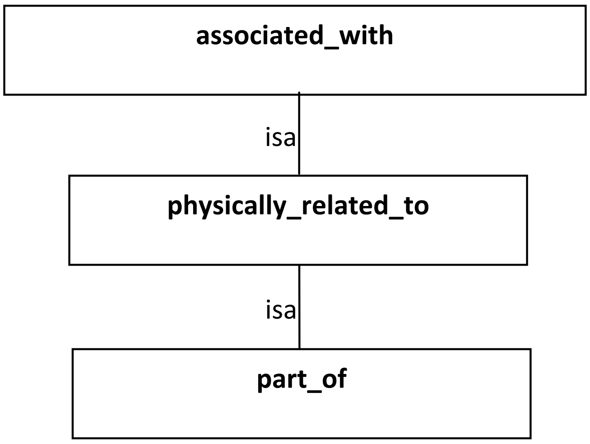
RL|T186|isa|H|The basic hierarchical link in the Network. If one item "isa" another item then the first item is more specific in meaning than the second item.||||IS|inverse_isa|

***In SRSTR file*:**

part_of|isa|physically_related_to|D|

physically_related_to|isa|associated_with|D|

***Graphical Representation of the STYs Hierarchy*:**



***In a NetON OWL Extended document*:**

The following example shows the description of ‘part_of’ relation in both OWL Full-Extended (WOD) and OWL DL-Extended (WOD) documents. It can be seen that ‘part_of’ is a ‘physically_related_to’. Since ‘physically_related_to’ is an ‘associated_with’, it implies that ‘part_of’ is also a ‘associated_with’ too (The entire description of the RL can be seen at Appendix G):

***Part_of relation definition*:**

```
<owl:ObjectProperty rdf:ID="part_of" >
  <rdfs:subPropertyOf rdf:resource="#physically_related_to"/>
  <rdfs:subPropertyOf rdf:resource="#associated_with"/>
</owl:ObjectProperty>
```

3.3.2.4 Validation of NetON Second Dimension Documents

SRFIL, SRFLD, SRDEF and SRSTR files in UMLS SN are inputs used by NetON algorithms. Those algorithms include reasoning rules on top of OWL Sublanguage rules to generate OWL Extended Species documents. OWL Full-Extended and OWL DL-Extended documents contain entire associative relations between STYs and hierarchy of STYs and RLs. The generated OWL Lite-Extended documents contain hierarchy of STYs and RLs and also include hierarchical relations between descendant(s) of STYs and RLs. As the described information in OWL Lite-Extended documents is also included both in OWLDL-Extended and OWL Full-Extended documents validation process is only applied to OWLDL-Extended and OWL Full-Extended documents.

The OWL Extended Species documents in any OWL sublanguage use the same OWL constructors with corresponding OWL Sublanguage Species documents while representing the descendant(s) of STYs and RLs for explicitly defined assertions. This implies that OWL DL-Extended documents are valid (use valid constructors) if corresponding OWL DL document is valid. The OWL Sublanguage documents generated by NetON were validated with as WonderWeb [25] (See section 3.3.1.2).

The SRSTRE1 and SRSTRE2 files (See Appendix E) in UMLS SN contain fully inherited set of valid relations with unique identifiers and with corresponding names, respectively. The entire relation number (including hierarchical relations too) in these files is 6864. The SRSTRE2 file is used to validate OWL Full-Extended and OWL DL-Extended documents generated by NetON. Each new associative relation between STYs and hierarchical relation between STYs and RLs are counted and collected into an array structure during the generation of OWL Extended Species documents. In addition, the SRSTRE2 file is parsed by an algorithm to capture its content. Then, the mentioned array is checked with the SRSTRE2 file content to see whether there are surplus or missing relations in generated OWL Full-Extended and OWL DL-Extended documents. (Figure 18) The same relation number and surplus or missing relations (if there is any) are counted and the result is showed in Table 17.

The information included in Table 17 is shared with users with a message box when the OWL Extended Species documents are chosen to be generated by NetON.

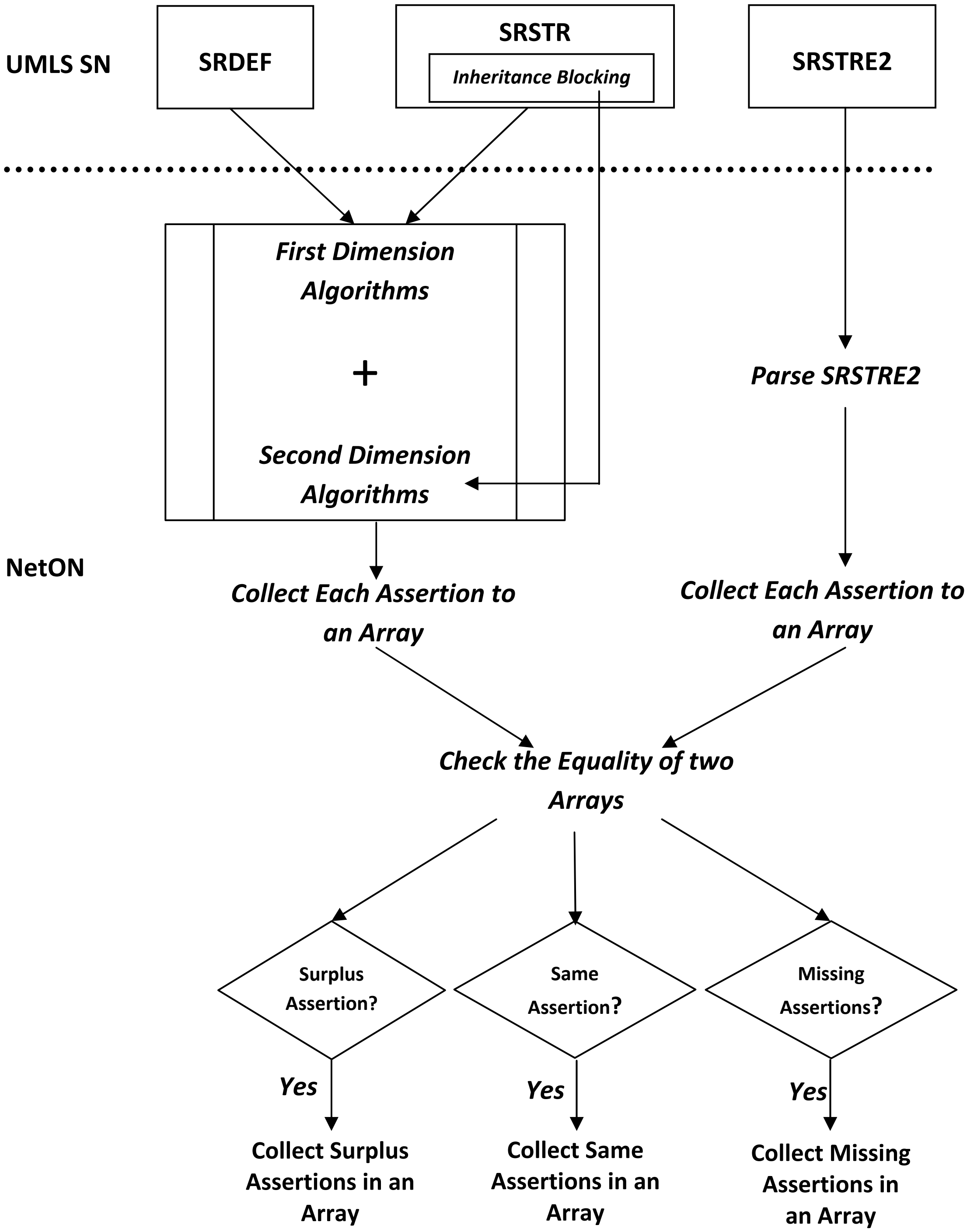

UMLS SN
SRDEF
SRSTR
Inheritance Blocking
SRSTRE2
First Dimension Algorithms
+
Second Dimension Algorithms
Parse SRSTRE2
NetON
Collect Each Assertion to an Array
Collect Each Assertion to an Array
Check the Equality of two Arrays
Surplus Assertion?
Same Assertion?
Missing Assertions?
Yes
Yes
Yes
Collect Surplus Assertions in an Array
Collect Same Assertions in an Array
Collect Missing Assertions in an Array

Figure 18 Validation of the OWL Extended Species

Table 17 Validation check results for NetON OWL Full/DL Extended Species

OWL Extended Species	Assertion Number
Total assertion number	*6864*
Total surplus assertion number	*0*
Total missing assertion number	*0*

3.3.3 Collective Demonstration of First and Second Dimensions

Thinking OWL DL, OWL Full and OWL Lite Species as first dimension and calling them as "Basic" in the second dimension, those two dimension restrictions are demonstrated in the following table. (Table 18)

Table 18 Demonstration of first and second dimensions collectively

Sublanguage	Type	With Hierarchical Relations of STYs & RLs	With Associative Relations Among STYs	With Detailed Information about STYs	With Detailed Information about RLs	With Inherited Assertions by Descendants of RLs	With Inherited Assertions by Descendants of STYs
OWL Full	*Basic*	√	√	√	√		
	Extended	√	√	√	√	√	√
OWLDL	*Basic*	√	√	√			
	Extended	√	√	√		√	√
OWL Lite	*Basic*	√					
	Extended	√				√	√

While generating OWL Species, in addition to two dimensions (Sublanguage and Type) mentioned above NetON users have also an option that includes the detailed information about STYs/RLs inside the document or not. (Table 19) The generated

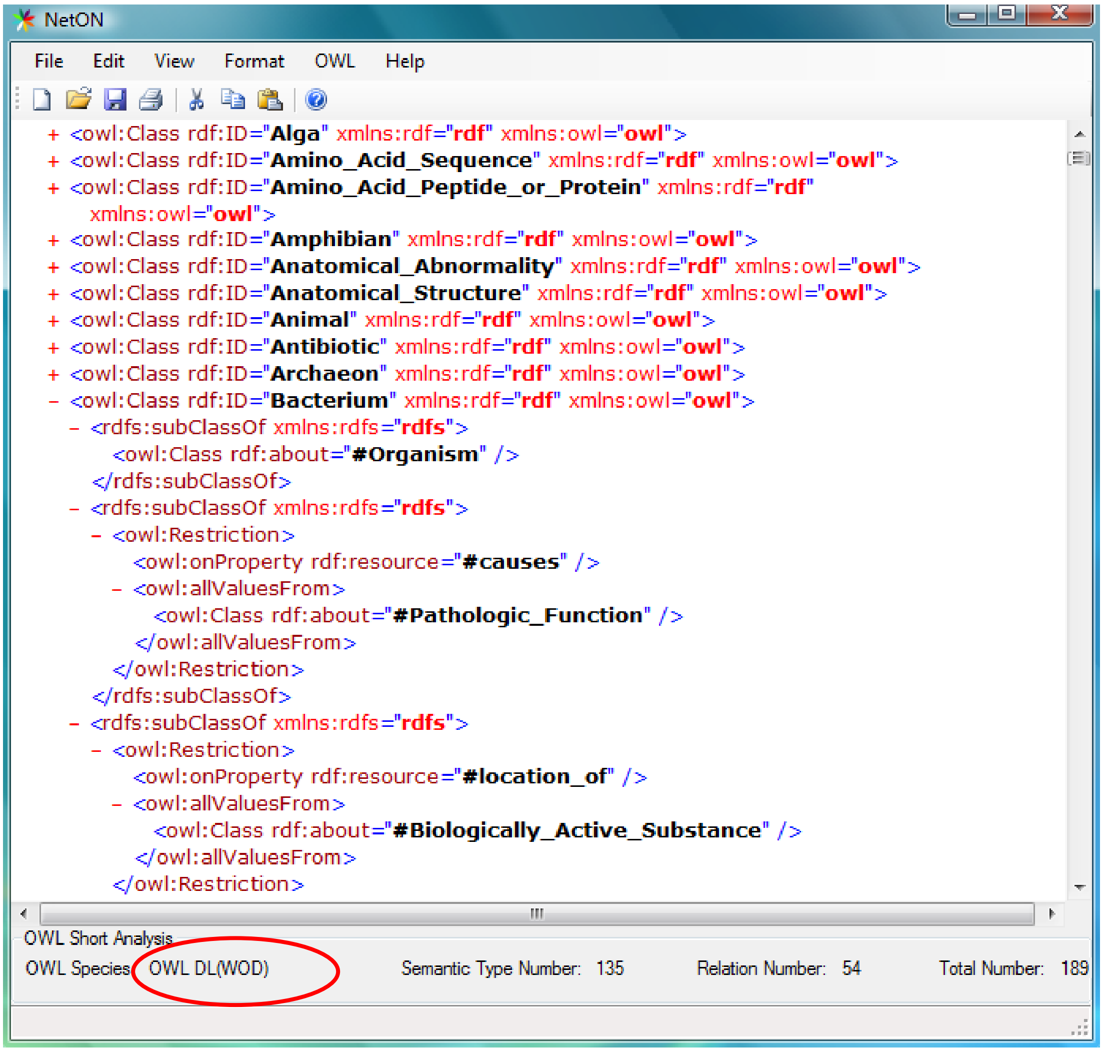
OWL documents without descriptive information for STYs and RLs is named by adding WOD (Without Descriptions) to end of the related OWL Species name.

Table 19 Demonstration of first and second dimensions collectively without detailed information about STYs/RLs

Sublanguage	Type	With Hierarchical Relations of STYs & RLs	With Associative Relations Among STYs	With Inherited Assertions by Descendants of RLs	With Inherited Assertions by Descendants of STYs
OWL Full	*Basic*	√	√		
	Extended	√	√	√	√
OWLDL	*Basic*	√	√		
	Extended	√	√	√	√
OWL Lite	*Basic*	√			
	Extended	√		√	√

3.3.4 An Embedded Web Browser for OWL Species

The NetON allows biomedical application developers to view generated OWL Species in the embedded web browser for rapid and easy browsing as well. This capability of the tool enables human users to follow smoothly the tags and therefore semantics in the interested documents. (Figure 19) In order to share a few transformation information of OWL Species, our tool also includes the OWL Short Analysis box at the bottom of the tool window. The name of generated OWL Species appears at the left side of this box.



```
+ <owl:Class rdf:ID="Alga" xmlns:rdf="rdf" xmlns:owl="owl">
+ <owl:Class rdf:ID="Amino_Acid_Sequence" xmlns:rdf="rdf" xmlns:owl="owl">
+ <owl:Class rdf:ID="Amino_Acid_Peptide_or_Protein" xmlns:rdf="rdf"
    xmlns:owl="owl">
+ <owl:Class rdf:ID="Amphibian" xmlns:rdf="rdf" xmlns:owl="owl">
+ <owl:Class rdf:ID="Anatomical_Abnormality" xmlns:rdf="rdf" xmlns:owl="owl">
+ <owl:Class rdf:ID="Anatomical_Structure" xmlns:rdf="rdf" xmlns:owl="owl">
+ <owl:Class rdf:ID="Animal" xmlns:rdf="rdf" xmlns:owl="owl">
+ <owl:Class rdf:ID="Antibiotic" xmlns:rdf="rdf" xmlns:owl="owl">
+ <owl:Class rdf:ID="Archaeon" xmlns:rdf="rdf" xmlns:owl="owl">
- <owl:Class rdf:ID="Bacterium" xmlns:rdf="rdf" xmlns:owl="owl">
  - <rdfs:subClassOf xmlns:rdfs="rdfs">
      <owl:Class rdf:about="#Organism" />
    </rdfs:subClassOf>
  - <rdfs:subClassOf xmlns:rdfs="rdfs">
    - <owl:Restriction>
        <owl:onProperty rdf:resource="#causes" />
      - <owl:allValuesFrom>
          <owl:Class rdf:about="#Pathologic_Function" />
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
  - <rdfs:subClassOf xmlns:rdfs="rdfs">
    - <owl:Restriction>
        <owl:onProperty rdf:resource="#location_of" />
      - <owl:allValuesFrom>
          <owl:Class rdf:about="#Biologically_Active_Substance" />
        </owl:allValuesFrom>
      </owl:Restriction>
```

OWL Short Analysis

OWL Species OWL DL(WOD)   Semantic Type Number: 135   Relation Number: 54   Total Number: 189

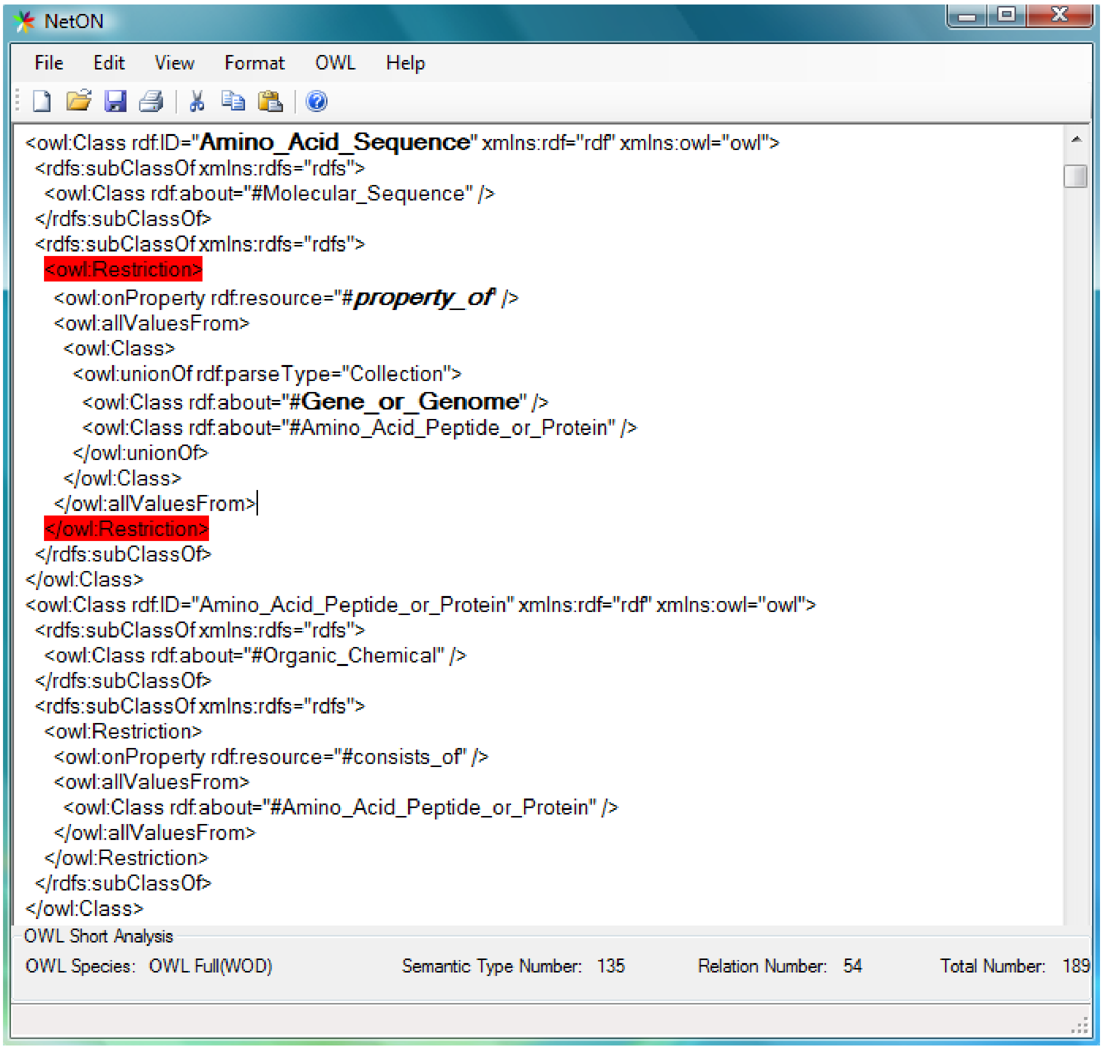
Figure 19 Web browser view of OWL Full Species (WOD)

3.3.5 Customization of OWL Species

When the mentioned OWL Species are not adequate for biomedical application developers, an application embedded in NetON can be used. The application allows them to customize the generated OWL Species. Being simple and flexible text editor for OWL Species, it has the basic text editing functionality and features. The developers can manage the modified OWL documents quickly and effectively to use them in their biomedical applications. The application allows biomedical software developers to create their own OWL Species. (Figure 20) In order to open the application Customization View menu item under View menu should be selected.

Further detailed information about customization options and their usage explanations can be found in Appendix F.

Many users will find the embedded application as a simple text editor for creating and editing documents on .txt, .rdf, or .owl format. Developers using NetON cannot unintentionally save particular formatting in documents that need to stay pure text, since the embedded text editor supports only very fundamental formatting. This is particularly useful when creating XML, RDF or OWL documents because special characters or other formatting may cause errors.



```
<owl:Class rdf:ID="Amino_Acid_Sequence" xmlns:rdf="rdf" xmlns:owl="owl">
 <rdfs:subClassOf xmlns:rdfs="rdfs">
  <owl:Class rdf:about="#Molecular_Sequence" />
 </rdfs:subClassOf>
 <rdfs:subClassOf xmlns:rdfs="rdfs">
  <owl:Restriction>
   <owl:onProperty rdf:resource="#property_of" />
   <owl:allValuesFrom>
    <owl:Class>
     <owl:unionOf rdf:parseType="Collection">
      <owl:Class rdf:about="#Gene_or_Genome" />
      <owl:Class rdf:about="#Amino_Acid_Peptide_or_Protein" />
     </owl:unionOf>
    </owl:Class>
   </owl:allValuesFrom>
  </owl:Restriction>
 </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="Amino_Acid_Peptide_or_Protein" xmlns:rdf="rdf" xmlns:owl="owl">
 <rdfs:subClassOf xmlns:rdfs="rdfs">
  <owl:Class rdf:about="#Organic_Chemical" />
 </rdfs:subClassOf>
 <rdfs:subClassOf xmlns:rdfs="rdfs">
  <owl:Restriction>
   <owl:onProperty rdf:resource="#consists_of" />
   <owl:allValuesFrom>
    <owl:Class rdf:about="#Amino_Acid_Peptide_or_Protein" />
   </owl:allValuesFrom>
  </owl:Restriction>
 </rdfs:subClassOf>
</owl:Class>
```

OWL Short Analysis

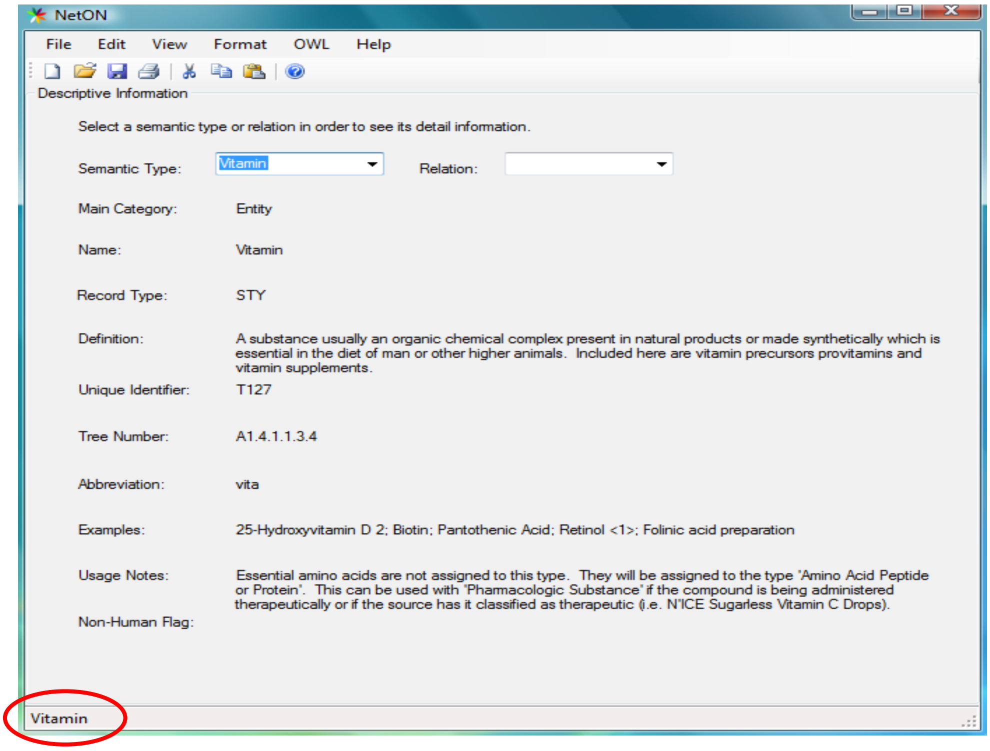
OWL Species: OWL Full(WOD)   Semantic Type Number: 135   Relation Number: 54   Total Number: 189

Figure 20 The embedded application in NetON for customization of OWL Species

3.3.6 An Embedded Application for Developers

There is also another application embedded in the NetON. As main biomedical knowledge is described by UMLS SN by the help of semantic types and relations, software developers may need to look at the descriptive information of STYs and RLs in order to decide how and where to use them before designing and building their own applications. In order to open the application Record Details View menu item under View menu should be selected.

With the intention of help to the biomedical application developers, the embedded application offers a user friendly interface to share the part of the UMLS SN. Before using this application the user needs to select an OWL Species with descriptive information (the name of the document should not include WOD abbreviation) in order to make the NetON generates the document. Then, the user can run the application in which the developed algorithm parses the generated OWL document and shows the descriptive information of STYs or RLs in the user friendly way. (Figure 21)



Descriptive Information

Select a semantic type or relation in order to see its detail information.

Semantic Type: Vitamin   Relation:

Main Category: Entity

Name: Vitamin

Record Type: STY

Definition: A substance usually an organic chemical complex present in natural products or made synthetically which is essential in the diet of man or other higher animals. Included here are vitamin precursors provitamins and vitamin supplements.

Unique Identifier: T127

Tree Number: A1.4.1.1.3.4

Abbreviation: vita

Examples: 25-Hydroxyvitamin D 2; Biotin; Pantothenic Acid; Retinol <1>; Folinic acid preparation

Usage Notes: Essential amino acids are not assigned to this type. They will be assigned to the type 'Amino Acid Peptide or Protein'. This can be used with 'Pharmacologic Substance' if the compound is being administered therapeutically or if the source has it classified as therapeutic (i.e. N'ICE Sugarless Vitamin C Drops).

Non-Human Flag:

Vitamin

Figure 21 User friendly share of descriptive information of STYs or RLs

3.3.7 Related Literature Studies

Recall that Schulz S. et.al. [22] manually transformed the UMLS SN to OWL DL by changing the original semantics represented in UMLS SN files. Jimenez-Ruiz E. [20] transformed some part of UMLS SN to OWL Full. Kashyap V., et.al. [24] recommended OWL constructors for representation of STYs and RLs and also discussed eight different potential interpretations of semantic relations between STYs in Description Logic format by concluding that applications should chose among diverse representation. Table 20 shows the primary constructors used in related literature studies. Constructors used for descriptive information of STYs and RLs in related literature studies can be seen at Table 21 and Table 22, respectively.

Table 20 Primary constructors used in related literature studies

Transformation Constructors		**Related Literature Studies**		
		Kashyap V., et.al. [24] (OWL DL)	**Schulz S. et.al. [22] (OWL DL)**	**Jimenez-Ruiz E. [20] (OWL Full)**
UMLS SN	**STY**	owl:Class (proposed)	owl:Class	owl:Class
	RL	owl:objectProperty (proposed)	owl:Class	owl:objectProperty
	Assertion	8 different possibilities in DL format were proposed by concluding that applications should choose among them	owl:Class	owl:Restriction, owl:onProperty, owl:allValues
	Inheritance Blocking	-	-	-

Table 21 Constructors used for descriptive information of STYs in related literature studies

Constructors for descriptive information of STYs		**Related Literature Studies**		
		Kashyap V., et.al. [24] (OWL DL)	**Schulz S. et.al. [22] (OWL DL)**	**Jimenez-Ruiz E. [20] (OWL Full)**
UMLS SN	***record type***	-	<snRecordType>	-
	unique identifier	-	-	-
	STY name	-	-	<OriginalName>
	STY tree number	-	<treeNumber>	-
	definition	-	<rdfs:comment>	<rdfs:comment>
	examples	-	-	<rdfs:label>
	usage note	-	-	-
	non-human flag	-	-	-
	abbreviation	-	-	-

Table 22 Constructors used for descriptive information of RLs in related literature studies

Constructors for descriptive information of RLs		**Related Literature Studies**		
		Kashyap V., et.al. [24] (OWL DL)	**Schulz S. et.al. [22] (OWL DL)**	**Jimenez-Ruiz E. [20] (OWL Full)**
UMLS SN	***record type***	-	<snRecordType>	-
	unique identifier	-	-	-
	RL name	-	-	-
	RL tree number	-	<treeNumber>	-
	definition	-	<rdfs:comment>	-
	examples	-	-	-
	usage note	-	-	-
	non-human flag	-	-	-
	abbreviation	-	-	-

Kashyap V., et.al. [24] recommended OWL constructors for representation of STYs and RLs and also discussed eight different potential interpretations of semantic relations between STYs in Description Logic format by concluding that applications should chose among diverse representation. However, Kashyap V., et. al. [24] did not offer an OWL-based UMLS SN document. Schulz S. et.al. [22] manually transformed the UMLS SN to OWL DL by changing the original semantics represented in UMLS SN files. The OWL DL version UMLS SN document basically included 612 explicitly specified assertions (from SRSTR file). Since they changed the original semantics, their document was hard to understand and to follow the intended meaning of the primitives they used. Additionally, it is not possible for an OWL reasoner to use their OWL DL document for inferring the hidden assertions. Jimenez-Ruiz E. [20] transformed some part of UMLS SN to OWL Full. The OWL-based UMLS SN document basically included 612 explicitly specified assertions (from SRSTR file) and 234 inverse assertions. When an OWL reasoner uses the OWL document developed by Jimenez-Ruiz E. [20] to infer the hidden assertions, the result will include 78 false positives. (Table 23)

Table 23 Number of false positive assertions when an OWL reasoner uses documents developed by Schulz S. et.al. [22] or by Jimenez-Ruiz E. [20]

	OWL Reasoner Input Document by Schulz S. et.al. [22]	**OWL Reasoner Input Document by Jimenez-Ruiz E. [20]**
False Positive Assertion Number	Not applicable	78

Schulz S. et.al. [22] and Jimenez-Ruiz E. [20] allocated class tag for modeling of the UMLS SN types in OWL. Kashyap V., et.al. [24] proposed a class tag for representation of UMLS SN types. Schulz S. et.al. [22] and Jimenez-Ruiz E. [20] modeled taxonomic hierarchy of the semantic types by OWL subclasses to affirm taxonomic subsumption. Kashyap V., et.al. [24] proposed the same class axiom for hierarchy representation of UMLS SN types.

Schulz S. et.al. [22]

```
<owl:Class rdf:about="#T001">
  <rdfs:subClassOf rdf:resource="#T072" />
  ...

</owl:Class>
```

Jimenez-Ruiz E. [20]

```
<owl:Class rdf:ID="Organism">
 <rdfs:subClassOf>
  <owl:Class rdf:about="#Physical_Object" />
 </rdfs:subClassOf>
  ...
</owl:Class >
```

Kashyap V., et.al. [24] suggested the translation of UMLS Semantic Network relations to OWL object properties. Jimenez-Ruiz E. [20] showed that the same property tag was chosen for representation of UMLS SN relations. Schulz S. et.al. [22] , on the other hand, used only two object properties in the representation of UMLS Semantic Network in OWL. These used object properties (hasDomain and hasRange) were not included in the original UMLS Semantic Network files [44]. Schulz S., et al., alternatively, decided to represent RLs with owl:Class constructor[22] by changing the original semantics represented in UMLS SN. Kashyap V., et.al. [24] proposed OWL subproperties for representation of taxonomic hierarchy of the semantic relations (RLs). Jimenez-Ruiz E. [20] chose the same property axiom for hierarchy representation of UMLS SN relations. Schulz S. et.al., however, presented the relation hierarchy with rdfs:subClassOf since they represented the relations as classes in their representation of UMLS Semantic Network in OWL [22].

Schulz S. et.al. [22]

```
<owl:Class rdf:about="#T147">
   <rdfs:label xml:lang="en">causes</rdfs:label>
   <rdfs:subClassOf rdf:resource="#T187" />
   ...
</owl:Class>

<owl:Class rdf:about="#T187">
  <rdfs:label xml:lang="en">brings_about</rdfs:label>
  <rdfs:subClassOf rdf:resource="#T139" />
  ...
</owl:Class>
```

Jimenez-Ruiz E. [20]

```
<owl:ObjectProperty rdf:ID="causes">
  <rdfs:subPropertyOf rdf:resource="#brings_about" />
  ...
</owl:ObjectProperty>
```

Kashyap V., et.al. [24] proposed inverseof role constructor for specification of inverses of relations in the UMLS Semantic Network. Jimenez-Ruiz E. [20] used the same property axiom for representation of UMLS SN inverse relations. Schulz S. et.al. [22], however, did not present the inverse of relations as they did not even represent the relations (object properties) without changing their semantics represented in UMLS Semantic Network.

Jimenez-Ruiz E. [20]

```
<owl:ObjectProperty rdf:ID="causes">
  ...
  <owl:inverseOf rdf:resource="#caused_by" />
</owl:ObjectProperty>
```

Schulz S. et.al. [22] looked also for representation of UMLS Semantic Network covering the range and domain limitations of the semantic relations between STYs (assertions) by using BioTop [23]. Recall that BioTop was developed by Schulz S. et.al. [23] as an abstract model of biomedical domain. Since BioTop was in OWL DL, first the authors manually performed the representation of UMLS SN in OWL DL formalism. They used only two object properties in the representation of UMLS Semantic Network in OWL. These used object properties (hasDomain and hasRange) were not included in the original UMLS Semantic Network files [44]. As the main aim of the authors was mapping process, relations between semantic types in UMLS SN were represented as classes by using hasDomain and hasRange properties restriction. In a relation description (with owl:class), Schulz S. et.al. [22] also included the explicitly defined assertions related to this relation. The below notation in OWL does not actually represent the semantics of the information included in UMLS Semantic Network.

Schulz S. et.al. [22]

```
<owl:Class rdf:about="#T147_T005_T046">
   <rdfs:label xml:lang="en">
     causes with domain Virus and range Pathologic Function
   </rdfs:label>
   <rdfs:subClassOf rdf:resource="#T147" />
   <rdfs:subClassOf>
       <owl:Restriction>
          <owl:onProperty rdf:resource="#hasDomain" />
          <owl:allValuesFrom rdf:resource="#T005" />
       </owl:Restriction>
   </rdfs:subClassOf>
   <rdfs:subClassOf>
       <owl:Restriction>
          <owl:onProperty rdf:resource="#hasRange" />
          <owl:allValuesFrom rdf:resource="#T046" />
       </owl:Restriction>
   </rdfs:subClassOf>
```

```
</owl:Class>

<owl:ObjectProperty rdf:about="#hasDomain" />
<owl:ObjectProperty rdf:about="#hasRange" />

<owl:Class rdf:about="#T147">
  <rdfs:label xml:lang="en">causes</rdfs:label>
  <owl:equivalentClass>
  <owl:Class>
    <owl:unionOf rdf:parseType="Collection">
       <rdf:Description rdf:about="#T147_T004_T046" />
       <rdf:Description rdf:about="#T147_T005_T046" />
       <rdf:Description rdf:about="#T147_T006_T046" />
       <rdf:Description rdf:about="#T147_T007_T046" />
       <rdf:Description rdf:about="#T147_T009_T046" />
       <rdf:Description rdf:about="#T147_T073_T037" />
       <rdf:Description rdf:about="#T147_T073_T046" />
       <rdf:Description rdf:about="#T147_T073_T190" />
       <rdf:Description rdf:about="#T147_T167_T037" />
       <rdf:Description rdf:about="#T147_T167_T046" />
       <rdf:Description rdf:about="#T147_T167_T190" />
    </owl:unionOf>
  </owl:Class>
  </owl:equivalentClass>
  ...
</owl:Class>
```

Jimenez-Ruiz E. [20] used the restriction on property representation for the semantic relations between STYs (assertions).

Jimenez-Ruiz E. [20]

```
<owl:Class rdf:ID="Tissue">
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#contained_in" />
        <owl:allValuesFrom>
```

```
          <owl:Class rdf:about="#Body_Space_or_Junction" />
        </owl:allValuesFrom>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#constitutes" />
        <owl:allValuesFrom>
          <owl:Class rdf:about="#Tissue" />
        </owl:allValuesFrom>
    </owl:Restriction>
  </rdfs:subClassOf>
   ...
</owl:Class>
```

Schulz S. et.al. [22] included a few descriptive information about the RLs and STYs in their OWL version of UMLS SN:

Schulz S. et.al. [22]

```
<owl:Class rdf:about="#T001">
  ...
  <rdfs:label xml:lang="en">Organism</rdfs:label>
  <snRecordType>STY</snRecordType>
  <treeNumber>A1.1</treeNumber>
  <rdfs:comment xml:lang="en">Generally, a living
      individual, including all plants and animals.
  </rdfs:comment>
</owl:Class>

<owl:Class rdf:about="#T147">
  ...
  <rdfs:label xml:lang="en">causes</rdfs:label>
  <snRecordType>RL</snRecordType>
  <treeNumber>R3.2.2</treeNumber>
  <rdfs:comment xml:lang="en">Brings about a condition or an
          effect. Implied here is that an agent, such as for
          example, a pharmacologic substance or an organism,
          has brought about the effect. This includes induces,
          effects, evokes, and etiology.
  </rdfs:comment>
</owl:Class>
```

Jimenez-Ruiz E. [20] did not include descriptive information about the RLs, but some descriptive information about the STYs was identified as attributes of classes in his OWL version of UMLS SN. Since individuals and classes were disjoint domains in OWL Lite and DL, when the attributes (OriginalName, HeC_Level) represented like in the below example the class (`Cell_or_Molecular_Dysfunction)` would be both a class and an individual at the same time. Therefore, the mentioned OWL description could not be either OWL Lite or OWL DL but only could be OWL Full [20]:

Jimenez-Ruiz E. [20]

```
<owl:Class rdf:ID="Cell_or_Molecular_Dysfunction">
   ...
   <rdfs:comment>
     A pathologic function inherent to cells, parts of cells,
     or molecules.
   </rdfs:comment>
   <rdfs:label>
     Examples of Metathesaurus concepts: DNA Damage;
     Wallerian Degeneration; Atypical squamous metaplasia
   </rdfs:label>
   <OriginalName>Cell or Molecular Dysfunction</OriginalName>
   <HeC_Level>Cellular</HeC_Level>
 </owl:Class>
```

Schulz S. et.al. [22] used `owl:disjointWith` constructor to describe that any semantic type and its subclasses were mutually exclusive. In UMLS SN, however, there is not any explicit declaration on mutual exclusivity among STYs. The below notation in OWL does not actually represent the semantics of the information included in UMLS Semantic Network.

Schulz S. et.al. [22]

```
<owl:Class rdf:about="#T021Remaining">
  <rdfs:label xml:lang="en">Fully Formed Anatomical Structure
      rest class</rdfs:label>
  <rdfs:subClassOf rdf:resource="#T021" />
  <owl:disjointWith rdf:resource="#T023" />
  <owl:disjointWith rdf:resource="#T024" />
  <owl:disjointWith rdf:resource="#T025" />
  <owl:disjointWith rdf:resource="#T026" />
  <owl:disjointWith rdf:resource="#T028" />
</owl:Class>
```

Jimenez-Ruiz E. [20] also included the inverse relations for the description of STYs in OWL. This information can actually be inferred by any OWL Reasoner [42, 43]. The inverse relations added to the description of STYs by Jimenez-Ruiz E. [20] are redundant information in fact.

```
<owl:Class rdf:ID="Anatomical_Abnormality">
  ...
  <rdfs:subClassOf>
   <owl:Restriction>
    <owl:onProperty rdf:resource="#treated_by" />
      <owl:allValuesFrom>
       <owl:Class>
       <owl:unionOf rdf:parseType="Collection">
       <owl:Class rdf:about=“#Pharmacologic_Substance" />
       <owl:Class rdf:about=“#Medical_Device" />
      <owl:Class rdf:about=“#Therapeutic_or_Preventive_Procedure"/>
       </owl:unionOf>
       </owl:Class>
      </owl:allValuesFrom>
```

```
     </owl:Restriction>
    </rdfs:subClassOf>
</owl:Class>
```

3.3.8 Contribution of NetON OWL Species

The literature survey reveals that there is no study for automatically transforming the maximum amount of UMLS SN information into OWL Sublanguages without changing the biomedical abstract model semantics. There was not also any study on transforming whole information included in UMLS SN to OWL without changing the owned semantics.

There wasn’t any study on representation of the maximum amount of UMLS SN information in OWL Lite standard. Schulz S. et.al. [22] manually transformed the UMLS SN to OWL DL by changing the original semantics represented in UMLS SN files. The OWL DL version UMLS SN document basically included 612 explicitly specified assertions (from SRSTR file). As they changed the original semantics their document was hard to understand and to follow the intended meaning of the primitives they used. Jimenez-Ruiz E. [20] transformed some part of UMLS SN to OWL Full. The OWL-based UMLS SN document basically included 612 explicitly specified assertions (from SRSTR file) and 234 inverse assertions which could also be inferred by any OWL reasoner [42, 43]. (See section 3.3.7 for inverse relation representation by Jimenez-Ruiz E. [20]) Kashyap V., et.al. [24] recommended OWL constructors for representation of STYs and RLs and also discussed eight different potential interpretations of semantic relations between STYs in Description Logic format by concluding that applications should chose among diverse representation. However, Kashyap V., et. al. [24] did not offer an OWL-based UMLS SN document.

NetON transforms UMLS SN to OWL in order to achieve a semantically convenient representation of upper-level abstraction of the biomedical domain. NetON was designed and built for the automatic transformation of UMLS Semantic Network to OWL sublanguages to support semantic operations between biomedical systems.

The tool generates accessible OWL contents for both humans and software applications so that, respectively, developers can easily understand the generated OWL documents and biomedical applications can efficiently interoperate semantically.

The major contribution of NetON is the automatic transformation of UMLS SN to OWL sublanguages that use the first dimension algorithms in NetON. These algorithms were developed with the aim of maximum possible information transformation from UMLS SN to OWL sublanguages without changing the biomedical abstract model semantics. The only information that is not able to be transformed to any OWL Basic Species due to the lack of appropriate constructors in OWL standard is inheritance blockings in UMLS SN. (See section 3.3.1.3)

The generated OWL Lite Species support developers who require uncomplicated restrictions and classification hierarchy in UMLS SN. Having simpler formal representation than relative OWL sublanguages, the OWL Lite Species offer basic approach for UMLS SN taxonomy. Facility on comprehension and implementation by application developers is a benefit of OWL Lite Species which also has a drawback of limited interpretation. OWL Lite Species offer a smallest practical part of UMLS Semantic Network with comparatively precise manner to uphold application developers. The subdued complexity will pave the way for a constructive influence on the effectiveness of entire reasoners on OWL Lite Species.

The OWL DL Species are generated by NetON to support developers who require utmost articulation from UMLS SN while ensuring inferences to be computable and promising decidable reasoning for an OWL reasoner [42, 43]. Encompassing the representation in OWL Lite Species, assertions and STYs detailed information are also transformed into OWL DL Species in order to make them available to the users. Being based on description logic, OWL DL Species prohibit the usage of OWL constructs to one another and consequently have well-organized support on reasoning and had attractive assets for reasoning systems. RDF restriction in one respect and extension in another, however, permits the lost of entire compatibility of OWL DL Species with RDF. Since the inheritance blocking information in UMLS SN cannot represented in OWL DL Species due to the lack of appropriate constructors in OWL standard, the users interesting with reasoning applications should be aware of the fact that a few deduction outcomes will be false positives. (See section 3.3.1.3)

The generated OWL Full Species support developers who require utmost articulation from UMLS SN while ensuring unlimited RDF usage without assurance of accurate or complete inferences. OWL Full Species comprise the OWL DL Species representation with different notations for STYs detailed information and also include the RLs detailed information in order to make them available to users. Inheritance blockings are the only information that is not able to be transformed into any OWL Species due to the lack of appropriate constructors in OWL standard.

OWL DL and OWL Lite Species use limited part of RDFS and RDF, while OWL Full Species extend them. OWL ontology developers must decide on the most appropriate OWL Species according to their requirements. Requirements of developers or applications on information representation from UMLS SN will determine the selection between OWL Lite, OWL DL and OWL Full Species.

In UMLS SN, there are unseen assertions that can be inferred by using inference rules on explicitly specified assertions in SRSTR file that includes the structure of UMLS SN. However, explicitly declared relations between semantic types are not essentially valid for all the descendants. Deduction outcomes of any OWL reasoners on NetON OWL Basic Species will also include false positives due to the lack of inheritance blocking information. The algorithms of the second (Extended) dimension consider inheritance blocking information while executing inference rules. As this cannot be done by any OWL reasoner [42, 43], the second dimension offers a solution for application developers. The whole information included in UMLS SN is transformed to OWL without changing the biomedical abstract model semantics by using developed second dimension algorithms.

The OWL Lite Extended documents support developers who require machine processable form of UMLS SN information including entire classification hierarchy of UMLS SN in OWL Lite Format. The OWL Lite Extended documents are able to be very helpful for the NetON users who do not know or not want to deal with reasoning algorithms. The OWL DL/Full Extended documents support developers who do not know or not want to deal with reasoning algorithms, but require machine processable form of UMLS SN information including whole concealed and valid information in OWL DL/Full Format. The OWL DL Extended documents can also be used by applications, which utilize reasoning on UMLS SN information, as a control file including all valid relations.

The NetON users should determine the most appropriate OWL Species among first and second dimension documents. OWL Sublanguage (First Dimension) documents will be helpful for those NetON users who only need explicitly defined information in UMLS SN. OWL Extended (Second Dimension) documents will be helpful for those who also need inherited information for descendants of STYs and RLs in addition to explicitly defined information in UMLS SN.

OWL Extended Species (Second Dimension) documents can be used by application developers in two different ways. In one way, developers can directly use OWL Extended Species (Second Dimension) documents, which contain entire valid assertions and their descendants, in their applications. In second way, developers can use OWL Extended Species documents as control files in reasoning applications in order to control and eliminate the false positive results.

Proposed UMLS SN transformed by NetON promises reliable, correct, comprehensive, and significant representation of relations and classes. Consequently, dealings different biomedical activities can be carried out in a more efficient, generic and easier way over OWL content of UMLS SN.

CHAPTER 4

DISCUSSION AND CONCLUSION

4.1 The NetON Role in Migration to Contemporary Use of UMLS SN for Biomedical Applications and Developer(s)

The UMLS SN semantics are distributed among different files [44] and cannot be effectively used as such to perform any semantic activity. In the conventional use of UMLS SN, each application developer(s) must understand the semantics of UMLS SN and then use different subsets (depicted as diverse colors in Figure 22) in his/her software formalism (depicted as different shapes in Figure 22).

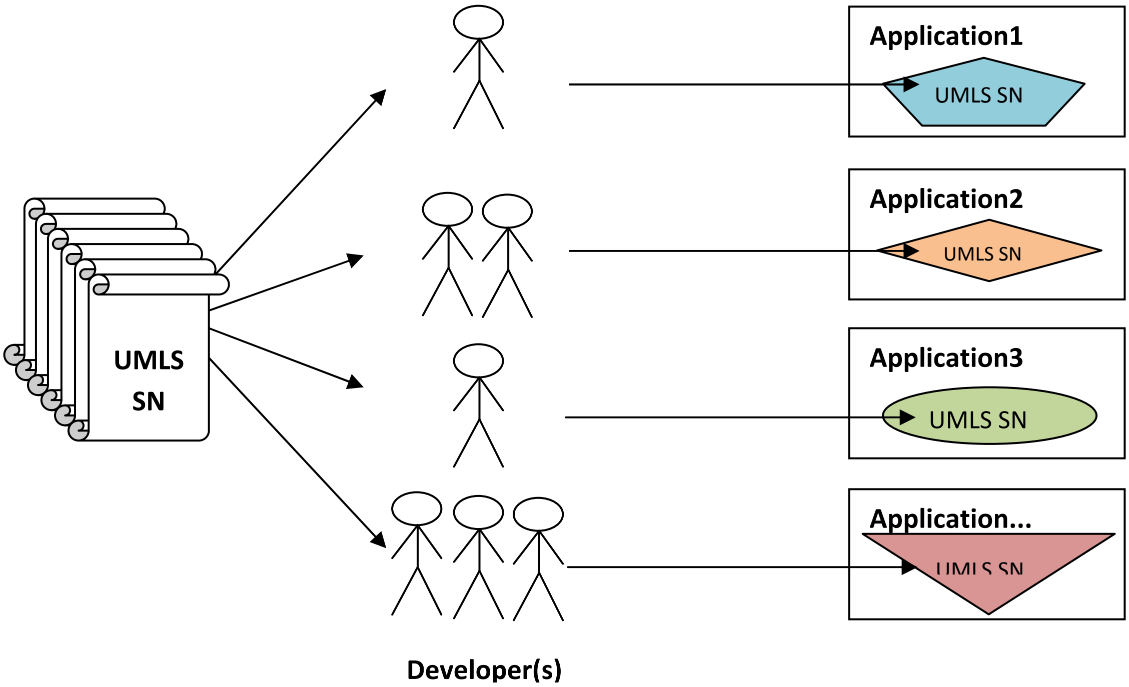


Figure 22 Conventional use of UMLS SN

NetON, being designed and developed for contemporary use of UMLS SN, allows migration from semantic formalism on a per-application basis to a standardized semantics representation (depicted as same shapes in Figure 32). Our tool allows computer application developers to transform UMLS SN to OWL Species with the aim of achieving semantic machine interoperability between biomedical information systems. The approach used by NetON allows biomedical applications to be compatible with Semantic Web standards. Our tool can help in upgrading already deployed systems by offering the generated UMLS SN OWL Species as biomedical data components. Since NetON provides an agreed-upon vocabulary for expressing semantics in a machine-processable way, application developers will need to write less code, will have less chance of misinterpretation, and will be able to understand the semantics of generated OWL Species. (Figure 23)

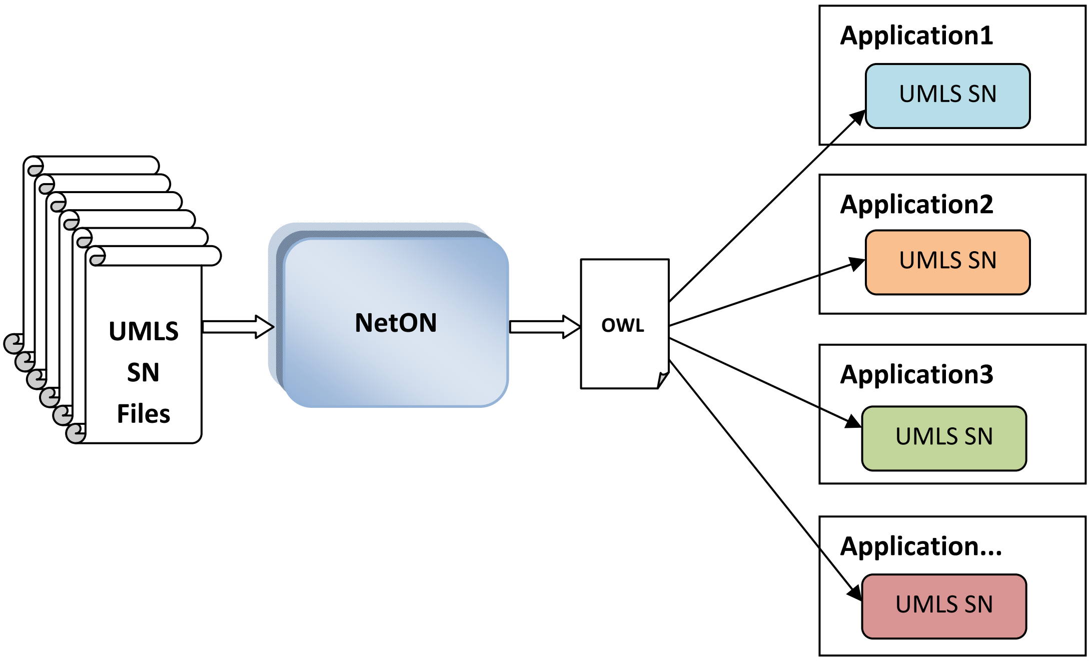


Figure 23 Updated NetON approach for biomedical application(s)

4.2 The NetON Role in Exchanging Information between Biomedical Systems

Biomedical information is typically allocated in diverse independent systems having semantic and/or syntactic incompatibility. The use of biomedical information generally calls for integrating, classifying and/or comparing biomedical data originating in these diverse systems. In an ideal world, biomedical information would be represented by common standards in all information systems. However, this expectation is overly idealistic as each biomedical and health domain or corresponding system might have different information representation requirements using diverse formalism. For a realistic contemporary approach, there is a need for a middle layer using common standards to exchange information between systems. We recommend a semantic layer to facilitate the exchange of information among technologically independent biomedical systems. A major advantage of having a semantic middle layer is that existing technologies and applications won't be required to change; rather, a new layer using existent infrastructures will be added. (Figure 24) Advances in the Semantic Web domain make it a suitable standard for works demanding semantic knowledge associated with biomedical systems. In the figure below, different shapes of arrows denote the diverse formalism notations of biomedical information.

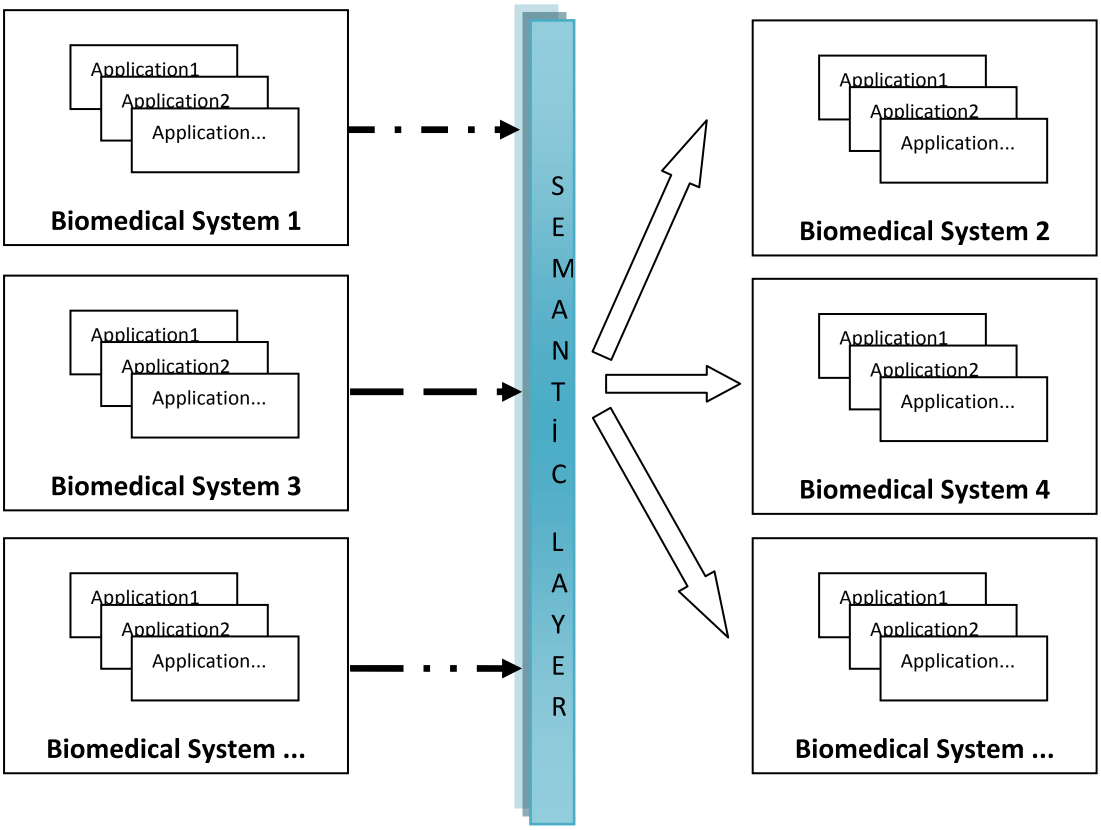


Figure 24 Proposed architecture for exchanging information among biomedical systems

Biomedical concepts/categories cannot be completely defined due to the fact that new circumstances and relations will be constantly explored in unrestrictedly large biology and medicine domains. This dynamic nature affects the content of UMLS Semantic Network. This implies that there is a need of a dynamic information model that is adjusted to continually enlarging biomedical necessities. Nevertheless, format limitations of information description in UMLS SN make the achievement of this goal difficult, as described in Section 3.1.1. In view of the fact that representing UMLS SN content for semantic actions is a significant issue, NetON combines UMLS SN and Semantic Web to achieve a semantically convenient representation of upper-level abstraction of the biomedical domain. Our tool provides an implementation of and support for the above mentioned middle semantic layer to facilitate semantic interoperability. (Figure 25)

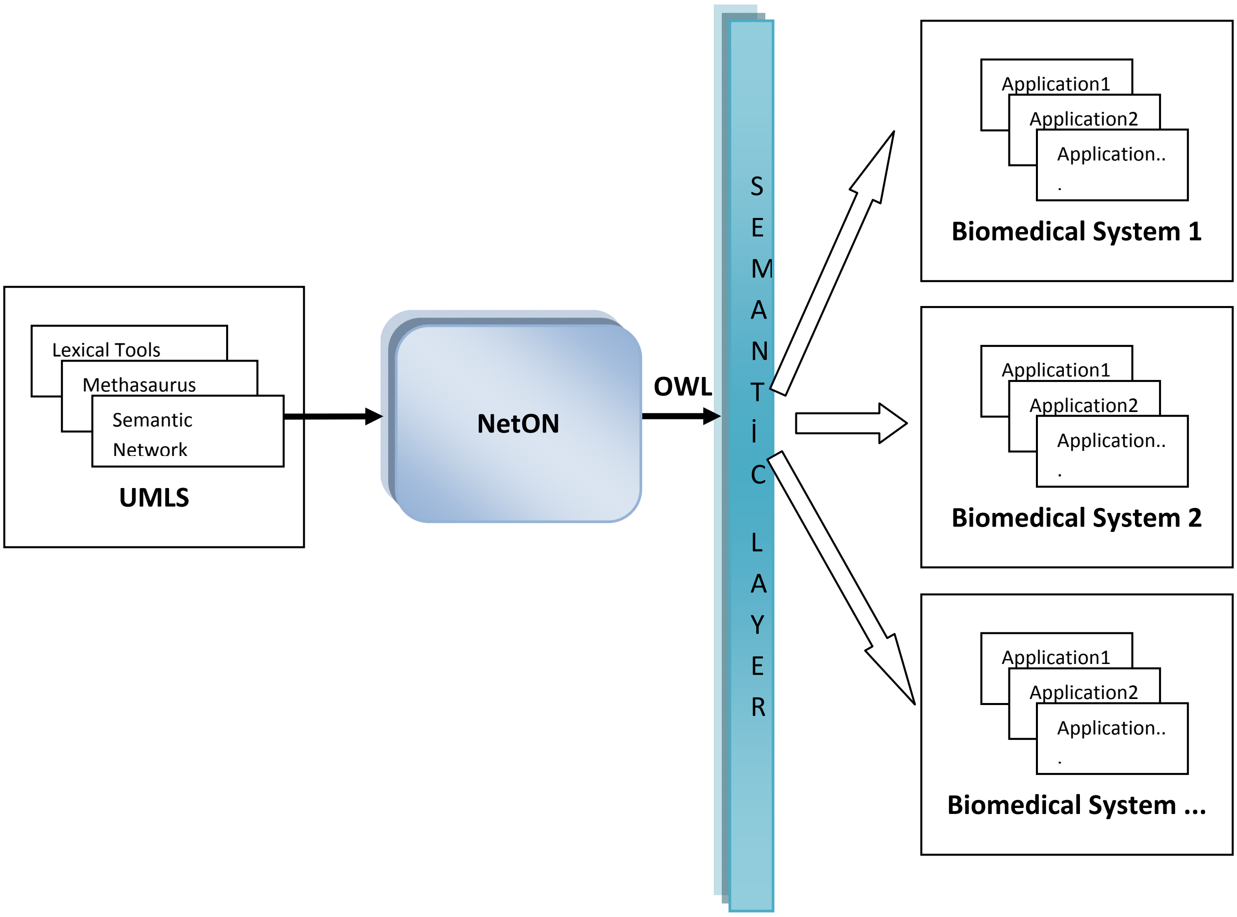


Figure 25 The contribution of NetON for interoperability of biomedical systems

4.3 Conclusion

In biological and medical information systems, biomedical vocabularies have been essential at all times. Even though a variety of related knowledge sources containing vast amounts of valuable domain knowledge have been developed so far (e.g. SNOMED and UMLS), they cannot be directly integrated into real-world biomedical information systems. In addition to being too comprehensive, most of them are also not formalized in a suitable representation language to be reused and shared.

Being an all inclusive source of upper-level abstraction for the biomedical domain, UMLS Semantic Network, due to considerably many relationships, do have a rather complex structure thus making it difficult for human orientation. It is therefore that its use appears limited even though the semantic network is a valuable source of modeling contents in the biomedical domain. The need for knowledge-intensive activities in biomedical applications makes it necessary to transform UMLS SN current formalism to a proper format that uses a knowledge-oriented language.

The benefits of OWL (See section 3.1.2) indicates that Ontology Web Language is a suitable candidate standard for representation of UMLS SN. It is seen that major restrictions of UMLS SN content can be described by OWL constructors, with detailed inspection of OWL and UMLS SN.

The discovery of new information, relationships, and details necessitates the continuous evolvement of UMLS SN to adapt to biomedical requirements. Therefore, it is seen that any transformation effort on the formalism of UMLS SN should be done automatically to immediately reflect the change to the new formalism.

The literature survey revealed that there is no study on automatically transforming the maximum amount of UMLS SN information into OWL Sublanguages without changing the biomedical abstract model semantics. There was not also any study on transforming whole information included in UMLS SN to OWL without changing the owned semantics.

There wasn't any study for representing the maximum amount of UMLS SN information in OWL Lite standard. Schulz S. et.al. [22] manually transformed the UMLS SN to OWL DL by changing the original semantics represented in UMLS SN files. The OWL DL version UMLS SN document basically included 612 explicitly specified assertions (from SRSTR file). As they changed the original semantics their document was hard to understand and to follow the intended meaning of the primitives they used. Jimenez-Ruiz E. [20] transformed some part of UMLS SN to OWL Full. The OWL-based UMLS SN document basically included 612 explicitly specified assertions (from SRSTR file) and 234 inverse assertions which could also be inferred by any OWL reasoner [42, 43]. (See section 3.3.7 for inverse relation representation by Jimenez-Ruiz E. [20]) Kashyap V., et.al. [24] recommended OWL constructors for representation of STYs and RLs and also discussed eight different potential interpretations of semantic relations between STYs in Description Logic format by concluding that applications should chose among diverse representation. However, Kashyap V., et. al. [24] did not offer an OWL-based UMLS SN document.

The major contribution of NetON is the automatic transformation of UMLS SN to OWL sublanguages (Lite, DL, Full) that use the first dimension algorithms in NetON. These algorithms were developed with the aim of maximum possible information transformation from UMLS SN to OWL sublanguages without changing the biomedical abstract model semantics. The only information that is not able to be transformed to any OWL Basic Species due to the lack of appropriate constructors in OWL standard is inheritance blockings in UMLS SN. (See section 3.3.1.3)

OWL DL-Basic and OWL Lite-Basic Species use limited parts of RDFS and RDF, while OWL Full-Basic Species extend them. OWL ontology developers must decide on the most appropriate OWL Basic Species according to their requirements. Requirements of developers or applications in reasoning support and articulation from UMLS SN will determine the selection between OWL Lite-Basic, OWL DL-Basic and OWL Full-Basic Species.

The OWL Lite-Basic Species offers a basic approach for UMLS SN taxonomy. Easy of comprehension and implementation by application developers is a benefit of OWL Lite-Basic Species, which also have a drawback of limited interpretation. Bontas EP et.al. [18, 19] in both references stated that they mainly utilized the taxonomic hierarchy of semantic types and relations from UMLS SN in OWL Format. This taxonomic hierarchy is included in the generated OWL Lite-Basic Species. Therefore, OWL Lite-Basic Species could be used by Bontas EP et.al. [19] in developing a Semantic Web-driven retrieval system for the 'lung pathology' domain by using ontology-based natural language processing. Bontas EP, et al. [18] could also utilize OWL Lite-Basic Species generated by NetON to carry out a content-driven retrieval and storage of medical images and reports.

The generated OWL DL-Basic Species support application developers requiring utmost articulation from UMLS SN while ensuring inferences are computable. Since inheritance blocking information in UMLS SN is not able to be represented in OWL DL-Basic Species due to the lack of appropriate constructors in the OWL standard, users interested in reasoning applications should be aware that a few deduction outcomes will be false positives. The OWL DL-Basic Species generated by NetON, for instance, could be used by Schulz S. et.al. [22] to generate mapping between the BioTop [23], which is an abstract biomedical model developed by Schulz S. et.al., and UMLS Semantic Network [60]. As the main aim of the authors was mapping process, relations between semantic types in UMLS SN were required to be represented with domain and range axioms in related property descriptions like in BioTop. The OWL DL-Basic documents needed to be customized by using the NetON customization editor or be processed by a simple algorithm to change property constraints with value restrictions representations to domain and range axioms as.

The OWL Full-Basic Species is generated principally for application developers requiring utmost articulation from UMLS SN while ensuring unlimited RDF usage, without assurance of accurate or complete inferences. Jimenez-Ruiz E. [21] could use the OWL Full-Basic Species in their tool developed to extend Protégé-OWL and manage a collection of related ontologies.

In UMLS SN, there are unseen assertions that can be inferred by using inference rules on explicitly specified assertions. However, explicitly declared relations between semantic types are not essentially valid for all the descendants. Deduction outcomes of any OWL reasoners (such as Pellet or Racer [42, 43]) on NetON OWL Sublanguage Species (except OWL Lite) will also include false positives due to the lack of inheritance blocking information on assertions in NetON OWL DL and OWL Full Species. The algorithms of the second dimension consider the inheritance blocking information Blocked (B), Descendant Not Included (DNI) and Defined (D) while executing inference rules. As this cannot be done by any OWL reasoner, the second dimension offers a solution for application developers.

The NetON users should determine the most appropriate OWL Species among first and second dimension documents. OWL Sublanguage (First Dimension) documents will be helpful for those NetON users who only need explicitly defined information in UMLS SN. OWL Extended (Second Dimension) documents will be helpful for those who also need inherited information for descendants of STYs and RLs in addition to explicitly defined information in UMLS SN.

OWL Extended Species (Second Dimension) documents can be used by application developers in two different ways. In one way, developers can directly use OWL Extended Species (Second Dimension) documents, which contain entire valid assertions and their descendants, in their applications. In second way, developers can use OWL Extended Species documents as control files in reasoning applications in order to control and eliminate the false positive results.

The OWL Lite-Extended documents include the entire inherited classification hierarchy of UMLS SN in OWL Lite Format. They are helpful for NetON users who do not know or do not want to deal with reasoning algorithms. The OWL DL/Full Extended documents are generated principally for application developers who do not know or not want to deal with reasoning algorithms but require a machine processable form of UMLS SN information including whole inherited and valid information in OWL DL/Full Format. The OWL DL Extended documents can also be used by applications that use reasoning on UMLS SN information as a control file including all valid relations.

Any generated OWL Species can also be altered by the customization application embedded in NetON. (See section 3.3.5) These altered species can use OWL relaxed primitives for constraint representation. Even such kind of alteration will not compute specific assertions concerning complexity and consistency; they might provide collective operability of databases, reasoning or non-reasoning applications using RDF Schema representation and OWL systems.

The transformation of UMLS SN into OWL Species makes it possible to easily perform semantic actions like selection, classification and comparison. The generated OWL Species can be used in natural language processing (NLP) and text mining to suggest validating relations among entities identified in text. They can also be used in information retrieval and navigation to search and display of UMLS concepts with respect to semantic types and alignment and be used in interoperability to help find similarity among concepts based on similarity in high-level categories. The OWL Species can also be used by NLP applications being developed with semantic web technologies to decide on relationships between mined terms by means of particular semantic models. In addition, generated OWL Species can be used to support software agents and knowledge management.

NetON is designed to be a bridge between the current presentation of UMLS Semantic Network and future technology where information will have exact meaning and can be understood and processed by computers. NetON enables the implementation of semantically manageable and interoperable biomedical systems by generating the OWL Species of UMLS Semantic Network as a component of the proposed semantic layer. OWL Species separates data from formatting and simplifies data sharing, data transport and platform changes. They provide machine-processable semantic models for UMLS SN and can be interpreted by biomedical applications. They are also shareable and reusable in different biomedical applications. Our tool targets accessible semantic content to allow efficient semantic interoperation and integration among these applications.

4.4 Future Work

The RLs between STYs are not necessarily valid for all instances of Metathesaurus concepts assigned to these STYs. That is to say, the RL may or may not be applicable between any particular pair of concepts. Therefore, transforming the data in the UMLS Metathesaurus as instance data in the SN OWL ontology alone won't be adequate for accurate representation of the UMLS Metathesaurus. Due to the complexity of biomedicine, any attempt at modeling like the UMLS SN will encounter the above mentioned problem. Therefore, higher-level modeling primitives (especially relations) in UMLS SN should be customized when using them with Metathesaurus concepts in biomedical applications. These customization efforts are open research areas.

Since OWL 2.0 is backward compatible with OWL 1.0, all OWL 1.0 ontologies are also valid OWL 2 [44]. Nonetheless, advances in both UMLS SN and OWL standard must be tracked to reflect in the NetON algorithms as needed.

REFERENCES

1. Bethesda, M.D. (2009). Unified Medical Language System (UMLS) Reference Manual. U.S. Department of Health and Human Services, National Institutes of Health, National Library of Medicine, NCBI.

2. U.S. Department of Health and Human Services, National Institutes of Health, National Library of Medicine (2009). Unified Medical Language System (UMLS). Retrieved December 30, 2009, from: http://www.nlm.nih.gov/research/umls/

3. U.S. Department of Health and Human Services, National Institutes of Health, National Library of Medicine (2009). Unified Medical Language System (UMLS). Retrieved December 29, 2009, from: http://www.nlm.nih.gov/pubs/techbull/ma09/ma09_umls_2009aa_release.html

4. Rubin, D.L., Lewis, S.E., Mungall C.J., Misra, S., Westerfield, M., Ashburner, M., Sim, I., Chute, C.G., Solbrig, H., Storey, M.A., Smith, B., Day-Richter, J., Noy, N.F. & Musen, M.A. (2006) National Center for Biomedical Ontology: Advancing biomedicine through structured organization of scientific knowledge. OMICS-A Journal of Integrative Biology; 10(2), 185-198.

5. Christopher, D.W. (2007) Agency and the Semantic Web. Oxford: Oxford University Press.

6. Antoniou, G. & Van Harmelen, F. (2004) A Semantic Web Primer, MIT Press.

7. Lee, B., Parr , C.S., Plaisant , C., Bederson , B.B., Veksler , V.D., Gray, W.D. & Kotfila, C. (2006) TreePlus: Interactive Exploration of Networks with Enhanced

Tree Layouts. IEEE Transactions On Visualization And Computer Graphics. Special Issue on Visual Analytics.

8. Berners-Lee, T., Chen, Y., Chilton, L., Connolly, D., Dhanaraj, R., Hollenbach, J., Lerer, A. & Sheets, D. (2006) Tabulator: Exploring and Analyzing linked data on the Semantic Web. The 3rd International Semantic Web User Interaction Workshop, Athens, Georgia, USA.

9. W3C. Extensible Markup Language (XML). Retrieved October 28, 2009, from: http://www.w3.org/XML/.

10. W3C. Extensible Markup Language (XML) 1.0 (Fifth Edition). Retrieved October 15, 2009, from: http://www.w3.org/TR/2008/REC-xml-20081126/.

11. W3C. Extensible Markup Language Schema. Retrieved October 10, 2009, from: http://www.w3.org/XML/Schema.

12. W3C. Resource Description Framework (RDF). Retrieved October 16, 2009, from: http://www.w3.org/RDF/.

13. W3C. RDF Primer. Retrieved October 17, 2009, from: http://www.w3.org/TR/2004/REC-rdf-primer-20040210/.

14. W3C. RDF Vocabulary Description Language 1.0: RDF Schema. Retrieved November 21, 2009, from: http://www.w3.org/TR/rdf-schema/.

15. Bechhofer, S., Van Harmelen, F., Hendler, J., Horrocks, I., McGuinness, D.L., Patel-Schneider, P.F. & Stein, L.A. OWL Web Ontology Language Reference. Retrieved November 20, 2009, from: http://www.w3.org/TR/owl-ref/.

16. W3C. OWL Web Ontology Language Overview. Retrieved November 23, 2009, from: http://www.w3.org/TR/2004/REC-owl-features-20040210/.

17. W3C. OWL Web Ontology Language Guide. Retrieved November 19, 2009, from: http://www.w3.org/TR/2004/REC-owl-guide-20040210/.

18. Bontas, E.P., Tietz, S., Tolksdorf, R. & Schrader, T. (2004) Generation and management of a medical ontology in a Semantic Web retrieval system. On the Move Confederated International Workshop and Conference, CYPRUS, v.3290, 637-653.

19. Bontas, E.P. Tietz, S. & Schrader, T. (2004) Experiences Using Semantic WebTechnologies to Realize an Information Retrieval System for Pathology. In Proc.of the Berliner XML Days Conference.

20. Jimenez-Ruiz, E. The UMLS Semantic Network in OWL. Retrieved November 10, 2009, from: http://krono.act.uji.es/people/Ernesto/UMLS_SN_OWL**.**

21. Jimenez-Ruiz, E., Berlanga, R., Nebot, V. & Sanz, I. (2007) A Protege Plug-in-Based System to Manage and Query Large Domain Ontologies. 10th International Protege Conference.

22. Schulz, S., Beisswanger, E., Hoek, L., Bodenreider, O. & Mulligen, E.M. (2009) Alignment of the UMLS semantic network with BioTop. Bioinformatics; 25(12), 69-76.

23. Beisswanger, E., Schulz, S., Stenzhorn, H., & Hahn, U. (2008). BioTop: an upper domain ontology for the life sciences. A description of its current structure, contents and interfaces to OBO ontologies. Applied Ontology; 3, 205-212.

24. Kashyap, V. & Borgida, A. (2003) Representing the UMLS Semantic Network Using OWL: (Or "What's in a Semantic Web link?"). 2nd International Semantic Web Conference, Sanibel, Florida.

25. Bechhofer, S. & Volz, R. WonderWeb OWL Ontology Validator, EU IST Project, University of Manchester and University of Karlsruhe. Retrieved December 14, 2009, from: http://www.mygrid.org.uk/OWL/Validator

26. National Library of Medicine (NLM), Protégé Retrieved December 11, 2009, from: http://protege.stanford.edu/

27. Noy, N.F., Shah, N.H., Whetzel, P.L., Dai, B., Dorf, M., Griffith, N., Jonquet, C., Rubin, D.L., Storey, M.A., Chute, C.G. & Musen, M.A. (2009) BioPortal:

ontologies and integrated data resources at the click of a mouse. Nucleic Acids Research; 37, 170-173.

28. Prosdocimi, F., Chisham, B., Pontelli, E., Thompson, J.D. & Stoltzfus, A. (2009) Initial Implementation of a Comparative Data Analysis Ontology. Evolutionary Bioinformatics; 5, 47-66**.**

29. Moreira, D.A. & Musen, M.A. (2007) OBO to OWL: a protégé OWL tab to read/save OBO ontologies. Bioinformatics; 23(14), 1868-1870**.**

30. Aitken, S., Chen, Y. & Bard, J. (2008) OBO Explorer: an editor for open biomedical ontologies in OWL. Bioinformatics; 24(3), 443-444**.**

31. National Center for Biomedical Ontologies (NCBO). Mapping OBO in OWL. Retrieved December 12, 2009, from: http://www.bioontology.org/wiki/index.php/OboInOwl:Main_Page.

32. Aitken, S., Chen, Y., Webber, B., Fan, W. & Bard, J. (2007) Managing the transition from OBO to OWL: The COBrA-CT bio-ontology tools. Proc. UK e-Science All-Hands Meeting; 94-101.

33. Van Assem, M., Menken, M. R., Schreiber G., Wielemaker J. & Wielinga, B. (2004) A Method for Converting Thesauri to RDF/OWL. Proceedings of the Third International Semantic Web Conference (ISWC 2004), Hiroshima, Japan.

34. Supekar, K., Chute, C. G. & Solbrig H. (2005) Representing Lexical Components of Medical Terminologies in OWL. AMIA Annual Symposium proceedings, 719-723.

35. Soualmia, L., Golbreich, C. & Darmoni, S. (2004) Representing the MeSH in OWL: Towards a semiautomatic migration. Proceedings of the KR 2004 Workshop on Formal Biomedical Knowledge Representation, 81–87.

36. Sidhu, A.S., Dillon, T.S. & Chang E. (2005) An ontology for protein data models. Conf Proc IEEE Eng Med Biol Soc.; 6:6120-6123.

37. Sattanathan, S., Narendra, N.C. & Maamar, Z. (2005) Ontologies for Specifying and Reconciling Contexts of Web Services. Electronic Notes in Theoretical Computer Science; 146(1), 43-57.

38. Harris, M.A., Clark, J., Ireland, A., Lomax, J., Ashburner, M., Foulger, R., Eilbeck, K., Lewis, S., Marshall, B., Mungall, C., Richter, J., Rubin, G.M., Blake, J.A., Bult, C., Dolan, M., Drabkin, H., Eppig, J.T., Hill, D.P., Ni, L., Ringwald, M., Balakrishnan, R., Cherry, J.M., Christie, K.R., Costanzo, M.C., Dwight, S.S., Engel, S., Fisk,D.G., Hirschman, J.E., Hong, E.L., Nash, R.S., Sethuraman, A., Theesfeld, C.L., Botstein, D., Dolinski, K., Feierbach, B., Berardini, T., Mundodi, S., Rhee, S.Y., Apweiler, R., Barrell, D., Camon, E., Dimmer, E., Lee, V., Chisholm, R., Gaudet, P., Kibbe, W., Kishore, R., Schwarz, E.M., Sternberg, P., Gwinn, M., Hannick, L., Wortman, J., Berriman, M., Wood, V., de la Cruz, N., Tonellato, P., Jaiswal, P., Seigfried, T. & White, R. (2004) The Gene Ontology (GO) database and informatics resource. Nucleic Acids Res.; 32, 258–261.

39. Beckett, D. RDF/XML Syntax Specification. Retrieved December 9, 2009, from: http://www.w3.org/TR/rdf-syntax-grammar/

40. Antoniou, G., Baldoni, M., Baroglio, C., Patti, V., Baumgartner, R., Eiter, T., Herzog, M., Schindlauer, R., Tompits, H., Bry, F., Schaffert, S., Henze, N. & May W. (2004) Reasoning Methods for Personalization on the Semantic Web. Annals of Mathematics, Computing & Teleinformatics; 1(2), 1-24.

41. Costello, R. L. & Jacobs, D. B. (2003) A Quick Introduction to OWL Web Ontology Language. OWL Tutorial from the MITRE Corporation, sponsored by DARPA.

42. Sirin, E., Parsia, B., Grau, B. C., Kalyanpur, A. & Katz, Y. (2007) Pellet: A practical OWL reasoner. Journal Of Web Semantics; 5(2):51-53.

43. Haarslev, V. & Möller, R. (2003) Racer: A Core Inference Engine for the Semantic Web. Proceeding of the 2nd International Workshop on Evaluation of Ontology-based Tools, 27-36.

44. National Library of Medicine (NLM). The UMLS Semantic Network Files, Retrieved December 27, 2009, from: http://semanticnetwork.nlm.nih.gov/Download/index.html.

45. Denecke, K. (2008) Semantic Structuring of and Information Extraction from Medical Documents Using the UMLS. Methods of Information in Medicine; 47, 425-434.

46. Gaudinat, A., Joubert, M., Aymard, S., Falco, L., Boyer, C. & Fieschi, M. (2004) WRAPIN: New Generation Health Search Engine Using UMLS Knowledge Sources for MeSH Term Extraction from Health Documentation. Medinfo; 11(1):356-360.

47. European Project. WRAPIN - Worldwide online Reliable Advice to Patients and Individuals. Retrieved December 21, 2009, from: http://www.wrapin.org/.

48. Chen, Y., Perl, Y., Geller, J., Hripcsak, G. & Zhang, L. (2008) Comparing and consolidating two heuristic metaschemas. Journal Of Biomedical Informatics; 41(2), 293-317.

49. Kumar, A., Smith, B., Stefanelli, M., Quaglinia, S. & Piazza, M. (2003) Implementing clinical guidelines in an organizational setup. Proceedings of the workshop on model-based and qualitative reasoning in biomedicine, p. 39–44.

50. Zhang, L., Hripcsak, G., Perl, Y., Halper, M. & Geller, J. (2005) An expert study evaluating the UMLS lexical metaschema. Artificial Intelligence in Medicine; 34(3):219–233.

51. Sánchez, D. & Moreno A. (2008) Learning non-taxonomic relationships from web documents for domain ontology construction. Data & Knowledge Engineering; 64(3), 600-623.

52. Fellbaum, C. (1998) WordNet: An Electronic Lexical Database, MIT Press.

53. Rajapakse, M., Kanagasabai, R., Tiong Ang, W., Veeramani, A., Schreiber, M. J. & Baker C. J.O. (2008) Ontology-centric integration and navigation of the dengue literature. Journal of Biomedical Informatics, 41(5), 806-815.

54. Chen, L., Morrey, C. P., Gu, H., Halper, M. & Perl, Y. (2009) Modeling Multi-typed Structurally Viewed Chemicals with the UMLS Refined Semantic Network. Journal of the American Medical Informatics Association, 16, 116 –131.

55. Geller, J., Gu, H., Perl, Y. & Halper, M. (2003) Semantic refinement and error correction in large terminological knowledge bases. Data & Knowledge Engineering, 45, 1–32.

56. Golbreich, C. Zhang, S. & Bodenreider, O. (2006) The foundational model of anatomy in OWL: Experience and perspectives. Journal of Web Semantics; 4, 181–195.

57. Chen, Y., Gu, H.H., Perl, Y., Geller, J. & Halper, M. (2009) Structural group auditing of a UMLS semantic type's extent. Journal of Biomedical Informatics; 42(1), 41–52.

58. Humphreys, B. L. & Lindberg, D. A. B. (1998) Schoolman HM and Barnett GO. The Unified Medical Language System: An informatics research collaboration. JAMIA; 5(1), 1–11.

59. McCray, A.T. & Hole, W. T. (November 1990) The scope and structure of the first version of the UMLS Semantic Network. Proceedings of 14th annual SCAMC, Los Alamitos, CA, 126–130.

60. McCray, A. T. (2003) An upper-level ontology for the biomedical domain. Comparative and Functional Genomics, 4, 80-84.

61. McCray, A. T. & Nelson, S. J. (1995) The representation of meaning in the UMLS. Methods of Information in Medicine; 34, 193–201.

62. U.S. National Library of Medicine. (April 2005) The Future of the UMLS Semantic Network Workshop, Bethesda, Maryland. Retrieved November 10, 2009, from: http://mor.nlm.nih.gov/snw/.

63. Perl, Y., Chen, Z., Halper, M., Geller, J., Zhang, L.& Peng, Y. (2003) The cohesive metaschema: a higher-level abstraction of the UMLS Semantic Network. Journal of Biomedical Informatics, 35(3), 194-212.

64. Schuyler, P. L., Hole, W. T., Tuttle, M. S. & Sherertz D. D. (1993) The UMLS Metathesaurus: representing different views of biomedical concepts. Bulletin of the Medical Library Association; 81(2), 217–22.

65. Fan, J. W. & Friedman, C. (2007) Semantic classification of biomedical concepts using distributional similarity. Journal of the American Medical Informatics Association; 14(4): 467–477.

66. U.S. Department of Health and Human Services, National Institutes of Health, National Library of Medicine. Unified Medical Language System (UMLS). Retrieved November 29, 2009, from: www.nlm.nih.gov/research/umls.

67. Perl, Y. & Geller, J. (April 2005) Semantic Network Issues in UMLS Study. The Future of the UMLS Semantic Network Workshop, Bethesda, Maryland. Retrieved November 17, 2009, from: http://mor.nlm.nih.gov/snw/.

68. Batres, R., West, M., Leal, D., Price, D., Masaki, K., Shimada, Y., Fuchino, T. & Naka, Y. (2007) An upper ontology based on ISO 15926. Computers & Chemical Engineering; 31(5-6), 519–534.

69. Lacy, L. W. (2005) OWL: Representing information using the Web ontology language, Trafford Publishing.

70. Fa, J. W., Xu, H. & Friedman, C. (2007) Using contextual and lexical features to restructure and validate the classification of biomedical concepts. BMC Bioinformatics; 8, p.264.

71. U.S. Department of Health and Human Services, National Institutes of Health, National Library of Medicine. Unified Medical Language System (UMLS). Retrieved September 16, 2009, from: http://www.nlm.nih.gov/pubs/techbull/ma09/ma09_umls_2009aa_release.html

72. Horrocks, I. (2002) DAML+OIL: A Description Logic for the Semantic Web. IEEE Bulletin for the Technical Committee on Data Engineering.

73. Antoniou, G. & Van Harmelen F. A. (2004) Semantic Web Primer. MIT Press.

74. Cimino, J. J., Min, H. & Perl, Y. (2003) Consistency across the hierarchies of the UMLS Semantic Network and Metathesaurus. Journal of Biomedical Informatics; 36(6), 450–461.

75. Burgun, A., Botti, G., Fieschi, M. & Le Beux, P. (1999) Sharing knowledge in medicine: semantic and ontologic facets of medical concepts. IEEE Proceeding System, Man and Cybernetics; 300–305.

76. McCray, A. T., Burgun, A. & Bodenreider, O. (2001) Aggregating UMLS semantic types for reducing conceptual complexity. Medinfo, 216–220.

77. Chen, Z., Perl, Y., Halper, M., Geller, J. & Gu, H. (2002) Partitioning the UMLS semantic network. IEEE Transactions on Information Technology in Biomedicine; 6(2), 102–108.

78. Zhang, L., Perl, Y., Halper, M., Geller, J. & Hripcsak, G. A. (2005) A lexical metaschema for the UMLS semantic network. Artificial Intelligence in Medicine; 33(1), 41–59.

79. Gu, H., Perl, Y., Elhanan, G., Min, H., Zhang, L., & Peng, Y. (2004) Auditing concept categorizations in the UMLS. Artificial Intelligence in Medicine; 31(1), 29–44.

80. Cimino, J. J. (1998) Auditing the Unified Medical Language System with semantic methods. Journal of the American Medical Informatics Association; 5(1):41–51.

81. Peng, Y., Halper, M.H., Perl, Y. & Geller, J. (2002) Auditing the UMLS for redundant classifications. Proceedings of AMIA Annual Symposium, 612–616.

82. Martínez-Costa, C., Menárguez-Tortosa, M., Fernández-Breis, J. T. & Maldonado, J. A. (2009) A model-driven approach for representing clinical archetypes for Semantic Web environments. Journal of Biomedical Informatics; 42, 150–164.

83. Berners-Lee, T., Hendler, J. & Lassila, O. (May 2001) The Semantic Web – A new form of Web content that is meaningful to computers will unleash a revolution of new possibilities. Scientic American; 284(5), 34-43.

84. Almendros-Jimenez, J. M. (2008) An RDF Query Language based on Logic Programming. Electronic Notes in Theoretical Computer Science; 200, 67–85.

85. Huang, S. L. & Yang, C. W. (2009) Designing a semantic bliki system to support different types of knowledge and adaptive learning. Computers & Education; 53, 701–712.

86. Aranguren, M.E., Bechhofer, S., Lord, P., Sattler, U. & Stevens, R. (2007) Understanding and using the meaning of statements in a bio-ontology: recasting the Gene Ontology in OWL. BMC Bioinformatics; 8(57).

87. Goodwin, J. (2005) Experiences of using OWL at the Ordnance Survey. Proceeding of OWL Experiences and Directions.

88. Schulz, S. & Hahn, U. (2005) Part-whole representation and reasoning in formal biomedical ontologies. Artificial Intelligence in Medicine; 34(3), 179–200.

89. Smith, B. (2006) From concepts to clinical reality: an essay on the benchmarking of biomedical terminologies. Journal of Biomedical Informatics ; 39(3), 288–298.

90. Ashburner, M., Ball, C.A., Blake, J.A., Botstein, D., Butler, H., Cherry, J.M., Davis, A.P., Dolinski, K., Dwight, S.S., Eppig, J.T., Harris, M.A., Hill, D.P., Issel-Tarver, L., Kasarskis, A., Lewis, S., Matese, J.C., Richardson, J.E., Ringwald, M., Rubin, G.M. & Sherlock, G. (2000) Gene ontology: tool for the unification of biology. Nature Genetics; 25, 25–29.

91. OBO.The Open Biomedical Ontologies; Retrieved December 16, 2009, from: http://www.obofoundry.org/.

92. Gruber, T. R. (1993) A translation approach to portable ontology specifications. Knowledge Acquisition; 5, 199–220.

93. Van Heijst, G., Schreiber, A. T. & Wielinga, B. J. (1997) Using explicit ontologies in KBS development. International Journal of Human-Computer Studies; 45, 183–292.

94. Brewster, C., O'Hara, K., Fuller, S., Wilks, Y., Franconi, E., Musen, M.A., Ellman, J. & Buckingham Shum, S. (2004) Knowledge representation with ontologies: the present and future. IEEE Intelligent Systems; 19(1), 72–81.

95. Fernández-Breis, J. T. & Martínez-Béjar, R. (2002) A cooperative framework for integrating ontologies. International Journal of Human-Computer Studies; 56(6), 662–717.

96. Zanstra, P., Stroetmann, V. N. & Rodrigues, J. M. (2007) Conceptual Framework for eHealth Interoperability. Semantic Health International Project; Retrieved December 2, 2009, from: http://www.semantichealth.org/DELIVERABLES/SemanticHEALTH_D1_1_finalC.pdf

97. Paterson, G. I. (2004) Semantic Interoperability for Decision Support Using Case Formalism and Controlled Vocabulary. Health'04.

98. Semantic Interoperability Community of Practice (SICoP). (2005) Introducing Semantic Technologies and the Vision of the Semantic Web. White Paper Series Module 1.

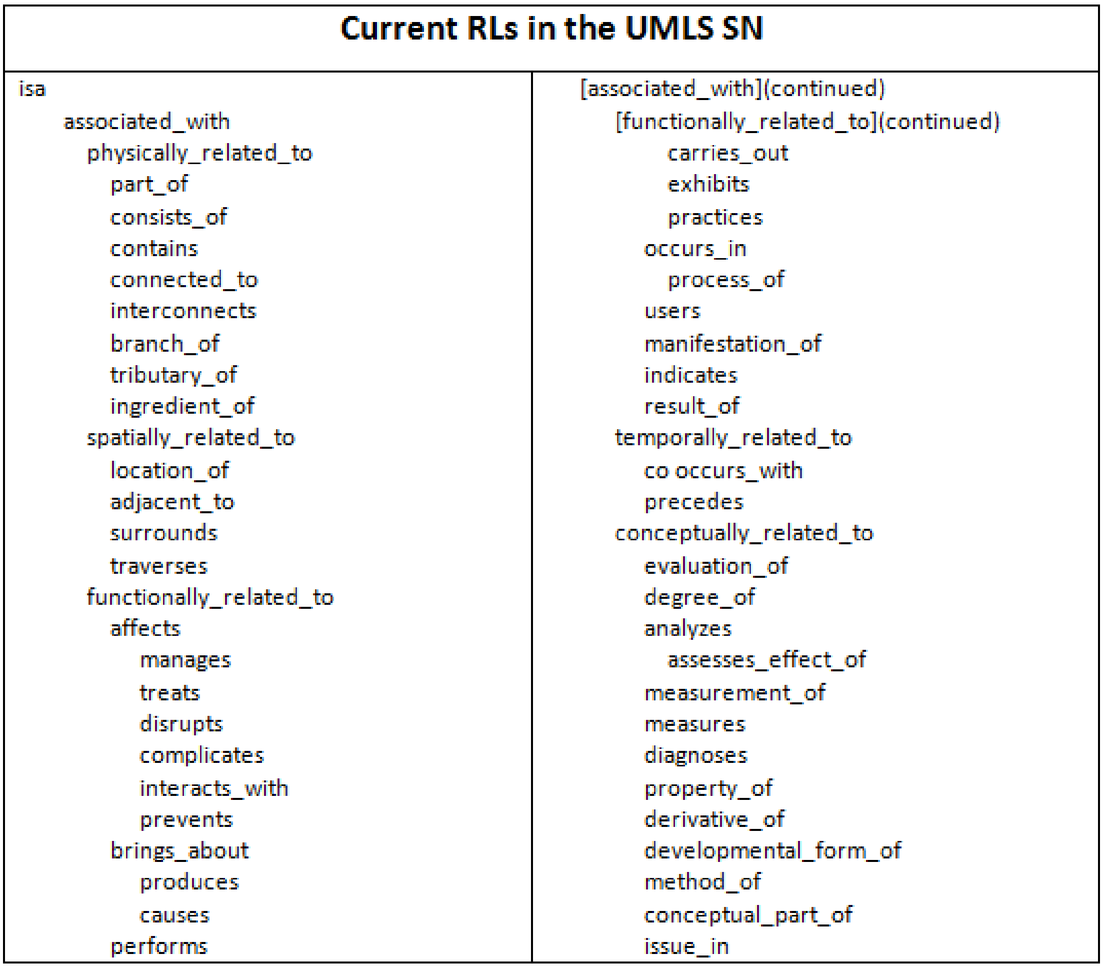
99. ONTOLOG. Collaborative Work Environment. Retrieved December 12, 2009, from: http://ontolog.cim3.net/.

100. Noy, N. F., & Rubin, D. L. (2008) Translating the Foundational Model of Anatomy into OWL. Web Semantics: Science, Services and Agents on the World Wide Web, 6(2), 133-136.

101. Abu-Hanna, A., Cornet, R., Keizer, N., Crubézy, M. & Tu, S. (2005) Protégé as a vehicle for developing medical terminological systems. International Journal of Human-Computer Studies, 62(5), 639-663.

APPENDICES

APPENDIX A. Semantic Relations in UMLS Semantic Network [1]



Current RLs in the UMLS SN

- isa
  - associated_with
    - physically_related_to
      - part_of
      - consists_of
      - contains
      - connected_to
      - interconnects
      - branch_of
      - tributary_of
      - ingredient_of
    - spatially_related_to
      - location_of
      - adjacent_to
      - surrounds
      - traverses
    - functionally_related_to
      - affects
        - manages
        - treats
        - disrupts
        - complicates
        - interacts_with
        - prevents
      - brings_about
        - produces
        - causes
      - performs

- [associated_with](continued)
  - [functionally_related_to](continued)
        - carries_out
        - exhibits
        - practices
      - occurs_in
        - process_of
      - users
      - manifestation_of
      - indicates
      - result_of
  - temporally_related_to
      - co occurs_with
      - precedes
  - conceptually_related_to
      - evaluation_of
      - degree_of
      - analyzes
        - assesses_effect_of
      - measurement_of
      - measures
      - diagnoses
      - property_of
      - derivative_of
      - developmental_form_of
      - method_of
      - conceptual_part_of
      - issue_in

APPENDIX B. Entity Semantic Types in UMLS Semantic Network [1]

ENTITY

- Physical Object
  - Organism
    - Plant
      - Alga
    - Fungus
    - Virus
    - Rickettsia or Chlamydia
    - Bacterium
    - Archaeon
    - Animal
      - Invertebrate
      - Vertebrate
        - Amphibian
        - Bird
        - Fish
        - Reptile
        - Mammal
          - Human
  - Anatomical Structure
    - Embryonic Structure
    - Anatomical Abnormality
      - Congenital Abnormality
      - Acquired Abnormality
    - Fully Formed Anatomical Structure
      - Body Part, Organ, or Organ Component
      - Tissue
      - Cell
      - Cell Component
      - Gene or Genome
  - Manufactured Object
    - Medical Device
    - Research Device
    - Clinical Drug
  - Substance
    - Chemical
      - Chemical Viewed Functionally
        - Pharmacologic Substance
          - Antibiotic
        - Biomedical or Dental Material
        - Biologically Active Substance
          - Neuroreactive Substance or Biogenic Amine
          - Hormone
          - Enzyme
          - Vitamin
          - Immunologic Factor
          - Receptor
        - Indecator, Reagent, or Diagnostic Acid
        - Hazardous or Poisonous Substance
      - Chemical Viewed Structurally
        - Organic Chemical
          - Nucleic Acid, Nucleoside, or Nucleotide
          - Organophosphorus Compound
          - Amino Acid, Peptide, or Protein
          - Carbohydrate
          - Lipid
            - Steroid
            - Eicosanoid
        - Inorganic Chemical
        - Element, Ion, or Isotope
    - Body Substance
    - Food
- Conceptual Entity
  - Idea or Concept
    - Temporal Concept
    - Qualitative Concept
    - Quantitative Concept
    - Functional Concept
      - Body System
    - Spatial Concept
      - Body Space or Junction
      - Body Location or Region
      - Molecular Sequence
        - Nucleotide Sequence
        - Amino Acid Sequence
        - Carbohydrate Sequence
      - Geographic Area
  - Finding
    - Laboratory or Test Result
    - Sign or Symptom
  - Organism Attribute
    - Clinical Attribute
  - Intellectual Product
    - Classification
    - Regulation or Law
  - Language
  - Occupation or Discipline
    - Biomedical Occupation or Discipline
  - Organization
    - Health Care Related Organization
    - Professional Society
    - Self-help or Relief Organization
  - Group Attribute
  - Group
    - Professional or Occupational Group
    - Population Group
    - Family Group
    - Age Group
    - Patient or Disabled Group

APPENDIX C. Event Semantic Types in UMLS Semantic Network [1]

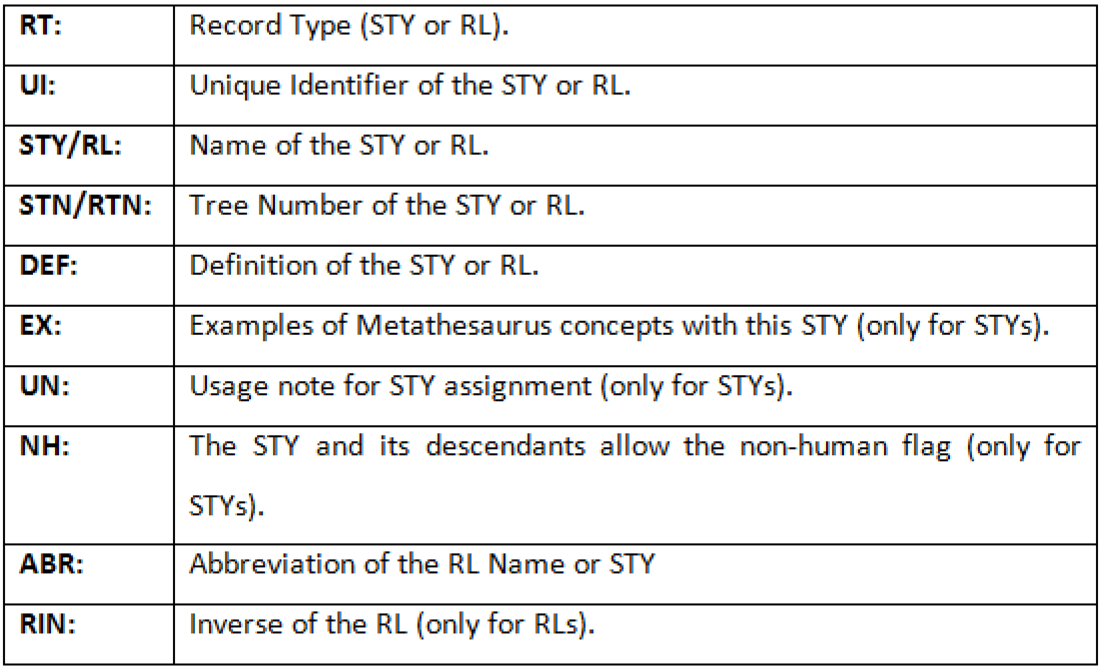
EVENT

- Activity
  - Behavior
    - Social Behavior
    - Individual Behavior
  - Daily or Recreational Activity
  - Occupational Activity
    - Health Care Activity
      - Laboratory Procedure
      - Diagnostic Procedure
      - Therapeutic or Preventive Procedure
    - Research Activity
      - Molecular Biology Research Technique
    - Governmental or Regulatory Activity
    - Educational Activity
  - Machine Activity
- Phenomenon or Process
  - Human-caused Phenomenon or Process
    - Environmental Effect of Humans
  - Natural Phenomenon or Process
    - Biologic Function
      - Physiologic Function
        - Organism Function
          - Mental Process
        - Organ or Tissue Function
        - Cell Function
        - Molecular Function
          - Genetic Function
      - Pathologic Function
        - Disease or Syndrome
          - Mental or Behavioral Dysfunction
          - Neoplastic Process
        - Cell or Molecular Dysfunction
        - Experimental Model or Disease
  - Injury or Poisoning

APPENDIX D. Specific Descriptions of ASCII Relational Format Tables [1]

SRDEF File

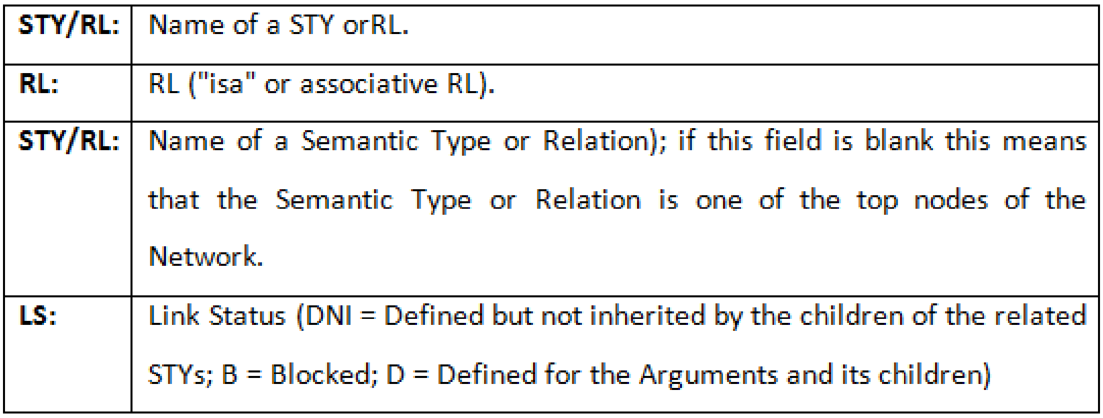
Table 24 SRDEF specific description



RT:	Record Type (STY or RL).
UI:	Unique Identifier of the STY or RL.
STY/RL:	Name of the STY or RL.
STN/RTN:	Tree Number of the STY or RL.
DEF:	Definition of the STY or RL.
EX:	Examples of Metathesaurus concepts with this STY (only for STYs).
UN:	Usage note for STY assignment (only for STYs).
NH:	The STY and its descendants allow the non-human flag (only for STYs).
ABR:	Abbreviation of the RL Name or STY
RIN:	Inverse of the RL (only for RLs).

SRSTR File

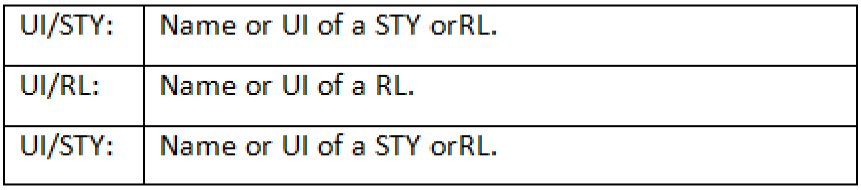
Table 25 SRSTR specific description



STY/RL:	Name of a STY orRL.
RL:	RL ("isa" or associative RL).
STY/RL:	Name of a Semantic Type or Relation); if this field is blank this means that the Semantic Type or Relation is one of the top nodes of the Network.
LS:	Link Status (DNI = Defined but not inherited by the children of the related STYs; B = Blocked; D = Defined for the Arguments and its children)

SRSTRE1 or SRSTRE2 Files

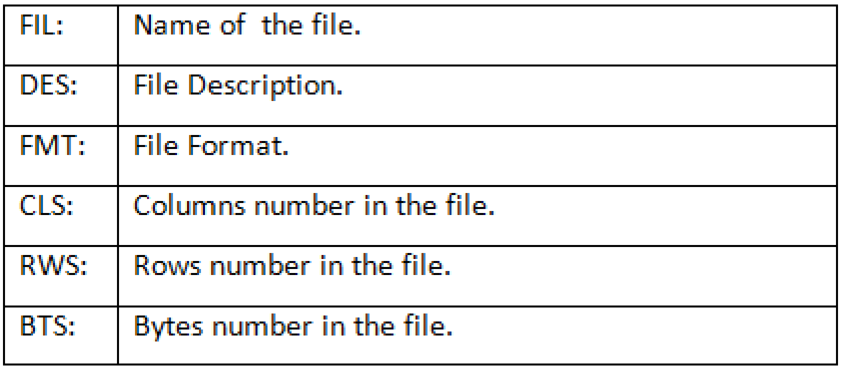
Table 26 SRSTRE1 or SRSTRE2 specific description



UI/STY:	Name or UI of a STY orRL.
UI/RL:	Name or UI of a RL.
UI/STY:	Name or UI of a STY orRL.

SRFIL File

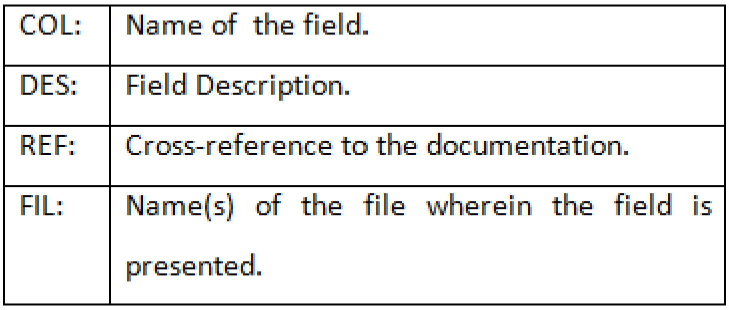
Table 27 SRFIL specific description



FIL:	Name of the file.
DES:	File Description.
FMT:	File Format.
CLS:	Columns number in the file.
RWS:	Rows number in the file.
BTS:	Bytes number in the file.

SRFLD File

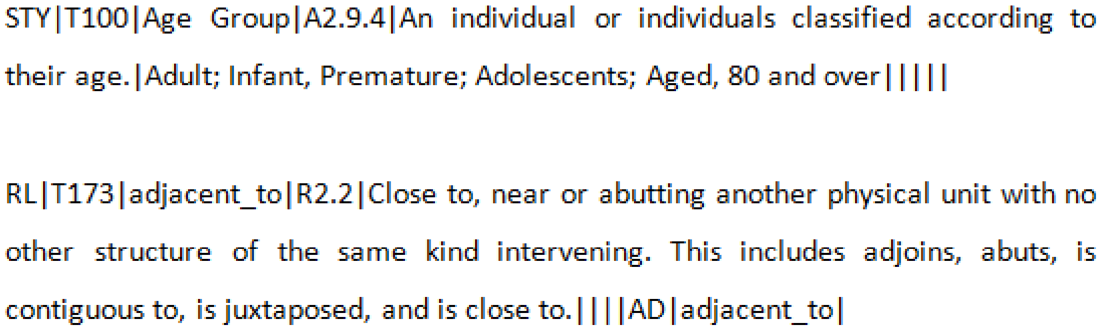
Table 28 SRFLD specific description



COL:	Name of the field.
DES:	Field Description.
REF:	Cross-reference to the documentation.
FIL:	Name(s) of the file wherein the field is presented.

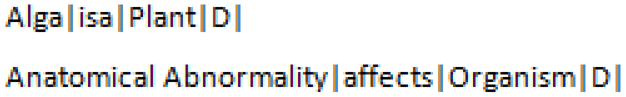
APPENDIX E. Sample Relational Records [44]

SRDEF File



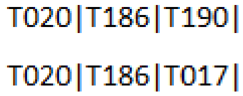
```
STY|T100|Age Group|A2.9.4|An individual or individuals classified according to their age.|Adult; Infant, Premature; Adolescents; Aged, 80 and over|||||

RL|T173|adjacent_to|R2.2|Close to, near or abutting another physical unit with no other structure of the same kind intervening. This includes adjoins, abuts, is contiguous to, is juxtaposed, and is close to.||||AD|adjacent_to|
```

SRSTR File



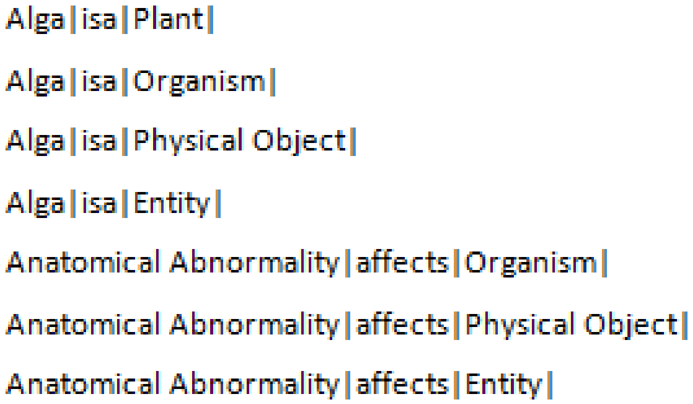
```
Alga|isa|Plant|D|
Anatomical Abnormality|affects|Organism|D|
```

SRSTRE1 File



```
T020|T186|T190|
T020|T186|T017|
```

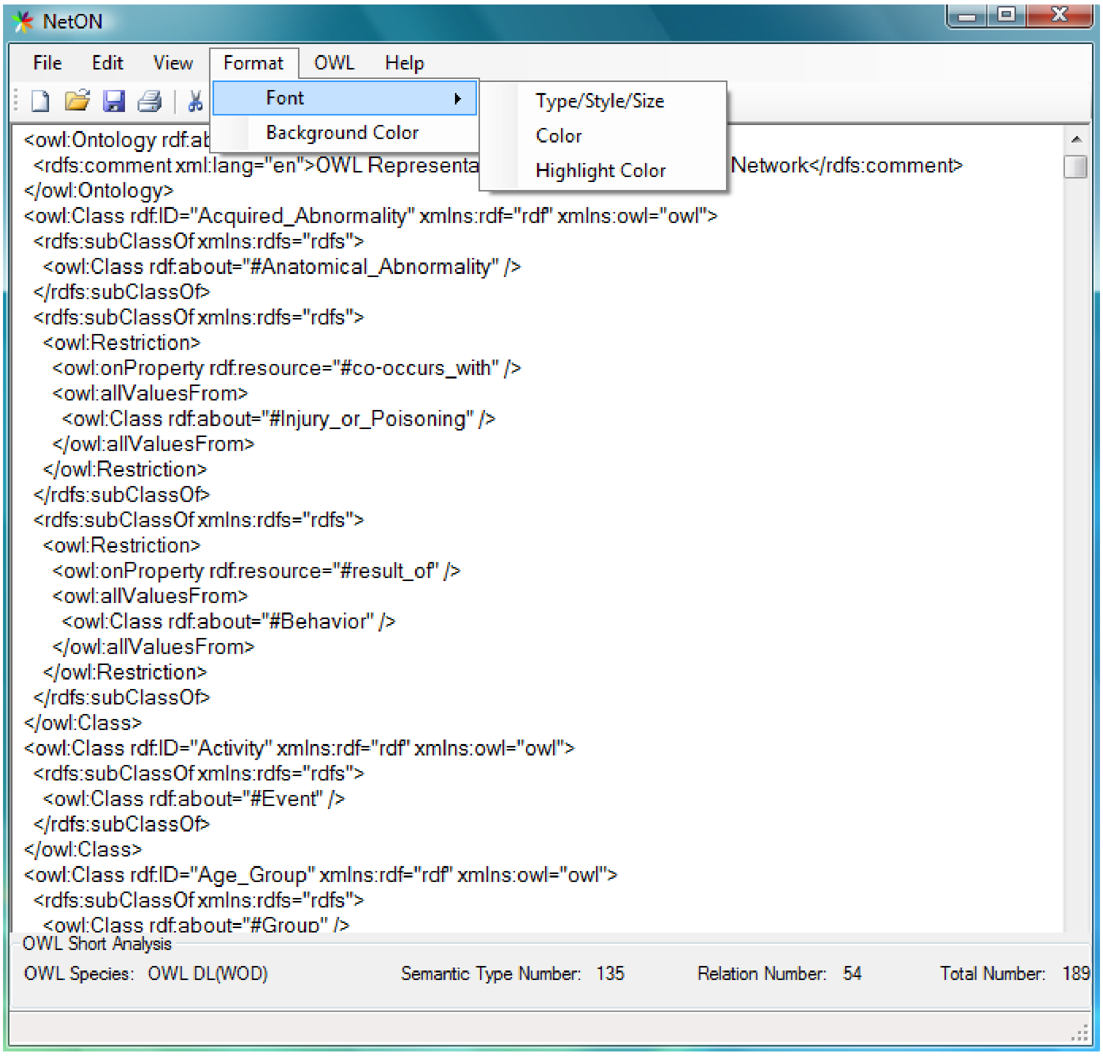
SRSTRE2 File



```
Alga|isa|Plant|
Alga|isa|Organism|
Alga|isa|Physical Object|
Alga|isa|Entity|
Anatomical Abnormality|affects|Organism|
Anatomical Abnormality|affects|Physical Object|
Anatomical Abnormality|affects|Entity|
```

APPENDIX F. Customization Details of NetON OWL Species

While working on customization of OWL Species, the embedded basic text application allows developers to change the font type, style, size, color. Subsequent to selecting the interested part of the document, in order to change the font type, style and size the Type/Style/Size sub-item needed to be chosen from Font menu item under the Format menu. (Figure 26) After the selection, the Font dialog box is popped up which also permits the strikeout and underline options for font customization.



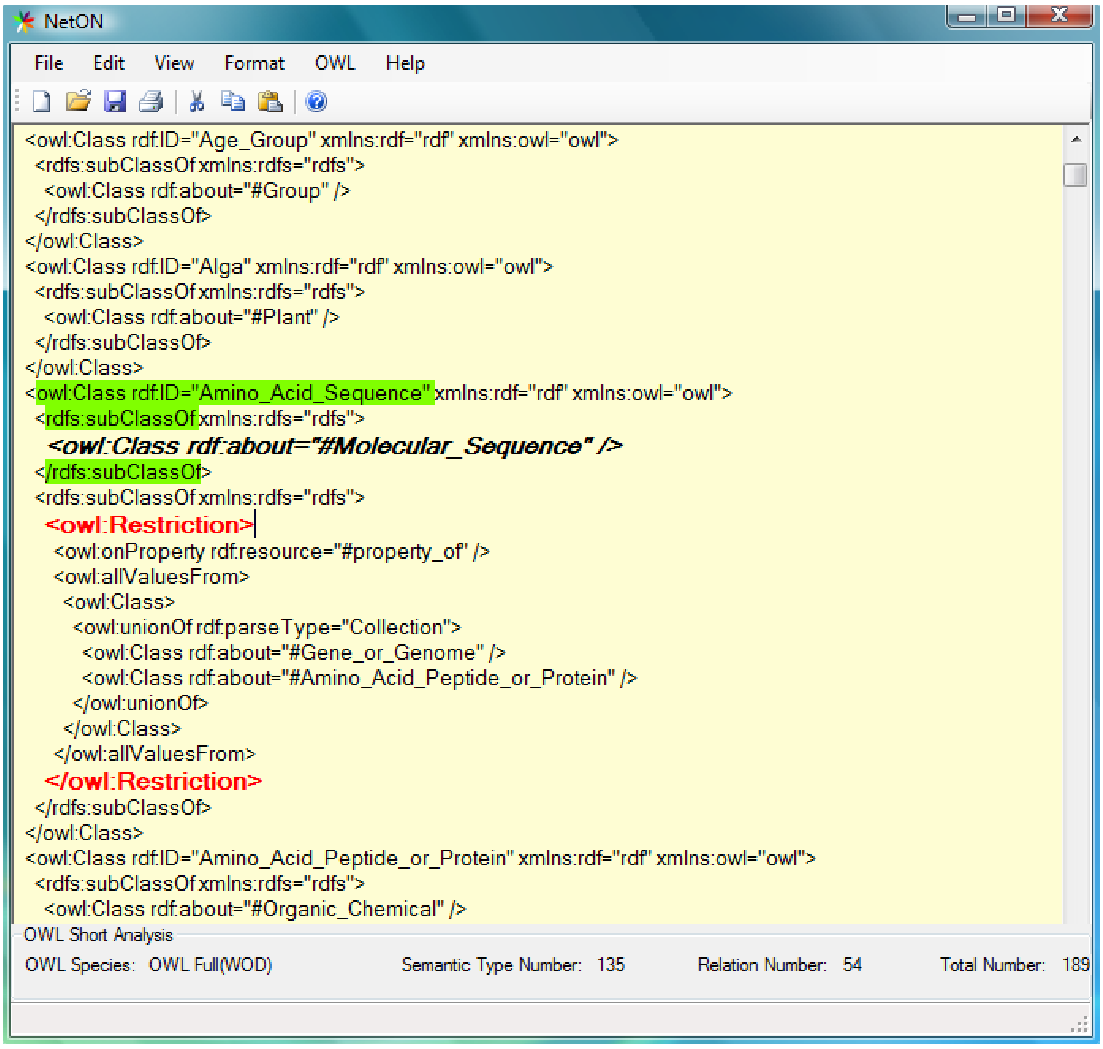
```
<owl:Ontology rdf:ab
  <rdfs:comment xml:lang="en">OWL Representa                    Network</rdfs:comment>
</owl:Ontology>
<owl:Class rdf:ID="Acquired_Abnormality" xmlns:rdf="rdf" xmlns:owl="owl">
  <rdfs:subClassOf xmlns:rdfs="rdfs">
    <owl:Class rdf:about="#Anatomical_Abnormality" />
  </rdfs:subClassOf>
  <rdfs:subClassOf xmlns:rdfs="rdfs">
    <owl:Restriction>
      <owl:onProperty rdf:resource="#co-occurs_with" />
      <owl:allValuesFrom>
        <owl:Class rdf:about="#Injury_or_Poisoning" />
      </owl:allValuesFrom>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf xmlns:rdfs="rdfs">
    <owl:Restriction>
      <owl:onProperty rdf:resource="#result_of" />
      <owl:allValuesFrom>
        <owl:Class rdf:about="#Behavior" />
      </owl:allValuesFrom>
    </owl:Restriction>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="Activity" xmlns:rdf="rdf" xmlns:owl="owl">
  <rdfs:subClassOf xmlns:rdfs="rdfs">
    <owl:Class rdf:about="#Event" />
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="Age_Group" xmlns:rdf="rdf" xmlns:owl="owl">
  <rdfs:subClassOf xmlns:rdfs="rdfs">
    <owl:Class rdf:about="#Group" />
```

OWL Short Analysis

OWL Species: OWL DL(WOD)   Semantic Type Number: 135   Relation Number: 54   Total Number: 189

Figure 26 Format menu items and sub-items

The embedded basic text application also allows developers to change the font color and highlight the preferred text. In order to change the font color, subsequent to selecting the interested part of the document, the Color sub-item needed to be chosen from Font menu item under the Format menu. The selected part of the document can also be highlighted with color by choosing Highlight Color sub-item from Font menu item under the Format menu. After either selection mentioned above, the Color dialog box is popped up for font color or highlight color customization. The developers can also define custom colors if the offered colors are not the ones that they want by clicking the “Define Custom Colors” button in order to pop up Color dialog box with Custom Colors.



```
<owl:Class rdf:ID="Age_Group" xmlns:rdf="rdf" xmlns:owl="owl">
 <rdfs:subClassOf xmlns:rdfs="rdfs">
  <owl:Class rdf:about="#Group" />
 </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="Alga" xmlns:rdf="rdf" xmlns:owl="owl">
 <rdfs:subClassOf xmlns:rdfs="rdfs">
  <owl:Class rdf:about="#Plant" />
 </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="Amino_Acid_Sequence" xmlns:rdf="rdf" xmlns:owl="owl">
 <rdfs:subClassOf xmlns:rdfs="rdfs">
  <owl:Class rdf:about="#Molecular_Sequence" />
 </rdfs:subClassOf>
 <rdfs:subClassOf xmlns:rdfs="rdfs">
  <owl:Restriction>
   <owl:onProperty rdf:resource="#property_of" />
   <owl:allValuesFrom>
    <owl:Class>
     <owl:unionOf rdf:parseType="Collection">
      <owl:Class rdf:about="#Gene_or_Genome" />
      <owl:Class rdf:about="#Amino_Acid_Peptide_or_Protein" />
     </owl:unionOf>
    </owl:Class>
   </owl:allValuesFrom>
  </owl:Restriction>
 </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="Amino_Acid_Peptide_or_Protein" xmlns:rdf="rdf" xmlns:owl="owl">
 <rdfs:subClassOf xmlns:rdfs="rdfs">
  <owl:Class rdf:about="#Organic_Chemical" />
```

OWL Short Analysis

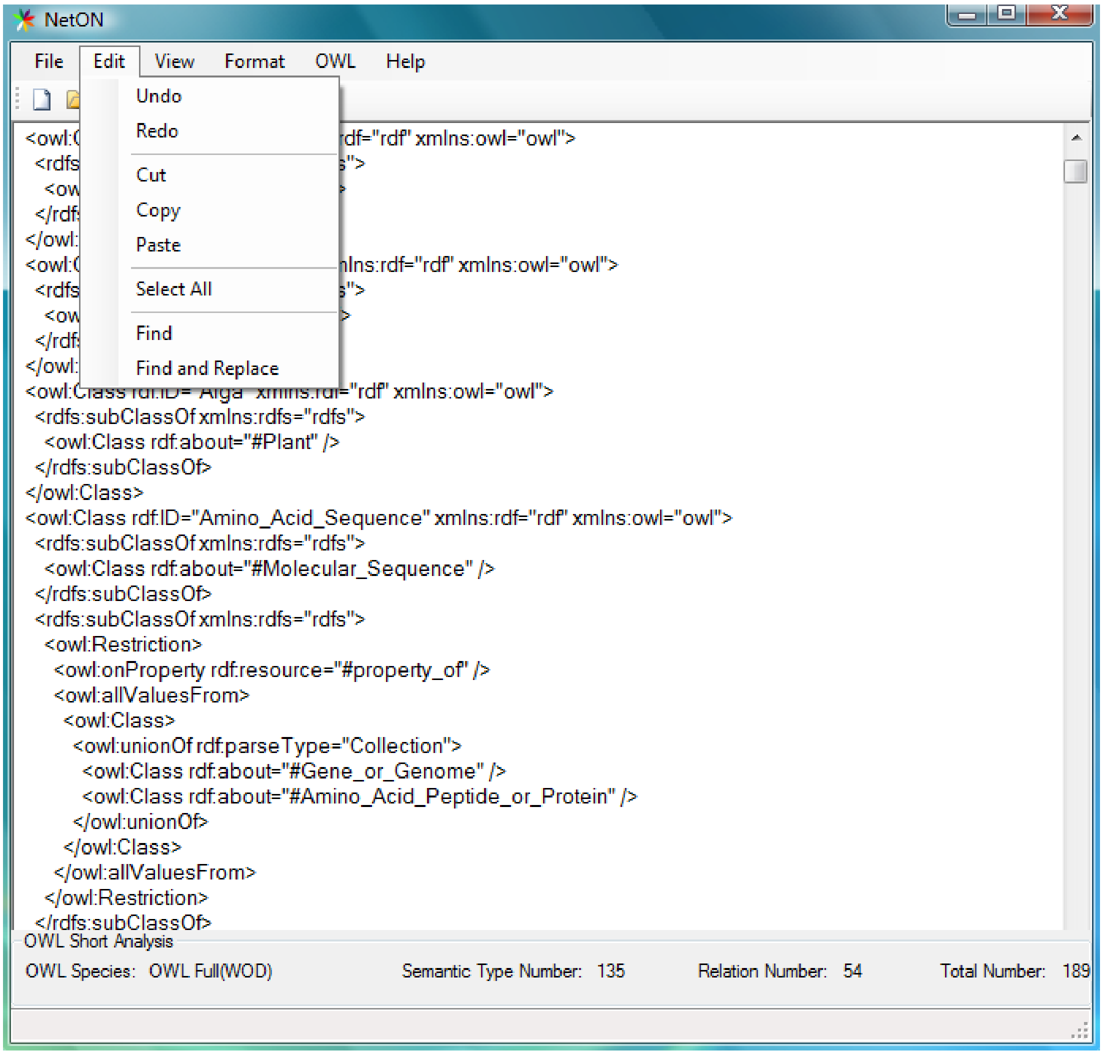
OWL Species: OWL Full(WOD) Semantic Type Number: 135 Relation Number: 54 Total Number: 189

Figure 27 An example for format customization of OWL Species

The embedded basic text application also allows developers to change the background color. In order to change the background color, the Background Color item needed to be chosen under the Format menu. After choosing related menu item, the Color dialog box is popped up for background customization. The developers can also define custom colors if the offered colors are not the ones that they want by clicking the “Define Custom Colors” button in order to pop up Custom Colors dialog box. Figure 27 shows us an example of format customization in font type, style, size, color changes and highlighting and also in background color of the generated OWL Species document.

The developers can also edit the interested document by the help of the embedded basic text application. To cut the part of the document so developers can move it to another location, subsequent to selection on the Edit menu Cut should be clicked. To copy the part of the document so developers can paste it in another location, subsequent to selection on the Edit menu Copy should be clicked. To paste the part of the document that developers have cut or copied the cursor needs to be placed where they want to paste the selected part, and then on the Edit menu Paste should be clicked. To delete the selected part of the document, subsequent to selection on the Edit menu Delete should be clicked. To undo the last action(s) on the Edit menu Undo should be clicked. To redo the undone action(s) on the Edit menu Redo should be clicked. Select All under Edit menu allows to select whole document. (Figure 28)

While working on customization of OWL Species, the embedded basic text application also allows developers to find and replace the interested part of the document. In order to find specific characters or words Find menu item under the Format menu is needed to be chosen. (Figure 28) This selection pops up the Find dialog box. In textbox following the Find what label the characters or words that users want to find are typed. Then, Find Next button should be clicked. To find and replace specific characters or words on the Edit menu Find and Replace should be clicked. In textbox following the Find what label the characters or words that users want to find are typed. In textbox following the Replace with label the replacement text is typed. To replace all instances of the text at once, Replace All should be clicked. Or, to replace one instance at a time, first Find Next is clicked, and then Replace should be clicked.



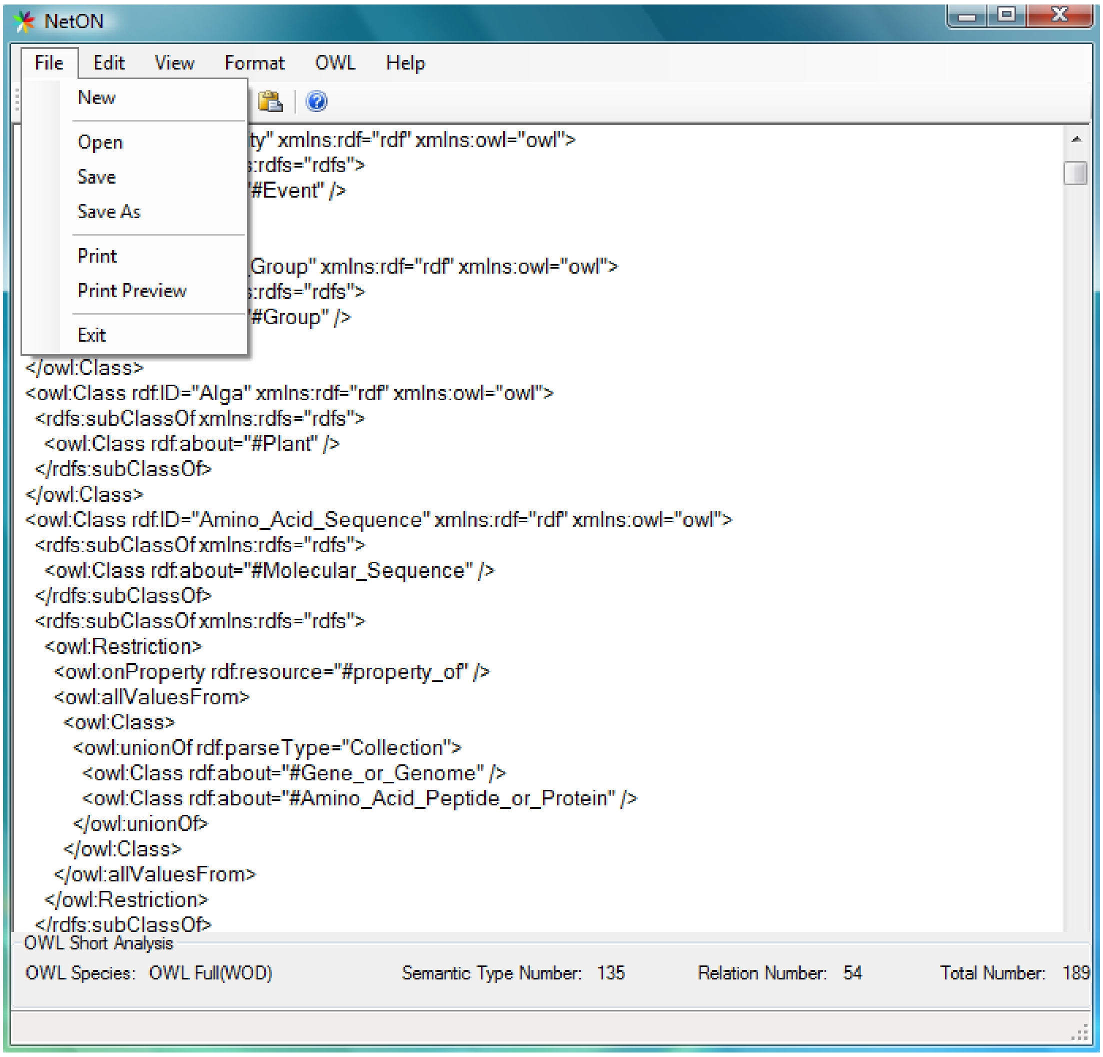
```
<owl:(                              df="rdf" xmlns:owl="owl">
 <rdfs                              s">
  <ow                               >
 </rdf
</owl:
<owl:(                              nlns:rdf="rdf" xmlns:owl="owl">
 <rdfs                              s">
  <ow                               >
 </rdf
</owl:
<owl:Class rdf:ID="Alga" xmlns:rdf="rdf" xmlns:owl="owl">
 <rdfs:subClassOf xmlns:rdfs="rdfs">
  <owl:Class rdf:about="#Plant" />
 </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="Amino_Acid_Sequence" xmlns:rdf="rdf" xmlns:owl="owl">
 <rdfs:subClassOf xmlns:rdfs="rdfs">
  <owl:Class rdf:about="#Molecular_Sequence" />
 </rdfs:subClassOf>
 <rdfs:subClassOf xmlns:rdfs="rdfs">
  <owl:Restriction>
   <owl:onProperty rdf:resource="#property_of" />
   <owl:allValuesFrom>
    <owl:Class>
     <owl:unionOf rdf:parseType="Collection">
      <owl:Class rdf:about="#Gene_or_Genome" />
      <owl:Class rdf:about="#Amino_Acid_Peptide_or_Protein" />
     </owl:unionOf>
    </owl:Class>
   </owl:allValuesFrom>
  </owl:Restriction>
 </rdfs:subClassOf>
```

OWL Short Analysis

OWL Species: OWL Full(WOD)   Semantic Type Number: 135   Relation Number: 54   Total Number: 189

Figure 28 Edit menu items

The developers can also create, open, save and print the customized OWL Species documents by the aid of the embedded basic text application. (Figure 29) To create a new document on the File menu, New menu item should be clicked. This makes the embedded application to open a blank document. Then, the users can begin typing. They can either take notes on their works or write their own OWL document. While activating New menu item under the File menu, the application controls if there is any document that the user is working on. If the answer is positive, then the Caution dialog box is popped up in order to allow the user to save the document.



```
ty" xmlns:rdf="rdf" xmlns:owl="owl">
:rdfs="rdfs">
#Event" />

Group" xmlns:rdf="rdf" xmlns:owl="owl">
:rdfs="rdfs">
#Group" />
</owl:Class>
<owl:Class rdf:ID="Alga" xmlns:rdf="rdf" xmlns:owl="owl">
 <rdfs:subClassOf xmlns:rdfs="rdfs">
  <owl:Class rdf:about="#Plant" />
 </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="Amino_Acid_Sequence" xmlns:rdf="rdf" xmlns:owl="owl">
 <rdfs:subClassOf xmlns:rdfs="rdfs">
  <owl:Class rdf:about="#Molecular_Sequence" />
 </rdfs:subClassOf>
 <rdfs:subClassOf xmlns:rdfs="rdfs">
  <owl:Restriction>
   <owl:onProperty rdf:resource="#property_of" />
   <owl:allValuesFrom>
    <owl:Class>
     <owl:unionOf rdf:parseType="Collection">
      <owl:Class rdf:about="#Gene_or_Genome" />
      <owl:Class rdf:about="#Amino_Acid_Peptide_or_Protein" />
     </owl:unionOf>
    </owl:Class>
   </owl:allValuesFrom>
  </owl:Restriction>
 </rdfs:subClassOf>
```

OWL Short Analysis

OWL Species: OWL Full(WOD)   Semantic Type Number: 135   Relation Number: 54   Total Number: 189

Figure 29 File menu items

To open a document on the File menu, Open menu item should be clicked. This makes the embedded application to pop up Open dialog box. Then, the users can begin typing. They can either take notes on their works or write their own OWL document. In Look in, the drive should be clicked that contains the document they want to open. After locating and selecting the document, Open button should be clicked. If users do not see the document they want, they can click a different file type in Files of type. For example, clicking All Files will allow users to see .txt, .rdf, and .owl type of files on the list. Or users can open a document they opened recently by clicking its name on the File menu.

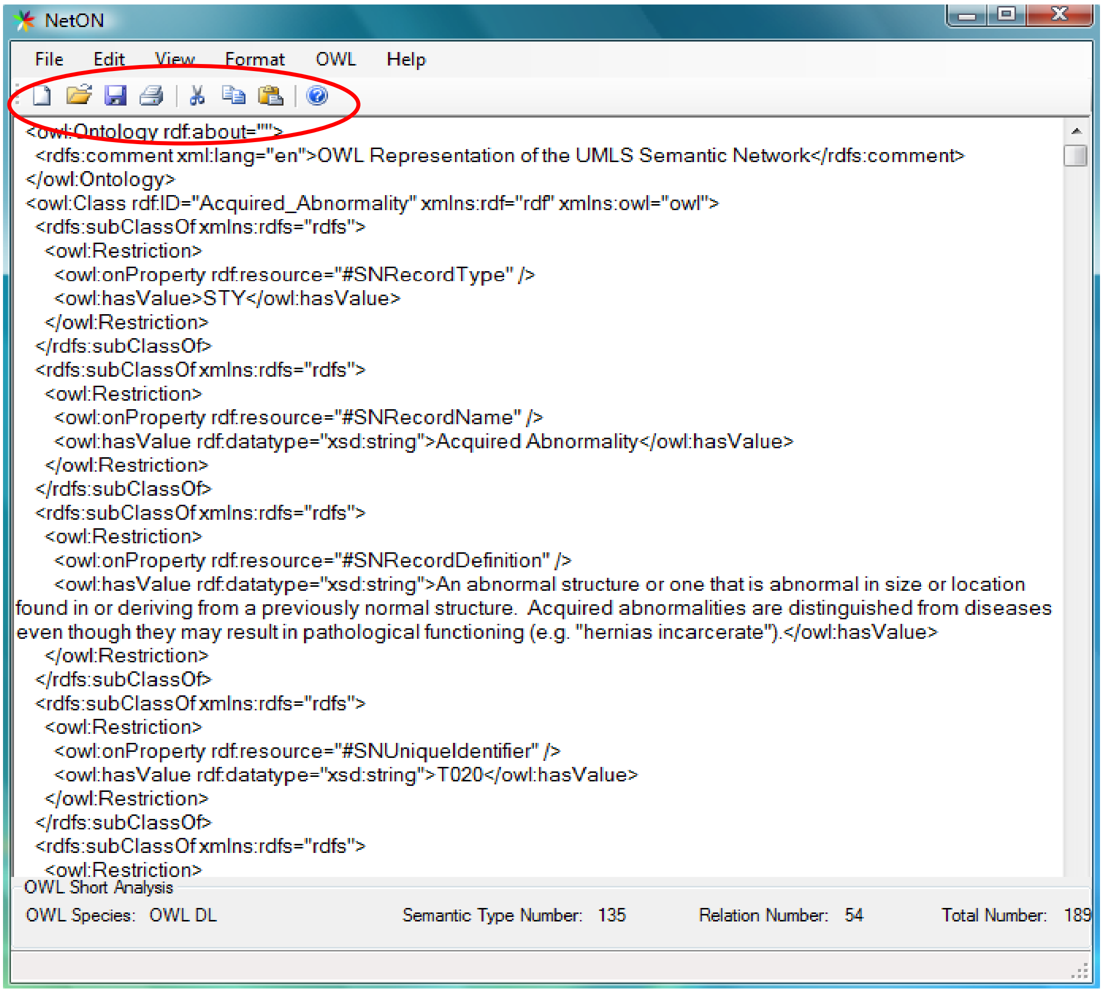
To save change to a document on the File menu, Save menu item should be clicked. While activating Save menu item, the application controls if there is any document that the user is working on. If the answer is positive, then the application controls whether the document is saved before or not. If it is saved before, the embedded application saves the document to the same place. If it is not saved before, the application popped up Save as dialog box in order to allow the user to save the document. To save the file with a new name, on the File menu, Save As should be selected; a different name in File name is typed, and then Save button is clicked. The embedded simple text editor automatically saves documents as text files, but user can change the default file type at any time. After clicking Save As on the File menu, the document format can be selected that users want to set as the default in Save as type dropdown list. For example, to always save documents as OWL files, OWL Document is selected, and then Save button should be clicked. This saves the current document in the selected file format and sets the default file format for future documents until users change it again.

The developers can also print the customized OWL Species documents by the aid of the embedded basic text application. To print a document on the File menu, Print menu item should be chosen. This makes the embedded application to pop up Print dialog box. On the General tab, the printer and preferences users want are selected, and then Print button should be clicked. Users must have a default printer set in the Printers folder. To see how the document will look before printing it, on the File menu, Print Preview can be clicked. Then, Print Preview dialog box is popped up. In the dialog box, user can click Close to return to the document.

The developers can close the NetON by clicking the Exit menu item under the File Menu. While activating Exit menu item, the application controls if there is any document that the user is working on. If the answer is positive, then the Caution dialog box is popped up in order to allow the user to save the document.

The embedded basic text application also gives brief information on ontology, OWL, NetON and how to use our tool to help developers. In order to pop up the Help window, the Help menu needed to be chosen. When using the Help feature in our tool, the Help window appears in a center location. User can then change the way the window is displayed.

While allowing developers to customize the generated OWL Species, the embedded application also offers the most commonly used functionality and features on tool stript menu buttons (Figure 30). File open, new, save, print and cut, copy, paste the selected text. The developers can use them to manage the modified OWL documents quickly and effectively.



```
<owl:Ontology rdf:about="">
  <rdfs:comment xml:lang="en">OWL Representation of the UMLS Semantic Network</rdfs:comment>
</owl:Ontology>
<owl:Class rdf:ID="Acquired_Abnormality" xmlns:rdf="rdf" xmlns:owl="owl">
  <rdfs:subClassOf xmlns:rdfs="rdfs">
    <owl:Restriction>
      <owl:onProperty rdf:resource="#SNRecordType" />
      <owl:hasValue>STY</owl:hasValue>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf xmlns:rdfs="rdfs">
    <owl:Restriction>
      <owl:onProperty rdf:resource="#SNRecordName" />
      <owl:hasValue rdf:datatype="xsd:string">Acquired Abnormality</owl:hasValue>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf xmlns:rdfs="rdfs">
    <owl:Restriction>
      <owl:onProperty rdf:resource="#SNRecordDefinition" />
      <owl:hasValue rdf:datatype="xsd:string">An abnormal structure or one that is abnormal in size or location
found in or deriving from a previously normal structure.  Acquired abnormalities are distinguished from diseases
even though they may result in pathological functioning (e.g. "hernias incarcerate").</owl:hasValue>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf xmlns:rdfs="rdfs">
    <owl:Restriction>
      <owl:onProperty rdf:resource="#SNUniqueIdentifier" />
      <owl:hasValue rdf:datatype="xsd:string">T020</owl:hasValue>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf xmlns:rdfs="rdfs">
    <owl:Restriction>
```

OWL Species: OWL DL   Semantic Type Number: 135   Relation Number: 54   Total Number: 189

Figure 30 Most commonly used tool trip menu buttons

APPENDIX G. The OWL Descriptions of Various STYs in NetON OWL Full-Extended or NetON OWL DL-Extended documents

The OWL Description of 'Mental_Process' STY:

```
<owl:Class rdf:ID="Mental_Process" >
    <rdfs:subClassOf >
      <owl:Class rdf:about="#Organism_Function" />
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Class rdf:about="#Physiologic_Function" />
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Class rdf:about="#Biologic_Function" />
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Class rdf:about="#Natural_Phenomenon_or_Process" />
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Class rdf:about="#Phenomenon_or_Process" />
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Class rdf:about="#Event" />
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Restriction>
        <owl:onProperty rdf:resource="#affects" />
        <owl:allValuesFrom>
          <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
              <owl:Class rdf:about="#Mental_Process" />
              <owl:Class rdf:about="#Organism_Function" />
              <owl:Class rdf:about="#Organ_or_Tissue_Function" />
              <owl:Class rdf:about="#Genetic_Function" />
              <owl:Class rdf:about="#Molecular_Function" />
              <owl:Class rdf:about="#Cell_Function" />
              <owl:Class rdf:about="#Physiologic_Function" />
              <owl:Class rdf:about="#Experimental_Model_of_Disease" />
              <owl:Class rdf:about="#Neoplastic_Process" />
              <owl:Class rdf:about="#Mental_or_Behavioral_Dysfunction" />
              <owl:Class rdf:about="#Disease_or_Syndrome" />
              <owl:Class rdf:about="#Cell_or_Molecular_Dysfunction" />
              <owl:Class rdf:about="#Pathologic_Function" />
              <owl:Class rdf:about="#Biologic_Function" />
              <owl:Class rdf:about="#Natural_Phenomenon_or_Process" />
              <owl:Class rdf:about="#Virus" />
              <owl:Class rdf:about="#Rickettsia_or_Chlamydia" />
              <owl:Class rdf:about="#Alga" />
              <owl:Class rdf:about="#Plant" />
              <owl:Class rdf:about="#Fungus" />
              <owl:Class rdf:about="#Bacterium" />
              <owl:Class rdf:about="#Archaeon" />
              <owl:Class rdf:about="#Reptile" />
              <owl:Class rdf:about="#Human" />
              <owl:Class rdf:about="#Mammal" />
              <owl:Class rdf:about="#Fish" />
              <owl:Class rdf:about="#Bird" />
              <owl:Class rdf:about="#Amphibian" />
              <owl:Class rdf:about="#Vertebrate" />
              <owl:Class rdf:about="#Invertebrate" />
              <owl:Class rdf:about="#Animal" />
              <owl:Class rdf:about="#Organism" />
              <owl:Class rdf:about="#Clinical_Attribute" />
              <owl:Class rdf:about="#Organism_Attribute" />
              <owl:Class rdf:about="#Social_Behavior" />
              <owl:Class rdf:about="#Individual_Behavior" />
              <owl:Class rdf:about="#Behavior" />
            </owl:unionOf>
          </owl:Class>
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Restriction>
        <owl:onProperty rdf:resource="#process_of" />
        <owl:allValuesFrom>
          <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
              <owl:Class rdf:about="#Mental_Process" />
              <owl:Class rdf:about="#Organism_Function" />
              <owl:Class rdf:about="#Organ_or_Tissue_Function" />
              <owl:Class rdf:about="#Genetic_Function" />
              <owl:Class rdf:about="#Molecular_Function" />
              <owl:Class rdf:about="#Cell_Function" />
              <owl:Class rdf:about="#Physiologic_Function" />
              <owl:Class rdf:about="#Experimental_Model_of_Disease" />
              <owl:Class rdf:about="#Neoplastic_Process" />
              <owl:Class rdf:about="#Mental_or_Behavioral_Dysfunction" />
              <owl:Class rdf:about="#Disease_or_Syndrome" />
              <owl:Class rdf:about="#Cell_or_Molecular_Dysfunction" />
              <owl:Class rdf:about="#Pathologic_Function" />
              <owl:Class rdf:about="#Biologic_Function" />
              <owl:Class rdf:about="#Natural_Phenomenon_or_Process" />
              <owl:Class rdf:about="#Archaeon" />
              <owl:Class rdf:about="#Reptile" />
              <owl:Class rdf:about="#Human" />
              <owl:Class rdf:about="#Mammal" />
              <owl:Class rdf:about="#Fish" />
              <owl:Class rdf:about="#Bird" />
              <owl:Class rdf:about="#Amphibian" />
              <owl:Class rdf:about="#Vertebrate" />
              <owl:Class rdf:about="#Invertebrate" />
              <owl:Class rdf:about="#Animal" />
              <owl:Class rdf:about="#Organism" />
            </owl:unionOf>
          </owl:Class>
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Restriction>
        <owl:onProperty rdf:resource="#result_of" />
        <owl:allValuesFrom>
          <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
              <owl:Class rdf:about="#Mental_Process" />
              <owl:Class rdf:about="#Organism_Function" />
              <owl:Class rdf:about="#Organ_or_Tissue_Function" />
              <owl:Class rdf:about="#Genetic_Function" />
              <owl:Class rdf:about="#Molecular_Function" />
              <owl:Class rdf:about="#Cell_Function" />
              <owl:Class rdf:about="#Physiologic_Function" />
              <owl:Class rdf:about="#Experimental_Model_of_Disease" />
              <owl:Class rdf:about="#Neoplastic_Process" />
              <owl:Class rdf:about="#Mental_or_Behavioral_Dysfunction" />
              <owl:Class rdf:about="#Disease_or_Syndrome" />
              <owl:Class rdf:about="#Cell_or_Molecular_Dysfunction" />
              <owl:Class rdf:about="#Pathologic_Function" />
              <owl:Class rdf:about="#Biologic_Function" />
              <owl:Class rdf:about="#Natural_Phenomenon_or_Process" />
              <owl:Class rdf:about="#Injury_or_Poisoning" />
              <owl:Class rdf:about="#Environmental_Effect_of_Humans" />
              <owl:Class rdf:about="#Human-caused_Phenomenon_or_Process" />
              <owl:Class rdf:about="#Phenomenon_or_Process" />
              <owl:Class rdf:about="#Congenital_Abnormality" />
              <owl:Class rdf:about="#Acquired_Abnormality" />
              <owl:Class rdf:about="#Anatomical_Abnormality" />
              <owl:Class rdf:about="#Social_Behavior" />
              <owl:Class rdf:about="#Individual_Behavior" />
              <owl:Class rdf:about="#Behavior" />
            </owl:unionOf>
          </owl:Class>
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Restriction>
        <owl:onProperty rdf:resource="#degree_of" />
        <owl:allValuesFrom>
          <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
              <owl:Class rdf:about="#Mental_Process" />
              <owl:Class rdf:about="#Organism_Function" />
            </owl:unionOf>
          </owl:Class>
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Restriction>
        <owl:onProperty rdf:resource="#produces" />
        <owl:allValuesFrom>
          <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
              <owl:Class rdf:about="#Body_Substance" />
              <owl:Class rdf:about="#Vitamin" />
              <owl:Class rdf:about="#Receptor" />
              <owl:Class rdf:about="#Neuroreactive_Substance_or_Biogenic_Amine"/>
              <owl:Class rdf:about="#Immunologic_Factor" />
              <owl:Class rdf:about="#Hormone" />
              <owl:Class rdf:about="#Enzyme" />
              <owl:Class rdf:about="#Biologically_Active_Substance" />
              <owl:Class rdf:about="#Family_Group" />
              <owl:Class rdf:about="#Age_Group" />
            </owl:unionOf>
          </owl:Class>
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Restriction>
        <owl:onProperty rdf:resource="#co-occurs_with" />
        <owl:allValuesFrom>
          <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
              <owl:Class rdf:about="#Mental_Process" />
              <owl:Class rdf:about="#Organism_Function" />
              <owl:Class rdf:about="#Organ_or_Tissue_Function" />
              <owl:Class rdf:about="#Genetic_Function" />
              <owl:Class rdf:about="#Molecular_Function" />
              <owl:Class rdf:about="#Cell_Function" />
              <owl:Class rdf:about="#Physiologic_Function" />
            </owl:unionOf>
          </owl:Class>
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Restriction>
        <owl:onProperty rdf:resource="#occurs_in" />
        <owl:allValuesFrom>
          <owl:Class rdf:about="#Temporal_Concept" />
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Restriction>
        <owl:onProperty rdf:resource="#precedes" />
        <owl:allValuesFrom>
          <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
              <owl:Class rdf:about="#Mental_Process" />
              <owl:Class rdf:about="#Organism_Function" />
              <owl:Class rdf:about="#Organ_or_Tissue_Function" />
              <owl:Class rdf:about="#Genetic_Function" />
              <owl:Class rdf:about="#Molecular_Function" />
              <owl:Class rdf:about="#Cell_Function" />
```

```
              <owl:Class rdf:about="#Physiologic_Function" />
            </owl:unionOf>
          </owl:Class>
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Restriction>
        <owl:onProperty rdf:resource="#issue_in" />
        <owl:allValuesFrom>
          <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
              <owl:Class rdf:about="#Biomedical_Occupation_or_Discipline" />
              <owl:Class rdf:about="#Occupation_or_Discipline" />
            </owl:unionOf>
          </owl:Class>
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
  </owl:Class>
```

The OWL Description of ‘Disease_or_Syndrome’ STY:

```
<owl:Class rdf:ID="Disease_or_Syndrome"  >
    <rdfs:subClassOf >
      <owl:Class rdf:about="#Pathologic_Function" />
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Class rdf:about="#Biologic_Function" />
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Class rdf:about="#Natural_Phenomenon_or_Process" />
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Class rdf:about="#Phenomenon_or_Process" />
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Class rdf:about="#Event" />
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Restriction>
        <owl:onProperty rdf:resource="#conceptually_related_to" />
        <owl:allValuesFrom>
          <owl:Class rdf:about="#Experimental_Model_of_Disease" />
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Restriction>
        <owl:onProperty rdf:resource="#produces" />
        <owl:allValuesFrom>
          <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
              <owl:Class rdf:about="#Body_Substance" />
              <owl:Class rdf:about="#Vitamin" />
              <owl:Class rdf:about="#Receptor" />
              <owl:Class rdf:about="#Neuroreactive_Substance_or_Biogenic_Amine"/>
              <owl:Class rdf:about="#Immunologic_Factor" />
              <owl:Class rdf:about="#Hormone" />
              <owl:Class rdf:about="#Enzyme" />
              <owl:Class rdf:about="#Biologically_Active_Substance" />
              <owl:Class rdf:about="#Tissue" />
            </owl:unionOf>
          </owl:Class>
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Restriction>
        <owl:onProperty rdf:resource="#associated_with" />
        <owl:allValuesFrom>
          <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
              <owl:Class rdf:about="#Experimental_Model_of_Disease" />
            <owl:Class rdf:about="#Neoplastic_Process" />
            <owl:Class rdf:about="#Mental_or_Behavioral_Dysfunction" />
            <owl:Class rdf:about="#Disease_or_Syndrome" />
            <owl:Class rdf:about="#Cell_or_Molecular_Dysfunction" />
            <owl:Class rdf:about="#Pathologic_Function" />
            <owl:Class rdf:about="#Clinical_Attribute" />
            <owl:Class rdf:about="#Organism_Attribute" />
          </owl:unionOf>
        </owl:Class>
      </owl:allValuesFrom>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf >
    <owl:Restriction>
      <owl:onProperty rdf:resource="#co-occurs_with" />
      <owl:allValuesFrom>
        <owl:Class>
          <owl:unionOf rdf:parseType="Collection">
            <owl:Class rdf:about="#Experimental_Model_of_Disease" />
            <owl:Class rdf:about="#Neoplastic_Process" />
            <owl:Class rdf:about="#Mental_or_Behavioral_Dysfunction" />
            <owl:Class rdf:about="#Disease_or_Syndrome" />
            <owl:Class rdf:about="#Cell_or_Molecular_Dysfunction" />
            <owl:Class rdf:about="#Pathologic_Function" />
            <owl:Class rdf:about="#Injury_or_Poisoning" />
            <owl:Class rdf:about="#Congenital_Abnormality" />
            <owl:Class rdf:about="#Acquired_Abnormality" />
            <owl:Class rdf:about="#Anatomical_Abnormality" />
          </owl:unionOf>
        </owl:Class>
      </owl:allValuesFrom>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf >
    <owl:Restriction>
      <owl:onProperty rdf:resource="#complicates" />
      <owl:allValuesFrom>
        <owl:Class>
          <owl:unionOf rdf:parseType="Collection">
            <owl:Class rdf:about="#Experimental_Model_of_Disease" />
            <owl:Class rdf:about="#Neoplastic_Process" />
            <owl:Class rdf:about="#Mental_or_Behavioral_Dysfunction" />
            <owl:Class rdf:about="#Disease_or_Syndrome" />
            <owl:Class rdf:about="#Cell_or_Molecular_Dysfunction" />
            <owl:Class rdf:about="#Pathologic_Function" />
            <owl:Class rdf:about="#Injury_or_Poisoning" />
            <owl:Class rdf:about="#Congenital_Abnormality" />
            <owl:Class rdf:about="#Acquired_Abnormality" />
            <owl:Class rdf:about="#Anatomical_Abnormality" />
          </owl:unionOf>
        </owl:Class>
      </owl:allValuesFrom>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf >
    <owl:Restriction>
      <owl:onProperty rdf:resource="#degree_of" />
      <owl:allValuesFrom>
        <owl:Class>
          <owl:unionOf rdf:parseType="Collection">
            <owl:Class rdf:about="#Experimental_Model_of_Disease" />
            <owl:Class rdf:about="#Neoplastic_Process" />
            <owl:Class rdf:about="#Mental_or_Behavioral_Dysfunction" />
            <owl:Class rdf:about="#Disease_or_Syndrome" />
            <owl:Class rdf:about="#Cell_or_Molecular_Dysfunction" />
            <owl:Class rdf:about="#Pathologic_Function" />
          </owl:unionOf>
        </owl:Class>
      </owl:allValuesFrom>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf >
    <owl:Restriction>
      <owl:onProperty rdf:resource="#manifestation_of" />
      <owl:allValuesFrom>
```

```
        <owl:Class>
          <owl:unionOf rdf:parseType="Collection">
            <owl:Class rdf:about="#Mental_Process" />
            <owl:Class rdf:about="#Organism_Function" />
            <owl:Class rdf:about="#Organ_or_Tissue_Function" />
            <owl:Class rdf:about="#Genetic_Function" />
            <owl:Class rdf:about="#Molecular_Function" />
            <owl:Class rdf:about="#Cell_Function" />
            <owl:Class rdf:about="#Physiologic_Function" />
            <owl:Class rdf:about="#Experimental_Model_of_Disease" />
            <owl:Class rdf:about="#Neoplastic_Process" />
            <owl:Class rdf:about="#Mental_or_Behavioral_Dysfunction" />
            <owl:Class rdf:about="#Disease_or_Syndrome" />
            <owl:Class rdf:about="#Cell_or_Molecular_Dysfunction" />
            <owl:Class rdf:about="#Pathologic_Function" />
            <owl:Class rdf:about="#Injury_or_Poisoning" />
          </owl:unionOf>
        </owl:Class>
      </owl:allValuesFrom>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf >
    <owl:Restriction>
      <owl:onProperty rdf:resource="#occurs_in" />
      <owl:allValuesFrom>
        <owl:Class>
          <owl:unionOf rdf:parseType="Collection">
            <owl:Class rdf:about="#Injury_or_Poisoning" />
            <owl:Class rdf:about="#Professional_or_Occupational_Group" />
            <owl:Class rdf:about="#Population_Group" />
            <owl:Class rdf:about="#Patient_or_Disabled_Group" />
            <owl:Class rdf:about="#Family_Group" />
            <owl:Class rdf:about="#Age_Group" />
            <owl:Class rdf:about="#Group" />
            <owl:Class rdf:about="#Neoplastic_Process" />
            <owl:Class rdf:about="#Mental_or_Behavioral_Dysfunction" />
            <owl:Class rdf:about="#Disease_or_Syndrome" />
          </owl:unionOf>
        </owl:Class>
      </owl:allValuesFrom>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf >
    <owl:Restriction>
      <owl:onProperty rdf:resource="#precedes" />
      <owl:allValuesFrom>
        <owl:Class>
          <owl:unionOf rdf:parseType="Collection">
            <owl:Class rdf:about="#Experimental_Model_of_Disease" />
            <owl:Class rdf:about="#Neoplastic_Process" />
            <owl:Class rdf:about="#Mental_or_Behavioral_Dysfunction" />
            <owl:Class rdf:about="#Disease_or_Syndrome" />
            <owl:Class rdf:about="#Cell_or_Molecular_Dysfunction" />
            <owl:Class rdf:about="#Pathologic_Function" />
          </owl:unionOf>
        </owl:Class>
      </owl:allValuesFrom>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf >
    <owl:Restriction>
      <owl:onProperty rdf:resource="#result_of" />
      <owl:allValuesFrom>
        <owl:Class>
          <owl:unionOf rdf:parseType="Collection">
            <owl:Class rdf:about="#Mental_Process" />
            <owl:Class rdf:about="#Organism_Function" />
            <owl:Class rdf:about="#Organ_or_Tissue_Function" />
            <owl:Class rdf:about="#Genetic_Function" />
            <owl:Class rdf:about="#Molecular_Function" />
            <owl:Class rdf:about="#Cell_Function" />
            <owl:Class rdf:about="#Physiologic_Function" />
            <owl:Class rdf:about="#Experimental_Model_of_Disease" />
            <owl:Class rdf:about="#Neoplastic_Process" />
            <owl:Class rdf:about="#Mental_or_Behavioral_Dysfunction" />
```

```xml
          <owl:Class rdf:about="#Disease_or_Syndrome" />
          <owl:Class rdf:about="#Cell_or_Molecular_Dysfunction" />
          <owl:Class rdf:about="#Pathologic_Function" />
          <owl:Class rdf:about="#Biologic_Function" />
          <owl:Class rdf:about="#Natural_Phenomenon_or_Process" />
          <owl:Class rdf:about="#Injury_or_Poisoning" />
          <owl:Class rdf:about="#Environmental_Effect_of_Humans" />
          <owl:Class rdf:about="#Human-caused_Phenomenon_or_Process" />
          <owl:Class rdf:about="#Phenomenon_or_Process" />
          <owl:Class rdf:about="#Congenital_Abnormality" />
          <owl:Class rdf:about="#Acquired_Abnormality" />
          <owl:Class rdf:about="#Anatomical_Abnormality" />
          <owl:Class rdf:about="#Therapeutic_or_Preventive_Procedure" />
          <owl:Class rdf:about="#Laboratory_Procedure" />
          <owl:Class rdf:about="#Diagnostic_Procedure" />
          <owl:Class rdf:about="#Health_Care_Activity" />
          <owl:Class rdf:about="#Social_Behavior" />
          <owl:Class rdf:about="#Individual_Behavior" />
          <owl:Class rdf:about="#Behavior" />
        </owl:unionOf>
      </owl:Class>
    </owl:allValuesFrom>
  </owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf >
  <owl:Restriction>
    <owl:onProperty rdf:resource="#affects" />
    <owl:allValuesFrom>
      <owl:Class>
        <owl:unionOf rdf:parseType="Collection">
          <owl:Class rdf:about="#Mental_Process" />
          <owl:Class rdf:about="#Organism_Function" />
          <owl:Class rdf:about="#Organ_or_Tissue_Function" />
          <owl:Class rdf:about="#Genetic_Function" />
          <owl:Class rdf:about="#Molecular_Function" />
          <owl:Class rdf:about="#Cell_Function" />
          <owl:Class rdf:about="#Physiologic_Function" />
          <owl:Class rdf:about="#Experimental_Model_of_Disease" />
          <owl:Class rdf:about="#Neoplastic_Process" />
          <owl:Class rdf:about="#Mental_or_Behavioral_Dysfunction" />
          <owl:Class rdf:about="#Disease_or_Syndrome" />
          <owl:Class rdf:about="#Cell_or_Molecular_Dysfunction" />
          <owl:Class rdf:about="#Pathologic_Function" />
          <owl:Class rdf:about="#Biologic_Function" />
          <owl:Class rdf:about="#Natural_Phenomenon_or_Process" />
          <owl:Class rdf:about="#Virus" />
          <owl:Class rdf:about="#Rickettsia_or_Chlamydia" />
          <owl:Class rdf:about="#Alga" />
          <owl:Class rdf:about="#Plant" />
          <owl:Class rdf:about="#Fungus" />
          <owl:Class rdf:about="#Bacterium" />
          <owl:Class rdf:about="#Archaeon" />
          <owl:Class rdf:about="#Reptile" />
          <owl:Class rdf:about="#Human" />
          <owl:Class rdf:about="#Mammal" />
          <owl:Class rdf:about="#Fish" />
          <owl:Class rdf:about="#Bird" />
          <owl:Class rdf:about="#Amphibian" />
          <owl:Class rdf:about="#Vertebrate" />
          <owl:Class rdf:about="#Invertebrate" />
          <owl:Class rdf:about="#Animal" />
          <owl:Class rdf:about="#Organism" />
        </owl:unionOf>
      </owl:Class>
    </owl:allValuesFrom>
  </owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf >
  <owl:Restriction>
    <owl:onProperty rdf:resource="#process_of" />
    <owl:allValuesFrom>
      <owl:Class>
        <owl:unionOf rdf:parseType="Collection">
          <owl:Class rdf:about="#Mental_Process" />
          <owl:Class rdf:about="#Organism_Function" />
```

```
              <owl:Class rdf:about="#Organ_or_Tissue_Function" />
              <owl:Class rdf:about="#Genetic_Function" />
              <owl:Class rdf:about="#Molecular_Function" />
              <owl:Class rdf:about="#Cell_Function" />
              <owl:Class rdf:about="#Physiologic_Function" />
              <owl:Class rdf:about="#Experimental_Model_of_Disease" />
              <owl:Class rdf:about="#Neoplastic_Process" />
              <owl:Class rdf:about="#Mental_or_Behavioral_Dysfunction" />
              <owl:Class rdf:about="#Disease_or_Syndrome" />
              <owl:Class rdf:about="#Cell_or_Molecular_Dysfunction" />
              <owl:Class rdf:about="#Pathologic_Function" />
              <owl:Class rdf:about="#Biologic_Function" />
              <owl:Class rdf:about="#Natural_Phenomenon_or_Process" />
              <owl:Class rdf:about="#Virus" />
              <owl:Class rdf:about="#Rickettsia_or_Chlamydia" />
              <owl:Class rdf:about="#Alga" />
              <owl:Class rdf:about="#Plant" />
              <owl:Class rdf:about="#Fungus" />
              <owl:Class rdf:about="#Bacterium" />
              <owl:Class rdf:about="#Archaeon" />
              <owl:Class rdf:about="#Reptile" />
              <owl:Class rdf:about="#Human" />
              <owl:Class rdf:about="#Mammal" />
              <owl:Class rdf:about="#Fish" />
              <owl:Class rdf:about="#Bird" />
              <owl:Class rdf:about="#Amphibian" />
              <owl:Class rdf:about="#Vertebrate" />
              <owl:Class rdf:about="#Invertebrate" />
              <owl:Class rdf:about="#Animal" />
              <owl:Class rdf:about="#Organism" />
            </owl:unionOf>
          </owl:Class>
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Restriction>
        <owl:onProperty rdf:resource="#issue_in" />
        <owl:allValuesFrom>
          <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
              <owl:Class rdf:about="#Biomedical_Occupation_or_Discipline" />
              <owl:Class rdf:about="#Occupation_or_Discipline" />
            </owl:unionOf>
          </owl:Class>
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
  </owl:Class>
```

The OWL Description of ‘Neoplastic_Process’ STY:

```
<owl:Class rdf:ID="Neoplastic_Process"  >
    <rdfs:subClassOf >
      <owl:Class rdf:about="#Disease_or_Syndrome" />
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Class rdf:about="#Pathologic_Function" />
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Class rdf:about="#Biologic_Function" />
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Class rdf:about="#Natural_Phenomenon_or_Process" />
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Class rdf:about="#Phenomenon_or_Process" />
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Class rdf:about="#Event" />
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Restriction>
        <owl:onProperty rdf:resource="#produces" />
        <owl:allValuesFrom>
          <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
              <owl:Class rdf:about="#Body_Substance" />
              <owl:Class rdf:about="#Vitamin" />
              <owl:Class rdf:about="#Receptor" />
              <owl:Class rdf:about="#Neuroreactive_Substance_or_Biogenic_Amine"/>
              <owl:Class rdf:about="#Immunologic_Factor" />
              <owl:Class rdf:about="#Hormone" />
              <owl:Class rdf:about="#Enzyme" />
              <owl:Class rdf:about="#Biologically_Active_Substance" />
              <owl:Class rdf:about="#Tissue" />
            </owl:unionOf>
          </owl:Class>
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Restriction>
        <owl:onProperty rdf:resource="#associated_with" />
        <owl:allValuesFrom>
          <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
              <owl:Class rdf:about="#Experimental_Model_of_Disease" />
              <owl:Class rdf:about="#Neoplastic_Process" />
              <owl:Class rdf:about="#Cell_or_Molecular_Dysfunction" />
              <owl:Class rdf:about="#Pathologic_Function" />
              <owl:Class rdf:about="#Clinical_Attribute" />
              <owl:Class rdf:about="#Organism_Attribute" />
            </owl:unionOf>
          </owl:Class>
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Restriction>
        <owl:onProperty rdf:resource="#co-occurs_with" />
        <owl:allValuesFrom>
          <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
              <owl:Class rdf:about="#Experimental_Model_of_Disease" />
              <owl:Class rdf:about="#Neoplastic_Process" />
              <owl:Class rdf:about="#Cell_or_Molecular_Dysfunction" />
              <owl:Class rdf:about="#Pathologic_Function" />
              <owl:Class rdf:about="#Injury_or_Poisoning" />
              <owl:Class rdf:about="#Congenital_Abnormality" />
              <owl:Class rdf:about="#Acquired_Abnormality" />
              <owl:Class rdf:about="#Anatomical_Abnormality" />
            </owl:unionOf>
          </owl:Class>
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Restriction>
        <owl:onProperty rdf:resource="#complicates" />
        <owl:allValuesFrom>
          <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
              <owl:Class rdf:about="#Experimental_Model_of_Disease" />
              <owl:Class rdf:about="#Neoplastic_Process" />
              <owl:Class rdf:about="#Mental_or_Behavioral_Dysfunction" />
              <owl:Class rdf:about="#Disease_or_Syndrome" />
              <owl:Class rdf:about="#Cell_or_Molecular_Dysfunction" />
              <owl:Class rdf:about="#Pathologic_Function" />
              <owl:Class rdf:about="#Injury_or_Poisoning" />
              <owl:Class rdf:about="#Congenital_Abnormality" />
              <owl:Class rdf:about="#Acquired_Abnormality" />
              <owl:Class rdf:about="#Anatomical_Abnormality" />
            </owl:unionOf>
          </owl:Class>
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf >
```

```
  <owl:Restriction>
    <owl:onProperty rdf:resource="#degree_of" />
    <owl:allValuesFrom>
      <owl:Class>
        <owl:unionOf rdf:parseType="Collection">
          <owl:Class rdf:about="#Experimental_Model_of_Disease" />
          <owl:Class rdf:about="#Neoplastic_Process" />
          <owl:Class rdf:about="#Mental_or_Behavioral_Dysfunction" />
          <owl:Class rdf:about="#Disease_or_Syndrome" />
          <owl:Class rdf:about="#Cell_or_Molecular_Dysfunction" />
          <owl:Class rdf:about="#Pathologic_Function" />
        </owl:unionOf>
      </owl:Class>
    </owl:allValuesFrom>
  </owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf >
  <owl:Restriction>
    <owl:onProperty rdf:resource="#manifestation_of" />
    <owl:allValuesFrom>
      <owl:Class>
        <owl:unionOf rdf:parseType="Collection">
          <owl:Class rdf:about="#Mental_Process" />
          <owl:Class rdf:about="#Organism_Function" />
          <owl:Class rdf:about="#Organ_or_Tissue_Function" />
          <owl:Class rdf:about="#Genetic_Function" />
          <owl:Class rdf:about="#Molecular_Function" />
          <owl:Class rdf:about="#Cell_Function" />
          <owl:Class rdf:about="#Physiologic_Function" />
          <owl:Class rdf:about="#Experimental_Model_of_Disease" />
          <owl:Class rdf:about="#Neoplastic_Process" />
          <owl:Class rdf:about="#Mental_or_Behavioral_Dysfunction" />
          <owl:Class rdf:about="#Disease_or_Syndrome" />
          <owl:Class rdf:about="#Cell_or_Molecular_Dysfunction" />
          <owl:Class rdf:about="#Pathologic_Function" />
          <owl:Class rdf:about="#Injury_or_Poisoning" />
        </owl:unionOf>
      </owl:Class>
    </owl:allValuesFrom>
  </owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf >
  <owl:Restriction>
    <owl:onProperty rdf:resource="#occurs_in" />
    <owl:allValuesFrom>
      <owl:Class>
        <owl:unionOf rdf:parseType="Collection">
          <owl:Class rdf:about="#Injury_or_Poisoning" />
          <owl:Class rdf:about="#Professional_or_Occupational_Group" />
          <owl:Class rdf:about="#Population_Group" />
          <owl:Class rdf:about="#Patient_or_Disabled_Group" />
          <owl:Class rdf:about="#Family_Group" />
          <owl:Class rdf:about="#Age_Group" />
          <owl:Class rdf:about="#Group" />
          <owl:Class rdf:about="#Neoplastic_Process" />
          <owl:Class rdf:about="#Mental_or_Behavioral_Dysfunction" />
          <owl:Class rdf:about="#Disease_or_Syndrome" />
        </owl:unionOf>
      </owl:Class>
    </owl:allValuesFrom>
  </owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf >
  <owl:Restriction>
    <owl:onProperty rdf:resource="#precedes" />
    <owl:allValuesFrom>
      <owl:Class>
        <owl:unionOf rdf:parseType="Collection">
          <owl:Class rdf:about="#Experimental_Model_of_Disease" />
          <owl:Class rdf:about="#Neoplastic_Process" />
          <owl:Class rdf:about="#Mental_or_Behavioral_Dysfunction" />
          <owl:Class rdf:about="#Disease_or_Syndrome" />
          <owl:Class rdf:about="#Cell_or_Molecular_Dysfunction" />
          <owl:Class rdf:about="#Pathologic_Function" />
        </owl:unionOf>
      </owl:Class>
    </owl:allValuesFrom>
  </owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf >
  <owl:Restriction>
    <owl:onProperty rdf:resource="#result_of" />
    <owl:allValuesFrom>
      <owl:Class>
        <owl:unionOf rdf:parseType="Collection">
          <owl:Class rdf:about="#Mental_Process" />
          <owl:Class rdf:about="#Organism_Function" />
          <owl:Class rdf:about="#Organ_or_Tissue_Function" />
          <owl:Class rdf:about="#Genetic_Function" />
          <owl:Class rdf:about="#Molecular_Function" />
          <owl:Class rdf:about="#Cell_Function" />
          <owl:Class rdf:about="#Physiologic_Function" />
          <owl:Class rdf:about="#Experimental_Model_of_Disease" />
          <owl:Class rdf:about="#Neoplastic_Process" />
          <owl:Class rdf:about="#Mental_or_Behavioral_Dysfunction" />
          <owl:Class rdf:about="#Disease_or_Syndrome" />
          <owl:Class rdf:about="#Cell_or_Molecular_Dysfunction" />
          <owl:Class rdf:about="#Pathologic_Function" />
          <owl:Class rdf:about="#Biologic_Function" />
          <owl:Class rdf:about="#Natural_Phenomenon_or_Process" />
          <owl:Class rdf:about="#Injury_or_Poisoning" />
          <owl:Class rdf:about="#Environmental_Effect_of_Humans" />
          <owl:Class rdf:about="#Human-caused_Phenomenon_or_Process" />
          <owl:Class rdf:about="#Phenomenon_or_Process" />
          <owl:Class rdf:about="#Congenital_Abnormality" />
          <owl:Class rdf:about="#Acquired_Abnormality" />
          <owl:Class rdf:about="#Anatomical_Abnormality" />
          <owl:Class rdf:about="#Therapeutic_or_Preventive_Procedure" />
          <owl:Class rdf:about="#Laboratory_Procedure" />
          <owl:Class rdf:about="#Diagnostic_Procedure" />
          <owl:Class rdf:about="#Health_Care_Activity" />
          <owl:Class rdf:about="#Social_Behavior" />
          <owl:Class rdf:about="#Individual_Behavior" />
          <owl:Class rdf:about="#Behavior" />
        </owl:unionOf>
      </owl:Class>
    </owl:allValuesFrom>
  </owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf >
  <owl:Restriction>
    <owl:onProperty rdf:resource="#affects" />
    <owl:allValuesFrom>
      <owl:Class>
        <owl:unionOf rdf:parseType="Collection">
          <owl:Class rdf:about="#Mental_Process" />
          <owl:Class rdf:about="#Organism_Function" />
          <owl:Class rdf:about="#Organ_or_Tissue_Function" />
          <owl:Class rdf:about="#Genetic_Function" />
          <owl:Class rdf:about="#Molecular_Function" />
          <owl:Class rdf:about="#Cell_Function" />
          <owl:Class rdf:about="#Physiologic_Function" />
          <owl:Class rdf:about="#Experimental_Model_of_Disease" />
          <owl:Class rdf:about="#Neoplastic_Process" />
          <owl:Class rdf:about="#Mental_or_Behavioral_Dysfunction" />
          <owl:Class rdf:about="#Disease_or_Syndrome" />
          <owl:Class rdf:about="#Cell_or_Molecular_Dysfunction" />
          <owl:Class rdf:about="#Pathologic_Function" />
          <owl:Class rdf:about="#Biologic_Function" />
          <owl:Class rdf:about="#Natural_Phenomenon_or_Process" />
          <owl:Class rdf:about="#Virus" />
          <owl:Class rdf:about="#Rickettsia_or_Chlamydia" />
          <owl:Class rdf:about="#Alga" />
          <owl:Class rdf:about="#Plant" />
          <owl:Class rdf:about="#Fungus" />
          <owl:Class rdf:about="#Bacterium" />
          <owl:Class rdf:about="#Archaeon" />
          <owl:Class rdf:about="#Reptile" />
          <owl:Class rdf:about="#Human" />
          <owl:Class rdf:about="#Mammal" />
```

```
              <owl:Class rdf:about="#Fish" />
              <owl:Class rdf:about="#Bird" />
              <owl:Class rdf:about="#Amphibian" />
              <owl:Class rdf:about="#Vertebrate" />
              <owl:Class rdf:about="#Invertebrate" />
              <owl:Class rdf:about="#Animal" />
              <owl:Class rdf:about="#Organism" />
            </owl:unionOf>
          </owl:Class>
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Restriction>
        <owl:onProperty rdf:resource="#process_of" />
        <owl:allValuesFrom>
          <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
              <owl:Class rdf:about="#Mental_Process" />
              <owl:Class rdf:about="#Organism_Function" />
              <owl:Class rdf:about="#Organ_or_Tissue_Function" />
              <owl:Class rdf:about="#Genetic_Function" />
              <owl:Class rdf:about="#Molecular_Function" />
              <owl:Class rdf:about="#Cell_Function" />
              <owl:Class rdf:about="#Physiologic_Function" />
              <owl:Class rdf:about="#Experimental_Model_of_Disease" />
              <owl:Class rdf:about="#Neoplastic_Process" />
              <owl:Class rdf:about="#Mental_or_Behavioral_Dysfunction" />
              <owl:Class rdf:about="#Disease_or_Syndrome" />
              <owl:Class rdf:about="#Cell_or_Molecular_Dysfunction" />
              <owl:Class rdf:about="#Pathologic_Function" />
              <owl:Class rdf:about="#Biologic_Function" />
              <owl:Class rdf:about="#Natural_Phenomenon_or_Process" />
              <owl:Class rdf:about="#Virus" />
              <owl:Class rdf:about="#Rickettsia_or_Chlamydia" />
              <owl:Class rdf:about="#Alga" />
              <owl:Class rdf:about="#Plant" />
              <owl:Class rdf:about="#Fungus" />
              <owl:Class rdf:about="#Bacterium" />
              <owl:Class rdf:about="#Archaeon" />
              <owl:Class rdf:about="#Reptile" />
              <owl:Class rdf:about="#Human" />
              <owl:Class rdf:about="#Mammal" />
              <owl:Class rdf:about="#Fish" />
              <owl:Class rdf:about="#Bird" />
              <owl:Class rdf:about="#Amphibian" />
              <owl:Class rdf:about="#Vertebrate" />
              <owl:Class rdf:about="#Invertebrate" />
              <owl:Class rdf:about="#Animal" />
              <owl:Class rdf:about="#Organism" />
            </owl:unionOf>
          </owl:Class>
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Restriction>
        <owl:onProperty rdf:resource="#issue_in" />
        <owl:allValuesFrom>
          <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
              <owl:Class rdf:about="#Biomedical_Occupation_or_Discipline" />
              <owl:Class rdf:about="#Occupation_or_Discipline" />
            </owl:unionOf>
          </owl:Class>
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
  </owl:Class>
```

The OWL Description of 'Mental_or_Behavioral_Dysfunction' STY:

```
<owl:Class rdf:ID="Mental_or_Behavioral_Dysfunction"  >
    <rdfs:subClassOf >
```

```
  <owl:Class rdf:about="#Disease_or_Syndrome" />
</rdfs:subClassOf>
<rdfs:subClassOf >
  <owl:Class rdf:about="#Pathologic_Function" />
</rdfs:subClassOf>
<rdfs:subClassOf >
  <owl:Class rdf:about="#Biologic_Function" />
</rdfs:subClassOf>
<rdfs:subClassOf >
  <owl:Class rdf:about="#Natural_Phenomenon_or_Process" />
</rdfs:subClassOf>
<rdfs:subClassOf >
  <owl:Class rdf:about="#Phenomenon_or_Process" />
</rdfs:subClassOf>
<rdfs:subClassOf >
  <owl:Class rdf:about="#Event" />
</rdfs:subClassOf>
<rdfs:subClassOf >
  <owl:Restriction>
    <owl:onProperty rdf:resource="#affects" />
    <owl:allValuesFrom>
      <owl:Class>
        <owl:unionOf rdf:parseType="Collection">
          <owl:Class rdf:about="#Mental_Process" />
          <owl:Class rdf:about="#Organism_Function" />
          <owl:Class rdf:about="#Organ_or_Tissue_Function" />
          <owl:Class rdf:about="#Genetic_Function" />
          <owl:Class rdf:about="#Molecular_Function" />
          <owl:Class rdf:about="#Cell_Function" />
          <owl:Class rdf:about="#Physiologic_Function" />
          <owl:Class rdf:about="#Experimental_Model_of_Disease" />
          <owl:Class rdf:about="#Neoplastic_Process" />
          <owl:Class rdf:about="#Mental_or_Behavioral_Dysfunction" />
          <owl:Class rdf:about="#Disease_or_Syndrome" />
          <owl:Class rdf:about="#Cell_or_Molecular_Dysfunction" />
          <owl:Class rdf:about="#Pathologic_Function" />
          <owl:Class rdf:about="#Biologic_Function" />
          <owl:Class rdf:about="#Natural_Phenomenon_or_Process" />
          <owl:Class rdf:about="#Virus" />
          <owl:Class rdf:about="#Rickettsia_or_Chlamydia" />
          <owl:Class rdf:about="#Alga" />
          <owl:Class rdf:about="#Plant" />
          <owl:Class rdf:about="#Fungus" />
          <owl:Class rdf:about="#Bacterium" />
          <owl:Class rdf:about="#Archaeon" />
          <owl:Class rdf:about="#Reptile" />
          <owl:Class rdf:about="#Human" />
          <owl:Class rdf:about="#Mammal" />
          <owl:Class rdf:about="#Fish" />
          <owl:Class rdf:about="#Bird" />
          <owl:Class rdf:about="#Amphibian" />
          <owl:Class rdf:about="#Vertebrate" />
          <owl:Class rdf:about="#Invertebrate" />
          <owl:Class rdf:about="#Animal" />
          <owl:Class rdf:about="#Organism" />
          <owl:Class rdf:about="#Social_Behavior" />
          <owl:Class rdf:about="#Individual_Behavior" />
          <owl:Class rdf:about="#Behavior" />
        </owl:unionOf>
      </owl:Class>
    </owl:allValuesFrom>
  </owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf >
  <owl:Restriction>
    <owl:onProperty rdf:resource="#process_of" />
    <owl:allValuesFrom>
      <owl:Class>
        <owl:unionOf rdf:parseType="Collection">
          <owl:Class rdf:about="#Mental_Process" />
          <owl:Class rdf:about="#Organism_Function" />
          <owl:Class rdf:about="#Organ_or_Tissue_Function" />
          <owl:Class rdf:about="#Genetic_Function" />
          <owl:Class rdf:about="#Molecular_Function" />
          <owl:Class rdf:about="#Cell_Function" />
          <owl:Class rdf:about="#Physiologic_Function" />
          <owl:Class rdf:about="#Experimental_Model_of_Disease" />
          <owl:Class rdf:about="#Neoplastic_Process" />
          <owl:Class rdf:about="#Mental_or_Behavioral_Dysfunction" />
          <owl:Class rdf:about="#Disease_or_Syndrome" />
          <owl:Class rdf:about="#Cell_or_Molecular_Dysfunction" />
          <owl:Class rdf:about="#Pathologic_Function" />
          <owl:Class rdf:about="#Biologic_Function" />
          <owl:Class rdf:about="#Natural_Phenomenon_or_Process" />
          <owl:Class rdf:about="#Archaeon" />
          <owl:Class rdf:about="#Reptile" />
          <owl:Class rdf:about="#Human" />
          <owl:Class rdf:about="#Mammal" />
          <owl:Class rdf:about="#Fish" />
          <owl:Class rdf:about="#Bird" />
          <owl:Class rdf:about="#Amphibian" />
          <owl:Class rdf:about="#Vertebrate" />
          <owl:Class rdf:about="#Invertebrate" />
          <owl:Class rdf:about="#Animal" />
          <owl:Class rdf:about="#Organism" />
        </owl:unionOf>
      </owl:Class>
    </owl:allValuesFrom>
  </owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf >
  <owl:Restriction>
    <owl:onProperty rdf:resource="#produces" />
    <owl:allValuesFrom>
      <owl:Class>
        <owl:unionOf rdf:parseType="Collection">
          <owl:Class rdf:about="#Body_Substance" />
          <owl:Class rdf:about="#Vitamin" />
          <owl:Class rdf:about="#Receptor" />
          <owl:Class rdf:about="#Neuroreactive_Substance_or_Biogenic_Amine" />
          <owl:Class rdf:about="#Immunologic_Factor" />
          <owl:Class rdf:about="#Hormone" />
          <owl:Class rdf:about="#Enzyme" />
          <owl:Class rdf:about="#Biologically_Active_Substance" />
          <owl:Class rdf:about="#Tissue" />
        </owl:unionOf>
      </owl:Class>
    </owl:allValuesFrom>
  </owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf >
  <owl:Restriction>
    <owl:onProperty rdf:resource="#associated_with" />
    <owl:allValuesFrom>
      <owl:Class>
        <owl:unionOf rdf:parseType="Collection">
          <owl:Class rdf:about="#Experimental_Model_of_Disease" />
          <owl:Class rdf:about="#Neoplastic_Process" />
          <owl:Class rdf:about="#Mental_or_Behavioral_Dysfunction" />
          <owl:Class rdf:about="#Cell_or_Molecular_Dysfunction" />
          <owl:Class rdf:about="#Pathologic_Function" />
          <owl:Class rdf:about="#Clinical_Attribute" />
          <owl:Class rdf:about="#Organism_Attribute" />
        </owl:unionOf>
      </owl:Class>
    </owl:allValuesFrom>
  </owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf >
  <owl:Restriction>
    <owl:onProperty rdf:resource="#co-occurs_with" />
    <owl:allValuesFrom>
      <owl:Class>
        <owl:unionOf rdf:parseType="Collection">
          <owl:Class rdf:about="#Experimental_Model_of_Disease" />
          <owl:Class rdf:about="#Neoplastic_Process" />
          <owl:Class rdf:about="#Mental_or_Behavioral_Dysfunction" />
          <owl:Class rdf:about="#Cell_or_Molecular_Dysfunction" />
          <owl:Class rdf:about="#Pathologic_Function" />
          <owl:Class rdf:about="#Injury_or_Poisoning" />
          <owl:Class rdf:about="#Congenital_Abnormality" />
          <owl:Class rdf:about="#Acquired_Abnormality" />
          <owl:Class rdf:about="#Anatomical_Abnormality" />
        </owl:unionOf>
      </owl:Class>
    </owl:allValuesFrom>
  </owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf >
  <owl:Restriction>
    <owl:onProperty rdf:resource="#complicates" />
    <owl:allValuesFrom>
      <owl:Class>
        <owl:unionOf rdf:parseType="Collection">
          <owl:Class rdf:about="#Experimental_Model_of_Disease" />
          <owl:Class rdf:about="#Neoplastic_Process" />
          <owl:Class rdf:about="#Mental_or_Behavioral_Dysfunction" />
          <owl:Class rdf:about="#Disease_or_Syndrome" />
          <owl:Class rdf:about="#Cell_or_Molecular_Dysfunction" />
          <owl:Class rdf:about="#Pathologic_Function" />
          <owl:Class rdf:about="#Injury_or_Poisoning" />
          <owl:Class rdf:about="#Congenital_Abnormality" />
          <owl:Class rdf:about="#Acquired_Abnormality" />
          <owl:Class rdf:about="#Anatomical_Abnormality" />
        </owl:unionOf>
      </owl:Class>
    </owl:allValuesFrom>
  </owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf >
  <owl:Restriction>
    <owl:onProperty rdf:resource="#degree_of" />
    <owl:allValuesFrom>
      <owl:Class>
        <owl:unionOf rdf:parseType="Collection">
          <owl:Class rdf:about="#Experimental_Model_of_Disease" />
          <owl:Class rdf:about="#Neoplastic_Process" />
          <owl:Class rdf:about="#Mental_or_Behavioral_Dysfunction" />
          <owl:Class rdf:about="#Disease_or_Syndrome" />
          <owl:Class rdf:about="#Cell_or_Molecular_Dysfunction" />
          <owl:Class rdf:about="#Pathologic_Function" />
        </owl:unionOf>
      </owl:Class>
    </owl:allValuesFrom>
  </owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf >
  <owl:Restriction>
    <owl:onProperty rdf:resource="#manifestation_of" />
    <owl:allValuesFrom>
      <owl:Class>
        <owl:unionOf rdf:parseType="Collection">
          <owl:Class rdf:about="#Mental_Process" />
          <owl:Class rdf:about="#Organism_Function" />
          <owl:Class rdf:about="#Organ_or_Tissue_Function" />
          <owl:Class rdf:about="#Genetic_Function" />
          <owl:Class rdf:about="#Molecular_Function" />
          <owl:Class rdf:about="#Cell_Function" />
          <owl:Class rdf:about="#Physiologic_Function" />
          <owl:Class rdf:about="#Experimental_Model_of_Disease" />
          <owl:Class rdf:about="#Neoplastic_Process" />
          <owl:Class rdf:about="#Mental_or_Behavioral_Dysfunction" />
          <owl:Class rdf:about="#Disease_or_Syndrome" />
          <owl:Class rdf:about="#Cell_or_Molecular_Dysfunction" />
          <owl:Class rdf:about="#Pathologic_Function" />
          <owl:Class rdf:about="#Injury_or_Poisoning" />
        </owl:unionOf>
      </owl:Class>
    </owl:allValuesFrom>
  </owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf >
  <owl:Restriction>
    <owl:onProperty rdf:resource="#occurs_in" />
    <owl:allValuesFrom>
```

```
      <owl:Class>
        <owl:unionOf rdf:parseType="Collection">
          <owl:Class rdf:about="#Injury_or_Poisoning" />
          <owl:Class rdf:about="#Professional_or_Occupational_Group" />
          <owl:Class rdf:about="#Population_Group" />
          <owl:Class rdf:about="#Patient_or_Disabled_Group" />
          <owl:Class rdf:about="#Family_Group" />
          <owl:Class rdf:about="#Age_Group" />
          <owl:Class rdf:about="#Group" />
          <owl:Class rdf:about="#Neoplastic_Process" />
          <owl:Class rdf:about="#Mental_or_Behavioral_Dysfunction" />
          <owl:Class rdf:about="#Disease_or_Syndrome" />
        </owl:unionOf>
      </owl:Class>
    </owl:allValuesFrom>
  </owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf >
  <owl:Restriction>
    <owl:onProperty rdf:resource="#precedes" />
    <owl:allValuesFrom>
      <owl:Class>
        <owl:unionOf rdf:parseType="Collection">
          <owl:Class rdf:about="#Experimental_Model_of_Disease" />
          <owl:Class rdf:about="#Neoplastic_Process" />
          <owl:Class rdf:about="#Mental_or_Behavioral_Dysfunction" />
          <owl:Class rdf:about="#Disease_or_Syndrome" />
          <owl:Class rdf:about="#Cell_or_Molecular_Dysfunction" />
          <owl:Class rdf:about="#Pathologic_Function" />
        </owl:unionOf>
      </owl:Class>
    </owl:allValuesFrom>
  </owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf >
  <owl:Restriction>
    <owl:onProperty rdf:resource="#result_of" />
    <owl:allValuesFrom>
      <owl:Class>
        <owl:unionOf rdf:parseType="Collection">
          <owl:Class rdf:about="#Mental_Process" />
          <owl:Class rdf:about="#Organism_Function" />
          <owl:Class rdf:about="#Organ_or_Tissue_Function" />
          <owl:Class rdf:about="#Genetic_Function" />
          <owl:Class rdf:about="#Molecular_Function" />
          <owl:Class rdf:about="#Cell_Function" />
          <owl:Class rdf:about="#Physiologic_Function" />
          <owl:Class rdf:about="#Experimental_Model_of_Disease" />
          <owl:Class rdf:about="#Neoplastic_Process" />
          <owl:Class rdf:about="#Mental_or_Behavioral_Dysfunction" />
          <owl:Class rdf:about="#Disease_or_Syndrome" />
          <owl:Class rdf:about="#Cell_or_Molecular_Dysfunction" />
          <owl:Class rdf:about="#Pathologic_Function" />
          <owl:Class rdf:about="#Biologic_Function" />
          <owl:Class rdf:about="#Natural_Phenomenon_or_Process" />
          <owl:Class rdf:about="#Injury_or_Poisoning" />
          <owl:Class rdf:about="#Environmental_Effect_of_Humans" />
          <owl:Class rdf:about="#Human-caused_Phenomenon_or_Process" />
          <owl:Class rdf:about="#Phenomenon_or_Process" />
          <owl:Class rdf:about="#Congenital_Abnormality" />
          <owl:Class rdf:about="#Acquired_Abnormality" />
          <owl:Class rdf:about="#Anatomical_Abnormality" />
          <owl:Class rdf:about="#Therapeutic_or_Preventive_Procedure" />
          <owl:Class rdf:about="#Laboratory_Procedure" />
          <owl:Class rdf:about="#Diagnostic_Procedure" />
          <owl:Class rdf:about="#Health_Care_Activity" />
          <owl:Class rdf:about="#Social_Behavior" />
          <owl:Class rdf:about="#Individual_Behavior" />
          <owl:Class rdf:about="#Behavior" />
        </owl:unionOf>
      </owl:Class>
    </owl:allValuesFrom>
  </owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf >
```

```
      <owl:Restriction>
        <owl:onProperty rdf:resource="#issue_in" />
        <owl:allValuesFrom>
          <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
              <owl:Class rdf:about="#Biomedical_Occupation_or_Discipline" />
              <owl:Class rdf:about="#Occupation_or_Discipline" />
            </owl:unionOf>
          </owl:Class>
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
  </owl:Class>
```

The OWL Description of 'Pharmacologic_Substance' STY:

```
<owl:Class rdf:ID="Pharmacologic_Substance"  >
    <rdfs:subClassOf >
      <owl:Class rdf:about="#Chemical_Viewed_Functionally" />
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Class rdf:about="#Chemical" />
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Class rdf:about="#Substance" />
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Class rdf:about="#Physical_Object" />
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Class rdf:about="#Entity" />
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Restriction>
        <owl:onProperty rdf:resource="#complicates" />
        <owl:allValuesFrom>
          <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
              <owl:Class rdf:about="#Injury_or_Poisoning" />
              <owl:Class rdf:about="#Mental_Process" />
              <owl:Class rdf:about="#Organism_Function" />
              <owl:Class rdf:about="#Organ_or_Tissue_Function" />
              <owl:Class rdf:about="#Genetic_Function" />
              <owl:Class rdf:about="#Molecular_Function" />
              <owl:Class rdf:about="#Cell_Function" />
              <owl:Class rdf:about="#Physiologic_Function" />
              <owl:Class rdf:about="#Experimental_Model_of_Disease" />
              <owl:Class rdf:about="#Neoplastic_Process" />
              <owl:Class rdf:about="#Mental_or_Behavioral_Dysfunction" />
              <owl:Class rdf:about="#Disease_or_Syndrome" />
              <owl:Class rdf:about="#Cell_or_Molecular_Dysfunction" />
              <owl:Class rdf:about="#Pathologic_Function" />
              <owl:Class rdf:about="#Biologic_Function" />
              <owl:Class rdf:about="#Congenital_Abnormality" />
              <owl:Class rdf:about="#Acquired_Abnormality" />
              <owl:Class rdf:about="#Anatomical_Abnormality" />
            </owl:unionOf>
          </owl:Class>
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Restriction>
        <owl:onProperty rdf:resource="#diagnoses" />
        <owl:allValuesFrom>
          <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
              <owl:Class rdf:about="#Experimental_Model_of_Disease" />
              <owl:Class rdf:about="#Neoplastic_Process" />
              <owl:Class rdf:about="#Mental_or_Behavioral_Dysfunction" />
              <owl:Class rdf:about="#Disease_or_Syndrome" />
              <owl:Class rdf:about="#Cell_or_Molecular_Dysfunction" />
              <owl:Class rdf:about="#Pathologic_Function" />
            </owl:unionOf>
          </owl:Class>
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Restriction>
        <owl:onProperty rdf:resource="#disrupts" />
        <owl:allValuesFrom>
          <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
              <owl:Class rdf:about="#Mental_Process" />
              <owl:Class rdf:about="#Organism_Function" />
              <owl:Class rdf:about="#Organ_or_Tissue_Function" />
              <owl:Class rdf:about="#Genetic_Function" />
              <owl:Class rdf:about="#Molecular_Function" />
              <owl:Class rdf:about="#Cell_Function" />
              <owl:Class rdf:about="#Physiologic_Function" />
              <owl:Class rdf:about="#Tissue" />
              <owl:Class rdf:about="#Gene_or_Genome" />
              <owl:Class rdf:about="#Cell" />
              <owl:Class rdf:about="#Cell_Component" />
              <owl:Class rdf:about="#Body_Part_Organ_or_Organ_Component" />
              <owl:Class rdf:about="#Fully_Formed_Anatomical_Structure" />
              <owl:Class rdf:about="#Embryonic_Structure" />
            </owl:unionOf>
          </owl:Class>
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Restriction>
        <owl:onProperty rdf:resource="#prevents" />
        <owl:allValuesFrom>
          <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
              <owl:Class rdf:about="#Experimental_Model_of_Disease" />
              <owl:Class rdf:about="#Neoplastic_Process" />
              <owl:Class rdf:about="#Mental_or_Behavioral_Dysfunction" />
              <owl:Class rdf:about="#Disease_or_Syndrome" />
              <owl:Class rdf:about="#Cell_or_Molecular_Dysfunction" />
              <owl:Class rdf:about="#Pathologic_Function" />
            </owl:unionOf>
          </owl:Class>
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Restriction>
        <owl:onProperty rdf:resource="#treats" />
        <owl:allValuesFrom>
          <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
              <owl:Class rdf:about="#Sign_or_Symptom" />
              <owl:Class rdf:about="#Experimental_Model_of_Disease" />
              <owl:Class rdf:about="#Neoplastic_Process" />
              <owl:Class rdf:about="#Mental_or_Behavioral_Dysfunction" />
              <owl:Class rdf:about="#Disease_or_Syndrome" />
              <owl:Class rdf:about="#Cell_or_Molecular_Dysfunction" />
              <owl:Class rdf:about="#Pathologic_Function" />
              <owl:Class rdf:about="#Injury_or_Poisoning" />
              <owl:Class rdf:about="#Congenital_Abnormality" />
              <owl:Class rdf:about="#Acquired_Abnormality" />
              <owl:Class rdf:about="#Anatomical_Abnormality" />
            </owl:unionOf>
          </owl:Class>
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Restriction>
        <owl:onProperty rdf:resource="#affects" />
        <owl:allValuesFrom>
          <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
              <owl:Class rdf:about="#Mental_Process" />
              <owl:Class rdf:about="#Organism_Function" />
              <owl:Class rdf:about="#Organ_or_Tissue_Function" />
              <owl:Class rdf:about="#Genetic_Function" />
              <owl:Class rdf:about="#Molecular_Function" />
              <owl:Class rdf:about="#Cell_Function" />
              <owl:Class rdf:about="#Physiologic_Function" />
              <owl:Class rdf:about="#Experimental_Model_of_Disease" />
              <owl:Class rdf:about="#Neoplastic_Process" />
              <owl:Class rdf:about="#Mental_or_Behavioral_Dysfunction" />
              <owl:Class rdf:about="#Disease_or_Syndrome" />
              <owl:Class rdf:about="#Cell_or_Molecular_Dysfunction" />
              <owl:Class rdf:about="#Pathologic_Function" />
              <owl:Class rdf:about="#Biologic_Function" />
              <owl:Class rdf:about="#Natural_Phenomenon_or_Process" />
            </owl:unionOf>
          </owl:Class>
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Restriction>
        <owl:onProperty rdf:resource="#interacts_with" />
        <owl:allValuesFrom>
          <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
              <owl:Class rdf:about="#Antibiotic" />
              <owl:Class rdf:about="#Pharmacologic_Substance" />
              <owl:Class rdf:about="#Indicator_Reagent_or_Diagnostic_Aid" />
              <owl:Class rdf:about="#Hazardous_or_Poisonous_Substance" />
              <owl:Class rdf:about="#Biomedical_or_Dental_Material" />
              <owl:Class rdf:about="#Vitamin" />
              <owl:Class rdf:about="#Receptor" />
              <owl:Class rdf:about="#Neuroreactive_Substance_or_Biogenic_Amine"/>
              <owl:Class rdf:about="#Immunologic_Factor" />
              <owl:Class rdf:about="#Hormone" />
              <owl:Class rdf:about="#Enzyme" />
              <owl:Class rdf:about="#Biologically_Active_Substance" />
              <owl:Class rdf:about="#Chemical" />
            </owl:unionOf>
          </owl:Class>
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Restriction>
        <owl:onProperty rdf:resource="#causes" />
        <owl:allValuesFrom>
          <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
              <owl:Class rdf:about="#Experimental_Model_of_Disease" />
              <owl:Class rdf:about="#Neoplastic_Process" />
              <owl:Class rdf:about="#Mental_or_Behavioral_Dysfunction" />
              <owl:Class rdf:about="#Disease_or_Syndrome" />
              <owl:Class rdf:about="#Cell_or_Molecular_Dysfunction" />
              <owl:Class rdf:about="#Pathologic_Function" />
              <owl:Class rdf:about="#Injury_or_Poisoning" />
              <owl:Class rdf:about="#Congenital_Abnormality" />
              <owl:Class rdf:about="#Acquired_Abnormality" />
              <owl:Class rdf:about="#Anatomical_Abnormality" />
            </owl:unionOf>
          </owl:Class>
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Restriction>
        <owl:onProperty rdf:resource="#ingredient_of" />
        <owl:allValuesFrom>
          <owl:Class rdf:about="#Clinical_Drug" />
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Restriction>
        <owl:onProperty rdf:resource="#issue_in" />
```

```
        <owl:allValuesFrom>
          <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
              <owl:Class rdf:about="#Biomedical_Occupation_or_Discipline" />
              <owl:Class rdf:about="#Occupation_or_Discipline" />
            </owl:unionOf>
          </owl:Class>
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
  </owl:Class>
```

The OWL Description of ‘Antibiotic’ STY:

```
<owl:Class rdf:ID="Antibiotic"  >
    <rdfs:subClassOf >
      <owl:Class rdf:about="#Pharmacologic_Substance" />
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Class rdf:about="#Chemical_Viewed_Functionally" />
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Class rdf:about="#Chemical" />
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Class rdf:about="#Substance" />
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Class rdf:about="#Physical_Object" />
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Class rdf:about="#Entity" />
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Restriction>
        <owl:onProperty rdf:resource="#complicates" />
        <owl:allValuesFrom>
          <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
              <owl:Class rdf:about="#Injury_or_Poisoning" />
              <owl:Class rdf:about="#Mental_Process" />
              <owl:Class rdf:about="#Organism_Function" />
              <owl:Class rdf:about="#Organ_or_Tissue_Function" />
              <owl:Class rdf:about="#Genetic_Function" />
              <owl:Class rdf:about="#Molecular_Function" />
              <owl:Class rdf:about="#Cell_Function" />
              <owl:Class rdf:about="#Physiologic_Function" />
              <owl:Class rdf:about="#Experimental_Model_of_Disease" />
              <owl:Class rdf:about="#Neoplastic_Process" />
              <owl:Class rdf:about="#Mental_or_Behavioral_Dysfunction" />
              <owl:Class rdf:about="#Disease_or_Syndrome" />
              <owl:Class rdf:about="#Cell_or_Molecular_Dysfunction" />
              <owl:Class rdf:about="#Pathologic_Function" />
              <owl:Class rdf:about="#Biologic_Function" />
              <owl:Class rdf:about="#Congenital_Abnormality" />
              <owl:Class rdf:about="#Acquired_Abnormality" />
              <owl:Class rdf:about="#Anatomical_Abnormality" />
            </owl:unionOf>
          </owl:Class>
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Restriction>
        <owl:onProperty rdf:resource="#diagnoses" />
        <owl:allValuesFrom>
          <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
              <owl:Class rdf:about="#Experimental_Model_of_Disease" />
              <owl:Class rdf:about="#Neoplastic_Process" />
              <owl:Class rdf:about="#Mental_or_Behavioral_Dysfunction" />
              <owl:Class rdf:about="#Disease_or_Syndrome" />
              <owl:Class rdf:about="#Cell_or_Molecular_Dysfunction" />
              <owl:Class rdf:about="#Pathologic_Function" />
            </owl:unionOf>
          </owl:Class>
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Restriction>
        <owl:onProperty rdf:resource="#disrupts" />
        <owl:allValuesFrom>
          <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
              <owl:Class rdf:about="#Mental_Process" />
              <owl:Class rdf:about="#Organism_Function" />
              <owl:Class rdf:about="#Organ_or_Tissue_Function" />
              <owl:Class rdf:about="#Genetic_Function" />
              <owl:Class rdf:about="#Molecular_Function" />
              <owl:Class rdf:about="#Cell_Function" />
              <owl:Class rdf:about="#Physiologic_Function" />
              <owl:Class rdf:about="#Tissue" />
              <owl:Class rdf:about="#Gene_or_Genome" />
              <owl:Class rdf:about="#Cell" />
              <owl:Class rdf:about="#Cell_Component" />
              <owl:Class rdf:about="#Body_Part_Organ_or_Organ_Component" />
              <owl:Class rdf:about="#Fully_Formed_Anatomical_Structure" />
              <owl:Class rdf:about="#Embryonic_Structure" />
            </owl:unionOf>
          </owl:Class>
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Restriction>
        <owl:onProperty rdf:resource="#prevents" />
        <owl:allValuesFrom>
          <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
              <owl:Class rdf:about="#Experimental_Model_of_Disease" />
              <owl:Class rdf:about="#Neoplastic_Process" />
              <owl:Class rdf:about="#Mental_or_Behavioral_Dysfunction" />
              <owl:Class rdf:about="#Disease_or_Syndrome" />
              <owl:Class rdf:about="#Cell_or_Molecular_Dysfunction" />
              <owl:Class rdf:about="#Pathologic_Function" />
            </owl:unionOf>
          </owl:Class>
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Restriction>
        <owl:onProperty rdf:resource="#treats" />
        <owl:allValuesFrom>
          <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
              <owl:Class rdf:about="#Sign_or_Symptom" />
              <owl:Class rdf:about="#Experimental_Model_of_Disease" />
              <owl:Class rdf:about="#Neoplastic_Process" />
              <owl:Class rdf:about="#Mental_or_Behavioral_Dysfunction" />
              <owl:Class rdf:about="#Disease_or_Syndrome" />
              <owl:Class rdf:about="#Cell_or_Molecular_Dysfunction" />
              <owl:Class rdf:about="#Pathologic_Function" />
              <owl:Class rdf:about="#Injury_or_Poisoning" />
              <owl:Class rdf:about="#Congenital_Abnormality" />
              <owl:Class rdf:about="#Acquired_Abnormality" />
              <owl:Class rdf:about="#Anatomical_Abnormality" />
            </owl:unionOf>
          </owl:Class>
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Restriction>
        <owl:onProperty rdf:resource="#affects" />
        <owl:allValuesFrom>
          <owl:Class>
```

```
        <owl:unionOf rdf:parseType="Collection">
          <owl:Class rdf:about="#Mental_Process" />
          <owl:Class rdf:about="#Organism_Function" />
          <owl:Class rdf:about="#Organ_or_Tissue_Function" />
          <owl:Class rdf:about="#Genetic_Function" />
          <owl:Class rdf:about="#Molecular_Function" />
          <owl:Class rdf:about="#Cell_Function" />
          <owl:Class rdf:about="#Physiologic_Function" />
          <owl:Class rdf:about="#Experimental_Model_of_Disease" />
          <owl:Class rdf:about="#Neoplastic_Process" />
          <owl:Class rdf:about="#Mental_or_Behavioral_Dysfunction" />
          <owl:Class rdf:about="#Disease_or_Syndrome" />
          <owl:Class rdf:about="#Cell_or_Molecular_Dysfunction" />
          <owl:Class rdf:about="#Pathologic_Function" />
          <owl:Class rdf:about="#Biologic_Function" />
          <owl:Class rdf:about="#Natural_Phenomenon_or_Process" />
        </owl:unionOf>
      </owl:Class>
    </owl:allValuesFrom>
  </owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf >
  <owl:Restriction>
    <owl:onProperty rdf:resource="#interacts_with" />
    <owl:allValuesFrom>
      <owl:Class>
        <owl:unionOf rdf:parseType="Collection">
          <owl:Class rdf:about="#Antibiotic" />
          <owl:Class rdf:about="#Indicator_Reagent_or_Diagnostic_Aid" />
          <owl:Class rdf:about="#Hazardous_or_Poisonous_Substance" />
          <owl:Class rdf:about="#Biomedical_or_Dental_Material" />
          <owl:Class rdf:about="#Vitamin" />
          <owl:Class rdf:about="#Receptor" />
          <owl:Class rdf:about="#Neuroreactive_Substance_or_Biogenic_Amine"/>
          <owl:Class rdf:about="#Immunologic_Factor" />
          <owl:Class rdf:about="#Hormone" />
          <owl:Class rdf:about="#Enzyme" />
          <owl:Class rdf:about="#Chemical" />
        </owl:unionOf>
      </owl:Class>
    </owl:allValuesFrom>
  </owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf >
  <owl:Restriction>
    <owl:onProperty rdf:resource="#causes" />
    <owl:allValuesFrom>
      <owl:Class>
        <owl:unionOf rdf:parseType="Collection">
          <owl:Class rdf:about="#Experimental_Model_of_Disease" />
          <owl:Class rdf:about="#Neoplastic_Process" />
          <owl:Class rdf:about="#Mental_or_Behavioral_Dysfunction" />
          <owl:Class rdf:about="#Disease_or_Syndrome" />
          <owl:Class rdf:about="#Cell_or_Molecular_Dysfunction" />
          <owl:Class rdf:about="#Pathologic_Function" />
          <owl:Class rdf:about="#Injury_or_Poisoning" />
          <owl:Class rdf:about="#Congenital_Abnormality" />
          <owl:Class rdf:about="#Acquired_Abnormality" />
          <owl:Class rdf:about="#Anatomical_Abnormality" />
        </owl:unionOf>
      </owl:Class>
    </owl:allValuesFrom>
  </owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf >
  <owl:Restriction>
    <owl:onProperty rdf:resource="#ingredient_of" />
    <owl:allValuesFrom>
      <owl:Class rdf:about="#Clinical_Drug" />
    </owl:allValuesFrom>
  </owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf >
  <owl:Restriction>
    <owl:onProperty rdf:resource="#issue_in" />
```

```
        <owl:allValuesFrom>
          <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
              <owl:Class rdf:about="#Biomedical_Occupation_or_Discipline" />
              <owl:Class rdf:about="#Occupation_or_Discipline" />
            </owl:unionOf>
          </owl:Class>
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
  </owl:Class>
```

The OWL Description of 'Temporal_Concept' STY:

```
<owl:Class rdf:ID="Temporal_Concept" >
    <rdfs:subClassOf  >
      <owl:Class rdf:about="#Idea_or_Concept" />
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Class rdf:about="#Conceptual_Entity" />
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Class rdf:about="#Entity" />
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Restriction>
        <owl:onProperty rdf:resource="#conceptual_part_of" />
        <owl:allValuesFrom>
          <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
              <owl:Class rdf:about="#Temporal_Concept" />
              <owl:Class rdf:about="#Organism_Function" />
              <owl:Class rdf:about="#Organ_or_Tissue_Function" />
              <owl:Class rdf:about="#Cell_Function" />
            </owl:unionOf>
          </owl:Class>
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Restriction>
        <owl:onProperty rdf:resource="#result_of" />
        <owl:allValuesFrom>
          <owl:Class rdf:about="#Mental_Process" />
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf >
      <owl:Restriction>
        <owl:onProperty rdf:resource="#issue_in" />
        <owl:allValuesFrom>
          <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
              <owl:Class rdf:about="#Biomedical_Occupation_or_Discipline" />
              <owl:Class rdf:about="#Occupation_or_Discipline" />
            </owl:unionOf>
          </owl:Class>
        </owl:allValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
  </owl:Class>
```

The OWL Description of 'Part_of' RL:

```
<owl:ObjectProperty rdf:ID="part_of" >
  <rdfs:subPropertyOf rdf:resource="#physically_related_to"/>
  <rdfs:subPropertyOf rdf:resource="#associated_with"/>
  <owl:inverseOf rdf:resource="#has_part" />
</owl:ObjectProperty>
```

CURRICULUM VITAE

PERSONAL INFORMATION

Surname, Name:	Gülden Özdemir, Birsen
Nationality:	Turkish (TC)
Date and Place of Birth:	June 6, 1975, Filibe
Marital Status:	Married
Phone:	+ 90 216 544 5555/1238
Fax:	+ 90 216 544 55 35
E-mail:	bgozdemir@dogus.edu.tr birsengulden@gmail.com

EDUCATION

Degree	**Institution**	**Year of Graduation**
MS	METU Information Systems	2004
MS	METU Food Engineering	2002
BS	METU Food Engineering	1999

WORK EXPERIENCE

Year	**Place**	**Enrollment**
Sep2007-Present	Doğuş University, Department of Computer Engineering	Instructor
Dec2007-Present	Doğuş University	Course Program Commission, Chief Executive

Nov2006- May2007	METU, Informatics Institute, AVICENNA Project	Course Material Developer
2000-2006	METU, Department of Information Systems	Research Assistant

PUBLICATIONS

O. Saka, KH. Gülkesen, B. Gülden & OD. Koçgil.” Evaluation of two search methods in PubMed; the Regular Search and Search by MeSH Terms”, ***Acta Informatica Medica: Journal of Society for Medical Informatics of B&H***, December2005, vol. 13, no. 4, pp. 180-183.

B. Gülden, E. Mumcuoğlu & N. Baykal, “A GIS System for Ambulatory Transportation”, ***IASTED International Conference on Biomedical Engineering***, Innsbruck, Austria, February 2004.

B. Gülden, E. Mumcuoğlu & N. Baykal, “A GIS Application”, ***BIOMED 2003 - 10th Biomedical Science and Technology Symposium***, Northern Cyprus, October 2003.

B.Gülden, P.Varoquaux, F.Yıldız, “Effects of High Oxygen Partial Pressure on the Respiration rates of the Apple slices(Malus sylvestris) and Mushroom(Agaricus bisporus)” ***Food Technology****,* 57, 2003, p.1.

RESEARCH INTEREST

Ontology Engineering, Semantic Web, Biomedical Applications