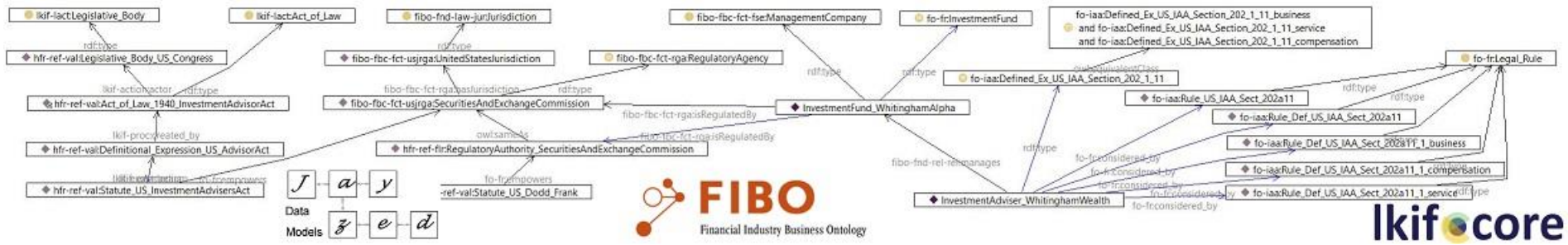
Financial Regulation Ontology

Tutorial chapter one – preface and introduction

Data Models

> “Alignment of Legal and Finance is the foundation for the Semantic Web approach to compliance.”

Jurgen Ziemer, Jayzed Data Models Inc., http://finregont.com



Compliance overwhelms Financial Institutions

Increasing complexity

The regulatory regime for Financial Institutions and Funds has tightened worldwide:
US Banks are subject to Capital Adequacy, Holding Company and Prudential Standards regulations.
The Dodd Frank Act brought Hedge Funds under supervision of the Securities & Exchange Commission . European countries implemented the Alternative Investment Manager Directive.

approached with

more systems, artifacts and staff

Consolidated Life New York offices, 1960. (Jack Lemon in Billy Wilder's "The Apartment")

=

Conventional compliance architecture is entangled in numerous systems, transformations, and mappings. At major banks each new compliance program brings more systems, software, warehouses, and literally hundreds of Word documents, spreadsheets, and PowerPoints. Hundreds of people compiling **heterogeneous artifacts**[1] and try to main consistency between them.

Neither the business, nor regulators have trust and proof that the code accurately implements the logic of the rules, and that reported numbers accurately trace back to their data sources.

Bio and Medical domain have mastered complexity

Bio/Medical is more complex than Finance

The human genome contains some 30,000 genes, approximately 3 billion DNA base pairs. Genes direct the production of over a million analyzed proteins.
More than 9500 terms define human phenotype anomalies, which describe over10, 000 diseases. Medical service providers exchange detailed clinical information. Almost half a million drugs are approved for treatment.

resolved with

Semantic Web and Ontology

The Semantic Web is the evolution from the old web of documents to a web of data that enables computers to navigate content and derive information.
The GENE ontology project started in 1998 with yeast and fruit flies.
"Our **vision** is that all biomedical knowledge and data are disseminated on the Internet using principled ontologies in such a way that the knowledge and data are semantically interoperable and useful for furthering biomedical science and clinical care." (National Center for Biomedical Ontology[1] (NCBO)
Today, the NCBO's Bio Portal alone lists 530 ontologies with millions of classes.
Research, clinical and increasingly medical information is stored and published in a homogeneous way.

[1] http://www.bioontology.org/about-ncbo

Finance can master the compliance challenge.

Financial data, regulations, reports, and metadata can be stored in a uniform way. The Ontology Web Language (OWL) defines the semantics of concepts, their relationships, and axioms. Compliance crosses the domains or Finance and Legal. We select the best domain reference ontologies.

"The Financial Industry Business Ontology (FIBO) is a collaboration between the Enterprise Data Management Council (EDMC) and the Object Management Group. The EDMC leads design in collaboration with major Financial Institutions. OMG provides governance and publishes FIBO as a formal standard."[2]

"The Legal Knowledge Interchange Format (LKIF) models legal rules of the kind found in legislation and regulations."[3]
We populated LKIF with the Code of Federal Regulation (CFR) and the United State Code (USC) titles related to investment and banking industry.

The Financial Regulation Ontology (FRO) extends and aligns the two domain reference ontologies. Extensions define subclasses to specific financial and regulatory data.
Ontology Alignment establishes the relationship of equivalent concepts in FIBO and LKIF. It enables queries and reasoning across the domains.

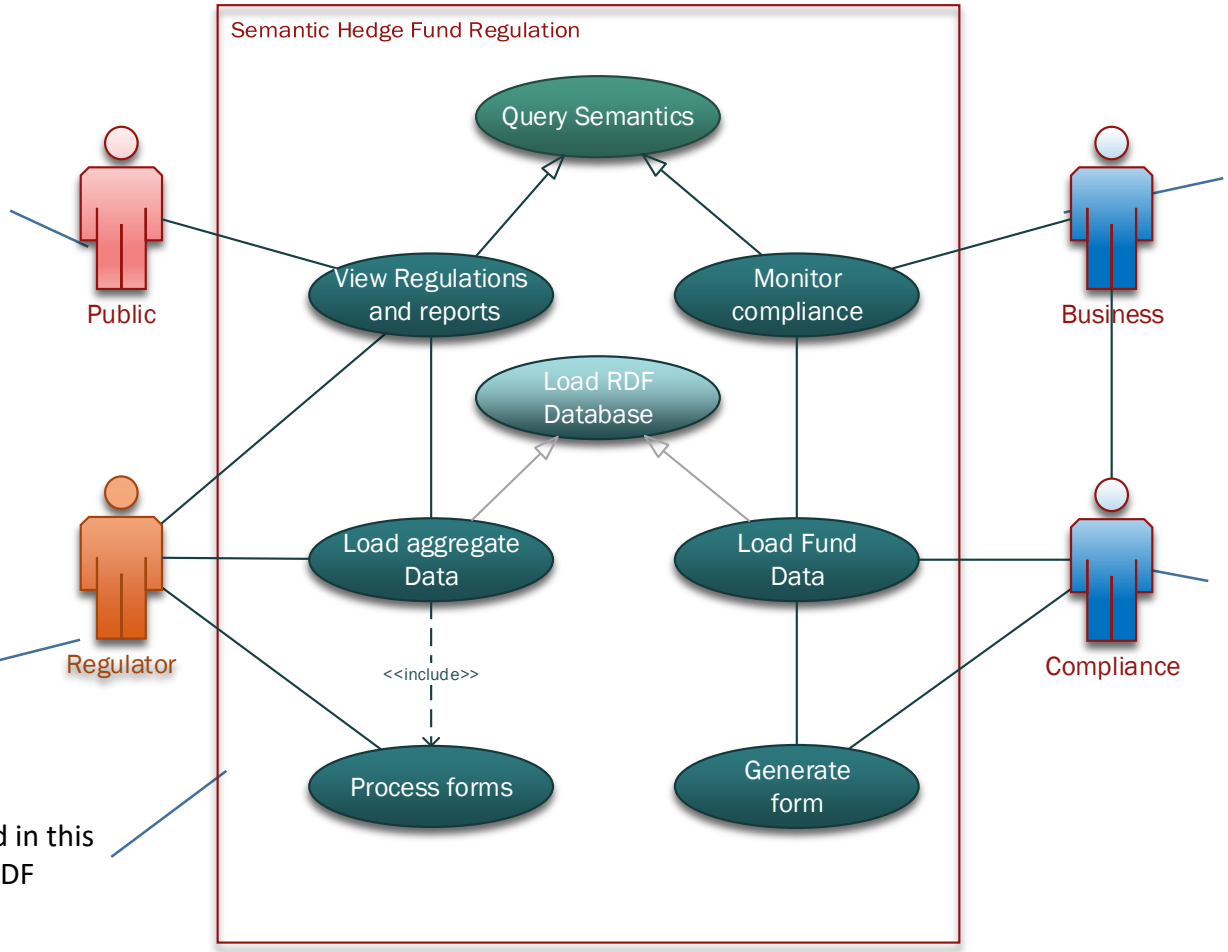
[2] http://www.edmcouncil.org/financialbusiness
[3] http://www.estrellaproject.org/

Tutorial sections for specific use cases.

The Securities and Exchange Commission (SEC) provides investors with Adviser data in XML. Publication in OWL on a semantic endpoint would serve the public even better and facilitate queries across other information sources. The tutorial introduction explains ontology fundamentals.

Chapter II "Loading the Law" explains, how to design ontology classes and how extract, transform and load data from the XML.

Note:
The Use "Load RDF Database is not covered in this tutorial. We highly recommend Cure/Blin RDF Database Systems (2015).



The fund manager monitors compliance and can assess the impact of changes to fund structure, investments and client base. Chapter I has a Finance example based ontology primer and business overview of the reference ontologies..

The fund's compliance officer can populate forms using inference and SPARQL rules. The RDF database of the fund holds internal financial data, positions, investors, structure. Chapter III shows ontology inference and rules in practice.

Implementing Financial Regulation Ontology

This slide-doc tutorial explains FRO, LKIF and FIBO design, and how to extend the ontologies for data population. We recommend to study the text end-to-end. Even beginner sections show finance related classes and data.

1 Introduction

The first chapter introduces Ontology Web Language (OWL) to business and the beginning ontologist. The example “Black Rock manages Emerging Markets ETF” is a good FIBO use case.

Getting Started continues with step-by-step instructions for Ontology Editor, Protégé and Query tools.

FRO foundations explains the core classes relevant to regulatory compliance.

2 Loading the law

Chapter two shows how to populate FRO from XML files for law and regulations. The details of LKIF Legal Document are mainly for government agencies, who want to make correction or add more regulations and laws.

However, loaded content is basis for Legal reasoning in chapter three and process and design is similar for all data sources. All metadata, mapping, and lineage are stored within the ontology

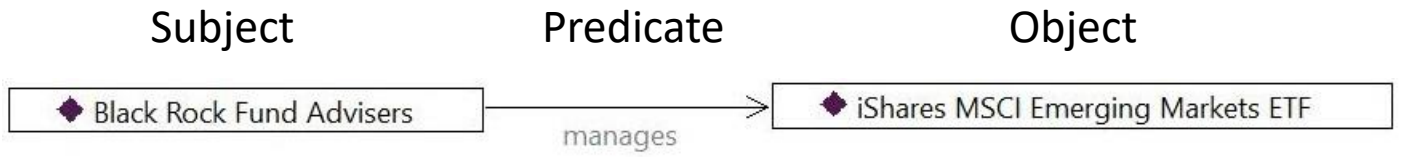
3 Legal Reasoning

Inferencing is the process to derive new knowledge from asserted facts. The FRO example infers, whether an Investment Funds must register with the Securities & Exchange Commission (SEC).

Chapter three starts with a deep-dive into LKIF Legal Expressions and how FRO defined classes encode SEC mandate and exceptions.

FRO tutorial chapter I - Introduction

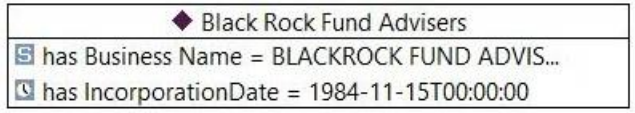
The Resource Description Framework (RDF) uses an elementary grammar to store information in **triples**.



Black Rock manages the iShares MSCI Emerging Markets ETF. Both are web resources identified by a URI.
The **Object Property "***manages"* connects the two resources.
This is similar to creating

- an association between objects. E.g. associating two Java instances.
- a Foreign Key reference between two database records.

The **Data Property** assigns a value to the subject resource.
Here is the Ontology Web Language (OWL) code. The colon separates the Namespace from the resource name.
rdfs:label - a build in RDF data property
fibo-fnd-rel-rel:manage – an object property defined in the Financial Services Business Ontology (FIBO).



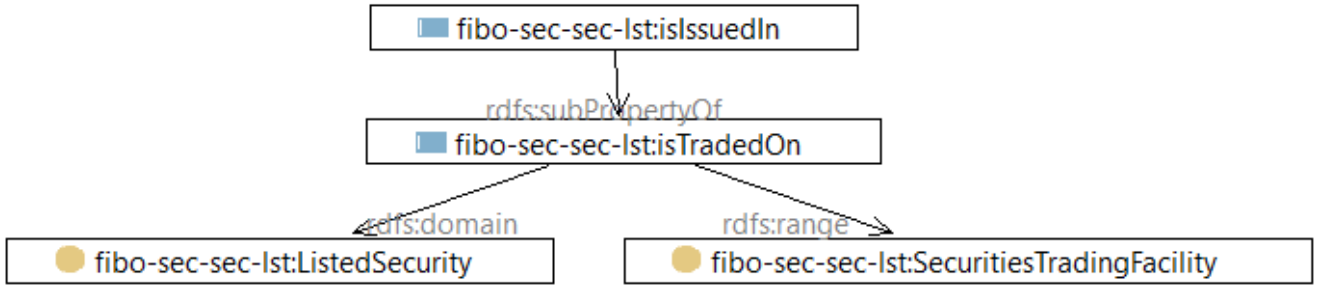

```
:Black_Rock_Fund_Advisers
  rdf:type fo-fr:InvestmentAdviser ;
  :hasBusinessName "BLACKROCK FUND ADVISORS"^^xsd:string ;
  :hasIncorporationDate "1984-11-15T00:00:00"^^xsd:dateTime ;
  fibo-fnd-rel-rel:manages :iShares_MSCI_Emerging_Markets_ETF ;
  rdfs:label "Black Rock Fund Advisers"^^xsd:string ;
```

Triples define the schema

RDF Schema (RDFS) extends RDF to allow describing taxonomies of classes and properties.

- OWL classes specify sets.
  A FIBO class defines Management Companies.
  FRO has a class for Investment Advisers.
- Instances or Individuals are the members of the class. The RDF object property rdf:type asserts that the individual is an instance of the class. E.g. Black Rock is an Investment Adviser.

We can define data and object properties of the class. The RDFS property's Domain restricts the type of the subject, Range restricts the type of the object. We can populate the schema with "iShares Emerging Markets is traded on the New York Stock Exchange".

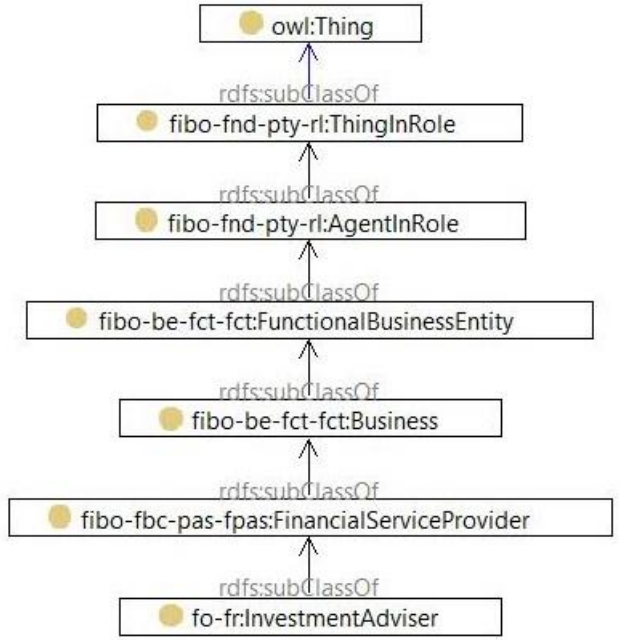


Triples define hierarchies and taxonomies

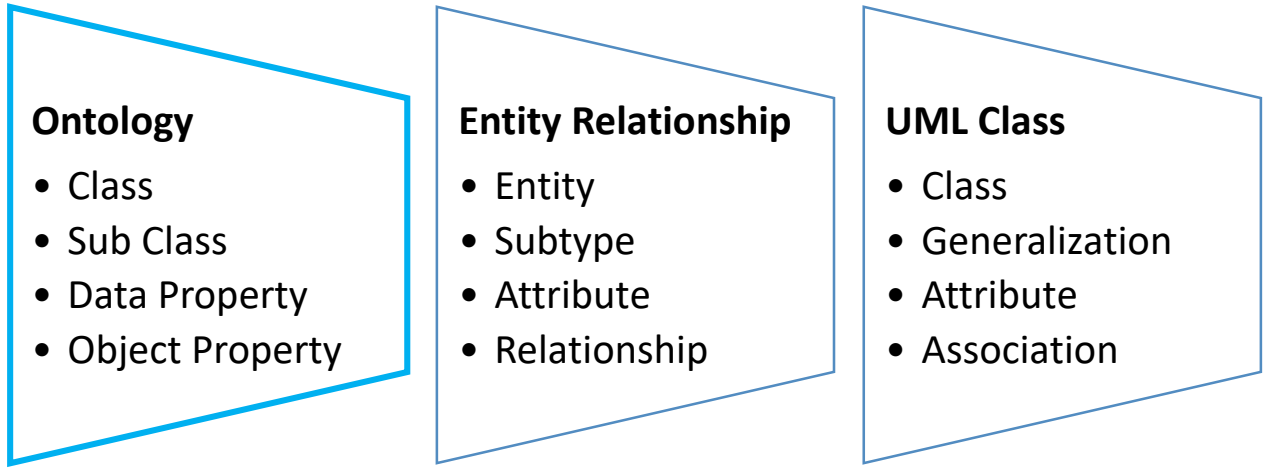
RDF Schema (RDFS) extends RDF to allow describing taxonomies of classes and properties.

The RDF-schema property subClassOf enables rich hierarchies of classes.
The subPropertyOf build hierarchies of class associations:
The *FRO Investment Adviser* is a subclass of a FIBO Financial Service Provider, which in turn is a subclass of *Agent in Role*. The owl:Thing is the root of all classes.

The core difference between RF and object models is the notion subtyping vs. inheritance.
In Java we must explicitly assign Black Rock to be an object of Investment Adviser. The object inherits methods of the base class, but does not become an object of it.
Likewise, in a Logical Data Model we can create a hierarchy subtype entities. However in the database, we must explicitly insert records into all hierarchy tables and create foreign key constraints.



Metadata is stored in triples.



RDFS provides constructs similar to UML class / object and data modeling Entity Relationship models.

Modeling tools provide model transformations, import and export between OWL, ER and UML.

- Data modelers still need to derive the Physical model from the Logical Model.
- Object modelers generate code from UML

The ontology already contains schema and data expressed in triples and we can use metadata in our SPARQL queries.

```
SELECT ?investment_adviser ?management_company ?FS_provider
?ETF_class
WHERE {
    ?investment_adviser fibo-fnd-rel-rel:manages ?managementcompany .
    ?investment_adviser rdf:type ?FS_provider .
    ?management_company rdf:type ?class .
    ?management_company rdf:type ?ETF_class .
    ?FS_provider rdfs:subClassOf fibo-fbc-pas-
fpas:FinancialServiceProvider .
}
```

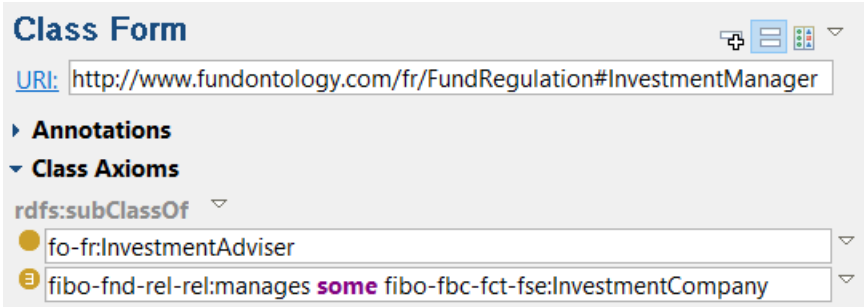
[investmentadviser]	Black_Rock_Fund_Advisers
[managementcompany]	iShares_MSCI_Emerging_Markets_ETF
FS_provider	fo-fr:InvestmentAdviser
ETF_class	fibo-fbc-fct-fse:ManagementCompany

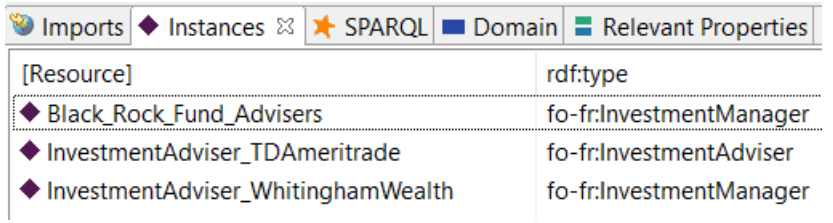
Class restrictions limit the set of individuals

A restriction describes a class of individuals based on the properties that instances of the class participate in. In other words a restriction is a kind of class, just like a named class. Typically, we apply restrictions to narrow down the set of instances from class to subclass.

An Investment Manager is an Investment Adviser, who manages some Investment Company. In the ontology editor (TopBraid Composer) we define the **existential restriction (some)** on the object property. The **universal restriction (only)** would mean that the adviser manages nothing else but Investment Companies. We can place restrictions on Quantifiers, Cardinalities and even Values. An example of a Value restriction is a Prime Bond with hasRating **only** 'AAA'.

Our example has Black Rock and Whitingham Wealth as Investment Advisers.
TD Ameritrade doesn't manage funds.

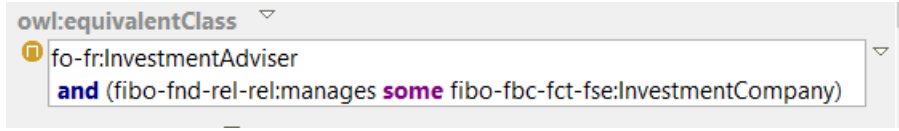




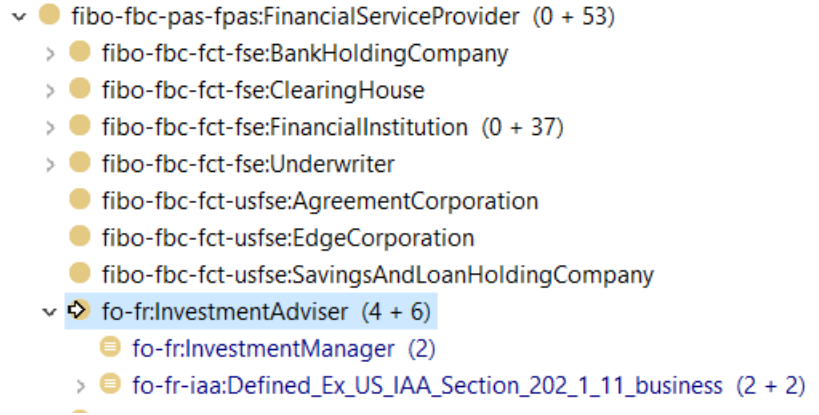
Defined Class drives the *Reasoner*

So far, the class restrictions although more refined than database constraints still depend on the implementation to assert that Black Rock is an Investment Manager and TDAmeritrade is only an Investment Adviser.
Changing Investment Manager to a **Defined Class** will have the Reasoner infer this new knowledge from existing asserted information.

We change the class restriction from rdfs:subTypeOf to owl:equivalent class.
Now the Investment Manager is defined as the intersection of all Investment Advisers **and** all things that manage an Investment Company.
The Reasoner tool processes class definitions and infers that a particular individual is a member of the class.



The class hierarchy now shows the defined classes in blue and we see two inferred instances: Black Rock and Whitingham Wealth.
Checking for class **subsumption** is a key task of the reasoner key differentiator to non-semantic technologies.



We formulate complex compliance rules as Defined Classes from class restriction building blocks.

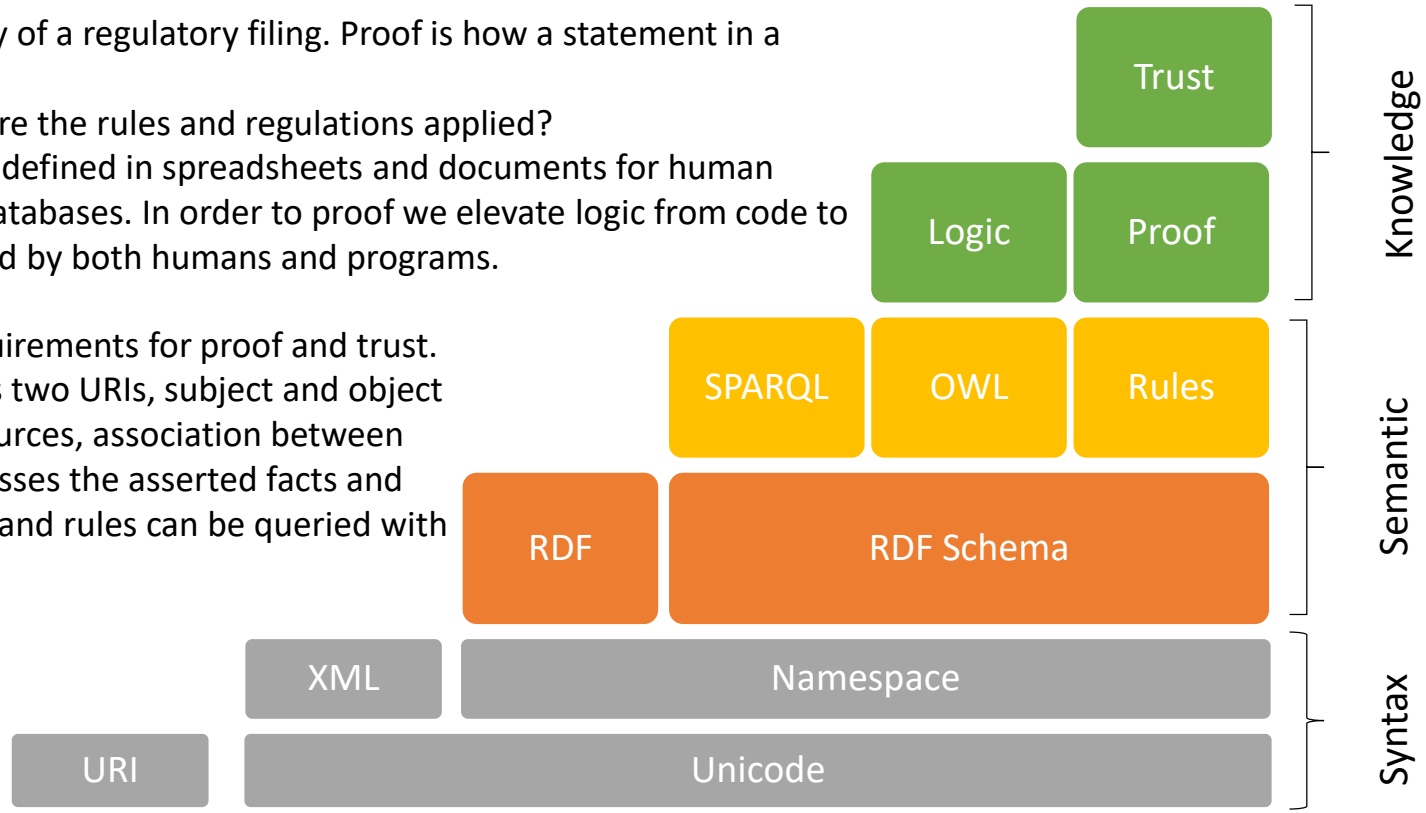
Semantic Web has 3 main layers

"The Semantic Web is an extension of the current Web in which information is given well-defined meaning, better enabling computers and people to work in cooperation." Sir Tim Berners-Lee, director or the World Wide Web Consortium (W3C)[4]

Trust is the firm belief in the integrity, reliability as accuracy of a regulatory filing. Proof is how a statement in a regulatory filing as been derived.
What are the underlying sources of Financial Data? What are the rules and regulations applied?
In traditional approaches to regulations Business Rules are defined in spreadsheets and documents for human readers. They get mapped and encoded in programs and databases. In order to proof we elevate logic from code to schema. So that a single artifact is understood and validated by both humans and programs.

The Semantic layer build upon the foundation to fulfill requirements for proof and trust. RDF removes ambiguity from XML. Unequivocal it connects two URIs, subject and object with a predicate. RDFS and OWL define classes of RDF resources, association between classes and class restrictions. The ontology Reasoner processes the asserted facts and infers new information. All information in including results and rules can be queried with SPARQL selects.

Today's traditional web provides the basic building blocks to encode text and Uniform Resource Identifiers. XML provides machine readable syntax. Namespaces facility XML-Schema definition, XSD.

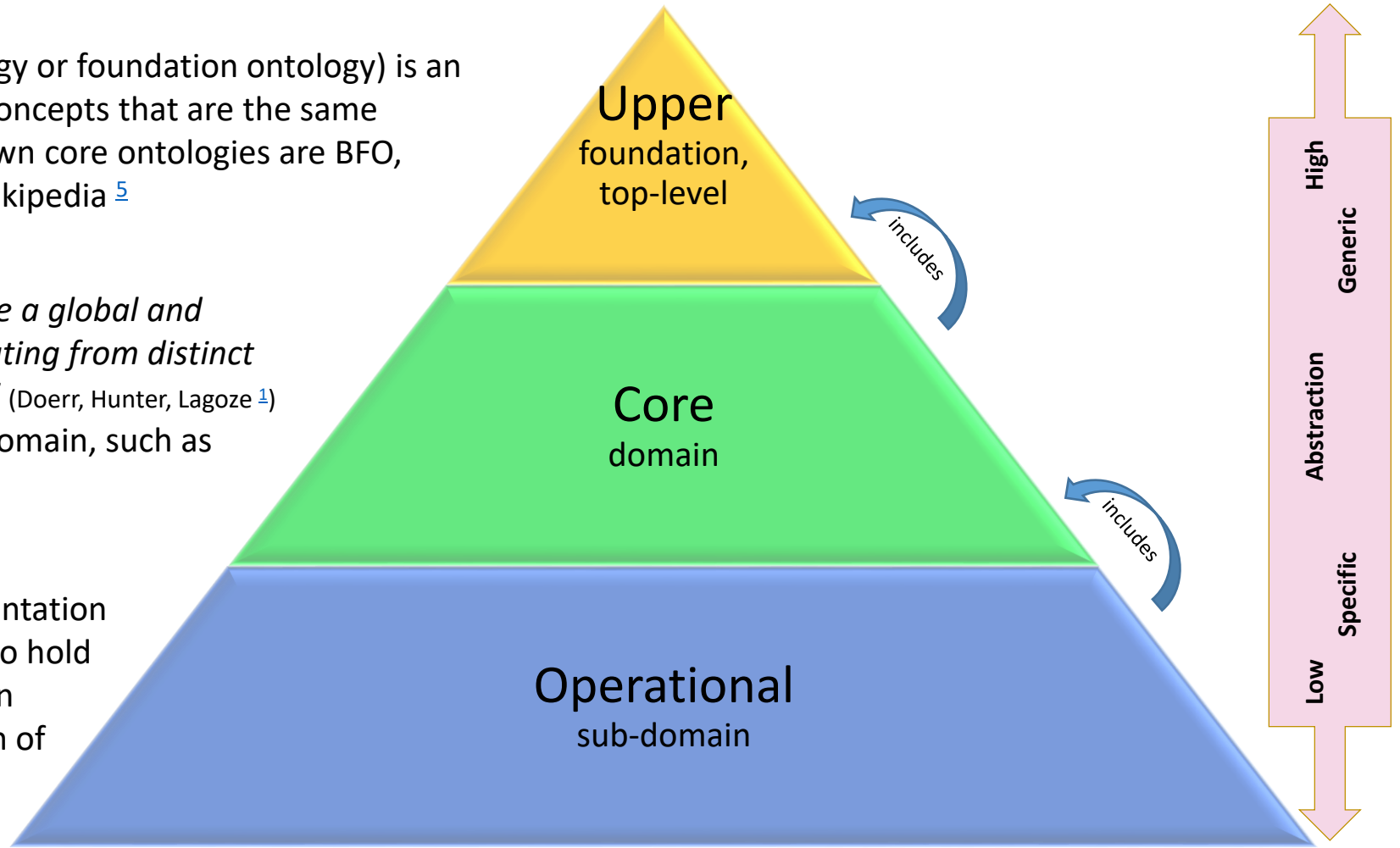


Three main levels classify ontologies

An upper ontology (aka top-level ontology or foundation ontology) is an ontology which describes very general concepts that are the same across all knowledge domains. Well known core ontologies are BFO, GFO, DOLCE, SUMO and Dublin Core. Wikipedia [5]

"The goal of a core ontology is to provide a global and extensible model into which data originating from distinct sources can be mapped and integrated." (Doerr, Hunter, Lagoze [1])
The core ontology applies to a specific domain, such as biology, medical, Legal and Finance.

The Operational Ontology is an implementation of a core ontology. It is specific enough to hold source data. Hedge Fund Regulation is an operational ontology for the sub-domain of (alternative) investment management.



FinRegOnt imports Reference Ontologies

In the Semantic Web a ontology must integrate with other ontologies in the domain.

It is a standard practice for Data and Object Architects to utilize Reference Models*.

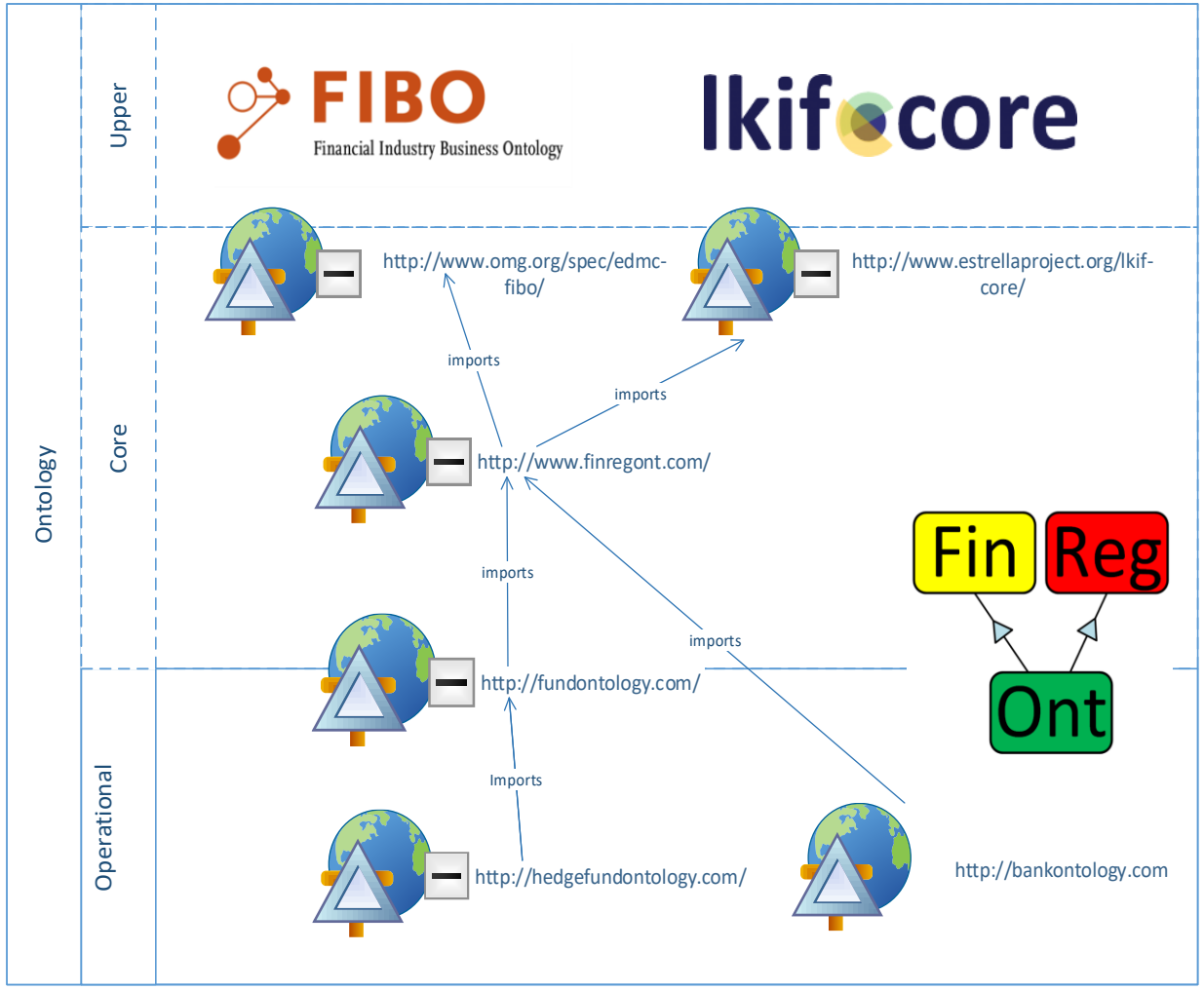
The diagram shows the base websites for reference ontologies, FIBO and LIKF and the set of FRO sites.

At the center is FinRegOnt, a core ontology integrating legal and financial information. http://finregont.com/

FinRegOnt includes:

- Financial Industry Business Ontology (FIBO). http://www.omg.org/spec/EDMC-FIBO/
- Legal Knowledge Interchange Format (LKIF). A core ontology for Legal domain. http://www.estrellaproject.org/

Unfortunately neither FIBO nor LKIF utilize an upper ontology for generic concepts. Thus the diagram places them overlapping Upper and Core.

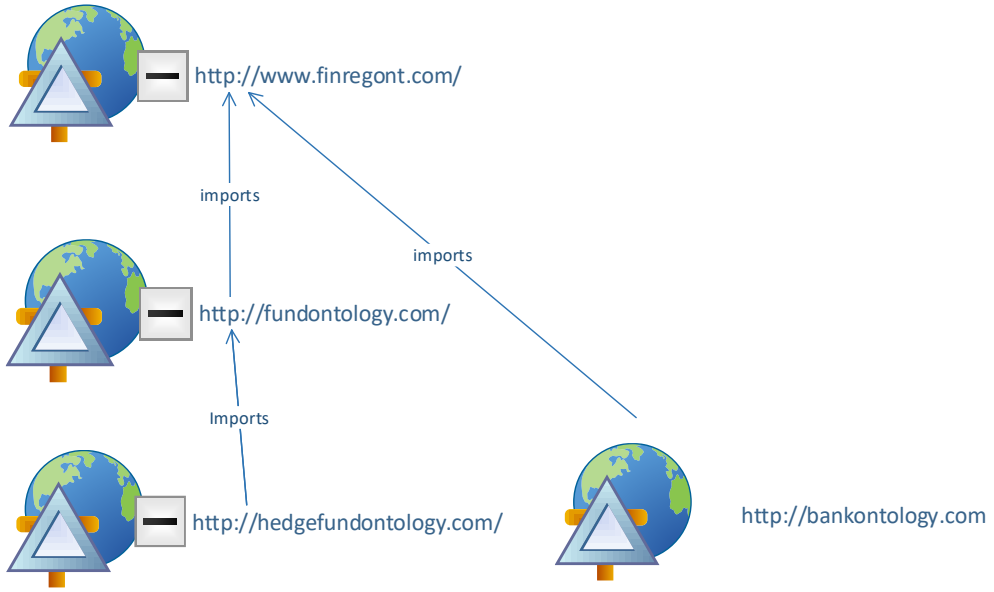


* Over 150 banks for example have licensed the IBM Banking Data Warehouse model

Operational ontologies import FinRegOnt

Financial Regulation Ontology is a set of related domain ontologies.

As a single import for Finance Subdomains FinRegOnt.com extends and aligns the two domain reference ontologies. Extensions define subclasses to specific financial and regulatory data. Ontology Alignment establishes the relationship of equivalent concepts in FIBO and LKIF. It enables queries and reasoning across the domains.
FinRegOnt is populated with the full text of relevant Code of Federal Regulations and United States Code.



Operational ontologies for regulatory compliance import FinRegOnt.

BankOntology.com covers Comprehensive Capital Assessment and Review (CCAR) and Bank Holding Companies.
FundOntotlogy.com overlaps core and operational layers. It has legal reasoning common to all investment vehicles.
HedgefundOntology.com is specific for alternative investment fund compliance.

* Over 150 banks for example have licensed the IBM Banking Data Warehouse model [6]

Financial Industry Business Ontology (FIBO)

FIBO is a collaboration between the Enterprise Data Management Council (EDMC) and the Object Management Group. The EDMC leads design in collaboration with major Financial Institutions. OMG provides governance and publishes FIBO as a formal standard.

"The EDM Council is a 501(c)(6) non-profit trade association founded by the financial industry to elevate the practice of data management as a business and operational priority. The Council is a leading advocate for the development and implementation of data content standards and the publication of data management best practices."[7]

"The Object Management Group® (OMG®) is an international, open membership, not-for-profit technology standards consortium, founded in 1989. OMG standards are driven by vendors, end-users, academic institutions and government agencies. OMG Task Forces develop enterprise integration standards for a wide range of technologies and an even wider range of industries."[8]

"FIBO™ is a business conceptual ontology standard providing a description of the structure and contractual obligations of financial instruments, legal entities, market data and financial processes. The primary application of the business conceptual ontology is for data harmonization and for the unambiguous sharing of meaning across data repositories"[3]

Legal Knowledge Interchange Format (LKIF)

The Legal Domain has several available reference ontologies. Nuria Casallas, "Legal Ontology Engineering"[1] provides a good introduction and comparison. We believe LKIF is the most advanced and the best choice for Legal Reasoning.

The **E**uropean project for **S**tandardized **T**ransparent **R**epresentations in order to **E**xtend **L**egal **A**ccessibility (Estrella, IST-2004-027655) aims to develop and validate an open, standards-based platform allowing public administrations to develop and deploy comprehensive legal knowledge management solutions, without becoming dependent on proprietary products of particular vendors.[3]

"LKIF is intended to model legal rules of the kind found in legislation and regulations."[9]
It is the main deliverable of the ESTRELLA project. LKIF is an Upper and Core ontology. The lead architect, Rinke Hoekstra made the OWL files are available on GitHub:
https://github.com/RinkeHoekstra/lkif-core

FIBO / LKIF ontology metrics

The statistics show the number of elements in the two reference ontologies.

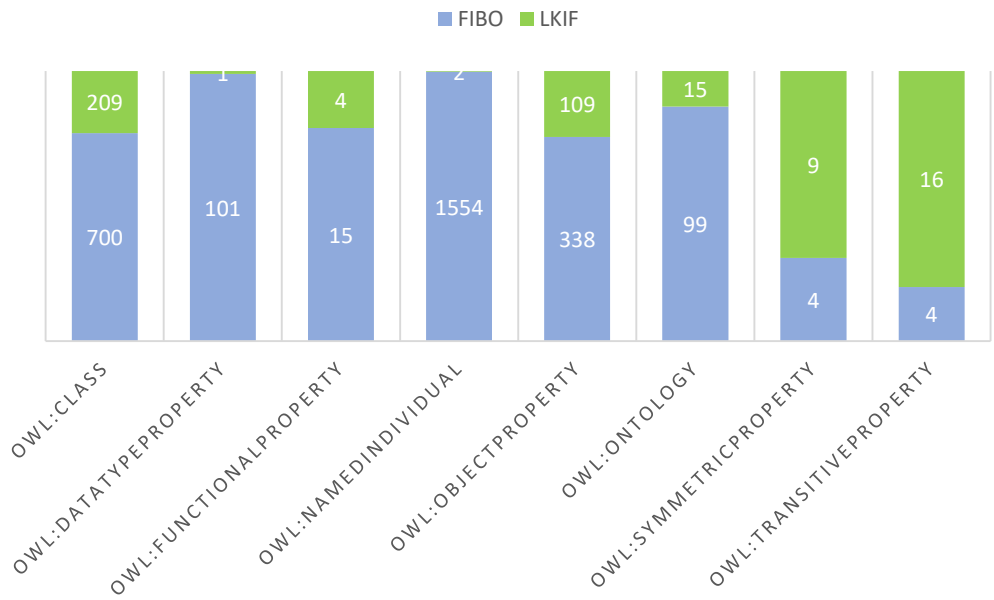
The two ontologies provide over 900 classes to utilize and leverage.

- LKIF has over 200 classes defining legal concepts.
- FIBO already has 700 classes and is still growing. (see following detail slides)

Note the absence of Data Properties and Named Individuals in LKIF. FIBO has some operational elements, data properties and individuals for Currencies, Countries and even the 12 Federal Reserve district banks.

The OWL:Ontology bar shows the actual number of OWL files.

The high number of symmetric and transitive properties show the high ontological commitment in LKIF.

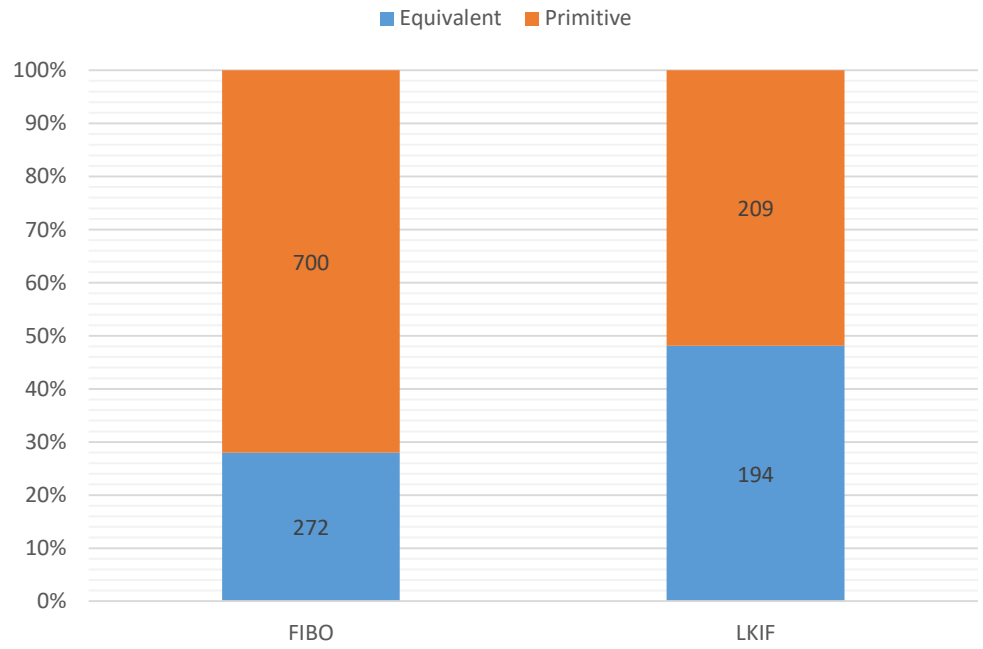


FIBO / LKIF design layer statistics

The ratio of Primitive to Equivalent classes show that LKIF has larger Complex layer than FIBO.

Three-layer ontology design pattern proposes (Dumontier, Villanueva-Rosales [10]):

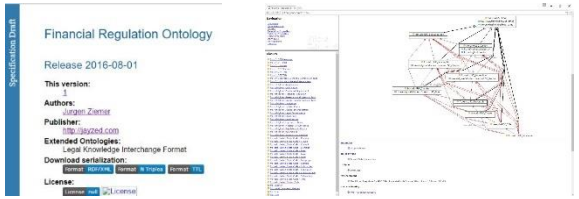
- Primitive layer consists of classes/properties forming taxonomic trees in which a single parent may be asserted.
- Complex layer refines the primitive layer by imposing restrictions such as necessary or necessary and sufficient conditions beyond the asserted subsumption.
- Application restriction layer applies highly restrictive constraints and may be used for the purposes of document validation and application interoperability.



First look at schema and data

This section is for the business users and beginning ontologist. To get an understanding of the Financial Regulation Ontology it is best to look at both schema and data.

Documentation



FinRegOnt website has the ontology documentation.

Browsable TopBraid report

http://finregont.com/fro/html/

PDF in W3C-style*
http://finregont.com/fro/html_widoco/index-en.html

Ontology Editor

protégé

Protégé is a free, open source ontology editor from Stanford University. It is the most widely used ontology design tool. Many tutorials and extensions (plugins) are available.

Semantic Endpoint

OPENLINK SOFTWARE

FinRegOnt schema modules can be queried on CKAN's Datahub.

Follow the link and instruction on the next pages.

* Thanks to Silvio Peroni for developing LODE Daniel Garijo for developing Widoco – see 6. Acknowledgements

Using an ontology editor

Ontology editors are applications that assist in the creation or manipulation of ontologies. They also provide an inference engine (Reasoner) and query interface. We use TopBraid, a commercial editor and Protégé for FRO.

TopBraid Maestro

"TopBraid Composer™ Maestro Edition (TBC-ME) combines world's leading semantic web modeling capabilities with the most comprehensive data conversion options and a powerful Integrated Development Environment (IDE) for building semantic web and Linked Data applications."[12]

This is our main tool. FRO's XML and Database imports, transformation rules and orchestration are implemented with Maestro. Most screenshots in this tutorial show TopBraid.

Protégé Desktop

We use Protégé for proofing – to make sure the ontology opens and passes Reasoner integrity checks. Protégé is free and thus the best way to get started.

Download Protégé at
http://protege.stanford.edu/products.php#desktop-protege

and follow the installation instructions
http://protegewiki.stanford.edu/wiki/Install_Protege5 .

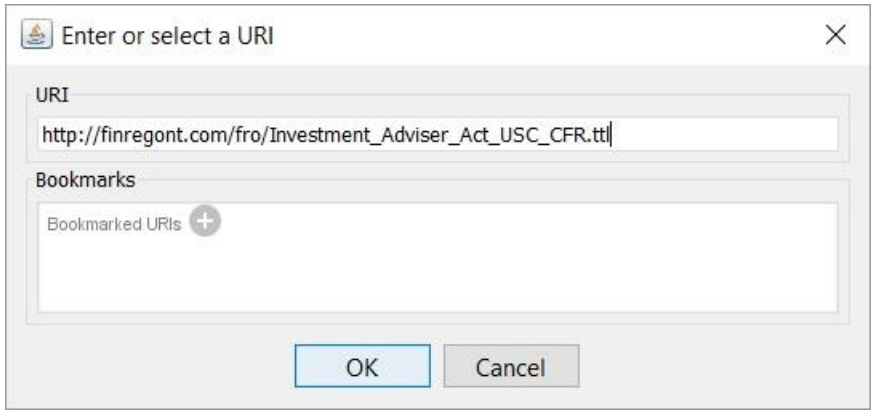
To get familiar with OWL concepts and the tool, we highly recommend the "Pizza" tutorial from Manchester university: http://owl.cs.manchester.ac.uk/publications/talks-and-tutorials/protg-owl-tutorial/

First steps with FinRegOnt in Protégé

The FinRegOnt website **fro** directory has all ontology files. There are subdirectories for Reference (fro/**ref**/), Code of Federal Regulations (fro/**cfr**/) and United States Code (fro/usc/). The Query directory (fro/**query**/) contains the SELECT statements used in this tutorial.

Launch Protégé and select File->Open from URL…

The root ontology file
http://finregont.com/fro/Investment_Adviser_Act_USC_CFR.ttl
Is a good starting point. It is populated with sections from the Code of Federal Regulations and United States Code relevant the Investment Adviser Act.



The tool loads the Investment Adviser Act ontology file and recursively all included ontologies. This may take a few minutes depending on your download speed.

First steps with FinRegOnt in Protégé

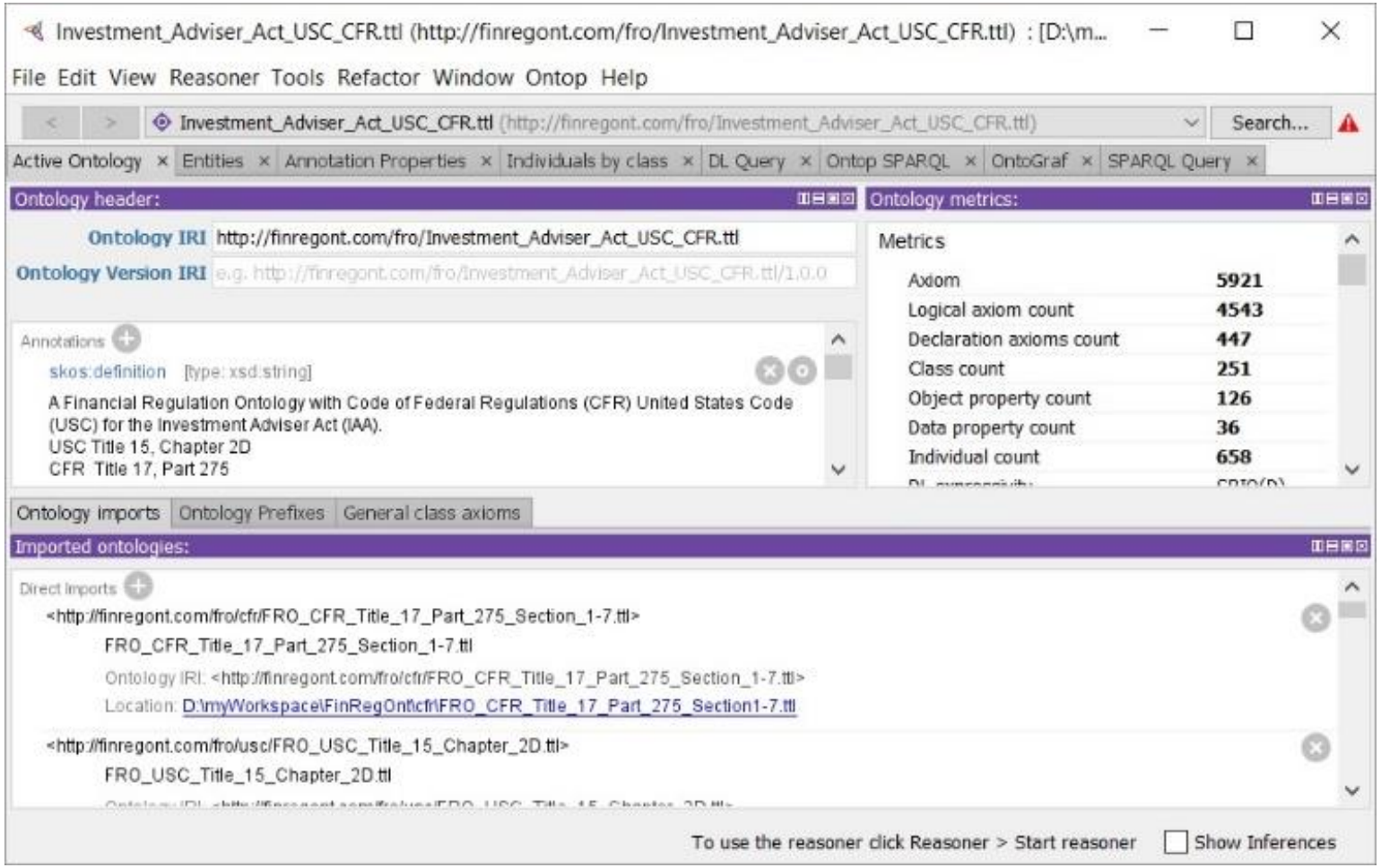
The active ontology tab displays header information, statistics and the list of includes.

The Ontology header windows shows the active ontology and it's Annotations. Here the SKOS definitions of the files. Annotations, are metadata properties attached to ontology, classes and properties. (Everything is a triple) .

The Metrics window lists statistics of the active plus included ontologies.

The bottom window, *Imported Ontologies* shows the direct and indirect imports.

Investment_Adviser_Act_USC_CFR.ttl includes FRO_CFR_Title_17_Part_275_Section_1-7, a data ontology file with the Code of Federal Regulations.



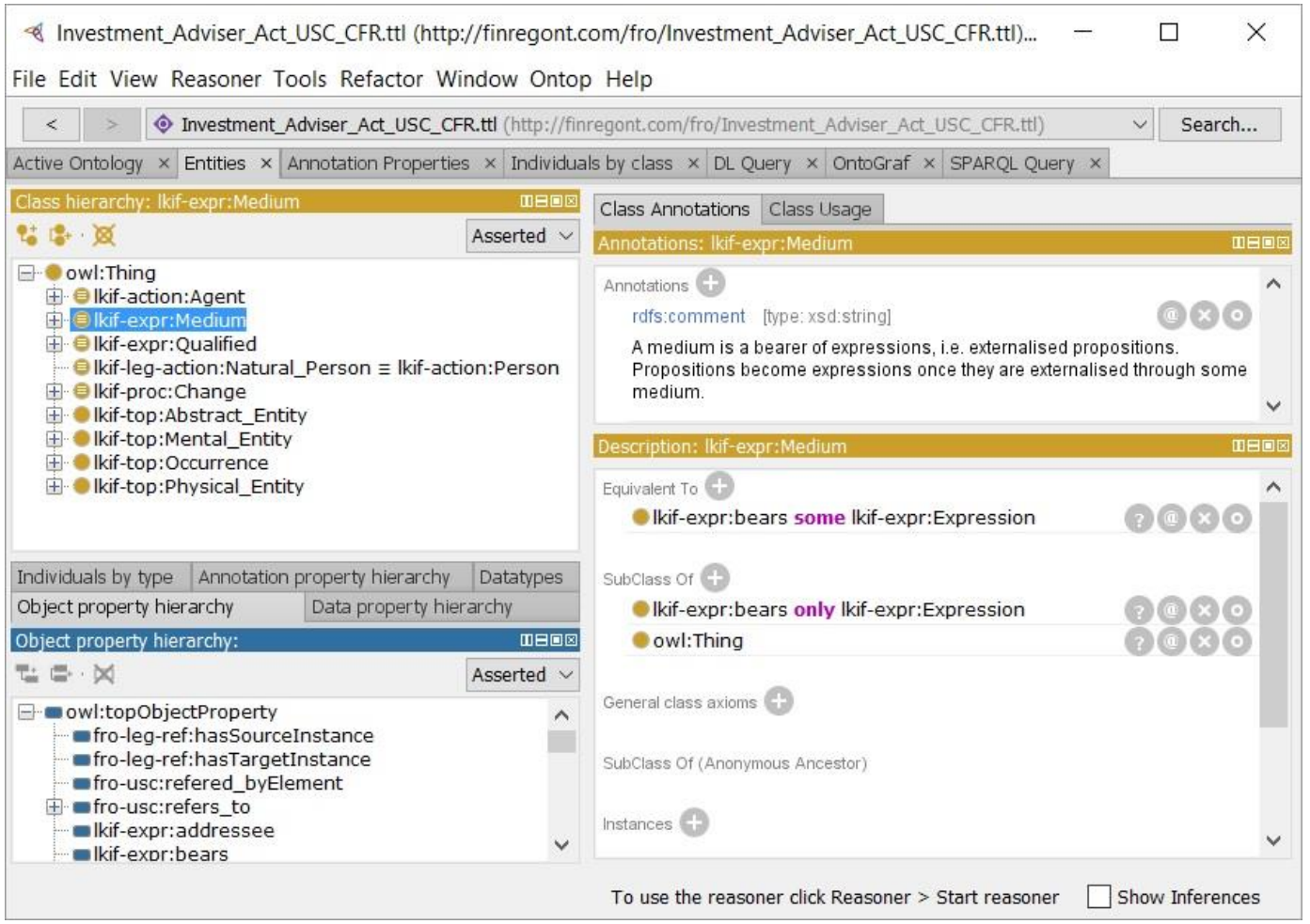
The Entities tab has windows for class and properties

The entities tab has a navigator window for the class and object hierarchy.

Note that all top-level classes, directly under owl:Thing are LKIF. Financial Regulation Ontology extends LKIF with United States Code (USC) and Code of Federal Regulation classes. We will see FRO subclasses of LKIF in the following pages.

Tip: Regulatory Document classes are lkif-expr:Medium. Lawmakers, supervisors are lkif-action:Agent.

Expand the “+” sign in the Navigator to drill down.



Ontology modules and prefixes

Ontologies LKIF, FIBO and FRO are broken down into modules. The modules contain OWL files.

A file will define a namespace for triples defined in the ontology. The prefix abbreviates the namespace.

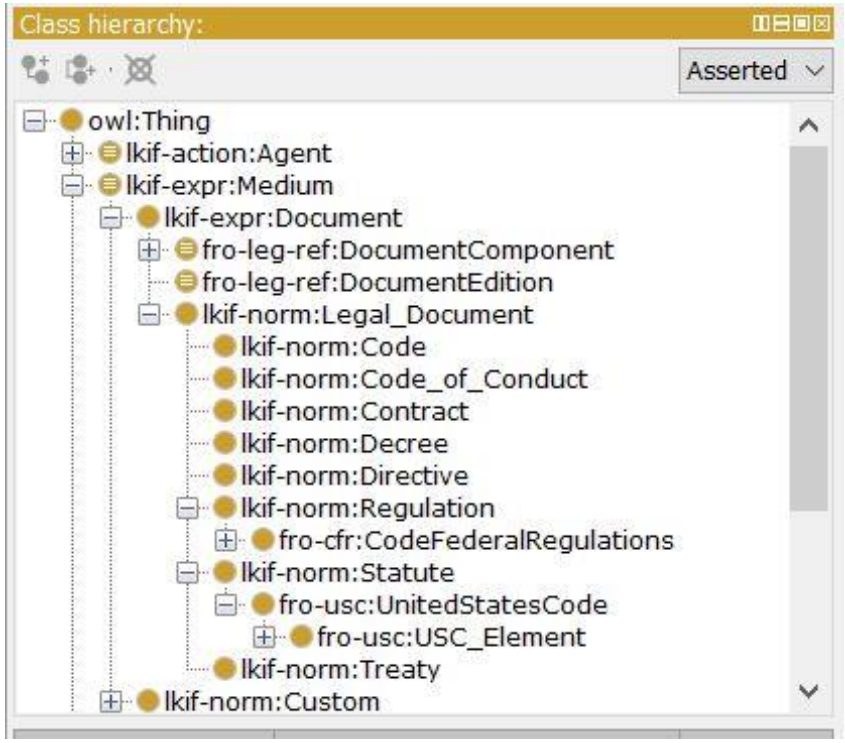
In Protégé select File->Preferences, Render tab and “Render by prefixed name to display the prefix for the Class names. The naming convention is: ontology-module: Name

The ontology is “lkif-”, “fibo-” or “fro-”.

Navigate down lkif-expr:Medium to expand fro-cfr:CodeFederalRegulations and fro-usc:UnitedStatesCode.

The Code of Federal Regulations is a LKIF Regulation; the United States Code is a LKIF Statute.

The defined classes fro-leg-ref:DocumentEdition and DocumentComponent enable generic queries on compliance documents.



Class details for CFR Section

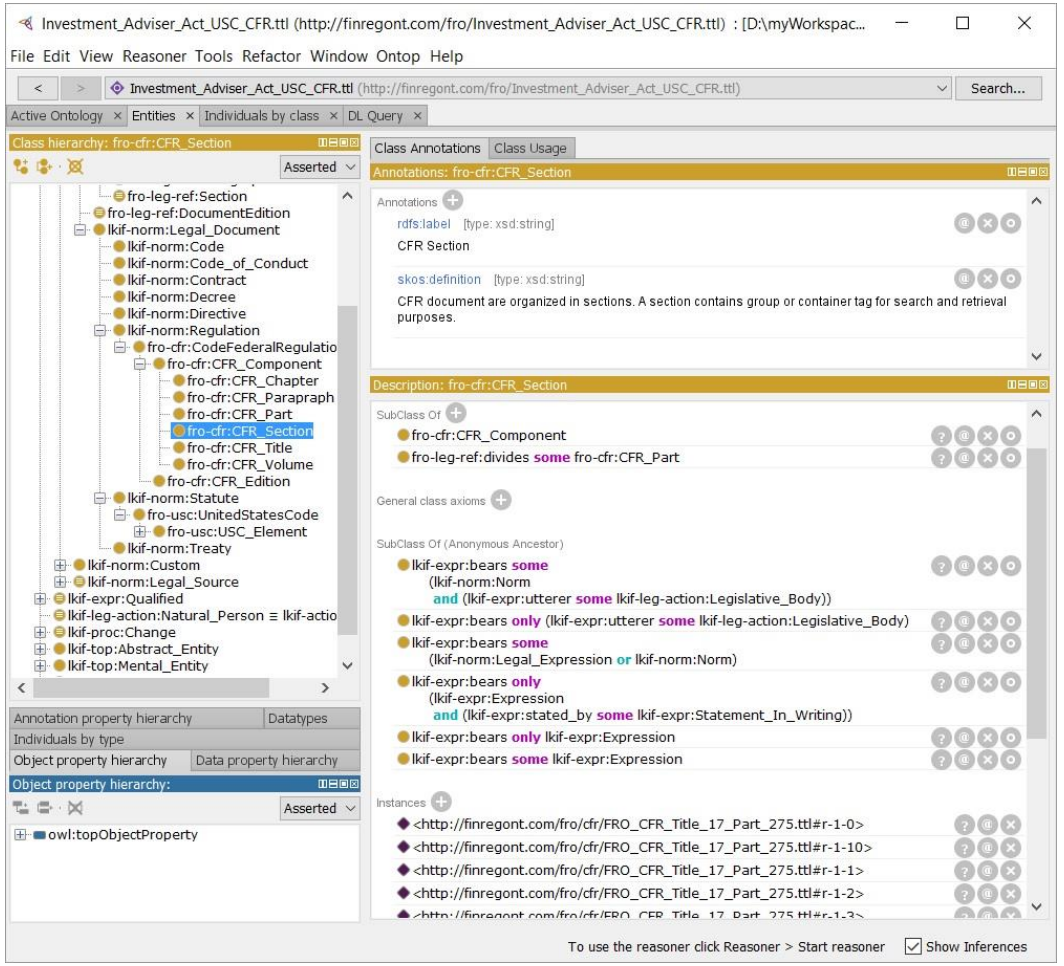
Drill down and select CFR_Section.

The Annotations window shows the class label and definition.

The Description window defines CFR_Section as a subclass of CFR_Component and the anonymous class “fro-leg-ref:divides some fro-cfr:CFR_Part.” The Code of Federal Regulations is a hierarchy of Title, Chapter, Section and Paragraph. The divides object property positions Individuals in the hierarchy.

The SubClass of (Anonymous Ancestor) axioms inherited from lkif-norm:LegalDocument place the regulation in the wider legal ontology context. It bears a Legal Expression that is a Statement in Writing made by a Legislative Body (the US Congress).

The Instances part lists 10 data records imported from the CFR XML file. The namespace has the FinRegOnt.com directory and filename. Click on http://finregont.com/fro/cfr/FRO_CFR_Title_17_Part275.ttl#r-1-0



Individual, instance details for CFR §275.0-3

The Property assertions tab shows Object and Data properties for the section individual.

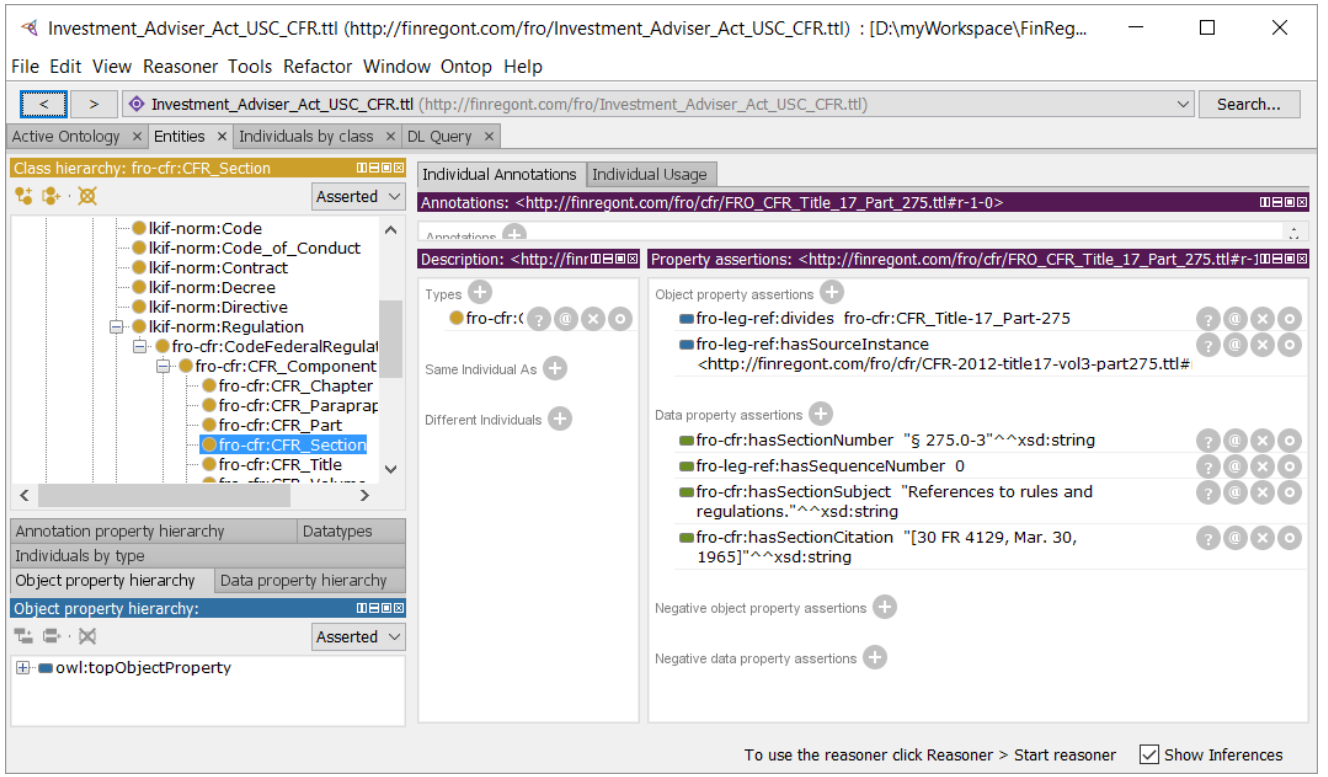
The Section divides Part 275 of the regulation. The object property *fro-leg-ref:hasSourceInstance* links to the original record in the XML import for this section.

The data properties lists *hasSectionNumber* "§ 275.0-3",

hasSubject "References to rules and regulations"

and *hasSectionCitation* "[30 FR 4129, Mar. 30, 1965]" as defined in CFR.

Sequence Number is system generated by the XML import. It is simply the order of elements in the CFR XML file. We use the property to ORDER BY in queries.



http://finregont.com © Jayzed Data Models Inc. 2016

Querying FRO instance data

We can query the ontology data in TopBraid Composer, Protégé or on a Semantic endpoint.

More sample SPARQL queries are explained in Chapter 2 of the tutorial and available on the website: http://finregont.com/fro/query/

SPARQL is a the query language for the Semantic Web, able to select and manipulate data stored in Resource Description Framework (RDF) format. Just like SQL is the query language for relational databases.

The Ontology editors have an interface to run SPARQL queries on ontology files.

The best book on SPARQL is Bob DuCharme's "Learning SPARQL", O'Reily, 2013.

RDF Database

Large volumes as data require dedicated database systems. All major database vendors provide add-ons to store RDF.
If you already have a license "Oracle Spatial and Graph" can be installed for free.

There are also native RDF systems like Open Source Virtuoso and Jena.

See Oliver Cure and Guillaume Blin's "RDF Database Systems", Morgan Kaufman, 2015 for comparison and excellent treatment of the internals.

Semantic Endpoint

A Semantic or SPARQL endpoint is a Web protocol service that enables users and applications to query a knowledge base. All RDF Databases provide endpoints.

Many well-known information providers have public endpoints. For example, http://wiki.dbpedia.org/ lets you query Wikipedia information. Governments publish data on Semantic Endpoint. For confidential information the Endpoint is only on the Financial Institution's intranet. The next pages show FinRegOnt on a CKANS public Endpoint.

FinRegOnt demo Semantic Endpoint

FinRegOnt is available for query on CKAN's Data Store Semantic Endpoint.

Open Knowledge

"Open Knowledge International is a worldwide non-profit network of people passionate about openness, using advocacy, technology and training to unlock information and enable people to work with it to create and share knowledge." https://okfn.org/about/

CKAN

CKAN is a an open source data portal widely used by governments and public services.

FinRegOnt is open source and published on the CKAN Datahub. https://datahub.io/dataset/financial-regulation-ontology

Although not supported CKAN has a Data Store, where samples of the Financial Regulation Ontology can be queried. http://semantic.ckan.net/isparql/

Query FinRegOnt on a SPARQL endpoint

To query FinRegOnt on a Semantic Endpoint, we need to provide FROM information of the graphs

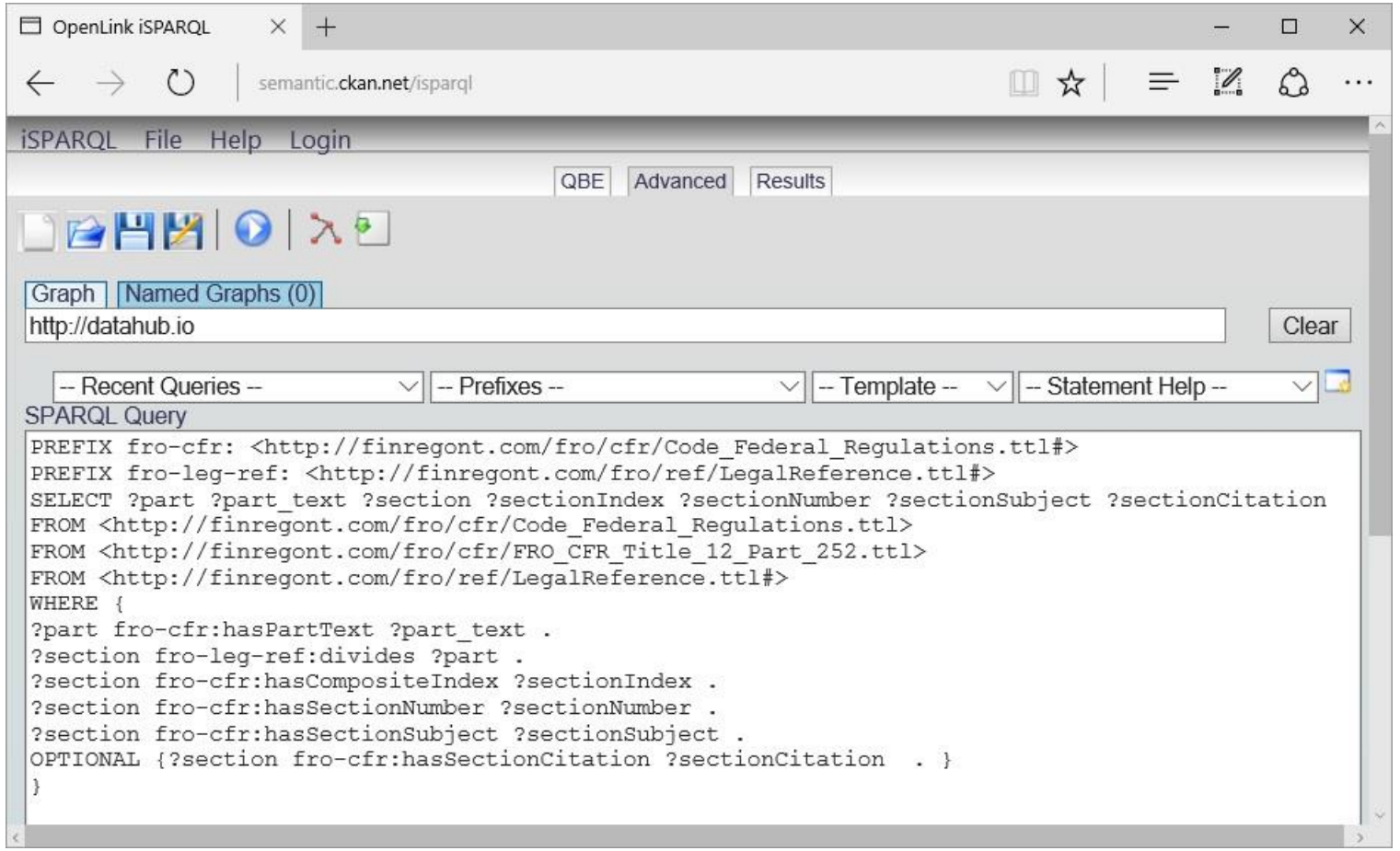
The CKAN endpoint is powered by Virtuoso's iSPARQL.

Launch the CKAN endpoint http://semantic.ckan.net/isparql/ on a web browser and click on the "Advanced" tab.

To get started just copy and paste the query "CFR 252 validation complete text (PREFIX FROM).rq" from the FinRegOnt query directory.

http://finregont.com/fro/query/CFR%20252%20validation%20complete%20text%20(PREFIX%20FROM).rq

Click on the execute icon.



SPARQL query results

The result set shows the record for Code of Federal Regulations Title 12 Part 252 – enhanced prudential standards. You can look up the ontology documentation and explore variations of the query.

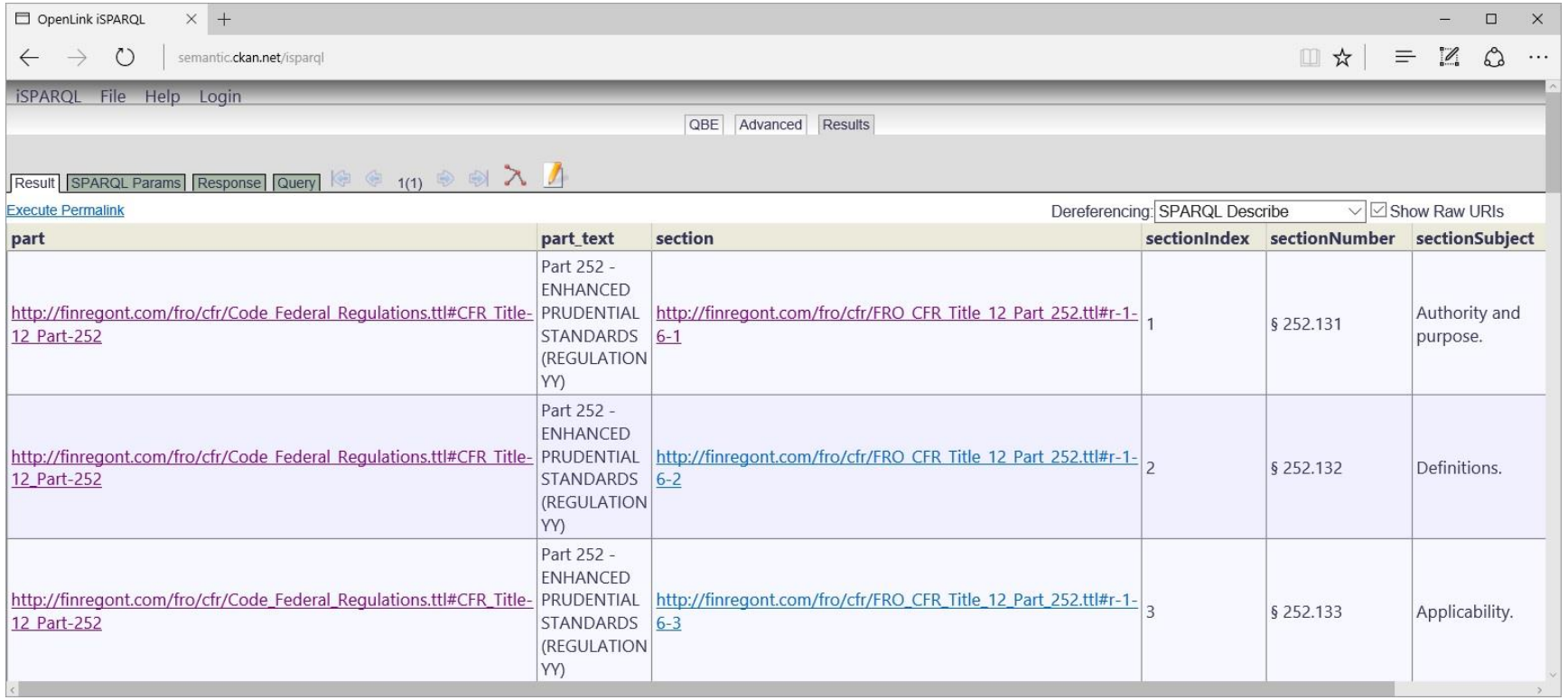
Columns in the Result tab are bound variables in the query:

"?part fro-cfr:hasPartText ?part_text" asks the query engine for all resources that are domain and range of the hasPartText data property.

"?section fro-leg-ref:divides ?part" is the object property join from CFR Part to CFR Section.

The query asks for hasSectionNumber and hasSectionSubject of the CFR Section.

Everything is a Triple!



part	part_text	section	sectionIndex	sectionNumber	sectionSubject
http://finregont.com/fro/cfr/Code_Federal_Regulations.ttl#CFR_Title-12_Part-252	Part 252 - ENHANCED PRUDENTIAL STANDARDS (REGULATION YY)	http://finregont.com/fro/cfr/FRO_CFR_Title_12_Part_252.ttl#r-1-6-1	1	§ 252.131	Authority and purpose.
http://finregont.com/fro/cfr/Code_Federal_Regulations.ttl#CFR_Title-12_Part-252	Part 252 - ENHANCED PRUDENTIAL STANDARDS (REGULATION YY)	http://finregont.com/fro/cfr/FRO_CFR_Title_12_Part_252.ttl#r-1-6-2	2	§ 252.132	Definitions.
http://finregont.com/fro/cfr/Code_Federal_Regulations.ttl#CFR_Title-12_Part-252	Part 252 - ENHANCED PRUDENTIAL STANDARDS (REGULATION YY)	http://finregont.com/fro/cfr/FRO_CFR_Title_12_Part_252.ttl#r-1-6-3	3	§ 252.133	Applicability.

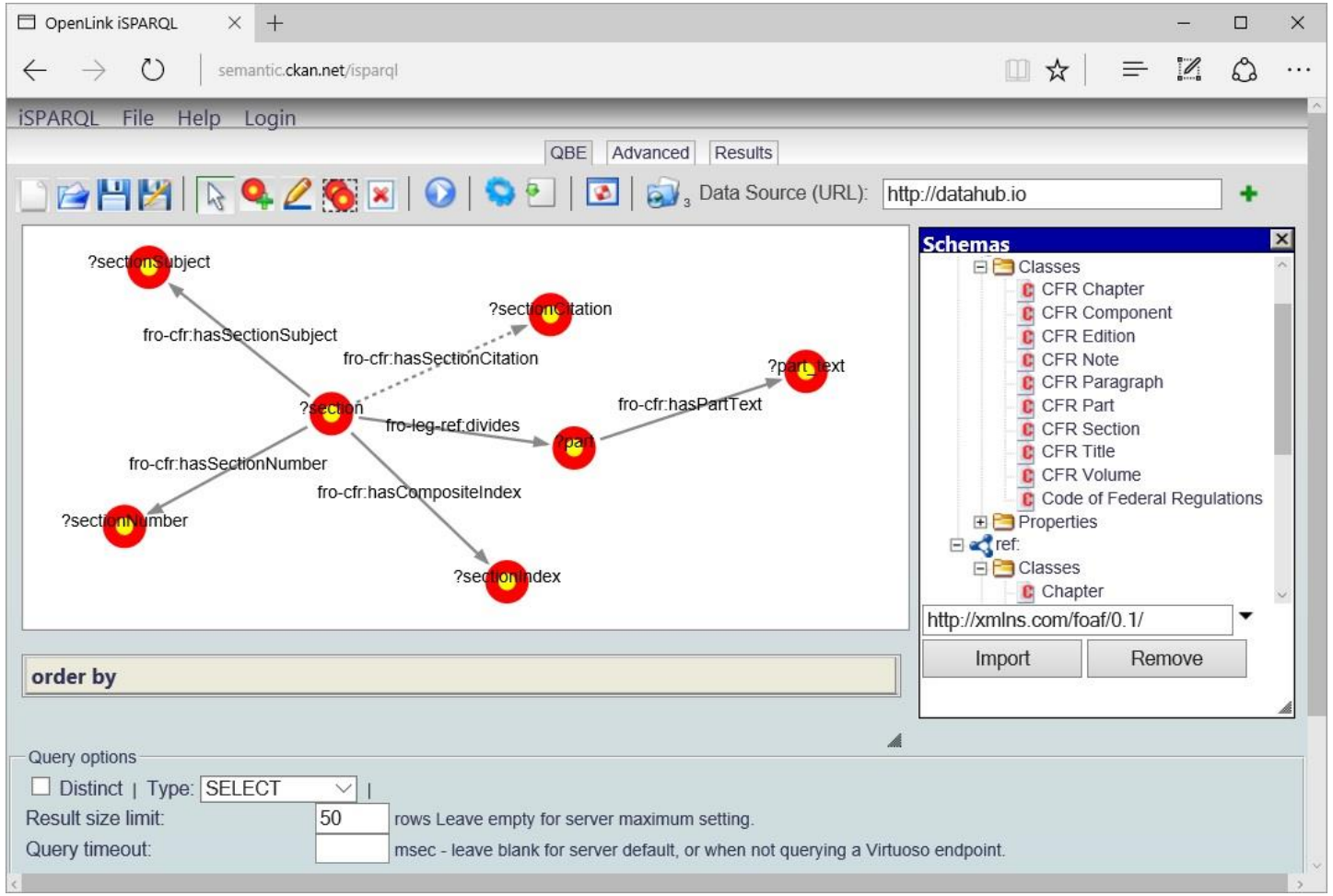
SPARQL query by example

The QBE tab shows the query's graph. Arrows depicting the properties connect circles for the variables.

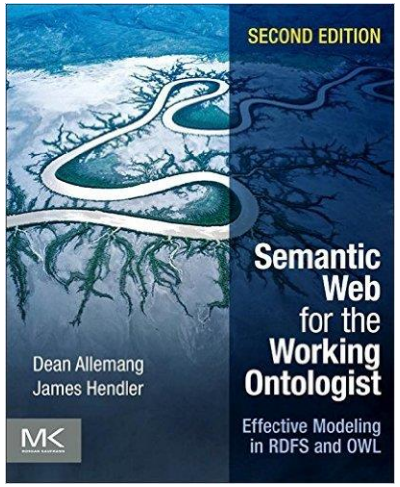
The Schema explorer box on the right hand side shows classes and properties defined in the ontologies.

In our example Code_Federal_Regulations.ttl graph defines the CFR classes for Title, Volume, Part, Chapter, Sections etc.

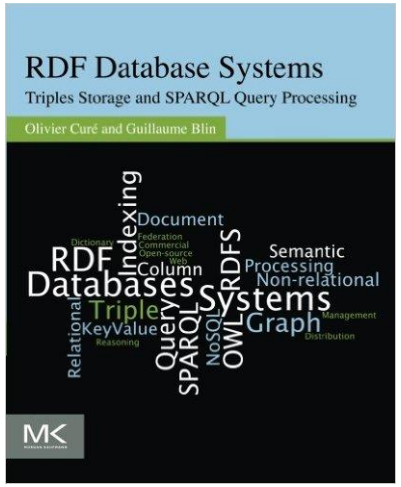
To modify the query, we drag additional entities into the diagram.



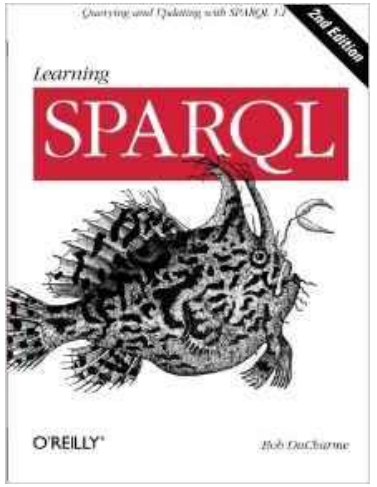
Chapter I – books, recommended companion reading



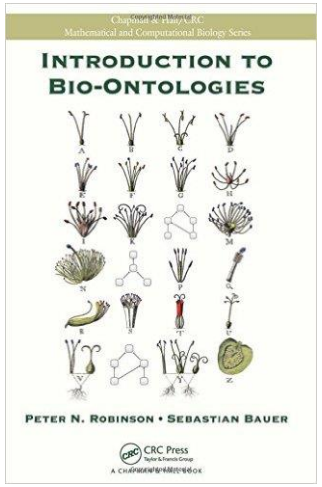
Semantic Web for the Working Ontologist, Second Edition: Effective Modeling in RDFS and OWL
Dean Allemang, James Hendler
Morgan Kaufman, 2011



RDF Database Systems: Triples Storage and SPARQL Query Processing
Peter N. Robinson
Chapman & Hall, 2014



Learning SPARQL
Bob DuCharme
O'Reilly, 2013



Introduction to Bio-Ontologies
Peter N. Robinson
Chapman & Hall, 2011

Chapter I - references

1. Towards a core ontology for information integration
   M Doerr, J Hunter, C Lagoze - Journal of Digital information, 2006 - journals.tdl.org
2. Financial Industry Business Ontology (FIBO) project website: http://www.edmcouncil.org/financialbusiness
3. ESTRELLA project website: http://www.estrellaproject.org/ GitHub: https://github.com/RinkeHoekstra/lkif-core
4. Sir Tim Berners-Lee, director or the World Wide Web Consortium (W3C), Scientific American 2001 http://www.scientificamerican.com/article/the-semantic-web/
5. Upper ontology Wikipedia - https://en.wikipedia.org/wiki/Upper_ontology
6. IBM Banking Data Warehouse – product page: http://www-03.ibm.com/software/products/en/banking/
7. Enterprise Data Management Council website: http://www.edmcouncil.org/
8. Object Management Group website: http://www.omg.org/
9. ESTRELLA Deliverable N°: 4.1 -The Legal Knowledge Interchange Format (LKIF): http://www.estrellaproject.org/doc/Estrella-D4.1.pdf
10. Three-Layer OWL Ontology Design
    Michel Dumontier, Natalia Villanueva-Rosales1 - School of Computer Science, Department of Biology, Carleton University, 2007
11. Cornell University Law School, Legal Information Institute website: https://www.law.cornell.edu/wex
12. Topbraid Composer website: http://www.topquadrant.com/tools/IDE-topbraid-composer-maestro-edition/
13. Protégé Stanford University website: http://protege.stanford.edu/
14. “Pizza” tutorial from Manchester university: http://owl.cs.manchester.ac.uk/publications/talks-and-tutorials/protg-owl-tutorial/

Intentionally left blank