Date: February 2016

Financial Industry Business Ontology Foundations

RTF – Final

OMG Document Number: dtc/2016-03-03

Standard document URL: http://www.omg.org/spec/FIBO/1.1

Normative Machine Consumable File(s):

http://www.omg.org/spec/EDMC-FIBO/FND/20160201/Accounting/AccountingEquity.rdf

http://www.omg.org/spec/EDMC-FIBO/FND/20160201/Accounting/CurrencyAmount.rdf

http://www.omg.org/spec/EDMC-FIBO/FND/20150801/Accounting/ISO4217-CurrencyCodes.rdf

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/AgentsAndPeople/Agents.rdf

http://www.omg.org/spec/EDMC-FIBO/FND/20160201/AgentsAndPeople/People.rdf

http://www.omg.org/spec/EDMC-FIBO/FND/20160201/Agreements/Agreements.rdf

http://www.omg.org/spec/EDMC-FIBO/FND/20160201/Agreements/Contracts.rdf

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Arrangements/Arrangements.rdf

http://www.omg.org/spec/EDMC-FIBO/FND/20150801/Arrangements/ClassificationSchemes.rdf

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Arrangements/Codes.rdf

http://www.omg.org/spec/EDMC-FIBO/FND/20160201/Arrangements/Documents.rdf

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Arrangements/IdentifiersAndIndices.rdf

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/DatesAndTimes/BusinessDates.rdf

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/DatesAndTimes/FinancialDates.rdf

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/DatesAndTimes/Occurrences.rdf

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/GoalsAndObjectives/Goals.rdf

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/GoalsAndObjectives/Objectives.rdf

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Law/Jurisdiction.rdf

http://www.omg.org/spec/EDMC-FIBO/FND/20150801/Law/LegalCapacity.rdf

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Law/LegalCore.rdf

http://www.omg.org/spec/EDMC-FIBO/FND/20160201/Organizations/FormalOrganizations.rdf

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Organizations/LegitimateOrganizations.rdf

http://www.omg.org/spec/EDMC-FIBO/FND/20160201/Organizations/Organizations.rdf

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/OwnershipAndControl/Control.rdf

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/OwnershipAndControl/Ownership.rdf

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/OwnershipAndControl/OwnershipAndControl.rdf

http://www.omg.org/spec/EDMC-FIBO/FND/20160201/Parties/Parties.rdf

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Parties/Roles.rdf

http://www.omg.org/spec/EDMC-FIBO/FND/20160201/Places/Addresses.rdf

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Places/Countries.rdf

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Places/Facilities.rdf

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Places/Locations.rdf

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Places/VirtualPlaces.rdf

http://www.omg.org/spec/EDMC-FIBO/FND/20160201/ProductsAndServices/PaymentsAndSchedules.rdf

http://www.omg.org/spec/EDMC-FIBO/FND/20160201/ProductsAndServices/ProductsAndServices.rdf

http://www.omg.org/spec/EDMC-FIBO/FND/20160201/Quantities/QuantitiesAndUnits.rdf

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Relations/Relations.rdf

http://www.omg.org/spec/EDMC-FIBO/FND/20160201/Utilities/Analytics.rdf

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Utilities/AnnotationVocabulary.rdf

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Utilities/BusinessFacingTypes.rdf

http://www.omg.org/spec/EDMC-FIBO/FND/20160201/Accounting/AccountingEquity.xml

http://www.omg.org/spec/EDMC-FIBO/FND/20160201/Accounting/CurrencyAmount.xml

http://www.omg.org/spec/EDMC-FIBO/FND/20150801/Accounting/ISO4217-CurrencyCodes.xml

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/AgentsAndPeople/Agents.xml

http://www.omg.org/spec/EDMC-FIBO/FND/20160201/AgentsAndPeople/People.xml

http://www.omg.org/spec/EDMC-FIBO/FND/20160201/Agreements/Agreements.xml

http://www.omg.org/spec/EDMC-FIBO/FND/20160201/Agreemeents/Contracts.xml

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Arrangements/Arrangements.xml

http://www.omg.org/spec/EDMC-FIBO/FND/20150801/Arrangements/ClassificationSchemes.xml

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Arrangements/Arrangements/Codes.xml

http://www.omg.org/spec/EDMC-FIBO/FND/20160201/Arrangements/Arrangements/Documents.xml

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Arrangements/IdentifiersAndIndices.xml

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/DatesAndTimes/BusinessDates.xml

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/DatesAndTimes/FinancialDates.xml

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/DatesAndTimes/Occurrences.xml

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/GoalsAndObjectives/Goals.xml

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/GoalsAndObjectives/Objectives.xml

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Law/Jurisdiction.xml

http://www.omg.org/spec/EDMC-FIBO/FND/20150801/Law/LegalCapacity.xml

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Law/LegalCore.xml

http://www.omg.org/spec/EDMC-FIBO/FND/20160201/Organizations/FormalOrganizations.xml

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Organizations/LegitimateOrganizations.xml

http://www.omg.org/spec/EDMC-FIBO/FND/20160201/Organizations/Organizations.xml

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/OwnershipAndControl/Control.xml

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/OwnershipAndControl/Ownership.xml

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/OwnershipAndControl/OwnershipAndControl.xml

http://www.omg.org/spec/EDMC-FIBO/FND/20160201/Parties/Parties.xml

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Parties/Roles.xml

http://www.omg.org/spec/EDMC-FIBO/FND/20160201/Places/Addresses.xml

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Places/Countries.xml

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Places/Facilities.xml

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Places/Locations.xml

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Places/VirtualPlaces.xml

http://www.omg.org/spec/EDMC-FIBO/FND/20160201/ProductsAndServices/PaymentsAndSchedules.xml

http://www.omg.org/spec/EDMC-FIBO/FND/20160201/ProductsAndServices/ProductsAndServices.xml

http://www.omg.org/spec/EDMC-FIBO/FND/20160201/Quantities/QuantitiesAndUnits.xml

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Relations/Relations.xml

http://www.omg.org/spec/EDMC-FIBO/FND/20160201/Utilities/Analytics.xml

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Utilities/AnnotationVocabulary.xml

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Utilities/BusinessFacingTypes.xml

http://www.omg.org/spec/EDMC-FIBO/FND/20160201/Accounting/AccountingEquity.xmi

http://www.omg.org/spec/EDMC-FIBO/FND/20160201/Accounting/CurrencyAmount.xmi

http://www.omg.org/spec/EDMC-FIBO/FND/20150801/Accounting/ISO4217-CurrencyCodes.xmi

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/AgentsAndPeople/Agents.xmi

http://www.omg.org/spec/EDMC-FIBO/FND/20160201/AgentsAndPeople/People.xmi

http://www.omg.org/spec/EDMC-FIBO/FND/20160201/Agreements/Agreements.xmi

http://www.omg.org/spec/EDMC-FIBO/FND/20160201/Agreements/Contracts.xmi

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Arrangements/Arrangements.xmi

http://www.omg.org/spec/EDMC-FIBO/FND/20150801/Arrangements/ClassificationSchemes.xmi

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Arrangements/Codes.xmi

http://www.omg.org/spec/EDMC-FIBO/FND/20160201/Arrangements/Documents.xmi

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Arrangements/IdentifiersAndIndices.xmi

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/DatesAndTimes/BusinessDates.xmi

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/DatesAndTimes/FinancialDates.xmi

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/DatesAndTimes/Occurrences.xmi

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/GoalsAndObjectives/Goals.xmi

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/GoalsAndObjectives/Objectives.xmi

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Law/Jurisdiction.xmi

http://www.omg.org/spec/EDMC-FIBO/FND/20150801/Law/LegalCapacity.xmi

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Law/LegalCore.xmi

http://www.omg.org/spec/EDMC-FIBO/FND/20160201/Organizations/FormalOrganizations.xmi

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Organizations/LegitimateOrganizations.xmi

http://www.omg.org/spec/EDMC-FIBO/FND/20160201/Organizations/Organizations.xmi

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/OwnershipAndControl/Control.xmi

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/OwnershipAndControl/Ownership.xmi

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/OwnershipAndControl/OwnershipAndControl.xmi

http://www.omg.org/spec/EDMC-FIBO/FND/20160201/Parties/Parties.xmi

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Parties/Roles.xmi

http://www.omg.org/spec/EDMC-FIBO/FND/20160201/Places/Addresses.xmi

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Places/Countries.xmi

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Places/Facilities.xmi

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Places/Locations.xmi

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Places/VirtualPlaces.xmi

http://www.omg.org/spec/EDMC-FIBO/FND/20160201/ProductsAndServices/PaymentsAndSchedules.xmi

http://www.omg.org/spec/EDMC-FIBO/FND/20160201/ProductsAndServices/ProductsAndServices.xmi

http://www.omg.org/spec/EDMC-FIBO/FND/20160201/Quantities/QuantitiesAndUnits.xmi

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Relations/Relations.xmi

http://www.omg.org/spec/EDMC-FIBO/FND/20160201/Utilities/Analytics.xmi

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Utilities/AnnotationVocabulary.xmi

http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Utilities/BusinessFacingTypes.xmi

---

This OMG document replaces the submission document (finance/2013-09-02, Alpha). It is an OMG Adopted Beta specification and is currently in the finalization phase. Comments on the content of this document are welcome, and should be directed to issues@omg.org by February 28, 2014.

You may view the pending issues for this specification from the OMG revision issues web page http://www.omg.org/issues/.

The FTF Recommendation and Report for this specification will be published on June 27, 2014. If you are reading this after that date, please download the available specification from the OMG Specifications Catalog.

Subject to all of the terms and conditions below, the owners of the copyright in this specification hereby grant you a fully-paid up, non-exclusive, nontransferable, perpetual, worldwide license (without the right to sublicense), to use this specification to create and distribute software and special purpose specifications that are based upon this specification, and to use, copy, and distribute this specification as provided under the Copyright Act; provided that: (1) both the copyright notice identified above and this permission notice appear on any copies of this specification; (2) the use of the specifications is for informational purposes and will not be copied or posted on any network computer or broadcast in any media and will not be otherwise resold or transferred for commercial purposes; and (3) no modifications are made to this specification. This limited permission automatically terminates without notice if you breach any of these terms or conditions. Upon termination, you will destroy immediately any copies of the specifications in your possession or control.

PATENTS

The attention of adopters is directed to the possibility that compliance with or adoption of OMG specifications may require use of an invention covered by patent rights. OMG shall not be responsible for identifying patents for which a license may be required by any OMG specification, or for conducting legal inquiries into the legal validity or scope of those patents that are brought to its attention. OMG specifications are prospective and advisory only. Prospective users are responsible for protecting themselves against liability for infringement of patents.

GENERAL USE RESTRICTIONS

Any unauthorized use of this specification may violate copyright laws, trademark laws, and communications regulations and statutes. This document contains information which is protected by copyright. No part of this work covered by copyright herein may be reproduced or used in any form or by any means--graphic, electronic, or mechanical, including photocopying, recording, taping, or information storage and retrieval systems--without permission of the copyright owner.

DISCLAIMER OF WARRANTY

WHILE THIS PUBLICATION IS BELIEVED TO BE ACCURATE, IT IS PROVIDED "AS IS" AND MAY CONTAIN ERRORS OR MISPRINTS. THE OBJECT MANAGEMENT GROUP AND THE COMPANIES LISTED ABOVE MAKE NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS PUBLICATION, INCLUDING BUT NOT LIMITED TO ANY WARRANTY OF TITLE OR OWNERSHIP, IMPLIED WARRANTY OF MERCHANTABILITY OR WARRANTY OF FITNESS FOR A PARTICULAR PURPOSE OR USE. IN NO EVENT SHALL THE OBJECT MANAGEMENT GROUP OR ANY OF THE COMPANIES LISTED ABOVE BE LIABLE FOR ERRORS CONTAINED HEREIN OR FOR DIRECT, INDIRECT, INCIDENTAL, SPECIAL, CONSEQUENTIAL, RELIANCE OR COVER DAMAGES, INCLUDING LOSS OF PROFITS, REVENUE, DATA OR USE, INCURRED BY ANY USER OR ANY THIRD PARTY IN CONNECTION WITH THE FURNISHING, PERFORMANCE, OR USE OF THIS MATERIAL, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

The entire risk as to the quality and performance of software developed using this specification is borne by you. This disclaimer of warranty constitutes an essential part of the license granted to you to use this specification.

RESTRICTED RIGHTS LEGEND

TRADEMARKS

COMPLIANCE

OMG's Issue Reporting Procedure

All OMG specifications are subject to continuous review and improvement. As part of this process we encourage readers to report any ambiguities, inconsistencies, or inaccuracies they may find by completing the Issue Reporting Form listed on the main web page http://www.omg.org, under Documents, Report a Bug/Issue (http://www.omg.org/report_issue.htm.)

Table of Contents

Preface

OMG

Founded in 1989, the Object Management Group, Inc. (OMG) is an open membership, not-for-profit computer industry standards consortium that produces and maintains computer industry specifications for interoperable, portable, and reusable enterprise applications in distributed, heterogeneous environments. Membership includes Information Technology vendors, end users, government agencies, and academia.

OMG member companies write, adopt, and maintain its specifications following a mature, open process. OMG's specifications implement the Model Driven Architecture® (MDA®), maximizing ROI through a full-lifecycle approach to enterprise integration that covers multiple operating systems, programming languages, middleware and networking infrastructures, and software development environments. OMG's specifications include: UML® (Unified Modeling Language™); CORBA® (Common Object Request Broker Architecture); CWM™ (Common Warehouse Metamodel); and industry-specific standards for dozens of vertical markets.

More information on the OMG is available at http://www.omg.org/.

OMG Specifications

As noted, OMG specifications address middleware, modeling and vertical domain frameworks. All OMG Specifications are available from the OMG website at:

http://www.omg.org/spec

Specifications are organized by the following categories:

Business Modeling Specifications

Middleware Specifications

- **CORBA/IIOP**
- **Data Distribution Services**
- **Specialized CORBA**

IDL/Language Mapping Specifications

Modeling and Metadata Specifications

- **UML, MOF, CWM, XMI**
- **UML Profile**

Modernization Specifications

Platform Independent Model (PIM), Platform Specific Model (PSM), Interface Specifications

- **CORBAServices**
- **CORBAFacilities**

OMG Domain Specifications

CORBA Embedded Intelligence Specifications

CORBA Security Specifications

All of OMG's formal specifications may be downloaded without charge from our website. (Products implementing OMG specifications are available from individual suppliers.) Copies of specifications, available in PostScript and PDF format, may be obtained from the Specifications Catalog cited above or by contacting the Object Management Group, Inc. at:

OMG Headquarters
109 Highland Avenue
Needham, MA 02494
USA
Tel: +1-781-444-0404
Fax: +1-781-444-0320
Email: *pubs@omg.org*

Certain OMG specifications are also available as ISO standards. Please consult http://www.iso.org

Typographical Conventions

The type styles shown below are used in this document to distinguish programming statements from ordinary English. However, these conventions are not used in tables or section headings where no distinction is necessary.

Times/Times New Roman - 10 pt.: Standard body text

Helvetica/Arial - 10 pt. Bold: OMG Interface Definition Language (OMG IDL) and syntax elements.

`Courier/Courier New - 10 pt. Bold:` Programming language elements.

Helvetica/Arial - 10 pt: Exceptions

NOTE: Terms that appear in italics are defined in the glossary. Italic text also represents the name of a document, specification, or other publication.

1 Scope

1.1 Overview

This specification is part of a family of specifications called the Financial Industry Business Ontology (FIBO).

FIBO is a modularized formal model of the concepts represented by finance industry terms as used in official financial organization documents such as contracts, product/service specifications and governance and regulatory compliance documents. This is referred to as a *Business Conceptual Model* as distinct from models or descriptions of data or IT implementations.

The scope of *finance industry* encompasses a broad range of organizations that manage money, including credit unions, banks, credit card companies, insurance companies, consumer finance companies, stock brokerages, investment funds and some government sponsored enterprises.

This particular specification defines the **Foundations** module of FIBO: a set of business concepts which are intended to support the financial industry terms semantics presented in other FIBO specifications.

Foundations is itself segmented into a number of models or ontologies.

The FIBO Foundations models define general concepts that are not unique to the financial industry, but needed to help define the financial concepts. FIBO Foundations therefore includes a number of basic legal, contractual and organizational concepts, among others. Concepts which are available in other industry standards are not included, but in some cases a “Proxy” concept is included for reference, for example for address and country concepts. The rationale for including these is two-fold:

- Concepts in the financial industry are generally specializations of more general, non-financial concepts such as contracts, commitments, transactions, organizations and so on, These are included in FIBO Foundations so that specializations of them may be defined in other FIBO specifications;
- Properties of financial industry concepts frequently need to be framed in terms of relationships to non-financial concepts such as countries, jurisdictions, addresses and the like. These are included in FIBO Foundations so that properties in other FIBO specifications may make reference to them.

FIBO concepts are documented using two forms of definition:

1. a structured ontology specification of the concept, and its relationships to others, represented using the Web Ontology Language (OWL).
2. natural language definitions which represent the concepts in natural language using the vocabulary of the finance industry.

This specification covers both the content of the models, and the underlying architecture employed for producing and presenting the model.

A number of informative annexes are provided to assist potential users with adoption and implementation of this and other FIBO specifications.

1.2 Applications and Uses of FIBO

One of the key benefits of FIBO with respect to data, message or reasoning metamodels is that it can provide a semantic anchor firmly rooted in the concepts as understood and used by people in the finance industry. FIBO enables the creation of logical data models such that those logical models derive their formal semantics from FIBO.

FIBO supports the derivation of ontologies to support semantic reasoning and querying applications. Since FIBO itself is framed using the formal constructs of the OWL language, such operational ontologies may be derived directly from the FIBO conceptual ontologies, with adaptation as necessary to support any application specific constraints.

FIBO allows disambiguation of new and existing regulation. To the extent that regulatory requirements reference the formal concepts in FIBO, terms referred to in these regulatory requirements, or in reports that are mandated, would be semantically unambiguous.

One important goal of FIBO is for the formal business definitions to be used in legal documents such as contracts, terms and conditions of sales and payment, IP protection, compliance reports; and to underpin less formal language used in advertising and customer-facing websites.

The business terms and definitions in this specification may be used as a reference model to which firms would tie their own proprietary models (semantic models or ontologies); and also as a catalog for all of the relevant data models.

1.3 How FIBO is Different from Operational Ontologies

Intended Audiences*: Technical modellers, data architects*

An ontology, regardless of how it is to be used, sets out formally a representation of items in a real-world domain of discourse. There are two distinct uses to which this applies:

- A business ontology (business conceptual model) as described in this specification – this uses the full expressive power of the chosen notation to formally define items in the domain of discourse, without taking application technical constraints into account
- An operational ontology is constrained to operate within the parameters of a specific semantic application. Typically, this will contain a sub-set of the constructs in the business conceptual ontology, and that sub-set will typically comprise a decidable ontology.

It is necessarily the case that when something is to be used in an application, there will be technical constraints imposed upon that application. This is just as true when the application includes an ontology, as for other technologies.

The technical constraints that may apply to an operational ontology, necessarily do not apply to a business conceptual ontology. That is, the existence of some technical constraint in the application domain should not in any way influence the way in which business facts are formally captured and modeled in a business conceptual ontology.

1.4 How FIBO is Different from Data Models

FIBO can be distinguished from document/message/data/reasoning schemas of all kinds.

- FIBO models things in the real or planned world of the finance industry.
- FIBO will only contain instances of its own concepts under the specific conditions listed below. With these exceptions, FIBO contains only concepts - even if those concepts have just single instances in the real or planned world of finance.
    - Instances which are needed in order to define properties which refer to them;
    - Classes of thing which are defined extensionally; and
    - Examples
- FIBO is not any kind of a data, message or reasoning model, although it adds great value to these. It does not model document/message/data content or schemas optimized for reasoning.

FIBO will not include concepts about the structure of content, messages, information or data, even if that data is in turn about the finance industry.

The FIBO model, is referred to here as a "Business Conceptual Model", corresponding to Level 2 of the Zachman Framework for Information Architecture.

The distinctions between the scope of the FIBO model, and that of both logical and physical models, are further described in Annex C.

1.5 Definitions

The human readable definitions have been constructed by and with the input of business subject matter experts.

Many definitions have been derived from definitions of data elements corresponding to those terms in industry data or messaging standards. These have been adapted where necessary to ensure that they are descriptive of the thing or fact itself and not of data elements for data about those things or facts, and have then been reviewed by industry subject

matter experts to ensure that such adaptation accurately captures the sense of the business concept. In cases where the definition in a data or message standard was incomplete, context-specific or tautologous, a fresh definition was framed by the industry subject matter experts who participated in these reviews, or a third party definition was proposed and adopted.

1.5.1. Definitions Policy

In some cases, definitions have been obtained from third party sources. The policy for arriving at definitions for the FIBO industry terms was as follows (and remains so for future iterations and extensions):

1. In the absence of a definition endorsed by the subject matter experts for a term, "Barrons DICTIONARY OF FINANCE AND INVESTMENT TERMS, 8th Edition John Downes and Jordan Elliot Goodman" shall be used.

2. If a term and its acceptable definition is not in the Barrons Dictionary, then http://www.investopedia.com/dictionary/ shall be the authoritative source, subject to licensing requirements being met.

3. If a term and its acceptable definition is not in either the Barrons Dictionary or the investopedia dictionary, then http://www.bankersalmanac.com/addcon/dictionary/ shall be the authoritative source.

4. If a term has no acceptable definition in these Financial Industry sources or does not exist in these Financial Industry sources then http://www.merriam-webster.com shall be the authoritative source.

5. When there is a conflict with the definition of a Financial Industry term with the same term in another Industry, the Financial Industry definition will be used within FIBO.

In all cases the source from which the definition was obtained, or from which it was adapted, is recorded in annotation metadata for that concept.

2 Conformance

2.1 Overview

This clause defines conformance points for the following types of artifacts:

- Technical applications of FIBO such as logical data models, XML schemas, operational ontologies, code, and other technical artifacts
- Extensions of FIBO
- Representations of FIBO for business consumption
    - In diagrams
    - In spreadsheets or tables

Conformance of technical applications of FIBO is the most important conformance point, because it addresses the core issue of what it means to conform to the ontologies that FIBO defines. In comparison, conformance of extensions and representations, while still important, are somewhat secondary concerns.

Note that in addition to conformant applications, there are a number of scenarios in which someone may make use of the FIBO ontologies as a business conceptual model while applying their own design to meet their requirements. It is not possible to define specific conformance points for each of the possible ways in which one may legitimately develop a conventional database application or an operational OWL ontology that would be a good application. The non-normative annex [Annex E] describes a number of acceptable model architectures which may adequately reflect the material in FIBO Foundations and any of the other FIBO specifications.

2.2 Conformant Technical Applications of Model Content

Technical applications of FIBO content are logical data models, XML schemas, operational ontologies, code artifacts, and other technical artifacts that purport to conform to FIBO.

2.2.1 Assessing FIBO Model Conformance

Given that a technical application includes a set of information elements some of which correspond to the concepts in FIBO, then the application is *FIBO Model Conformant* if and only if:

- At least one of those information elements corresponds to a concept in the FIBO ontology for which conformance is claimed
- The application does not permit actual data to exist which would not be valid set of instances of those corresponding FIBO concepts: in other words if the data is represented as a set of individuals of the corresponding FIBO concepts then they will constitute a valid FIBO model with no contradictions

It is permissible for the information elements to have additional information or to be more constrained than those in FIBO.

2.2.1.1 Full FIBO Foundations Model Conformance

If a technical application is FIBO Model Conformant with the complete set of FIBO Foundations ontologies, then the application satisfies Full FIBO Foundations Model Conformance.

2.2.1.2 FIBO Ontology Model Conformance

If a technical application is FIBO Model Conformant with a particular FIBO Foundations ontology, then the application satisfies FIBO Ontology Conformance for that particular ontology. There is thus a separate compliance point for each ontology in Clause 10.

2.2.2 Assessing FIBO ODM Conformance

An extension of FIBO is FIBO ODM conformant if it is expressed in ODM (the OMG Ontology Definition Metamodel) and also restricts itself to using only the sub-set of ODM modeling constructs defined in the Architecture clause of this specification (8)

If the technical application is not an OWL ontology, then by definition the application is not FIBO ODM Conformant.

2.3 FIBO Extension Conformance

This definition of conformance points applies both to extension of the model content for use locally and to the preparation for submission of new model content for FIBO itself. The following conformance points may be asserted for each ontology that extends FIBO itself:

- FIBO-Full Extension in ODM: Satisfies FIBO Extension Conformance (see below) and FIBO ODM Conformance
- FIBO-Full Extension in OWL: Satisfies FIBO Extension Conformance (see below) and OWL2 Conformance

In turn, for *FIBO Extension Conformance* an ontology must satisfy FIBO Model Conformance (see 2.2.1) and the rules in the following three sub-clauses related to labeling, model consistency and relationship to subject matter.

2.3.1 Labeling

Business-facing labels shall be provided for all named model constructs. These labels must conform to the following formal requirements:

- Labels shall use normal English expression including spaces and punctuation, using lowercase except for proper nouns.
- Labels shall represent a plain English name (in US English spelling) which is that most commonly used by the finance industry.
- Labels do not need to be unique across the model.

- At least one business-facing label shall be present which is not in the form of, or contain, acronyms (including business acronyms) except where these are the only means by which the concept may be referred in the business domain (for example "CDO Squared").

2.3.2 Model Consistency

Reasoning is the mechanism by which the logical assertions made in an ontology and related knowledge base are evaluated by an inference engine. A logical assertion is simply an explicit statement that declares that a certain premise is true. Such assertions, taken together, form a logical theory, and a consistent theory is one that does not contain any logical contradictions. This means that there is at least one interpretation of the theory in which all of the axioms contained therein are provably true. The logical assertions expressed in the FIBO Foundations ontologies have been checked using multiple inference engines, designed specifically to support OWL 2, for internal logical consistency *(i.e.*, for consistency within that single ontology), and for logical consistency with imports closure (meaning, consistency including all axioms in any imported ontology in addition to those in the single ontology in question).

In order for any extension to FIBO to be conformant, it must be verified as being logically consistent (internally and with respect to imports) in addition to syntactically correct according to the OWL specifications. Examples of reasoning engines that can be used to verify logical consistency of an OWL 2 ontology are discussed in an article on Wikipedia[1]. Members of the OMG Ontology Special Interest Group (ontology@omg.org) can also make recommendations for tooling that might assist FIBO users in verifying their extensions.

In addition to being logically consistent, a conformant FIBO extension must be a conservative extension of each FIBO ontology that it imports i.e. the extension must not prove new logical assertions about the concepts defined in the imported ontologies. More formally, any logical assertion regarding concepts drawn exclusively from the vocabulary of an imported FIBO ontology is provable in a conformant extension if and only if it is provable within the imported ontology. This condition ensures that conformant FIBO extensions use the concepts defined in the imported FIBO ontologies without changing their meanings by narrowing or constraining them and supports composability of conformant FIBO extensions. As for logical consistency, reasoners can be used to verify that an OWL2 ontology is a conservative extension of an imported ontology but in general it is a more difficult problem so reasoners will take longer to determine this. Pragmatic guidelines like prohibiting restrictions on imported concepts can help ensure that extensions are conservative but in general it is possible to restrict imported concepts indirectly in subtle ways and so a reasoner should be used to verify conformance.

2.3.3 Relationship to Subject Matter

In any extension to FIBO model content each model element which is a class, an object property or a datatype property shall correspond to some item in the real world. No model element shall refer to some technical construct such as a database field, internal identifier, database key and the like.

An exception is made for information constructs which are themselves important and publicly shared parts of the business domain, such as publicly issued identifiers, security identifiers, ratings codes and the like. In each such case, there shall be some formally identified scheme in which the code in question is defined.

A suitable test for types of "Information" to be considered real is whether that information is publicly shared or, if private, made available across the business supply chain. Examples include Legal Entity Identifier, securities prospectuses, published indices, interest rates.

2.4 Conformant Business Presentation of Model Content

There are two conformance points for presentation of FIBO content:

- FIBO Business Diagram
- FIBO Business Table

[1] http://en.wikipedia.org/wiki/Semantic_reasoner

Any tool which asserts support for one or other or both of business presentation conformance points must be able to import the available FIBO content in at least one of the available serialization formats (UML XMI, ODM XMI or OWL), and produce diagrams and/or tables which conform with the requirements defined for the conformance point.

For the avoidance of doubt, this sub-clause describes diagrams which are to be presented to business subject matter expert and not diagrams which form part of this specification itself. FIBO Model Conformance may be asserted without reference to this conformance point, and model content may be presented to various audiences including business audiences without asserting conformance to this sub-clause.

2.4.1 General Requirements

It is a requirement of this specification that content of the models is made available to people in the business domain in one or more of a set of diagrams and tables which are described in this specification. A presentation of FIBO model content is not a conformant FIBO Business Presentation (i.e. a conformant FIBO Business Diagram or a conformant FIBO Business Table) if the only means for the reader to view the model's terms, definitions and relationships is one which requires some formal understanding of some model language such as UML or OWL, beyond the knowledge conveyed by the annexes to this specification. For the avoidance of doubt, a non-conformant business presentation is any format which contains symbols, whether diagrammatic or textual, which have a meaning other than the meaning which a reasonably educated but non-technical person would ascribe to those items

2.4.2 FIBO Business Diagram Conformance

OWL features such as restrictions on properties or classes, where these are present in the model content, shall be rendered in some way that communicates their business intent without reference to the way in which the OWL syntax represents these constructions.

OWL constructs shall be represented by simple constructs which do not require specialist technical training, such as boxes, arrows and lines.

All notation on all diagrams shall only represent features of OWL, except where this is clearly identified as additional annotation (intended to enhance an understanding of the business content of the model and not part of the model itself).

In diagrams generated from OWL tools or other non UML based tooling, no features shall be present which do not represent some feature of OWL except where these are clearly identified as visual decorations intended to enhance an understanding of the business content of the model.

If UML Generalization notation is used, this shall be laid out with the "arrowhead" pointing vertically upwards, in either the vertical tree style or direct style of routing. Generalization relationships may also be represented using more intuitive, non UML notations, in which case this requirement shall not apply.

2.4.3 FIBO Business Table Conformance

This sub clause concerns tabular presentations. These may range from basic presentation of terms, definitions and synonyms, to tables which represent all of the model content or all model content except relationships between relationships. Conformant FIBO Business Tables may be rendered as spreadsheets or as textual documents in a tabular layout.

2.4.3.1 Basic Tables

A conformant FIBO Business Table using the "Basic" tabular format shall show only the following entries:

- Term (preferred label for concept)
    - Classes and properties may be in the same column or different columns
    - Synonyms can be in the same column (in parentheses under the preferred label) or a separate column headed Synonyms
- Definition

These shall be labeled as such.

This table shall only show those constructs from the FIBO model content which represent meaningful business concepts, and not the additional constructs which deal with the set theoretic logic of the model. That is, the basic table shall show only (differentiating between them):

- Class
- Relationship Property
- Simple Property

2.4.3.2 Extended Tables

A conformant FIBO Business Table using the Extended Tabular format shall conform with the following requirements:

The extended table shall encompass each of the basic model features, as follows (they may be covered by separate columns or multiple features may be grouped into a smaller number of columns, distinguished by sub-headings)

- Term
- Definition
- Synonym
- Range of Simple Properties (titled as "Simple Type")
- Range of Relationship Properties (titled as "Related Thing")
- Property type
- Super (class or property) (can be labeled as “Parent”)
- Disjoints (labeled "mutually exclusive")
- Additional metadata may or may not be shown, at the discretion of the modeler and as appropriate to the intended usage (for example, review notes annotations).

The following model constructs shall be included in the Extended Table reports, in or near the following order:

- Class (including classes which are equivalent to logical unions and classes which are enumerations of individuals)
- Relationship Property
- Simple Property
- The relationship between any named class which represents a logical union, and those classes which are members of that union
- Individuals
    - 'typeOf ' relationships from Individual to Class (labeled "type of")
- Enumerated sets of individuals
    - ‘oneOf relationships from the enumerated class to each of the individuals members of that set
- Annotations – there are no specific requirements for how these are presented.

Object Properties and Datatype Properties shall be included once in all reports across the model, and this shall be for the ontology which defines that property.

The intention of these requirements is that the report shows each type of fact, once only and in a logical order.

3 References

3.1 Normative References

The following normative documents contain provisions which, through reference in this text, constitute provisions of this specification. For dated references, subsequent amendments to, or revisions of, any of these publications do not apply.

Reference	Description
[Dublin Core]	DCMI Metadata Terms, Issued 2013-06-14 by the Dublin Core Metadata Initiative. Available at http://www.dublincore.org/documents/dcmi-terms/.
[ISO 1087]	ISO 1087-1:2000 Terminology — Vocabulary — Part 1: Theory and application
[ISO 4217a]	ISO 4217 Codes for the representation of currencies and funds, Seventh edition, 2008-07-15
[ISO 4217b]	ISO 4217 Currency and funds code list, 2014-03-28
[MOF Core]	Meta Object Facility (MOF™) Core, v2.4.1. OMG Available Specification, formal/2011-08-07. Available at http://www.omg.org/spec/MOF/2.4.1/.
[MOF XMI]	MOF 2/XMI (XML Metadata Interchange) Mapping Specification, v2.4.1. OMG Available Specification, formal/2011-08-09. Available at http://www.omg.org/spec/XMI/2.4.1/.
[ODM 1.0]	Ontology Definition Metamodel (ODM), v1.0. Available Specification, formal/2009-05-01. Available at http://www.omg.org/spec/ODM/1.0/.
[ODM 1.1]	Convenience Specification for the Ontology Definition Metamodel (ODM), v1.1, available from the ODM 1.1 RTF.
[OMG AB Specification Metadata]	OMG Architecture Board recommendations for specification of ontology metadata, Available at http://www.omg.org/techprocess/ab/SpecificationMetadata/
[OWL 2]	OWL 2 Web Ontology Language Quick Reference Guide (Second Edition), W3C Recommendation 11 December 2012. Available at http://www.w3.org/TR/2012/REC-owl2-quick-reference-20121211/.
[RDF 1.1]	RDF 1.1 Concepts and Abstract Syntax, W3C Last Call Working Draft. Latest version Available at http://www.w3.org/TR/2013/WD-rdf11-concepts-20130723/
[RDF Concepts]	Resource Description Framework (RDF): Concepts and Abstract Syntax. Graham Klyne and Jeremy J. Carroll, Editors. W3C Recommendation, 10 February 2004. Latest version is available at http://www.w3.org/TR/rdf-concepts/.
[RDF Schema]	RDF Vocabulary Description Language 1.0: RDF Schema. Dan Brickley and R.V. Guha, Editors. W3C Recommendation, 10 February 2004. Latest version is available at http:// www.w3.org/TR/rdf-schema/.
[SKOS]	SKOS Simple Knowledge Organization System Reference, W3C Recommendation 18 August 2009. Available at http://www.w3.org/TR/2009/REC-skos-reference-20090818/.

Reference	Description
[UML2]	Unified Modeling Language™ (UML®). . Available at http://www.uml.org/index.htm
[Unicode]	*The Unicode Standard, Version 3*, The Unicode Consortium, Addison-Wesley, 2000. ISBN 0-201-61633-5, as updated from time to time by the publication of new versions. (See http:// www.unicode.org/unicode/standard/versions/ for the latest version and additional information on versions of the standard and of the Unicode Character Database).
[UTF-8]	RFC 3629: UTF-8, a transformation format of ISO 10646. F. Yergeau. IETF, November 2003, http://www.ietf.org/rfc/rfc3629.txt
[W3C Datatypes in RDF and OWL]	XML Schema Datatypes in RDF and OWL, W3C Working Group Note 14 March 2006, Available at http://www.w3.org/TR/2006/NOTE-swbp-xsch-datatypes-20060314/.
[XML Schema Datatypes]	XML Schema Part 2: Datatypes. W3C Recommendation 02 May 2000. Latest version is available at http://www.w3.org/TR/xmlschema-2/.

3.2 Non Normative References

The following informative documents are referenced throughout this text or in parts of the Annexes:

Reference	Description
[DOLCE]	A. Gangemi, N. Guarino, C. Masolo, A. Oltramari, and L. Schneider. Sweetening ontologies with DOLCE. In Proceedings of EKAW, Siguenza, Spain, 2002.
[ISO Common Logic]	Information Technology - Common Logic ISO/IEC 24707:2007 http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=39175
[ISO MDR]	Information technology — Metadata registries (MDR) — Part 3: Registry metamodel and basic attributes, ISO/IEC 11179-3:2013, Available at http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=50340
[Knowledge Representation]	*Knowledge Representation: Logical, Philosophical and Computational Foundations*, Sowa, John F., Brooks/Cole. 2000
[Model Theory]	*Mathematical Logic: An Introduction to Model Theory*, Lightstone, A. H., New York: Plenum Press, 1978, H. B. Enderton (ed).
[OMV]	Ontology Metadata Vocabulary (OMV) - http://omv2.sourceforge.net/ (a standard giving metadata for ontology-level information)
[C S Peirce]	*A Comprehensive Bibliography and Index of the Published Works of Charles Sanders Peirce, with a Bibliography of Secondary Studies*, Ketner, K. L. et al., Johnson Associates (Greenwich, Connecticut): 1977
[W3C Organization Ontology]	W3C Organization Ontology. Available at: http://www.w3.org/TR/vocab-org/
[Zachman]	Zachman Framework http://www.zachman.com/

3.3 Changes to Adopted OMG Specifications

This specification does not change or replace any OMG specifications. It does, however, depend on pending changes to the Ontology Definition Metamodel (ODM), in support of OWL 2 and RDF 1.1.

4 Terms and Definitions

For the purposes of this specification, the following terms and definitions apply.

Content

Definition: Subject matter or meta-content.

Business conceptual model

Definition: A model which represents and only represents business subject matter without reference to the design of any solution or data model representation.

Business publication

Definition: Representation of a subject matter view in a form that is understandable and usable by business users.

Example: Text document, web page, audio recording, interactive search dialog

Business subject matter

Definition: Subject matter that defines and describes the kinds of people (and the roles they play), organizations and other things that an enterprise has to deal with in the course of its operational business, regardless of how this content is presented to the people in the organization (e.g. in text documents, web pages, audio broadcasts).

Example: Business concepts, such as: *OTC derivative*, *business day*

Example: Relationships between business concepts, such as: *swap transaction has ISDA confirmation*

Example: Constraints, such as: Each ISDA confirmation is of exactly one swap transaction

Example: Descriptions, such as: ISDA is the largest trade organization of participants in the OTC derivatives market.

Example: Business processes (defined in terms of the business concepts), such as:

If a Disputing Party reasonably disputes the Value of any transfer of Eligible Credit Support, then the Disputing Party will notify the other party not later than the close of business on the Local Business Day following.

Note: Business subject matter is mainly about kinds of thing, but may include individuals, in three roles: (1) as one-of-a-kind things referenced in the subject matter, such as *ISDA*, *Dodd-Frank Act*, *EC Treaty*; (2) As types defined by enumeration, such as the currencies in which a trading business maintains accounts; (3) in examples.

Note: Business subject matter is usually scoped by area of business jurisdiction (or something similar), such as, say, derivatives trading. The business subject matter is about the business of derivatives trading.

Other areas of responsibility in the enterprise have different subject matter. For example, the IS department's subject matter includes information models of things in the operational business (including derivatives trading). The finance department's subject matter includes financial models of things in the operational business.

From the derivatives trading perspective (the relevant parts of) these information and financial models would be considered meta-content.

Business subject matter view

Definition: Subset of business subject matter that is intended to be presented in some business publication.

Example: Concept definitions; relationship definitions with constraints.

Extension

Definition: The membership of some class of thing. This is distinct from its intension, that is the properties intrinsic to that class of thing. In applying the intension of some class to some collection of individuals, one arrives at the extension of that class for that collection.

Extensional

Definition: Logic explicable solely in terms of extensions; ignoring differences of meaning that do not affect the extension.

Extensional Definition of Class Membership

Definition: The definition of membership of a class by direct articulation of those members (that is, by articulation of the Extension of that class.

Intension

Definition: The properties intrinsic to some class of thing.

Intensional

Definition: Logic (of a predicate) incapable of explanation solely in terms of the set of objects to which it is applicable; requiring explanation in terms of meaning or understanding.

Intensional Definition of Class Membership

Definition: The definition of membership of a class according to properties intrinsic to members of that class.

Meta-content

Definition: Information about subject matter

Example: Control information, such as: date and author of last update, external source, owner

Example: Connection of subject matter items to content outside the subject matter scope, such as data model elements that correspond to them (and point to the storage of instance data).

Model-Theoretic Conformance

Definition: The manner in which some model conforms with some theory about what it is intended to model and how it is intended to model it.

Ontology

Definition: A formalization of a conceptualization. For the purposes of this specification the formalization is in OWL, using ODM as a means to render this, and the conceptualization is that of business subject matter.

Operational Ontology

Definition: An ontology which is intended for use within some application.

Subject matter

Definition: Information about things in the universe of discourse; the essential facts, data, or ideas that constitute the basis of spoken, written, or artistic expression or representation; often : the substance as distinguished from the form especially of an artistic or literary production.

Taxonomy

Definition: A set of terms which stand in some classification relation to one another.

Terminology

Definition: The overall disposition of ontologies of concepts and vocabularies of terms, in relation to one another.

Vocabulary

Definition: A set of words, each giving one or more formal definitions which apply to a meaningful concept that is referred to by that word.

5 Symbols and Abbreviations

5.1 Symbols

There are no symbols introduced by this specification.

5.2 Abbreviations

The following abbreviations are used throughout this specification:

- OWL – Web Ontology Language
- ODM – Ontology Definition Metamodel
- RDF – Resource Definition Framework
- SME – Subject Matter Expert
- UML – Unified Modeling Language
- URI – Uniform Resource Identifier
- URL – Uniform Resource Locator
- XMI – XML Metadata Interchange
- XML – eXtensible Markup Language

Additional symbols and abbreviations that are used only in annexes to this specification are given in those annexes.

6 Additional Information

6.1 How to Read this Specification

6.1.1 Audiences

This specification has the following audiences:

- The standards community
- The finance industry business community
- The regulatory community
- Technical architects
- Semantic Modelers

Each clause opens with a statement identifying the intended audience for that clause. The language in that clause is then framed appropriately for readers from that audience. Where "Intended Audience" is not stated the material in that clause is intended to be comprehensible to all general readers.

6.1.1.1 Standards Community

This audience is intended to be able to follow and validate the way in which this specification sets out the arrangements for the production and maintenance of model content, and the production of business facing reports and diagrams representing parts of that content.

6.1.1.2 The Finance Industry Business Community

As noted in the clause on conformance (2) this specification includes detailed requirements for the production of diagrams and reports that are intended for consumption by business subject matter experts. This specification also contains material addressed at this audience, this being an informative annex on "Interpreting Model Content". This audience is not intended to read and understand the remaining parts of this specification.

6.1.1.3 The Regulatory Community

As for Finance Industry Business Community.

6.1.1.4 Technical Architects

These include but are not limited to:

- Tooling vendors and developers
- Other content providers / enriched content providers
- Business Analysts – anyone who use the model on site, whether they are a modeler, a metadata analyst, etc.
- Technology Management

The bulk of the "Architecture" clause is intended to be read and understood by these audiences and by the 'Semantic Modelers' audience.

6.1.1.5 Semantic Modelers

Much of the material in this specification is intended to be read and understood by semantic modelers. This includes the 'Conformance' clause (2), the ‘Architecture’ clause (8) and the non normative Annex D on implementing and extending this model and proposing new model content.

The Semantic modeler audience is not the same as the technical audience, although some individuals may possess skills in both. clauses of this specification which are written for a semantic modeling audience do not require any training in any formal technology in order to understand and act upon their contents. These clauses do require a clear understanding of semantics and formal logic. It is not necessarily the case that technical readers are expected to be able to read and understand all aspects of the semantic modeling material. It should also be noted that some terms which have specific meanings in one or more technology environments, may have different (or often only subtly different) meanings to the semantic modeling audience. Where both semantics and technical audiences are intended to read a clause, care has been taken to try to use all of the applicable terms and qualify words which have multiple different usages to these audiences.

6.2 Acknowledgements

The following organization submitted this specification:

- Enterprise Data Management Council

The following companies have provided significant expertise and resources in the development of its content and architecture:

- Adaptive Inc.
- Australia and New Zealand Banking Group
- AVOX/DTCC
- Bank of America
- Barclays Capital
- BBH
- Bloomberg
- Business Semantics
- CIBC
- Citigroup Inc.
- Credit Suisse Group AG
- CUSIP
- The Federal National Mortgage Association (Fannie Mae)
- David Frankel Consulting
- FacetApp
- Fidelity
- GoldenSource Corporation
- HSBC Holdings plc
- JPMorgan Chase & Co.
- The Manufacturers Life Insurance Company
- Michigan State University
- Model Driven Solutions
- Model Systems

- Morgan Stanley
- MphasiS
- National Australia Bank
- No Magic
- Nomos Software
- Nordea Bank
- Oakland University
- OntoAge
- OpenFinance
- PricewaterhouseCoopers LLP
- Revelytix
- Sallie Mae
- SAP
- Semantic Arts
- State Street
- Sungard
- SWIFT
- Tahoe Blue
- Thematix Partners LLC
- Thomson Reuters
- UBS AG
- University of British Columbia
- University College Cork
- Wells Fargo
- Wizdom Systems, Inc.

6.3 Notation

The diagrams included herein are ODM-compliant UML diagrams, in other words, they conform to the UML Profiles for RDF and OWL specified in the OMG's Ontology Definition Metamodel [ODM] Specification. This includes the set of UML stereotypes and graphical notation used in the diagrams provided.

The color scheme employed in these diagrams includes:

- Basic OWL Classes: white for classes defined within the current (local) ontology, amber for classes defined within an imported (referenced) ontology
- OWL Restriction Classes and other Class Expressions (unions, intersection, complements): green
- OWL Object Properties: blue
- OWL Data Properties: dark gray
- OWL Datatypes: pink

- OWL Individuals: light gray

Within the context of this specification (and the FIBO specifications on which it depends), a module is group of ontologies, organized as a subdomain with respect to the FBC namespace and as a folder from a file management perspective. One or more ontologies are contained in each of the three modules in this specification, which include FinancialInstruments, FunctionalEntities, and ProductsAndServices. For each module there is an "about" file, which provides metadata about the module, specified in tabular form. Each of the primary ontologies in a given module is defined as an ODM-compliant UML model as well as in OWL (aside from the "about" file, which is expressed in RDF/XML only). The normative ontology is expressed in ODM XMI (*i.e.*, XMI that conforms to the ODM metamodels for RDF and OWL), ODM UML XMI (*i.e.*, that conforms to the UML Profiles for RDF and OWL in the ODM specification), and in RDF/XML serialized OWL 2.

The notation used to represent description logic expressions (*i.e.*, the expressions in the Parent columns in class tables containing ontology details) is consistent with the notation defined in the Description Logic Handbook [DL Handbook]. Some of the basics are described in Table 6-1, below. Note that this is not intended to be comprehensive, but includes the primary patterns that are used in the FIBO FBC specification, for property restrictions in particular.

Table 6-1 Description Logic Expressions Notation

Construct	Description	Notation
Boolean Connectives and Enumeration		
intersection	The intersection of two classes consists of exactly those individuals which are instances of both classes.	C ∩ D
union	The union of two classes contains every individual which is contained in at least one of these classes.	C ∪ D
enumeration	An enumeration defines a class by enumerating all its instances.	oneOf (i_1, i_2, i_3, … i_n)
Property Restrictions		
universal quantification	Universal quantification is used to describe a class of individuals for which all related individuals must be instances of a given class (*i.e.*, allValuesFrom in OWL).	∀R.C, where R is the relation (property) and C is the class that constrains all values for related individuals
existential quantification	Existential quantification is used to define a class as the set of all individuals that are connected via a particular property to at least one individual which is an instance of a certain class (*i.e.*, someValuesFrom in OWL).	∃R.C, where R is the relation (property) and C is the class that constrains some values of related individuals
individual value	Individual value restrictions are used to describe classes of individuals that are related to one particular individual (*i.e.*, hasValue in OWL).	∀R.I, where R is the relation (property) and I is the individual
exact cardinality	Cardinality (number) restrictions define classes by restricting the cardinality on the sets of fillers for roles (relationships, or properties in OWL). Exact cardinality restrictions restrict the cardinality of possible fillers to exactly the number specified.	= n R (for unqualified restrictions) = n R.C (for qualified restrictions, i.e., including onClass or on DataRange)
maximum cardinality	Maximum cardinality restrictions restrict the cardinality of possible fillers to at most the number specified (inclusive).	≤ n R (for unqualified restrictions) ≤ n R.C (for qualified restrictions)
minimum cardinality	Minimum cardinality restrictions restrict the cardinality of possible fillers to at least the number specified (inclusive).	≥ n R (for unqualified restrictions) ≥ n R.C (for qualified restrictions)
Class Axioms		

equivalent classes	Two classes are considered equivalent if they contain exactly the same individuals.	≡ C
disjoint classes	Disjointness means that membership in one class specifically excludes membership in another.	¬ C

Within the tabular representation for restrictions in the tables included herein, the identifiers for the restrictions shown in the diagrams are included parenthetically following the logic expressions. These are not part of the logic, but are included for comparison purposes. The identifiers are named based on the precedent set in the FIBO Foundations [Foundations] specification, which includes the namespace prefix for the ontology followed by a unique number.

Additionally, some restrictions are nested, whereby the content of an embedded (nested) restriction is also included parenthetically. In these cases, all of the identifiers will be included, also parenthetically, following the complete specification of the complex restriction. Note too that in the case of complex restrictions, where there are nested elements in parentheses, the “dot notation” used as a separator between a property and the role filler is replaced with the embedded parenthetical filler definition. A “role” from a description logic perspective is essentially a property in OWL, and the role “filler” is the class or individual that provides the value for that role in a given axiom (*i.e.* in a restriction or other logic expression).

For the vast majority of the property restrictions specified in FIBO, the restrictions are defined as necessary conditions for class membership, rathe r than sufficient conditions. As a result, the tables assume that necessary conditions is the default and only in cases where a restriction imposes sufficient conditions will that be stated.

6.4 Interpreting the Business Model Content

Intended Audiences*: Business Subject Matter experts*

6.4.1 Introduction

The model content is intended by read and understood by business domain experts with knowledge of business entities and legal concepts. It requires no knowledge of modeling theory, technical modeling languages, technology development or data modeling.

The following knowledge is required to interpret the model content:

- Set theory
- Logic
- Business (commerce, law, finance)

6.4.2 The Model

6.4.2.1 What the Model Contains

The model described in this specification contains elements called 'Things', Simple Properties about those things in the form of unstructured information, and Relationship Properties in the form of relationships between one 'Thing' and another. Things, Simple Properties and Relationship Properties all have as a minimum the definition for the term that they represent, plus additional information on usage, review history, sources of terms and definitions and so forth.

6.4.2.2 Model Views

Whereas the information given in this specification conveys all of the model content, supplementary diagrams and tables may be created for a business audience which will not show all of this information, but only a sub-set. This sub clause makes reference to such formats and views, and is to be read by a business audience to understand what those views show. This sub clause contains no technical language about OWL or other modeling constructs but uses plain English alternative terms for those concepts.

The content of the model may be rendered in two basic forms: visual information in the form of diagrams, and textual

information in the form of tables. Diagrams may be made available in varying levels of detail and created to show different sets of terms and relationships across or within sections of the model. Textual information may be created as web based tabular reports and as spreadsheets. These should contain basic information of term, definition and synonym and in some cases will contain additional information about the types of thing or the types of information to which facts in the model refer. Business tables and spreadsheets need not show relationships between relationships as such information would be difficult to visualize in the tabular format.

Diagrams and tables should reflect the information retained in the underlying model repository directly. For example, if two 'Thing' elements have a relationship between them and they appear on the same diagram, the relationship between them should always appear.

6.4.2.3 Business Diagrams

Business diagrams reflect any set of terms in the model, within or across sections of the content. These may be rendered with varying levels of detail. Diagrams created during reviews of the subject matter will typically contain a greater range of terms than diagrams created for presentation to the wider community of potential users.

6.4.3 Interpretation

The model conveys 'Things' and 'Facts'. Facts are in two forms:

- 'Simple Properties': these are a statement about something which is framed in terms of some simple type of information, such as textual entries, yes/no answers, dates, numbers and selections of textual information
- 'Relationship Properties': these are a statement about something which is framed in terms of something else, that other thing also being framed as a kind of 'Thing'.

In addition, there are relationships which represent additional set theory concepts, notably logical unions, mutual exclusivity.

Each 'Thing' also has a 'Parent' relationship, with the sense of 'is a', typically shown as an upward point arrow on the diagrams. This relationship indicates that the thing from the non-arrowed end is “a kind of” the thing at the end with the arrow.

These concepts are described in the sub clauses which follow.

6.4.3.1 Thing

A Thing is a set theory construct. This is shown on the diagrams as a box with a name. On some diagrams, additional textual entries in the box show the Simple Properties about that thing.

A Thing is defined as the set of individuals which are defined according the facts (properties) given for that kind of thing. Membership of the set is defined in the sense that any individual in the world of which the stated facts are true or applicable, is a member of that set. In terms of logical theory, these sets are defined intensionally. It is also possible to define a set explicitly as a list of its members (in logical theoretic terms, an extensional definition).

6.4.3.2 Inheritance: the Parent 'is a' relationship

Each Thing in the model has one or more parent Things. The relationship between the Thing and its parent may be interpreted as an 'is a' form of relationship, meaning that the thing of which the parent relationship is shown is a kind of the thing to which the arrow in the Parent relationship is pointing.

This relationship formally indicates that the thing that has the Parent, inherits all of the facts about that parent. In addition, this relationship is transitive, meaning that the parent relationships of the parent are passed on to the child term. For example, if a share is a security and a security is a transferable contract then a share is a transferable contract.

The relationships of this type create a formal inheritance structure called a Taxonomy. Taxonomies in this sense may be single inheritance (as is often seen in technical model designs) or multiple inheritance. In the FIBO models these are multiple inheritance, meaning that types of thing (such as types of contract) may be classified in more than one way. So for example an interest rate swap is both a swap and an interest rate derivative.

As an example of multiple inheritance, one might say that in terms of the Linnaeus Taxonomy of Species, a whale is a mammal, while one may also create a set of taxonomic classifications based on habitat, in terms of which a whale may also be a marine animal.

On a technical note, the Parent relationship is functionally identical to the relationship known as 'Generalization' in the UML modeling language.

6.4.3.3 Simple Properties

Simple Properties are assertions about things in a class, which may be framed in terms of some simple type of information.

Types of information about which Simple Properties are asserted are:

- Text
- Date
- Number
- Whole number
- Yes/no answer
- Selection of textual descriptors

To a technical person these may easily be identified with what are called 'datatypes'. However these represent the types of information not data as such. A special case is the selection of possible answers - this refers to a list of entries (see Selection Lists).

6.4.3.4 Relationship Properties

A Relationship Property is defined as a fact about something which is framed in terms of a relationship to some other thing.

Relationship Properties are of the form subject-relationship-object where the subject is the Thing from which the relationships is drawn and the object is the thing to which the relationship points.

There are additional pieces of information about these Relationship Properties, such as whether they are symmetric, transitive and so on. The use and interpretation of these refinements to Relationship Properties are beyond the scope of this explanatory sub clause.

6.4.3.5 Logical Unions

Logical unions indicate that any individual which is a member of any of the classes of 'Thing' of which the union is a union, are members of that union.

Graphical methods for presenting a logical union for a business audience may vary, but should ideally present as a named class of Thing with relationships clearly identifiable as being “in union” relations, to the classes of thing which are members of the union. The underlying model consists of an anonymous union element and a named class, with an equivalence asserted between these two.

Relationship Properties may refer to unions in the same way that they refer to other classes of Thing.

6.4.3.6 Mutually Exclusive sets

Given that each thing is a set of potential members defined by their properties (facts), it is possible for any one thing in the world to be defined as being a member of more than one set, if the properties asserted for one set are not related to the properties asserted for another set.

Where membership of one set necessarily precludes membership of another set (that is, where a set is defined such as to specifically exclude members of another set), this should be shown by a relationship on the diagrams, labeled 'mutually exclusive'.

Where classes of 'Thing' are not indicated as being mutually exclusive (or have parents which belong to classes of Thing which are mutually exclusive), then any individual in the domain of discourse (the world) may belong to both sets.

This is formally known as a 'disjoint' relationship.

6.4.3.7 Relationship Properties hierarchies

Relationship Properties are themselves disposed in a hierarchy similar to that given for the classes of 'Thing'. These are indicated on more advanced diagrams by a relationship in a similar style to the Parent relationship line. The Relationship Property to which the arrow refers represents a more general meaning, of which the Relationship Property at the bottom of the relationship represents a narrower definition of the same meaning.

This is formally known as a “sub property of’ relationship.

In many cases Relationship Properties are reused in a narrower context without change to the meaning of the property. Such reuses are implemented in the model by the use of a kind of logical assertion called a “Restriction”. Business facing diagrams should aim to represent these in a clearly understandable way which indicates what property is reused and how its use is narrowed down.

The narrowing of these meanings frequently occurs in conjunction with the narrowing of the meanings of classes of 'Thing' in the taxonomy. For example, types of bond are classified (a narrowing or specialization of the meaning of 'bond') according to, among other things, a narrowing of the relationship 'issued by' with the latter relationships being distinguished form one another by the nature of the kind of party which is the issuer.

6.4.3.8 Inverse relationships

These are need only be shown on diagrams that relationships between relationships.

Relationship Properties in the model are all one-directional, by virtue of their being framed as 'subject-verb-object' triples. In the business domain, meaningful terms and definitions may exist in either direction between one class of thing and another (for example, a bank has a customer versus a person has an account at the bank.

In theoretical terms, this relationship only applies between relationships which are known as 'functional' relationships. An explanation of this is beyond the scope of this sub clause.

6.4.3.9 Selection Lists

A list of possible entries for a simple type is displayed as a box on the diagrams, with a list of the possible entries. These are displayed as text, and generally refer to lists of possible textual values for the Simple Property.

It should be noted that these do not or should not represent lists of kinds of 'Thing' - those would be represented as a taxonomy of actual things. This is an important difference between this and a data model, since many data models have similar selection lists, called 'enumerations' in the data modeling world, which may represent kinds of thing or classifications of the thing which has these as a property.

6.4.3.10 Selections of Things

This is a class or set of things of which the members are explicitly listed (in theoretical terms, an extensional definition of the class).

An extensional set of things as described above (also known as an enumeration – this is not the same thing as a data enumeration) takes the form of a named class of Thing and a set of “Individuals”, with a “one of” relationship between each individual thing and the named set of which it is a member.

7 Introduction

7.1.1 Reading this Standard

Technical audiences (in both conventional and semantic technology) are directed at the “Architecture” clause (8).

Business audiences (financial industry participants, regulators and others) are directed at the sub clause on interpreting model content (6.3) and the model content itself in Clause 10.

The business content defined in this standard is intended to be presented both in a business-facing format and in a complete, technical format. The latter is intended for consumption by technical and standards audiences only. This specification defines the content of the standard and the ways in which it is to be presented to business readers.

7.2 Usage Scenarios

Intended Audiences: *Technical implementers (conventional and semantic technology); technology management*

The model defined in this specification is intended for use as a business conceptual model.

The uses envisaged for the model are as follows:

- Model driven development
    - Of database schemas
    - Of message schemas
    - Of common messaging across a business unit or organization
- Semantic Technology development
- Integration of systems and/or data feeds

In addition, the model may be extended locally to extend the scope of what is modeled, prior to using such local extensions in any of the above usage scenarios.

7.2.1 Model Driven Development

Model Driven Development refers to the top town development of technical artifacts starting with a high level, business view of the requirements (for programs) or the data semantics (for data).

The model defined in this specification is intended to be situated within any model driven development framework, as a conceptual model and potentially extended locally with additional concepts. This is the case whether the development is for databases, messages or a combination of the two.

Analysis of the model and metadata provided may enable the automation or partial automation of the production of logical data models, or at least of a candidate starting point for the development of the logical data model prior to the addition of keys and other database requirements.

The model described and presented within this specification supports multiple inheritance between classes, whereas most logical data models would be developed using a single inheritance taxonomy (as this is often a constraint on the logical or physical models development). This model will contain metadata which defines, for multiple inheritance taxonomies, Such information can be interrogated to extract from the model a suitable single inheritance taxonomy appropriate to the requirements of the development.

If this model is used within a UML tool, users may create formal mappings between logical data model constructs and the semantics corresponding to these in the FIBO model content. This simplifies the validation and verification of technical data model artifacts.

7.2.2 Semantic Technology Development

As part of this specification, model content is made available in the Web Ontology Language (OWL) format, which is the format used in semantic technology applications.

However, semantic technology developers should be aware that the physical and technical constraints, which rightly apply to semantic technology applications, have not been imposed, since its primary purpose is to serve as a conceptual model at the business level.

Similarly, it should be noted that in defining the formal meanings of terms in the business domain, most of those meanings are “grounded” with reference to legal constructs, accounting constructs and so on. This may or may not correspond to instance data in the application. Typically a semantic technology application, like any other application, will operate on actual data.

There is therefore a distinct difference between the terms defined in this model to satisfy the requirements of a business conceptual model, and the terms required or to be found in an ontology that would be used in a semantic technology application.

Semantic Technology developers will therefore need to extract from the model content, some suitable and decidable sub-set of that content.

This specification does not detail exactly how to derive decidable sub-sets of the content, such as OWL-DL. It is left to the semantic technology developer to make the necessary transformations.

Some of the metadata provided with this model may assist in this.

7.2.3 Integration of systems and/or data feeds

The simplest application of this conceptual model is to simply use the terms as a common point of reference when comparing terms within different logical or physical data models. This would be of value for example when integrating different systems.

Many systems may not have a formally stated ontology for the data elements that they use, or the database schema may be considered to be the only record of the meanings of the terms therein. Typically, whenever two or more systems need to be integrated, there is a time consuming and almost open ended “mapping” exercise in which the meanings of each of the terms in each of the databases or message schemas involved in the integration, are guessed and perhaps written down.

In reality, even when the intended meanings of the elements in each database and message schema are known, there is not an easy one-to-one mapping between one system and another. This is typically the result of good design: the more the designs have made use of reusable common data structures, the more efficient that design is, but correspondingly the less explicit is the semantics of the terms.

In an integration project that brings together data elements from more than two systems or data feeds, the number of mappings that need to be carried out between one system or feed and another is a geometrical function of the number of such data sources and feeds. In order to have a mapping exercise which is only arithmetically related to the number of data sources and feeds, it is necessary to have a single “hub” of terms which are able to be used as a common point of reference between each of the data models.

While this can often be achieved using a single data model, in practice the limitations on data models (such as single inheritance taxonomies in many cases, though not all) mean that no one model can be found against which all terms in all data models and feeds may be cross referenced. The model presented as part of this specification, being a semantic model, contains full definitions of the meaningful concepts which may be referred to by any of the data elements in the data sources or feeds that need to be integrated, as long as this model may be extended locally to cover areas of scope which are not part of the current specification.

8 Architecture

Intended Audience: *Technical, including Enterprise and Information Architects, Implementers.*

This clause provides an overview of the ontology architecture and modeling strategies used to develop the Foundations ontology.

- Usage and restriction of the Ontology Definition Metamodel (ODM) standard
- Notional architecture and intended use of the Foundations ontologies
- Application and adaptation of semantic modeling techniques and notations for business presentation.

The technical content, including diagrams, incorporated in Clause 10 of this specification, was generated from the same models used to generate the RDF/XML serialized OWL, further ensuring correctness and completeness of the specification itself.

8.1 Ontology Definition Metamodel (ODM) Usage and Adaptations

The model content is developed and maintained using the Unified Modeling Language as a modeling tool framework, but with all model content built using the formal constructs of the Web Ontology Language (OWL). This is achieved using the OMG's Ontology Definition Metamodel (ODM) specification.

The Ontology Definition Metamodel (ODM) specification provides a means to represent OWL constructs using UML tools. This is achieved using UML's extension capability called 'profiles' for OWL and for RDF Schema. The ODM UML Profiles define a number of stereotypes which apply to standard UML metaclasses and may be used to represent OWL constructs in a consistent and meaningful way.

8.2 Ontology Architecture and Namespaces

The ontology architecture for FIBO is designed to facilitate reuse and ontology evolution to the degree possible. It is also designed to facilitate mapping to other standards, in particular, to financial industry domain standards, such as FpML (Financial Products Mark-up Language[2]). There are countless standards used for financial reporting, many of which are complex and lengthy, with overlap and jurisdiction-specific semantics. An approach to the foundational terminology that provides very high-level, abstract conceptual knowledge designed to facilitate mapping is an important design goal of FIBO Foundations.

Proxy concepts for Goal, Objective, Address, and Country, for example, that are included in the Foundations with little embellishment, are designed to provide hooks for mapping to the OMG's Business Motivation Model, ISO standards for Country code representations, US Publication 28 and other national postal addressing standards, and so forth. The basic building blocks for the Foundations Ontology are shown in Figure 8.1, below.

As shown in the diagram, the Foundations ontologies are divided up into a number of *modules*. For example, the Utilities module includes: a general purpose BusinessTypes.owl ontology, a general Relations.owl ontology, and an AnnotationVocabulary.owl ontology, that captures FIBO-specific annotations.

The Foundations modules will ultimately depend on (1) Basic Terminology and Ontology Metadata (in light gray in the figure), and (2) a number of external modules, representing concepts for Natural Language, Geopolitical Entities (for example ISO 3166 Country codes, regional and municipal designations), Postal Addressing (from standards such as US Publication 28), and concepts defining dates, times, calendars, and schedules. A sample set of these anticipated external resources are given in the dark gray layer in the figure.

In this initial version, the Foundations standard reuses metadata definitions, as highlighted in Figure 8.1 in the Basic Terminology and Ontology Metadata layer, from:

[2] See http://www.fpml.org/.

- The Dublin Core Metadata Terms Standard
- The W3C Simple Knowledge Organization System (SKOS)
- The OMG Architecture Board's Specification Metadata Recommendation

SKOS and the OMG Specification Metadata are explicitly imported, while the Dublin Core is not, due to the fact it is an RDF Vocabulary and only OWL ontologies may be formally imported.

Figure 8.1 Foundations Ontology Architecture

Table 8.1 lists the prefixes and namespaces considered external to FIBO.

Table 8-1. Prefix and Namespaces for referenced/external vocabularies

Namespace Prefix	Namespace
rdf	**http://www.w3.org/1999/02/22-rdf-syntax-ns#**
rdfs	http://www.w3.org/2000/01/rdf-schema#
owl	http://www.w3.org/2002/07/owl#
xsd	http://www.w3.org/2001/XMLSchema#
dct	http://purl.org/dc/terms/
skos	http://www.w3.org/2004/02/skos/core#
sm	http://www.omg.org/techprocess/ab/SpecificationMetadata/
lcc-lr	http://www.omg.org/spec/LCC/Languages/LanguageRepresentation/
lcc-639-1	http://www.omg.org/spec/LCC/Languages/ISO639-1-LanguageCodes/

Namespace Prefix	Namespace
lcc-cr	http://www.omg.org/spec/LCC/Countries/CountryRepresentation/
lcc-3166-1	http://www.omg.org/spec/LCC/Countries/ISO3166-1-CountryCodes/

The namespace approach taken for FIBO is based on OMG guidelines and is constructed as follows:

- A standard prefix `http://www.omg.org/spec/`
- The family name, EDMC-`FIBO`
- The abbreviation for the specification: in this case `FND`
- The module name
- The ontology name

Note that the URI/IRI strategy for the ontologies in FIBO takes a "slash" rather than "hash" approach, in order to accommodate server-side applications. Though not technically necessary, this specification does mandate namespace prefixes to be used. These are constructed as follows with the components separate by "-":

- The specification family name `fibo`
- The specification abbreviation: `fnd`
- An abbreviation for the module name
- An abbreviation for the ontology name

The namespaces and prefixes corresponding to FIBO Foundations ontologies are summarized in Table 8-2 for convenience. These are given in alphabetical order, by module, rather than with any intent to show imports relationships.

Table 8-2. Prefix and Namespaces for FIBO Foundations

Namespace Prefix	Namespace
fibo-fnd-acc-mod	http://www.omg.org/spec/EDMC-FIBO/FND/Accounting/AboutAccounting/
fibo-fnd-acc-aeq	http://www.omg.org/spec/EDMC-FIBO/FND/Accounting/AccountingEquity/
fibo-fnd-acc-cur	http://www.omg.org/spec/EDMC-FIBO/FND/Accounting/CurrencyAmount/
fibo-fnd-acc-4217	http://www.omg.org/spec/EDMC-FIBO/FND/Accounting/ISO4217-CurrencyCodes/
fibo-fnd-aap-mod	http://www.omg.org/spec/EDMC-FIBO/FND/AgentsAndPeople/AboutAgentsAndPeople/
fibo-fnd-aap-agt	http://www.omg.org/spec/EDMC-FIBO/FND/AgentsAndPeople/Agents/
fibo-fnd-aap-ppl	http://www.omg.org/spec/EDMC-FIBO/FND/AgentsAndPeople/People/
fibo-fnd-agr-mod	http://www.omg.org/spec/EDMC-FIBO/FND/Agreements/AboutAgreements/

Namespace Prefix	Namespace
fibo-fnd-agr-agr	http://www.omg.org/spec/EDMC-FIBO/FND/Agreements/Agreements/
fibo-fnd-agr-ctr	http://www.omg.org/spec/EDMC-FIBO/FND/Agreements/Contracts/
fibo-fnd-arr-mod	http://www.omg.org/spec/EDMC-FIBO/FND/Arrangements/AboutArrangements/
fibo-fnd-arr-arr	http://www.omg.org/spec/EDMC-FIBO/FND/Arrangements/Arrangements
fibo-fnd-arr-cls	http://www.omg.org/spec/EDMC-FIBO/FND/Arrangements/ClassificationSchemes/
fibo-fnd-arr-cd	http://www.omg.org/spec/EDMC-FIBO/FND/Arrangements/Codes
fibo-fnd-arr-doc	http://www.omg.org/spec/EDMC-FIBO/FND/Arrangements/Documents
fibo-fnd-arr-id	http://www.omg.org/spec/EDMC-FIBO/FND/Arrangements/IdentifiersAndIndices
fibo-fnd-dt-mod	http://www.omg.org/spec/EDMC-FIBO/FND/DatesAndTimes/AboutDatesAndTimes/
fibo-fnd-dt-bd	http://www.omg.org/spec/EDMC-FIBO/FND/DatesAndTimes/BusinessDates
fibo-fnd-dt-fd	http://www.omg.org/spec/EDMC-FIBO/FND/DatesAndTimes/FinancialDates
fibo-fnd-dt-oc	http://www.omg.org/spec/EDMC-FIBO/FND/DatesAndTimes/Occurrences
fibo-fnd-gao-mod	http://www.omg.org/spec/EDMC-FIBO/FND/GoalsAndObjectives/AboutGoalsAndObjectives/
fibo-fnd-gao-gl	http://www.omg.org/spec/EDMC-FIBO/FND/GoalsAndObjectives/Goals/
fibo-fnd-gao-obj	http://www.omg.org/spec/EDMC-FIBO/FND/GoalsAndObjectives/Objectives/
fibo-fnd-law-mod	http://www.omg.org/spec/EDMC-FIBO/FND/Law/AboutLaw/
fibo-fnd-law-jur	http://www.omg.org/spec/EDMC-FIBO/FND/Law/Jurisdiction/
fibo-fnd-law-lcap	http://www.omg.org/spec/EDMC-FIBO/FND/Law/LegalCapacity/
fibo-fnd-law-cor	http://www.omg.org/spec/EDMC-FIBO/FND/Law/LegalCore/
fibo-fnd-org-mod	http://www.omg.org/spec/EDMC-FIBO/FND/Organizations/AboutOrganizations/
fibo-fnd-org-fm	http://www.omg.org/spec/EDMC-FIBO/FND/Organizations/FormalOrganizations/

Namespace Prefix	Namespace
fibo-fnd-org-lg	http://www.omg.org/spec/EDMC-FIBO/FND/Organizations/LegitimateOrganizations/
fibo-fnd-org-org	http://www.omg.org/spec/EDMC-FIBO/FND/Organizations/Organizations/
fibo-fnd-oac-mod	http://www.omg.org/spec/EDMC-FIBO/FND/OwnershipAndControl/AboutOwnershipAndControl/
fibo-fnd-oac-ctl	http://www.omg.org/spec/EDMC-FIBO/FND/OwnershipAndControl/Control/
fibo-fnd-oac-oac	http://www.omg.org/spec/EDMC-FIBO/FND/OwnershipAndControl/OwnershipAndControl
fibo-fnd-oac-own	http://www.omg.org/spec/EDMC-FIBO/FND/OwnershipAndControl/Ownership/
fibo-fnd-pty-mod	http://www.omg.org/spec/EDMC-FIBO/FND/Parties/AboutParties/
fibo-fnd-pty-pty	http://www.omg.org/spec/EDMC-FIBO/FND/Parties/Parties/
fibo-fnd-pty-rl	http://www.omg.org/spec/EDMC-FIBO/FND/Parties/Roles/
fibo-fnd-plc-mod	http://www.omg.org/spec/EDMC-FIBO/FND/Places/AboutPlaces/
fibo-fnd-plc-adr	http://www.omg.org/spec/EDMC-FIBO/FND/Places/Addresses/
fibo-fnd-plc-cty	http://www.omg.org/spec/EDMC-FIBO/FND/Places/Countries/
fibo-fnd-plc-fac	http://www.omg.org/spec/EDMC-FIBO/FND/Places/Facilities
fibo-fnd-plc-loc	http://www.omg.org/spec/EDMC-FIBO/FND/Places/Locations/
fibo-fnd-plc-vrt	http://www.omg.org/spec/EDMC-FIBO/FND/Places/VirtualPlaces
fibo-fnd-pas-mod	http://www.omg.org/spec/EDMC-FIBO/FND/ProductsAndServices/AboutProductsAndServices/
fibo-fnd-pas-psch	http://www.omg.org/spec/EDMC-FIBO/FND/ProductsAndServices/PaymentsAndSchedules/
fibo-fnd-pas-pas	http://www.omg.org/spec/EDMC-FIBO/FND/ProductsAndServices/ProductsAndServices/
fibo-fnd-qt-mod	http://www.omg.org/spec/EDMC-FIBO/FND/Quantities/AboutQuantities/
fibo-fnd-qt-qtu	http://www.omg.org/spec/EDMC-FIBO/FND/Quantities/QuantitiesAndUnits/
fibo-fnd-rel-mod	http://www.omg.org/spec/EDMC-FIBO/FND/Relations/AboutRelations/

Namespace Prefix	Namespace
fibo-fnd-rel-rel	http://www.omg.org/spec/EDMC-FIBO/FND/Relations/Relations/
fibo-fnd-utl-mod	http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/AboutUtilities/
fibo-fnd-utl-alx	http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/Analytics
fibo-fnd-utl-av	http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/AnnotationVocabulary/
fibo-fnd-utl-bt	http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/BusinessFacingTypes/

8.3 FIBO-Based Reporting

8.3.1 Business-Facing Approach

There are a number of ways of presenting the ontology to domain experts, and the intent is to standardize two of these.

Diagrammatic Presentation

The FIBO ontologies (model) may be presented to business domain experts in a number of forms, with views that express different levels of detail and different aspects of the model to aid in understanding. Critical requirements for business-facing diagrams include limiting or eliminating technical detail while retaining it in the underlying model, and hiding, to the degree possible:

- stereotype names on diagrams, although English labels and icons may be used where important to express the meaning of a line or box,
- technical tags, such as visibility, and optionally names, on property endpoints,
- empty partitions in boxes representing classes and association classes, and
- the class in an association class representation of an object, data, or annotation property.

This does not preclude the incorporation of diagramming elements to represent fundamental concepts from set theory, first order logic, etc., that are needed to understand the ontology. Other requirements for diagramming style will be forthcoming as the specification achieves broader adoptions.

An example, showing a simplified OWL diagram, is given in Figure 8.2.

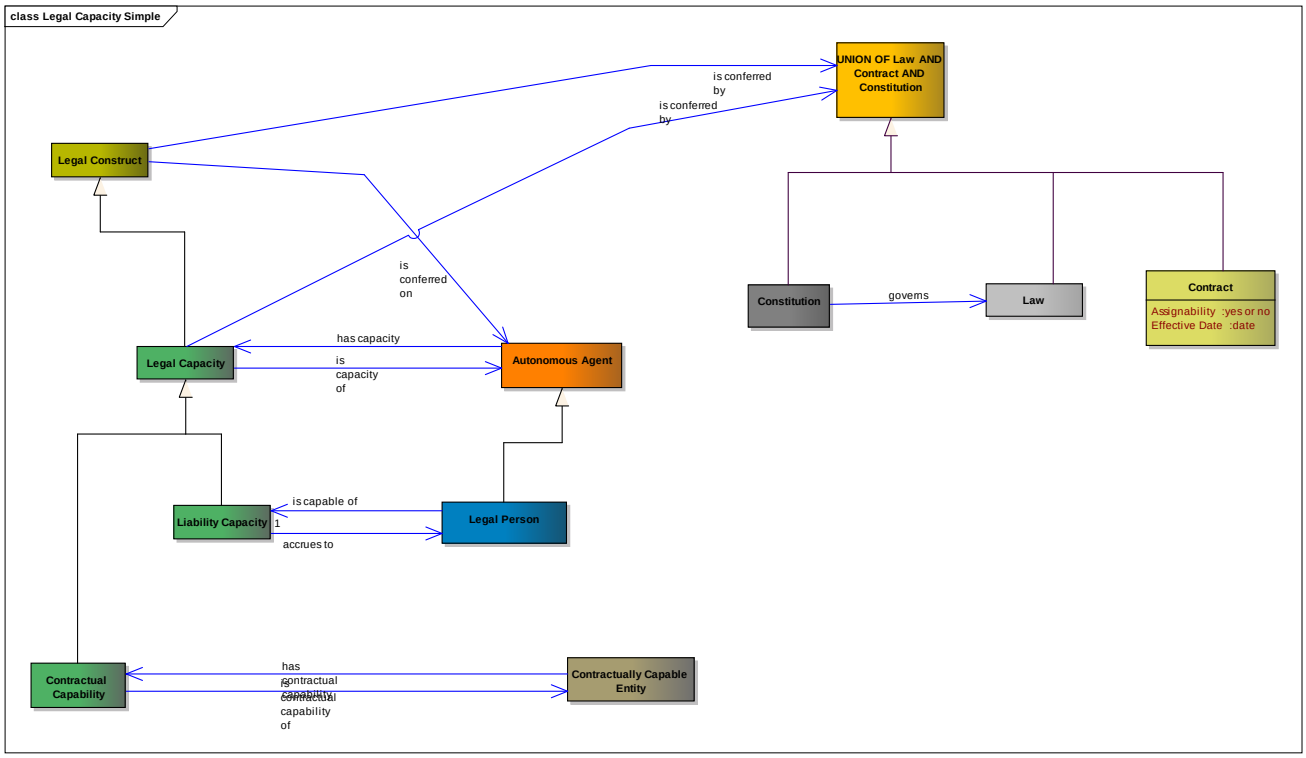

Figure 8.2 Example Business-Facing FIBO Diagram

The strategy for representation for subject matter experts may include use of color to highlight certain lines, in addition to labeling them in English, for example, by using blue lines for object properties, green lines for data properties (if they are not shown using an attribute style, inside the class box), dashed red dependency for disjointness, and so forth.

Tabular or Textual Presentation

In addition to the presentation via diagrams, there is a need to provide business domain experts with a more spreadsheet-like view of the terms, relationships, formal definitions, and other annotations in particular, for review, understanding, and use.

There are two levels of detail that shall be made available in reports. These are the 'Basic' view of Term, Definition and Synonym, and an extended view giving most or all of the same information that is seen in the diagrams. This shall include line entries for each thing and each fact (Relationship Property and Simple Property) as well as the set theory constructs and relationships modeled (unions, parent terms etc.). It is not necessary to show relationships between relationships in these tables, such as sub property hierarchies or property inverses.

The constructs shall be represented with an English language name, including spaces between words rather than camel case; those that are substantially different from their OWL language equivalents include: “Is A” for subclass relationships, “Type” for datatypes, “type of” rather than rdfType, “Simple Property” for datatype properties, “Relationship Property” for object properties, and “mutually exclusive” for disjointness relationships. These names are in US English and may be replaced in reports with definitionally equivalent labels in other natural languages and possibly additional terms that may be added to support parallel, collaborative development processes required for FIBO financial product-specific ontologies.

9 Additional Metadata

9.1 Introduction

As discussed in Clause 8, the FIBO Foundations and specifications that depend on it reuse existing metadata standards, including:

- The Dublin Core Metadata Terms Standard
- The W3C Simple Knowledge Organization System (SKOS)
- The OMG Architecture Board's Specification Metadata Recommendation

These metadata definitions are not inherent elements of RDF Schema or OWL, although the standard makes extensive use of `rdfs:label` in particular. This clause of the specification describes the metadata used throughout the standard and provides examples where appropriate for clarification purposes.

9.2 Family and Specification Metadata

The Foundations specification has a set of common metadata which is specified in this sub clause and is given in the specification and version 'About' ontologies as described in the sub-clause on the About ontologies. This information is included regardless of whether the About ontologies are serialized as RDF/XML OWL, UML/XMI with the ODM profiles for RDF and OWL applied, or as ODM XMI.

The use of the "sm" namespace prefix in the abbreviated IRI for the metadata term refers to the Specification Metadata ontology, as described in Table 8-1, above.

Table 9-1. FIBO Foundations Specification Family Metadata

Metadata Term	Value
sm:familyTitle	Financial Industry Business Ontology (FIBO)
sm:familyAbbreviation	FIBO
sm:familyURL	http://www.omg.org/spec/EDMC-FIBO/
sm:familyAbstract	The content that comprises the Financial Industry Business Ontology (FIBO) is documentation, interpretable in formal logic, of the concepts represented by finance industry terms as used in official financial organization documents such as contracts, product/service specifications and governance and regulatory compliance documents.
sm:technologyArea	**formal semantics**
sm:topicArea	**finance**
sm:keyword	**Financial Industry Business Ontology, FIBO, ontology, vocabulary**

Table 9-2. FIBO Foundations Specification Metadata

Metadata Term	Value
sm:specificationTitle	Financial Industry Business Ontology (FIBO) Foundations Specification
sm:specificationAbbreviation	FIBO-FND
sm:specificationURL	http://www.omg.org/spec/EDMC-FIBO/FND/
sm:specificationAbstract	FIBO Foundations is a set of business concepts which are intended to support the financial industry terms semantics presented in other FIBO specifications. The FIBO Foundations models define concepts which are not unique to the financial services industry. From these, financial industry terms in other FIBO specifications may be derived by extension. Terms are also included which may be referred to by properties of things in those specifications. FIBO Foundations therefore includes a number of basic terms about legal, contractual and organizational concepts, among others.
sm:dependsOn	http://www.omg.org/techprocess/ab/SpecificationMetadata/
sm:keyword	Foundational vocabulary

Table 9-3. FIBO Foundations Specification Version Metadata

Metadata Term	Value
sm:thisVersion	1.0
sm:publicationDate	2013-08-26T18:00:00
sm:specificationVersionURL	http://www.omg.org/spec/EDMC-FIBO/FND/1.0/
sm:specificationVersionStatus	Request For Comments (RFC)
skos:historyNote	This version of the FIBO Foundations Specification was revised primarily to reflect comments received at the March 2013 OMG Technical Meeting in Reston and reflected in the Errata discussed at the June 2013 OMG Tech-

Metadata Term	Value
	nical Meeting in Berlin. Revisions to FIBO Foundations are managed per the process outlined in the Policies and Procedures for OMG standards, with the intent to maintain backwards compatibility in the ontologies to the degree possible. The RDF/XML serialized OWL for the Foundations ODM/OWL ontologies have been checked for syntactic errors and logical consistency with Protege 4 (http://protege.stanford.edu/), HermiT 1.3.7 (http://www.hermit-reasoner.com/) and Pellet 2.2 (http://clarkparsia.com/pellet/).
sm:addressForComments	http://www.omg.org/issues/

9.3 Module Metadata

Every module will have unique metadata specific to that module, as given in Clause 10, below. Additionally, every ontology will include curation metadata. Explicit use of the MIT License[3] for software (including OWL ontologies, UML models, ODM XMI) is intended to assure users of the ontologies that the ontologies are freely available, for use with attribution, and without warranty. This module metadata is given in the module "About" ontology as described in the sub-clause on the About ontologies.

9.4 Ontology-Level Metadata

Each Foundations ontology has a set of curation and rights metadata which is specified in this sub clause rather than being repeated for each ontology. This information is included regardless of whether the ontology is serialized as RDF/XML OWL, UML/XMI with the ODM profiles for RDF and OWL applied, or as ODM XMI.

Table 9-4. FIBO Foundations Specification Curation and Rights Metadata

Metadata Term	Value
sm:copyright	Copyright (c) 2013-2014 EDM Council, Inc. Copyright (c) 2013-2014 Object Management Group, Inc.
dct:license	http://www.omg.org/techprocess/ab/SpecificationMetadata/MITLicense
sm:responsibleTaskForce	http://fdtf.omg.org/

Finally, each ontology will include ontology-specific metadata, using the OMG Specification Metadata ontology. These details are provided with the individual ontologies in Clause 10.

[3] See http://opensource.org/licenses/mit-license.php

9.5 The ‘About’ Files

Each FIBO submission is to be accompanied by a set of files containing the metadata for the FIBO Family, the individual FIBO specification and the specifications for each module. This is so that metadata for each of these elements of a FIBO submission do not need to be repeated in each ontology. These are collectively known as “About files”

The About files are:

- About the EDMC-FIBO Family
- About the Specification
- About the specification version
- About each Module

9.5.1 EDM Council FIBO Family About File

This is to be included with each FIBO specification submission. It has the filename AboutTheEDMC-FIBOFamily.rdf and includes the family level metadata described in sub-clause 9.2.

9.5.2 Specification About File

This is unique to each FIBO specification and is named according to the specification 3- or 4-letter abbreviation, for example (for Foundations) AboutFND.rdf

This file includes the specification-level metadata described in sub-clause 9.2.

9.5.3 Specification Version About File

This is unique to each FIBO specification and is named according to the specification 3- or 4-letter abbreviation and its version, for example (for Foundations) AboutFND-1.0.rdf

This file includes the specification version metadata described in sub-clause 9.2.

9.5.4 Module About File

This is unique to each FIBO module and is named according to the module full name, for example (for Foundations): AboutAgreements.rdf

This file includes the module-level metadata described in sub-clause 9.3.

9.6 Ontology Entity-Level Metadata

This sub clause describes the metadata that are applied to each named concept (Class and Property) in the ontologies.

9.6.1 Definitions, Notes, and Labels

Table 9-5. Definitions, Labeling, and Notes

Term Requirement	Term Type	Annotation	Usage Notes
Definition	Definition	`skos:definition`	Main formal definition of term. Must always be present
Change history	Note	`skos:changeNote`	Notes indicating why something was modified
General note, editorial comment	Note	`skos:editorialNote`	The bulk of the “Further Notes” narrative is expressed this way

Examples	Note	skos:example	Examples
Explanatory note	Note	fibo-utl-av:explanatoryNote	Notes providing additional explanation about the concept
Historical note	Note	skos:historyNote	Notes regarding the history of the concept
Note	Note	skos:note	Used when no specific note annotation is appropriate
Scope note	Note	skos:scopeNote	Clarifying information about the scope of the term or concept
Usage note	Note	fibo-utl-av:usageNote	Used to suggest how a particular concept is intended to be used
Preferred Label	Label	skos:prefLabel	Replaces rdfs:label if there is a preferred label for the concept
Alternate Label	Label	skos:altLabel	Alternate label additional to prefLabel. Should be used instead of rdfs:label for alternatives

9.6.2 Synonymous Terms

Synonyms are fundamental to the reporting required for business domain view and review of the ontologies, which, at a basic level, may only require the concept, a label, its formal definition in text form, and any synonyms.

Fundamentally, an ontology, and any extensions derived from it, should contain only a single element defining a given concept, with synonyms captured using the `fibo-utl-av:synonym` annotation property. Within a given ontology, use of separate classes with the same meaning, together with the OWL construct for class equivalence (equivalentClass) is not considered best practice. Such an approach may be necessary to align or map ontologies to one another, however, where the same concepts exist in different namespaces. `fibo-utl-av:abbreviation` may be used to specify abbreviations and acronyms associated with concepts as appropriate.

9.6.3 Provenance and Cross-reference Annotation

Where possible, every effort is made in the FIBO ontologies to provide references for the origin of terms and their definitions, including cases where those definitions have been adapted for FIBO usage. While less important for Foundations, any FIBO ontology that includes terminology from a particular standard, such as FpML, ISO 20022, any regulatory publication, and so forth should note it as the source for a given concept or its definition.

Four annotation properties are provided in the FIBO AnnotationVocabulary to facilitate provenance documentation for the terminology and definitions specified in the standard. These are:

- `fibo-utl-av:adaptedFrom` – used where the text in the `skos:definition` is adapted from the definition of the term defined in the range of this property (range can be a string, URI, or BibliographicCitation). Note that this initial version of Foundations does not recommend a specific standard for citatations. There are a number of ontologies that might be considered for this purpose, and the OMG Specification Metadata provides a class called BibliographicCitation that can be used as the range of this annotation and can be mapped to the preferred citation definition for a given application, organization, or repository.
- `fibo-utl-av:definitionOrigin` – used where the text in the `skos:definition` is a direct copy of the definition of the term defined in the range of this property (range can be a string, URI, or BibliographicCitation).
- `fibo-utl-av:termOrigin` – which provides the means to document the source of a term, in a standard, in some other document, or by some organization. The range of this property is the document and / or organization from which the term was derived (range can be a string, URI, or BibliographicCitation).

- `fibo-utl-av:nameOrigin` – which provides the means to document the name of the original term in the standard, other document or organization referenced via the annotation `fibo-utl-av:termOrigin`

9.6.4 Change Management Annotation

In addition to the version information provided at the specification level for a given FIBO ontology, additional annotations for change management purposes may be appropriate at the concept level. These may include:

- `skos:changeNote`
- `fibo-utl-av:modifiedBy` – identifying the person and/or organization responsible for the change
- `fibo-utl-av:modifiedOn` – identifying the date and time of the change

10. Model Content Reports

Intended Audience: *Business Analysts, other business stakeholders*

This clause shows the content of the model from a business perspective. Model content is presented both as diagrams and as tables. Readers do not need to be conversant with the Web Ontology Language or other modeling languages in order to be able to interpret what is presented here. However some familiarity with the “set theoretic” interpretation of the model content is required.

This clause has a sub clause for each ontology that is automatically generated from the ODM representation of that ontology, and is designed to be more human-readable than the raw OWL file.

10.1 Module: Utilities

Table 10-1. Utilities Module Metadata

Metadata Term	Value
sm:moduleName	Utilities
sm:moduleAbbreviation	FIBO-FND-UTL
sm:moduleVersion	1.0
sm:moduleAbstract	Ontologies which provide annotations and business facing datatypes to be used in other ontologies. These ontologies are not expected to be used directly by business stakeholders and are for the definition of material which is used by semantic modelers in Foundations and in other FIBO ontologies.

10.1.1 Ontology: Annotation Vocabulary

This vocabulary provides a set of metadata annotations for use in describing FIBO ontology elements. The annotations extend properties defined in the OMG's Specification Metadata Recommendation, in the Dublin Core Metadata Terms Vocabulary and in the W3C Simple Knowledge Organization System (SKOS) Vocabulary, and have been customized to suit the FIBO specification development process.

Note that any of the original properties provided in Dublin Core and SKOS can be used in addition to the terms provided herein. However, any Dublin Core terms that are not explicitly defined as OWL annotation properties in this ontology or in any of its imports must be so declared in the ontologies that use them.

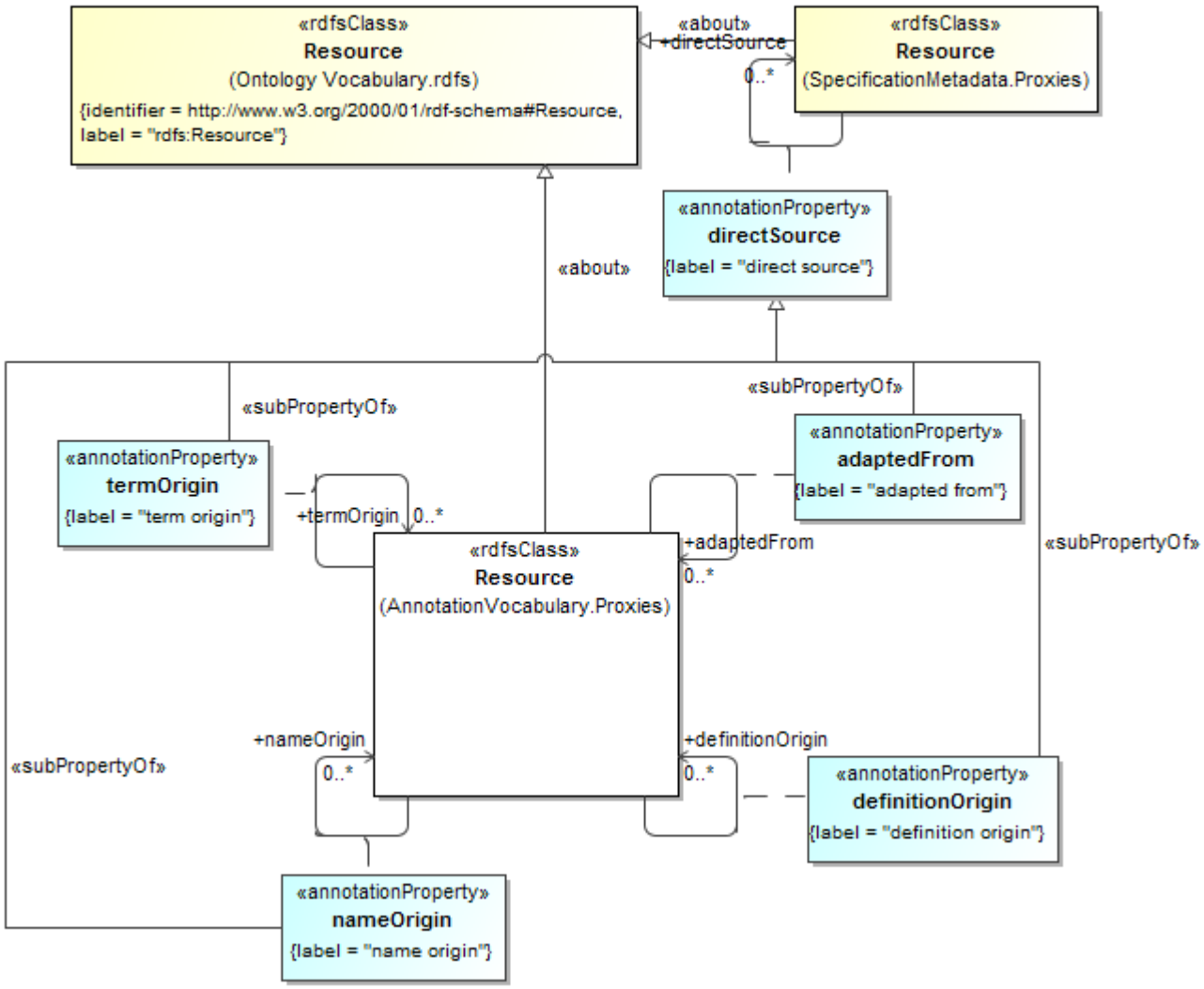

Figure 10.1 Term and Definition Annotations

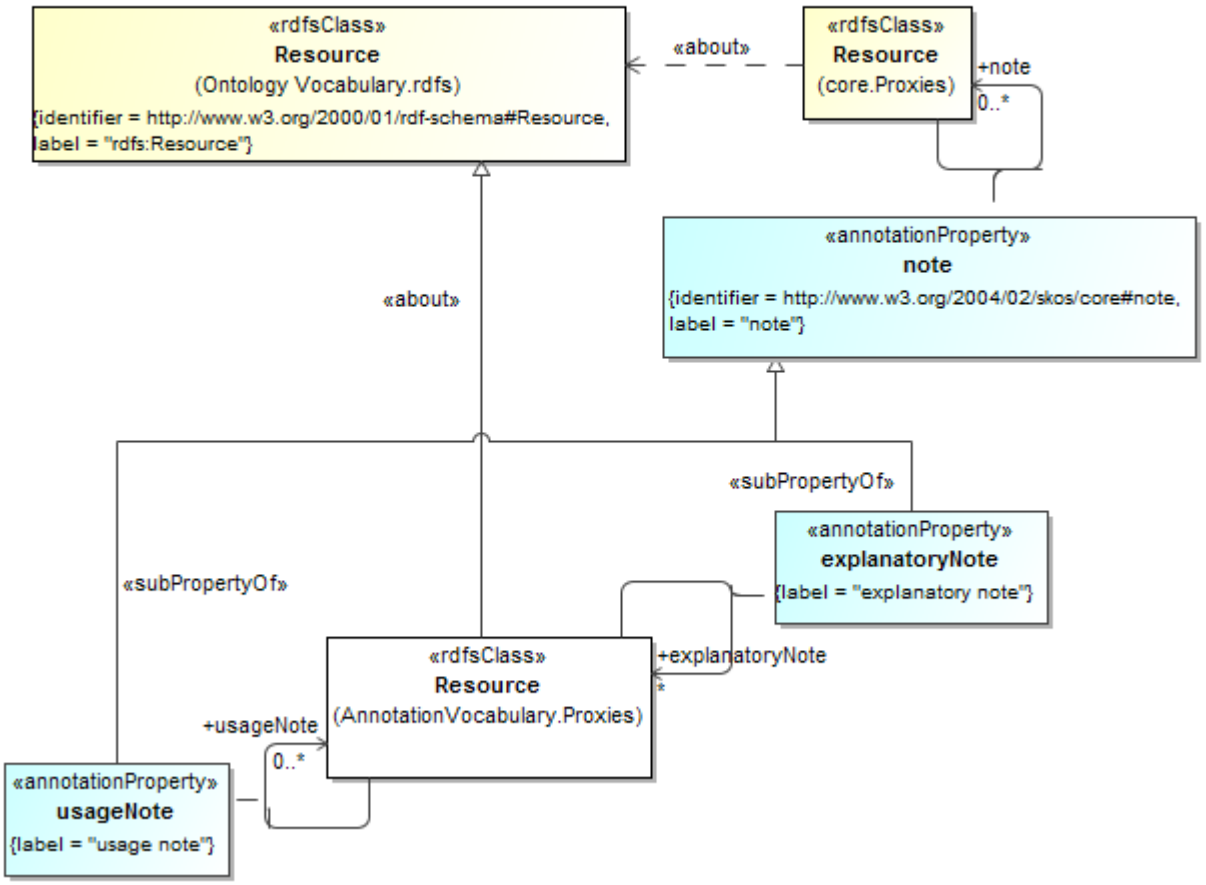

Figure 10.2 Explanatory Annotations

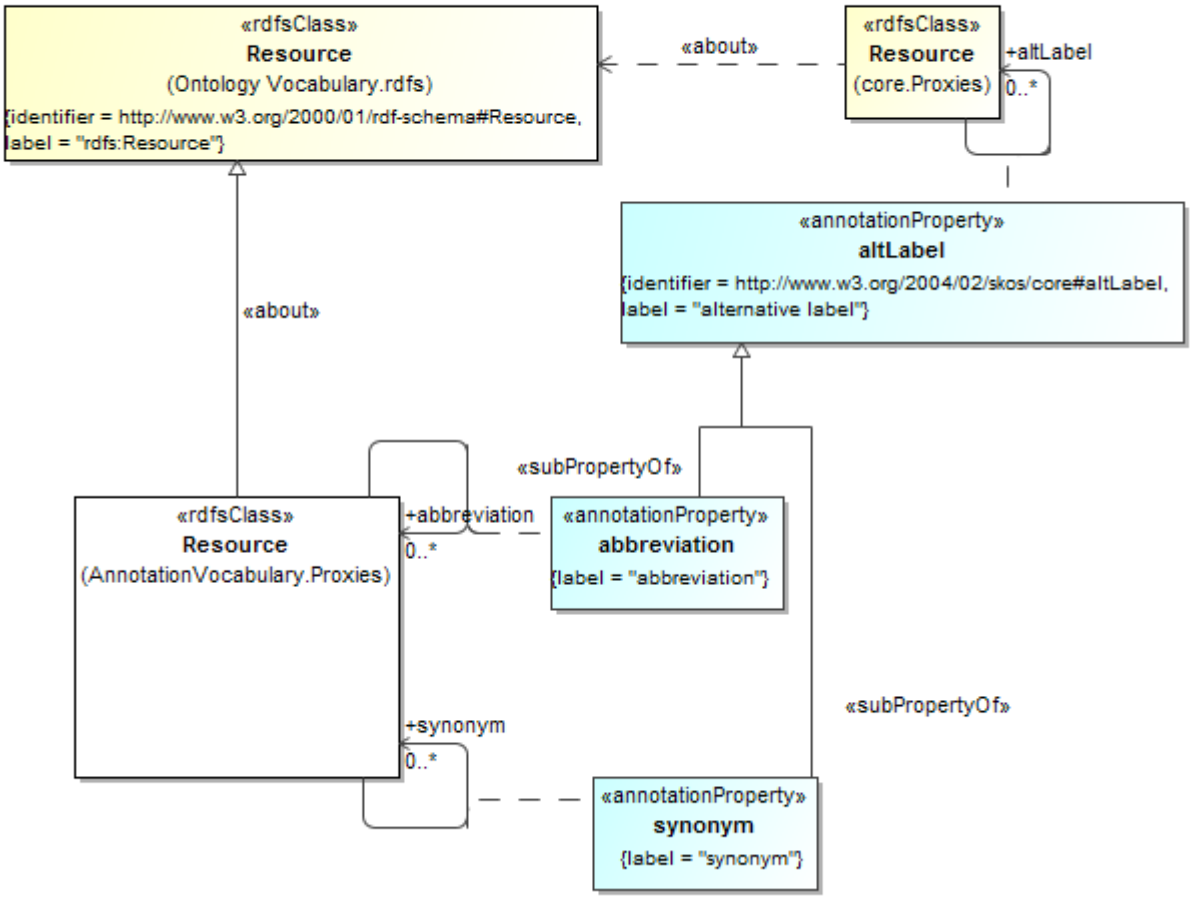

Figure 10.3 Alternate Label Annotations

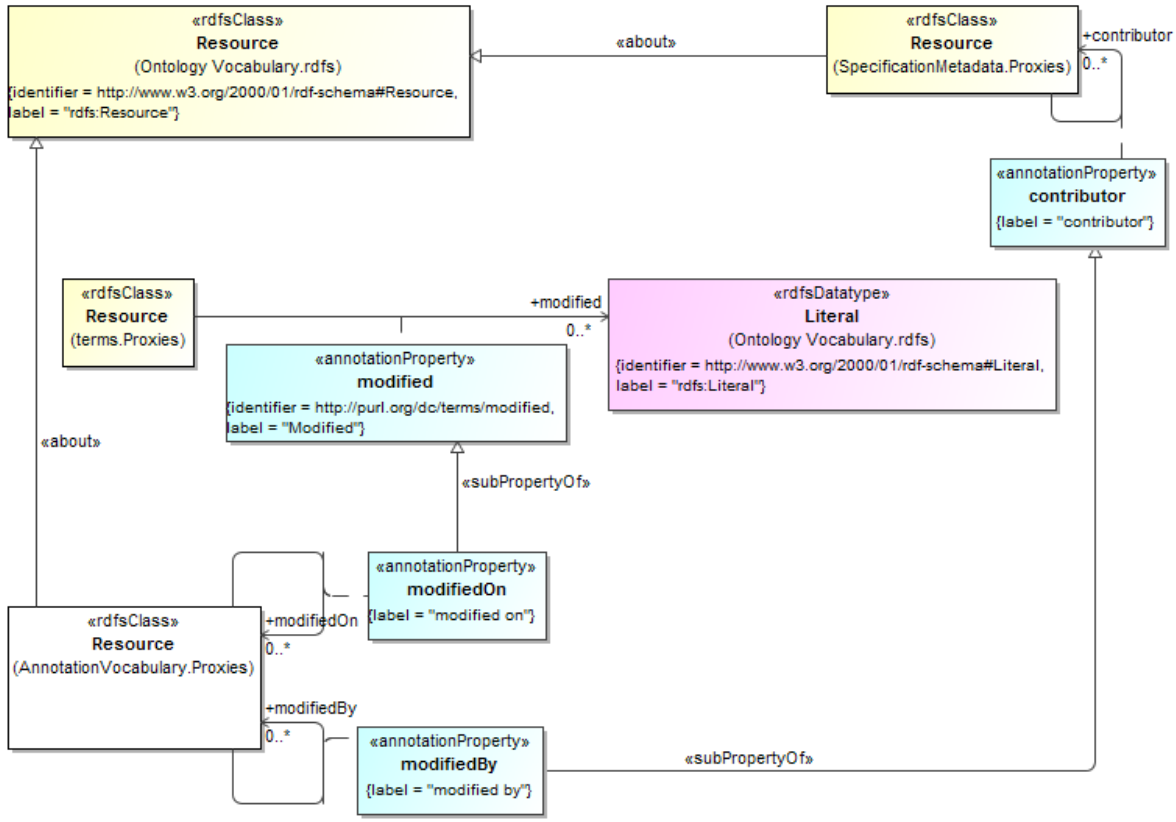

Figure 10.4 Element Change Management Annotations

Table 10-2. Annotation Vocabulary Metadata

Metadata Term	Value
sm:filename	Annotation Vocabulary
sm:fileAbbreviation	fibo-fnd-utl-av
OntologyIRI	http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/AnnotationVocabulary/
owl:versionIRI	http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Utilities/AnnotationVocabulary/

Table 10-3. Annotation Vocabulary Details

Annotation Properties

Name	Annotations	Property Axioms
termOrigin (term origin)	Definition: Document from which a given term was taken directly; the range for this annotation can be a string (either xsd:string or fibo-fnd-utl-bt:text), URI (either xsd:anyURI or fibo-fnd-utl-bt:uri), or Bibli-	Parent Property: sm:directSource

Name	Annotations	Property Axioms
	ographicCitation	
nameOrigin (name origin)	Definition: provides the means to document the name of the original term in the source referenced via termOrigin; the range for this annotation can be a string (either xsd:string or fibo-fnd-utl-bt:text), URI (either xsd:anyURI or fibo-fnd-utl-bt:uri), or BibliographicCitation	Parent Property: sm:directSource
definitionOrigin (definition origin)	Definition: document from which a given definition was taken directly; the range for this annotation can be a string (either xsd:string or fibo-fnd-utl-bt:text), URI (either xsd:anyURI or fibo-fnd-utl-bt:uri), or BibliographicCitation	Parent Property: sm:directSource
adaptedFrom (adapted from)	Definition: the document from which a given term (or its definition) was adapted; the range for this annotation can be a string (either xsd:string or fibo-fnd-utl-bt:text), URI (either xsd:anyURI or fibo-fnd-utl-bt:uri), or BibliographicCitation	Parent Property: sm:directSource
ModifiedOn (modified on)	Definition: identifies the date a model element in the body of an ontology was changed	Parent Property: terms:modified
modifiedBy (modified by)	Definition: identifies the organization or person responsible for making a change to a model element in the body of an ontology	Parent Property: sm:contributor
abbreviation (abbreviation)	Definition: an abbreviation is short form for a particular designation that can be substituted for the primary representation. Adapted from: ISO 1087-1 Terminology work – Vocabulary	Parent Property: core:altLabel
synonym (synonym)	Definition: a synonym is another designation that can be substituted for the primary representation. It is a designation for the same concept. Adapted from: ISO 1087-1 Terminology work - Vocabulary	Parent Property: core:altLabel
explanatoryNote (explanatory note)	Definition: a note that provides additional explanatory information about a given concept	Parent Property: core:note
usageNote (usage note)	Definition: a note that provides information about how a given concept is used in the FIBO context	Parent Property: core:note

10.1.2 Ontology: Business Facing Types

This ontology provides high level definitions for business facing datatypes for use in other FIBO ontology elements. These types are essentially aliases of existing RDF datatypes, and are provided in order to be able to present datatype properties to a business audience with non technical names, for example yes or no in place of boolean and text in place of string. All datatype properties in the FIBO ontologies are framed in terms of these business-facing types and not in terms of the underlying technically-named datatypes.

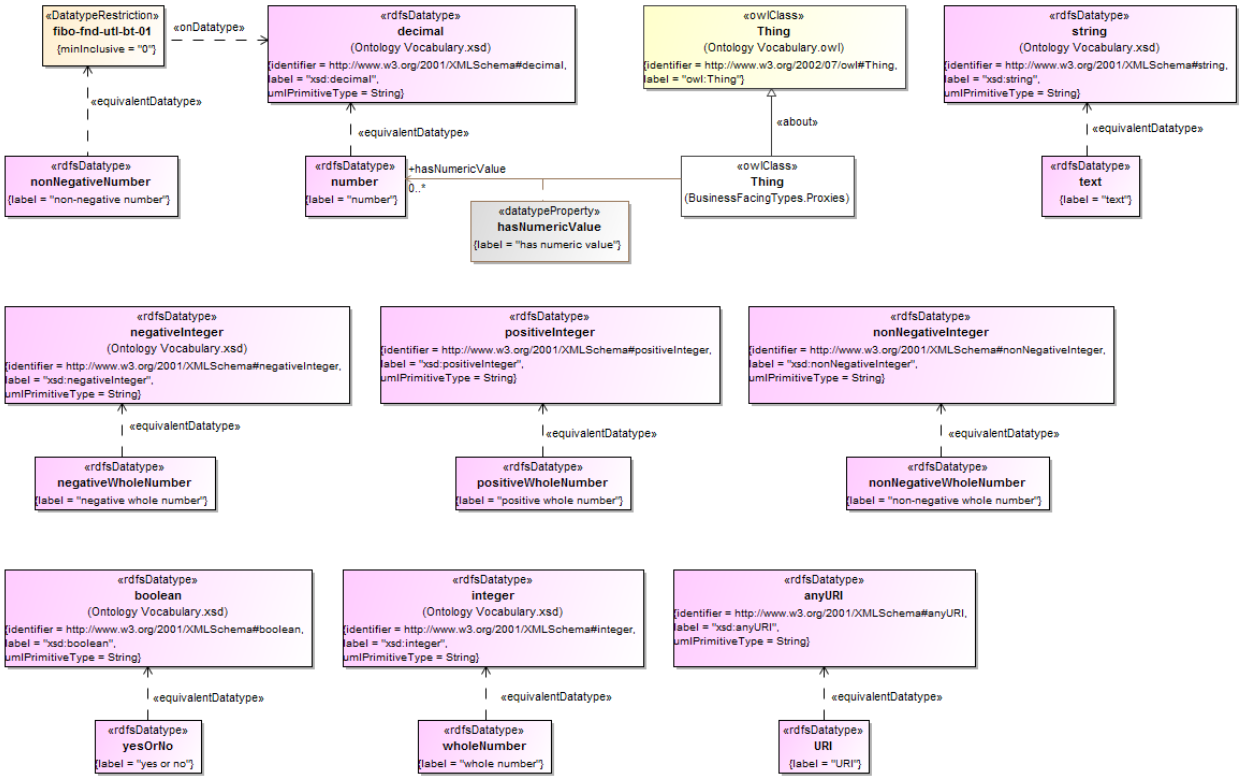

Figure 10.5 General Purpose Business Types

As shown in Figure 10.5, a number of business types are defined for use in other definitions in FIBO. These are provided to facilitate understanding by business subject matter experts, using common language rather than technical nomenclature. Note that a non-negative number is defined using a datatype restriction, fibo-fnd-utl-bt-01, that can be interpreted as saying that a non-negative number is declared to be of type xsd:decimal whose values must be greater than or equal to zero.

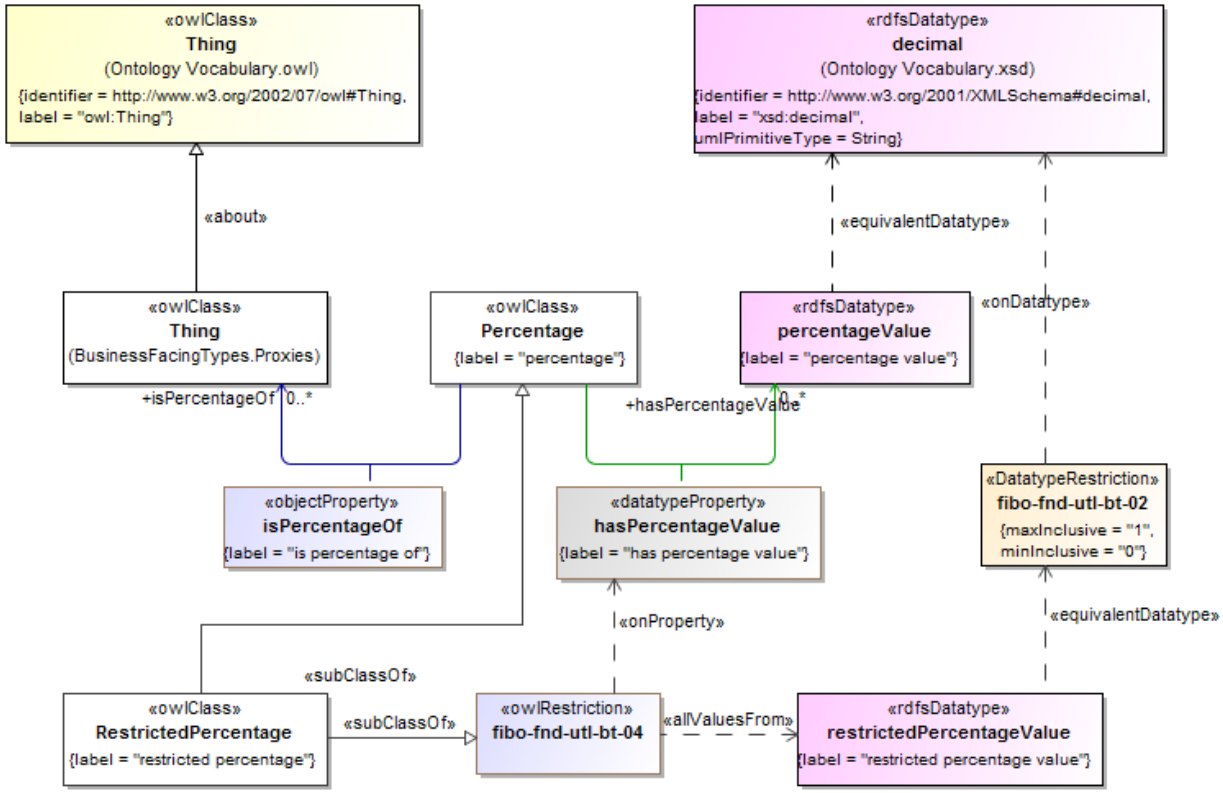

Figure 10.6 Percentage Definitions

Figure 10.6 depicts the definition of two datatypes that can be used in other FIBO definitions for the purposes of specifying percentage values. These include (1) percentage, which is declared as an xsd:decimal (and can be negative), and (2) restricted percentage, which must be an xsd:decimal whose values are restricted (in fibo-fnd-utl-bt-02) to range from zero to one.

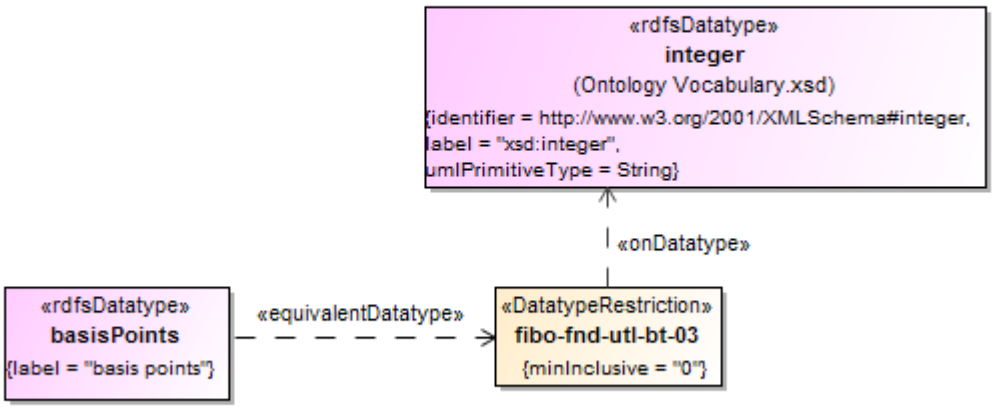

Figure 10.7 Basis Points Definition

Figure 10.7 declares the definition of basis points to be an xsd:integer whose values must be greater than or equal to zero (fibo-fnd-utl-bt-03).

Table 10-4. Business-Facing Types Ontology Metadata

Metadata Term	Value
sm:filename	Business Facing Types Ontology
sm:fileAbbreviation	fibo-fnd-utl-bt
OntologyIRI	http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/BusinessFacingTypes/
owl:versionIRI	http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Utilities/BusinessFacingTypes/
sm:dependsOn	http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/AnnotationVocabulary/

Table 10-5. Business Facing Types Definitions

Datatype	Definition	Equivalent Datatype	Concept Type	Definition Source
basisPoints	A basis point is a unit equal to one hundredth of a percentage point, or one part per ten thousand, 1/10000.	fibo-fnd-utl-bt-03	Datatype	
negativeWholeNumber		negativeInteger	Datatype	
nonNegativeNumber		fibo-fnd-utl-bt-01	Datatype	
nonNegativeWholeNumber		nonNegativeInteger	Datatype	
number	A number is a mathematical object used to count, label, and measure.	decimal	Datatype	
percentageValue	the value of a percentage expressed as a number or ratio as a fraction of 100	decimal	Datatype	
positiveWholeNumber		positiveInteger	Datatype	
restrictedPercentageValue	A type defining a percentage specified as decimal from 0 to 1. A percentage of 5% would be represented as 0.05. The maximum value is 100%, i.e., 1.	fibo-fnd-utl-bt-02	Datatype	
text	the contents of an ordinary sequential file readable as textual material without much processing	String	Datatype	
URI	a uniform resource identifier (URI) is a string of characters used to identify a name or a web resource.	anyURI	Datatype	
wholeNumber		integer		
yesOrNo	something which has two states, interpreted as Yes and No in the context in which this is used	boolean		

Table 10-6. Business Facing Types Details

Classes

Name	Annotations	Class Expressions
Percentage (percentage)	Definition: a proportion of something expressed as a percentage amount	
RestrictedPercentage (Restricted percentage)	Definition: a proportion of something expressed as a percentage amount and not exceeding 100% of that of which it is the percentage	Parent Class: Percentage Property Restriction: ∀ **hasPercentageValue.RestrictedPercentageValue** (fibo-fnd-utl-bt-04)

Properties

Name	Annotations	Property Axioms
hasPercentageValue (has percentage value)	Definition: indicates a value expressed as a percentage	Domain: Percentage Range: Percentage value
isPercentageOf (is percentage of)	Definition: that of which the percentage is a ratio, expressed as a fraction of 100 where 100 represents the whole of that thing or quantity	Domain: Percentage

10.1.3 Ontology: Analytics

This ontology provides mathematical abstractions for use in other ontologies, including for example the basic components of formulae, parameters and values.

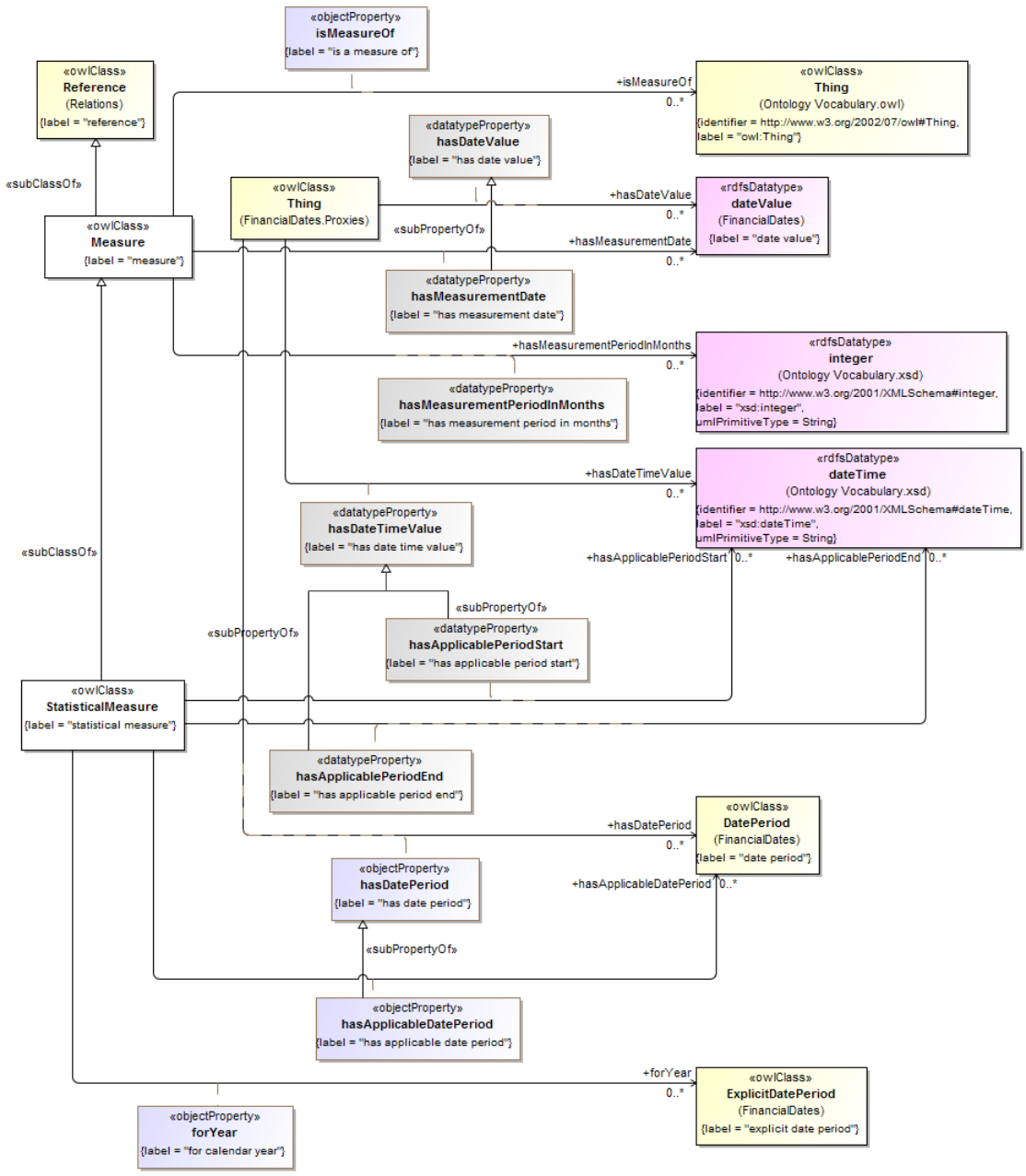

Figure 10.8 Definition of Measure and Statistical Measure

Figure 10.8 provides core definitions for concepts including measure and statistical measure, which for extension by numerous domain-specific concepts in FIBO, including but not limited to rates such as interest rates and foreign exchange rates, as well as for specifying indices and indicators, market valuation related concepts, commodities, for comparison of the performance of various instruments, and so forth.

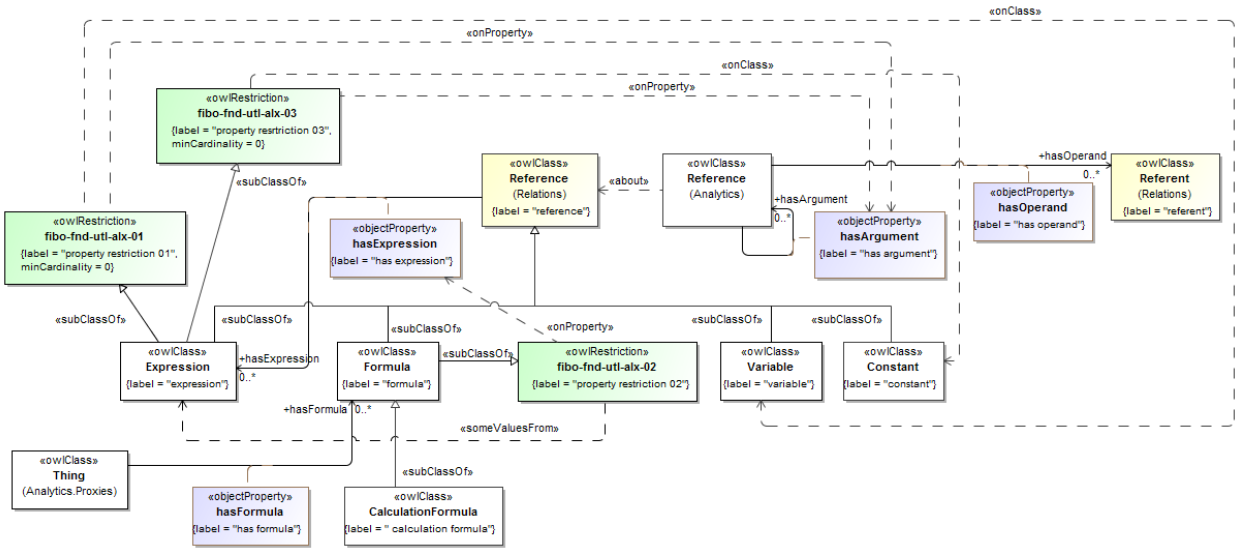

Figure 10.9 Expressions and Formulae

Figure 10.9 provides a view of the concepts needed to express various kinds of formulae, calculations, and other expressions. As was the case with measure and statistical measure, these properties are needed for extension in a number of FIBO domains, to document valuation formulas, interest rate calculations, comparisons between values for a number of concepts, etc.

Table 10-7. Analytics Ontology Metadata

Metadata Term	Value
sm:filename	Analytics
sm:fileAbbreviation	fibo-fnd-utl-alx
OntologyIRI	http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/Analytics/
owl:versionIRI	http://www.omg.org/spec/EDMC-FIBO/FND/20160201/Utilities/Analytics/
sm:dependsOn	http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/AnnotationVocabulary/ http://www.omg.org/spec/EDMC-FIBO/FND/Relations/Relations/ http://www.omg.org/spec/EDMC-FIBO/FND/DatesAndTimes/FinancialDates/

Table 10-8. Analytics Details

Classes

Name	Annotations	Class Expressions
CalculationFormula (calculation formula)	Definition: a mathematical formula that transforms one or more inputs into an amount or number of something	Parent Class: Formula

	Adapted from: http://www.oxforddictionaries.com/definition/english/calculation Adapted from: https://en.wikipedia.org/wiki/Calculation	
Constant (constant)	Definition: a symbol that represents a constant in a formula or expression	Parent Class: Reference
Expression (expression)	Definition: a finite combination of symbols that are well-formed according to applicable rules	Parent Class: Reference Property Restriction: ≥ 0 **hasArgument**.**Constant** (fibo-fnd-utl-alx-03) Property Restriction: ≥ 0 **hasArgument**.**Variable** (fibo-fnd-utl-alx-01)
Formula (formula)	Definition: a general fact or rule expressed in letters and symbols; may consist of one or more expressions Adapted from: http://www.merriam-webster.com/dictionary/formula	Parent Class: Reference Property Restriction: ∃ **hasExpression**.**Expression** (fibo-fnd-utl-alx-02)
Measure (measure)	Definition: an amount or degree of something; the dimensions, capacity, or amount of something ascertained by measuring Adapted from: http://www.merriam-webster.com/dictionary/measure	Parent Class: Reference
StatisticalMeasure (statistical measure)	Definition: A summary (means, mode, total, index, etc.) of the individual quantitative variable values for the statistical units in a specific group (study domains). Definition origin: http://stats.oecd.org/glossary/detail.asp?ID=5068	Parent Class: Measure
Variable (variable)	Definition: a symbol that represents a parameter in a formula or expression	Parent Class: Measure

Properties

Name	Annotations	Property Axioms
forYear (for year)	Definition: a predicate indicating the calendar year for which the statistical measure is applicable	Domain: StatisticalMeasure Range: ExplicitDatePeriod
hasApplicableDatePeriod (has applicable	Definition: a predicate indicating the date period for which the statistical measure is applicable	Parent Property: hasDatePeriod

date period)		Domain: StatisticalMeasure Range: DatePeriod
hasArgument (has argument)	Definition: indicates a specific input to a function, formula or expression, also known as an independent variable	Parent Property: has Domain: Reference Range: Reference
hasExpression (has expression)	Definition: indicates a mathematical or other formal expression, which may be part of a formula	Parent Property: has Domain: Reference Range: Expression
hasFormula (has formula)	Definition: indicates a concise way of expressing information symbolically, as in a mathematical or chemical formula	Parent Property: has Range: Formula
hasOperand (has operand)	Definition: indicates the operand of a mathematical argument, parameter or other similar concept	Parent Property: has Domain: Reference Range: Referent
isMeasureOf (is a measure of)	Definition: a predicate indicating the concept being measured	Parent Property: appliesTo Domain: Measure
hasApplicablePeriodEnd (has applicable period end)	Definition: a predicate indicating the end of the period for which the statistical measure is applicable	Parent Property: hasDateTimeValue Domain: StatisticalMeasure Range: dateTime
hasApplicablePeriodStart (has applicable period start)	Definition: a predicate indicating the start of the period for which the statistical measure is applicable	Parent Property: hasDateTimeValue Domain: StatisticalMeasure Range: dateTime
hasMeasurementDate (has measurement date)	Definition: a predicate indicating the date on which the measure was taken	Parent Property: hasDateValue Domain: Measure

		Range: dateValue
hasMeasurementPeriodInMonths (has measurement period in months)	Definition: a predicate indicating the coverage period for which the measure is applicable expressed in months	Domain: Measure Range: integer

10.2 Module: Relations

Table 10-9. Relations Module Metadata

Metadata Term	Value
sm:moduleN6me	Relations
sm:moduleAbbreviation	FIBO-FND-REL
sm:moduleVersion	1.0
sm:moduleAbstract	This module contains an ontology defining a number of reusable relationships. These are used, refined or restricted to define relationships among more specific concepts in other FIBO ontologies. Some of these relationships stand in for relationships which are defined in external standards ontologies.

10.2.1 Ontology: Relations

This ontology defines a set of general-purpose relations for use in other FIBO ontology elements. These include a number of properties required for reuse across the foundations and business entities models.

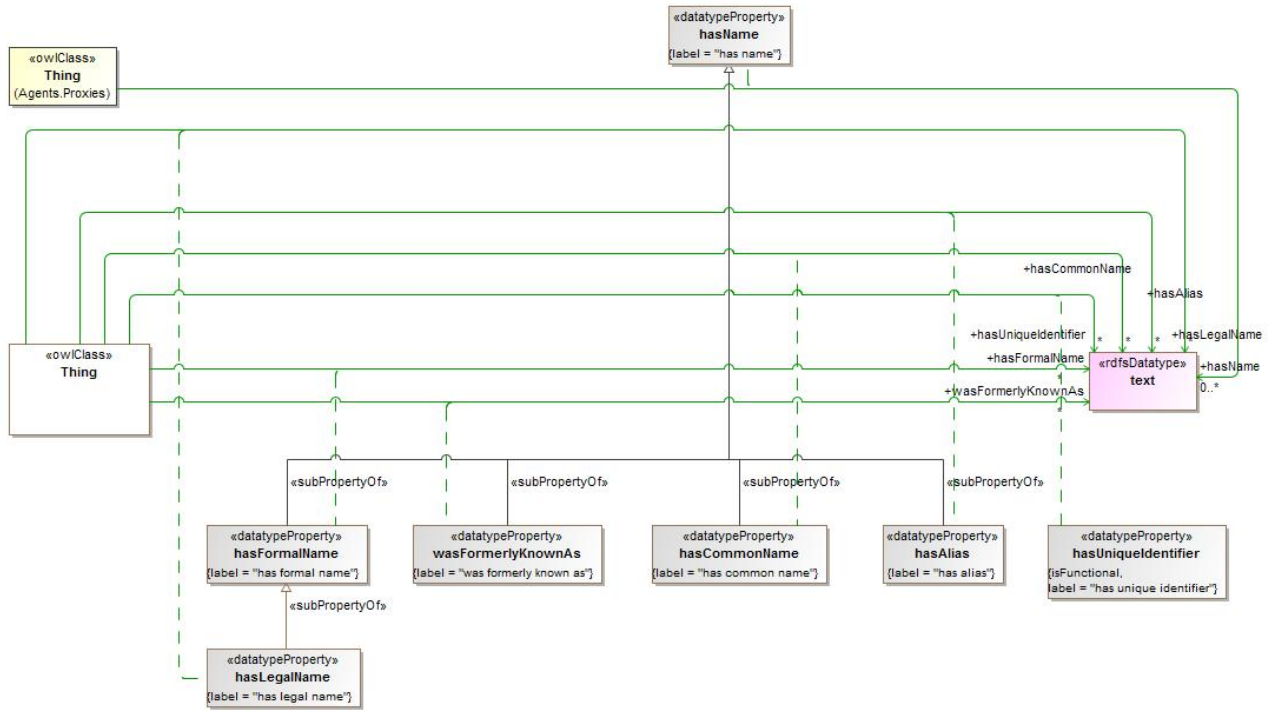

Figure 10.10 Data Properties Textual

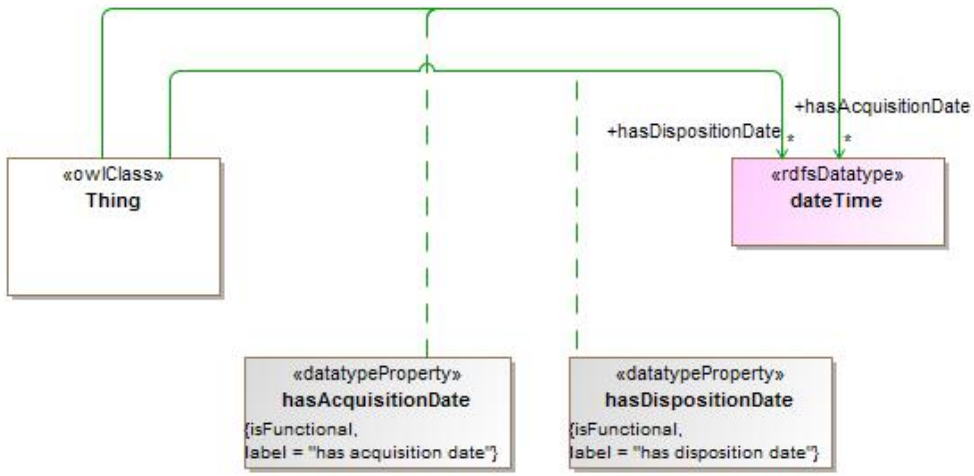

Figure 10.11 Data Properties Numeric

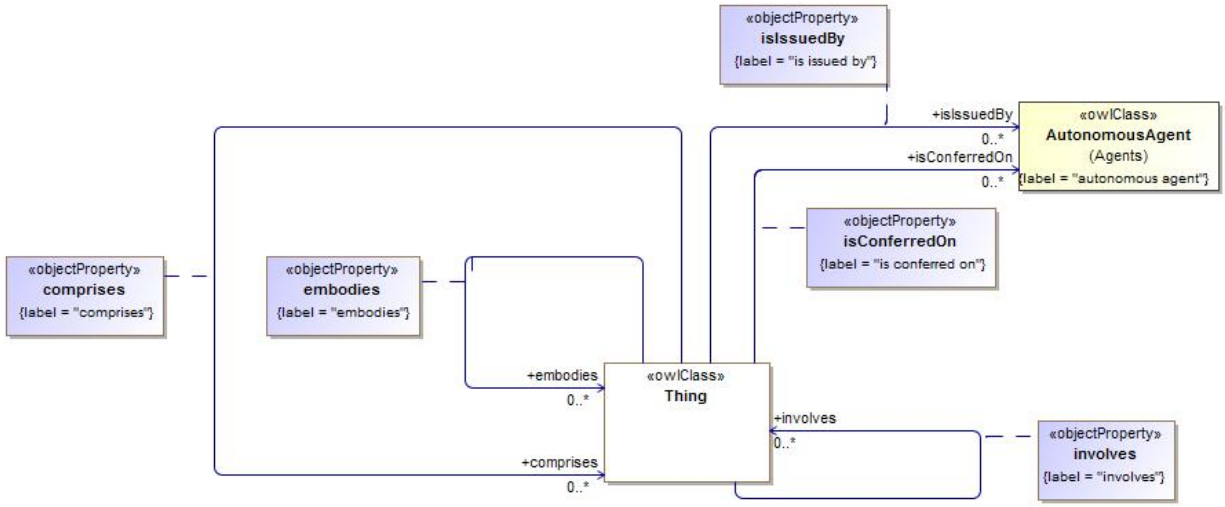

Figure 10.12 Singular Relations

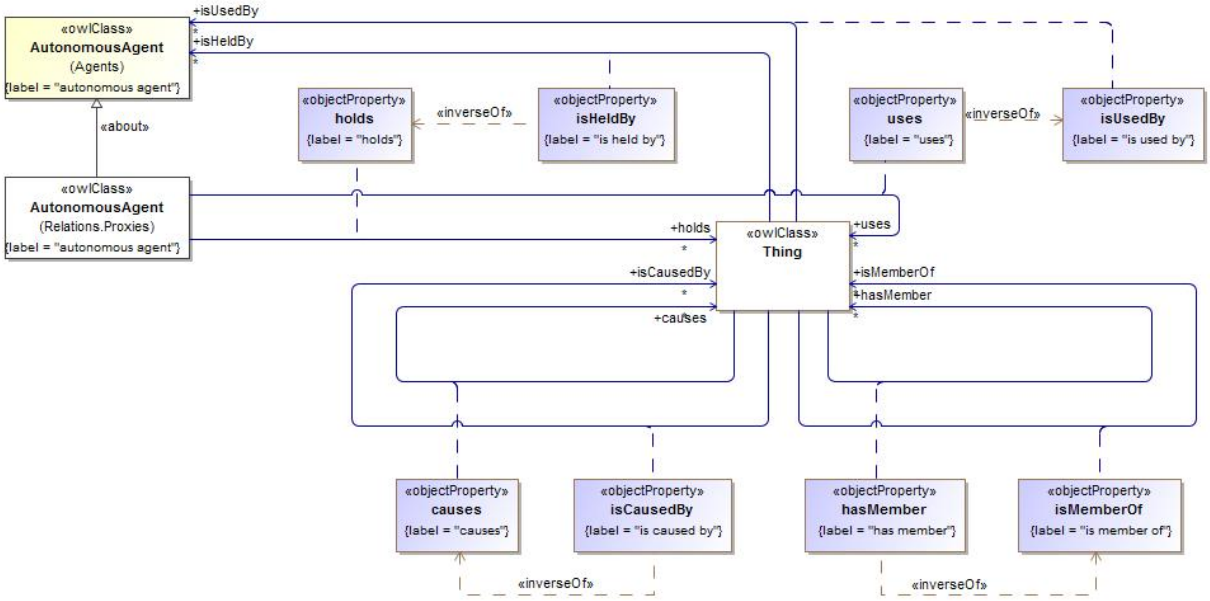

Figure 10.13 Simple Physical Relations

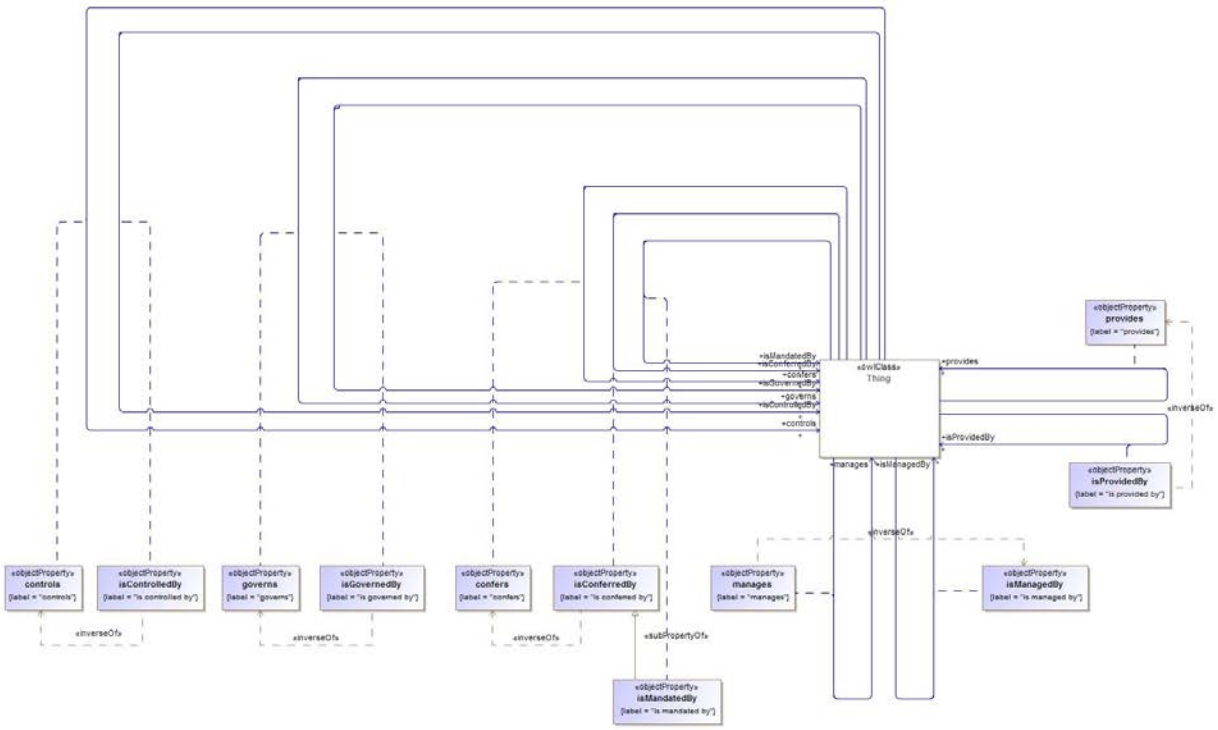

Figure 10.14 Construct Relations

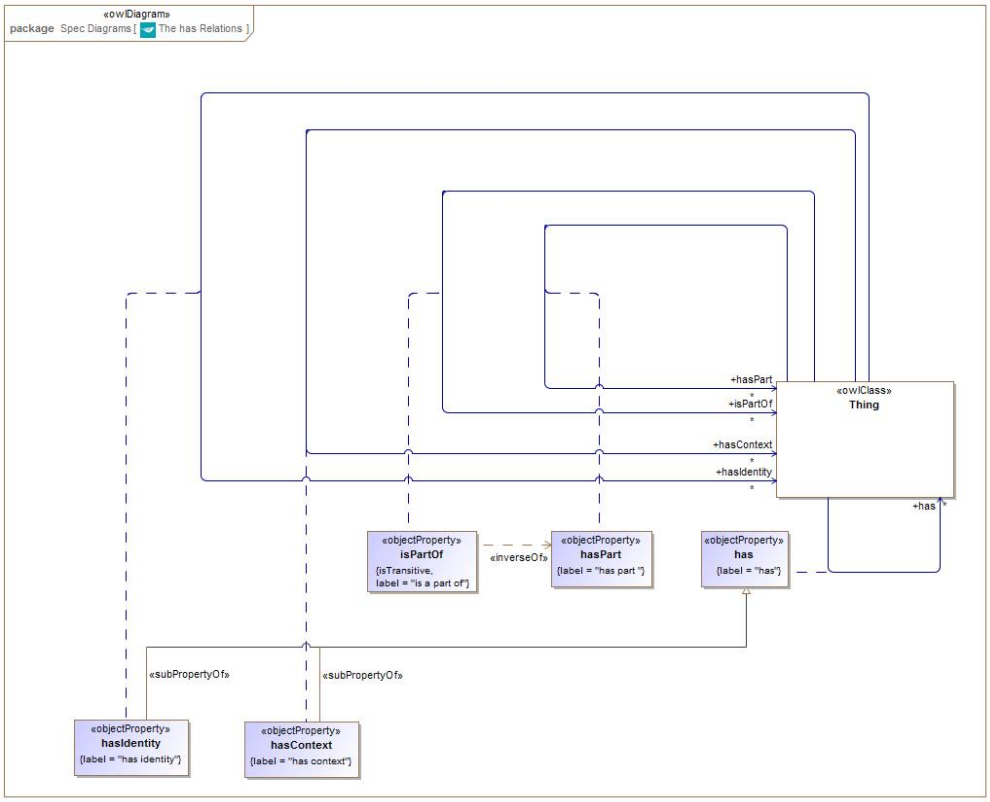

Figure 10.15 The 'has' Relations

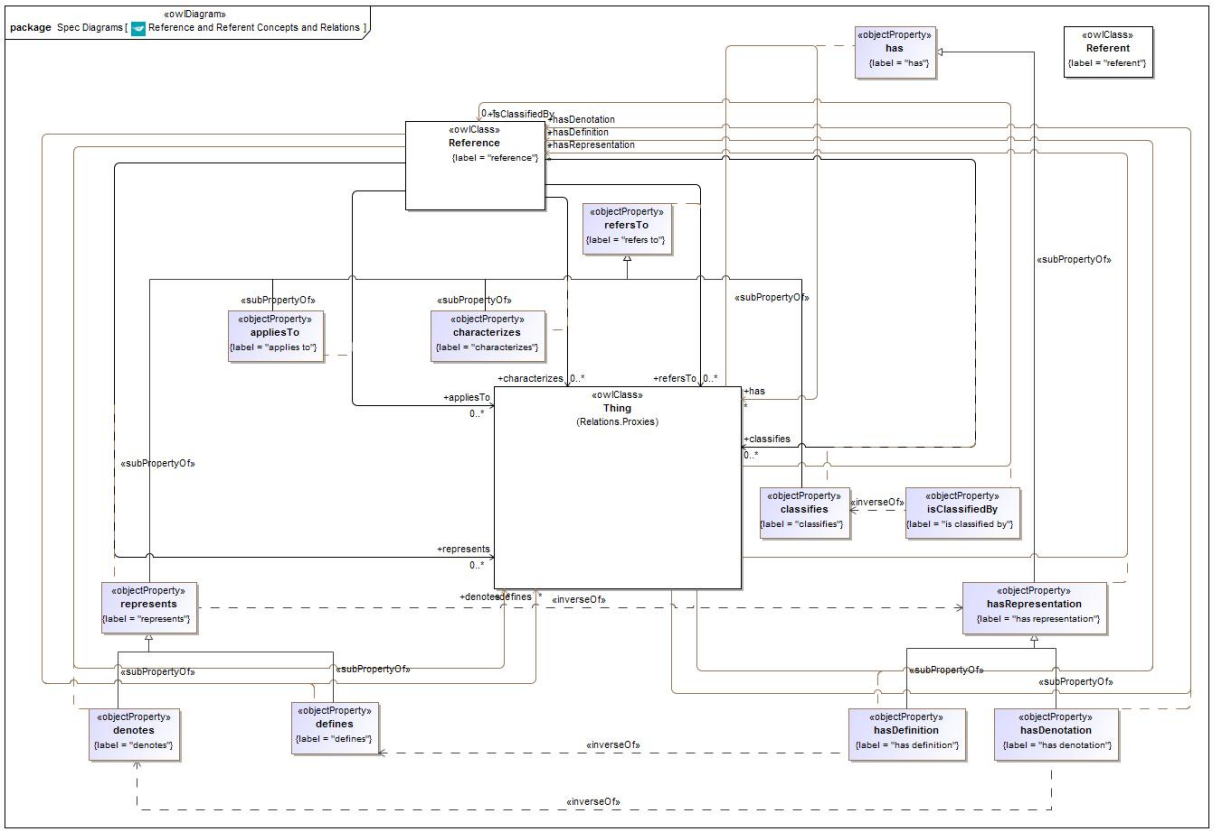

Figure 10.16 Reference and Referent Concepts and Relations

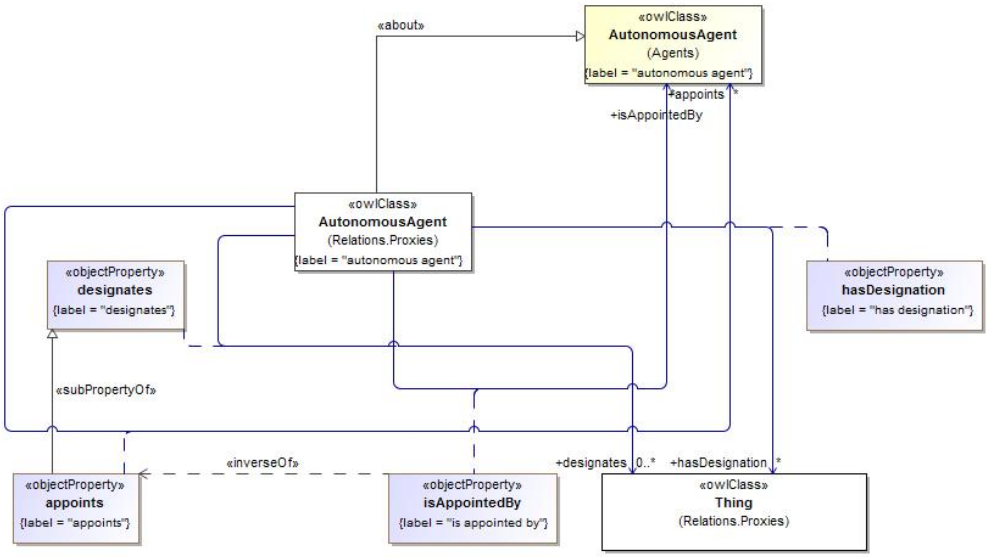

Figure 20.17 Designation and Appointment

Table 10-10. Relations Ontology Metadata

Metadata Term	Value

Metadata Term	Value
sm:filename	Relations Ontology
sm:fileAbbreviation	fibo-fnd-rel-rel
OntologyIRI	http://www.omg.org/spec/EDMC-FIBO/FND/Relations/Relations/
owl:versionIRI	http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Relations/Relations/
sm:dependsOn	http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/AnnotationVocabulary/ http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/BusinessFacingTypes/ http://www.omg.org/spec/EDMC-FIBO/FND/AgentsAndPeople/Agents/

Table 10-11. Relations Details

Classes

Name	Annotations	Class Expressions
AutonomousAgent (autonomous agent)	Definition: Proxy for AutonomousAgent	
Reference (reference)	Definition: a concept that refers to (or stands in for) another concept Adapted From: http://grammar.about.com/od/rs/g/referenceterm.htm	
Referent (referent)	Definition: the concept that another concept stands for or refers to Adapted From: http://grammar.about.com/od/rs/g/referentterm.htm	

Properties

Name	Annotations	Property Axioms
appliesTo (applies to)	Definition: a relation indicating something that is pertinent or relevant to the concept	Parent Property: refers to Domain: Reference
causes (causes)	Definition: the relationship between an event (the cause) and a second event (the effect), where the second event is understood as a consequence of the first; also, the relationship between a set of factors (causes) and a phenomenon (the effect)	Inverse: is caused by
characterizes (characterizes)	Definition: describes the character or quality of	Parent Property: refers to Domain: Reference
classifies (classifies)	Definition: arranges in classes; assigns to a category Adapted From: Merriam-Webster Dictionary	Parent Property: refers to Domain: Reference

Name	Annotations	Property Axioms
		Inverse: is classified by
comprises (comprises)	Definition: includes, especially within a particular scope, is made up of	
confers (confers)	Definition: grants or bestows by virtue of some authority Explanatory Note: This property should be read as describing the conferral of some legal power or duty, some commitment or some social construct, and is a property of some social construct such as an agreement or some legal authority. These concepts, which would describe the kind of thing of which this is a property, and the kinds of thing in terms of which this property is framed, are outside the scope of this mode land so are not shown. Adapted From: Merriam-Webster Online Dictionary	Inverse: is conferred by
defines (defines)	Definition: determines or identifies the essential qualities or meaning of, discovers and sets forth the meaning of, fixes or marks the limits of, demarcates	Parent Property: represents Domain: Reference Inverse: has definition
denotes (denotes)	Definition: represents, calls by a distinctive title, term, or expression	Parent Property: represents Domain: Reference Inverse: has denotation
designates (designates)	Definition: to name something officially or appoint someone to a position officially Adapted From: http://www.dictionarycentral.com/definition/designate.html	Domain: AutonomousAgent
embodies (embodies)	Definition: is an expression of, or gives a tangible or visible form to (an idea, quality, or feeling), makes concrete and perceptible Explanatory Note: This property should be interpreted as being the property of a union of concrete things and information constructs, and as referring to some abstract thing or to some mediating thing or both, in the informative ontology of conceptual abstractions. Adapted From: http://www.merriam-webster.com/dictionary/govern	
governs (governs)	Definition: prevails or has decisive influence over; exercises authority Explanatory Note: This property should be read as being the property of a logical union of social construct (in the informative abstractions ontology) and legal person, and as referring to 'thing'. Adapted From: http://www.merriam-webster.com/dictionary/govern	Inverse: is governed by
has (has)	Definition: indicates that someone (or something) possesses something, as a characteristic, attribute, feature, capability, and so forth Editorial Note: As used in FIBO, this definition of has specifically excludes possession in the sense of ownership. Adapted From: Encarta Webster's Dictionary of the English Language (2004)	
hasAcquisitionDate (has acquisition date)	Definition: links an asset or owner/controller/controllee to the date of acquisition	Range: xsd:dateTime
hasAlias (has alias)	Definition: Any other name by which an individual or organization is known	Parent Property: has name

Name	Annotations	Property Axioms
	Editorial Note: Added at SME Review, to meet AML requirements	Range: text
hasCommonName (has common name)	Definition: a name by which something is frequently referred, without reference to any formal usage or structure	Parent Property: has name Range: text
hasContext (has context)	Definition: provides a context in which something is defined, expressed, or represented Explanatory Note: This property should be read as referring to some context (known as 'mediating thing') in the informative upper ontology which is not included in this model. It should also be read as being the property of some contextually defined thing (known in the informative upper ontology as 'relative thing').	Parent Property: has
hasDefinition (has definition)	Definition: specifies a form of words that conveys the meaning associated with something	Parent Property: has representation Range: Reference Inverse: defines
hasDenotation (has denotation)	Definition: relates a concept (or something else, but typically a concept) to a representation or denotation for that concept	Parent Property: has representation Range: Reference Inverse: denotes
hasDesignation (has designation)	Definition: relates an individual or organization to a position, role, or other designation	Domain: AutonomousAgent
hasDispositionDate (has disposition date)	Definition: links something, such as an asset or its owner/controller/controllee to the date something was sold, transferred, destroyed, etc.	Range: xsd:dateTime
hasFormalName (has formal name)	Definition: a name by which something is known for some official purpose or context, or which is structured in some way such as to always follow the same format regardless of usage	Parent Property: has name Range: text
hasIdentity (has identity)	Definition: provides a means for identifying something that fills a particular role Explanatory Note: This property should be read as being a property of some kind of 'relative thing' as defined externality to this ontology. The property is usually but not exclusively framed with reference to some 'independent thing' but may take other forms and so should be regarded as having a target of 'thing'.	Parent Property: has
hasLegalName (has legal name)	Definition: the name used to refer to an person or organization in legal communications	Parent Property: has formal name Range: text

Name	Annotations	Property Axioms
hasMember (has member)	Definition: relates something, typically a group or organization, to some discrete thing identified as a part (member) of itExplanatory Note: This property should be read as being the property of a logical union of group and organization (not shown).	Inverse: is member of
hasRepresentation (has representation)	Definition: relates a concept to some textual or other symbol which is intended to convey the sense of that concept or to some form of words which sets out the meaning of that concept	Parent Property: has Range: Reference Inverse: represents
hasUniqueIdentifier (has unique identifier)	Definition: has some textual or numeric information which when taken in combination with some associated scheme is unique to the thing and may be used to distinguish it from other things of the same or different type Explanatory Note: With reference to a given (possibly implicit) set of objects, a unique identifier (UID) is any identifier which is guaranteed to be unique among all identifiers used for those objects and for a specific purpose. The uniqueness requires and is guaranteed by the existence of a scheme associated with the identifier.	Range: text
holds (holds)	Definition: is the relationship between an individual or organization and something it possesses, or over which it exercises some ownership or control or has at its discretion the ability to dispose of it as it sees fit	Domain: AutonomousAgent Inverse: is held by
involves (involves)	Definition: (of a situation or event) includes (something) as a necessary part or result	
isAppointedBy (is appointed by)	Definition: indicates the individual or group that has assigned or appointed someone to an office or position	Domain: AutonomousAgent Range: AutonomousAgent Inverse: appoints
isCausedBy (is caused by)	Definition: is the relationship between an event (the effect) and a second event (the cause), where the first event is understood as a consequence of the second; also, the relationship between a set of factors (causes) and a phenomenon (the effect)	Inverse: causes
isClassifiedBy (is classified by)	Definition: indicates the classification scheme used to classify something	Range: Reference Inverse: classifies
isConferredBy (is conferred by)	Definition: a relationship between a right or obligation and the vehicle, such as an agreement or contract, that vests (or confers) said right or obligation Explanatory Note: This property should be read as describing some legal power or duty, some commitment or some social construct being conferred as a result of some social construct such as an agreement or some legal authority. These concepts, which would describe the kind of thing of which this is a property, and the kinds of thing in terms of which this property is framed, are outside the scope of this model and so are not shown.	Inverse: confers
isConferredOn (is conferred on)	Definition: that on which the conferred thing is conferred	Range: AutonomousAgent
isControlledBy (is	Definition: is influenced, managed, or directed by	Inverse: controls

Name	Annotations	Property Axioms
controlled by)		
isGovernedBy (is governed by)	Definition: a relationship between a contract, agreement, jurisdiction, or other legal construct and the regulation, policy, procedure, or legal person that regulates or oversees (governs) it Explanatory Note: This property should be read as being the property of some thing and as referring to a logical union of social construct (in the informative abstractions ontology) and legal person.	Inverse: governs
isHeldBy (is held by)	Definition: something that is possessed by and at least partially under the control of something which can be used or acted on by the holder, regardless of ownership	Range: AutonomousAgent Inverse: holds
isIssuedBy (is issued by)	Definition: identifies an office or organization responsible for circulating, distributing, or publishing something Adapted From: http://www.thefreedictionary.com/issue	Range: AutonomousAgent
isManagedBy (is managed by)	Definition: relates something to another thing that has some role in directing its affairs Explanatory Note: The target or range of this property should be read as always being some kind of 'relative thing', that is a thing defined in some context. Generally this will be a 'party in role'. This property is not intended to be used to relate a thing to some independent thing which it is managed by, only to something in the role of being that which manages it.	Inverse: manages
isMandatedBy (is mandated by)	Definition: relates a responsibility, capacity, or action to that which requires it Explanatory Note: This prooerty should be read as being a property of some social construct as defined in the informative ontology for conceptual abstractions, to some other social construct such as a legal instrument or an agreement.	Parent Property: is conferred by
isMemberOf (is member of)	Definition: belonging, either individually or collectively, to a group Explanatory Note: This property should be read as being framed in terms of a logical union of group and organization (not shown).	Inverse: has member
isProvidedBy (is provided by)	Definition: is made available by Explanatory Note: The target or range of this property should be read as always being some kind of 'relative thing', that is a thing defined in some context. Generally this will be a 'party in role'. This property is not intended to be used to relate a thing to some independent thing which it is provided by, only to something in the role of being that which provides it.	Inverse: provides
isUsedBy (is used by)	Definition: relates something to a thing that has the ability to employ or deploy it as appropriate	Range: AutonomousAgent Inverse: uses
manages (manages)	Definition: relates an autonomous agent to something that it directs in some way Explanatory Note: This property should be read as always being a property of some kind of 'relative thing', that is a thing defined in some	Inverse: is managed by

Name	Annotations	Property Axioms
	context. Generally this will be a 'party in role'. This property is not intended to be used to relate some independent thing to that which it manages, instead it must only be a property of something in the role of being that which manages some thing.	
provides (provides)	Definition: makes something available to Explanatory Note: This property should be read as always being a property of some kind of 'relative thing', that is a thing defined in some context. Generally this will be a 'party in role'. This property is not intended to be used to relate some independent thing to that which it provides, instead it must only be a property of something in the role of being that which provides some thing.	Inverse: is provided by
refersTo (refers to)	Definition: the relationship between a reference and the concept it stands for or refers to, i.e., the referent for that reference	Domain: Reference
represents (represents)	Definition: relates some textual or other symbol or some set of words to some concept that has the sense or meaning the representation is intended to convey	Parent Property: refers to Domain: Reference Inverse: has representation
uses (uses)	Definition: relates an autonomous agent to something that it has the ability to employ in some way	Domain: AutonomousAgent Inverse: is used by
wasFormerlyKnownAs (was formerly known as)	Definition: a name by which something was known in the past	Parent Property: has name Range: text
appoints (appoints)	Definition: assigns a job or role to someone, selects or designates to fill an office or a position, fixes or sets by authority or by mutual agreement Adapted From: Free Online Dictionary	Parent Property: designates Domain: AutonomousAgent Range: AutonomousAgent Inverse: is appointed by
hasPart (has part)	Definition: indicates any portion of a thing, regardless of whether the portion itself is attached to the remainder or detached; cognitively salient or arbitrarily demarcated; self-connected or disconnected; homogeneous or gerrymandered; material or immaterial; extended or unextended; spatial or temporal Explanatory Note: This property relates a thing to anything which is a proper part of that thing. This is not parthood in the sense of a the role of part which may be played by interchangeable things such as wheels; instead this property relates an independent thing to something which makes up a part of it. Adapted From: Stanford Encyclopedia of Philosophy at http://plato.stanford.edu/entries/mereology/	Inverse: is a part of
isPartOf (is a part of)	Definition: relates something to a thing that it is some component or portion of, regardless of how that whole-part relationship is manifested, i.e., attached to the remainder or detached; cognitively salient or arbitrarily demarcated; self-connected or disconnected; homogeneous or gerrymandered; material or immaterial; extended or unextended; spatial or temporal; the most generic part relation, reflexive, asymmetric, and transitive	Inverse: has part

Name	Annotations	Property Axioms
	Explanatory Note: This property represents what is also known in the literature of 'proper parthood',that is the recursive (transitive) relation-ship whereby things have parts which have parts and so on. This is distinct from a separate meaning of 'has part' which would refer to an item playing the named role of a part such as a nearside front wheel. for the avoidance of doubt, this is not that relationship, and this property applies betwen independent things and other independent things which may make up their parts. Adapted From: Stanford Encyclopedia of Philosophy at http://plato.stanford.edu/entries/mereology/	
controls (controls)	Definition: exercises authoritative or dominating influence over; directs Adapted From: The American Heritage(R) Dictionary of the English Language, Fourth Edition	Inverse: is controlled by

10.3 Module: Goals and Objectives

Table 10-12. Goals and Objectives Module Metadata

Metadata Term	Value
sm:moduleName	Goals and Objectives
sm:moduleAbbreviation	FIBO-FND-GAO
sm:moduleVersion	1.0
sm:moduleAbstract	This module includes ontologies for goals and objectives which may be pursued by people or organizations. Goals form the basis for the definition of an organization, and objectives and related concepts are required for describing business plans.

10.3.1 Ontology: Goals

This ontology defines the concept of a goal, for use in other FIBO ontology elements. Goal is defined in general terms and forms one of the basic properties of organizations.

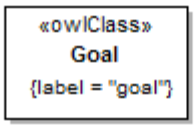

Figure 10.18 Goals Concepts

Table 10-13. Goals Ontology Metadata

Metadata Term	Value
sm:filename	Goals Ontology
sm:fileAbbreviation	fibo-fnd-gao-gl
OntologyIRI	http://www.omg.org/spec/EDMC-FIBO/FND/GoalsAndObjectives/Goals/
owl:versionIRI	http://www.omg.org/spec/EDMC-FIBO/FND/20141101/GoalsAndObjectives/Goals/
sm:dependsOn	http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/AnnotationVocabulary/

Table 10-14. Goals Details

Classes

Name	Annotations	Class Expressions
Goal (goal)	Definition: a goal is a desired result that a person, organization or system envisions or plans, or to which it commits, in order to achieve a desired state Explanatory Note: Many people endeavor to reach goals within a finite time by setting deadlines. Adapted From: http://en.wikipedia.org/wiki/Goal	

10.3.2 Ontology: Objectives

This ontology defines the concept of an objective, for use in other FIBO ontology elements. Objectives are defined as being distinct from goals, in that they constitute time limited and measurable targets which some entity may seek to attain in pursuit of its goals.

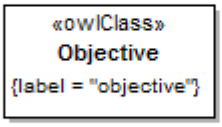

Figure 10.19 Objectives Concepts

Table 10-15. Objectives Ontology Metadata

Metadata Term	Value
sm:filename	Objectives Ontology
sm:fileAbbreviation	fibo-fnd-gao-obj
OntologyIRI	http://www.omg.org/spec/EDMC-FIBO/FND/GoalsAndObjectives/Objectives/
owl:versionIRI	http://www.omg.org/spec/EDMC-FIBO/FND/20141101/GoalsAndObjectives/Objectives/
sm:dependsOn	http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/AnnotationVocabulary/

Table 10-16. Objectives Details

Classes

Name	Annotations	Class Expressions
Objective (objective)	Definition: a statement of a quantitative, measurable result that a person, organization, or system seeks to attain in order to achieve its goals Explanatory Note: This provides an attainable, time-limited, and measurable result that defines strategy and that a person, organization, or system seeks to meet in order to achieve its goals. Definition Origin: Forrester Research	

10.4 Module: Parties

Table 10-17. Parties Module Metadata

Metadata Term	Value
sm:moduleName	Parties
sm:moduleAbbreviation	FIBO-FND-PTY
sm:moduleVersion	1.0
sm:moduleAbstract	This module includes ontologies defining concepts that are highly contextual in nature, such as the meaning of a party in a role, an agent playing a role, and so on. Also covers independent roles themselves. The definitions for agents and parties in roles provide general, reusable patterns for talking about agents performing roles in specific contexts. For example the same person in the context of aviation could be a pilot, and in the context of family could be a mother. These pattern will be refined in other FIBO ontologies to define concepts such as issuer, counterparty, underwriter, etc.

10.4.1 Ontology: Parties

This ontology defines the high-level concepts of parties in roles, for use in other FIBO ontology elements. The concept of a party in a role describes some entity defined specifically in terms of some role which it performs in some formal contractual or transactional relationship. The ontology includes one or more basic party in role concepts. The ontology also includes one or more logical combinations of types of autonomous entity which may perform some of the party roles defined elsewhere in this ontology, such as the role of ownership.

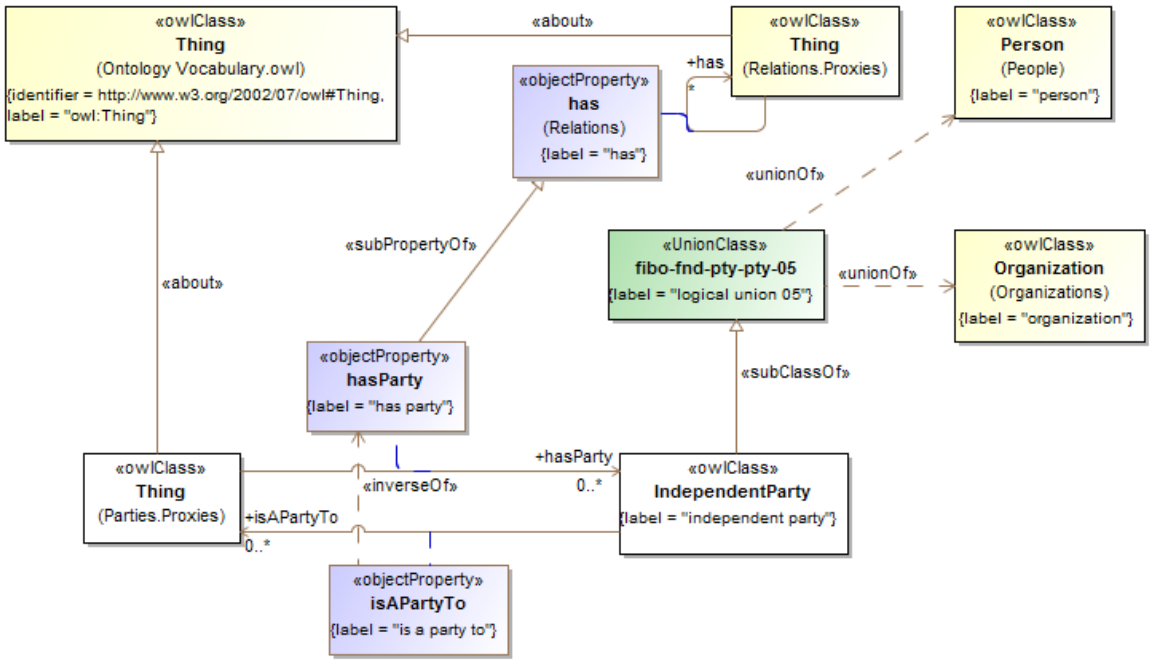

Figure 10.20 Independent Parties

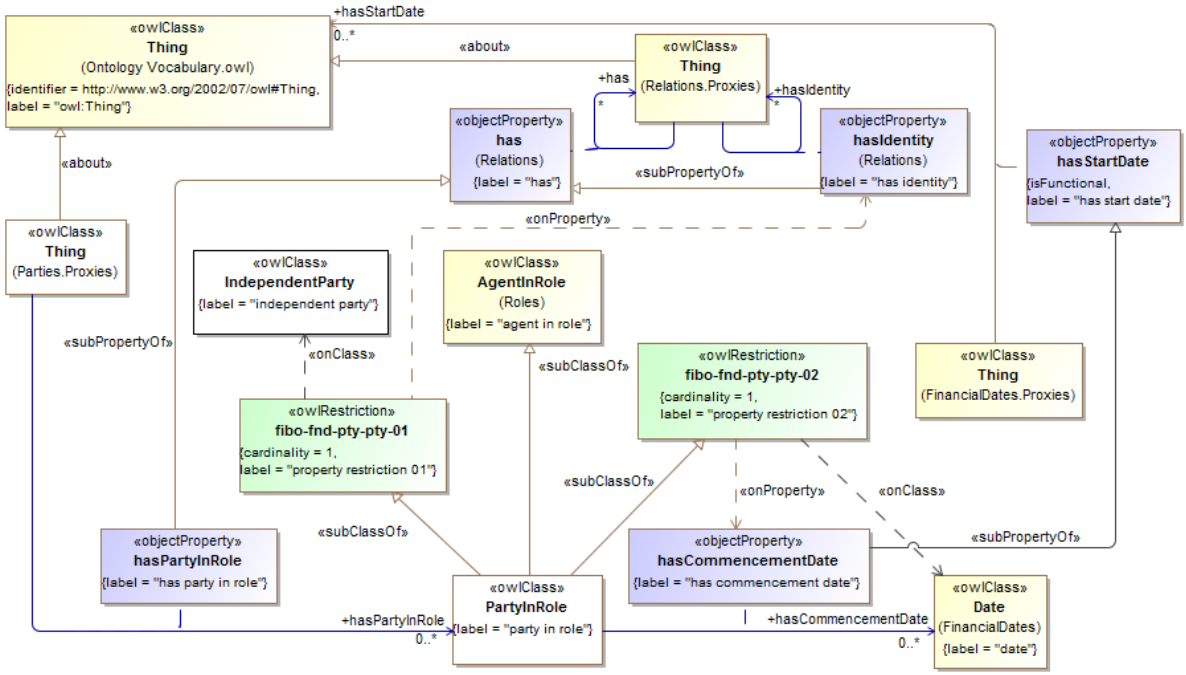

Figure 10.21 Definition of Party In Role

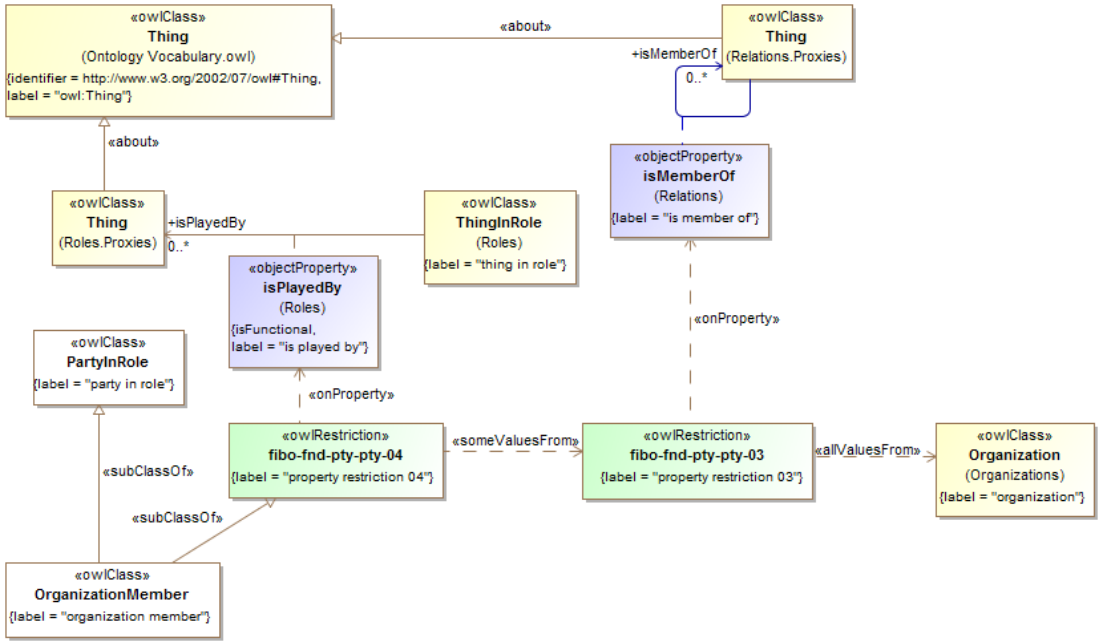

Figure 10.22 Organization Member

Table 10-18. Parties Ontology Metadata

Metadata Term	Value
sm:filename	Parties Ontology
sm:fileAbbreviation	fibo-fnd-pty-pty
OntologyIRI	http://www.omg.org/spec/EDMC-FIBO/FND/Parties/Parties/
owl:versionIRI	http://www.omg.org/spec/EDMC-FIBO/FND/20160201/Parties/Parties/
sm:dependsOn	http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/AnnotationVocabulary/ http://www.omg.org/spec/EDMC-FIBO/FND/Relations/Relations/ http://www.omg.org/spec/EDMC-FIBO/FND/DatesAndTimes/FinancialDates/ http://www.omg.org/spec/EDMC-FIBO/FND/Organizations/Organizations/ http://www.omg.org/spec/EDMC-FIBO/FND/AgentsAndPeople/People/ http://www.omg.org/spec/EDMC-FIBO/FND/Parties/Roles/

Table 10-19. Parties Details

Classes

Name	Annotations	Class Expressions
IndependentParty (independent party)	Definition: any person or organization capable of performing any business party role, such as an individual, a corporation, a partnership, an association, a joint-stock company, a business trust, or an unincorporated organization Adapted from: www.ecfr.gov	Parent Class: (Person ∪ Organization) (fibo-fnd-pty-pty-05)
OrganizationMember (organization member)	Definition: identifies that which has a membership role in some organization	Parent Class: PartyInRole Property Restriction: ∃ **isPlayedBy** (∀ **isMemberOf.Organization)** (fibo-fnd-pty-pty-03, fibo-fnd-pty-pty-04)
PartyInRole (party in role)	Definition: a relative concept that ties an independent party to a specific role they are standing in in which they play some part i.e. are party to Example: Examples include organization member, issuer, owner, partner in a partnership, shareholder, etc. Scope note: The concept of a party in a role refers only to those contexts in which in natural English one would call someone a 'party' for example being party to a contract or to a transaction; it does not cover entities as performing some role in some activity or process (the separate concept Actor covers that). A good test is whether the relative thing defined as PartyInRole can be sensibly said to have a part or play a part in something. Corresponds to the English (not data modeling) sense of the word 'Party'. Adapted from: OMG Property and Casualty Information Models, dtc/12-01-04, Annex A, Glossary of Data Model Terms and Definitions	Parent Class: AgentInRole Property Restriction: = 1 **hasIdentity.IndependentParty** (fibo-fnd-pty-pty-01) Property Restriction: = 1 **hasCommencementDate.Date** (fibo-fnd-pty-pty-02)

Properties

Name	Annotations	Property Axioms
hasCommencementDate (has commencement date)	Definition: the date a party relationship comes into force	Parent Property: hasStartDate Domain: PartyInRole Range: Date
hasParty (has party)	Definition: identifies an independent party associated with an agreement, contract, policy, regulation, or other business arrangement	Parent Property: has Range: IndependentParty

hasPartyInRole (has party in role)	Definition: identifies a party acting in a specific role as related to the particular agreement, contract, policy, regulation, or other business relationship	Parent Property: has Range: PartyInRole
isAPartyTo (is a party to)	Definition: identifies an agreement, contract, policy, regulation, or other business transaction that an independent party is associated with	Domain: IndependentParty Property Axiom: inverse of **hasParty**

10.4.2 Ontology: Roles

This ontology defines some high-level concepts of roles for use in other FIBO ontology elements. These concepts include the basic property whereby something has some role, along with the high-level concept of an agent in a role. The agent in role concept provides the basis for party in role concepts in the PartyRoles ontology and is framed as some entity defined specifically in respect to some role which it performs in some context.

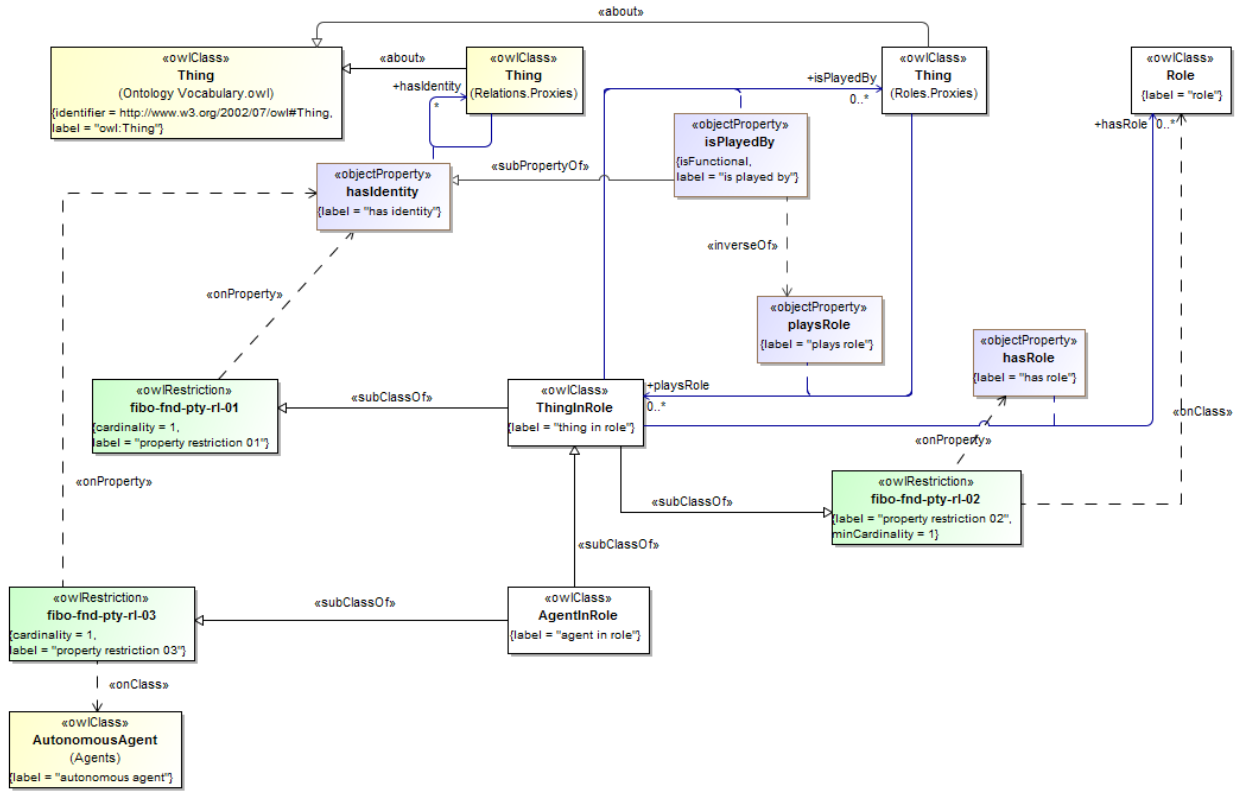

Figure 10.23 Role Definitions

Table 10-20. Roles Ontology Metadata

Metadata Term	Value
sm:filename	Roles Ontology
sm:fileAbbreviation	fibo-fnd-pty-rl
OntologyIRI	http://www.omg.org/spec/EDMC-FIBO/FND/Parties/Roles/
owl:versionIRI	http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Parties/Roles/
sm:dependsOn	http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/AnnotationVocabulary/ http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/BusinessFacingTypes/ http://www.omg.org/spec/EDMC-FIBO/FND/Relations/Relations/ http://www.omg.org/spec/EDMC-FIBO/FND/AgentsAndPeople/Agents/

Table 10-21. Roles Details

Classes

Name	Annotations	Class Expressions
Role (role)	Definition: A role is a set of connected behaviors, rights, obligations, beliefs, and norms as conceptualized by actors in the context of some situation.	
AgentInRole (agent in role)	Definition: An agent-in-role is a relative concept that ties an autonomous agent to a role they are playing in a given situational context.	Parent Class: ThingInRole Property Restriction: =1 **hasIdentity.AutonomousAgent** (fibo-fnd-pty-rl-03)
ThingInRole (thing in role)	Definition: A thing-in-role is a relative concept that ties some thing to a role it plays in a given situational context.	Property Restriction: =1 **hasIdentity** (fibo-fnd-pty-rl-01) Property Restriction: =1 **hasRole.Role** (fibo-fnd-pty-rl-02)

Properties

Name	Annotations	Property Axioms
hasRole (has role)	Definition: provides a means for relating a person, organization, group, or other entity to a role that entity plays in some relationship and context	Parent Property: has Domain: thing in role Range: role

Name	Annotations	Property Axioms
isPlayedBy (is played by)	Definition: indicates the actor (the independent thing) that performs a role.	Parent Property: has identity Domain: thing in role Inverse: playsRole
playsRole (plays role)	Definition: indicates the role that an actor (independent thing) performs.	Range: thing in role Inverse: isPlayedBy

10.5 Module: Arrangements

Table 10-22. Arrangements Module Metadata

Metadata Term	Value
sm:moduleName	Arrangements
sm:moduleAbbreviation	FIBO-FND-ARR
sm:moduleVersion	1.0
sm:moduleAbstract	This module contains ontologies that define abstract concepts, structures and schemata, such as identifiers and identification schemes, indices and indexing schemes, codes and coding schemes, classification strategies, and quantities.

10.5.1 Ontology: Arrangements

This ontology defines abstract structural concepts, including arrangement and collection, for use in other FIBO ontology elements. These abstract concepts are further refined to support definition of identifiers, codes, quantities, and schemata that organize and classify such identifiers and codes.

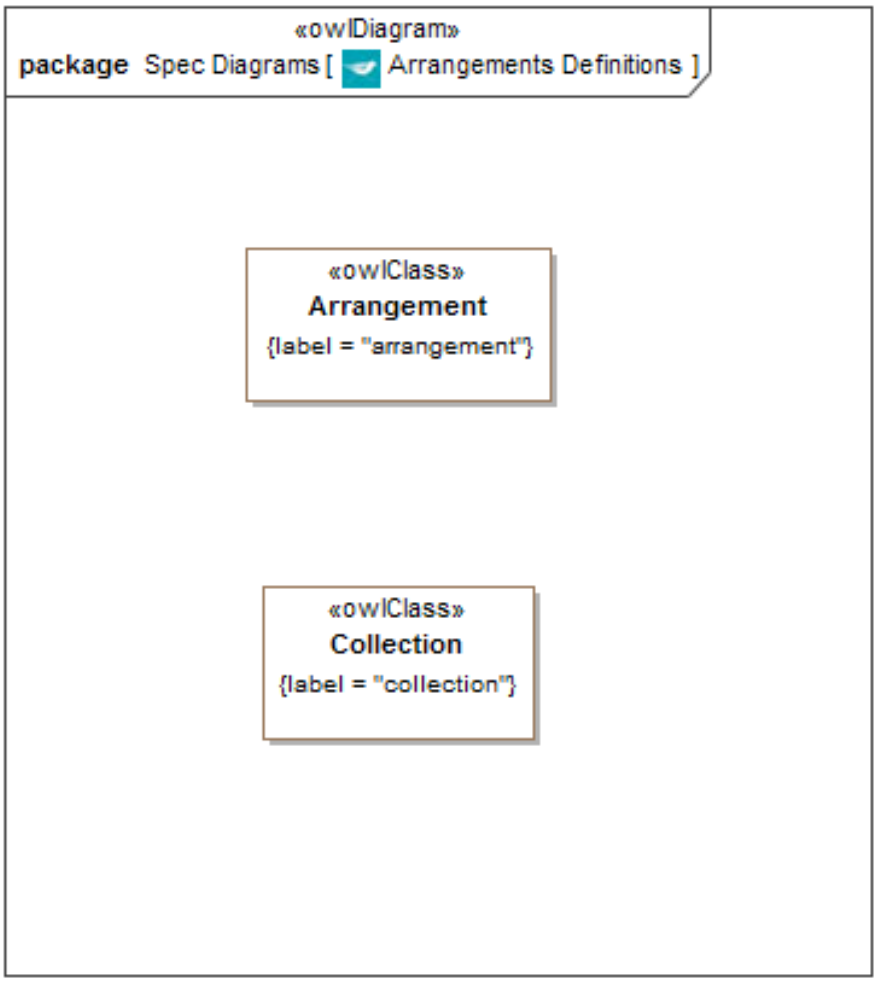

Figure 10.24 Arrangements Concepts

Table 10-23. Arrangements Ontology Metadata

Metadata Term	Value
sm:filename	Arrangements.rdf
sm:fileAbbreviation	fibo-fnd-arr-arr
OntologyIRI	http://www.omg.org/spec/EDMC-FIBO/FND/Arrangements/Arrangements/
owl:versionIRI	http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Arrangements/Arrangements/
sm:dependsOn	http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/AnnotationVocabulary/

Table 10-24. Arrangements Details

Classes

Name	Annotations	Class Expressions
Arrangement (arrangement)	Definition: a structure or means of organizing information such as a schema, numbering system, organization scheme, measurement system, taxonomy, or language for organizing information	
Collection (collection)	Definition: a grouping of some variable number of things (may be zero) that have some shared significance	

10.5.2 Ontology: Classification Schemes

This ontology defines abstract concepts for representation of classification schemes that themselves are intended to permit the classification of arbitrary concepts into hierarchies (or partial orders) for use in other FIBO ontology elements.

Metadata defining the primary metadata elements for the Classification Schemes ontology are given in Table 10-25, below.

Table 10-25 Classification Schemes Ontology Metadata

Metadata Term	Value
sm:filename	ClassificationSchemes.rdf
sm:fileAbbreviation	fibo-fnd-arr-cls
OntologyIRI	http://www.omg.org/spec/EDMC-FIBO/FND/Arrangements/ClassificationSchemes/
owl:versionIRI	http://www.omg.org/spec/EDMC-FIBO/FND/20150801/Arrangements/ClassificationSchemes/
sm:dependsOn	http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/AnnotationVocabulary/ http://www.omg.org/spec/EDMC-FIBO/FND/ Relations/Relations/ http://www.omg.org/spec/EDMC-FIBO/FND/Arrangements/Arrangements/

Figure 10.25 provides an overview of the concepts defined in the Classification Schemes ontology.

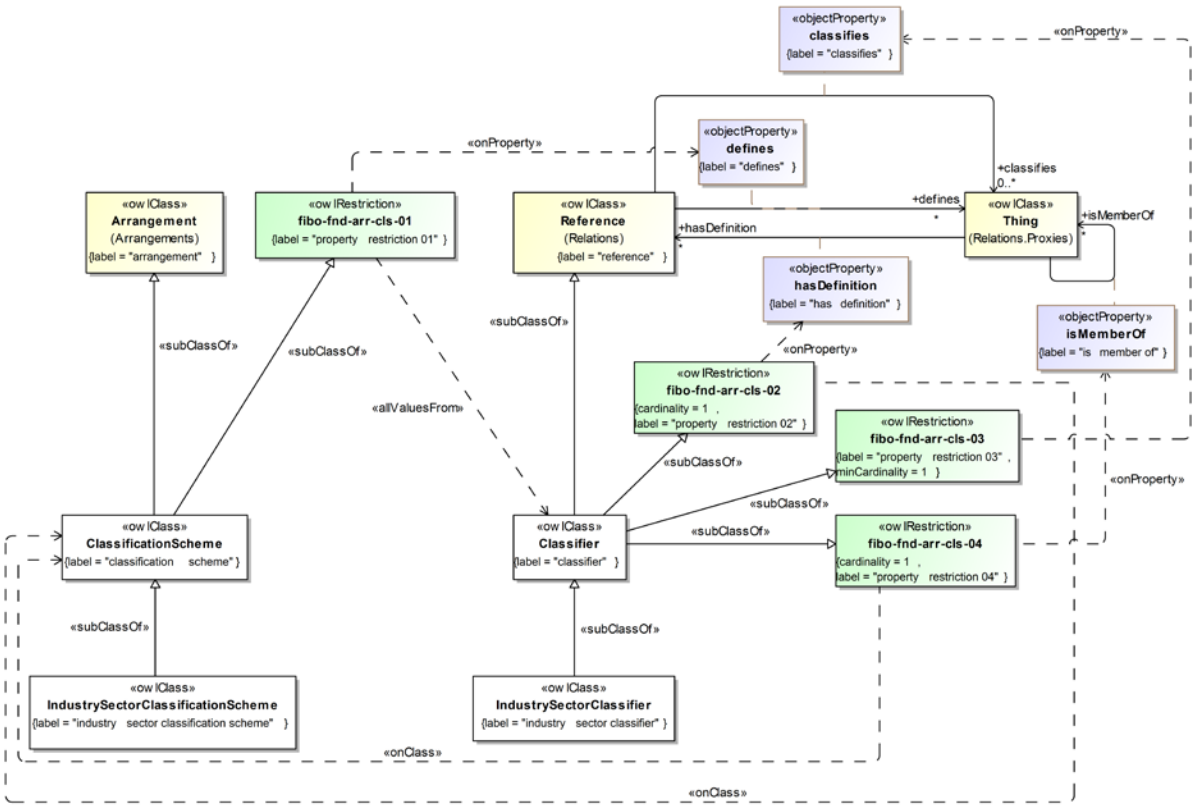

Figure 10.25. Classification Schemes Concepts

Detailed content of the Classification Schemes ontology is provided in Table 10-26, below.

Table 10-26 Classification Schemes Details

Classes

Name	Annotations	Class Expressions
ClassificationScheme (classification scheme)	Definition: a system for allocating classifiers (elements in a classification scheme) to objects, similar to identifiers in some cases; such classification schemes are intended to permit the classification of arbitrary objects into hierarchies (or partial orders) Explanatory note: A classification scheme may be a taxonomy, a network, an ontology, or any other terminological system. The classification may also be just a list of controlled vocabulary of property words (or terms). The list might be taken from the 'leaf level' of a taxonomy. Adapted from: ISO/IEC 11179-3 Information technology - Metadata registries (MDR) - Part 3: Registry metamodel and basic attributes, Third edition, 2013-02-15	Parent Class: Arrangement Property Restriction: ∀**defines**.**Classifier** (fibo-fnd-arr-cls-01)
Classifier (classifier)	Definition: a standardized classification or delineation for something, per some scheme for such delineation, within a specified context Adapted from: ISO/IEC 11179-3 Information technology - Metadata registries (MDR) - Part 3: Registry metamodel and basic attributes, Third edition, 2013-02-15	Parent Class: Reference Property Restriction: = 1 **isMemberOf**. **ClassificationScheme** (fibo-fnd-arr-cls-04) Property Restriction: = 1 **hasDefini-**

Name	Annotations	Class Expressions
		tion.**ClassificationScheme** (fibo-fnd-arr-cls-02) Restriction: ≥ 1 **classifies** (fibo-fnd-arr-cls-03)
IndustrySectorClassificationScheme (industry sector classification scheme)	Definition: a system for allocating classifiers to organizations by industry sector Example: Examples include the North American Industry Classification System (NAICS), and older Standardized Industry Classification (SIC), which is still used today by the US Securities and Exchange Commission (SEC), developed by government to classify industries, and commercial classification schemes, such as the Global Industry Standard Classification (GICS) developed jointly by Morgan Stanley Capital International (MSCI) and Standard and Poor's, or competing schemes including the Industry Classification Benchmark (ICB) system, maintained by Dow Jones and London's FTSE Group, among others. Adapted from: Barron's Dictionary of Business and Economics Terms, 2012 See Also: http://www.investopedia.com/terms/g/gics.asp	Parent Class: ClassificationScheme
IndustrySectorClassifier (industry sector classifier**)**	Definition: a standardized classification or delineation for an organization, or possibly for a security representing an interest in a given organization, per some scheme for such delineation, by industry Adapted from: Barron's Dictionary of Business and Economics Terms, 2012	Parent Class: Classifier

10.5.3 Ontology: Codes

This ontology defines abstract concepts for representation of codes and coding schemes for use in other FIBO ontology elements.

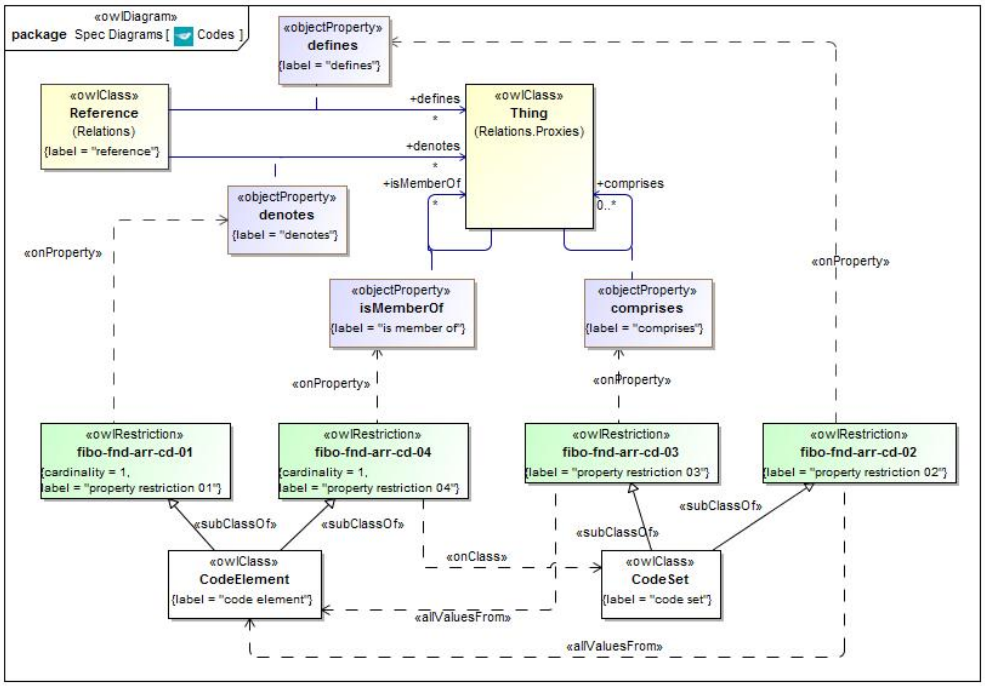

Figure 30.26 Codes

Table 10-27. Codes Ontology Metadata

Metadata Term	Value
sm:filename	Codes
sm:fileAbbreviation	fibo-fnd-arr-cd
OntologyIRI	http://www.omg.org/spec/EDMC-FIBO/FND/Arrangements/Codes/
owl:versionIRI	http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Arrangements/Codes/
sm:dependsOn	http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/AnnotationVocabulary/ http://www.omg.org/spec/EDMC-FIBO/FND/Relations/Relations/ http://www.omg.org/spec/EDMC-FIBO/FND/Arrangements/Arrangements/

Table 10-28. Codes Details

Classes

Name	Annotations	Class Expressions
CodeElement (code element)	Definition: a sequence of characters, capable of identifying that with which it is associated for some purpose, within a specified context, i.e., a code set, according to a pre-established set of rules	Property Restriction: =1 **denotes** (fibo-fnd-arr-cd-01) Property Restriction: =1 **isMemberOf.CodeSet**

Name	Annotations	Class Expressions
		(fibo-fnd-arr-cd-04)
CodeSet (code set)	A system of valid symbols that substitute for specified values, e.g., alpha, numeric, symbols and/or combinations	Property Restriction: ∀ **defines. CodeElement** (fibo-fnd-arr-cd-02) Property Restriction: ∀ **comprises.CodeElement** (fibo-fnd-arr-cd-03)

10.5.4 Ontology: IdentifiersAndIndices

This ontology defines abstract concepts for representation of identifiers, identification schemes, indices and indexing schemes for use in other FIBO ontology elements.

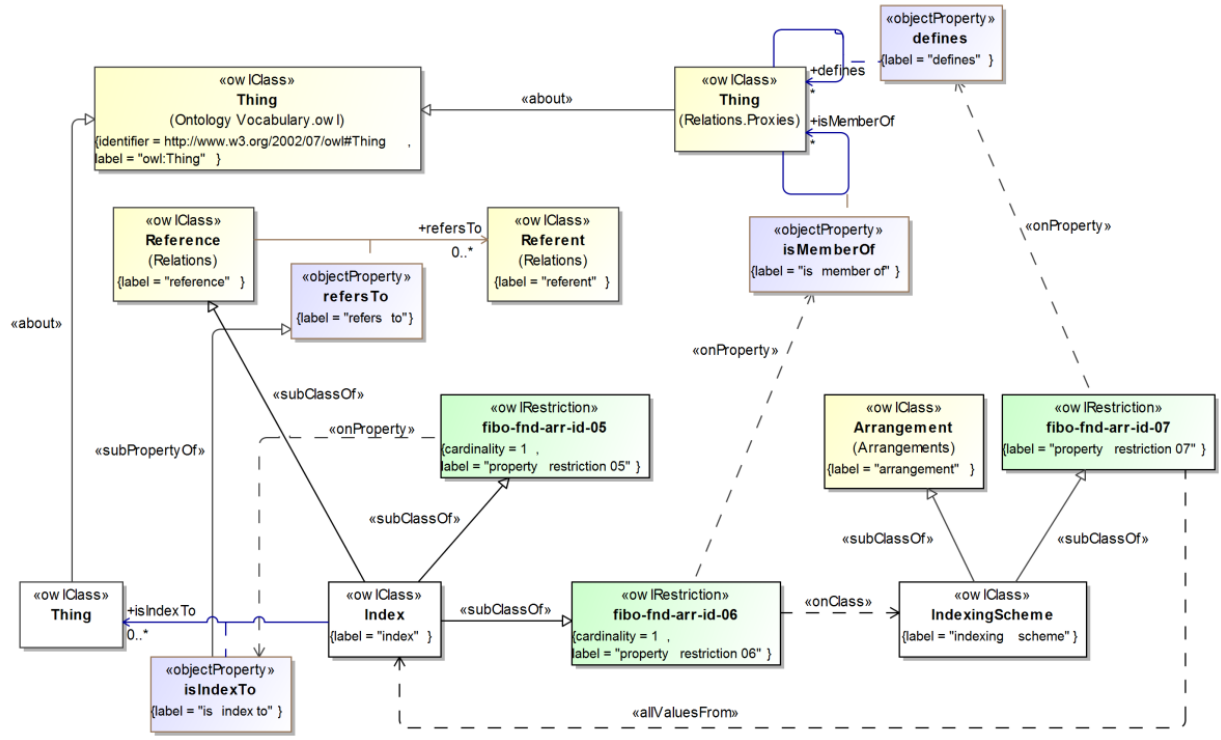

Figure 10.27 Indices and Indexing Schemes

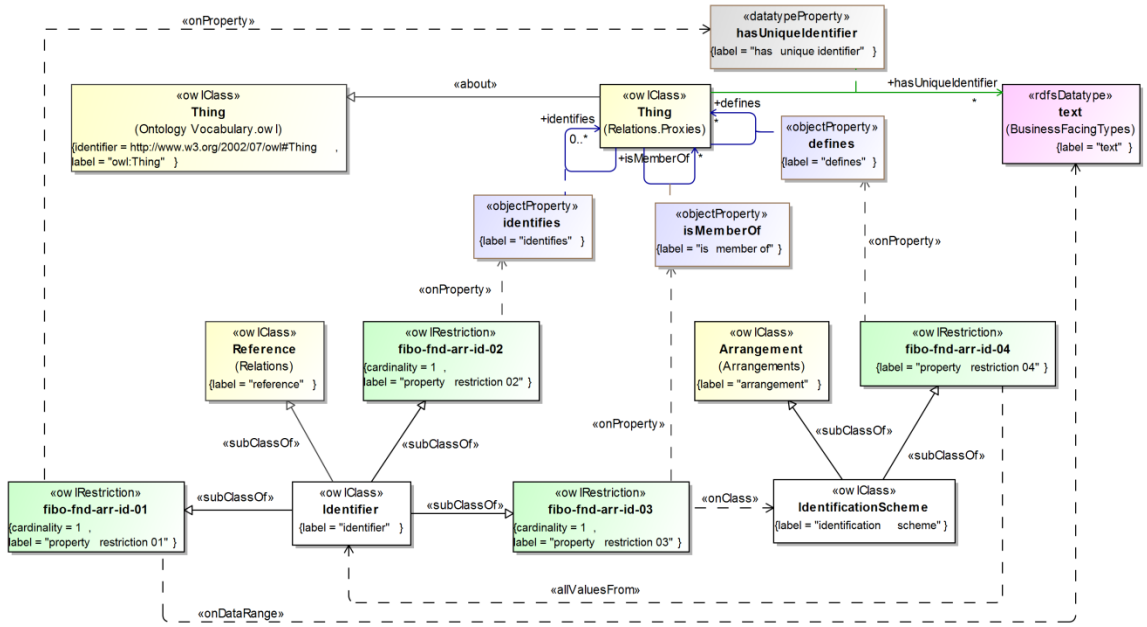

Figure 10.28 Identifiers and Identification Schemes

Table 10-29. IdentifiersAndIndices Ontology Metadata

Metadata Term	Value
sm:filename	IdentifiersAndIndices.rdf
sm:fileAbbreviation	fibo-fnd-arr-id
OntologyIRI	http://www.omg.org/spec/EDMC-FIBO/FND/Arrangements/IdentifiersAndIndices/
owl:versionIRI	http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Arrangements/IdentifiersAndIndices/
sm:dependsOn	http://www.omg.org/spec/EDMC-FIBO/FND/Arrangements/Arrangements/ http://www.omg.org/spec/EDMC-FIBO/FND/Relations/Relations/ http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/AnnotationVocabulary/ http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/BusinessFacingTypes/ http://www.omg.org/spec/EDMC-FIBO/FND/AgentsAndPeople/Agents/

Table 10-30. Identifiers And Indices Details

Classes

Name	Annotations	Class Expressions
IdentificationScheme (identification scheme)	Definition: system for allocating identifiers to objects Adapted from: ISO/IEC 11179-3 Information technology - Metadata registries (MDR) - Part 3: Registry metamodel and basic attributes, Third edition, 2013-02-15	Parent Class: Arrangement Property Restriction: ∀ **defines.Identifier** (fibo-fnd-arr-id-04)
Identifier (identifier)	Definition: sequence of characters, capable of uniquely identifying that with which it is associated, within a specified context Adapted from:	Parent Class: Reference Property Restriction: ≥1 **isMemberOf.IdentificationScheme** (fibo-fnd-arr-id-03) Property Restriction: =1 **identifies** (fibo-fnd-arr-id-02) Property Restriction: =1 **hasUniqueIdentifier.text** (fibo-fnd-arr-id-01)
Index (index)	Definition: an indirect shortcut derived from and pointing into, a greater volume of values, data, information or knowledge Adapted from:	Parent Class: Arrangement Property Restriction: =1 **isIndexTo** (fibo-fnd-arr-id-05) Property Restriction: =1 **isMemberOf.IndexingScheme** (fibo-fnd-arr-id-06)
IndexingScheme (indexing scheme)	Definition: system for indexing values, data, information, or knowledge	Parent Class: Reference Property Restriction: ∀ **defines.Index** (fibo-fnd-arr-id-07)

Properties

Name	Annotations	Property Axioms
isIndexTo (is index to)	Definition: that to which the index refers	Parent Property: refersTo Domain: Index

10.5.5 Ontology: Documents

This ontology defines abstract concepts for representation documents for use in other FIBO ontology elements.

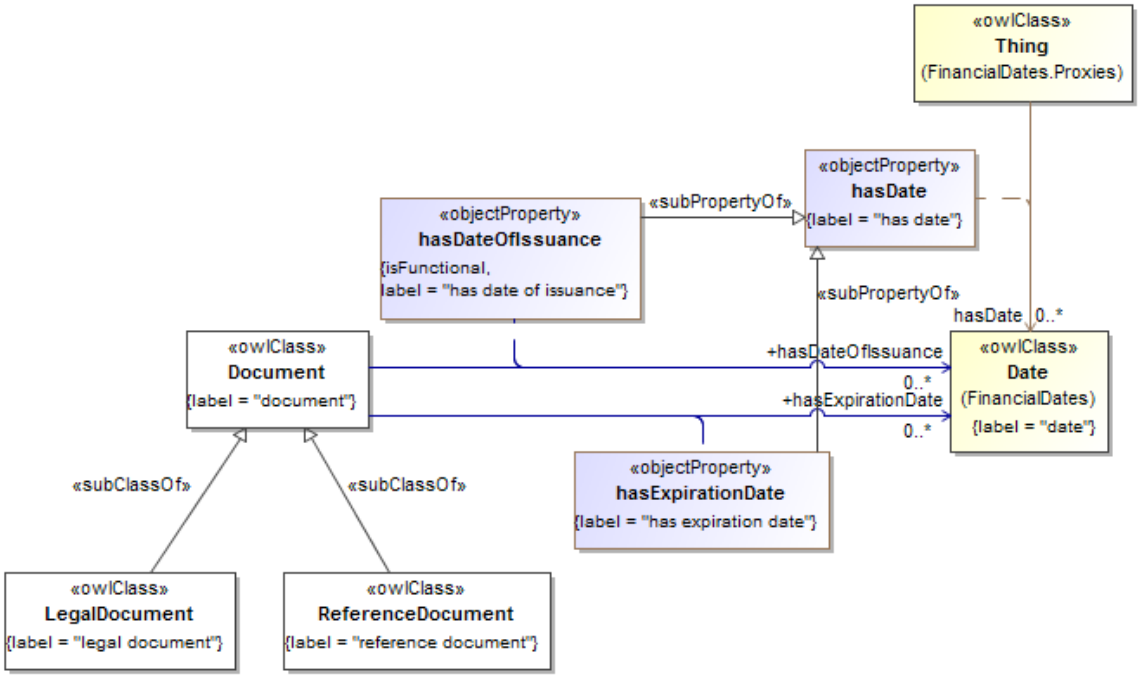

Figure 40.29 Documents Concepts

Table 10-31. Documents Ontology Metadata

Metadata Term	Value
sm:filename	Documents
sm:fileAbbreviation	fibo-fnd-arr-doc
OntologyIRI	http://www.omg.org/spec/EDMC-FIBO/FND/Arrangements/Documents/
owl:versionIRI	http://www.omg.org/spec/EDMC-FIBO/FND/20160201/Arrangements/Documents/
sm:dependsOn	http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/AnnotationVocabulary/ http://www.omg.org/spec/EDMC-FIBO/FND/DatesAndTimes/FinancialDates/

Table 10-32. Documents Details

Classes

Name	Annotations	Class Expressions
Document (document)	Definition: something, such as a recording or a photograph, or a writing that can be used to furnish evidence or information	
LegalDocument (legal document)	Definition: a written or printed paper that bears the original, official, or legal form of something and can be used to furnish decisive evidence or information	Parent Class: Document
ReferenceDocument (reference document)	Definition: a document that provides pertinent details for consultation about a subject Adapted from: ISO/IEC 11179-3 Information technology - Metadata registries (MDR) - Part 3: Registry metamodel and basic attributes, Third edition, 2013-02-15	Parent Class: Document

Properties

Name	Annotations	Property Axioms
hasDateOfIssuance (has date of issuance)	Definition: links something, typically an agreement, contract, or document, with the date it was issued	Parent Property: hasDate Type: FunctionalProperty Domain: Document Range: Date
hasExpirationDate (has expiration date)	Definition: links something, typically an agreement, contract, document, or perishable item, with an expiration date	Parent Property: hasDate Domain: Document Range: Date

10.6 Module: Agents and People

Table 10-33. Agents and People Module Metadata

Metadata Term	Value
sm:moduleName	Agents and People
sm:moduleAbbreviation	FIBO-FND-AAP
sm:moduleVersion	1.0

Metadata Term	Value
sm:moduleAbstract	This module contains ontologies of concepts relating to types of autonomous entity, that is things in the world which are able to determine their own behavior. Includes ontologies for people and for autononomous entities in general.

10.6.1 Ontology: Agents

This ontology defines the concept of autonomous agent for use in other FIBO ontology elements. As defined here, autonomous agent corresponds to what is often referred to as "agent" in software and other systems. It is defined as any entity which is able to act on its own part, and embraces all such things, including people, animals, software agents organizations and all forms of legal persons, although not all of these concepts are elaborated in FIBO as not all are relevant to financial services.

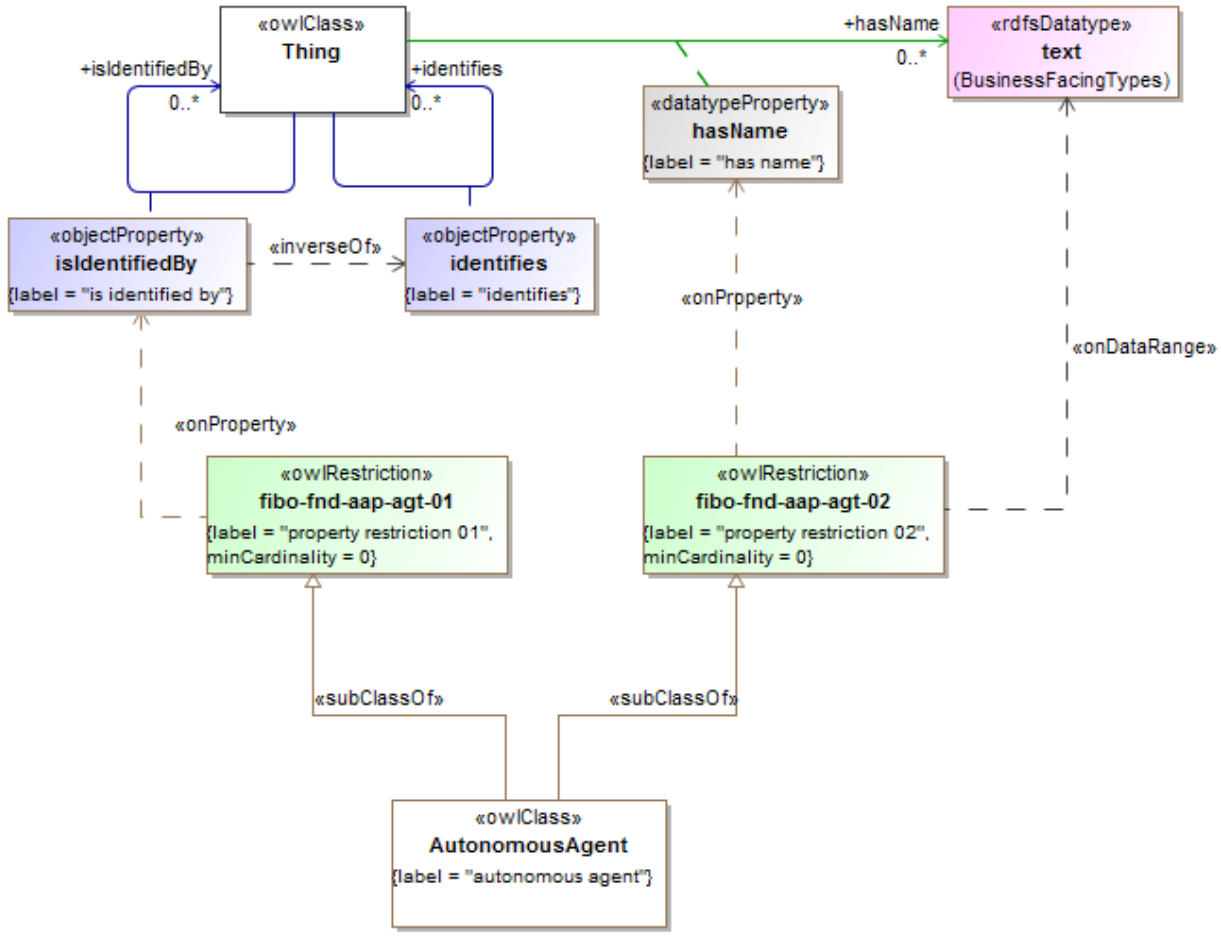

Figure 10.30 Agents Concepts

Table 10-34. Agents Ontology Metadata

Metadata Term	Value
sm:filename	Agents Ontology
sm:fileAbbreviation	fibo-fnd-aap-agt
OntologyIRI	http://www.omg.org/spec/EDMC-FIBO/FND/AgentsAndPeople/Agents/
owl:versionIRI	http://www.omg.org/spec/EDMC-FIBO/FND/20141101/AgentsAndPeople/Agents/
sm:dependsOn	http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/AnnotationVocabulary/ http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/BusinessFacingTypes/

Table 10-35. Agents Details

Classes

Name	Annotations	Class Expressions
AutonomousAgent (autonomous agent)	Definition: An agent is an autonomous individual that can adapt to and interact with its environment.	Property Restriction: ≥ 0 **isIdentifiedBy** (fibo-fnd-aap-agt-01) Property Restriction: ≥ 0 **hasName.text** (fibo-fnd-aap-agt-02)

Properties

Name	Annotations	Property Axioms
hasName (has name)	Definition: that by which some thing is known; may apply to anything	Range: text
identifies (identifies)	Definition: is the relationship between something and that which provides a unique reference for it	Inverse: isIdentifiedBy
isIdentifiedBy (is identified by)	Definition: provides a unique identifier for something	Inverse: identifies

10.6.2 Ontology: People

This ontology defines concepts for people and human related terms, for use in other FIBO ontology elements. People as defined here are human persons only. This ontology sets out a number of basic properties that are held by people or are definitive of a small number of specific types of people such as minors or adults. Primary use cases for determining the set of personal information definitions included are the common elements required to (1) open a bank account, (2) identify a sophisticated investor, and (3) establish foreign account ownership for money laundering purposes.

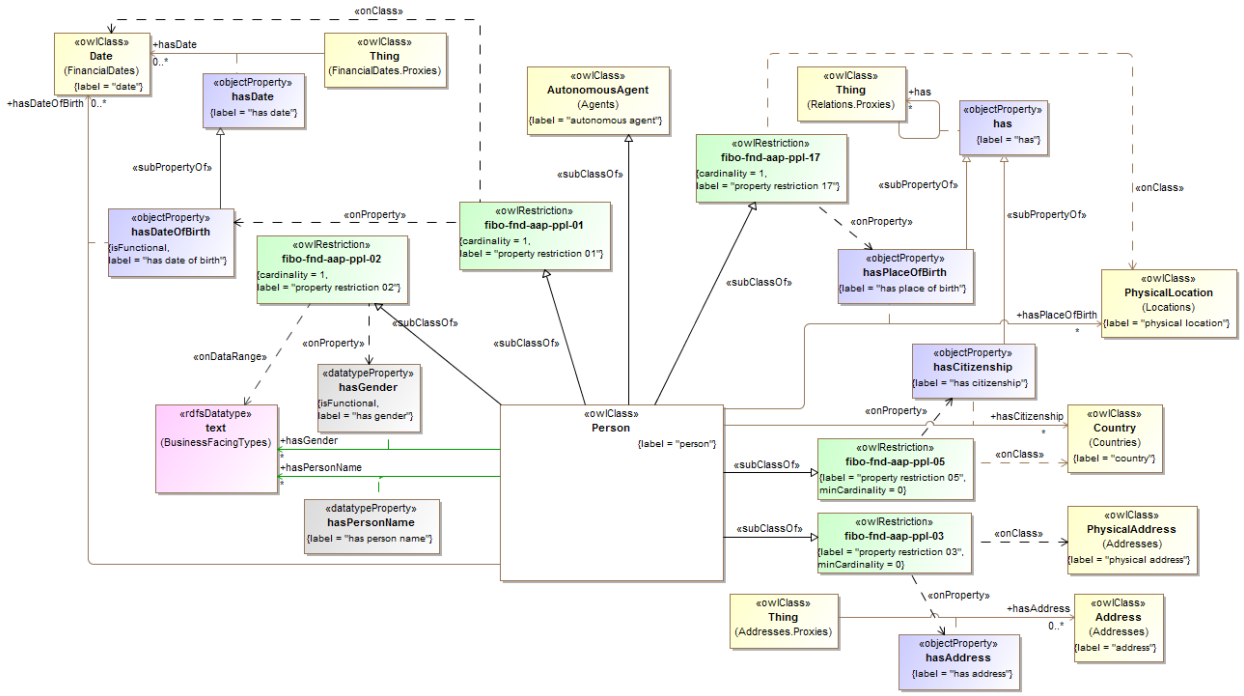

Figure 10.31 Person Fundamentals

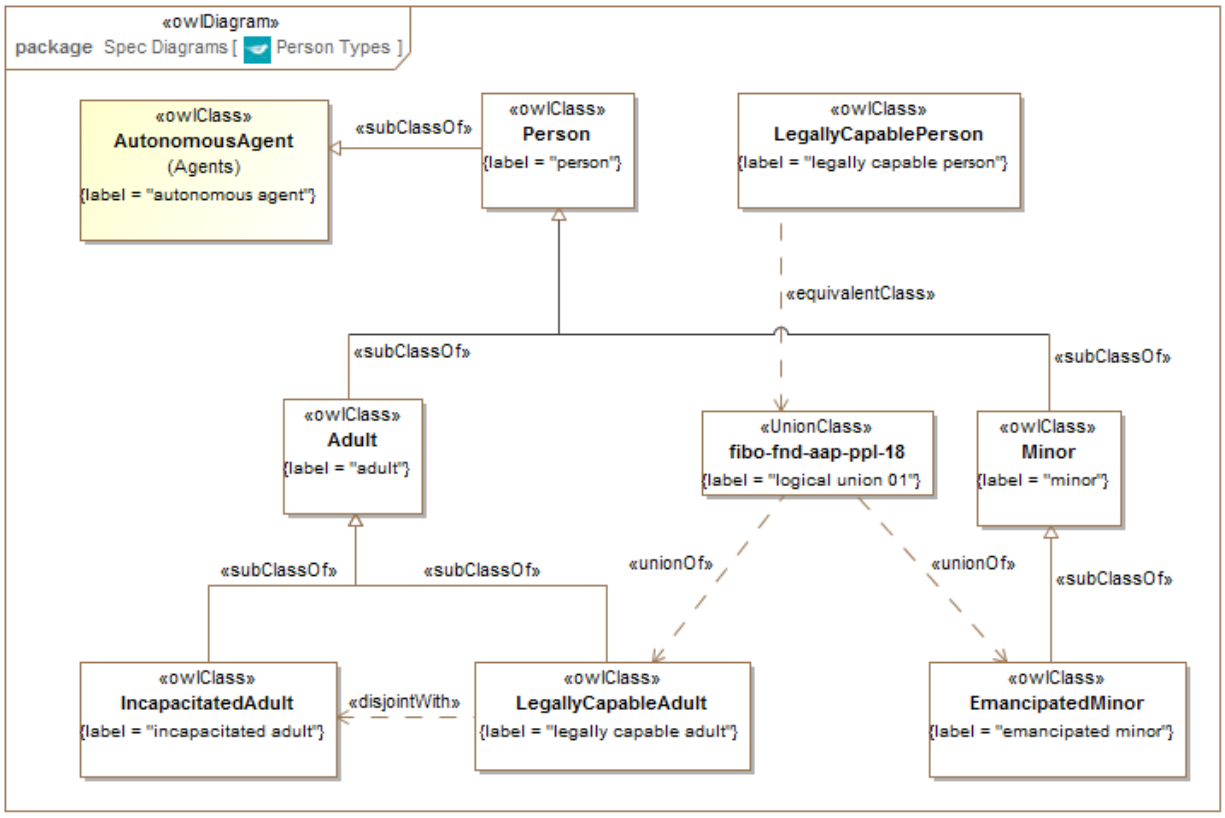

Figure 10.32 Person Types

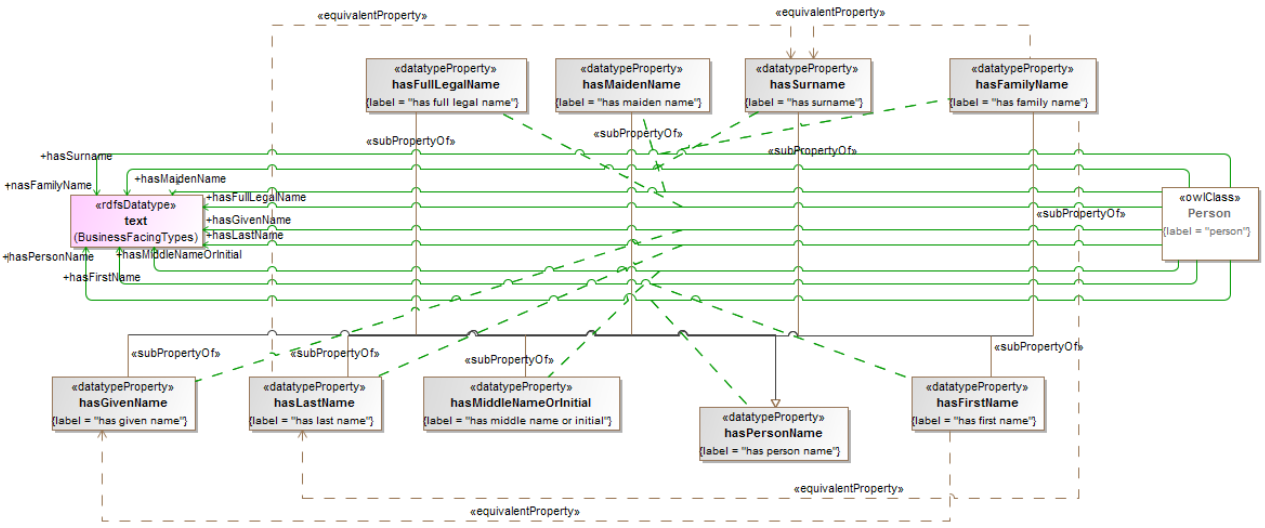

Figure 10.33 People Names

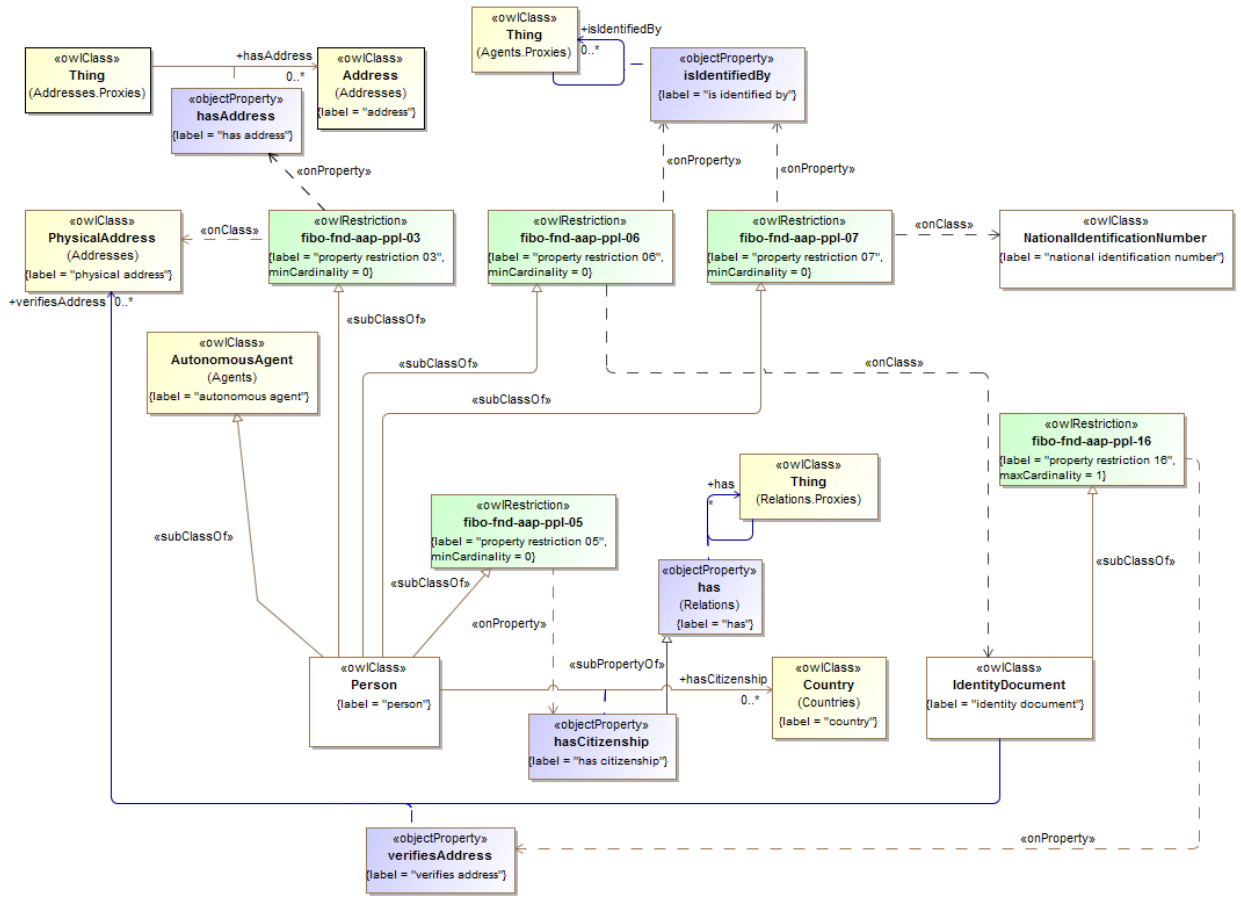

Figure 10.34 People Identification Basics

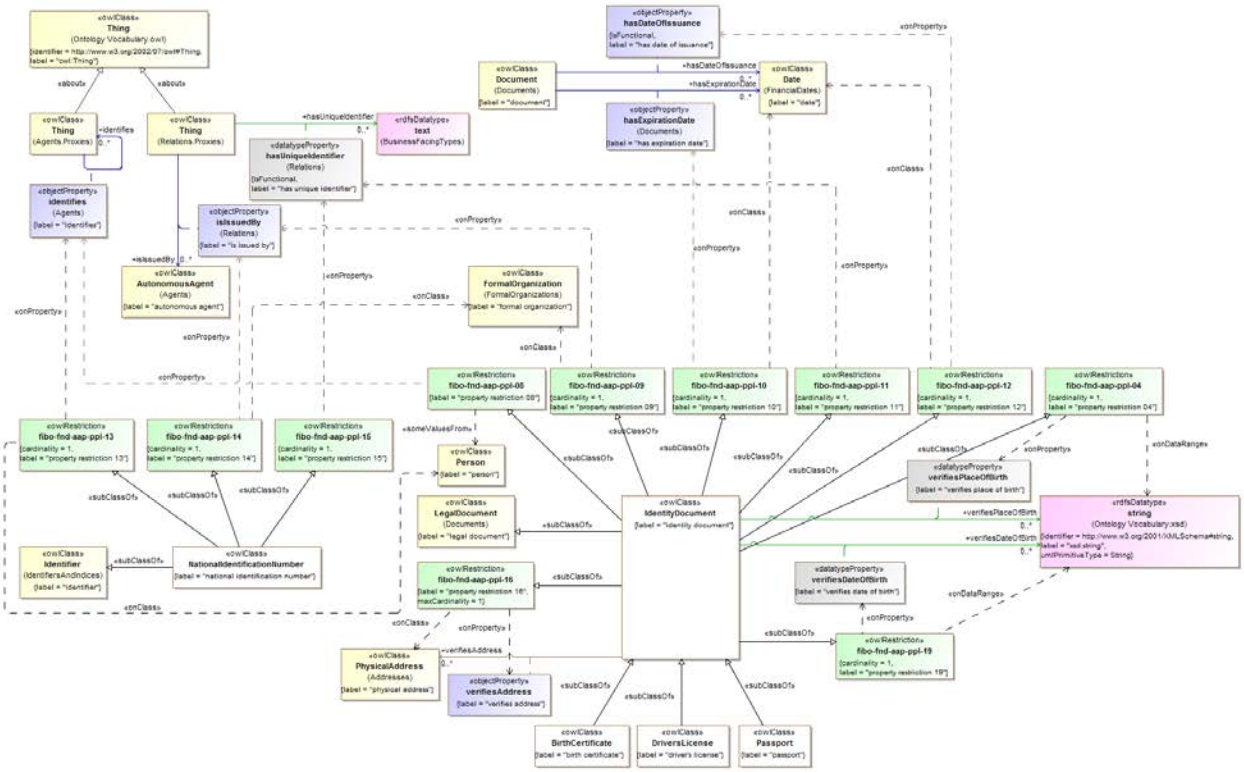

Figure 10.35 Identification Documents

Table 10-36. People Ontology Metadata

Metadata Term	Value
sm:filename	People Ontology
sm:fileAbbreviation	fibo-fnd-aap-ppl
OntologyIRI	http://www.omg.org/spec/EDMC-FIBO/FND/AgentsAndPeople/People/
owl:versionIRI	http://www.omg.org/spec/EDMC-FIBO/FND/20160201/AgentsAndPeople/People/
sm:dependsOn	http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/AnnotationVocabulary/ http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/BusinessFacingTypes/ http://www.omg.org/spec/EDMC-FIBO/FND/Relations/Relations/ http://www.omg.org/spec/EDMC-FIBO/FND/AgentsAndPeople/Agents/ http://www.omg.org/spec/EDMC-FIBO/FND/Arrangements/Documents/ http://www.omg.org/spec/EDMC-

Metadata Term	Value
	FIBO/FND/Arrangements/IdentifiersAndIndices/ http://www.omg.org/spec/EDMC-FIBO/FND/DatesAndTimes/FinancialDates/ http://www.omg.org/spec/EDMC-FIBO/FND/Places/Locations/ http://www.omg.org/spec/EDMC-FIBO/FND/Places/Countries/ http://www.omg.org/spec/EDMC-FIBO/FND/Places/Addresses/ http://www.omg.org/spec/EDMC-FIBO/FND/Organizations/FormalOrganizations/

Table 10-37. People Details

Classes

Name	Annotations	Class Expressions
Adult (adult)	Definition: a person who has attained the age of majority as defined by given jurisdiction Adapted from: https://en.wikipedia.org/wiki/Adult	Parent Class: Person
BirthCertificate (birth certificate, certificate of live birth)	Definition: an original document certifying the circumstances of the birth, or a certified copy of or representation of the ensuing registration of that birth Explanatory note: A birth certificate is a vital record that documents the birth of a child. Depending on the jurisdiction, a record of birth might or might not contain verification of the event by such as a midwife or doctor. Adapted from: https://en.wikipedia.org/wiki/Birth_certificate	Parent Class: IdentityDocument
DriversLicense (driver's license, driving license)	Definition: an official document which states that a person may operate a motorized vehicle, such as a motorcycle, car, truck or a bus, on a public roadway Adapted from: https://en.wikipedia.org/wiki/Non-driver_identification_card	Parent Class: IdentityDocument
EmancipatedMinor (emancipated minor)	Definition: a minor who is allowed to conduct a business or any other occupation on his or her own behalf or for their own account outside the control of a parent or guardian Explanatory note: The minor will then have full contractual capacity to conclude contracts with regard to the business. Whether parental consent is needed to achieve emancipated status varies from case to case. In some cases, court permission is necessary. Protocols vary by jurisdiction.	Parent Class: Minor

Name	Annotations	Class Expressions
	Adapted from: https://en.wikipedia.org/wiki/Emancipated_minor	
IdentityDocument (identity document, identity card)	Definition: any legal document which may be used to verify aspects of a person's identity Explanatory note: If issued in the form of a small, mostly standard-sized card, it is usually called an identity card (IC). Countries which do not have formal identity documents may require informal documents. In the absence of a formal identity document, driving licenses can be used in many countries as a method of proof of identity, although some countries do not accept driving licenses for identification, often because in those countries they don't expire as documents and can be old and easily forged. Most countries accept passports as a form of identification. Most countries have the rule that foreign citizens need to have their passport or occasionally a national identity card from their country available at any time if they do not have residence permit in the country. Adapted from: https://en.wikipedia.org/wiki/Identification_card	Parent Class: LegalDocument Property Restriction: = 1 **hasDateOfIssuance**.**Date** (fibo-fnd-aap-ppl-12) Property Restriction: = 1 **isIssuedBy**.**FormalOrganization** (fibo-fnd-aap-ppl-09) Property Restriction: = 1 **verifiesPlaceOfBirth**.**string** (fibo-fnd-aap-ppl-04) Property Restriction: = 1 **verifiesDateOfBirth**.**string** (fibo-fnd-aap-ppl-19) Property Restriction: ≤ 1 **verifiesAddress**.**PhysicalPlace** (fibo-fnd-aap-ppl-16) Property Restriction: = 1 **hasExpirationDate**.**Date** (fibo-fnd-aap-ppl-10) Property Restriction: = 1 **hasUniqueIdentifier** (fibo-fnd-aap-ppl-11) Property Restriction: ∃ **identifies**.**Person** (fibo-fnd-aap-ppl-08)
Incapacitated Adult (incapacitated adult)	Definition: an adult who is legally identified as not having legal capacity, typically as a result of some inherent physical or mental incapacity or as a result of having contracted some illness which temporarily deprives them of such capacity Explanatory note: Individuals may have an inherent physical condition which prevents them from achieving the normal levels of performance expected from persons of comparable age, or their inability to match current levels of performance may be caused by	Parent Class: Adult Class Axiom: ¬ **LegallyCapablePerson**

Name	Annotations	Class Expressions
	contracting an illness. Whatever the cause, if the resulting condition is such that individuals cannot care for themselves, or may act in ways that are against their interests, those persons are vulnerable through dependency and require the protection of the state against the risks of abuse or exploitation. Hence, any agreements that were made are voidable, and a court may declare that person a ward of the state and grant power of attorney to an appointed legal guardian. Adapted from: https://en.wikipedia.org/wiki/Capacity_(law)	
LegallyCapableAdult (legally capable adult)	Definition: a person who has attained the age of majority as defined by given jurisdiction and is allowed to conduct a business or any other occupation on his or her own behalf or for their own account	Parent Class: Adult
LegallyCapablePerson (legally capable person)	Definition: a person who is allowed to conduct a business or any other occupation on his or her own behalf or for their own account	Class Axiom: ≡ (EmancipatedMinor ∪ LegallyCapableAdult) (fibo-fnd-aap-ppl-18)
Minor (minor)	Definition: a person under a certain age, usually the age of majority in a given jurisdiction, which legally demarcates childhood from adulthood Explanatory note: The age depends upon jurisdiction and application, but is generally 18. Adapted from: https://en.wikipedia.org/wiki/Minor_(law)	Parent Class: Person
NationalIdentificationNumber (national identification number, national identity number, national insurance number)	Definition: a number or text which appears on an identity document issued by a country or jurisdiction Explanatory note: A national identification number, national identity number, or national insurance number is used by the governments of many countries as a means of tracking their citizens, permanent residents, and temporary residents for the purposes of work, taxation, government benefits, health care, and other governmentally-related functions. The number will appear on an identity document issued by a country. The ways in which such a system is implemented are dependent on the country, but in most cases, a citizen is issued an identification number at birth or when they reach a legal age (typically the age of 18). Non-citizens may be issued such numbers when they enter the country, or when granted a temporary or permanent residence permit. Many countries issued such numbers ostensibly for a singular purpose, but over time, they become a de facto national identification number. For example, the United States originally developed its Social Security number system as a means of disbursing Social Security benefits. However, due to function creep, the number has become utilized for other purposes to the point where it is almost essential to have one to, among other things, open a bank account,	Parent Class: LegalDocument Property Restriction: = 1 **isIssuedBy**.**FormalOrganization** (fibo-fnd-aap-ppl-14) Property Restriction: = 1 **hasUniqueIdentifier** (fibo-fnd-aap-ppl-15) Property Restriction: = 1 **identifies**.**Person** (fibo-fnd-aap-ppl-13)

Name	Annotations	Class Expressions
	obtain a credit card, or drive a car. Adapted from: http://en.wikipedia.org/wiki/National_identification_number	
Passport (passport)	Definition: a document, issued by a national government, which certifies the identity and nationality of its holder for the purpose of international travel Explanatory note: The elements of identity contained in all standardized passports include information about the holder, including name, date of birth, gender and place of birth. Adapted from: https://en.wikipedia.org/wiki/Passport	Parent Class: IdentityDocument
Person (person, natural person)	Definition: a person; any member of the species homo sapiens	Parent Class: AutonomousAgent Property Restriction: = 1 **hasDateOfBirth**.**Date** (fibo-fnd-aap-ppl-01) Property Restriction: = 1 **hasPlaceOfBirth.PhysicalLocation** (fibo-fnd-aap-ppl-17) Property Restriction: = 1 **hasGender**.**text** (fibo-fnd-aap-ppl-02) Property Restriction: ≥ 0 **isIdentifiedBy**.**NationalIdentificationNumber** (fibo-fnd-aap-ppl-07) Property Restriction: ≥ 0 **isIdentifiedBy**.**IdentityDocument** (fibo-fnd-aap-ppl-06) Property Restriction: ≥ 0 **hasAddress**.**PhysicalAddress** (fibo-fnd-aap-ppl-03) Property Restriction: ≥ 0 **hasCitizenship**.**Country** (fibo-fnd-aap-ppl-05)

Properties

Name	Annotations	Property Axioms
hasCitizenship (has citizenship)	Definition: links a person to their country of citizenship	Parent Property: has Domain: Person Range: Country
hasDateOfBirth (has date of birth)	Definition: links a person with their date of birth	Parent Property: hasDate Type: FunctionalProperty Domain: Person Range: Date
hasPlaceOfBirth (has place of birth)	Definition: links a person with their place of birth	Parent Property: has Domain: Person Range: PhysicalLocation
verifiesAddress (verifies address)	Definition: verifies a person's address as recorded in an identity document	Parent Property: hasAddress Domain: Person Range: PhysicalAddress
hasFamilyName (has family name)	Definition: the patronymic or family name of a person	Parent Property: **hasPersonName** Property Axiom: ≡ **hasLastName** Property Axiom: ≡ **hasSurname** Domain: Person Range: text
hasFirstName (has first name)	Definition: the given name or first name of a person, that is the name chosen for them at birth or changed by them subsequently from the name given at birth	Parent Property: **hasPersonName** Property Axiom: ≡ **has-**

		GivenName Domain*:* Person Range*:* text
hasFullLegalName (has full legal name)	Definition: the legally complete name of a person, as used in formal dealings of a legal or contractual nature	Parent Property: **hasPerson-Name** Domain*:* Person Range*:* text
hasGender (has gender)	Definition: links a particular gender value with a person	Type: FunctionalProperty Domain*:* Person Range*:* text
hasGivenName (has given name)	Definition: the given name or first name of a person, that is the name chosen for them at birth or changed by them subsequently from the name given at birth	Parent Property: **hasPerson-Name** Domain*:* Person Range*:* text
hasLastName (has last name)	Definition: the patronymic or family name of a person	Parent Property: **hasPerson-Name** Property Axiom: ≡ **hasSurname** Domain*:* Person Range*:* text
hasMaidenName (has maiden name)	Definition: the patronymic or family name which a person was born with and which predates any changes of name due to marriage	Parent Property: **hasPerson-Name** Domain*:* Person Range*:* text
hasMiddleNameOrInitial (has middle name or initial)		Parent Property: **hasPerson-Name** Domain*:* Person

		Range: text
hasPersonName (has person name)	Definition: links any sort of name to an individual person	Parent Property: **hasName** Domain: Person Range: text
hasSurname (has surname)	Definition: the patronymic or family name of a person	Parent Property: **hasPersonName** Domain: Person Range: text
verifiesDateOfBirth (verifies date of birth)	Definition: verifies the person's date of birth	Domain: IdentityDocument Range: string
verifiesPlaceOfBirth (verifies place of birth)	Definition: verifies the person's place of birth	Domain: IdentityDocument Range: string

10.7 Module: Places

Table 10-38. Places Module Metadata

Metadata Term	Value
sm:moduleName	Places
sm:moduleAbbreviation	FIBO-FND-PLC
sm:moduleVersion	1.0
sm:moduleAbstract	This module includes ontologies defining concepts to do with real or virtual places and the addresses to such places. Note that most of these terms are proxies for terms which exist or which are expected to be published in the future in formal ontologies for those concepts (e.g. geophysical, geopolitical, as well as the address components in physical standards like VCard).

10.7.1 Ontology: Locations

This ontology provides a placeholder for use in mapping geographic location-oriented concepts to the appropriate standards.

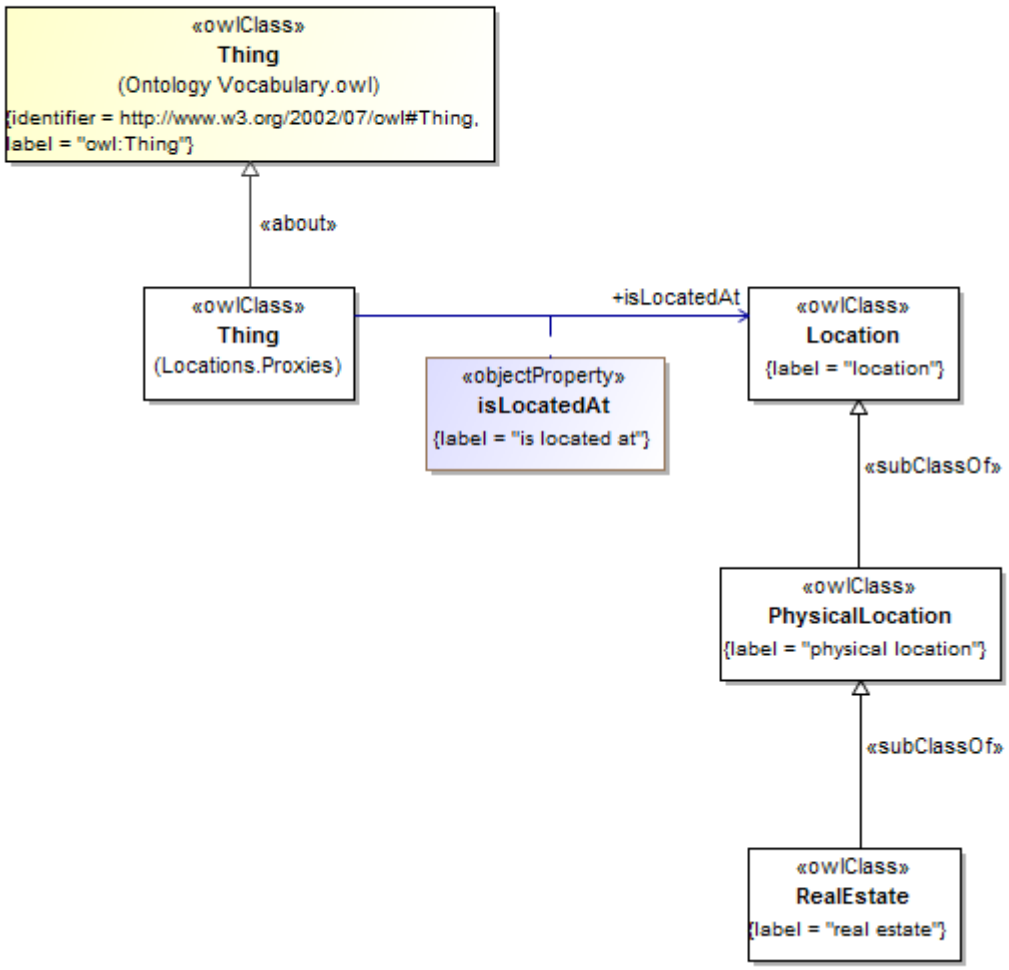

Figure 10.36 Locations Concepts

Table 10-39. Locations Ontology Metadata

Metadata Term	Value
sm:filename	Locations Ontology
sm:fileAbbreviation	fibo-fnd-plc-loc
OntologyIRI	http://www.omg.org/spec/EDMC-FIBO/FND/Places/Locations/
owl:versionIRI	http://www.omg.org/spec/EDMC-

Metadata Term	Value
	FIBO/FND/20141101/Places/Locations/
sm:dependsOn	http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/AnnotationVocabulary/

Table 10-40. Locations Details

Classes

Name	Annotations	Class Expressions
RealEstate (real estate)	Definition: Land plus anything permanently fixed to it, including buildings, sheds and other items attached to the structure. Explanatory Note: Although media often refers to the "real estate market" from the perspective of residential living, real estate can be grouped into three broad categories based on its use, namely residential, commercial and industrial. Examples of real estate include undeveloped land, houses, condominiums, townhomes, office buildings, retail store buildings and factories. Definition Origin: http://www.investopedia.com/terms/r/realestate.asp	Parent Class: PhysicalLocation
PhysicalLocation (physical location)	Definition: A location in physical space	Parent Class: Location
Location (location)	Definition: Anything that can be defined as the answer to a question of the form, Where is...? Scope Note: This includes physical places such as towns, cities, physical addresses, buildings etc. and also abstract places such as on-line trading venues or financial marketplaces.	

Properties

Name	Annotations	Property Axioms
isLocatedAt (is located at)	Definition: a property linking something to a location or place, which might be physical or virtual	Range: Location

10.7.2 Ontology: Countries

This ontology provides a very high level definition of country related concepts, essentially a placeholder for use in mapping countries and intra-country concepts to the appropriate regional standards or to some as yet undefined global address ontology, for use in other FIBO ontology elements. A minimal set of geopolitical and geophysical terms are included as required for financial risk management and other application use cases, and these are all to be considered as placeholders for suitable standard ontologies for these concepts as these become available. These terms may also be mapped to controlled vocabulary standards such as ISO 3166.

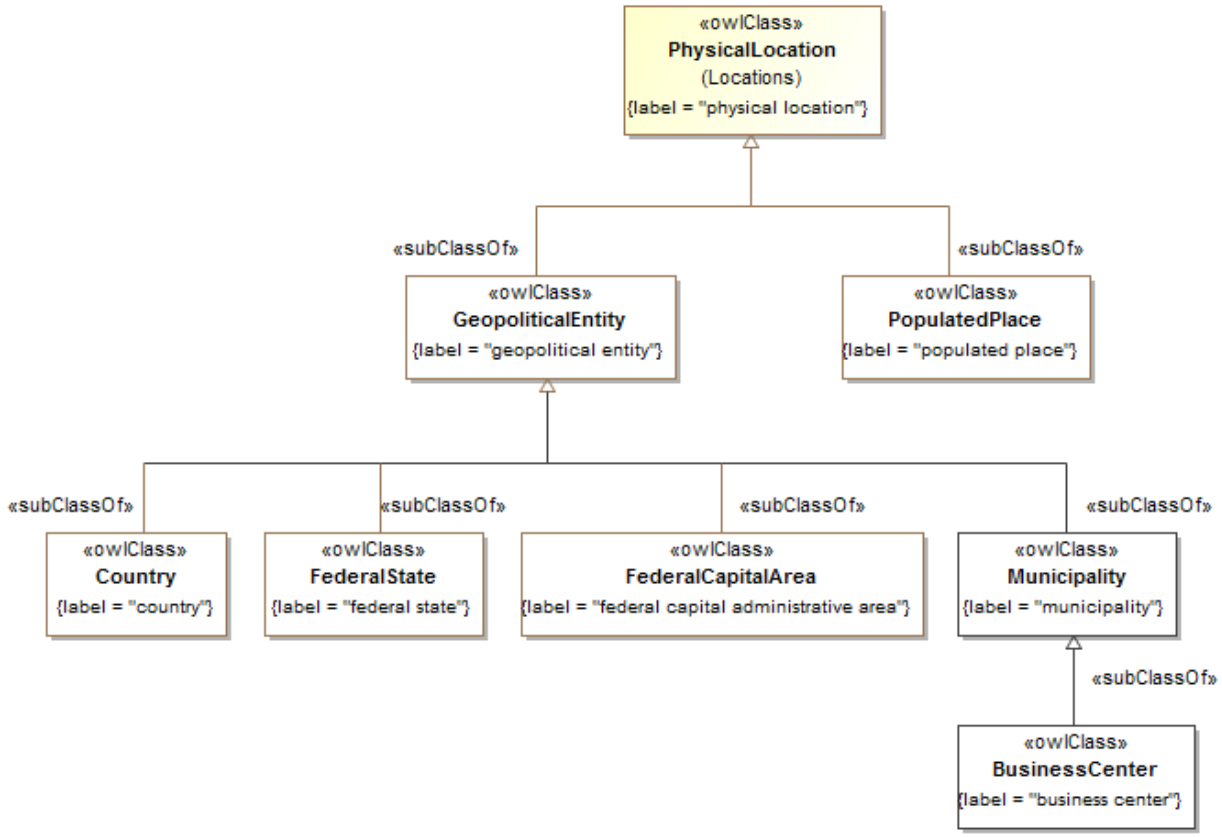

Figure 10.37 Countries Concepts

Table 10-41. Countries Ontology Metadata

Metadata Term	Value
sm:filename	Countries Ontology
sm:fileAbbreviation	fibo-fnd-plc-cty
OntologyIRI	http://www.omg.org/spec/EDMC-FIBO/FND/Places/Countries/
owl:versionIRI	http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Places/Countries/
sm:dependsOn	http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/AnnotationVocabulary/ http://www.omg.org/spec/EDMC-FIBO/FND/Places/Locations/

Table 10-42. Countries Details

Classes

Name	Annotations	Class Expressions
PopulatedPlace (populated place)	Definition: a community in which people live or have lived, without being specific as to size, population or importance	Parent Class: physical location
GeopoliticalEntity (geopolitical entity)	Definition: any country, federal province, city or other administrative unit which is both geographical and political in its identity	Parent Class: physical location
FederalState (federal state)	Definition: a self-governing geopolitical unit which forms part of a wider geopolitical unit that is recognized as a country Explanatory Note: This type of entity, variously referred to as a state, province or canton, has a level of self government including its own legal system and court jurisdiction, but cedes a level of autonomy to the federation of which it forms a part.	Parent Class: geopolitical entity
FederalCapitalArea (federal capital administrative area)	Definition: The capital administrative region of a country which is a federation, if the physical area of this region does not form a part of any of the states or pronvinces which make up the federal country.	Parent Class: geopolitical entity
Country (country)	Definition: A self-governing geopolitical unit that is recognized as a country by the United Nations	Parent Class: geopolitical entity
BusinessCenter (business center)	Definition: a municipality where business is conducted, especially one that is considered a financial center Adapted from: FpML Business Center and related codes, see http://www.fpml.org/coding-scheme/business-center-7-14.xml	Parent Class: municipality
Municipality (municipality)	Definition: an urban administrative division having corporate status and usually powers of self-government or jurisdiction Explanatory Note: A municipality is a general-purpose administrative subdivision, as opposed to a special-purpose district. Adapted from: http://en.wikipedia.org/wiki/Municipality	Parent Class: geopolitical entity

10.7.3 Ontology: Addresses

This ontology provides a very high level definition of address, essentially a placeholder for use in mapping addresses to the appropriate regional standards or to some as yet undefined global address ontology, for use in other FIBO ontology elements. A minimal set of address related terms are included as required for financial risk management and other application use cases, and these are all to be considered as placeholders for suitable global address standards as these become available.

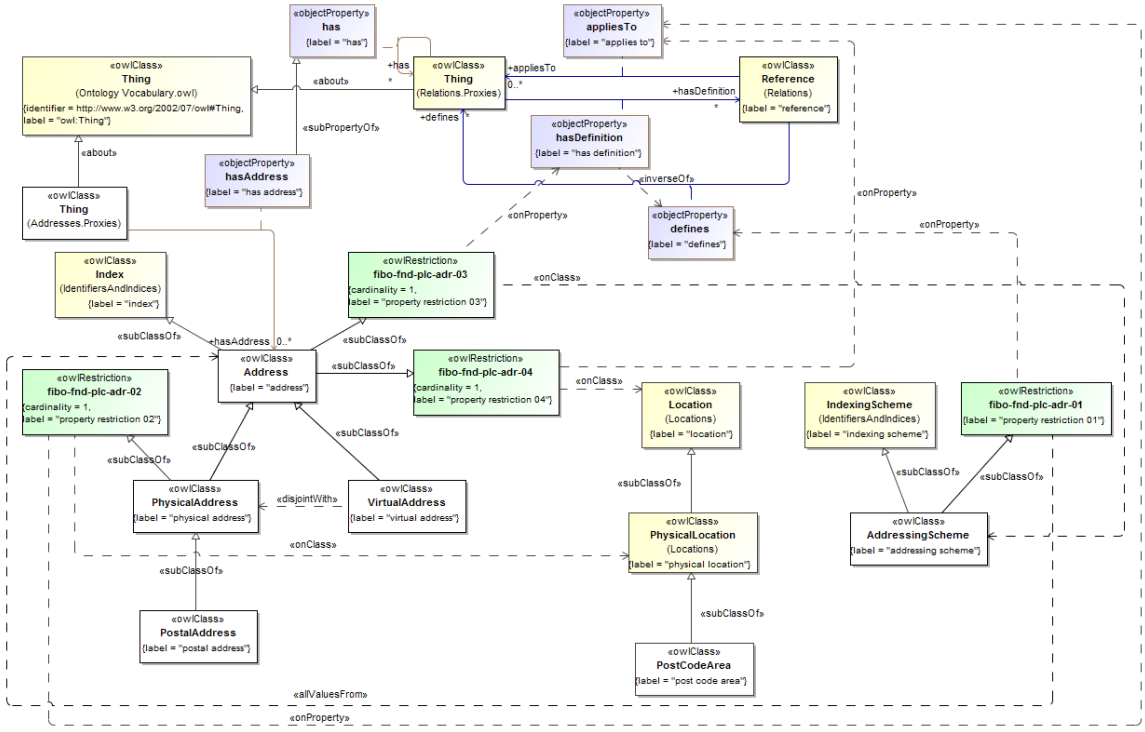

Figure 10.38 Addresses Concepts

Table 10-43. Addresses Ontology Metadata

Metadata Term	Value
sm:filename	Addresses Ontology
sm:fileAbbreviation	fibo-fnd-plc-adr
OntologyIRI	http://www.omg.org/spec/EDMC-FIBO/FND/Places/Addresses/
owl:versionIRI	http://www.omg.org/spec/EDMC-FIBO/FND/20160201/Places/Addresses/
sm:dependsOn	http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/AnnotationVocabulary/ http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/BusinessFacingTypes/ http://www.omg.org/spec/EDMC-FIBO/FND/Relations/Relations/ http://www.omg.org/spec/EDMC-FIBO/FND/Places/Locations/

Table 10-44. Addresses Details

Classes

Name	Annotations	Class Expressions
Address (address)	Definition: an index to a location to which communications may be delivered	Parent Class: Index Property Restriction: = 1 **appliesTo**.**Location** (fibo-fnd-plc-adr-04) Property Restriction: = 1 **hasDefinition**.**AddressingScheme** (fibo-fnd-plc-adr-03)
AddressingScheme (addressing scheme)	Definition: a system for allocating addresses to objects	Parent Class: IndexingScheme Property Restriction: ∀ **defines**.**Address** (fibo-fnd-plc-adr-01)
PostCodeArea (post code area)	Definition: a physical area uniquely identified by some postal code	Parent Class: PhysicalLocation
PostalAddress (postal address)	Definition: a physical address where postal communications can be addressed, for any kind of organization or person	Parent Class: PhysicalAddress
PhysicalAddress (physical address)	Definition: a physical address where communications can be addressed, papers served or representatives located for any kind of organization or person Scope note: An address is a collection of information, presented in a mostly fixed format, used for describing the location of a building, apartment, or other structure or a plot of land, generally using political boundaries and street names as references, along with other identifiers such as house or apartment numbers. Some addresses also contain special codes to aid routing of mail and packages, such as a ZIP code or post code. (Wikipedia)	Parent Class: Address Property Restriction: = 1 **appliesTo**.**PhysicalLocation** (fibo-fnd-plc-adr-02)
VirtualAddress (virtual address)	Definition: an address identifying a virtual, i.e. non-physical location	Parent Class: Address Class Axiom: ¬ **PhysicalAddress**

Properties

Name	Annotations	Property Axioms
hasAddress (has address)	Definition: has a means by which the entity may be located or contacted or may receive correspondence	Parent Property: has

Range: Address

10.7.4 Ontology: Facilities

This ontology provides scaffolding for use in describing concepts related to facilities, both virtual and physical, including physical sites that provide various facilities.

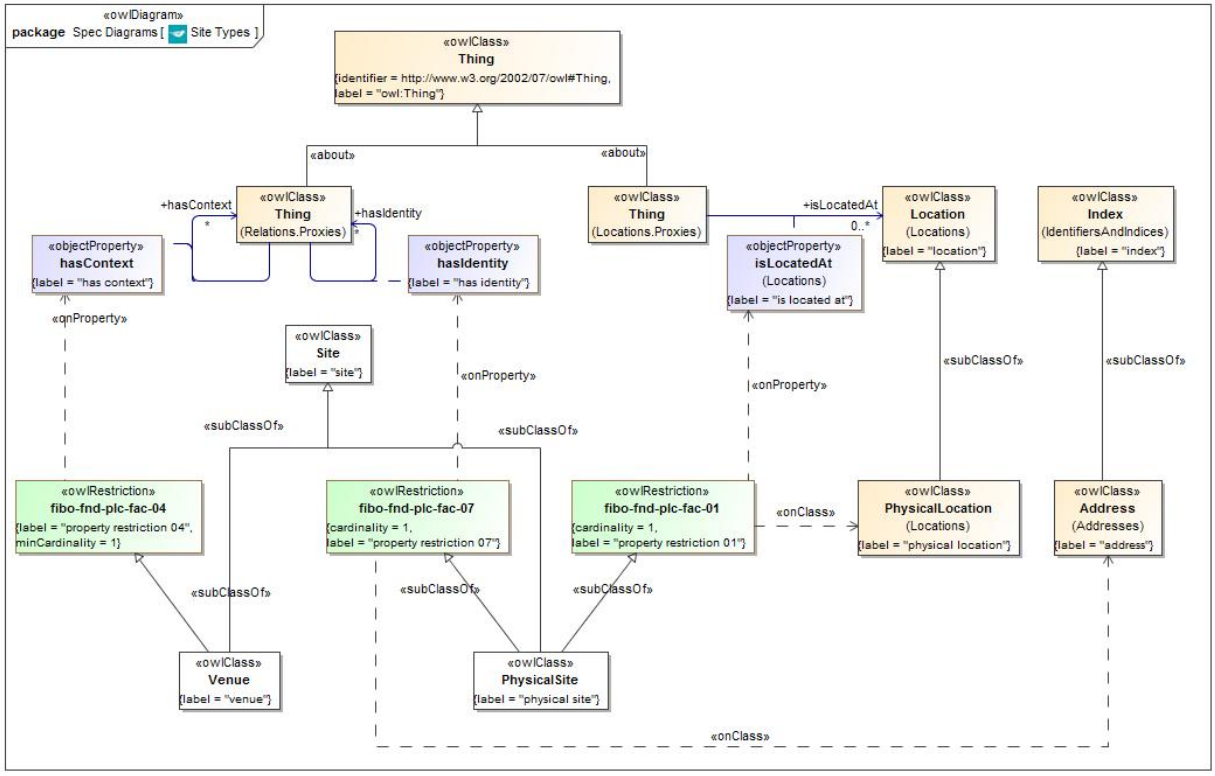

Figure 50.39 Site Types

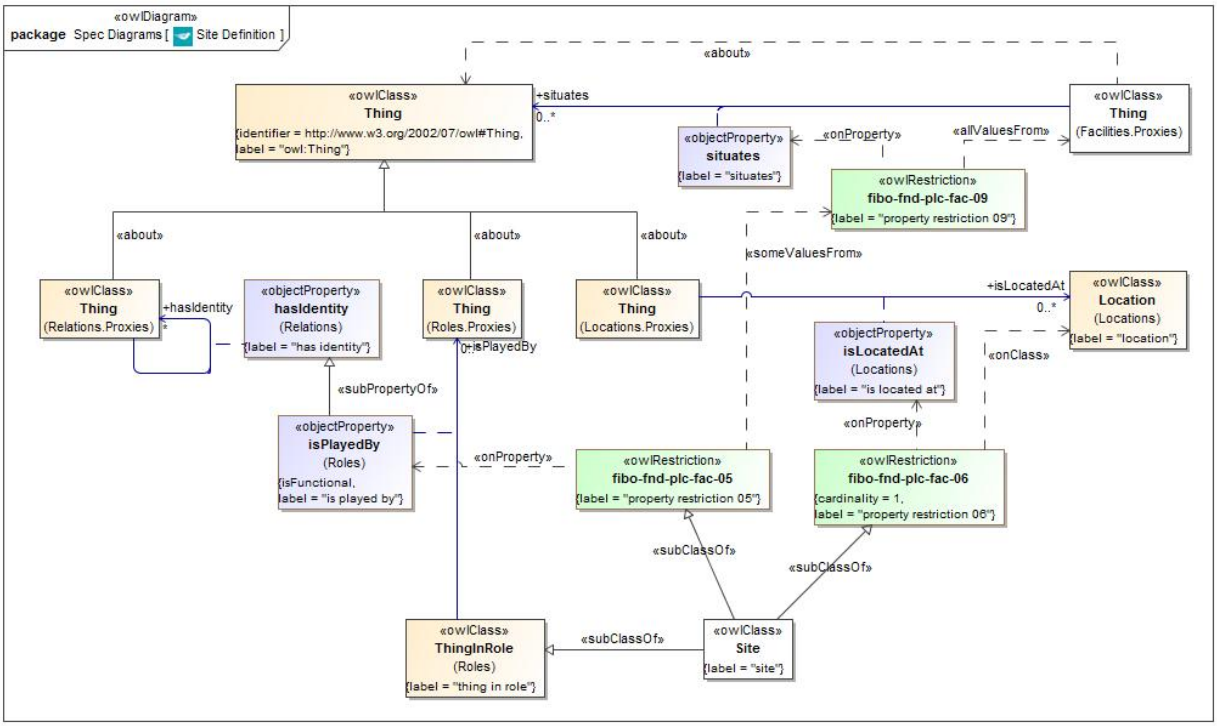

Figure 60.40 Site Definition

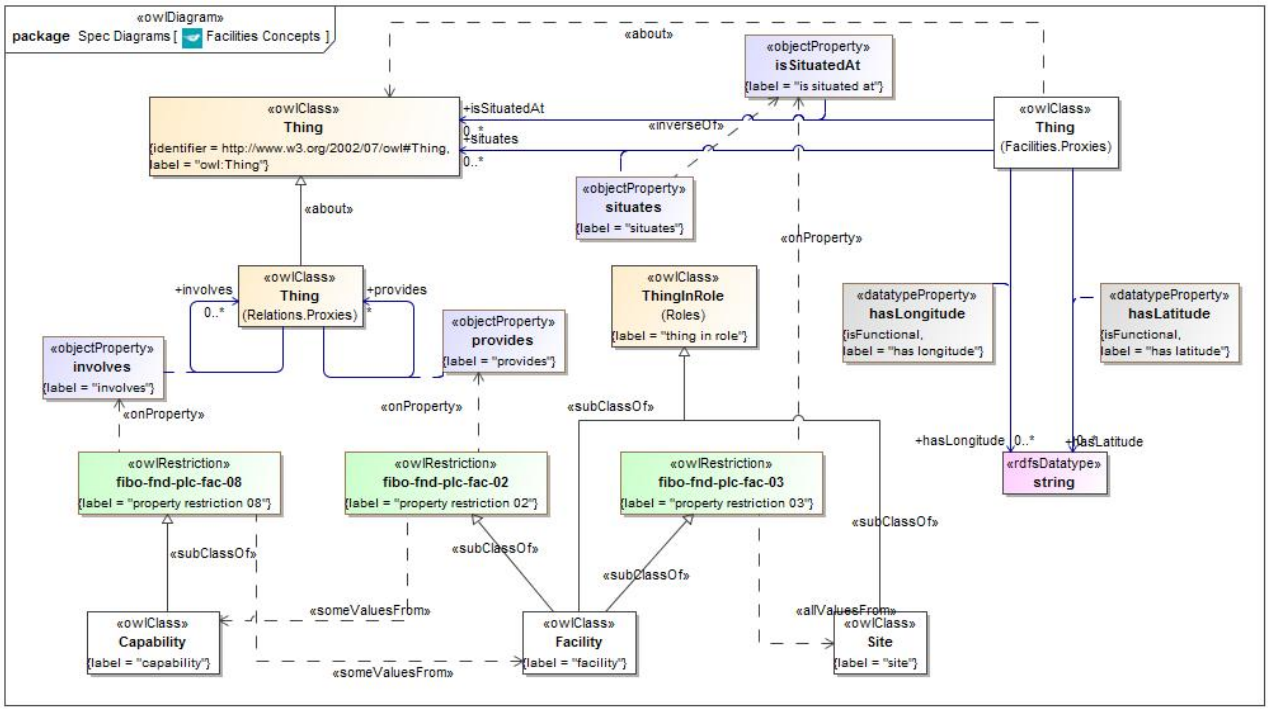

Figure 70.41 Facilities Concepts

Table 10-45. Facilities Ontology Metadata

Metadata Term	Value
sm:filename	Facilities
sm:fileAbbreviation	fibo-fnd-plc-fac
OntologyIRI	http://www.omg.org/spec/EDMC-FIBO/FND/Places/Facilities/
owl:versionIRI	http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Places/Facilities/
sm:dependsOn	http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/AnnotationVocabulary/ http://www.omg.org/spec/EDMC-FIBO/FND/Places/Locations/ http://www.omg.org/spec/EDMC-FIBO/FND/Relations/Relations/ http://www.omg.org/spec/EDMC-FIBO/FND/Parties/Roles/ http://www.omg.org/spec/EDMC-FIBO/FND/Places/Addresses/

Table 10-46. Facilities Concepts

Classes

Name	Annotations	Class Expressions
Facility (facility)	Definition: something that is built, contrived, established, or installed to serve a particular purpose, or make some course of action or operation easier, or provide some capability or service Explanatory Note: A facility may be concrete (as in a manufacturing facility) or abstract. Concrete facilities may be permanent, semi-permanent, or temporary structures, providing one or more capabilities at a given site. A single site may include multiple facilities and a given facility may span multiple sites.	Parent Class: ThingInRole Property Restriction: ∃ **provides.Capability** (fibo-fnd-plc-fac-02) Property Restriction: ∀ **isSituatedAt.Site** (fibo-fnd-plc-fac-03)
Venue (venue)	Definition: A place where something happens, described in the context of the event or activity that occurs there	Parent Class: Site Property Restriction: ≥ 1 **hasContext.** (fibo-fnd-plc-fac-04)
Capability (capability)	Definition: A capability represents the ability to perform a particular type of work and may involve people with particular skills and knowledge, intellectual property, defined practices, operating facilities, tools and equipment.	Property Restriction: ∃ **involves.Facility** (fibo-fnd-plc-fac-08)
PhysicalSite (physical site)	Definition: A physical site is a an actual location that situates something, typically a structure or building, archeological dig, landing location for an aircraft or spacecraft, etc. From biology, this could also be the site of a wound, and active site, and so forth. A physical site has certain characteristics that contribute to the context it provides, including area, shape, accessibility, and in the case of a geographic site, landforms, soil and ground conditions, climate, and so forth.	Parent Class: Site Property Restriction: = 1 **isLocatedAt.PhysicalLocation** (fibo-fnd-plc-fac-01) Property Restriction: = 1 **hasIdentity.Address** (fibo-fnd-plc-fac-07)
Site (site)	Definition: A site is a place, setting, or context in which something is situated.	Parent Class: ThingInRole

Name	Annotations	Class Expressions
		Property Restriction: ∃ **isPlayedBy.(∀ situates)** (fibo-fnd-plc-fac-05, fibo-fnd-plc-fac-09) Property Restriction: =1 **isLocatedAt.Location** fibo-fnd-plc-fac-06)

Properties

Name	Annotations	Property Axioms
hasLatitude **(has latitude)**	Definition: Latitude values indicate the angular distance between the Equator and points north or south of it on the surface of the Earth.	Range: string
hasLongitude **(has longitude)**	Definition: Lines of longitude, called meridians, run perpendicular to lines of latitude, and all pass through both poles. Longitude values indicate the angular distance between the Prime Meridian and points east or west of it on the surface of the Earth.	Range: string
isSituatedAt **(is situated at)**	Definition: indicates that something has been positioned, located or placed at some site, or in some setting, situation, or context	Inverse: situates
situates **(situates)**	Definition: indicates the place, setting, or context in which something is situated	Inverse: isSituatedAt

10.7.5 Ontology: VirtualPlaces

This ontology provides scaffolding for use in describing virtual location-oriented concepts.

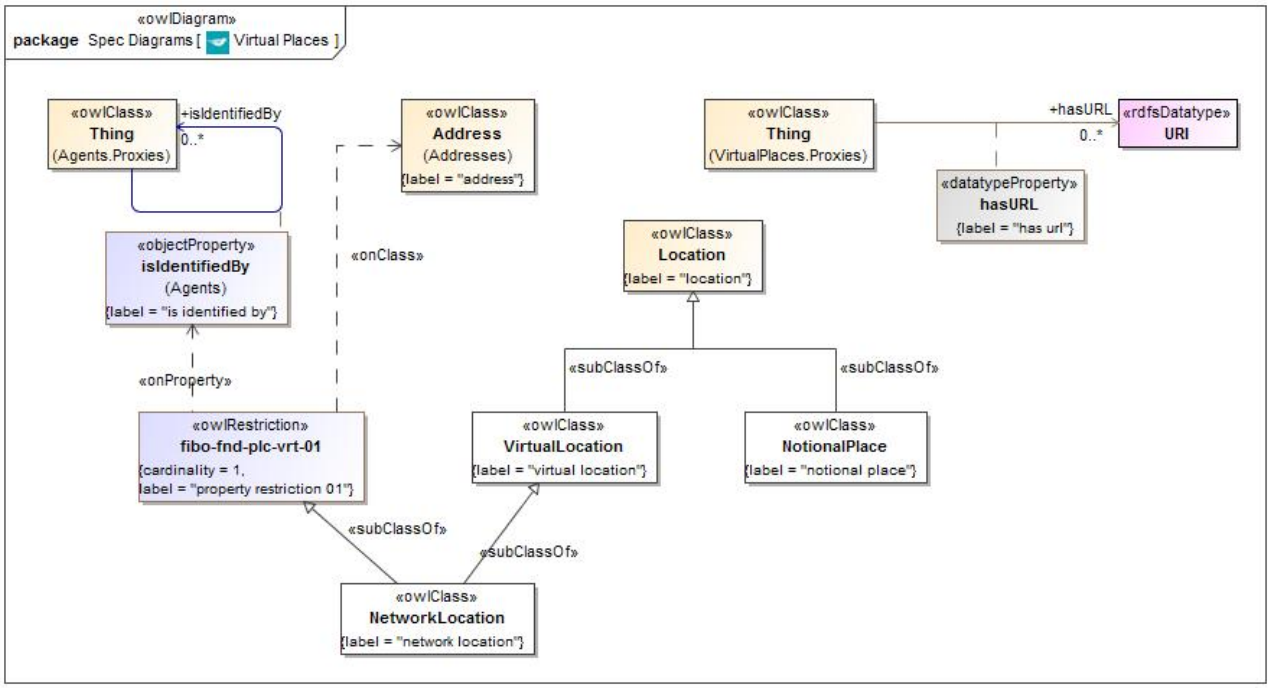

Figure 80.42 Virtual Places

Table 10-47. VirtualPlaces Ontology Metadata

Metadata Term	Value
sm:filename	VirtualPlaces
sm:fileAbbreviation	fibo-fnd-plc-vrt
OntologyIRI	http://www.omg.org/spec/EDMC-FIBO/FND/Places/VirtualPlaces/
owl:versionIRI	http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Places/VirtualPlaces/
sm:dependsOn	http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/AnnotationVocabulary/ http://www.omg.org/spec/EDMC-FIBO/FND/Places/Locations/ http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/BusinessFacingTypes/ http://www.omg.org/spec/EDMC-FIBO/FND/Relations/Relations/ http://www.omg.org/spec/EDMC-FIBO/FND/Places/Addresses/ http://www.omg.org/spec/EDMC-FIBO/FND/AgentsAndPeople/Agents/

Table 10-48. Virtual Places Concepts

Classes

Name	Annotations	Class Expressions
NotionalPlace (notional place)	Definition: A place described in terms of some abstract description or as a list of commonly understood concepts such as domestic, Eurozone etc.	Parent Class: Location
VirtualLocation (virtual location)	Definition: A place which has no physical location.	Parent Class: Location
NetworkLocation (network location)	Definition: A network address is location in a telecommunications network that may be identified by a network address (an identifier for a node or interface)	Parent Class: VirtualLocation Property Restriction: = 1 **isIdentifiedBy.Address** (fibo-fnd-plc-vrt-01)

Properties

Name	Annotations	Property Axioms
hasURL (has url)		Range: URI

10.8 Module: Organizations

Table 10-49. Organizations Module Metadata

Metadata Term	Value
sm:moduleName	Organizations
sm:moduleAbbreviation	FIBO-FND-ORG
sm:moduleVersion	1.0
sm:moduleAbstract	This module includes several ontologies defining organizations, features of an organization and different types of organization. These include formal versus informal organizations, legitimate and illicit organizations and so on. They are purposefully underspecified to facilitate mapping to specific organization ontologies, such as the emerging W3C organization and formal organization ontologies, organization from a BMM or BPMN perspective, organization from a records management (RMS) perspective, and so forth.

10.8.1 Ontology: Organizations

This ontology defines high-level concepts for organizations and related terms, for use in other FIBO ontology elements. It is purposefully underspecified to facilitate mapping to specific organization ontologies, such as the emerging W3C organization ontology, organization from a BMM or BPMN perspective, organization from a records management (RMS) perspective, and so forth.

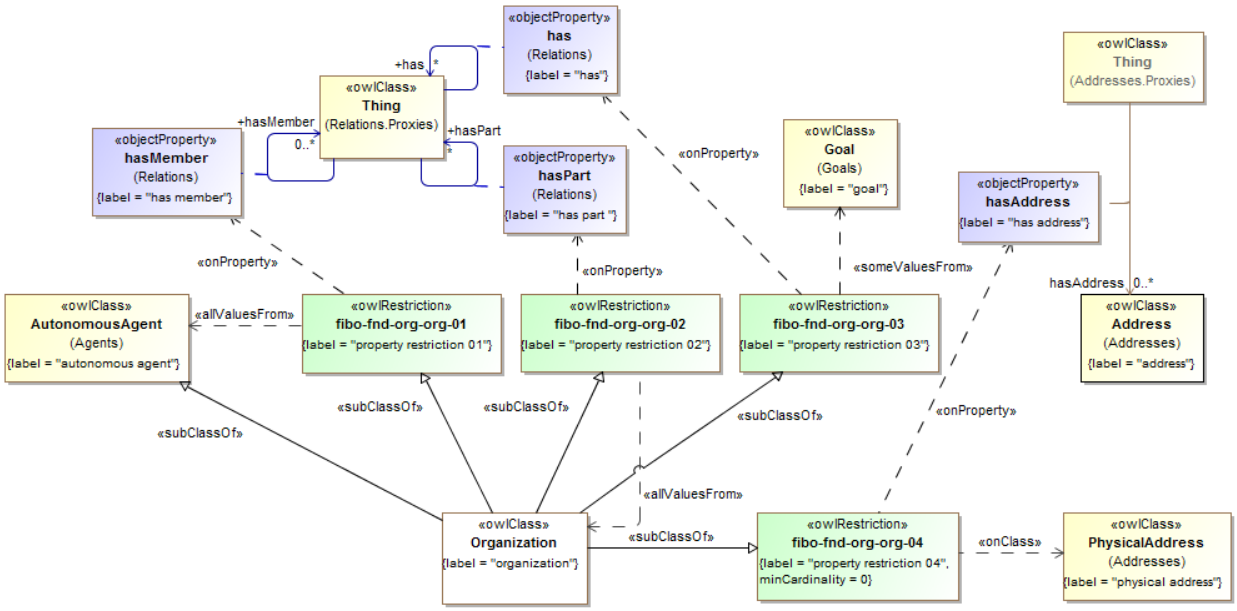

Figure 10.43 Organizations Concepts

Table 10-50. Organizations Ontology Metadata

Metadata Term	Value
sm:filename	Organizations Ontology
sm:fileAbbreviation	fibo-fnd-org-org
OntologyIRI	http://www.omg.org/spec/EDMC-FIBO/FND/Organizations/Organizations/
owl:versionIRI	http://www.omg.org/spec/EDMC-FIBO/FND/20160201/Organizations/Organizations/
sm:dependsOn	http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/AnnotationVocabulary/ http://www.omg.org/spec/EDMC-FIBO/FND/Relations/Relations/ http://www.omg.org/spec/EDMC-FIBO/FND/AgentsAndPeople/Agents/ http://www.omg.org/spec/EDMC-FIBO/FND/Places/Addresses/ http://www.omg.org/spec/EDMC-FIBO/FND/GoalsAndObjectives/Goals/

Table 10-51. Organizations Details

Classes

Name	Annotations	Class Expressions
Organization (organization)	Definition: a social unit of people, systematically structured and managed to meet a need or pursue collective goals on a continuing basis Example: This may be a business entity or a government, international or non profit organization. Definition origin: http://www.BusinessDictionary.com/	Parent Class: AutonomousAgent Property Restriction: ∀ **hasMember**.**AutonomousAgent** (fibo-fnd-org-org-01) Property Restriction: ∀ **hasPart**.**Organization** (fibo-fnd-org-org-02) Property Restriction: ∃ **has**.**Goal** (fibo-fnd-org-org-03) Property Restriction: ≥ 0 **hasAddress**.**PhysicalAddress** (fibo-fnd-org-org-04)

10.8.2 Ontology: Formal Organizations

This ontology defines the high level concept of formal organization for use in other FIBO ontology elements. It is purposefully underspecified to facilitate mapping to other formal organization ontologies, such as the emerging W3C formal organization ontology, or others defined for specific business and financial services standards. The concepts in this ontology extend those in the Organizations ontology.

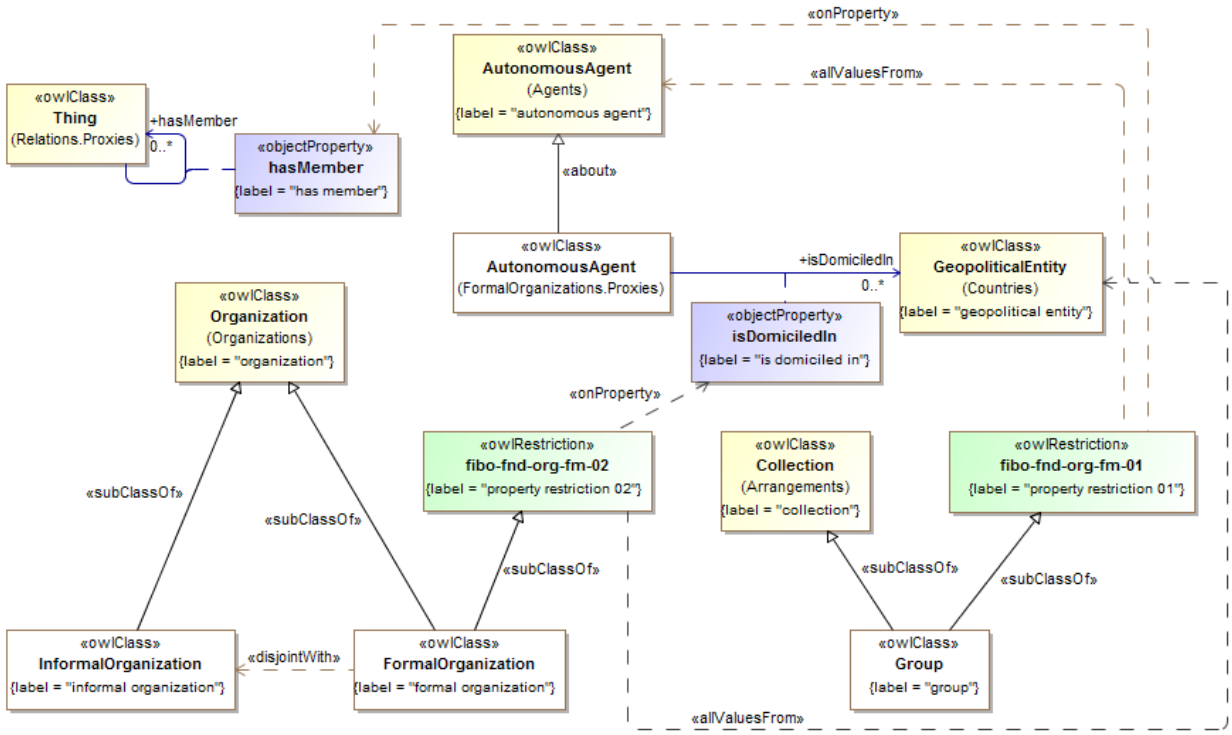

Figure 10.44 Formal Organizations Concepts

Table 10-52. Formal Organizations Ontology Metadata

Metadata Term	Value
sm:filename	Formal Organizations Ontology
sm:fileAbbreviation	fibo-fnd-org-fm
OntologyIRI	http://www.omg.org/spec/EDMC-FIBO/FND/Organizations/FormalOrganizations/
owl:versionIRI	http://www.omg.org/spec/EDMC-FIBO/FND/20160201/Organizations/FormalOrganizations/

Metadata Term	Value
sm:dependsOn	http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/AnnotationVocabulary/ http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/BusinessFacingTypes/ http://www.omg.org/spec/EDMC-FIBO/FND/Relations/Relations/ http://www.omg.org/spec/EDMC-FIBO/FND/AgentsAndPeople/Agents/ http://www.omg.org/spec/EDMC-FIBO/FND/Places/Locations/ http://www.omg.org/spec/EDMC-FIBO/FND/Places/Countries/ http://www.omg.org/spec/EDMC-FIBO/FND/Places/Addresses/ http://www.omg.org/spec/EDMC-FIBO/FND/GoalsAndObjectives/Goals/ http://www.omg.org/spec/EDMC-FIBO/FND/Organizations/Organizations/

Table 10-53. Formal Organizations Details

Classes

Name	Annotations	Class Expressions
FormalOrganization (formal organization)	Definition: an organization that is recognized in some legal jurisdiction, with associated rights and responsibilities Example: Examples include a Corporation, Charity, Government or Church. Adapted from: W3C Organization Ontology	Parent Class: Organization Class Axiom: ¬ InformalOrganization Property Restriction: ∀ **isDomiciledIn.GeopoliticalEntity** (fibo-fnd-org-fm-02)
Group (group)	Definition: a collection of autonomous entities	Parent Class: Collection Property Restriction: ∀ **hasMember.AutonomousAgent)** (fibo-fnd-org-fm-01)
InformalOrganization (informal organization)	Definition: An organization which is not formally constituted in some way.	Parent Class: Organization

Properties

Name	Annotations	Property Axioms
isDomiciledIn (is domiciled in)	Definition: identifies the permanent home or principal establishment of an individual or organization Explanatory note: Note that residence is not the same as domicile, as a person or organization can have many transient residences but only one legal domicile. The domicile of a formal organization is the address (location) where the establishment is maintained or where the governing power of the organization is exercised. Adapted from: Barron's Dictionary of Business and Economics Terms, Fifth Edition, 2012. Adapted from: http://thelawdictionary.org/domicile/	Domain: AutonomousAgent Range: GeopoliticalEntity

10.8.3 Ontology: Legitimate Organizations

This ontology defines the concepts of legitimate and illicit organizations for use in other FIBO ontology elements. These distinctions are provided in order to facilitate modeling of concepts relevant to money laundering. Legitimate organizations such as clubs are defined. These, along with the distinctions of formal versus informal organizations, provide the universe of possible kinds of organizations which may perform specific roles such as holding shares, having control of assets of companies and so on.

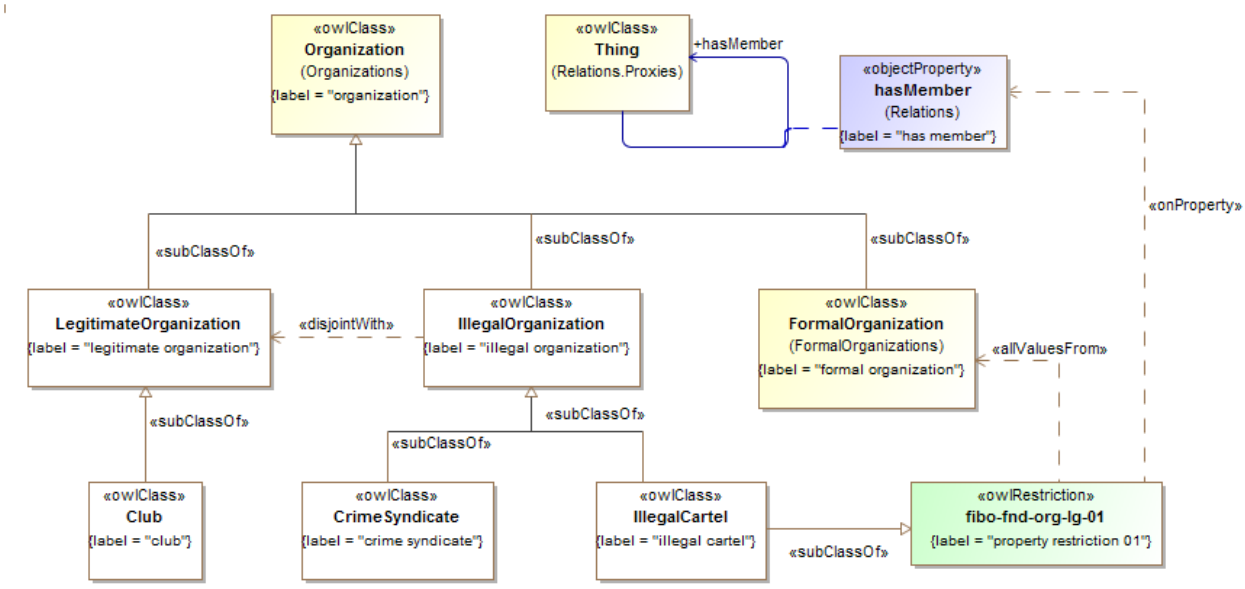

Figure 10.45 Legitimate and Illicit Organizations Concepts

Table 10-54. Legitimate Organizations Ontology Metadata

Metadata Term	Value
sm:filename	Legitimate Organizations Ontology
sm:fileAbbreviation	fibo-fnd-org-lg
OntologyIRI	http://www.omg.org/spec/EDMC-FIBO/FND/Organizations/LegitimateOrganizations/
owl:versionIRI	http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Organizations/LegitimateOrganizations/
sm:dependsOn	http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/AnnotationVocabulary/ http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/BusinessFacingTypes/ http://www.omg.org/spec/EDMC-FIBO/FND/Relations/Relations/ http://www.omg.org/spec/EDMC-FIBO/FND/AgentsAndPeople/Agents/ http://www.omg.org/spec/EDMC-FIBO/FND/Places/Locations/ http://www.omg.org/spec/EDMC-FIBO/FND/Places/Countries/ http://www.omg.org/spec/EDMC-FIBO/FND/Places/Addresses/ http://www.omg.org/spec/EDMC-FIBO/FND/GoalsAndObjectives/Goals/ http://www.omg.org/spec/EDMC-FIBO/FND/Organizations/Organizations/ http://www.omg.org/spec/EDMC-FIBO/FND/Organizations/FormalOrganizations/

Table 10-55. Legitimate and Illicit Organizations Details

Classes

Name	Annotations	Class Expressions
Club (club)	Definition: An informal organization formed to pursue some common interest among its members	Parent Class: LegitimateOrganization
CrimeSyndicate (crime syndicate)	Definition: An informal grouping formed for the purposes of organized criminal activities	Parent Class: IllegalOrganization
IllegalCartel (illegal cartel)	Definition: A collection of companies that come together to manipulate the market in some way, e.g. price fixing	Parent Class: IllegalOrganization Property Restriction: ∀ **hasMember.FormalOrganization** (fibo-fnd-org-lg-01)

Name	Annotations	Class Expressions
IllegalOrganization (illegal organization)	Definition: A kind of organization which has been set up specifically to perform illegal acts or has become such Explanatory Note: This is not to do with performing illicit acts. We can narrow down on a definition for Illicit Organization - one which has been set up specifically to perform illicit acts or has become such. This relates to the purpose of the organization, and the purposes of the entities which control that entity. And the acts which the entity may perform. (definition adopted from the above note, with Illicit changed to Illegal for clarity). Typically, a money laundering entity may perform (will perform) legal acts and is explicitly set up for such, but will also perform illicit acts. The definition of illicit is framed entirely with respect to law and not morality.	Parent Class: Organization Class Axiom: ¬ **LegitimateOrganization**
LegitimateOrganization (legitimate organization)	Definition: An organization that exists to serve some lawful purpose	Parent Class: Organization Class Axiom: ¬ **IllegalOrganization**

10.9 Module: Agreements

Table 10-56. Agreements Module Metadata

Metadata Term	Value
sm:moduleName	Agreements
sm:moduleAbbreviation	FIBO-FND-AGR
sm:moduleVersion	1.0
sm:moduleAbstract	This module includes ontologies describing agreements between parties and contracts that formalize those agreements. These cover written and verbal contracts, including contracts which may be transferred from one party to another. The latter form the basis for financial securities contracts. The Contracts ontology also describes fundamental properties of contracts such as contractual terms, contract parties and so on, many of which form the basis for more specialized financial industry concepts such as interest payment terms, bond issuers and so on.

10.9.1 Ontology: Agreements

This ontology defines concepts for agreements, for use in other ontology elements. Agreements as defined here are the actual agreements between parties, and this ontology is intended to be referred to in conjunction with the contracts ontology which defines the actual contracts which formalize such agreements. The concepts of agreement and contract are intended to be kept distinct in the FIBO ontologies, that is neither is intended to be regarded as a sub type of the other.

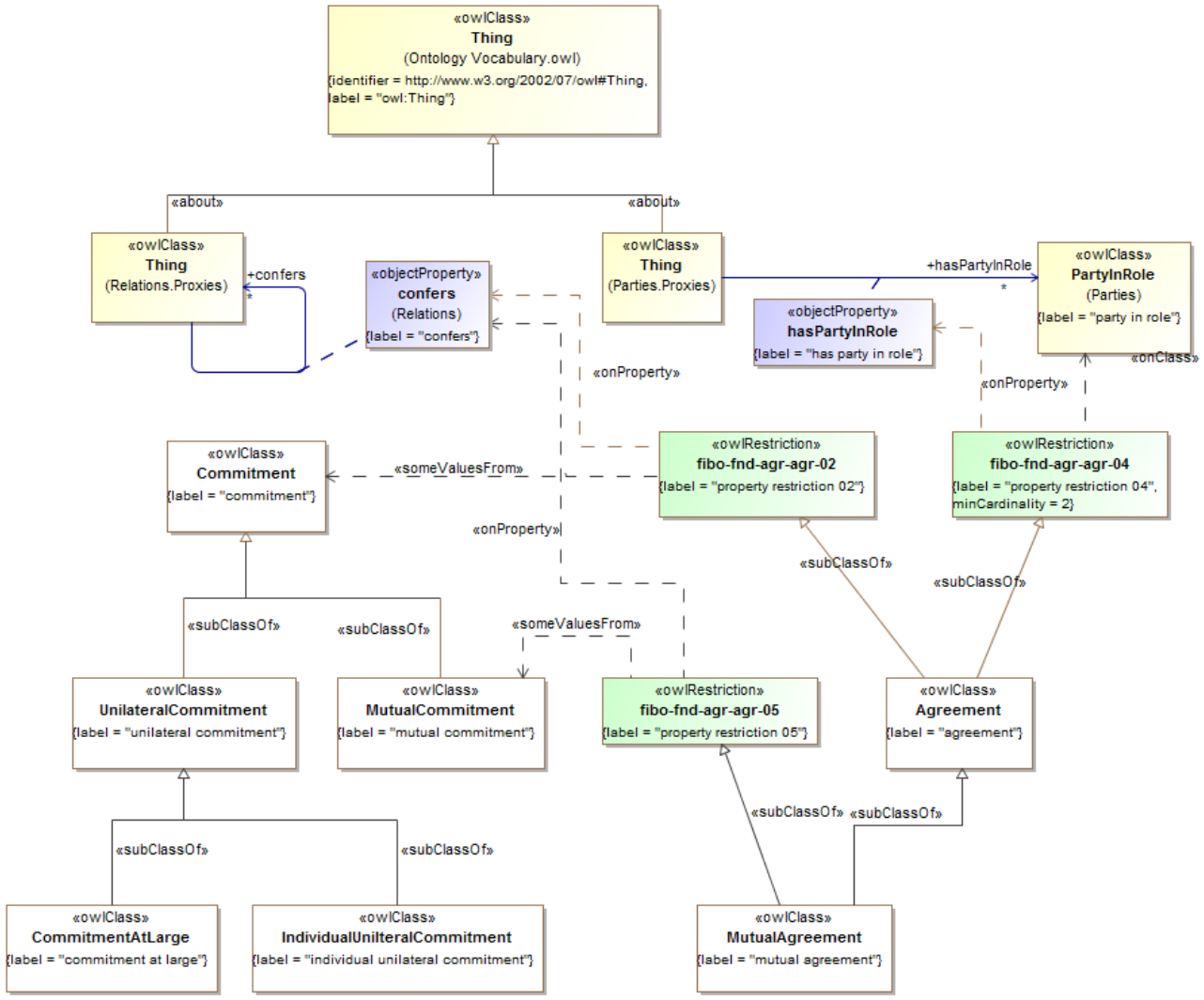

Figure 10.46 Agreements Concepts

Table 10-57. Agreements Ontology Metadata

Metadata Term	Value
sm:filename	Agreements Ontology
sm:fileAbbreviation	fibo-fnd-agr-agr
OntologyIRI	http://www.omg.org/spec/EDMC-FIBO/FND/Agreements/Agreements/
owl:versionIRI	http://www.omg.org/spec/EDMC-FIBO/FND/20160201/Agreements/Agreements/
sm:dependsOn	http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/AnnotationVocabulary/

Metadata Term	Value
	http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/BusinessFacingTypes/ http://www.omg.org/spec/EDMC-FIBO/FND/Relations/Relations/ http://www.omg.org/spec/EDMC-FIBO/FND/AgentsAndPeople/Agents/ http://www.omg.org/spec/EDMC-FIBO/FND/Places/Locations/ http://www.omg.org/spec/EDMC-FIBO/FND/Places/Countries/ http://www.omg.org/spec/EDMC-FIBO/FND/Places/Addresses/ http://www.omg.org/spec/EDMC-FIBO/FND/GoalsAndObjectives/Goals/ http://www.omg.org/spec/EDMC-FIBO/FND/Organizations/Organizations/ http://www.omg.org/spec/EDMC-FIBO/FND/Organizations/FormalOrganizations/ http://www.omg.org/spec/EDMC-FIBO/FND/AgentsAndPeople/People/ http://www.omg.org/spec/EDMC-FIBO/FND/Parties/Roles/ http://www.omg.org/spec/EDMC-FIBO/FND/Parties/Parties/

Table 10-58. Agreements Details

Classes

Name	Annotations	Class Expressions
Agreement (agreement)	Definition: A negotiated and usually legally enforceable understanding between two or more legally competent parties. Explanatory Note: Although a binding contract can (and often does) result from an agreement, an agreement typically documents the give-and-take of a negotiated settlement and a contract specifies the minimum acceptable standard of performance. Explanatory Note: An agreement provides language that defines the terms and conditions of a legally binding contract among the identified parties, ordinarily leading to a contract. Explanatory Note: An agreement may be formalized in the form of a Contract or other formal instrument, or it may not. In either case, the agreement is that which may be referred to as the agreement between or among the parties, and the contract is framed as defining (and usually as exclusively defining) the agreement between two parties. Definition Origin: http://www.businessdictionary.com/definition/agreement.html	Property Restriction: ∃ **confers.Commitment** (fibo-fnd-agr-agr-02) Property Restriction: = 2 **hasPartyInRole.PartyInRole** (fibo-fnd-agr-agr-04)
Commitment (commitment)	Definition: A legal construct which represents the undertaking on the part of some party to act or refrain from acting in some manner. Editorial Note: The undertaking by some party to act or refrain from acting results in an obligation on the part of that party, and usually results in the existence of some corresponding right on the party of some other party, in the event that the commitment is to such party. Thus Obligations and Rights are considered as reciprocal aspects of this	

Name	Annotations	Class Expressions
	Commitment concept.	
CommitmentAtLarge (commitment at large)	Definition: a commitment made by some party without direct involvement from the potential beneficiaries of that commitment Scope Note: Forms the basis for negotiable securities including transferable contracts and potentially other types of agreement such as software licenses.	Parent Class: UnilateralCommitment
IndividualUnilateralCommitment (Individual unilateral commitment)	Definition: a commitment made by some party unilaterally to another specific party	Parent Class: UnilateralCommitment
MutualAgreement (mutual agreement)	Definition: an agreement between two or more specific named parties. The rights and obligations pertaining to either party cannot be transferred to another party without prior agreement Scope Note: This may or may not be a contractual agreement - it also forms the basis of REA transaction models which may or may not refer to contractual agreements, since REA is also used to frame transactions internal to an individual organization.	Parent Class: Agreement Property Restriction: ∃ **confers.MutualCommitment** (fibo-fnd-agr-agr-05)
MutualCommitment (mutual commitment)	Definition: A commitment between two or more parties	Parent Class: Commitment
UnilateralCommitment (unilateral commitment)	Definition: A commitment made by one party without reference to the party to which the commitment is made.	Parent Class: Commitment

10.9.2 Ontology: Contracts

This ontology defines concepts relating to contracts, for use in other FIBO ontology elements. These include written contracts which are the concrete evidence of agreements between parties, along with verbal contracts. Contracts are further broken down into bilateral and transferable contracts, the latter being the basis for most financial instruments. Properties of contracts are also defined, in particular contractual terms and contract parties. These concepts all form the basis of concepts in the financial services industry, for example interest payment terms are a kind of contract terms set, and security holders are a kind of contract counterparty.

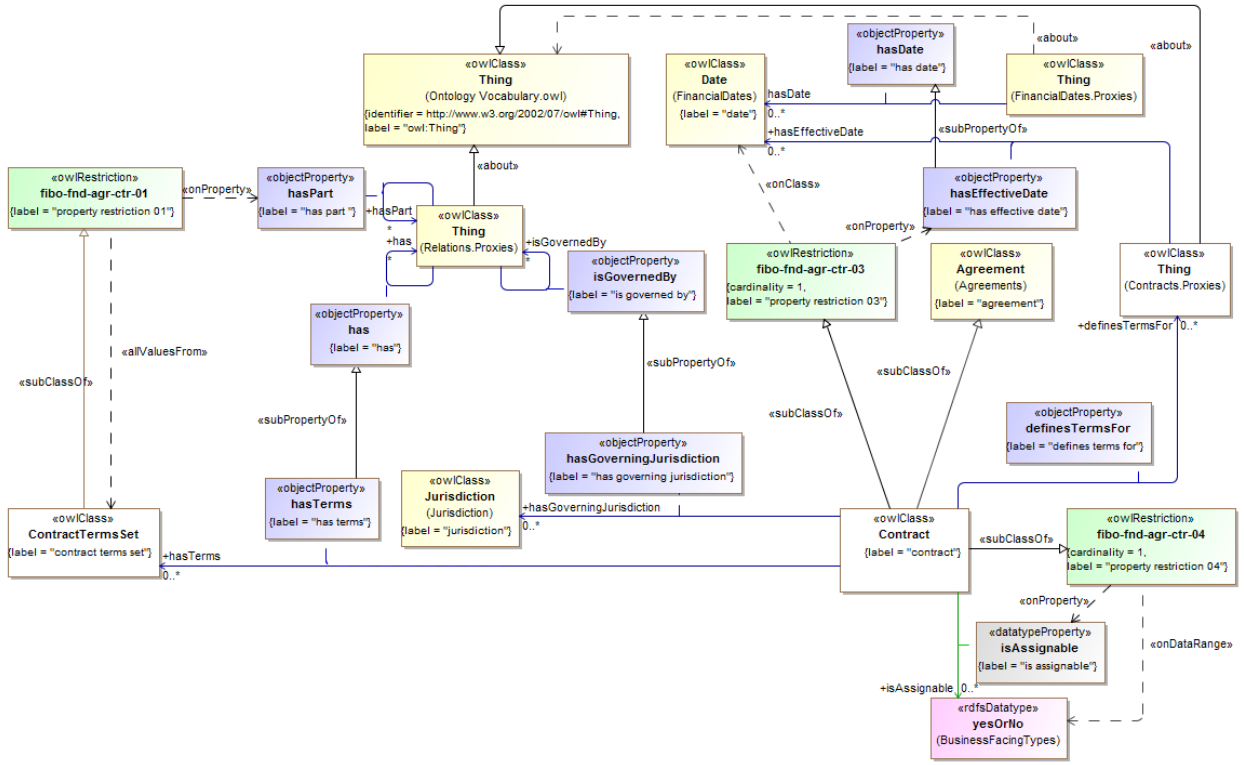

Figure 10.47 Contracts Basic Concepts

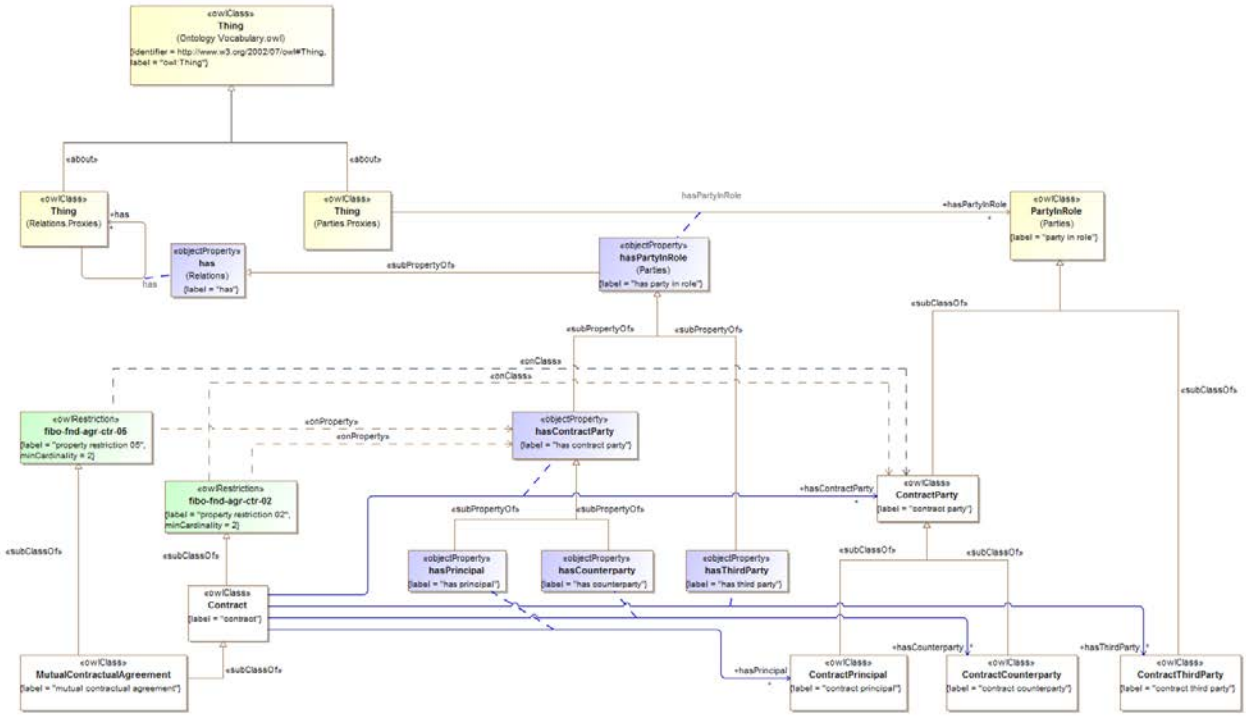

Figure 10.48 Contract Parties

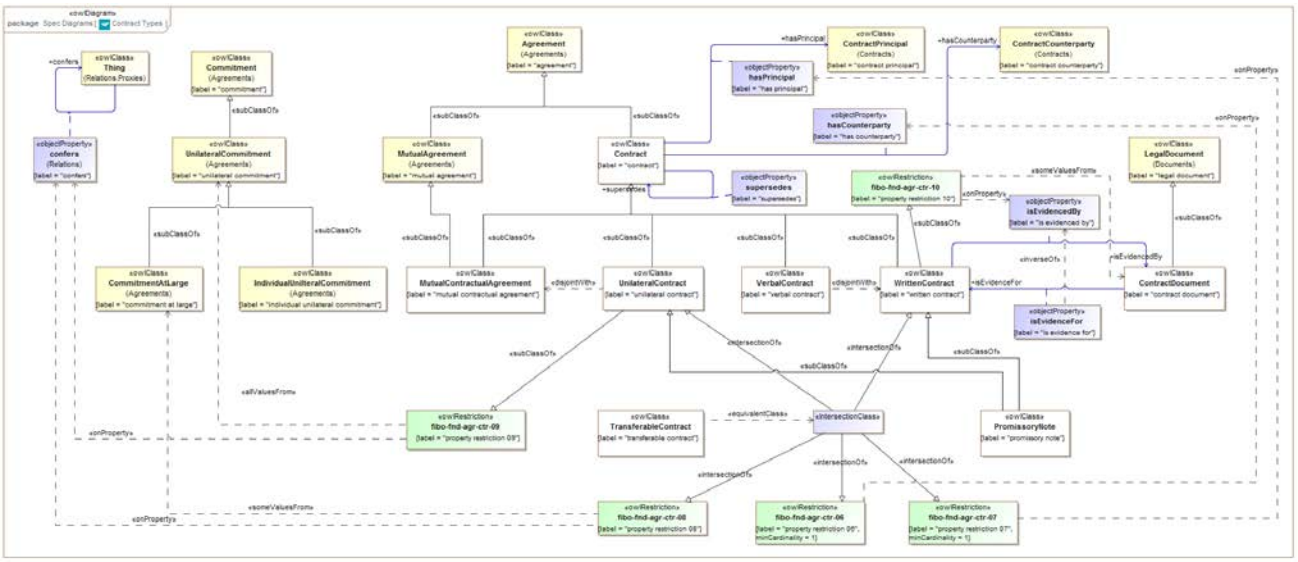

Figure 10.49 Contract Types

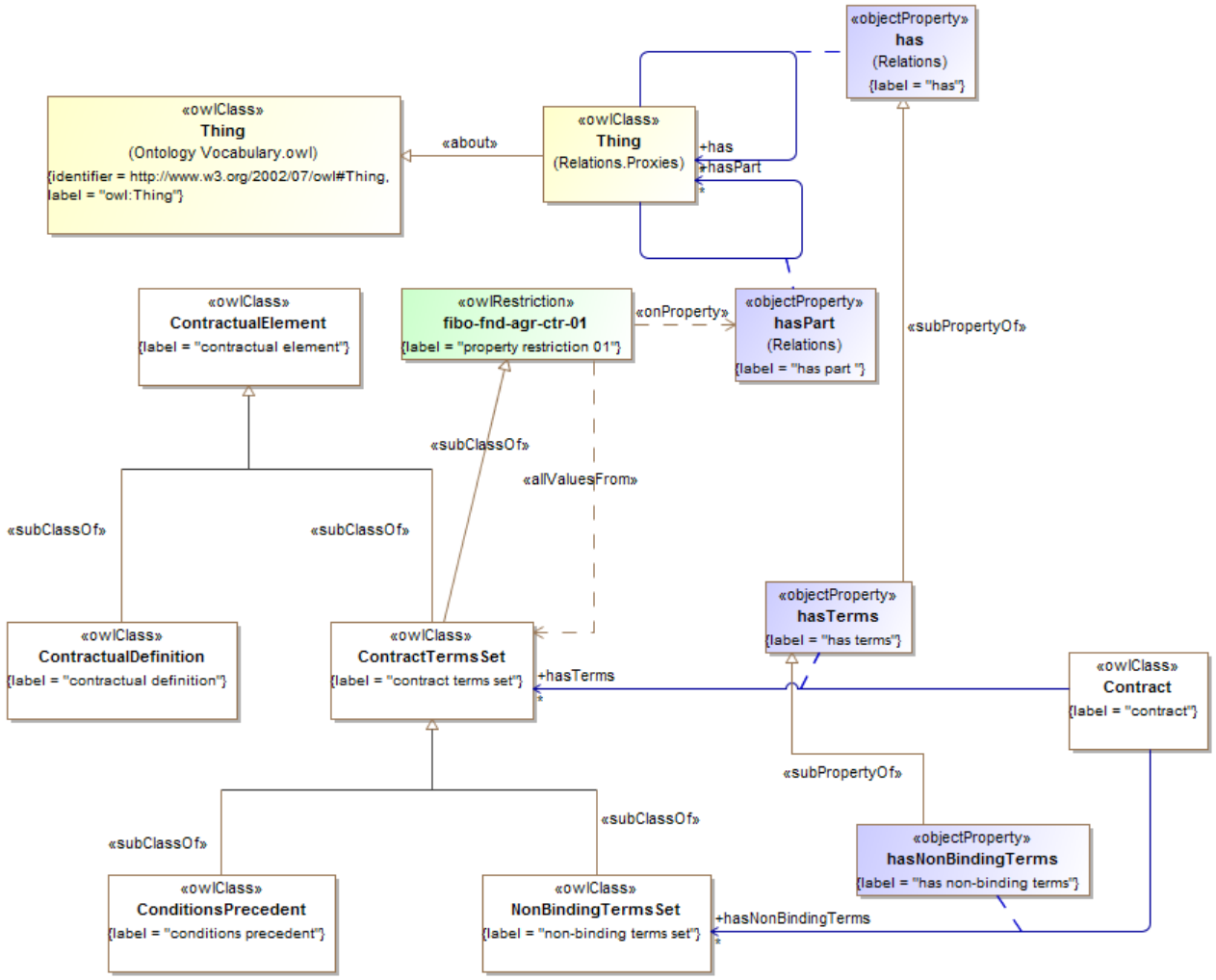

Figure 10.50 Contract Terms and Elements

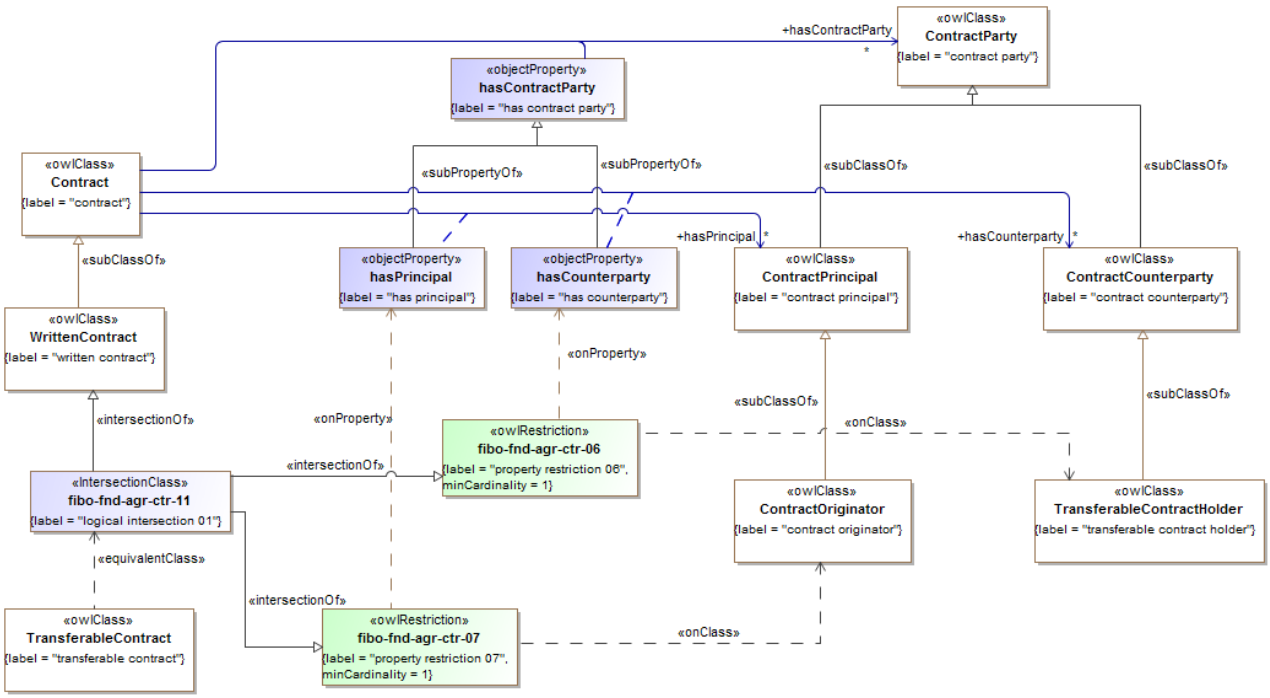

Figure 10:51 Transferable Contract parties

Table 10-59. Contracts Ontology Metadata

Metadata Term	Value
sm:filename	Contracts Ontology
sm:fileAbbreviation	fibo-fnd-agr-ctr
OntologyIRI	http://www.omg.org/spec/EDMC-FIBO/FND/Agreements/Contracts/
owl:versionIRI	http://www.omg.org/spec/EDMC-FIBO/FND/20160201/Agreements/Contracts/
sm:dependsOn	http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/AnnotationVocabulary/ http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/BusinessFacingTypes/ http://www.omg.org/spec/EDMC-FIBO/FND/Relations/Relations/ http://www.omg.org/spec/EDMC-FIBO/FND/Arrangements/Documents/ http://www.omg.org/spec/EDMC-FIBO/FND/DatesAndTimes/FinancialDates/ http://www.omg.org/spec/EDMC-FIBO/FND/Parties/Parties/ http://www.omg.org/spec/EDMC-FIBO/FND/Law/LegalCore/ http://www.omg.org/spec/EDMC-

Metadata Term	Value
	FIBO/FND/Law/Jurisdiction/ http://www.omg.org/spec/EDMC-FIBO/FND/Agreements/Agreements/

Table 10-60. Contracts Details

Classes

Name	Annotations	Class Expressions
ConditionsPrecedent (conditions precedent)	Definition: Conditions precedent on some obligation. These are conditions which would alter the Obligation as it is otherwise stated. Adapted from: https://en.wikipedia.org/wiki/Adult	Parent Class: ContractTermsSet
Contract (contract)	Definition: a voluntary, deliberate, and legally binding agreement between two or more competent parties Explanatory note: Contracts are usually written but may be spoken or implied, and generally have to do with employment, sale or lease, or tenancy. Definition origin: http://www.businessdictionary.com/definition/contract.html	Parent Class: Agreement Property Restriction: ≥ 2 **hasContractParty**.**ContractParty** (fibo-fnd-aap-ppl-01) Property Restriction: = 1 **isAssignable**.**yesOrNo** (fibo-fnd-aap-ppl-01) Property Restriction: = 1 **hasEffectiveDate**.**Date** (fibo-fnd-aap-ppl-01)
ContractCounterparty (contract counterparty)	Definition: in the event that a contract identifies either party to that contract as being the principal, this is the other party to that contract Explanatory note: Where no party is identified as the principal to a contract, both or all parties are simply identified as being parties to the contract. That is, the concept 'counterparty' as defined here is specifically in opposition to 'principal'; other usages of the word counterparty exist.	Parent Class: ContractParty
ContractDocument (contract document)	Definition: a written document, whether physical or electronic, which sets out the formal terms and conditions of some written contract Scope note: Written here does not necessarily mean a paper document but includes situations in which the contract is expressed electronically, whether as an electronic representation of a formal document such as in PDF form or as an electronic message, provided in the latter case that the message is expressly given formal contractual standing, for example as indicated in a separate cover-	Parent Class: LegalDocument

Name	Annotations	Class Expressions
	ing agreement between the parties.	
ContractOriginator (contract originator)	Definition: the party that originates the contract and acts as the principal in that contract regardless of the owner or counterparty	Parent Class: ContractPrincipal
ContractParty (contract party)	Definition: a party to the contract, that is a contractually capable person or organization which is a signatory to the contract, and which grants or concedes certain rights and obligations as defined in the contract	Parent Class: PartyInRole
ContractPrincipal (contract principal)	Definition: the party identified as being the principal or first party to a contract, in the event that the contract distinguishes any party as the principal Explanatory note: In law, the principal is the party that has the primary responsibility in a liability or obligation, as opposed to an endorser, guarantor, or surety. Definition origin: http://www.businessdictionary.com/definition/principal.html	Parent Class: ContractParty
ContractTermsSet (contract terms set)	Definition: the terms and conditions that set the rights and obligations of the contracting parties when a contract is awarded or entered into Scope note: These include general conditions which are common to all types of contracts, such as general and special arrangements, provisions, requirements, rules, specifications, and standards that form an integral part of an agreement or contract, as well as special conditions which are peculiar to a specific contract (such as, contract change conditions, payment conditions, price variation clauses, penalties). Definition origin: http://www.businessdictionary.com/definition/conditions-of-contract.html Definition origin: http://www.businessdictionary.com/definition/terms-and-conditions.html	Parent Class: ContractualElement Property Restriction: ∀ **hasPart**.**ContractTermsSet** (fibo-fnd-aap-ppl-01)
ContractThirdParty (contract third party)	Definition: someone who may be indirectly involved but is not a principal party to an arrangement, contract, deal, lawsuit, or transaction Definition origin: http://www.businessdictionary.com/definition/third-party.html	Parent Class: PartyInRole
ContractualDefinition (contractual definition)	Definition: a contractual element that defines something in a contract or other legal instrument	Parent Class: ContractualElement
ContractualElement (contractual element)	Definition: any element of a contract	
MutualContractualAgreement (mutual contractual agreement,	Definition: a contract between two or more specific named parties; the rights and obligations pertaining to either party cannot be transferred to another party without prior written permission or a	Parent Class: MutualAgreement

Name	Annotations	Class Expressions
bilateral contract)	change to the contract itself	Parent Class: Contract Property Restriction: ≥ 2 **hasContractParty**.**ContractParty** (fibo-fnd-aap-ppl-01) Class Axiom: ¬ UnilateralContract
NonBindingTermsSet (non-binding terms set)	Definition: terms that do not have binding legal standing on the issuer or holder	Parent Class: ContractTermsSet
PromissoryNote (promissory note)	Definition: a promissory note is a written, signed, unconditional, and unsecured promise by one party (the maker or promisor) to another (the payee or promisee) that commits the maker to pay a specified sum on demand, or on a fixed or a determinable date Explanatory note: Promissory notes (such as bank or currency notes) are negotiable instruments. Definition origin: http://www.businessdictionary.com/definition/promissory-note.html	Parent Class: UnilateralContract Parent Class: WrittenContract
TransferableContract (transferable contract)	Definition: a contract in which the rights and obligations of one party (the holder) may be transferred to another party, which thereby takes on the same rights and obligations with respect to the other party to the contract Explanatory note: Note that the ability to transfer ownership of one side of a contract, and the concept of assignability, are distinct. In one case the contract may be freely traded; in the other case, some legal transfer of rights to a third party takes place, without a change in who are the signatories of a (typically bilateral) contract. Definition origin: http://www.edmcouncil.org/	Class Axiom: ≡ (UnilateralContract ∪ WrittenContract ∪ (∃ **confers.CommitmentAtLarge**) ∪ (≥ 1 **hasCounterparty.TransferableContractHolder**) ∪ (≥ 1 **hasPrincipal.ContractOriginator**))
TransferableContractHolder (transferable contract holder)	Definition: a party that holds a transferable contract and enjoys the benefits defined in that contract while they hold it	Parent Class: ContractCounterparty
UnilateralContract (unilateral contract)	Definition: a contract in which only one party makes an express promise, or undertakes a performance without first securing a reciprocal agreement from the other party Explanatory note: In a unilateral, or one-sided, contract, one party, known as the offeror, makes a promise in exchange for an act (or abstention from acting) by another party, known as the offeree. If the offeree acts on the offeror's promise, the offeror is legally obligated to fulfill the contract, but an offeree cannot be forced to act (or not act), because no return promise has been made to the offeror. After an offeree has performed, only one enforceable promise exists, that of the offeror.	Parent Class: Contract Property Restriction: ∀ **confers**.**UnilateralCommitment** (fibo-fnd-aap-ppl-01)

Name	Annotations	Class Expressions
	A unilateral contract differs from a Bilateral Contract, in which the parties exchange mutual promises. Bilateral contracts are commonly used in business transactions; a sale of goods is a type of bilateral contract. Definition origin: http://legal-dictionary.thefreedictionary.com/Unilateral+contract	
VerbalContract (verbal contract)	Definition: a contract that exists as a result of some verbal exchange	Parent Class: Contract Class Axiom: ¬ WrittenContract
WrittenContract (written contract)	Definition: a formal contract that is written and signed by the parties thereto	Parent Class: Contract Property Restriction: ∃ **isEvidencedBy**.**ContractDocument** (fibo-fnd-aap-ppl-01)

Properties

Name	Annotations	Property Axioms
hasCitizenship (has citizenship)	Definition: links a person to their country of citizenship	Parent Property: has Domain: Person Range: Country
hasDateOfBirth (has date of birth)	Definition: links a person with their date of birth	Parent Property: hasDate Type: FunctionalProperty Domain: Person Range: Date
hasPlaceOfBirth (has place of birth)	Definition: links a person with their place of birth	Parent Property: has Domain: Person Range: PhysicalLocation
verifiesAddress (verifies address)	Definition: verifies a person's address as recorded in an identity document	Parent Property: hasAddress Domain: Person

Name	Annotations	Property Axioms
		Range: PhysicalAddress
hasFamilyName (has family name)	Definition: the patronymic or family name of a person	Parent Property: **hasPerson-Name** Property Axiom: ≡ **hasLast-Name** Property Axiom: ≡ **hasSurname** Domain: Person Range: text
hasFirstName (has first name)	Definition: the given name or first name of a person, that is the name chosen for them at birth or changed by them subsequently from the name given at birth	Parent Property: **hasPerson-Name** Property Axiom: ≡ **has-GivenName** Domain: Person Range: text
hasFullLegalName (has full legal name)	Definition: the legally complete name of a person, as used in formal dealings of a legal or contractual nature	Parent Property: **hasPerson-Name** Domain: Person Range: text
hasGender (has gender)	Definition: links a particular gender value with a person	Type: FunctionalProperty Domain: Person Range: text
hasGivenName (has given name)	Definition: the given name or first name of a person, that is the name chosen for them at birth or changed by them subsequently from the name given at birth	Parent Property: **hasPerson-Name** Domain: Person Range: text
hasLastName (has last name)	Definition: the patronymic or family name of a person	Parent Property: **hasPerson-Name**

Name	Annotations	Property Axioms
		Property Axiom: ≡ **hasSurname** Domain: Person Range: text
hasMaidenName (has maiden name)	Definition: the patronymic or family name which a person was born with and which predates any changes of name due to marriage	Parent Property: **hasPerson-Name** Domain: Person Range: text
hasMiddleNameOrInitial (has middle name or initial)		Parent Property: **hasPerson-Name** Domain: Person Range: text
hasPersonName (has person name)	Definition: links any sort of name to an individual person	Parent Property: **hasName** Domain: Person Range: text
hasSurname (has surname)	Definition: the patronymic or family name of a person	Parent Property: **hasPerson-Name** Domain: Person Range: text
verifiesDateOfBirth (verifies date of birth)	Definition: verifies the person's date of birth	Domain: IdentityDocument Range: string
verifiesPlaceOfBirth (verifies place of birth)	Definition: verifies the person's place of birth	Domain: IdentityDocument Range: string

10.10 Module: Law

Table 10-61. Law Module Metadata

Metadata Term	Value
sm:moduleName	Law
sm:moduleAbbreviation	FIBO-FND-LAW
sm:moduleVersion	1.0
sm:moduleAbstract	This module includes several ontologies defining legal concepts, including constitutions, laws and jurisdictions. It also includes the definition of legal capacities such as signatory capacity, contractual capability and the like.

10.10.1 Ontology: Legal Core

This ontology defines high-level legal concepts for use in other FIBO ontology elements. These concepts include law and constitution, both of which are framed at a more abstract level than national or state laws and constitutions, so that law forms the basis both for statutes and for company by-laws, and constitution forms the basis both for national or state constitutions and for instruments which are constitutive of incorporated legal entities. This ontology also defines some of the variants of these such as governmental constitutions and ordinances. Other types of law are provided in the Jurisdictions ontology as extensions of concepts in this ontology. Court of Law is also defined here.

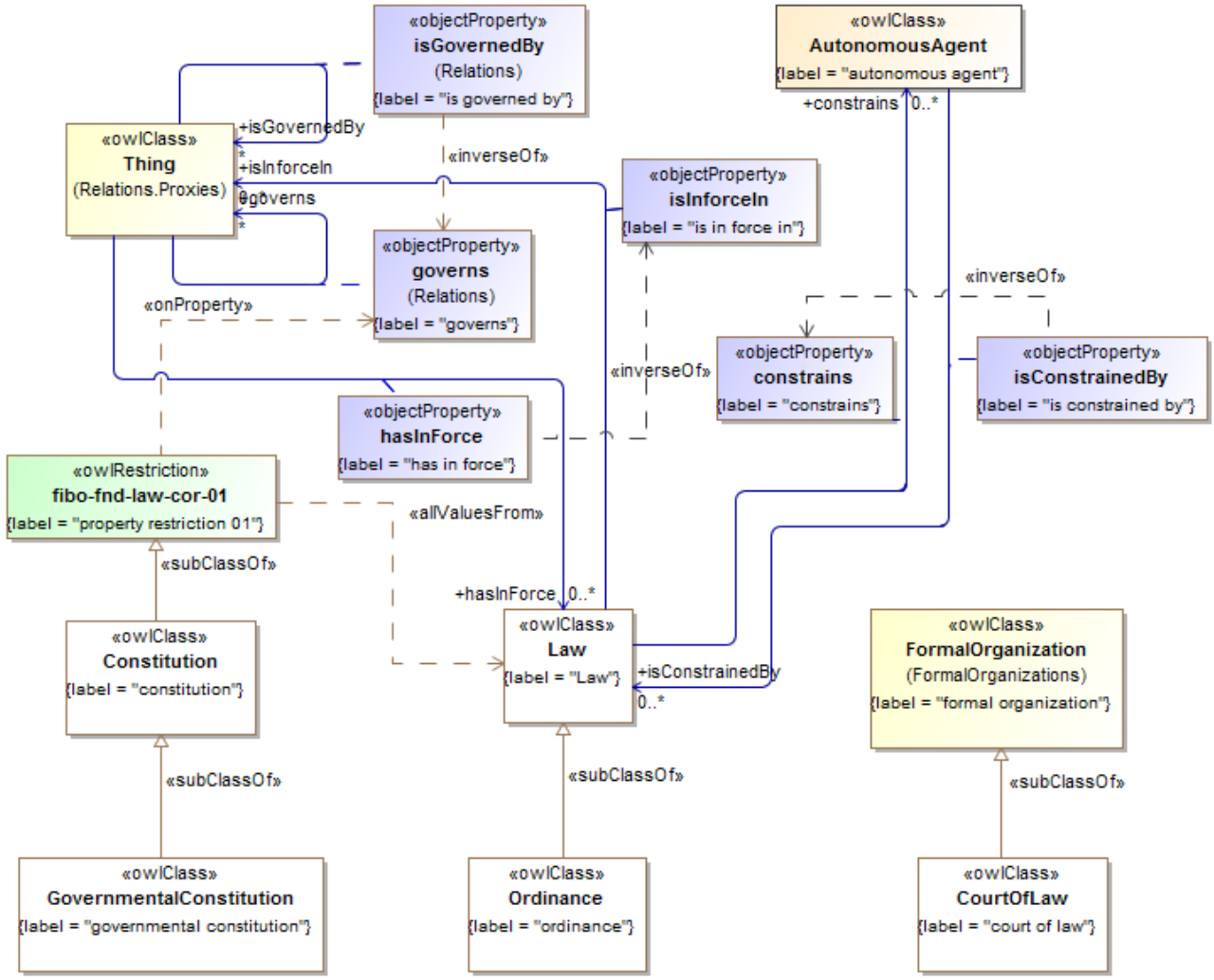

Figure 10.52 Legal Core Concepts

Table 10-62. Legal Core Ontology Metadata

Metadata Term	Value
sm:filename	Legal Core Ontology
sm:fileAbbreviation	fibo-fnd-law-cor
OntologyIRI	http://www.omg.org/spec/EDMC-FIBO/FND/Law/LegalCore/
owl:versionIRI	http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Law/LegalCore/
sm:dependsOn	http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/AnnotationVocabulary/ http://www.omg.org/spec/EDMC-FIBO/FND/Relations/Relations/

Metadata Term	Value
	http://www.omg.org/spec/EDMC-FIBO/FND/AgentsAndPeople/Agents/ http://www.omg.org/spec/EDMC-FIBO/FND/Organizations/FormalOrganizations/

Table 10-63. Legal Core Details

Classes

Name	Annotations	Class Expressions
Constitution (constitution)	Definition: A constitution defines the basic principles and laws of a nation, state, or social group that determine the powers and duties of the government and guarantee certain rights to the people in it. Editorial Note: This defines the framework in which laws (for a country constitution), rules and regulations (for a party or organization constitution) or contractual commitments are made and in which they have force. Definition Origin: Merriam-Webster Online Dictionary	Property Restriction: ∀ **governs.Law** (fibo-fnd-law-cor-01)
CourtOfLaw (court of law)	Definition: A court of law is a court that hears cases and decides them on the basis of statutes or the common law. Definition Origin: Merriam-Webster Online Dictionary	Parent Class: FormalOrganization
GovernmentalConstitution (governmental constitution)	Definition: a set of rules and principles that define the nature and extent of government. Editorial Note: This defines the framework in which laws are made and in which they have force. Explanatory Note: Most constitutions seek to regulate the relationship between institutions of the state, in a basic sense the relationship between the executive, legislature and the judiciary, but also the relationship of institutions within those branches. For example, executive branches can be divided into a head of government, government departments/ministries, executive agencies and a civil service/administration. Most constitutions also attempt to define the relationship between individuals and the state, and to establish the broad rights of individual citizens. It is thus the most basic law of a territory from which all the other laws and rules are hierarchically derived; in some territories it is in fact called Basic Law. Definition Origin: http://en.wikipedia.org/wiki/Constitution#Governmental_constitutions	Parent Class: Constitution
Law (law)	Definition: a system of rules and guidelines which are enforced through social institutions to govern behavior. Editorial Note: Any law or body of law, which may have force in some context, including national laws, company bylaws and the like. Explanatory Note: Law is a term which does not have a universally accepted definition. Certain Laws are made by governments, specifically by their legislatures. The formation of laws themselves may be influenced by a constitution (written or unwritten) and the rights encoded therein. The law shapes politics, economics and society in countless ways and serves as a social mediator of relations between people.	

Name	Annotations	Class Expressions
	Definition Origin: http://en.wikipedia.org/wiki/Law	
Ordinance (ordinance)	Definition: An authoritative rule or law; a decree or command; a public injunction or regulation, such as a city ordinance against excessive horn blowing. (Source: Dictionary.com)	Parent Class: Law

Properties

Name	Annotations	Property Axioms
constrains (constrains)	Definition: forces, compels, or obliges Definition Origin: http://dictionary.reference.com/browse/constrains	Domain:Law Range:AutonomousAgent Inverse: isConstrainedBy
hasInForce (has in force)	Definition: relates a jurisdiction or situation to a rule, regulation or law (collectively 'law') that is currently in force in that situation or jurisdiction	Range:Law Inverse: isInForceIn
isConstrainedBy (is constrained by)	Definition: identifies the policy, rule, regulation, contract, or other thing that compels or obliges someone to act in some way	Domain:AutonomousAgent Range:Law Inverse: constrains
isInforceIn (is in force in)	Definition: identifies a jurisdiction or similar context in which some law (including by-law, company by-law and state law) has effect	Domain:Law Inverse: hasInForce

10.10.2 Ontology: Jurisdiction

This ontology defines high level concepts relating to jurisdictions for use in other FIBO ontology elements. This includes a general definition of jurisdiction along with some basic types of jurisdiction, along with the factors which distinguish one type of jurisdiction from another. This ontology also defines basic types of legal system, and extends the basic concept of law which is in the LegalCore ontology.

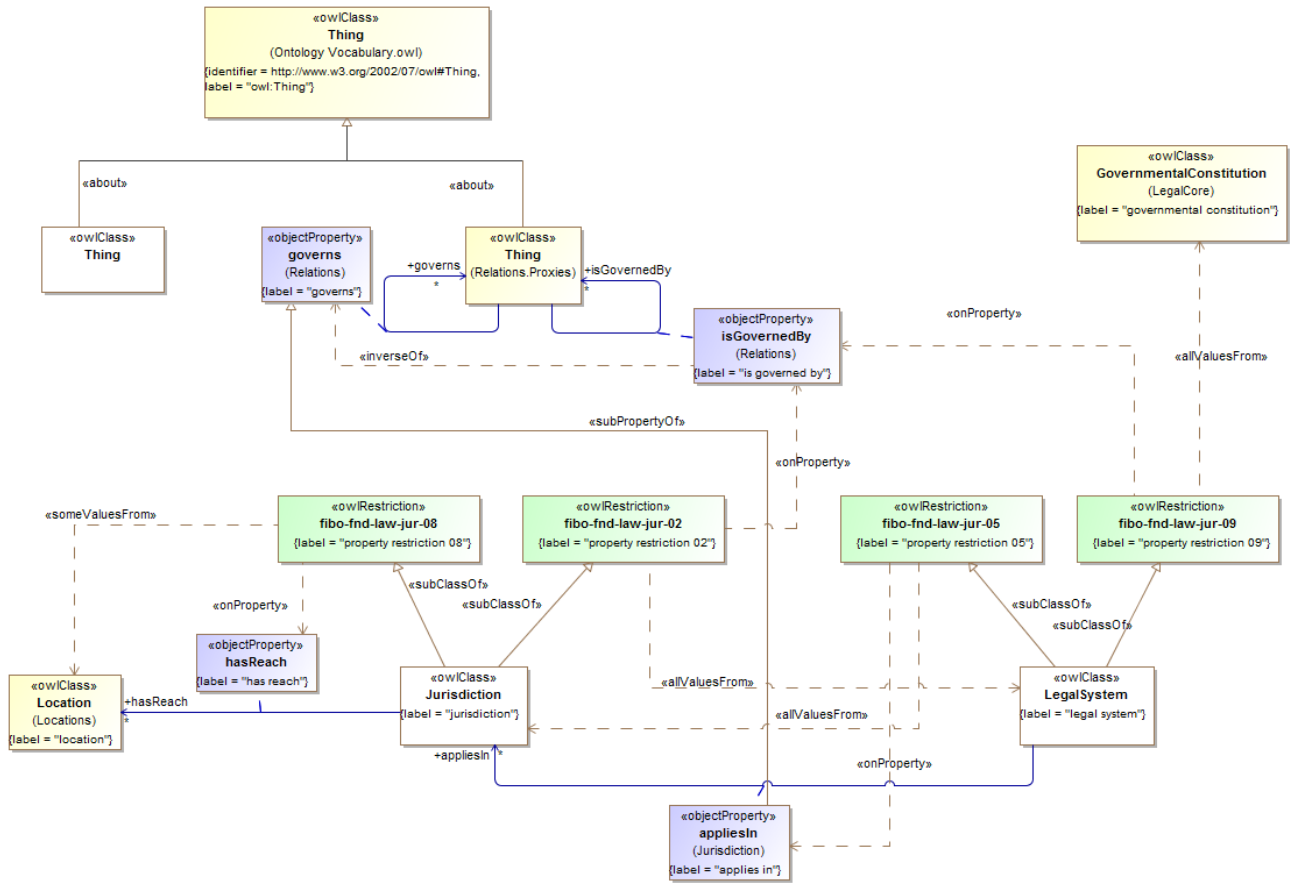

Figure 10.53 Jurisdiction Basic Concepts

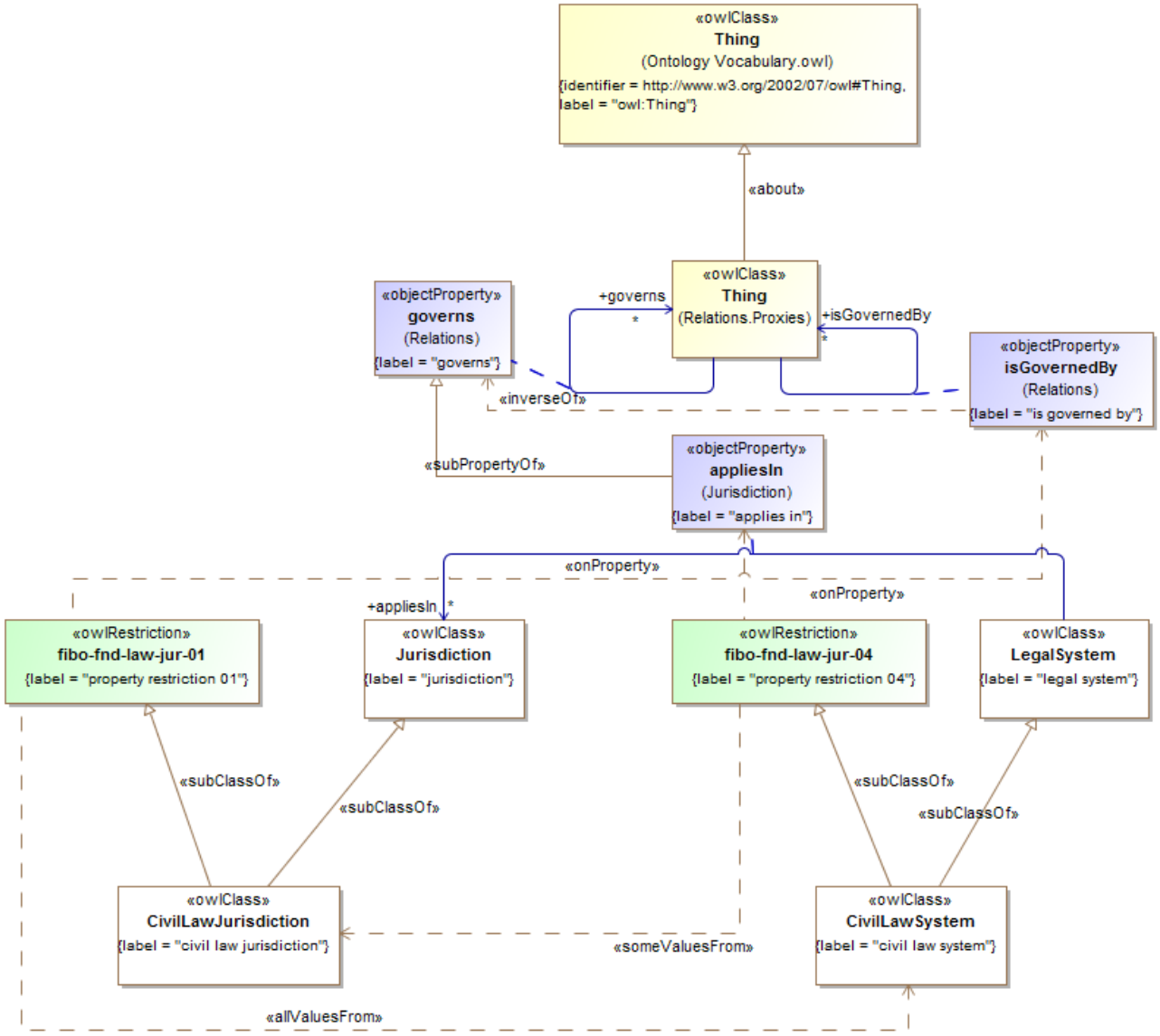

Figure 10.54 Civil Law Jurisdiction

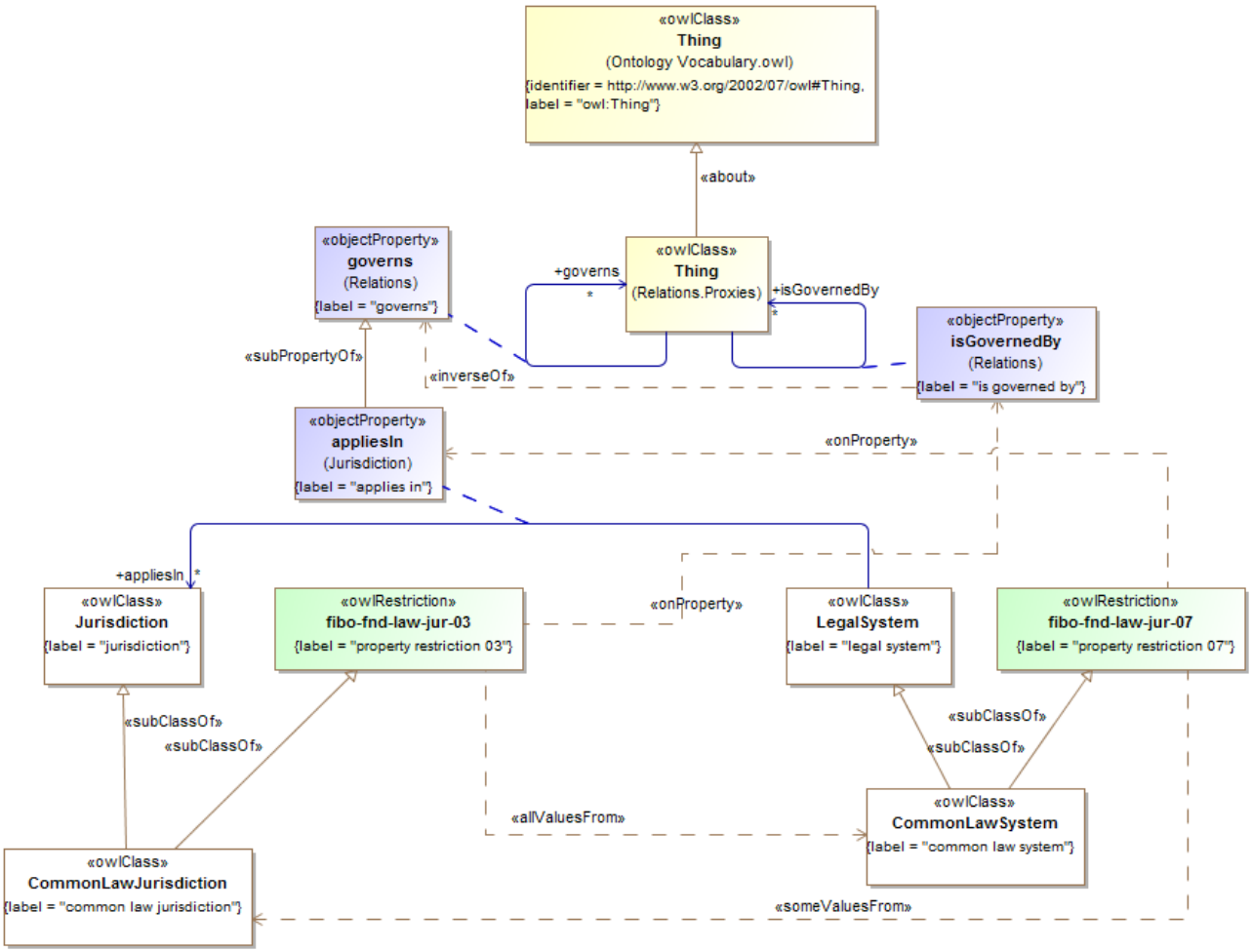

Figure 10.55 Common Law Jurisdiction

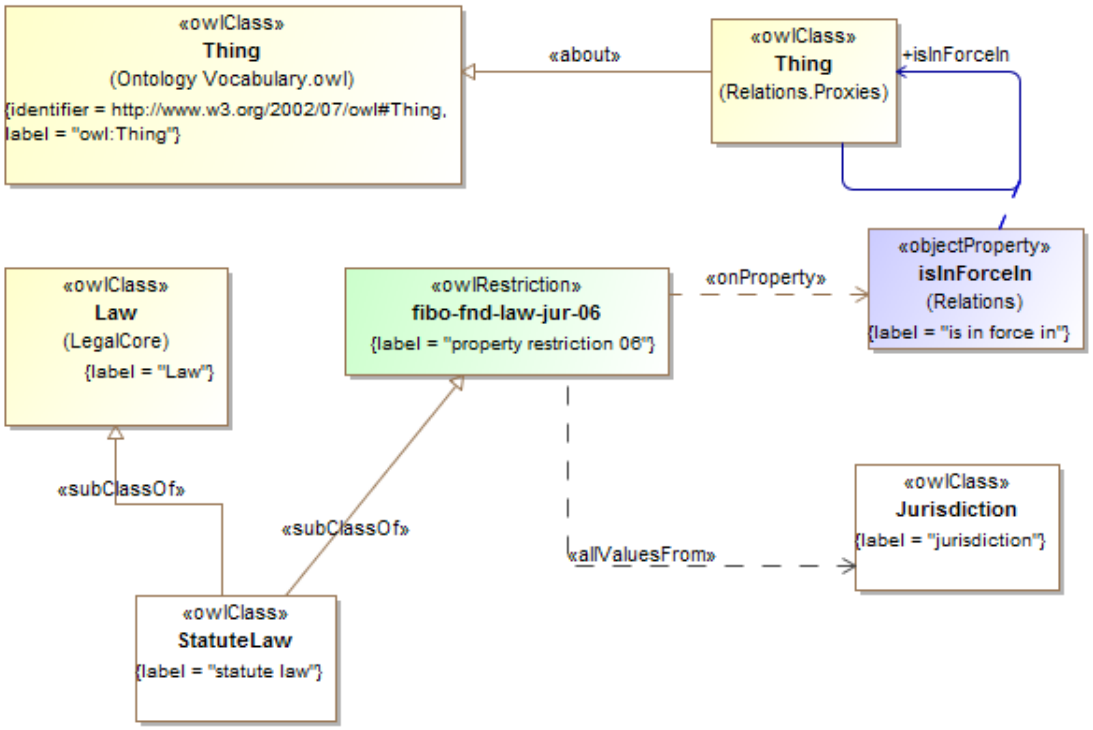

Figure 10.56 Statute Law

Table 10-64. Jurisdiction Ontology Metadata

Metadata Term	Value
sm:filename	Jurisdiction Ontology
sm:fileAbbreviation	fibo-fnd-law-jur
OntologyIRI	http://www.omg.org/spec/EDMC-FIBO/FND/Law/Jurisdiction/
owl:versionIRI	http://www.omg.org/spec/EDMC-FIBO/FND/20141101/Law/Jurisdiction/
sm:dependsOn	http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/AnnotationVocabulary/ http://www.omg.org/spec/EDMC-FIBO/FND/Relations/Relations/ http://www.omg.org/spec/EDMC-FIBO/FND/Places/Locations/ http://www.omg.org/spec/EDMC-FIBO/FND/Places/Countries/ http://www.omg.org/spec/EDMC-FIBO/FND/Law/LegalCore/

Table 10-65. Jurisdiction Details

Classes

Name	Annotations	Class Expressions
CivilLawJurisdiction (civil law jurisdiction)	Definition: a civil law jurisdiction	Parent Class: Jurisdiction Property Restriction: ∀ **isGovernedBy.CivilLawSystem** (fibo-fnd-law-jur-07)
CivilLawSystem (civil law system)	Definition: a legal system originating in Europe, intellectualized within the framework of late Roman law, and whose most prevalent feature is that its core principles are codified into a referable system which serves as the primary source of law Explanatory Note: This can be contrasted with common law systems whose intellectual framework comes from judge-made decisional law which gives precedential authority to prior court decisions on the principle that it is unfair to treat similar facts differently on different occasions (doctrine of judicial precedent). Definition Origin: http://en.wikipedia.org/wiki/Civil_law_(legal_system)	Parent Class: LegalSystem Property Restriction: ∃ **appliesIn.CivilLawJurisdiction** (fibo-fnd-law-jur-05)
CommonLawJurisdiction (common law jurisdiction)	Definition: a jurisdiction based on common law	Parent Class: Jurisdiction Property Restriction: ∀ **isGovernedBy.CommonLawSystem** (fibo-fnd-law-jur-08)
CommonLawSystem (common law system)	Definition: Common law, also known as case law or precedent, is law developed by judges through decisions of courts and similar tribunals Explanatory Note: A jurisdiction which is based in Common Law will also have alongside a legislature that passes statutes. Explanatory Note: By contrast, civil law (codified/continental law) is set on statutes adopted through the legislative/parliamentary process and/or regulations issued by the executive branch on base of the parliamentary statutes. A common law system is a legal system that gives great potential precedential weight to common law, on the principle that it is unfair to treat similar facts differently on different occasions. The body of precedent is called common law and it binds future decisions. In cases where the parties disagree on what the law is, a common law court looks to past precedential decisions of relevant courts. If a similar dispute has been resolved in the past, the court is bound to follow the reasoning used in the prior decision (this principle is known as stare decisis). If, however, the court finds that the current dispute is fundamentally distinct from all previous cases (called a matter of first impression), judges have the authority and duty to make law by creating precedent. Thereafter, the new decision becomes precedent, and will bind future courts. Definition Origin: http://en.wikipedia.org/wiki/Common_law	Parent Class: LegalSystem Property Restriction: ∃ **appliesIn.CommonLawJurisdiction** (fibo-fnd-law-jur-06)
Jurisdiction (jurisdiction)	Definition: the limits or territory within which authority may be exercised; the power, right, or authority to interpret and apply the law Definition Origin: Merriam-Webster Online Dictionary	Property Restriction: ∀ **isGovernedBy.LegalSystem** (fibo-fnd-law-jur-02) Property Restriction: ∃ **hasReach.Location** (fibo-fnd-law-jur-01)
LegalSystem (legal system)	Definition: Legal regimen of a country consisting of (1) a written or oral constitution, (2) primary legislation (statutes) enacted by the legis-	Property Restriction: ∀ **appliesIn.Jurisdiction**

Name	Annotations	Class Expressions
	lative body established by the constitution, (3) subsidiary legislation (bylaws) made by person or bodies authorized by the primary legislation to do so, (4) customs applied by the courts on the basis of traditional practices, and (5) principles or practices of civil, common, Roman, or other code of law. Editorial Note: This is a Mediating Thing, that is some context in which things have their meaning and existence - in this case, laws and the interpretation thereof by courts. Explanatory Note: The contemporary legal systems of the world are generally based on one of three basic systems: civil law, common law, and religious law, or combinations of these. However, the legal system of each country is shaped by its unique history and so incorporates individual variations. Definition Origin: http://www.businessdictionary.com/definition/legal-system.html	(fibo-fnd-law-jur-03) Property Restriction: ∀ **isGovernedBy.GovernmentalConstitution** (fibo-fnd-law-jur-04)
StatuteLaw (statute law)	Definition: written law (as opposed to oral or customary law) set down by a legislature or by a legislator (in the case of an absolute monarchy). Explanatory Note: Statutes may originate with national, state legislatures or local municipalities. Statutory laws are subordinate to the higher constitutional laws of the land. Definition Origin: http://en.wikipedia.org/wiki/Statute_law	Parent Class: Law Property Restriction: ∀ **isInForceIn.Jurisdiction** (fibo-fnd-law-jur-09)

Properties

Name	Annotations	Property Axioms
appliesIn (applies in)	Definition: indicates the jurisdiction in which a particular legal system applies	Parent Property: governs Domain: LegalSystem Range: Jurisdiction
hasReach (has reach)	Definition: indicates the geopolitical unit (country, federal province or municipality) or geophysical extent in which the jurisdiction has effect	Domain: Jurisdiction Range: Location

10.10.3 Ontology: Legal Capacity

This ontology defines high-level legal concepts, especially those related to legal responsibilities, for use in other FIBO ontology elements. The ontology defines things which are conferred upon some entity by some legal instrument, and elaborates this into a number of specific capacities, responsibilities and powers, each of which forms the basis for many of the concepts used elsewhere in FIBO in defining legal personhood, executive powers and the like.

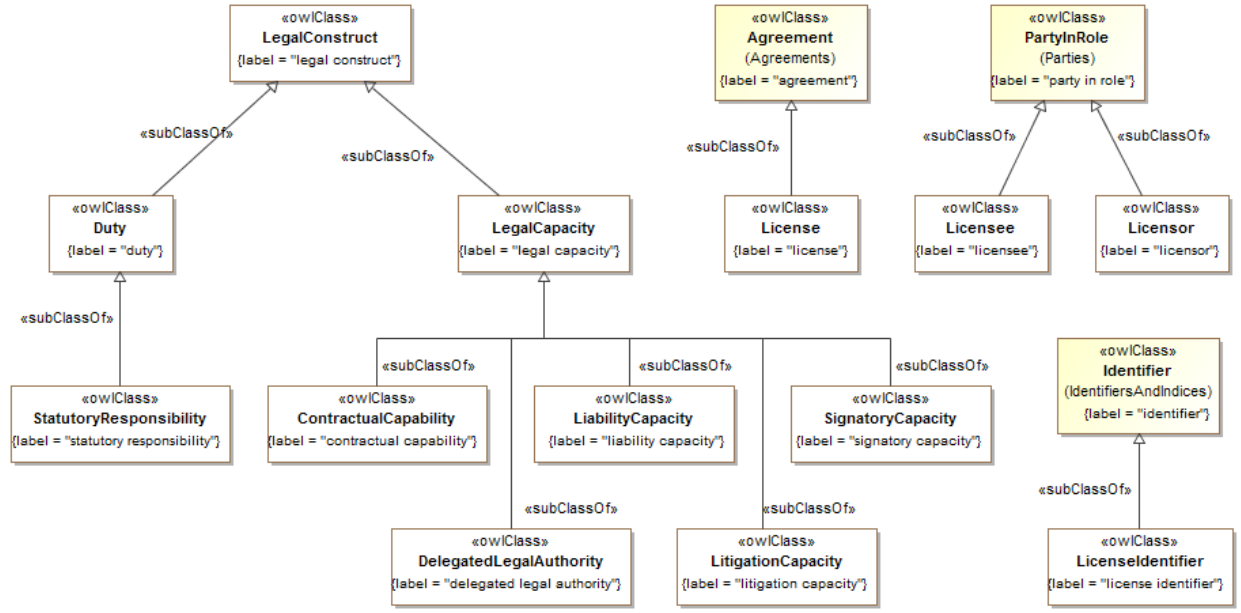

Figure 10.57 Legal Capacity Class Hierarchy

Figure 10.57 provides an overview of the concepts in the Legal Capacity ontology. Subsequent diagrams provide more detailed views on several of the concepts, with a focus on the logical restrictions that further define them.

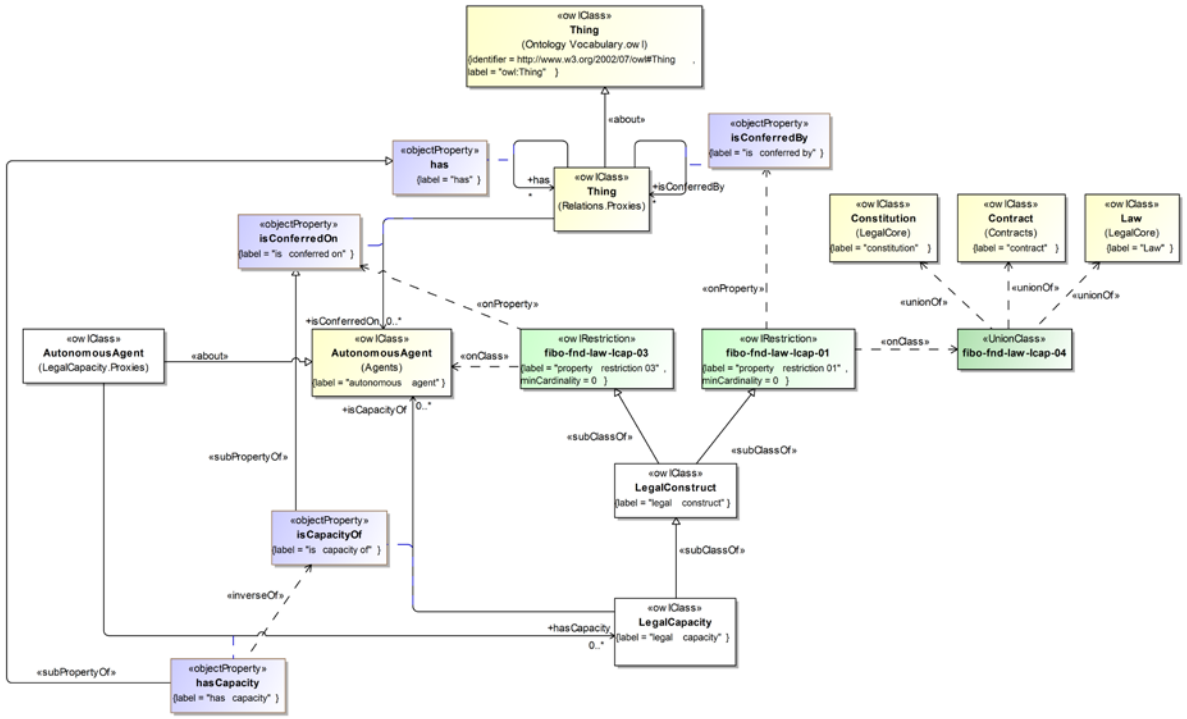

Figure 10.58 Definition of Legal Capacity

Figure 10.58 defines legal capacity in terms of who that capacity can be conferred on, who has that capacity, and the legal basis for that capacity.

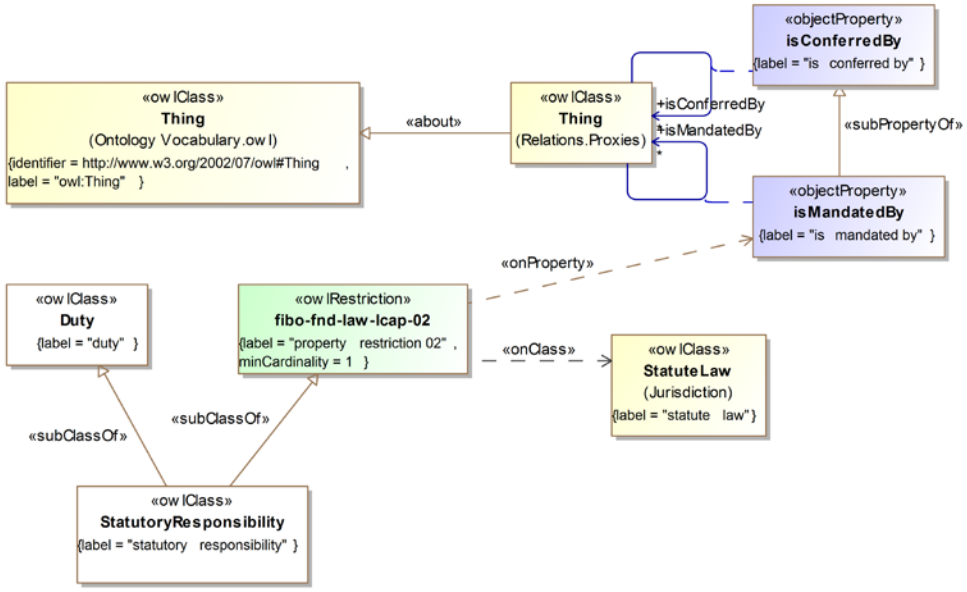

Figure 10.59 Definition of Statutory Responsibility

Figure 10.59 defines statutory responsibility as something that is mandated by at least one statute (statute law).

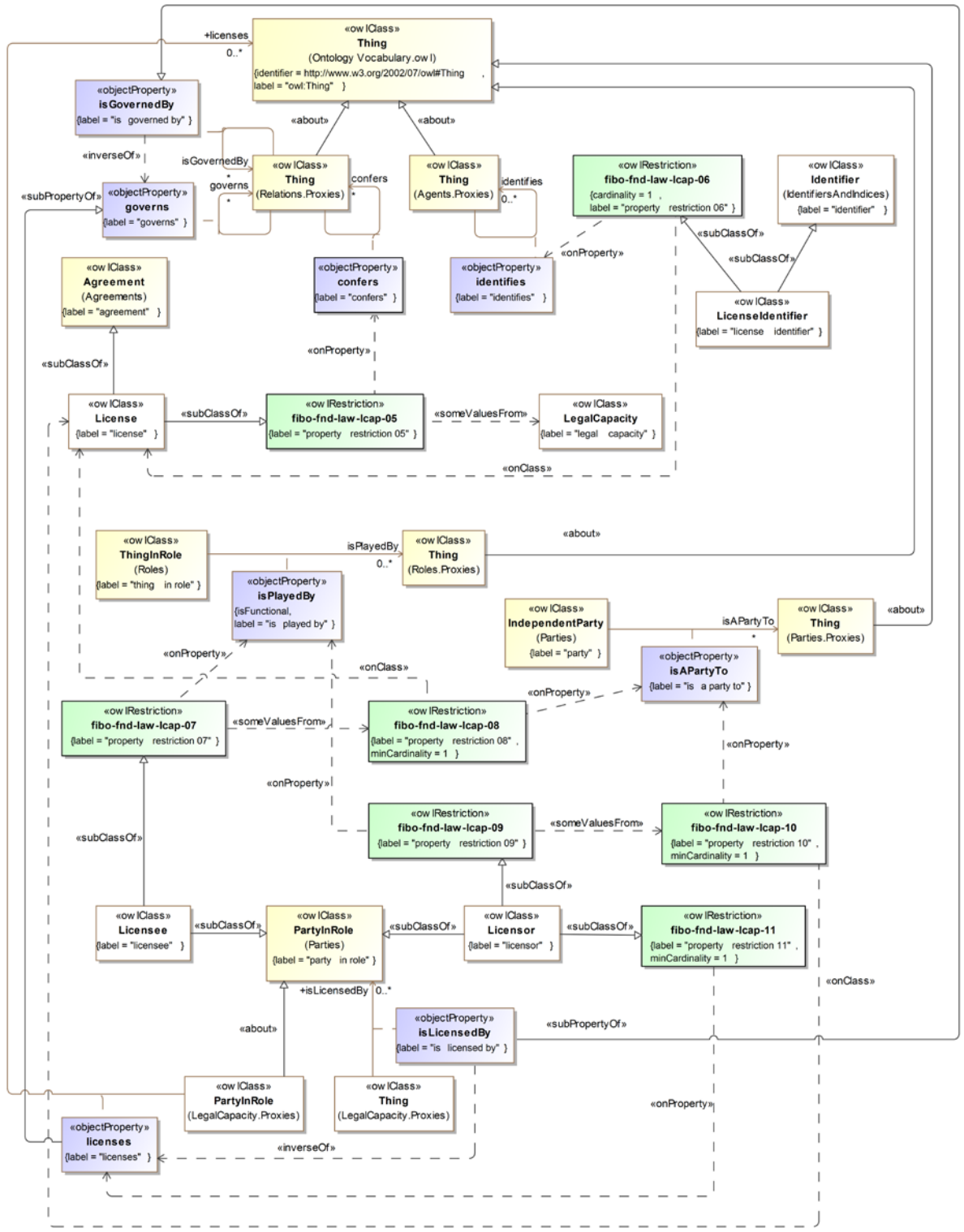

Figure 10.60 Definition of Licensing

Figure 10.60 provides the definitions for concepts and relationships for basic licensing.

Table 10-66. Legal Capacity Ontology Metadata

Metadata Term	Value
sm:filename	Legal Capacity Ontology
sm:fileAbbreviation	fibo-fnd-law-lcap
OntologyIRI	http://www.omg.org/spec/EDMC-FIBO/FND/Law/LegalCapacity/
owl:versionIRI	http://www.omg.org/spec/EDMC-FIBO/FND/20160201/Law/LegalCapacity/
sm:dependsOn	http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/AnnotationVocabulary/ http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/BusinessFacingTypes/ http://www.omg.org/spec/EDMC-FIBO/FND/Relations/Relations/ http://www.omg.org/spec/EDMC-FIBO/FND/AgentsAndPeople/Agents/ http://www.omg.org/spec/EDMC-FIBO/FND/Places/Locations/ http://www.omg.org/spec/EDMC-FIBO/FND/Places/Countries/ http://www.omg.org/spec/EDMC-FIBO/FND/Places/Addresses/ http://www.omg.org/spec/EDMC-FIBO/FND/GoalsAndObjectives/Goals/ http://www.omg.org/spec/EDMC-FIBO/FND/Organizations/Organizations/ http://www.omg.org/spec/EDMC-FIBO/FND/Organizations/FormalOrganizations/ http://www.omg.org/spec/EDMC-FIBO/FND/AgentsAndPeople/People/ http://www.omg.org/spec/EDMC-FIBO/FND/Parties/Roles/ http://www.omg.org/spec/EDMC-FIBO/FND/Parties/Parties/ http://www.omg.org/spec/EDMC-FIBO/FND/Law/LegalCore/ http://www.omg.org/spec/EDMC-FIBO/FND/Law/Jurisdiction/ http://www.omg.org/spec/EDMC-FIBO/FND/Agreements/Agreements/ http://www.omg.org/spec/EDMC-FIBO/FND/Agreements/Contracts/ http://www.omg.org/spec/EDMC-FIBO/FND/Arrangements/IdentifiersAndIndices/

Table 10-67. Legal Capacity Details

Classes

Name	Annotations	Class Expressions
ContractualCapability (contractual capability)	Definition: the capacity to enter into legally binding contracts	Parent Class: LegalCapacity
DelegatedLegalAuthority (delegated legal authority)	Definition: institutionalized and legal power inherent in a particular job, function, or position that is meant to enable its holder to successfully carry out his or her responsibilities, where such power has been delegated through some formal means Editorial note: This is always accompanied by an equal responsibility for one's actions or a failure to act. Scope note: This specifically means the authority to make legally binding commitments.	Parent Class: LegalCapacity
Duty (duty)	Definition: some obligation which exists and is imposed on some individual	Parent Class: LegalConstruct
LegalCapacity (legal capacity)	Definition: the capacity to carry out certain actions or to have certain rights	Parent Class: LegalConstruct
LegalConstruct (legal construct)	Definition: something which is conferred by way of law or contract, such as a right	Property Restriction: ≥ 0 **isConferredOn**.**AutonomousAgent** (fibo-fnd-law-lcap-03) Property Restriction: ≥ 0 **isConferredBy** (**Contract** ∪ **Constitution** ∪ **Law**) (fibo-fnd-law-lcap-01, fibo-fnd-law-lcap-04)
LiabilityCapacity (liability capacity)	Definition: the ability to be sued at law	Parent Class: LegalCapacity
LitigationCapacity (litigation capacity)	Definition: the legal capacity to pursue a litigation action in law	Parent Class: LegalCapacity
License (license)	Definition: grant of permission needed to do something Adapted from: Barron's Dictionary of Business and Economics Terms, Fifth Edition, 2012	Parent Class: Agreement Property Restriction: ∃ **confers**.**LegalCapacity** (fibo-fnd-law-lcap-05)
Licensee (licensee)	Definition: a party to whom a license has been granted Adapted from: Barron's Dictionary of Business and Economics Terms, Fifth Edition, 2012	Parent Class: PartyInRole Property Restriction: ∃ **isPlayedBy** (≥ 1 **isAPartyTo**.**License**) (fibo-fnd-law-lcap-07, fibo-fnd-law-lcap-08)
LicenseIdentifier (license identifier)	Definition: an identifier associated with a license	Parent Class: Identifier Property Restriction: = 1 **identifies**.**License** (fibo-fnd-law-lcap-06)
Licensor (licensor)	Definition: a party who grants a license Adapted from: Barron's Dictionary of Business and Economics Terms, Fifth Edition, 2012	Parent Class: PartyInRole Property Restriction: ∃ **isPlayedBy** (≥ 1 **isAPartyTo**.**License**) (fibo-fnd-law-lcap-09, fibo-fnd-law-lcap-10) Property Restriction: ≥ 1 **licenses** (fibo-fnd-law-lcap-11)

Name	Annotations	Class Expressions
SignatoryCapacity (signatory capacity)	Definition: the capacity of some natural person to sign agreements on the part of some organization or legal person	Parent Class: LegalCapacity
StatutoryResponsibility (statutory responsibility)	Definition: an obligation which is defined under some body of law (statute)	Parent Class: Duty Property Restriction: ≥ 1 **isMandatedBy**.**StatuteLaw** (fibo-fnd-law-lcap-02)

Properties

Name	Annotations	Property Axioms
hasCapacity (has capacity)	Definition: identifies an individual or organization that has some capability to carry out certain actions, or has certain rights or obligations	Parent Property: has Domain: AutonomousAgent Range: LegalCapacity Inverse: isCapacityOf
isCapacityOf (is capacity of)	Definition: identifies an individual or organization on which a given legal capacity has been conferred Scope note: This includes capacities specific to duties at law (such as those for corporate officers) as well as the ability or capacity to incur liability.	Parent Property: isConferredOn Domain: LegalCapacity Range: AutonomousAgent
isLicensedBy (is licensed by)	Definition: indicates the regulatory agency that regulates something	Parent Property: isGovernedBy Range: PartyInRole Inverse: licenses
licenses (licenses)	Definition: issues a license required in order to perform some task, provide some service, exercise some privilege, or pursue some line of business or occupation to some party Adapted from: http://www.merriam-webster.com/dictionary/regulate	Parent Property: governs Domain: PartyInRole

10.11 Module: Ownership and Control

Table 10-68. Ownership and Control Module Metadata

Metadata Term	Value
sm:moduleName	Ownership and Control
sm:moduleAbbreviation	FIBO-FND-OAC
sm:moduleVersion	1.0
sm:moduleAbstract	This module includes ontologies defining the meanings of ownership, asset and owner, and of types of control such as de jure and de facto control. These form the basis of ownership and control relationship hierarchies as well as what it means to own or to control something.

10.11.1 Ontology: Control

This ontology defines high-level, control-related concepts for use in other FIBO ontology elements. The ontology covers basic concepts around control, along with a distinction between de jure and de facto control, the former being derived with reference to terms in the LegalCapacity ontology.

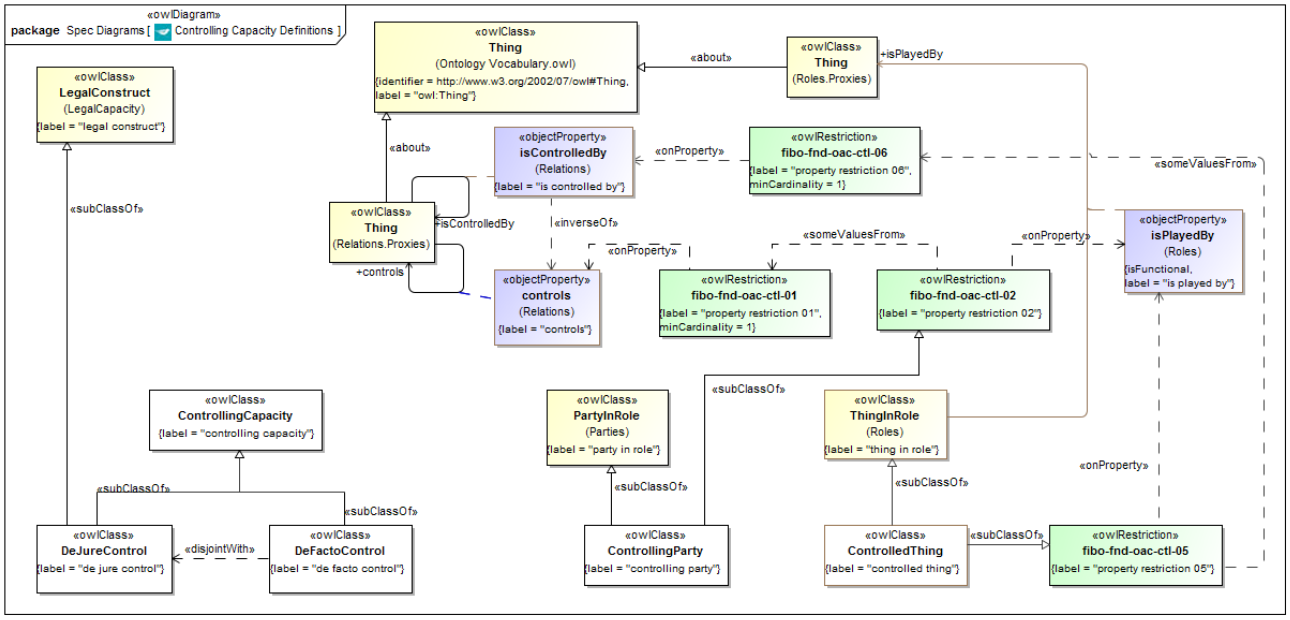

Figure 10.61 Controlling Capacity Definitions

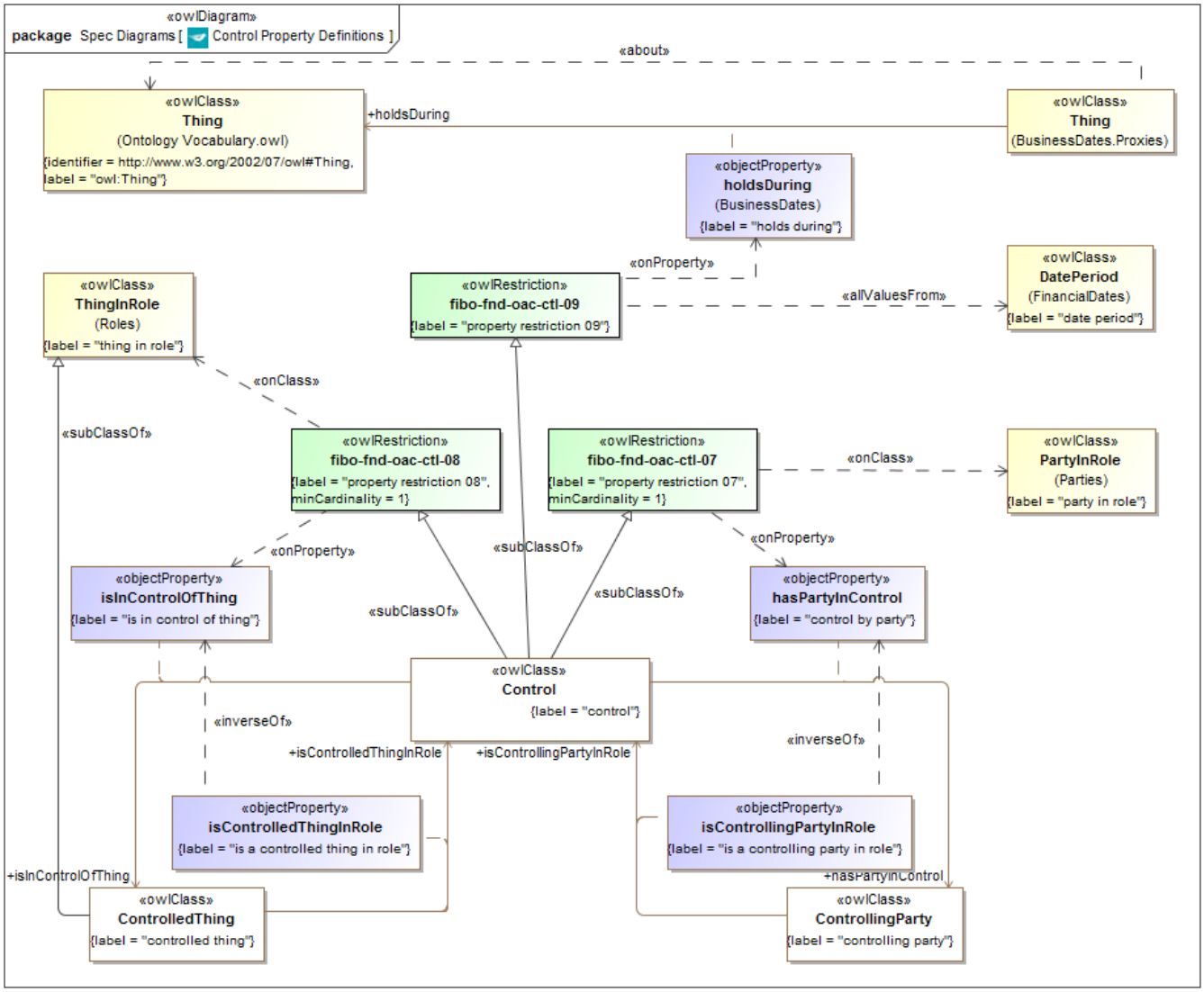

Figure 10.62 Control Property Definitions

Table 10-69. Control Ontology Metadata

Metadata Term	Value
sm:filename	Control Ontology
sm:fileAbbreviation	fibo-fnd-oac-ctl
OntologyIRI	http://www.omg.org/spec/EDMC-FIBO/FND/OwnershipAndControl/Control/
owl:versionIRI	http://www.omg.org/spec/EDMC-FIBO/FND/20141101/OwnershipAndControl/Control/

Metadata Term	Value
sm:dependsOn	http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/AnnotationVocabulary/ http://www.omg.org/spec/EDMC-FIBO/FND/Relations/Relations/ http://www.omg.org/spec/EDMC-FIBO/FND/DatesAndTimes/FinancialDates/ http://www.omg.org/spec/EDMC-FIBO/FND/DatesAndTimes/BusinessDates/ http://www.omg.org/spec/EDMC-FIBO/FND/Parties/Roles/ http://www.omg.org/spec/EDMC-FIBO/FND/Parties/Parties/ http://www.omg.org/spec/EDMC-FIBO/FND/Law/LegalCapacity/

Table 10-70. Control Details

Classes

Name	Annotations	Class Expressions
Control (control)	Definition: the possession by a party, direct or indirect, of the power to direct or cause the direction of the management and policies of a thing, whether through the ownership of voting shares, by contract, or otherwise	Property Restriction: ≥ 1 hasPartyInControl.PartyInRole (fibo-fnd-oac-ctl-07) Property Restriction: ≥ 1 isInControlOfThing.ThingInRole (fibo-fnd-oac-ctl-08) Property Restriction: ∀ holdsDuring.DatePeriod (fibo-fnd-oac-ctl-09)
ControlledThing (controlled thing)	Definition: thing over which some party exercises some form of control in some context	Parent Class: ThingInRole Property Restriction: ∃ isPlayedBy (≥ 1 isControlledBy) (fibo-fnd-oac-ctl-05, (fibo-fnd-oac-ctl-06)
Controlling-Capacity (Controlling capacity)	Definition: the possession, direct or indirect, of the power to direct or cause the direction of the thing which is controlled.	
ControllingParty (controlling party)	Definition: Party which exercises some form of control in some context. Editorial Note: At this level of abstraction it is not defined whether the control is some degree of controlling interest, or some level of actual control (asserted or calculated) in some entity.	Parent Class: PartyInRole Property Restriction: ∃ isPlayedBy (≥ 1 controls) (fibo-fnd-oac-ctl-02, fibo-fnd-oac-ctl-01)
DeFactoControl (de facto control)	Definition: control that is understood, due to condition or situation treated as standard or official, even if not explicitly stated (or actually standardized)	Parent Class: ControllingCapacity Class Axiom: ¬ DeJureControl
DeJureControl (de jure control)	Definition: control that is formalized in law, or codified in some legal instrument	Parent Class: LegalConstruct Parent Class: ControllingCapacity Class Axiom: ¬ DeFactoControl

Properties

Name	Annotations	Property Axioms
hasPartyInControl (control by party)	Definition: indicates the party in a control relationship where a party controls a thing	Domain: Control Range: ControllingParty Inverse: isControllingPartyInRole
isInControlOfThing (is in control of thing)	Definition: Indicates the thing in a control relationship where a party controls a thing	Domain: Control Range: ControlledThing Inverse: isControlledThingInRole
isControlledThingInRole (is a controlled thing in role)	Definition: indicates the context of control in which the thing plays the role of being controlled	Domain: ControlledThing Range: Control Inverse: isInControlOfThing
isControllingPartyInRole (is a controlling party in role)	Definition: indicates the context of control in which the party plays the role of controlling some thing	Domain: ControllingParty Range: Control Inverse: hasPartyInControl

10.11.2 Ontology: Ownership

This ontology defines high-level, ownership-related concepts for use in other FIBO ontology elements. These include the concept of owner, asset and ownership along with relationships between them whereby an asset is something owned by some owner.

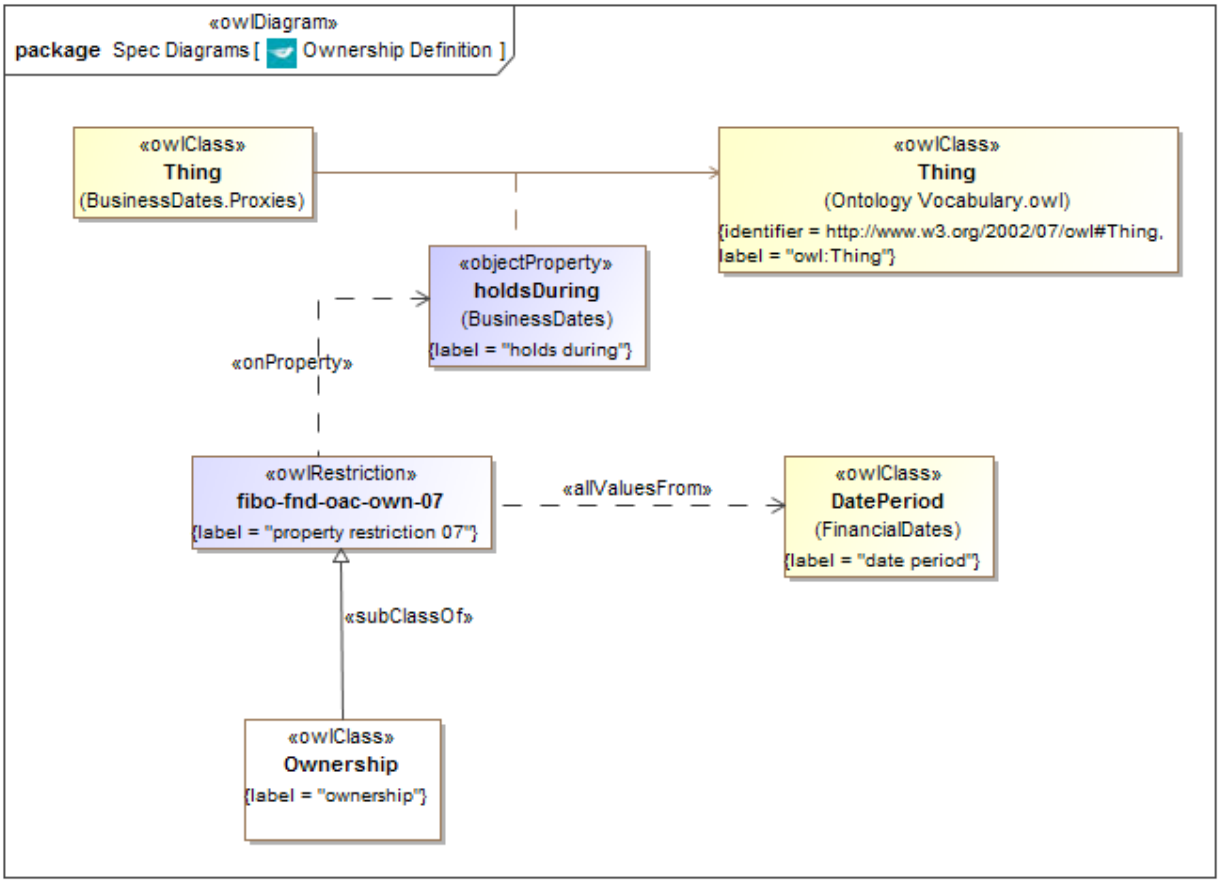

Figure 10.63 Ownership Definition

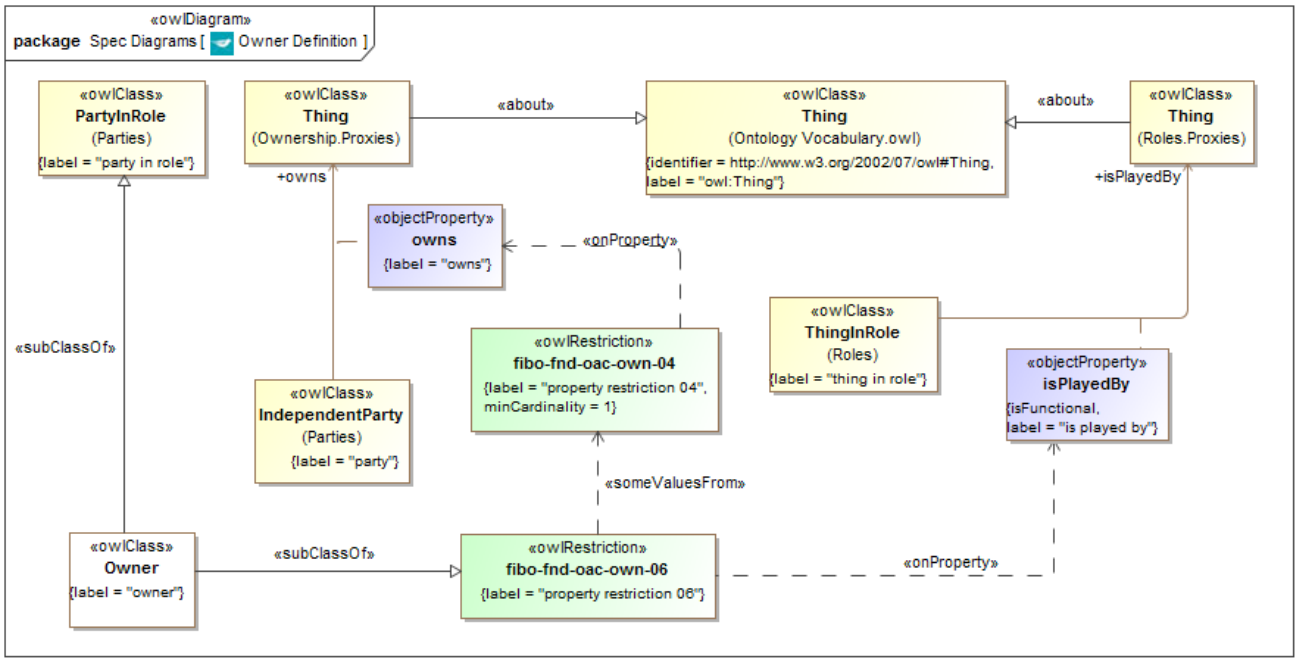

Figure 10.64 Owner Definition

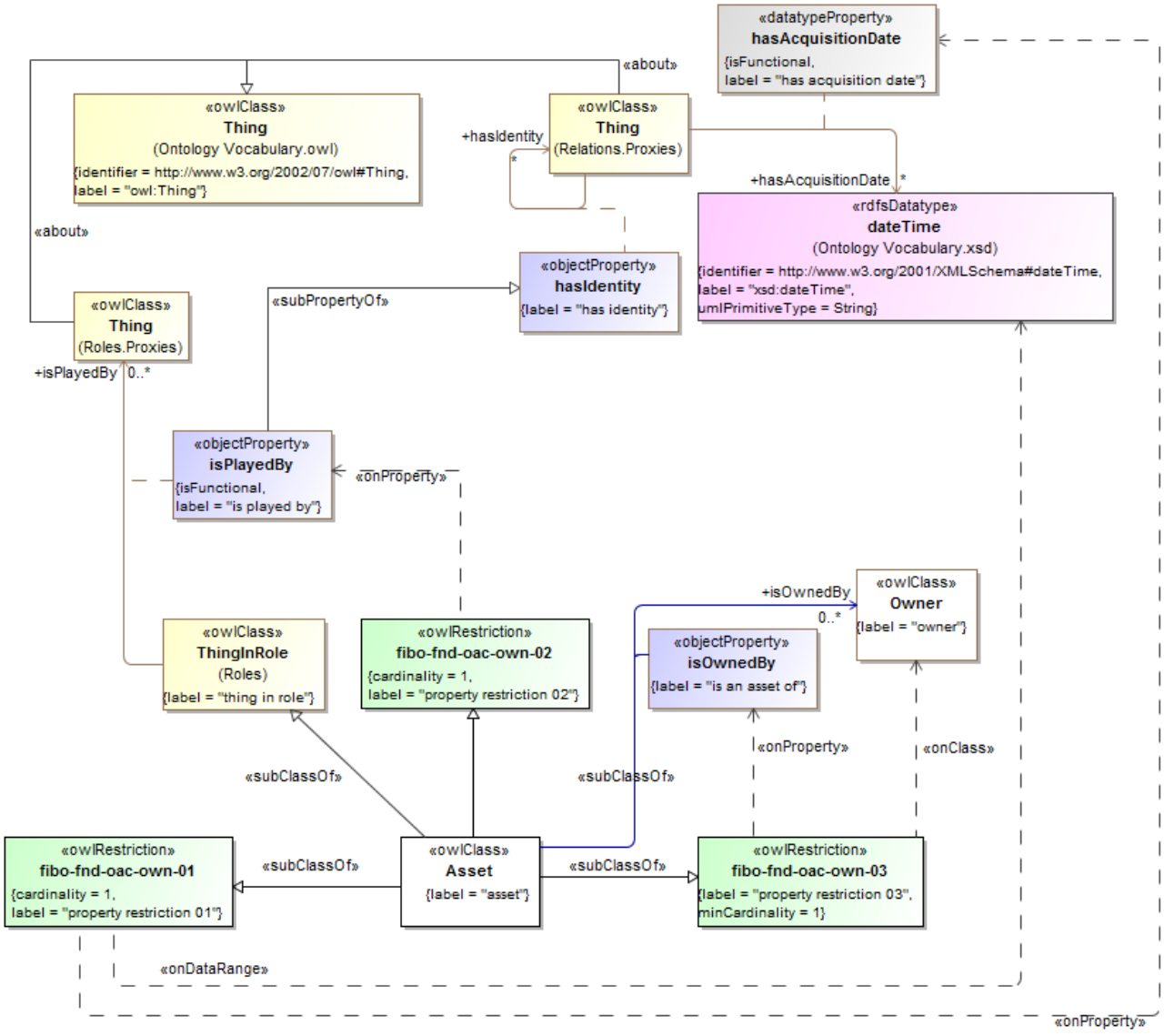

Figure 10.65 Asset Definition

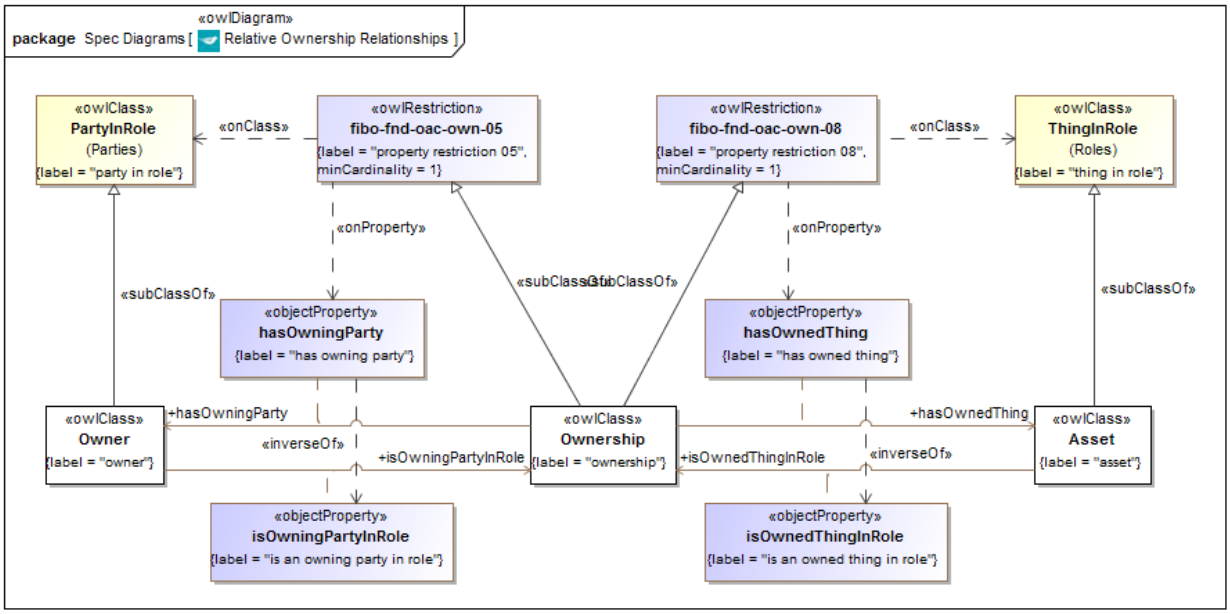

Figure 10.66 Relative Ownership Relations

Table 10-71 Ownership Ontology Metadata

Metadata Term	Value
sm:filename	Ownership Ontology
sm:fileAbbreviation	fibo-fnd-oac-own
OntologyIRI	http://www.omg.org/spec/EDMC-FIBO/FND/OwnershipAndControl/Ownership/
owl:versionIRI	http://www.omg.org/spec/EDMC-FIBO/FND/20141101/OwnershipAndControl/Ownership/
sm:dependsOn	http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/AnnotationVocabulary/ http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/BusinessFacingTypes/ http://www.omg.org/spec/EDMC-FIBO/FND/Relations/Relations/ http://www.omg.org/spec/EDMC-FIBO/FND/AgentsAndPeople/Agents/ http://www.omg.org/spec/EDMC-FIBO/FND/Places/Locations/ http://www.omg.org/spec/EDMC-FIBO/FND/Places/Countries/ http://www.omg.org/spec/EDMC-FIBO/FND/Places/Addresses/ http://www.omg.org/spec/EDMC-

Metadata Term	Value
	FIBO/FND/GoalsAndObjectives/Goals/ http://www.omg.org/spec/EDMC-FIBO/FND/Organizations/Organizations/ http://www.omg.org/spec/EDMC-FIBO/FND/Organizations/FormalOrganizations/ http://www.omg.org/spec/EDMC-FIBO/FND/AgentsAndPeople/People/ http://www.omg.org/spec/EDMC-FIBO/FND/Parties/Roles/ http://www.omg.org/spec/EDMC-FIBO/FND/Parties/Parties/

Table 10-72. Ownership Details

Classes

Name	Annotations	Class Expressions
Asset (asset)	A thing held by some party and having some value.	Property Restriction: = 1 **hasAcquisitionDate.dateTime** (fibo-fnd-oac-own-01) Property Restriction: = 1 **isPlayedBy** (fibo-fnd-oac-own-02) Property Restriction: ≥ 1 **isOwnedBy.Owner** (fibo-fnd-oac-own-03)
Owner (owner)	A party in the ownership role; one that owns something. The thing owned is an Asset to that Party.	Parent Class: party in role Property Restriction: ∃ **isPlayedBy (≥ 1 owns)** (fibo-fnd-oac-own-06, fibo-fnd-oac-own-04)
Ownership (ownership)	Ownership is the context in which some Party is said to own some Independent Thing. The Party is defined as such due to its being the owning party to that Thing.	Property Restriction: ≥ 1 **hasOwningParty.PartyInRole** (fibo-fnd-oac-own-05) Property Restriction: ∀ **holdsDuring.DatePeriod** (fibo-fnd-oac-own-07) Property Restriction: ≥ 1 **hasOwnedThing.ThingInRole** (fibo-fnd-oac-own-08)

Properties

Name	Annotations	Property Axioms
hasOwnedThing (has owned thing)	Definition: indicates the party in an ownership relationship where a party owns a thing	Domain: Ownership Range: Asset Inverse: isOwnedThingInRole
hasOwningParty (has owning party)	Definition: indicates the thing in an ownership relationship where a party owns a thing	Domain: Ownership Range: Owner

Name	Annotations	Property Axioms
		Inverse: isOwningPartyInRole
isOwnedBy (is an asset of)	Definition: identifies the party that owns the asset	Domain: Asset Range: Owner
isOwnedThingInRole (is an owned thing in role)	Definition: indicates the context of ownership in which the thing plays the role of an asset	Domain: Asset Range: Ownership Inverse: hasOwnedThing
isOwningPartyInRole (is an owning party in role)	Definition: indicates the context of ownership in which the party plays the role of owner	Domain: owner Range: ownership Inverse: hasOwningParty
owns (owns)	Definition: to have (something) as one's own, possess	Domain: independent party Inverse: isOwnedBy

10.11.3 Ontology: OwnershipAndControl

This ontology defines high-level, ownership-related concepts for use in other FIBO ontology elements. These include the concept of owner, asset and ownership along with relationships between them whereby an asset is some thing owned by some owner.

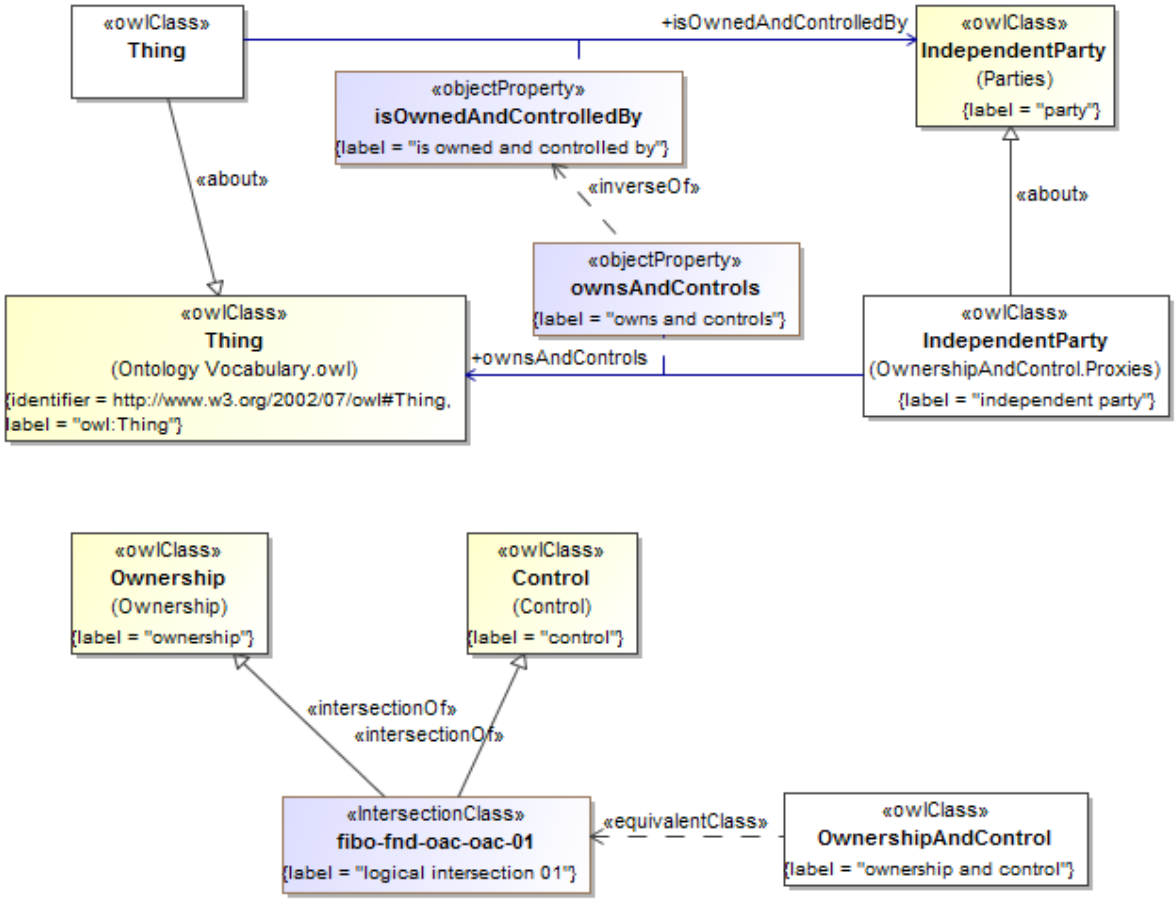

Figure 90.67 Ownership and Control

Table 10-73. OwnershipAndControl Ontology Metadata

Metadata Term	Value
sm:filename	OwnershipAndControl
sm:fileAbbreviation	fibo-fnd-oac-oac
OntologyIRI	http://www.omg.org/spec/EDMC-FIBO/FND/OwnershipAndControl/OwnershipAndControl/
owl:versionIRI	http://www.omg.org/spec/EDMC-FIBO/FND/20141101/OwnershipAndControl/OwnershipAndControl/
sm:dependsOn	http://www.omg.org/spec/EDMC-FIBO/FND/Relations/Relations/ http://www.omg.org/spec/EDMC-FIBO/FND/Parties/Parties/ http://www.omg.org/spec/EDMC-FIBO/FND/Parties/Roles/ http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/AnnotationVocabulary/

Table 10-74. Ownership and Control Details

Classes

Name	Annotations	Class Expressions
IndependentParty (independent party)	Definition: Proxy for IndependentParty	Class Axiom: ≡ **Independent-Party**
OwnershipAndControl (ownership and control)	Definition: The intersection of ownership and control reflects the unique case where an Iindependent Party both owns and controls another Independent Thing.	Class Axiom: ≡ (Ownership ∩ Control) (fibo-fnd-oac-oac-01)

Properties

Name	Annotations	Property Axioms
isOwnedAndControlledBy (is owned and controlled by)	Definition: a relationship between some thing and the party that owns, influences, manages and directs it	Range: IndependentParty
ownsAndControls (owns and controls)	Definition: directs and exercises authoritative or dominating influence over some thing that is also owned Editorial Note: basic rule: if x controls y and x owns y then x owns and controls y SWRL rule: controls(?x, ?y), owns(?x, ?y) -> ownsAndControls(?x, ?y)	Domain: IndependentParty Inverse: isOwnedAndControlledBy

10.12 Module: Accounting

Table 10-75. Accounting Module Metadata

Metadata Term	Value
sm:moduleName	Accounting
sm:moduleAbbreviation	FIBO-FND-ACC
sm:moduleVersion	1.0
sm:moduleAbstract	This module contains ontologies of general accounting concepts including debt, equity, interest and so on, as well as currency amounts.

10.12.1 Ontology: Accounting Equity

This ontology defines equity-related concepts for use in defining other FIBO ontology elements. These are based on basic accounting principles as they relate to equity, debt, assets and liabilities of a firm. Equity forms the basis for ownership of certain forms of corporate body.

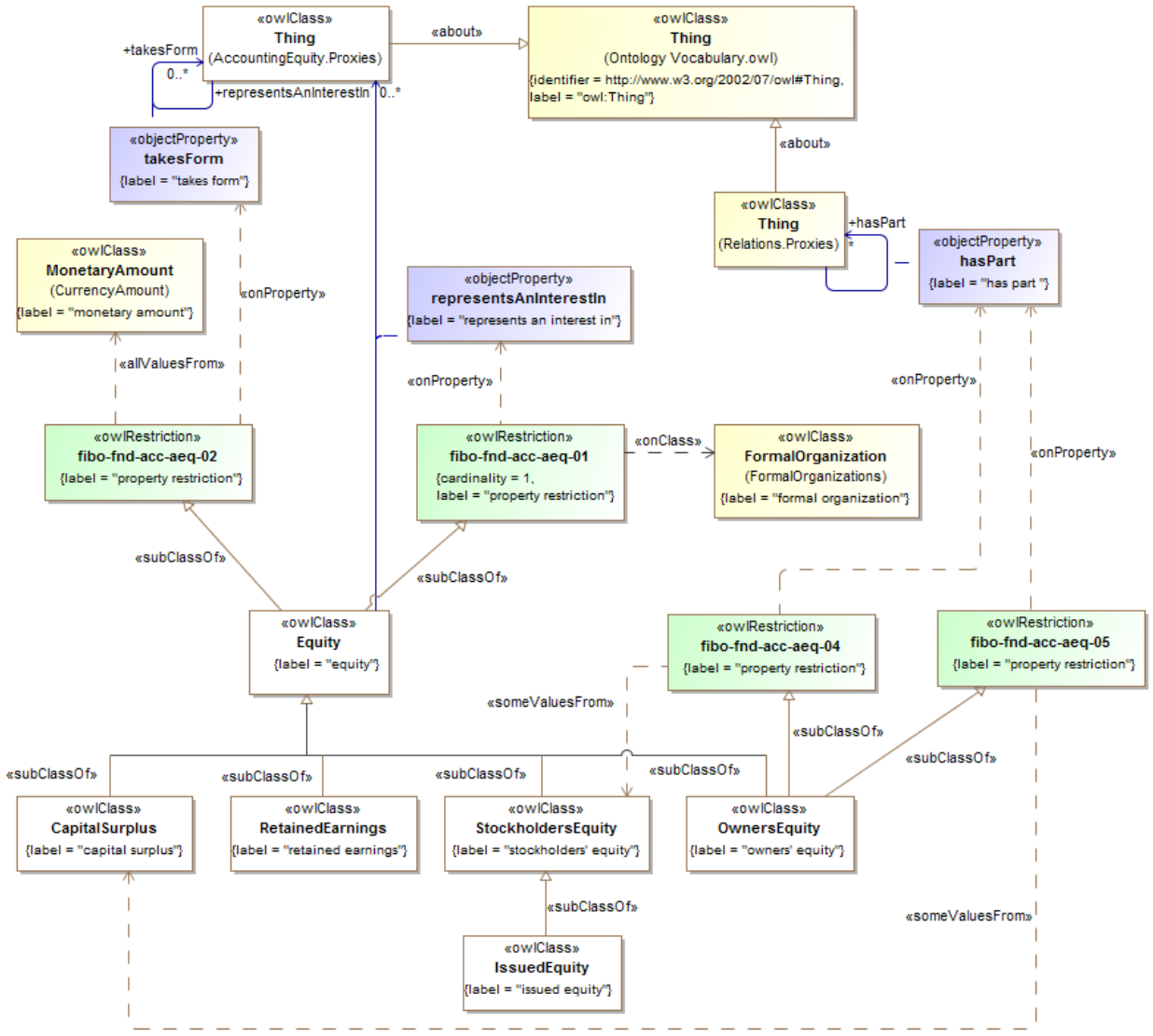

Figure 10.68 Equity Concepts

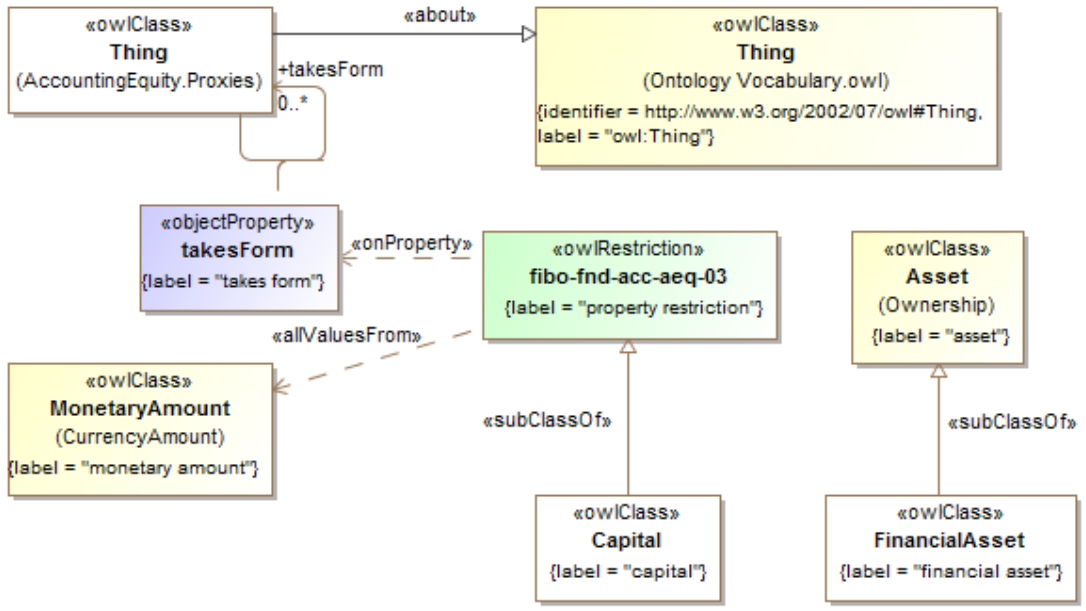

Figure 10.69 Capital and Asset

Table 10-76. Accounting Equity Ontology Metadata

Metadata Term	Value
sm:filename	Accounting Equity Ontology
sm:fileAbbreviation	fibo-fnd-acc-aeq
OntologyIRI	http://www.omg.org/spec/EDMC-FIBO/FND/Accounting/AccountingEquity/
owl:versionIRI	http://www.omg.org/spec/EDMC-FIBO/FND/20160201/Accounting/AccountingEquity/
sm:dependsOn	http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/AnnotationVocabulary/ http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/BusinessFacingTypes/ http://www.omg.org/spec/EDMC-FIBO/FND/Relations/Relations/ http://www.omg.org/spec/EDMC-FIBO/FND/AgentsAndPeople/Agents/ http://www.omg.org/spec/EDMC-FIBO/FND/Places/Locations/ http://www.omg.org/spec/EDMC-FIBO/FND/Places/Countries/ http://www.omg.org/spec/EDMC-FIBO/FND/Places/Addresses/ http://www.omg.org/spec/EDMC-FIBO/FND/GoalsAndObjectives/Goals/ http://www.omg.org/spec/EDMC-FIBO/FND/Organizations/Organizations/ http://www.omg.org/spec/EDMC-FIBO/FND/Organizations/FormalOrganizations/ http://www.omg.org/spec/EDMC-

Metadata Term	Value
	FIBO/FND/AgentsAndPeople/People/ http://www.omg.org/spec/EDMC-FIBO/FND/Parties/Roles/ http://www.omg.org/spec/EDMC-FIBO/FND/Parties/Parties/ http://www.omg.org/spec/EDMC-FIBO/FND/OwnershipAndControl/Ownership/ http://www.omg.org/spec/EDMC-FIBO/FND/Accounting/CurrencyAmount/

"The content of the AccountingEquity ontology is defined in Table 10-77, below."

Table 10-77. Accounting Equity Details

Classes

Name	Annotations	Class Expressions
Capital (capital)	Definition: financial capital, which represents obligations, and is liquidated as money for trade, and owned by legal entities Explanatory note: Financial capital is in the form of capital assets, traded in financial markets. Its market value is not based on the historical accumulation of money invested but on the perception by the market of its expected revenues and of the risk entailed. Definition origin: http://en.wikipedia.org/wiki/Capital_(economics).	Property Restriction: ∀ **takesForm**.**MonetaryAmount** (fibo-fnd-acc-aeq-03)
CapitalSurplus (capital surplus, additional paid in capital)	Definition: Capital surplus is that amount which a firm raises in excess of the par value (nominal value) of the shares (common stock). Explanatory note: Capital surplus is a term that frequently appears as a balance sheet item as a component of shareholders equity. Definition origin: http://en.wikipedia.org/wiki/Additional_paid_in_capital	Parent Class: Equity
Equity (equity)	Definition: the value of an ownership interest in property, including shareholders equity in a business Definition origin: http://en.wikipedia.org/wiki/Equity	Property Restriction: = 1 **representsAnInteresTIn**.**FormalOrganization** (fibo-fnd-acc-aeq-02) Property Restriction: ∀ **takesForm**.**MonetaryAmount** (fibo-fnd-acc-aeq-01)
FinancialAsset (financial asset)	Definition: An asset consisting of one or more financial instruments, treated as an asset	Parent Class: Asset
IssuedEquity (issued equity)	Definition: externally-held stockholders equity that may be transferred from one party to another	Parent Class: StockholdersEquity
OwnersEquity (owners' equity)	Definition: equity owned in some concern as recorded on the books of that concern	Parent Class: Equity Property Restriction: ∃ **has-**

Name	Annotations	Class Expressions
		Part.**StockholdersEquity** (fibo-fnd-acc-aeq-04) Property Restriction: ∃ **has-Part**.**CapitalSurplus** (fibo-fnd-acc-aeq-05)
RetainedEarnings (retained earnings)	Definition: the portion of net income which is retained by the corporation rather than distributed to its owners as dividends Explanatory note: If the corporation takes a loss, then that loss is retained and called variously retained losses, accumulated losses or accumulated deficit. Retained earnings and losses are cumulative from year to year with losses offsetting earnings. Definition origin: http://en.wikipedia.org/wiki/Retained_earnings	Parent Class: Equity
StockholdersEquity (stockholders' equity)	Definition: equity held in a concern by stockholders Explanatory note: When total assets are greater than total liabilities, stockholders have a positive equity (positive book value). Conversely, when total liabilities are greater than total assets, stockholders have a negative stockholders equity (negative book value, also sometimes called stockholders' deficit. Definition origin: http://en.wikipedia.org/wiki/Retained_earnings	Parent Class: Equity

Properties

Name	Annotations	Property Axioms
representsAnInterestIn (represents an interest in)	Definition: Equity always represents an interest in some business organization. This is the organization, company or venture in which the holder of the equity has a stake in by virtue of holding that equity	Domain: Equity
takesForm (takes form)	Definition: the form taken by some amount of money defined according to its purpose, such as capital or equity	

10.12.2 Ontology: Currency Amount

This ontology defines currency and monetary amount related concepts for use in defining other FIBO ontology elements. There are two distinct kinds of concepts that correspond to money and amounts: a concrete, actual amount of money, and the monetary measure of something denominated in some currency. These are dimensionally the same but whereas "money amount" is defined as an amount of money, "monetary amount" is an abstract monetary measure.

The definition of currency provided herein is compliant with the definitions given in ISO 4217. ISO 4217 provides universally applicable coded representations of names of currencies and funds, used internationally for financial transaction support. The ontology has been partitioned into 2 parts: (1) the essential concept system describing the standard (this module), and (2) ISO4217-1-CurrencyCodes, which contains all of the individuals specified in ISO 4217.

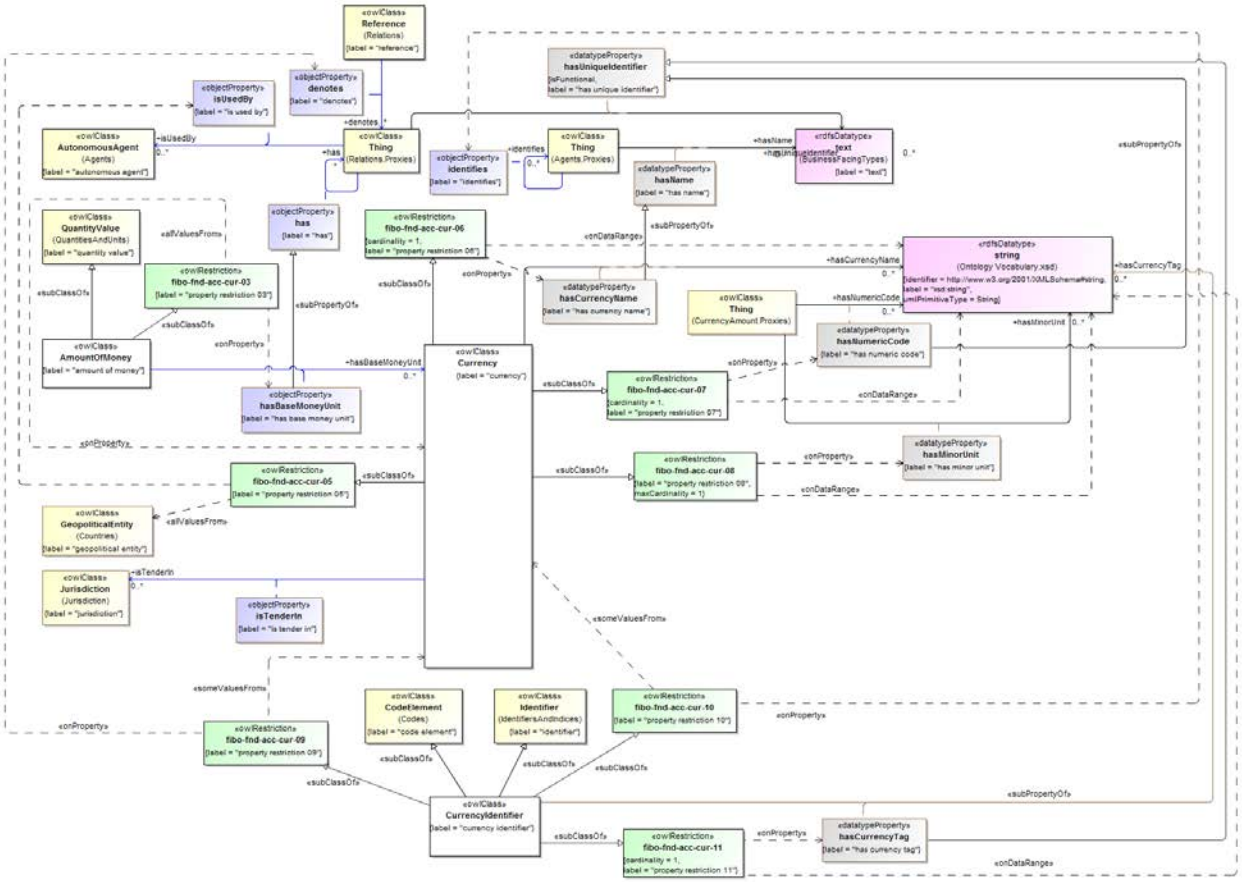

Figure 10.70 Definitions for Currency and Currency Identifiers

Figure 10.70 provides the complete definition of Currency and CurrencyIdentifier as specified in the ISO 4217 standard. It also depicts a MoneyAmount as a kind of QuantityValue that has a base money unit of Currency.

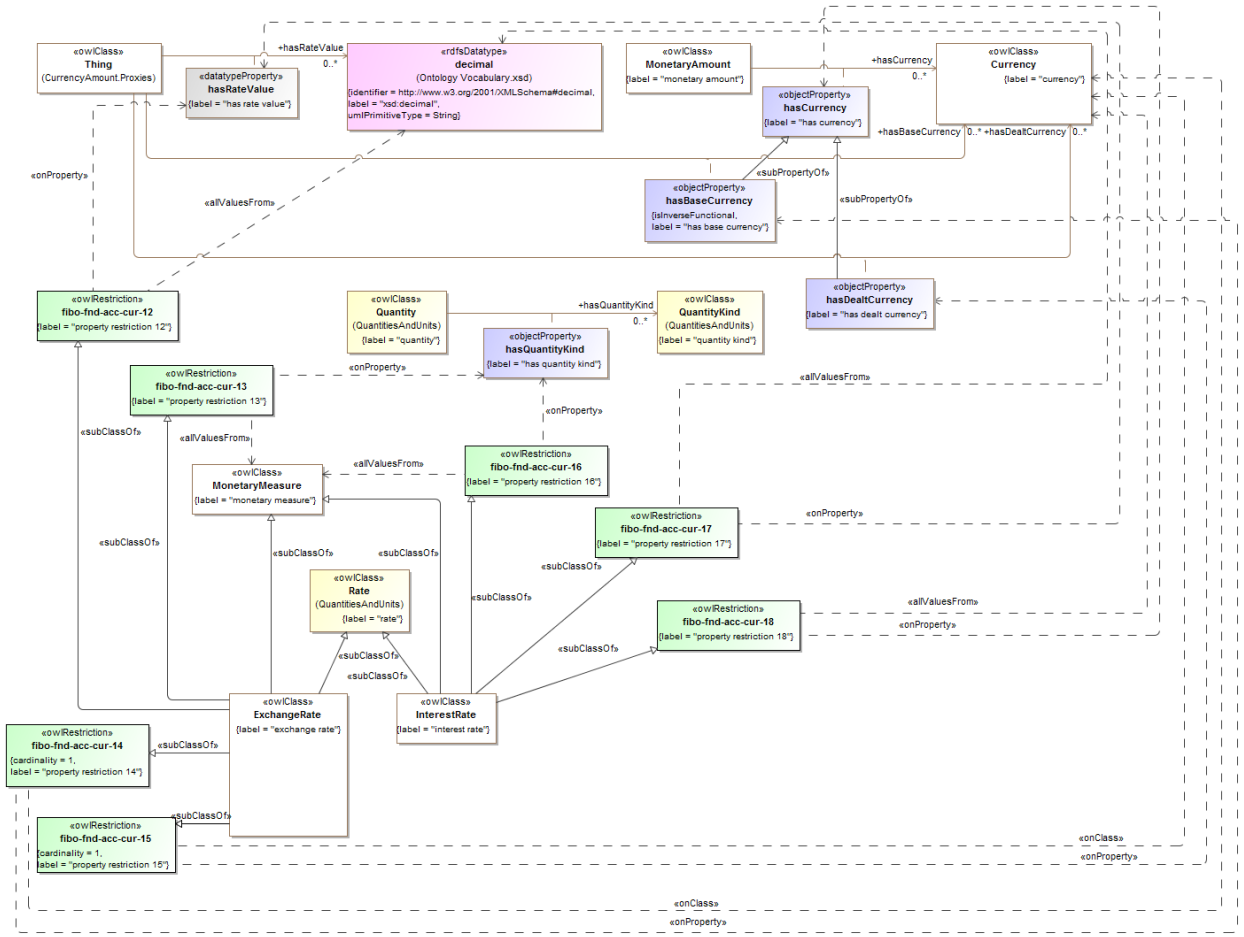

Figure 10.71 Definition of Exchange and Interest Rates

Figure 10.71 depicts definitions for interest and exchange rates for extension by other domain-specific concepts in FIBO, including but not limited to various reference rates (*e.g.*, the Fed Rate, LIBOR, EURIBOR, etc.).

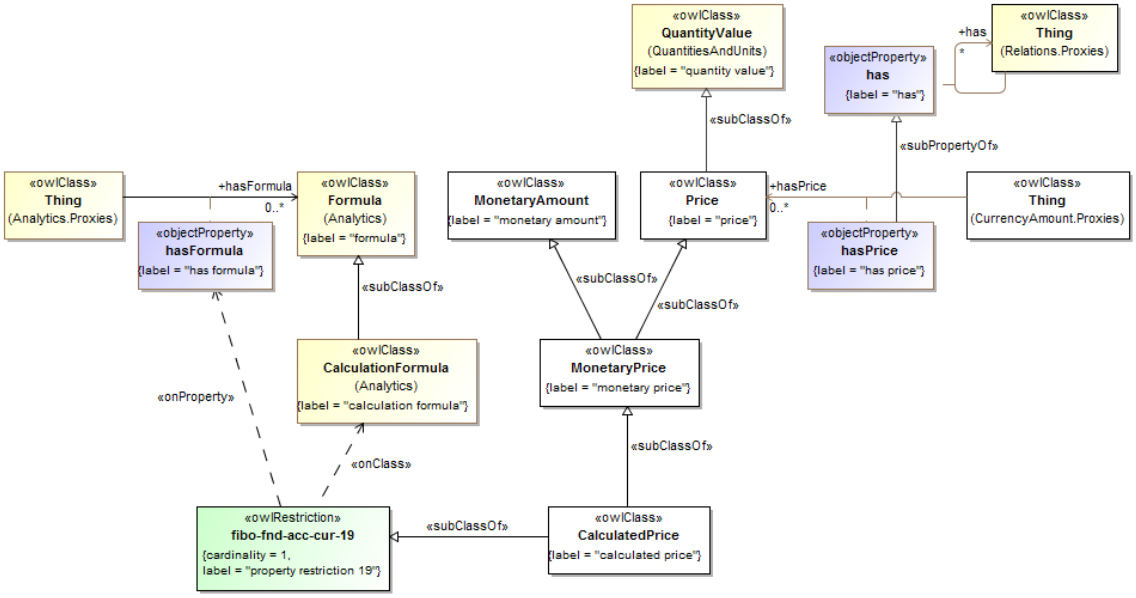

Figure 10.72 Definition of Price

Figure 10.72 shows the definitions for price, monetary price, and calculated price for extension by other domain-specific concepts in FIBO, including prices related to market valuation of securities, prices for financial services, prices associated with financial products including but not limited to securities, and so forth.

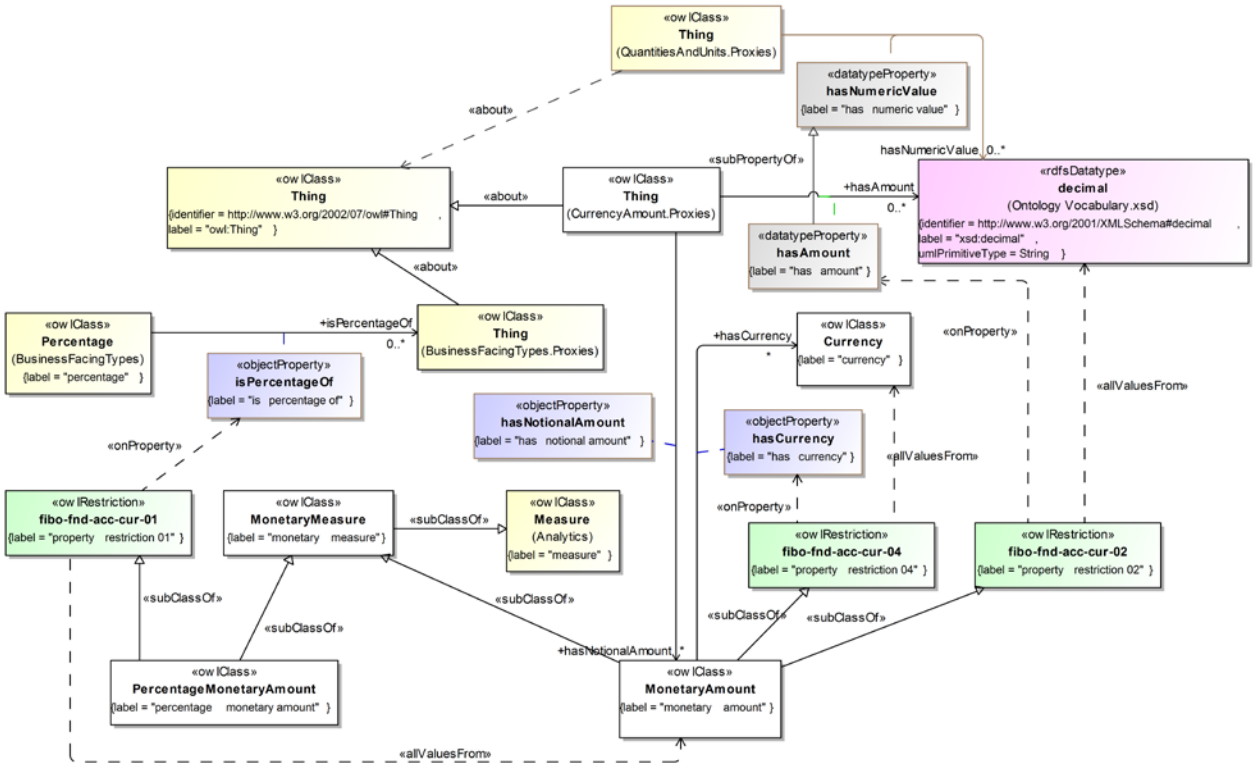

Figure 10.73Monetary Amounts and Measures

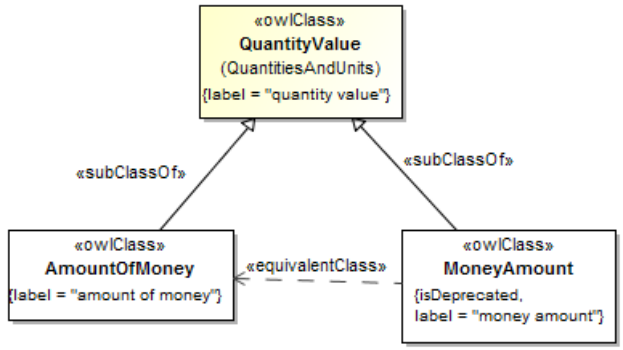

Figure 10.74 Deprecated Class Definition for MoneyAmount

The concept called MoneyAmount, shown in Figure 10.74 above, has been deprecated in favor of AmountOfMoney. Implementers should note that the MoneyAmount class will likely be eliminated from the FIBO Foundations standard in a future release.

Metadata defining the primary metadata elements for the CurrencyAmount ontology are given in Table 10-78, below.

Table 10-78. Currency Amount Ontology Metadata

Metadata Term	Value
sm:filename	CurrencyAmount.rdf

Metadata Term	Value
sm:fileAbbreviation	fibo-fnd-acc-cur
OntologyIRI	http://www.omg.org/spec/EDMC-FIBO/FND/Accounting/CurrencyAmount/
owl:versionIRI	http://www.omg.org/spec/EDMC-FIBO/FND/20160201/Accounting/CurrencyAmount/
sm:dependsOn	http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/AnnotationVocabulary/ http://www.omg.org/spec/EDMC-FIBO/FND/AgentsAndPeople/Agents/ http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/BusinessFacingTypes/ http://www.omg.org/spec/EDMC-FIBO/FND/Relations/Relations/ http://www.omg.org/spec/EDMC-FIBO/FND/Arrangements/Codes/ http://www.omg.org/spec/EDMC-FIBO/FND/Arrangements/IdentifiersAndIndices/ http://www.omg.org/spec/EDMC-FIBO/FND/Places/Countries/http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/Analytics http://www.omg.org/spec/EDMC-FIBO/FND/Quantities/QuantitiesAndUnits/

The content of the CurrencyAmount ontology is defined in Table 10-79, below.

Table 10-79. Currency Amount Details

Classes

Name	Annotations	Class Expressions
AmountOfMoney (amount of money, cash)	Definition: a sum of money Explanatory note: This is an actual sum of money, not the measure of a sum of money in monetary units, although it has the same basic properties (decimal number with a currency unit).	Parent Class: QuantityValue Property Restriction: ≥ 1 **hasBaseMoneyUnit.Currency** (fibo-fnd-acc-cur-03)
CalculatedPrice (calculated price)	Definition: a monetary price determined by a formula Adapted from: http://www.businessdictionary.com/definition/price.html	Parent Class: MonetaryPrice Property Restriction: = 1 **hasFormula.CalculationFormula** (fibo-fnd-acc-cur-19)
Currency (currency)	Definition: medium of exchange value, defined by reference to the geographical location of the authorities responsible for it Definition origin: Codes for the representation of currencies and funds, ISO 4217, Seventh edition, 2008-07-15, section 3.1.	Property Restriction: = 1 **hasCurrencyName.string** (fibo-fnd-acc-cur-06) Property Restriction: = 1 **hasNumericCode.string** (fibo-fnd-acc-cur-07) Property Restriction: ≤ 1 **hasMinorUnit.string** (fibo-fnd-acc-cur-08)

Name	Annotations	Class Expressions
		Property Restriction: ≤ 1 **representsRedenomination.boolean** (fibo-fnd-acc-cur-09) Property Restriction: ∀ **isUsedBy.GeopoliticalEntity** (fibo-fnd-acc-cur-05)
CurrencyIdentifier (currency identifier)	Definition: the trigraph representing the currency or fund Explanatory note: The first (left-most) two characters of the currency identifier provide a code unique to the currency authority to which it is assigned. Wherever practicable, it is derived from the geographical location of the currency authority, as described in ISO 3166. The third (right-most) character of the identifier (alphabetic code) is an indicator, preferably mnemonic, derived from the name of the major currency unit or fund. Where the currency is not associated with a single geographical entity as described in ISO 3166-1, a specially allocated identifier (alpha-2 code) will be used to describe the currency authority. This code will be allocated by the Maintenance Agency from within the user-assigned range of codes XA to XZ specified in 8.1.3 of ISO 3166-1:1997. The character following X will be a mnemonic, where possible, derived from the name of the geographical area concerned. Definition origin: http://www.omg.org/spec/DTV/1.1/	Parent Class: CodeElement Parent Class: Identifier Property Restriction: = 1 **hasCurrencyTag.string** (fibo-fnd-acc-cur-12) Property Restriction: ∀ **denotes.Currency** (fibo-fnd-acc-cur-10) Property Restriction: ∀ **identifies.Currency** (fibo-fnd-acc-cur-11)
ExchangeRate (exchange rate)	Definition: a rate at which one currency can be exchanged for another Example: The exchange rate between the U.S. dollar and British pound is distinct from the exchange rate between the U.S. dollar and the euro. Adapted from: Barron's Dictionary of Finance and Investment Terms, Ninth Edition, 2014.	Parent Class: MonetaryMeasure, Rate Property Restriction: ∀ **hasRateValue.decimal** (fibo-fnd-acc-cur-12) Property Restriction: ∀ **hasQuantityKind.MonetaryMeasure** (fibo-fnd-acc-cur-13) Property Restriction: = 1 **hasBaseCurrency.Currency** (fibo-fnd-acc-cur-14) Property Restriction: = 1 **hasDealtCurrency.Currency** (fibo-fnd-acc-cur-15)
InterestRate (interest rate)	Definition: an amount charged, expressed as a percentage of principal, by a lender to a borrower for the use of assets ExplanatoryNote: Interest rates are typically noted on an annual basis, known as the annual percentage rate (APR). The assets borrowed could include cash, consumer goods, and large assets such as a vehicle or building. The rate is derived by dividing the amount of interest by the amount of principal borrowed. Interest rates are quoted on bills, notes, bonds, credit cards, and many kinds of consumer and business loans. Adapted from: Barron's Dictionary of Finance and Investment Terms, Ninth Edition, 2014.	Parent Class: MonetaryMeasure, Rate Property Restriction: ∀ **hasRateValue.decimal** (fibo-fnd-acc-cur-17) Property Restriction: ∀ **hasQuantityKind.MonetaryMeasure** (fibo-fnd-acc-cur-16) Property Restriction: ∀ **hasCur-**

Name	Annotations	Class Expressions
		rency.Currency (fibo-fnd-acc-cur-18)
MonetaryAmount (monetary amount)	Definition: the measure which is an amount of money specified in monetary units Explanatory note: This is an abstract concept, not to be confused with a sum of money (Money Amount).	Parent Class: MonetaryMeasure Property Restriction: ∀ **hasAmount**.**decimal** (fibo-fnd-acc-cur-02) Property Restriction: ∀ **hasCurrency**.**Currency** (fibo-fnd-acc-cur-04)
MonetaryPrice (monetary price)	Definition: a price that that is expressed as a monetary amount ExplanatoryNote: As the consideration given in exchange for transfer of ownership, price forms the essential basis of commercial transactions. It may be fixed by a contract, left to be determined by an agreed upon formula at a future date, or discovered or negotiated during the course of dealings between the parties involved. In commerce, price is determined by what (1) a buyer is willing to pay, (2) a seller is willing to accept, and (3) the competition is allowing to be charged. Adapted from: http://www.businessdictionary.com/definition/price.html Adapted from: http://www.oxforddictionaries.com/definition/english/price	Parent Class: MonetaryAmount, Price
MonetaryMeasure (monetary measure)	Definition: some measure of some sum of money Explanatory note: This may be a measure expressed in terms of decimal plus currency, or it may be a measure expressed in terms of a percentage amount with reference to some other monetary amount or to some Money Amount (actual amount of money)..	Parent Class: Measure
MoneyAmount **(money amount)**	Definition: a sum of money Deprecated: true	Parent Class: QuantityValue Class Axiom: ≡ AmountOfMoney
PercentageMonetaryAmount (percentage monetary amount)	Definition: a measure of some amount of money expressed as a percentage of some other amount, some notional amount or some concrete money amount Explanatory note: This will have a relationship to what it is a percentage of. Alternatively and for some applications of this term, there may be an enumerated list of possible things it is a percentage of.	Parent Class: MonetaryMeasure Property Restriction: ∀ **isPercentageOf**.**MonetaryAmount** (fibo-fnd-acc-cur-01)
Price (price)	Definition: an amount of money, goods, or services requested, expected, required, or given in exchange for something else Adapted from: http://www.businessdictionary.com/definition/price.html Adapted from: http://www.oxforddictionaries.com/definition/english/price	Parent Class: QuantityValue

Properties

Name	Annotations	Property Axioms
hasBaseCurrency (has base currency)	Definition: a predicate indicating the base currency in an exchange rate; one unit of this currency represents R units of the dealt currency, where R is the exchange rate value	Parent Property: hasCurrency Object Property Axiom (Type): InverseFunctionalProperty Range: Currency

Name	Annotations	Property Axioms
hasBaseMoneyUnit (has base money unit)	Definition: the currency in which the money amount is denominated	Parent Property: has Domain: AmountOfMoney Range: Currency
hasCurrency (has currency)	Definition: the currency in which the monetary amount is defined	Parent Property: has Domain: AmountOfMoney Range: Currency
hasDealtCurrency (has dealt currency)	Definition: a predicate indicating the dealt currency in an exchange rate; R units of this currency represent one unit of the base currency	Parent Property: hasCurrency Range: Currency
hasNotionalAmount has notional amount)	Definition: has a notional value expressed as some monetary amount, that is a number and a currency in which that number is denominated	Parent Property: has Range: MonetaryAmount
hasPrice (has price)	Definition: indicates the value of something expressed as an amount of money or goods	Parent Property: has Range: Price
isTenderIn (is tender in)	Definition: a region or country in which the currency is exchangeable for goods and services Explanatory note: Commonly referred to also as legal tender, however this definition does not hold literally in some countries *e.g.* Scotland.	Domain: Currency Range: GeopoliticalEntity
hasAmount (has amount)	Definition: a total number or quantity	Parent Property: hasNumericValue Range: decimal
hasCurrencyName (has currency name)	Definition: provides the full currency name, including diacritical marks	Parent Property: hasName Domain: Currency Range: string
hasCurrencyTag (has currency tag)	Definition: relates a unique three-character string to the identifier for a currency	Parent Property: hasUniqueIdentifier Domain: CurrencyIdentifier Range: string
hasMinorUnit (has minor unit)	Definition: relates a code for the minor unit of currency to the currency or fund Scope note: Requirements sometimes arise for values to be expressed in terms of minor units of currency. When this occurs, it is necessary to know the decimal relationship that exists between the currency concerned and its minor unit. - 0 means that there is no minor unit for the currency; - 1, 2, and 3 signify a ratio of 10 to 1, 100 to 1 and 1000 to 1 respectively.	Domain: Currency Range: string
hasNumericCode (has numeric code)	Definition: relates a numeric code to the currency or fund Scope note: The numeric currency code is derived, where possible, from the United Nations Standard Country or Area Code. Additional codes to meet special requirements (as described in 5.1.3) and in respect of funds will be allocated as necessary from within the user-assigned range of codes 950 to 998. Funds codes are allocated in descending order commencing at 998.	Domain: Currency Range: string
hasRateValue (has rate value)	Definition: has a value for a rate expressed as a number (may be a percentage or raw number)	Range: decimal

Name	Annotations	Property Axioms
representsRedenomination (represents redenomination)	Definition: indicates that a currency has been redenominated Scope note: Where a currency is redenominated, the Maintenance Agency shall assign an alpha and a numeric code to the redenominated currency which will enable it to be distinguished from the original denomination of that currency.	Domain: Currency Range: boolean

10.13 Module: Dates and Times

Table 10-80. Dates and Times Module Metadata

Metadata Term	Value
sm:moduleName	Dates and Times
sm:moduleAbbreviation	fibo-fnd-dt
sm:moduleVersion	1.0
sm:moduleAbstract	This module includes ontologies describing date and time concepts which are of specific reference in financial services. These cover foundational date and time concepts in a form usable for financial subject matter ontologies, including occurrences and conventions for business days and the like. The business day convention concepts are to be further extended in specialized ontologies for securities and derivatives, building on the ontologies in this module.

10.13.1 Ontology: FinancialDates

This ontology provides definitions of date and schedule concepts for use in other FIBO ontologies.

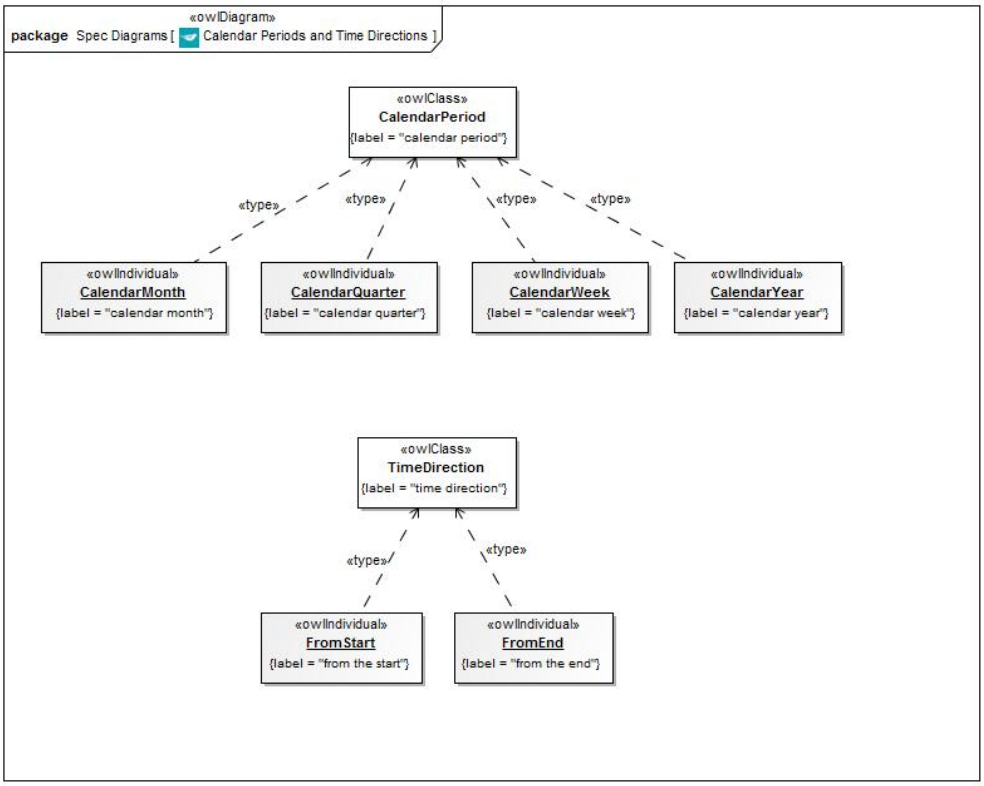

Figure 100.75 Calendar Periods and Time Directions

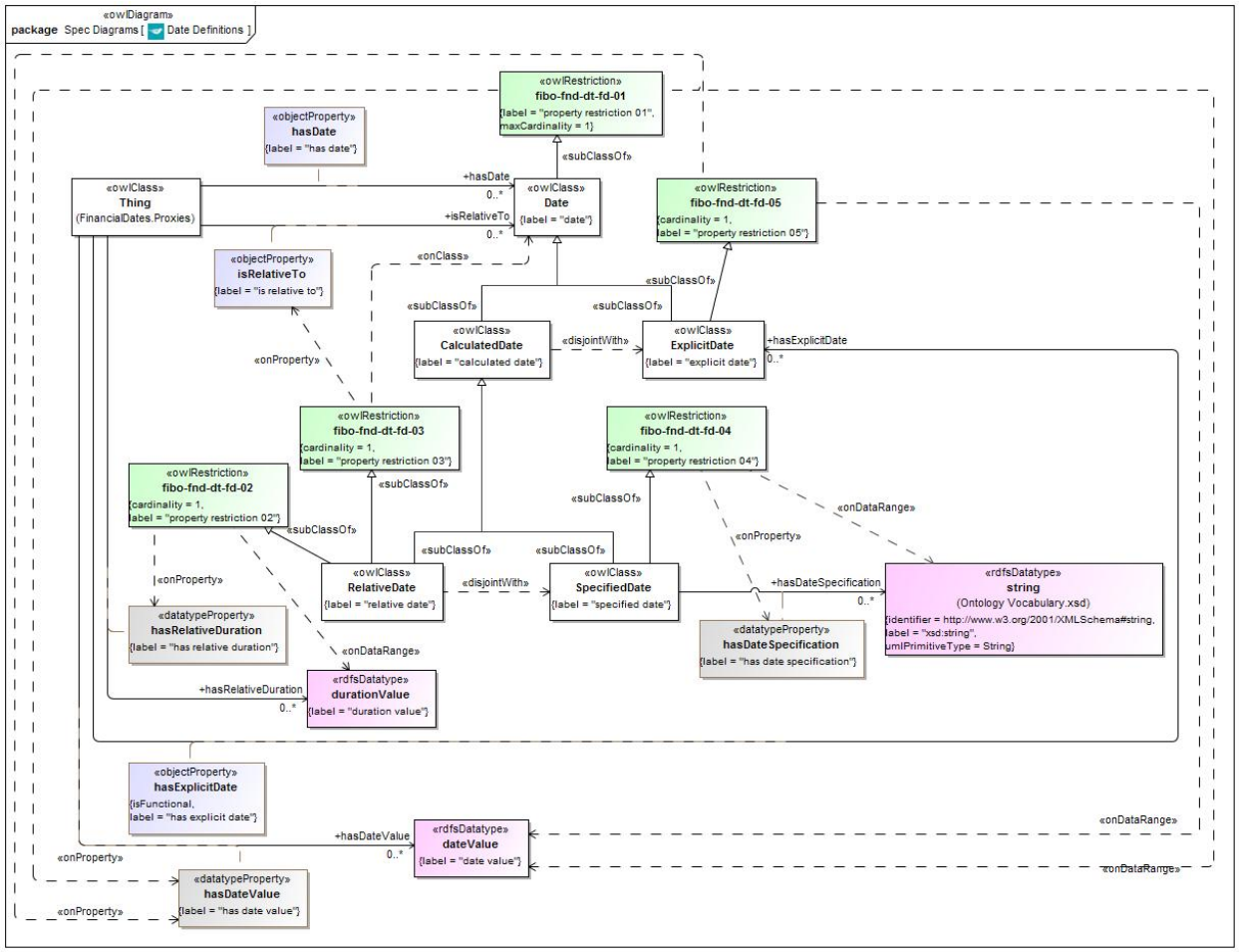

Figure 110.76 Date Definitions

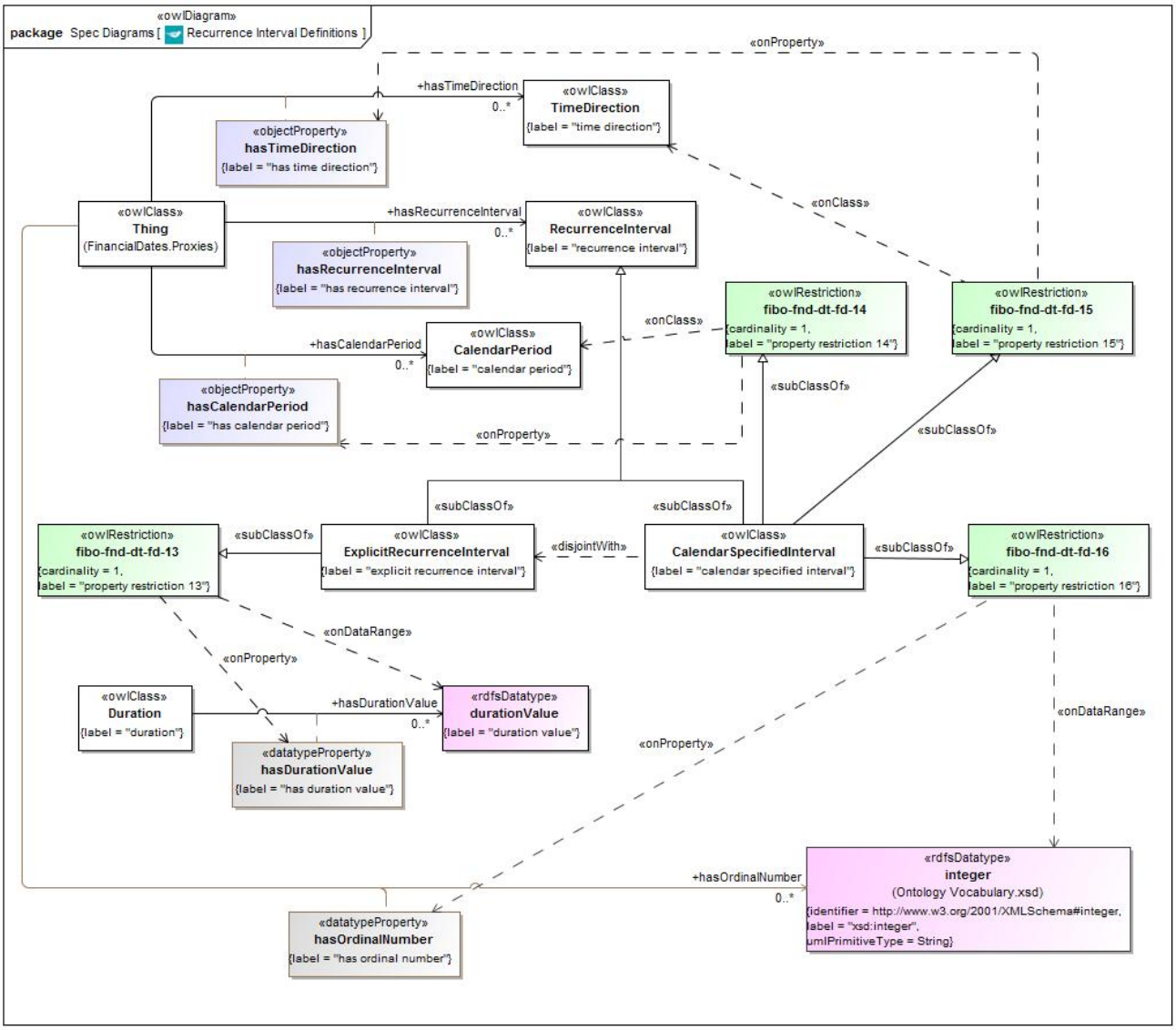

Figure 120.77 Recurrence Interval Definitions

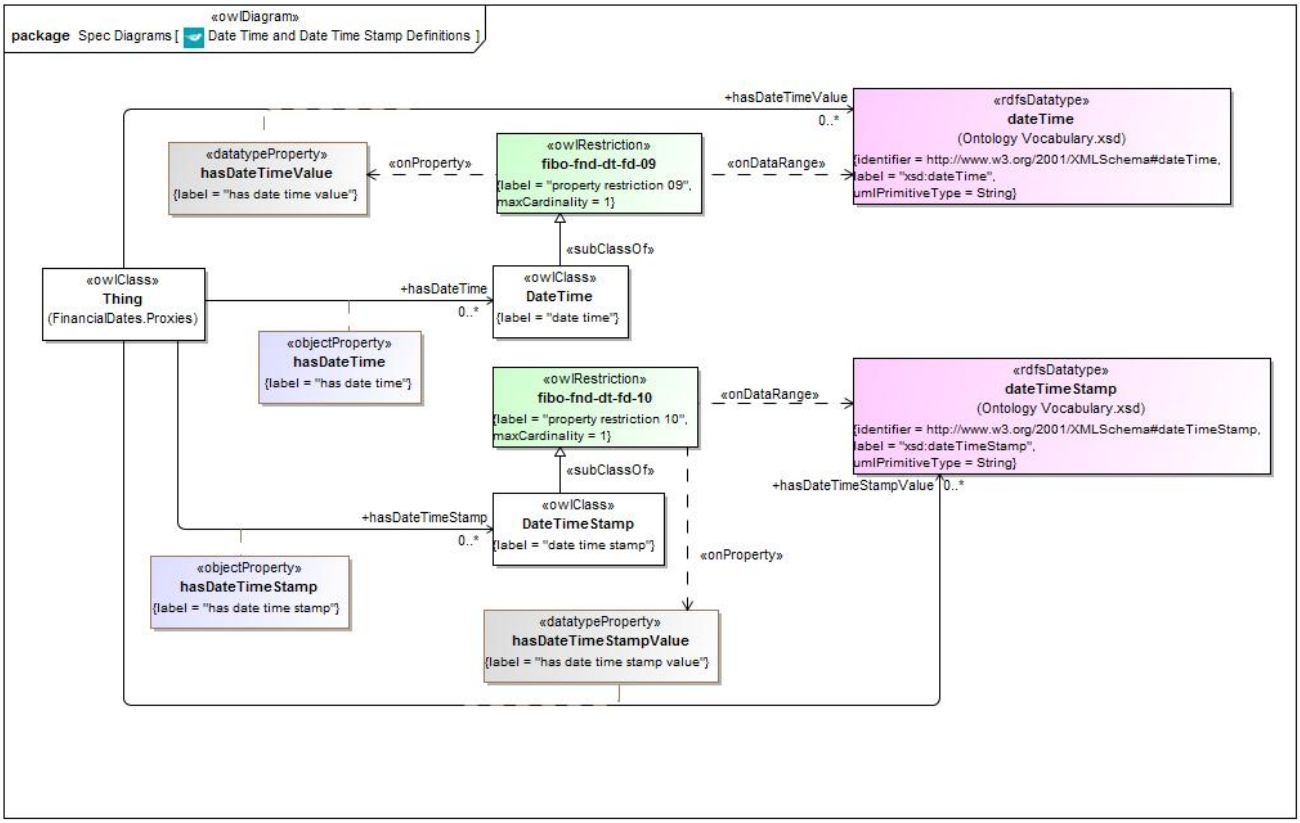

Figure 130.78 Date Time and Date Time Stamp Definitions

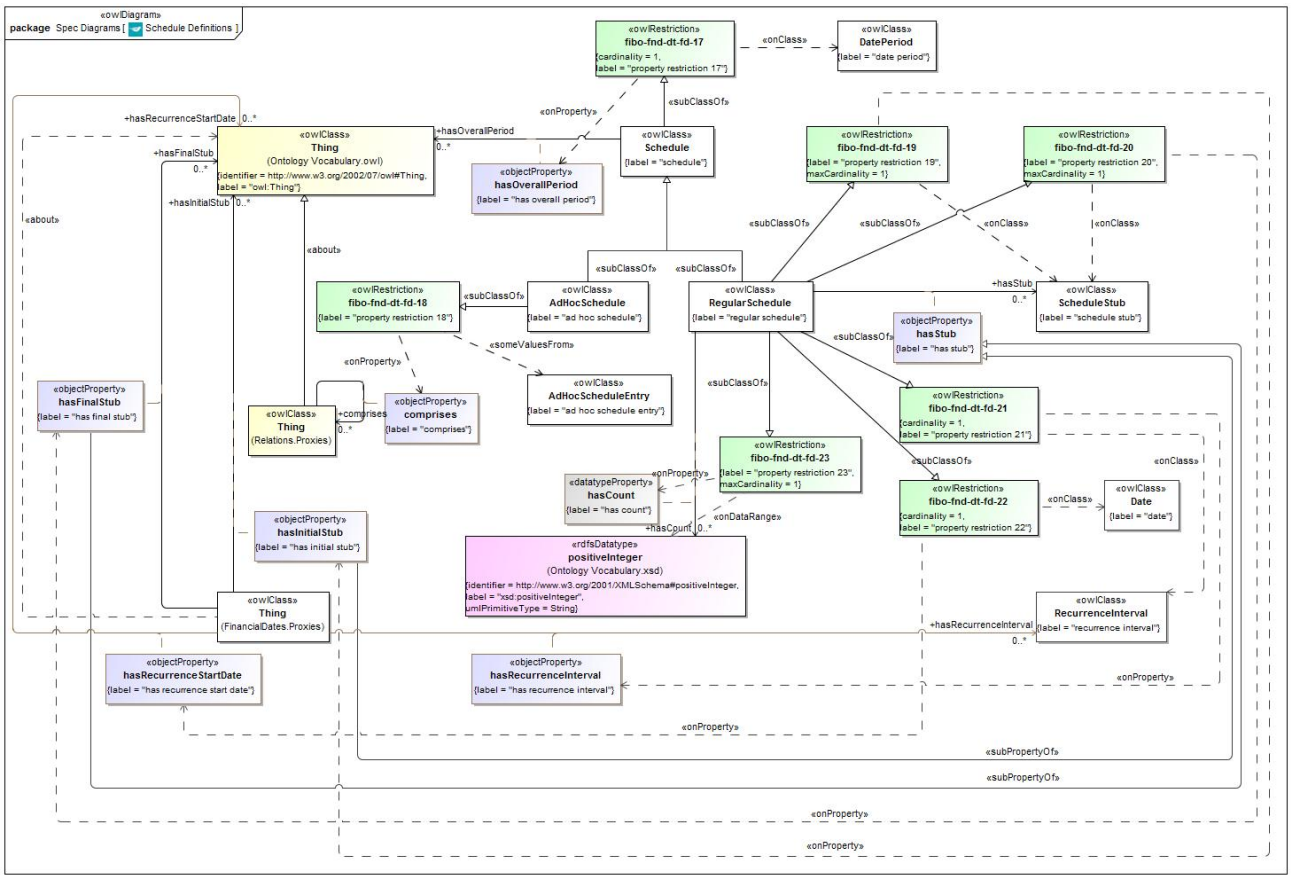

Figure 140.79 Schedule Definitions

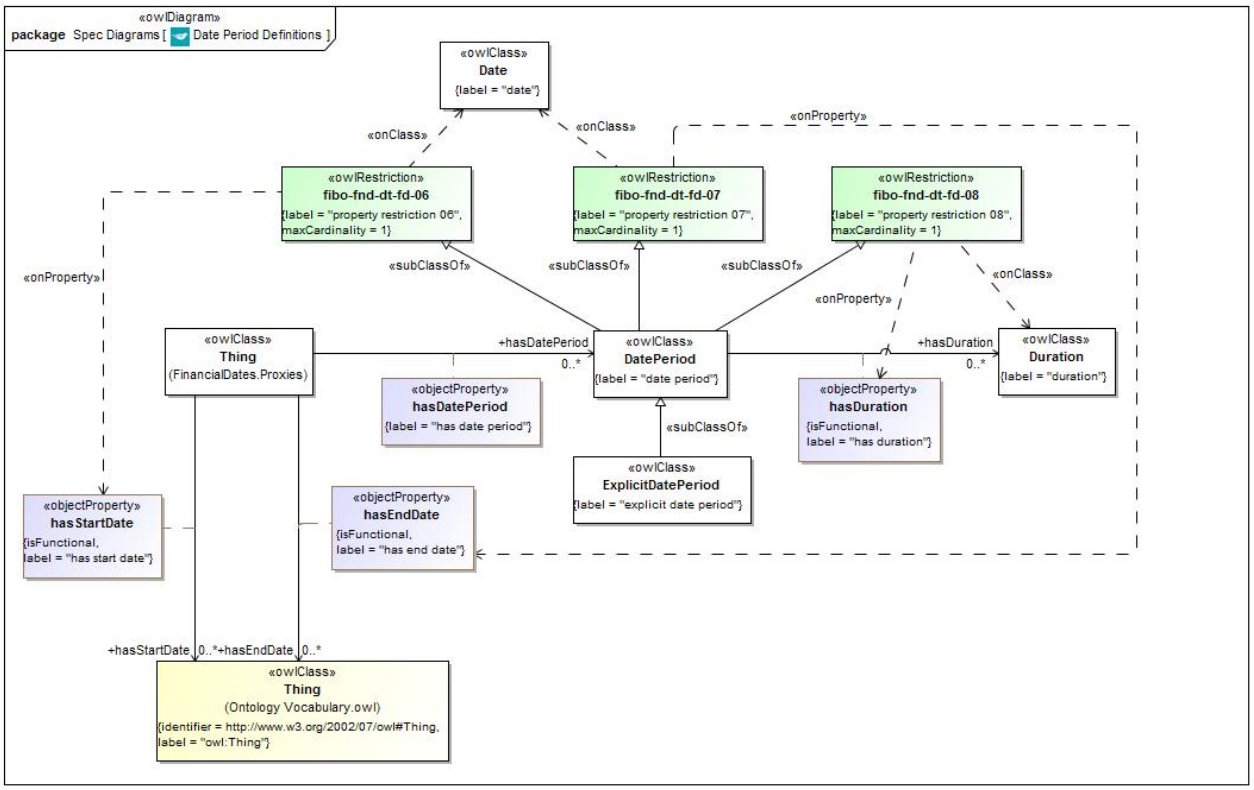

Figure 150.80 Date Period Definitions

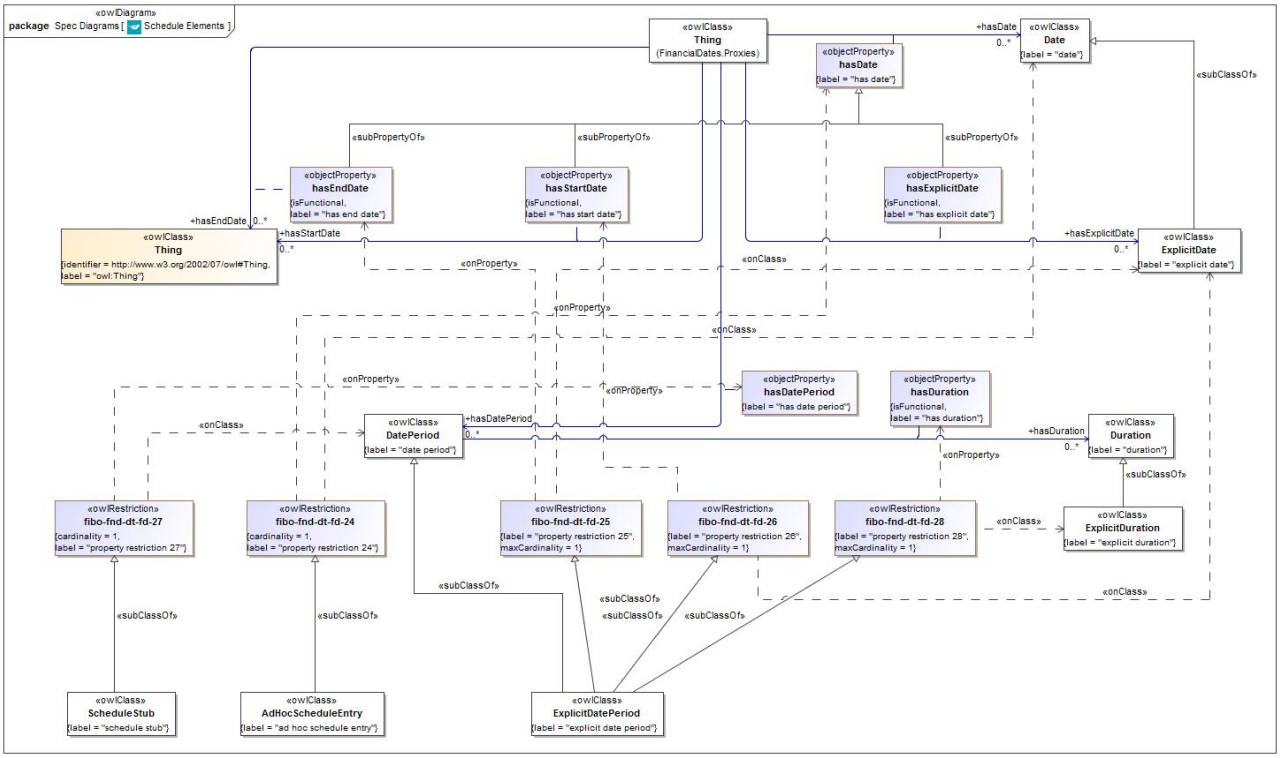

Figure 160.81 Schedule Elements

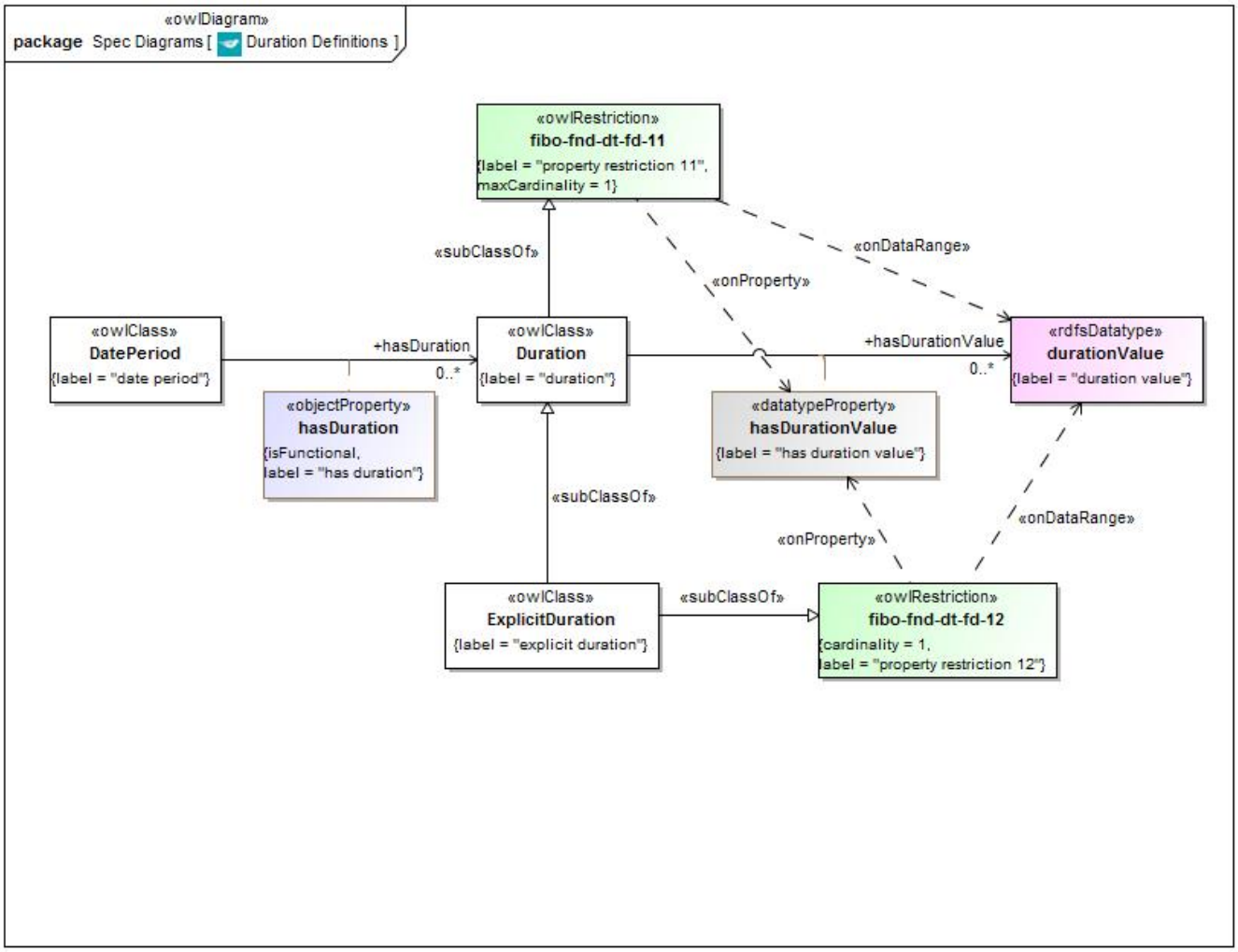

Figure 170.82 Duration Definitions

Table 10-81. FinancialDates Ontology Metadata

Metadata Term	Value
sm:filename	FinancialDates
sm:fileAbbreviation	fibo-fnd-dt-fd
OntologyIRI	http://www.omg.org/spec/EDMC-FIBO/FND/DatesAndTimes/FinancialDates/
owl:versionIRI	http://www.omg.org/spec/EDMC-FIBO/FND/20141101/DatesAndTimes/FinancialDates/
sm:dependsOn	http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/AnnotationVocabulary/ http://www.omg.org/spec/EDMC-FIBO/FND/Relations/Relations/

Table 10-82. Financial Dates Details

Classes

Name	Annotations	Class Expressions
Duration (duration)	Definition: An amount of time.	Property Restriction: ≤ 1 **hasDurationValue.durationValue** (fibo-fnd-dt-fd-11)
RegularSchedule (regular schedule)	Definition: A RegularSchedule is a Schedule that recurs regularly. Editorial Note: The BusinessDates ontology extends 'RegularSchedule' with an optional BusinessDayAdjustment that specifies what should happen if a scheduled date falls on a weekend or a holiday. Explanatory Note: A RegularSchedule is a Schedule defined as a set of Dates that start on a recurrence start date and repeat after each recurrence interval. The size of this set is defined by a count. The "initial ScheduleStub" associated with a RegularSchedule identifies any special treatment applied before the recurrence start date. Simiilarly, a "final ScheduleStub" identifies any special handling at the end of the recurrences. For example, a mortgage loan that is due each calendar month may have an initial payment due before the first calendar month, or a final payment due after the last monthly payment.	Parent Class: Schedule Property Restriction: ≤ 1 **hasInitialStub.ScheduleStub** (fibo-fnd-dt-fd-19) Property Restriction: ≤ 1 **hasFinalStub.ScheduleStub** (fibo-fnd-dt-fd-20) Property Restriction: = 1 **hasRecurrenceInterval.RecurrenceInterval** (fibo-fnd-dt-fd-21) Property Restriction: = 1 **hasRecurrenceStartDate.Date** (fibo-fnd-dt-fd-22) Property Restriction: = 1 **hasCount.positiveInteger** (fibo-fnd-dt-fd-23)
ExplicitDuration (explicit duration)	Definition: A Duration in which the 'hasSettledDuration' property is always set. This class is used when a Duration is guaranteed to be known when it is created.	Parent Class: Duration Property Restriction: = 1 **hasDurationValue.durationValue** (fibo-fnd-dt-fd-12)
CalendarPeriod (calendar period)	Definition: CalendarPeriod is an enumeration that indicates whether a CalendarSpecifiedDate is figured with respect to a calendar week, a calendar month, a calendar quarter, or a calendar year. Editorial Note: The terms "calendar xxx" are intended to reinforce that these are periods on a calendar, not durations. For example, a calendar year always starts on a January 1 and ends on a December 31. The term "calendar year" does not mean the same thing as a duration (an amount of time) of 1 year, nor can a calendar year start on any arbitrary day of a year. For example, a calendar year never starts on September 1. Similar points apply to other kinds of calendar periods, such as "calendar week", "calendar month", and "calendar quarter".	oneOf (*CalendarMonth*, *CalendarQuarter*, *CalendarWeek*, *CalendarYear*)
RecurrenceInterval (recurrence interval)	Definition: A RecurrenceInterval defines the time interval between each element of a RegularSchedule. Editorial Note: The BusinessDates ontology adds a 'BusinessRecurrenceInterval' subclass of RecurrenceInterval that specifies various intervals via an enumeration.	
Date (date)	Definition: A Date identifies a calendar day on some calendar.	Property Restriction: ≤ 1 **hasDateValue.dateValue** (fibo-fnd-dt-fd-01)
DateTimeStamp (date	Definition: A DateTimeStamp combines a Date, a time, and a time	Property Restriction: = 1

Name	Annotations	Class Expressions
time stamp)	zone.	**hasDateTimeStampValue.dateTimeStamp** (fibo-fnd-dt-fd-10)
AdHocScheduleEntry (ad hoc schedule entry)	Definition: An AdHocScheduleEntry identifies one Date among multiple AdHocScheduleEntries that jointly make up an AdHocSchedule.	Property Restriction: = 1 **hasDate.Date** (fibo-fnd-dt-fd-24)
ExplicitDate (explicit date)	Definition: An ExplicitDate is a Date in which the "hasDateValue" property is required.	Parent Class: Date Property Restriction: = 1 **hasDateValue.dateValue** (fibo-fnd-dt-fd-05)
SpecifiedDate (specified date)	Definition: A SpecifiedDate is a CalculatedDate that is defined by a rule, which is captured as a string by the "hasDateSpecification" property.	Parent Class: CalculatedDate Property Restriction: = 1 **hasDateSpecification.string** (fibo-fnd-dt-fd-04)
ExplicitDatePeriod (explicit date period)	Definition: A DatePeriod where the start date, end date, and duration are all explicit.	Parent Class: DatePeriod Property Restriction: ≤ 1 **hasEndDate.ExplicitDate** (fibo-fnd-dt-fd-25) Property Restriction: ≤ 1 **hasStartDate.ExplicitDate** (fibo-fnd-dt-fd-26) Property Restriction: ≤ 1 **hasDuration.ExplicitDuration** (fibo-fnd-dt-fd-28)
AdHocSchedule (ad hoc schedule)	Definition: An AdHocSchedule is a Schedule that consists of (comprises) AdHocScheduleEntries, each of which specifies a Date. Other ontologies can extend AdHocScheduleEntry to relate the Date to something.	Parent Class: Schedule Property Restriction: ∃ **comprises.AdHocScheduleEntry** (fibo-fnd-dt-fd-18)
DateTime (date time)	Definition: The combination of a Date and a time, without a time zone.	Property Restriction: = 1 **hasDateTimeValue.dateTime** (fibo-fnd-dt-fd-09)
DatePeriod (date period)	Definition: A time span over one or more calendar days, defined by at least two of three properties: 1. startDate 2. endDate 3. periodDuration If more than one of these properties is missing, the DatePeriod is invalid.	Property Restriction: ≤ 1 **hasStartDate.Date** (fibo-fnd-dt-fd-06) Property Restriction: ≤ 1 **hasEndDate.Date** (fibo-fnd-dt-fd-07) Property Restriction: ≤ 1 **hasDuration.Duration** (fibo-fnd-dt-fd-08)
CalculatedDate (calculated date)	Definition: A CalculatedDate is a Date that is or will be calculated in some way. The 'hasDateValue' property of a CalculatedDate is not set until the Date is calculated. Since the calculation may depend upon future events that may or may not ever happen, the 'hasDateValue' property may never be set. Editorial Note: The BusinessDates ontology extends 'CalculatedDate' with an optional BusinessDayAdjustment that specifies what should happen if a CalculatedDate falls on a weekend or a holiday.	Parent Class: Date Class Axiom: ¬ **ExplicitDate**
Schedule (schedule)	Definition: A Schedule is a table of Dates.	Property Restriction: = 1 **has-**

Name	Annotations	Class Expressions
	Explanatory Note: The overall period covers the entire DatePeriod of the Schedule, from the earliest Date to the final Date of the Schedule	**OverallPeriod.DatePeriod** (fibo-fnd-dt-fd-17)
CalendarSpecifiedInterval (calendar specified interval)	Definition: A CalendarSpecifiedInterval is a RecurrenceInterval that it is specified as the nth day of some CalendarPeriod (such as a calendar month), and a TimeDirection (forward from the beginning of the month, or backwards from the end). The nth day is an ordinal number, not a cardinal number. '1' means the first day of the calendar period.	Parent Class: RecurrenceInterval Property Restriction: = 1 **hasCalendarPeriod.CalendarPeriod** (fibo-fnd-dt-fd-14) Property Restriction: = 1 **hasTimeDirection.TimeDirection** (fibo-fnd-dt-fd-15) Property Restriction: = 1 **hasOrdinalNumber.integer** (fibo-fnd-dt-fd-16) Class Axiom: ¬ **ExplicitRecurrenceInterval**
TimeDirection (time direction)	Definition: TimeDirection is an enumeration class that indicates whether a CalendarSpecifiedDate is figured from the start or the end of a calendar period. The enumeration values of this class are modeled as instances of the class so that instances of CalendarSpecifiedDate can directly reference them.	oneOf (*FromStart*, *FromEnd*)
ExplicitRecurrenceInterval (explicit recurrence interval)	Definition: An ExplicitRecurrenceInterval defines a RecurrenceInterval via an ExplicitDuration.	Parent Class: RecurrenceInterval Property Restriction: = 1 **hasDurationValue.durationValue** (fibo-fnd-dt-fd-13)
ScheduleStub (schedule stub)	Definition: A ScheduleStub identifies a DatePeriod before the start of the recurring part of a Schedule or after the end of the recurring part, and an associated OccurrenceKind. Editorial Note: The Occurrences ontology extends ScheduleStub to 'comprise' an OccurrenceKind. The meaning is that a schedule stub comprises a date period and an event which is scheduled to occur during that date period; in other words that an Occurrence of the OccurrenceKind should happen during the DatePeriod of the ScheduleStub.	Property Restriction: = 1 **hasDatePeriod.DatePeriod** (fibo-fnd-dt-fd-27)
RelativeDate (relative date)	Definition: A RelativeDate is a CalculatedDate that is some Duration before or after another Date. When the 'hasRelativeDuration' property is negative, the RelativeDate is before the 'isRelativeTo' Date; otherwise the RelativeDate is after the 'isRelativeTo' Date.	Parent Class: CalculatedDate Property Restriction: = 1 **hasRelativeDuration.durationValue** (fibo-fnd-dt-fd-02) Property Restriction: = 1 **isRelativeTo.Date** (fibo-fnd-dt-fd-03) Class Axiom: ¬ **SpecifiedDate**

Properties

Name	Annotations	Property Axioms
hasCalendarPeriod (has calendar period)	Definition: CalendarPeriod identifies a calendar period used in computing a CalendarSpecifiedDate, such as a calendar week, calendar month, calendar quarter, or calendar year.	Parent Property: has Range: CalendarPeriod
hasCount (has count)	Definition: The count of the number of entries in a RegularSchedule.	Domain: RegularSchedule Range: xsd:positiveInteger
hasDate (has date)		Parent Property: has Range: Date
hasDatePeriod (has date period)		Parent Property: has Range: DatePeriod
hasDateSpecification (has date specification)	Definition: A rule that specifies how a SpecifiedDate is computed. Editorial Note: The rule is modeled as a simple String because OWL2 provides no way to model the semantics of such a rule.	Domain: SpecifiedDate Range: xsd:string
hasDateTime (has date time)		Parent Property: has Range: DateTime
hasDateTimeStamp (has date time stamp)		Parent Property: has Range: DateTimeStamp
hasDateTimeStampValue (has date time stamp value)		Range: xsd:dateTimeStamp
hasDateTimeValue (has date time value)		Range: xsd:dateTime
hasDateValue (has date value)	Definition: 'hasDatevValue' refers to actual date associated with any kind of Date, if the actual date has been established	Range: DateValue
hasDuration (has duration)	Definition: the duration of a DatePeriod	Parent Property: has Domain: DatePeriod Range: Duration
hasDurationValue (has duration value)	Definition: The amount of a Duration.	Domain: Duration Range: DurationValue
hasEndDate (has end date)	Definition: the ending date of some Schedule or DatePeriod	Parent Property: hasDate
hasExplicitDate (has explicit date)		Parent Property: hasDate Range: ExplicitDate
hasFinalStub (has final stub)	Definition: a final stub identifies any special period at the end of a RegularSchedule	Parent Property: hasStub
hasInitialStub (has initial stub)	Definition: An initial stub identifies any special period at the start of a RegularSchedule.	Parent Property: hasStub
hasOrdinalNumber (has ordinal number)	Definition: An ordinal number meaning 1st, 2nd, 3rd, etc. Explanatory Note: Negative ordinal numbers mean 1st before, 2nd before, etc.	Range: xsd:integer
hasOverallPeriod (has overall period)	Definition: the DatePeriod that includes all the Dates of a Schedule, including any ScheduleStubs	Parent Property: hasDatePeriod Domain: Schedule
hasRecurrenceInterval (has recurrence interval)		Parent Property: has Range: RecurrenceInterval
hasRecurrenceStartDate (has recur-	Definition: the starting Date of the first recurrence of a RegularSchedule	Parent Property: hasStartDate

Name	Annotations	Property Axioms
rence start date)		
hasRelativeDuration (has relative duration)	Definition: The Duration between two Dates. Explanatory Note: A relative duration may be negative.	Range: DurationValue
hasStartDate (has start date)	Definition: the starting Date of something	Parent Property: hasDate
hasStub (has stub)		Parent Property: has Domain:RegularSchedule Range: ScheduleStub
hasTimeDirection (has time direction)	Definition: A TimeDirection indicates whether a CalendarSpecified-Date is figured from the beginning or end of a calendar period.	Parent Property: has Range: TimeDirection
isRelativeTo (is relative to)	Definition: A RelativeDate or RelativeDatePeriod is defined relative to this Date.	Range: Date

10.13.2 Ontology: Occurrences

This ontology extends definitions of date and schedule concepts from the FinancialDates ontology with concepts defining occurrences (i.e., event-related concepts) for use in other FIBO ontologies.

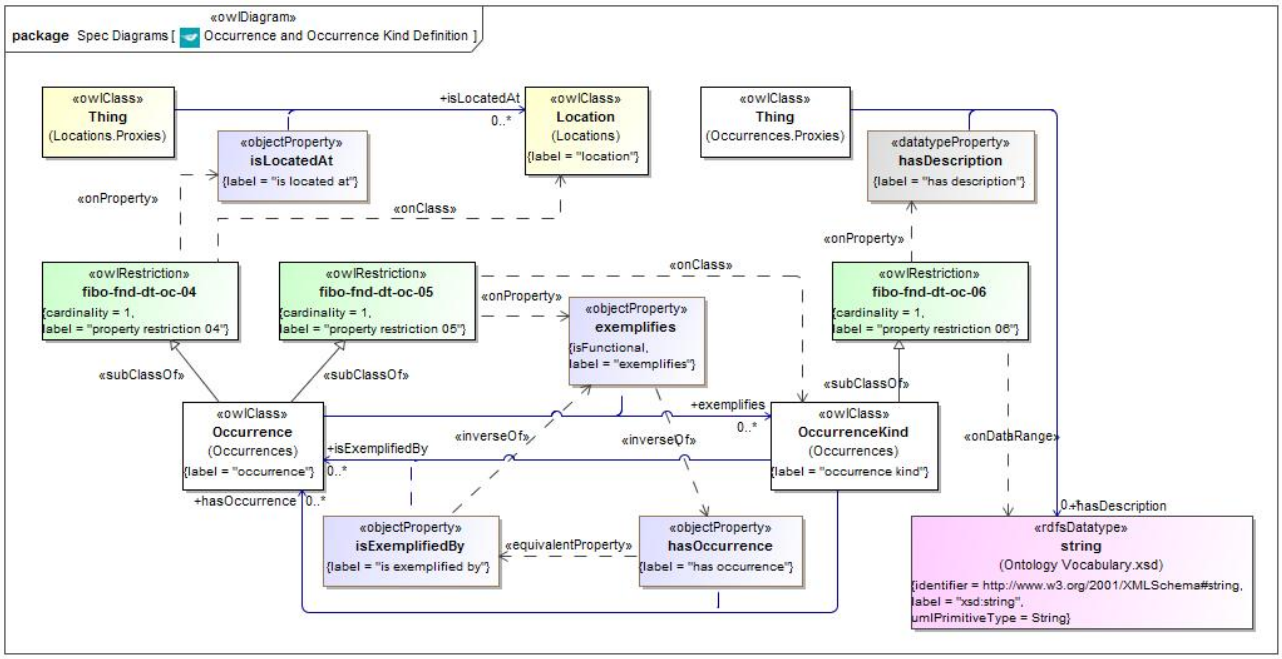

Figure 180.83 Occurrence and Occurrence Kind Definition

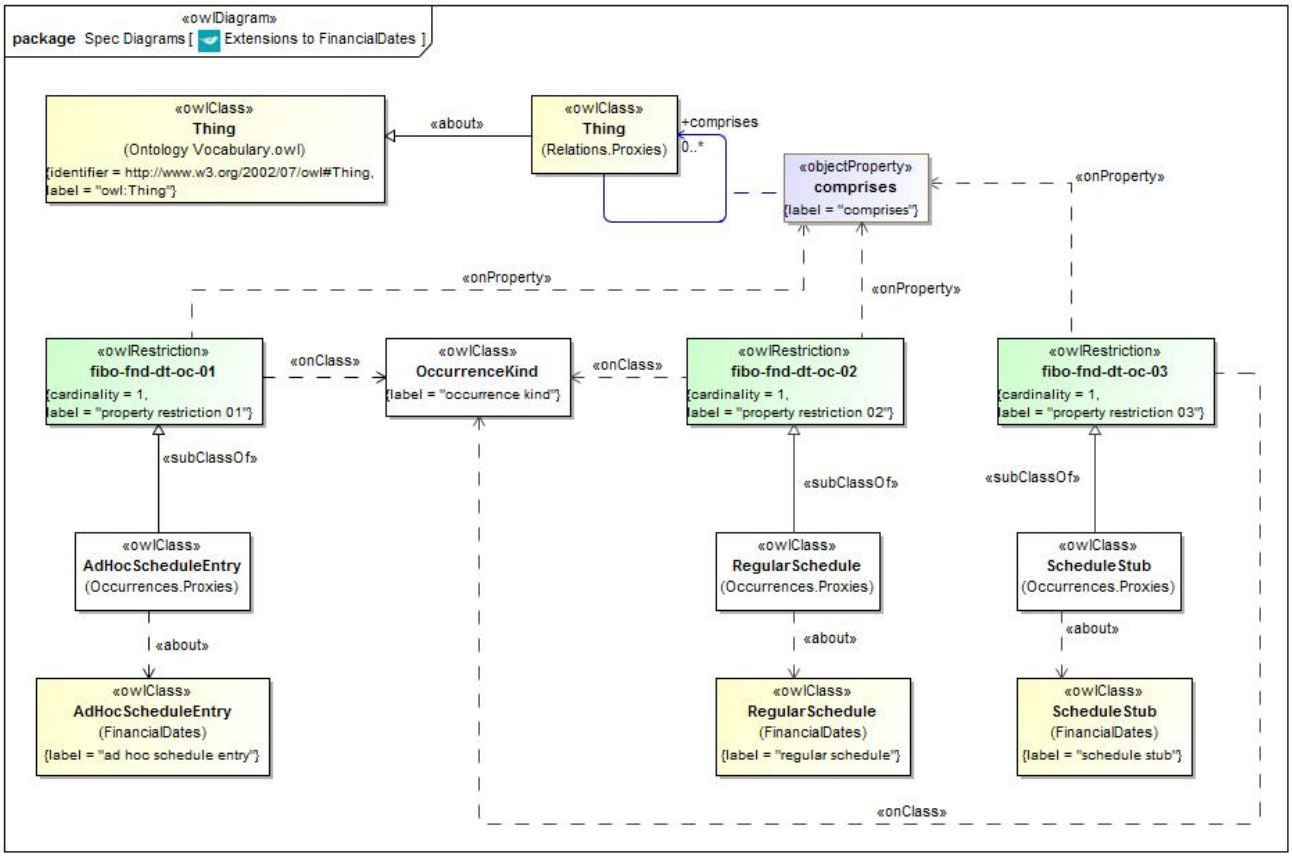

Figure 190.84 Extensions to FinancialDates

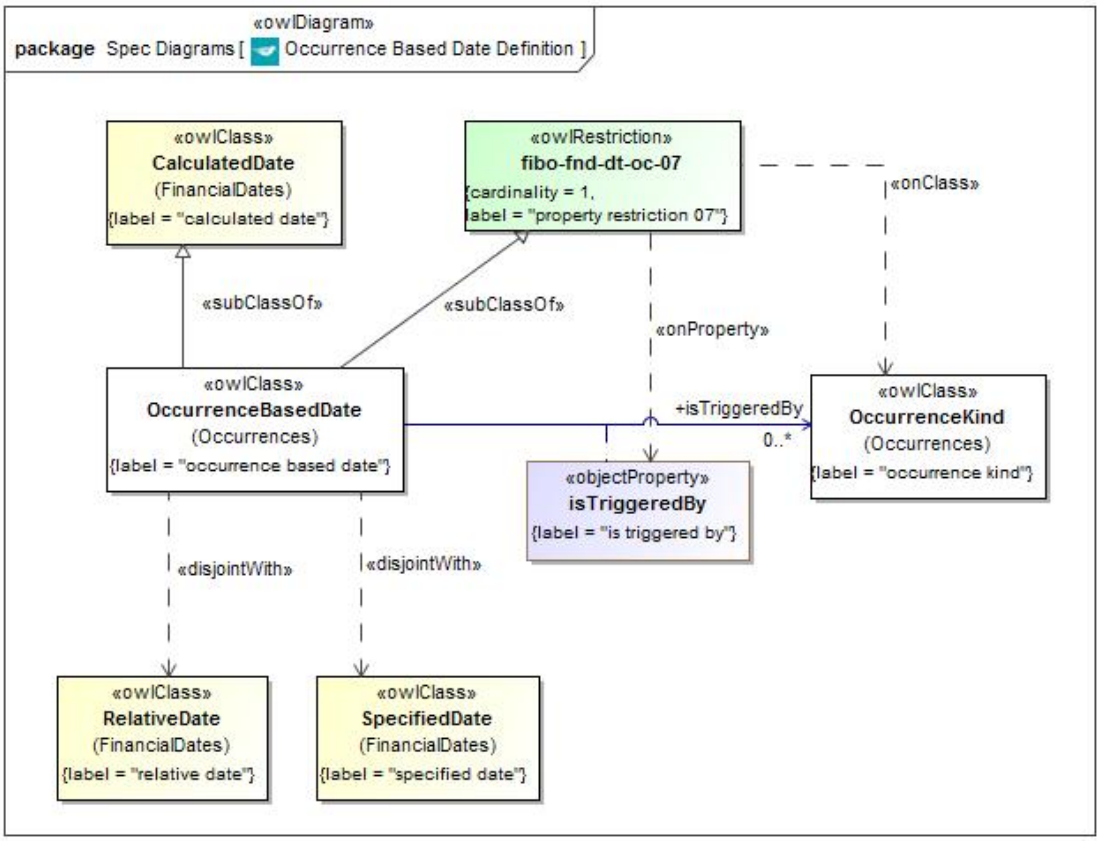

Figure 200.85 Occurrence Based Date Definition

Table 10-83. Occurrences Ontology Metadata

Metadata Term	Value
sm:filename	Occurrences
sm:fileAbbreviation	fibo-fnd-dt-oc
OntologyIRI	http://www.omg.org/spec/EDMC-FIBO/FND/DatesAndTimes/Occurrences/
owl:versionIRI	http://www.omg.org/spec/EDMC-FIBO/FND/20141101/DatesAndTimes/Occurrences/
sm:dependsOn	http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/AnnotationVocabulary/ http://www.omg.org/spec/EDMC-FIBO/FND/DatesAndTimes/FinancialDates/ http://www.omg.org/spec/EDMC-FIBO/FND/Places/Locations/ http://www.omg.org/spec/EDMC-FIBO/FND/Relations/Relations/

Table 10-84. Occurrences Details

Classes

Name	Annotations	Class Expressions
Occurrence (occurrence)	Definition: An Occurrence is a happening of an OccurrenceKind. Each Occurrence has a DateTimeStamp, which identifies when the Occurrence happened, and a Location (possibly virtual), that identifies where the Occurrence happened. Editorial Note: In order for other ontologies to accept FinancialDates without committing to the particular notions of 'Occurrence' and 'OccurrenceKind' that is modeled here, all aspects of Occurrences are captured in this ontology.	Property Restriction: = 1 **isLocatedAt.Location** (fibo-fnd-dt-oc-04) Property Restriction: = 1 **exemplifies.OccurrenceKind** (fibo-fnd-dt-oc-05)
OccurrenceBasedDate (occurrence based date)	Definition: An OccurrenceBaseDate is a CalculatedDate that is defined with respect to the Occurrence of some OccurrenceKind. The 'hasDateValue' property of an OccurrenceBasedDate is not set until the Occurrence happens. The 'triggeredBy' property relates an OccurrenceBasedDate to the OccurrenceKind that gives the meaning of the OccurrenceBasedDate.	Parent Class: CalculatedDate Property Restriction: = 1 **isTriggeredBy.OccurrenceKind** (fibo-fnd-dt-oc-07) Class Axiom: ¬ **RelativeDate** Class Axiom: ¬ **SpecificedDate**
OccurrenceKind (occurrence kind)	Definition: An OccurrenceKind is a type of event, which has a description. An OccurrenceKind may or may not ever happen, and thus does not have a Date. An OccurrenceKind happens as an Occurrence, which does have a Date. Editorial Note: In order for other ontologies to accept FinancialDates without committing to the particular notions of 'Occurrence' and 'OccurrenceKind' that is modeled here, all aspects of Occurrences are captured in this ontolog	Property Restriction: = 1 **hasDescription.string** (fibo-fnd-dt-oc-06)
AdHocScheduleEntry (ad hoc schedule entry)	Definition: Proxy for AdHocScheduleEntry	Property Restriction: = 1 **comprises.OccurrenceKind** (fibo-fnd-dt-oc-01)
RegularSchedule (regular schedule)	Definition: Proxy for RegularSchedule	Property Restriction: = 1 **comprises.OccurrenceKind** (fibo-fnd-dt-oc-02)
ScheduleStub (schedule stub)	Definition: Proxy for ScheduleStub	Property Restriction: = 1 **comprises.OccurrenceKind** (fibo-fnd-dt-oc-03)

Properties

Name	Annotations	Property Axioms
hasDescription (has description)	Definition: a textual description of something	Range: string
isTriggeredBy (is triggered by)	Definition: An OccurrenceBasedDate is triggered by an Occurrence that exemplifies the OccurrenceKind.	Domain: OccurrenceBasedDate Range: OccurrenceKind
hasOccurrence (has occurrence)	Definition: identifies occurrences of a given occurrence kind	Parent Property: has Domain: OccurrenceKind Range: Occurrence
isExemplifiedBy (is exemplified by)	Definition: identifies examples of a given concept	Domain: OccurrenceKind Range: Occurrence Inverse: exemplifies
exemplifies (exemplifies)	Definition: illustrates by example	Domain: Occurrence Range: OccurrenceKind Inverse: hasOccurrence

10.13.3 Ontology: BusinessDates

This ontology extends definitions of date and schedule concepts from the FinancialDates ontology with concepts defining dates that may be adjusted when they fall on weekends or holidays as defined in a given business center, for use in other FIBO ontologies.

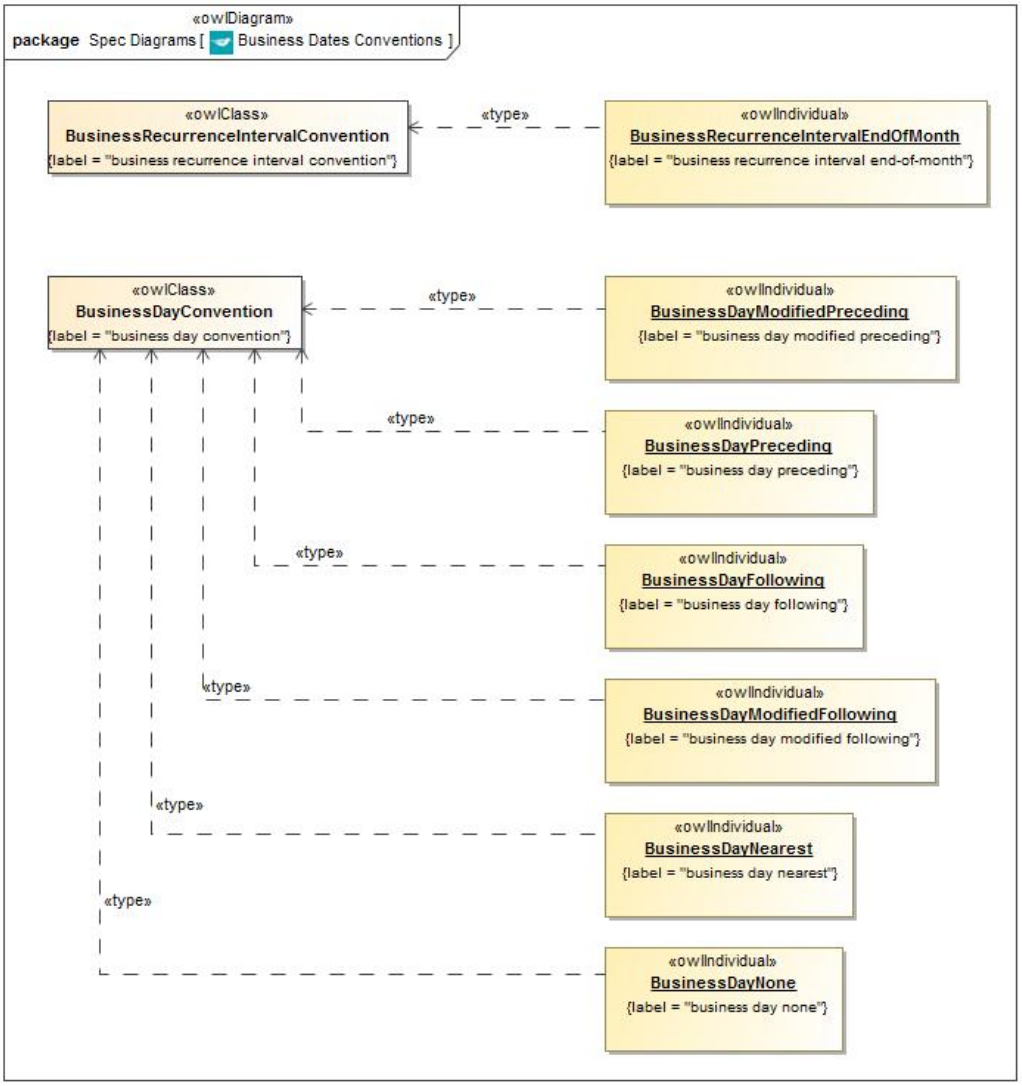

Figure 210.86 Business Dates Conventions

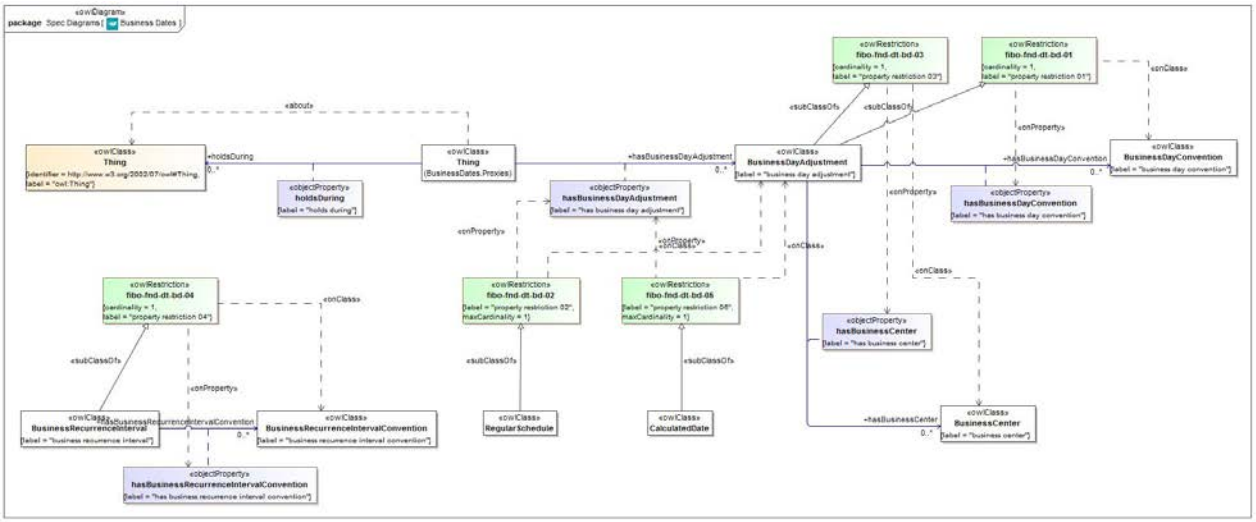

Figure 220.87 Business Dates

Table 10-85. BusinessDates Ontology Metadata

Metadata Term	Value
sm:filename	BusinessDates
sm:fileAbbreviation	fibo-fnd-dt-bd
OntologyIRI	http://www.omg.org/spec/EDMC-FIBO/FND/DatesAndTimes/BusinessDates/
owl:versionIRI	http://www.omg.org/spec/EDMC-FIBO/FND/20141101/DatesAndTimes/BusinessDates/
sm:dependsOn	http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/AnnotationVocabulary/ http://www.omg.org/spec/EDMC-FIBO/FND/DatesAndTimes/FinancialDates/ http://www.omg.org/spec/EDMC-FIBO/FND/Places/Countries/ http://www.omg.org/spec/EDMC-FIBO/FND/Relations/Relations/

Table 10-86. Business Dates Details

Classes

Name	Annotations	Class Expressions
BusinessRecurrenceInterval (business recurrence interval)	Definition: A BusinessRecurrenceInterval is a RecurrenceInterval that is specified using a BusinessRecurrenceIntervalConvention.	Parent Class: RecurrenceInterval Property Restriction: = 1 **hasBusinessRecurrenceIntervalConvention.BusinessRecurrenceInterv**

Name	Annotations	Class Expressions
		alConvention (fibo-fnd-dt-bd-04)
BusinessDayConvention (business day convention)	Definition: BusinessDayConvention is an enumeration of the possible ways to handle a Date that falls on a weekend or holiiday. BusinessDayTreatment combines a BusinessCenter with a BusinessDayConvention to determine what to do when business is not conducted in a particular business center on a particular calendar day.	
BusinessDayAdjustment (business day adjustment)	Definition: A BusinessDayAdjustment uses a BusinessDayConvention to specify what happens when a Date falls on a day that is a weekend or a holiday in some BusinessCenter.	Property Restriction: = 1 **hasBusinessDayConvention.BusinessDayConvention** (fibo-fnd-dt-bd-01) Property Restriction: = 1 **hasBusinessCenter.BusinessCenter** (fibo-fnd-dt-bd-03)
BusinessRecurrenceIntervalConvention (business recurrence interval convention)	Definition: BusinessRecurrenceIntervalConvention models various conventions for recurring days, such as "end of month", and "TBill auction date". These conventions are modelled as instances of the BusinessRecurrenceIntervalConvention class.	
RegularSchedule	Proxy for RegularSchedule	Property Restriction: ≤ 1 **hasBusinessDayAdjustment.BusinessDayAdjustment** (fibo-fnd-dt-bd-02)
CalculatedDate	Proxy for CalculatedDate	Property Restriction: ≤ 1 **hasBusinessDayAdjustment.BusinessDayAdjustment** (fibo-fnd-dt-bd-05)

Properties

Name	Annotations	Property Axioms
hasBusinessDayConvention (has business day convention)	Definition: A BusinessDayConvention identifies how a Date should be handled wihen it falls on a day that is not a business day.	Parent Property: has Domain: BusinessDayAdjustment Range: BusinessDayConvention
hasBusinessDayAdjustment (has business day adjustment)		Parent Property: has Range: BusinessDayAdjustment
holdsDuring (holds during)	Definition: a relationship that states that some condition or state holds during a specified date period	Parent Property: hasDatePeriod
hasBusinessCenter (has business center)	Definition: the location where business is conducted, and hence the business calendar used to adjust dates.	Parent Property: has Domain: BusinessDayAdjustment Range: BusinessCenter
hasBusinessRecurrenceIntervalConvention (has business recurrence interval convention)		Parent Property: has Domain: BusinessRecurrenceInterval Range: BusinessRecurrenceIntervalConvention

10.14 Module: Quantities

Metadata defining the primary metadata elements for the Quantities module are given in Table 10-87, below.

Table 10-87 Quantities Module Metadata

Metadata Term	Value
sm:moduleName	Quantities
sm:moduleAbbreviation	FIBO-FND-QT
sm:moduleVersion	1.0
sm:moduleAbstract	This module contains ontologies that define concepts related to quantities, units, dimensions, and quantity values.

10.14.1 Ontology: QuantitiesAndUnits

This ontology provides an initial set of concepts supporting the representation of quantities, units, systems of quantities, and systems of units for use in FIBO. It is compatible with and can be mapped directly to the OMG Date Time Vocabulary (DTV) Quantities Ontology, but has been integrated into FND to provide local coverage of quantities and measurements and eliminate the SBVR mark-up.

Metadata defining the primary metadata elements for the QuantitiesAndUnits ontology are given in Table 10-88, below.

Table 10-88 QuantitiesAndUnits Ontology Metadata

Metadata Term	Value
sm:filename	QuantitiesAndUnits.rdf
sm:fileAbbreviation	fibo-fnd-qt-qtu
OntologyIRI	http://www.omg.org/spec/EDMC-FIBO/FND/Quantities/QuantitiesAndUnits/
owl:versionIRI	http://www.omg.org/spec/EDMC-FIBO/FND/20160201/Quantities/QuantitiesAndUnits/
sm:dependsOn	http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/AnnotationVocabulary/ http://www.omg.org/spec/EDMC-FIBO/FND/Relations/Relations/ http://www.omg.org/spec/EDMC-FIBO/FND/Arrangements/Arrangements/

Figure 10.88 provides an overview of the concepts defined in the QuantitiesAndUnits ontology. Subsequent diagrams provide more detailed views on several of the concepts, with a focus on the logical restrictions that further define them.

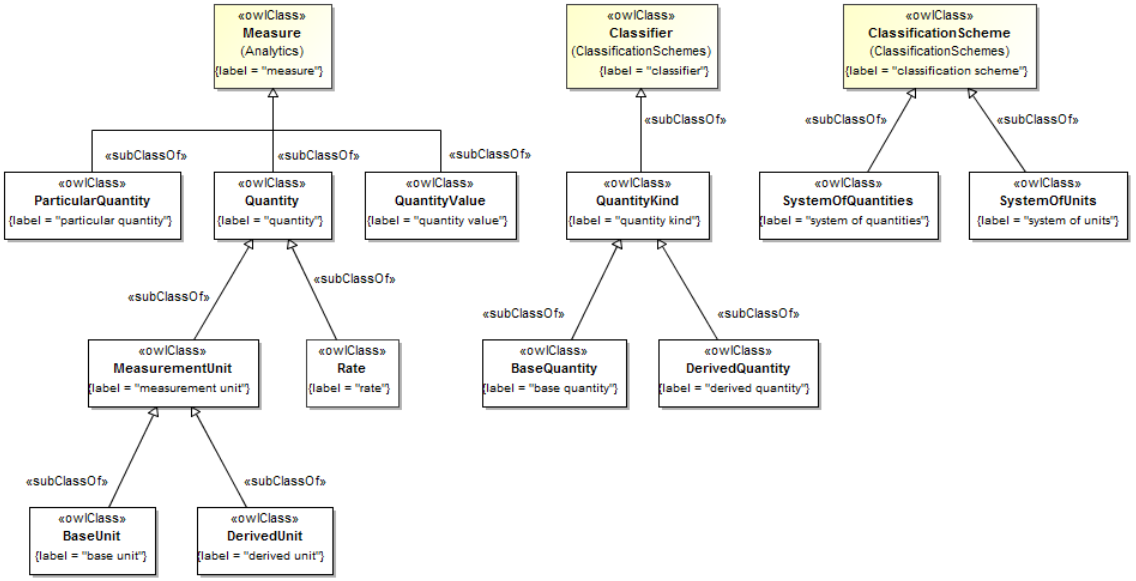

Figure 10.88 QuantitiesAndUnits Ontology Overview

Figure 10.89 defines quantities in more detail, including the properties and restrictions relevant to the Quantity, MeasurementUnit, BaseUnit, and DerivedUnit classes.

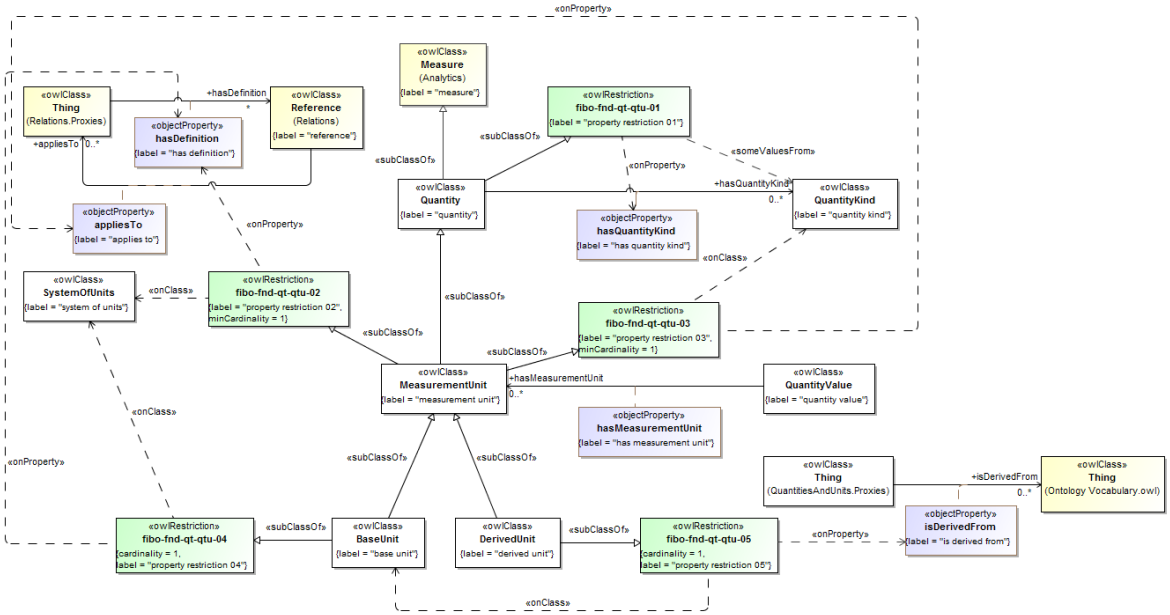

Figure 10.89 Definitions for Quantities and Measurement Units

Figure 10.90 defines quantity kinds in more detail, including the properties and restrictions relevant to the QuantityKind, BaseQuantity, and DerivedQuantity classes.

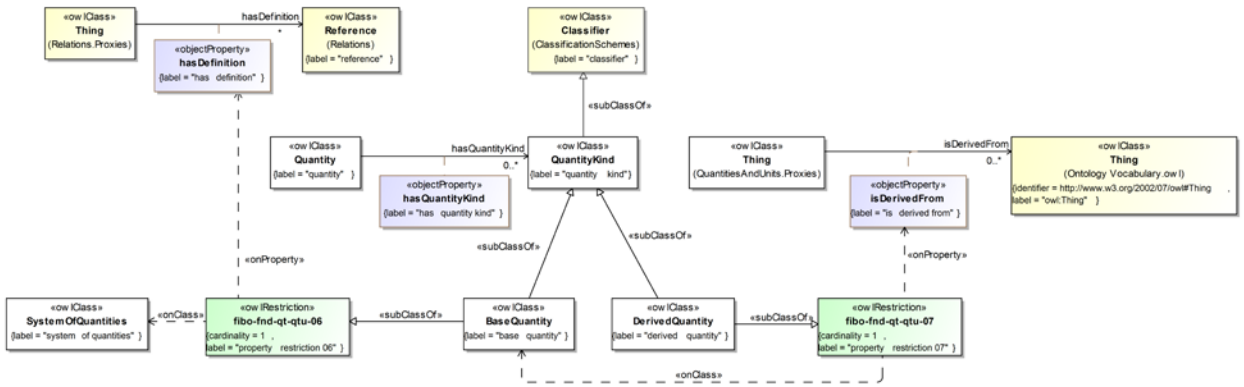

Figure 10.90 Definitions for Quantity Kinds

Figure 10.91 provides the set of definitions relevant to SystemsOfQuantities and SystemsOfUnits.

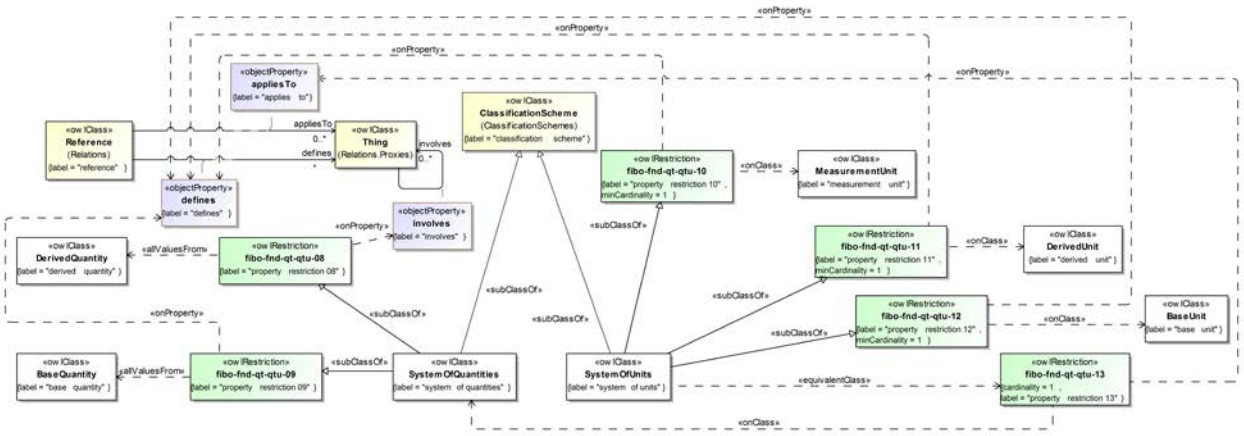

Figure 10.91 Definitions for SystemsOfQuantities and SystemsOfUnits

Finally, Figure 10.92 provides the set of definitions characterizing quantity values.

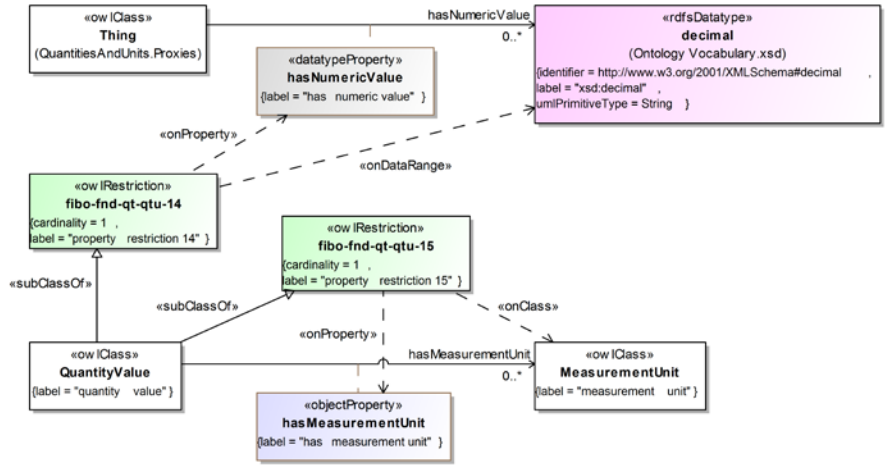

Figure 10.92 Definitions for Quantity Values

Detailed content of the QuantitiesAndUnits ontology is provided in Table 10-89, below.

Table 10-89 Quantities and Units Details

Classes

Name	Annotations	Class Expressions
BaseQuantity (base quantity)	Definition: a quantity kind in a conventionally chosen subset of a given system of quantities, where no subset quantity can be expressed in terms of the others Example: The International System of Quantities (ISQ) comprises these base quantities (with their SI base measurement units): length (meter), mass (kilogram), duration (second), electric current (ampere), thermodynamic temperature (kelvin), amount of substance (mole), and luminous intensity (candela). These base quantities are not mutually comparable. All quantities of any one of these kinds are, however, mutually comparable. Definition origin: http://www.omg.org/spec/DTV/1.1/	Parent Class: QuantityKind Property Restriction: = 1 **hasDefinition**.**SystemOfQuantities** (fibo-fnd-qt-qtu-06)
Base Unit (base unit)	Definition: a measurement unit that is defined by a system of units to be the reference measurement unit for a base quantity Explanatory note: Quantity units that are not base units are derived units. Definition origin: http://www.omg.org/spec/DTV/1.1/	Parent Class: MeasurementUnit Property Restriction: = 1 **hasDefinition**.**SystemOfUnits** (fibo-fnd-qt-qtu-04)
DerivedQuantity (derived quantity)	Definition: a quantity kind, in a system of quantities, that is not a base quantity of the system but may be defined in terms of base quantities of the system Example: velocity (length/time), mass density (mass/length^3) Definition origin: http://www.omg.org/spec/DTV/1.1/	Parent Class: QuantityKind Property Restriction: = 1 **isDerivedFrom**.**BaseQuantity** (fibo-fnd-qt-qtu-07)
DerivedUnit (derived unit)	Definition: a measurement unit for a derived quantity Explanatory note: Every derived unit is defined in terms of base units. Example: 1 minute = 60 seconds Definition origin: http://www.omg.org/spec/DTV/1.1/	Parent Class: MeasurementUnit Property Restriction: = 1 **isDerivedFrom**.**BaseUnit** (fibo-fnd-qt-qtu-05)
MeasurementUnit (measurement unit)	Definition: a quantity, defined and adopted by convention, with which any other quantity of the same kind can be compared to express the ratio of the two quantities as a number Example: week, day, hour, minute, second, kilogram, joule, meter Definition origin: http://www.omg.org/spec/DTV/1.1/	Parent Class: Quantity Property Restriction: ≥ 1 **appliesTo**.**QuantityKind** (fibo-fnd-qt-qtu-03) Property Restriction: ≥ 1 **hasDefinition**.**SystemOfUnits** (fibo-fnd-qt-qtu-02)
ParticularQuantity (particular quantity)	Definition: a property that is of an individual thing and is quantifiable as an instance of some quantity kind Explanatory note: A particular quantity is given by a definite description, which identifies the individual thing and the property. Particular quantities are properties of particular things and are generally expressed by a term for the property and a quantity value. Example: The weight of a given person, the mass of the Earth, the speed of light, and the distance between and are said to be 'particular quantities.' Definition origin: http://www.omg.org/spec/DTV/1.1/	Parent Class: Measure
Quantity (quantity)	Definition: a property of a phenomenon, body, or substance, to which a	Parent Class: Measure

Name	Annotations	Class Expressions
	number can be assigned with respect to a reference Explanatory note: A quantity as defined here is said to be a 'scalar' as distinct from a 'vector'. However, a vector or a tensor whose components are quantities is also considered to be a quantity. Example: second, kilogram, joule, meter. These are quantities in a general sense, which is what is meant here by quantity. Definition origin: http://www.omg.org/spec/DTV/1.1/	Property Restriction: ∃ **hasQuantityKind**.**QuantityKind** (fibo-fnd-qt-qtu-01)
QuantityKind (quantity kind)	Definition: a categorization type for "quantity" that characterizes quantities as being mutually comparable Explanatory note: Every instance of "quantity kind" is also a specialization of "quantity". So the concept "duration" is an instance of "quantity kind" and it is a specialization of "quantity", *i.e.*, it is a classifier of actual quantities. But a given duration (*i.e.*, the duration of something) is an instance of "duration" and thus a "particular quantity" not an instance of "quantity kind". For example, a "year" is not an instance of quantity kind; it is an instance of quantity, but not a category of quantity. Definition origin: http://www.omg.org/spec/DTV/1.1/	Parent Class: Classifier
QuantityValue (quantity value)	Definition: number and measurement unit together giving magnitude of a quantity Explanatory note: The quantity expressed by a quantity value is the quantity whose ratio to the measurement unit is the number. Example: 2 days, 3.5 hours, 150 lb, 45.5 miles Definition origin: http://www.omg.org/spec/DTV/1.1/	Parent Class: Measure Property Restriction: = 1 **hasMeasurementUnit**.**MeasurementUnit** (fibo-fnd-qt-qtu-15) Property Restriction: = 1 **hasNumericValue**.**decimal** (fibo-fnd-qt-qtu-14)
Rate (rate)	Definition: a quantity measured with respect to some other quantity Adapted from: http://www.thefreedictionary.com/rate Adapted from: http://www.icoachmath.com/math_dictionary/rate.html	Parent Class: Quantity
SystemOfQuantities (system of quantities)	Definition: set of quantities together with a set of non-contradictory equations relating those quantities Definition origin: http://www.omg.org/spec/DTV/1.1/	Parent Class: ClassificationScheme Property Restriction: ∀ **defines**.**BaseQuantity** (fibo-fnd-qt-qtu-09) Property Restriction: ∀ **involves**.**DerivedQuantity** (fibo-fnd-qt-qtu-08)
SystemOfUnits (system of units)	Definition: a set of measurement units associated with a system of quantities, together with a set of rules that assign one measurement unit to be the base unit for each base quantity in the system of quantities and a set of rules for the derivation of other units from the base units Example: The International System of Units (SI) is a system of units. Definition origin: http://www.omg.org/spec/DTV/1.1/	Parent Class: ClassificationScheme Property Restriction: = 1 **appliesTo**.**SystemOfQuantities** (fibo-fnd-qt-qtu-13) Property Restriction: ≥ 1 **defines**.**MeasurementUnit** (fibo-fnd-qt-qtu-10)

Name	Annotations	Class Expressions
		Property Restriction: ≥ 1 **defines**.**BaseUnit** (fibo-fnd-qt-qtu-12) Property Restriction: ≥ 1 **defines**.**DerivedUnit** (fibo-fnd-qt-qtu-11)

Properties

Name	Annotations	Property Axioms
hasMeasurementUnit (has measurement unit)	Definition: a quantity value is expressed in terms of a particular measurement unit Definition origin: http://www.omg.org/spec/DTV/1.1/	Parent Property: has Domain: QuantityValue Range: MeasurementUnit
hasQuantityKind (has quantity kind)	Definition: a quantity is an instance of the category of quantity that is the quantity kind Example: hour (the duration) is an instance of “duration” - a specific quantity of time. So the quantity kind of “hour” is “duration”. Definition origin: http://www.omg.org/spec/DTV/1.1/	Parent Property: has Domain: Quantity Range: QuantityKind
isDerivedFrom (is derived from)	Definition: a derived quantity is derived from a base quantity; a derived unit is derived from a base unit Definition origin: http://www.omg.org/spec/DTV/1.1/	
hasNumericValue (has numeric value)	Definition: a number associated with a quantity value Definition origin: http://www.omg.org/spec/DTV/1.1/	Range: decimal

10.15 Module: Products and Services

Metadata defining the primary metadata elements for the ProductsAndServices module are given in Table 10-90, below.

Table 10-90 ProductsAndServices Module Metadata

Metadata Term	Value
sm:moduleName	Products and Services
sm:moduleAbbreviation	FIBO-FND-PAS
sm:moduleVersion	1.0
sm:moduleAbstract	This module includes ontologies defining concepts such as buyers, sellers, customers, clients, products and services generally, as well as very high-level relationships between them, for use in other FIBO ontologies.

10.15.1 Ontology: Products and Services

This ontology defines fundamental concepts for buyers, sellers, clients, customers, products, goods and services for use in other FIBO ontologies.

Metadata defining the primary metadata elements for the ProductsAndServces ontology are given in Table 10-91, below.

Table 10-91 ProductsAndServices Ontology Metadata

Metadata Term	Value
sm:filename	ProductsAndServices.rdf
sm:fileAbbreviation	fibo-fnd-pas-pas
OntologyIRI	http://www.omg.org/spec/EDMC-FIBO/FND/ProductsAndServices/ProductsAndServices/
owl:versionIRI	http://www.omg.org/spec/EDMC-FIBO/FND/20160201/ProductsAndServices/ProductsAndServices/
sm:dependsOn	http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/AnnotationVocabulary/ http://www.omg.org/spec/EDMC-FIBO/FND/AgentsAndPeople/Agents/ http://www.omg.org/spec/EDMC-FIBO/FND/Relations/Relations/ http://www.omg.org/spec/EDMC-FIBO/FND/Arrangements/Documents/ http://www.omg.org/spec/EDMC-FIBO/FND/Arrangements/IdentifiersAndIndices/ http://www.omg.org/spec/EDMC-FIBO/FND/DatesAndTimes/Occurrences/ http://www.omg.org/spec/EDMC-FIBO/FND/Parties/Roles/ http://www.omg.org/spec/EDMC-FIBO/FND/Parties/Parties/ http://www.omg.org/spec/EDMC-FIBO/FND/Places/Locations/ http://www.omg.org/spec/EDMC-FIBO/FND/Places/Facilities/ http://www.omg.org/spec/EDMC-FIBO/FND/Accounting/CurrencyAmount/ http://www.omg.org/spec/EDMC-FIBO/FND/Agreements/Contracts/

Figure 10.93 provides an overview of the concepts in the ProductsAndServices ontology. Subsequent diagrams provide more detailed views on several of the concepts, with a focus on the logical restrictions that further define them.

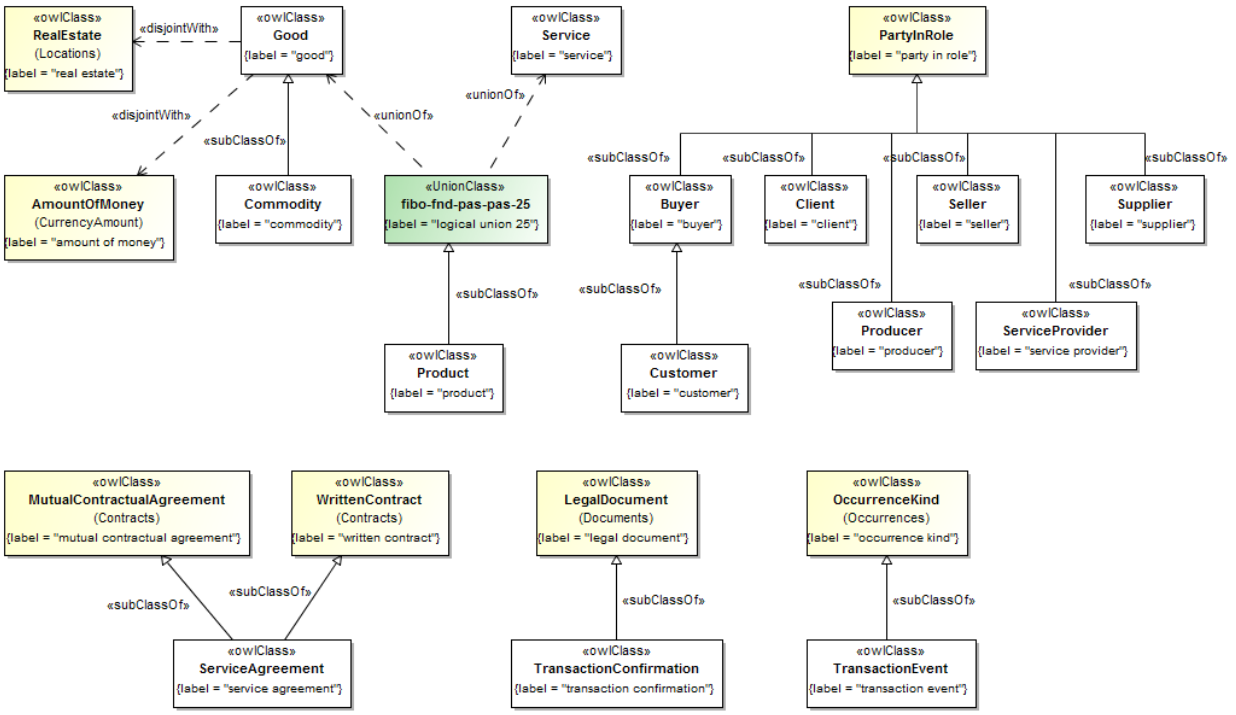

Figure 10.93 Products and Services Class Hierarchy

Figure 10.94 refines the definition of a product in terms of producers and suppliers.

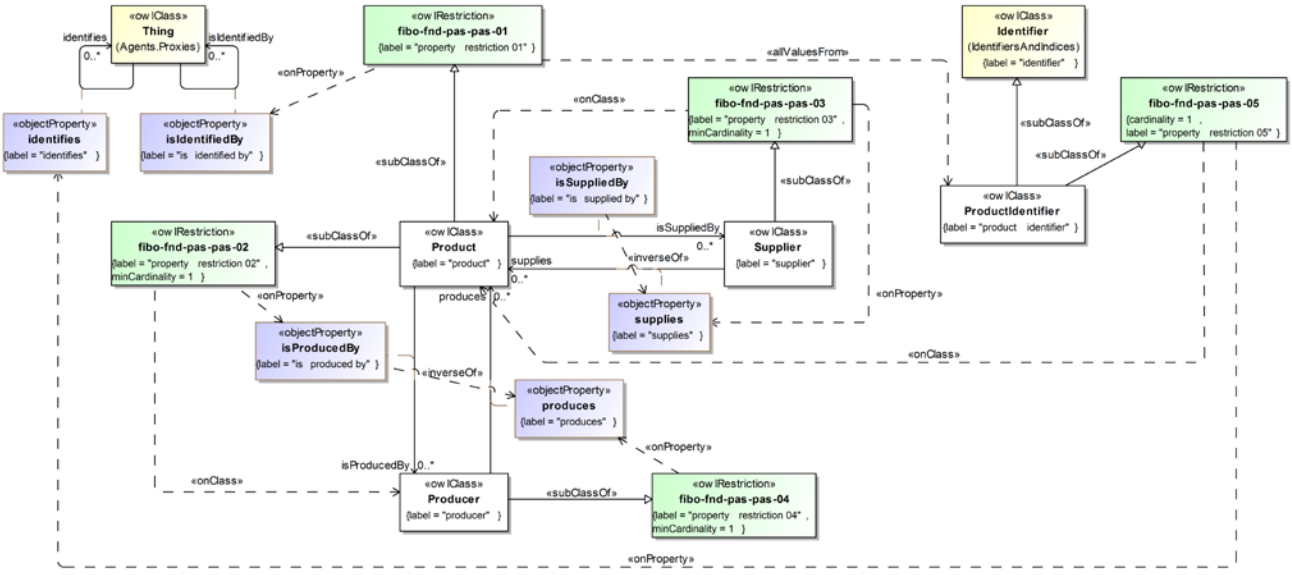

Figure 10.94 Definition of Product – Producers and Suppliers

Figure 10.95 refines the definition of a product in terms of buyers and sellers.

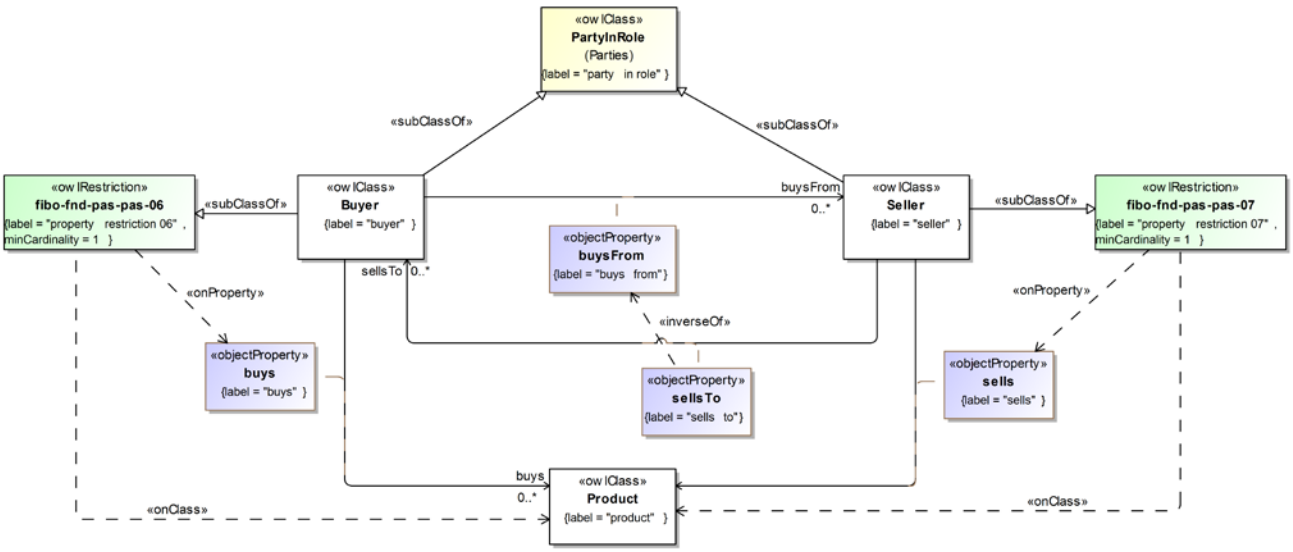

Figure 10.95 Definitions of Buyer and Seller

Figure 10.96 refines the definition of a service provider.

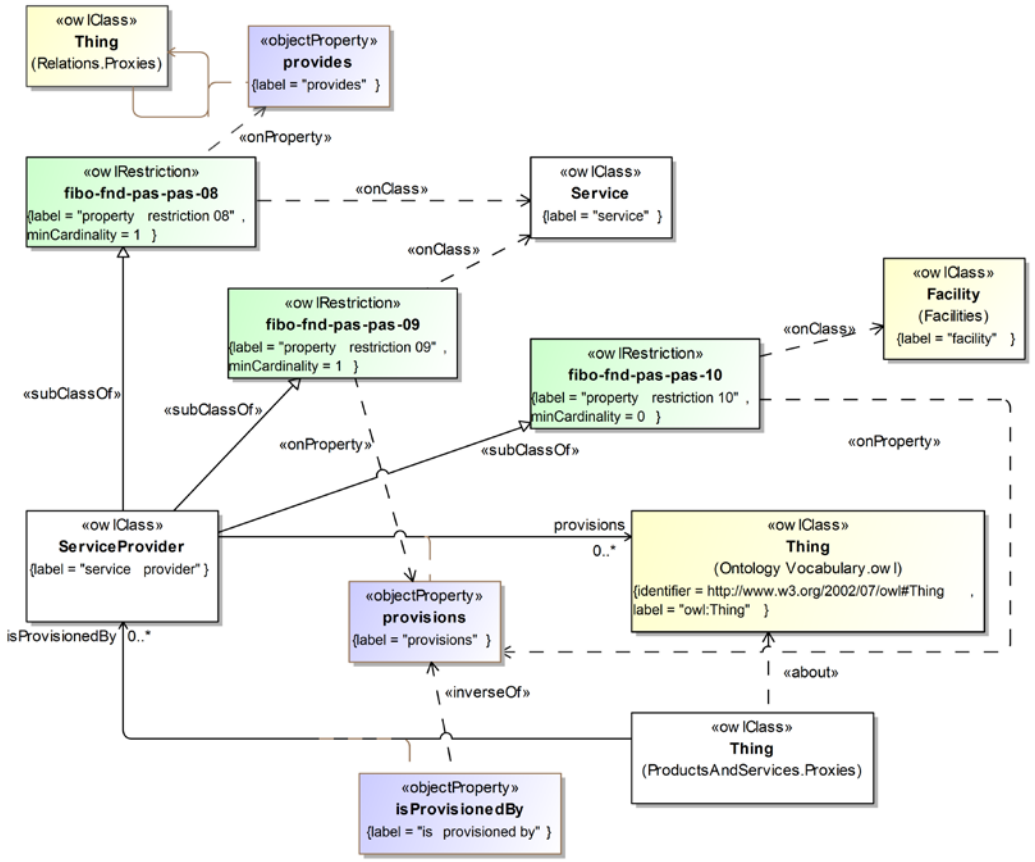

Figure 10.96 Definition of Service Provider

Figure 10.97 depicts the definition of a service agreement.

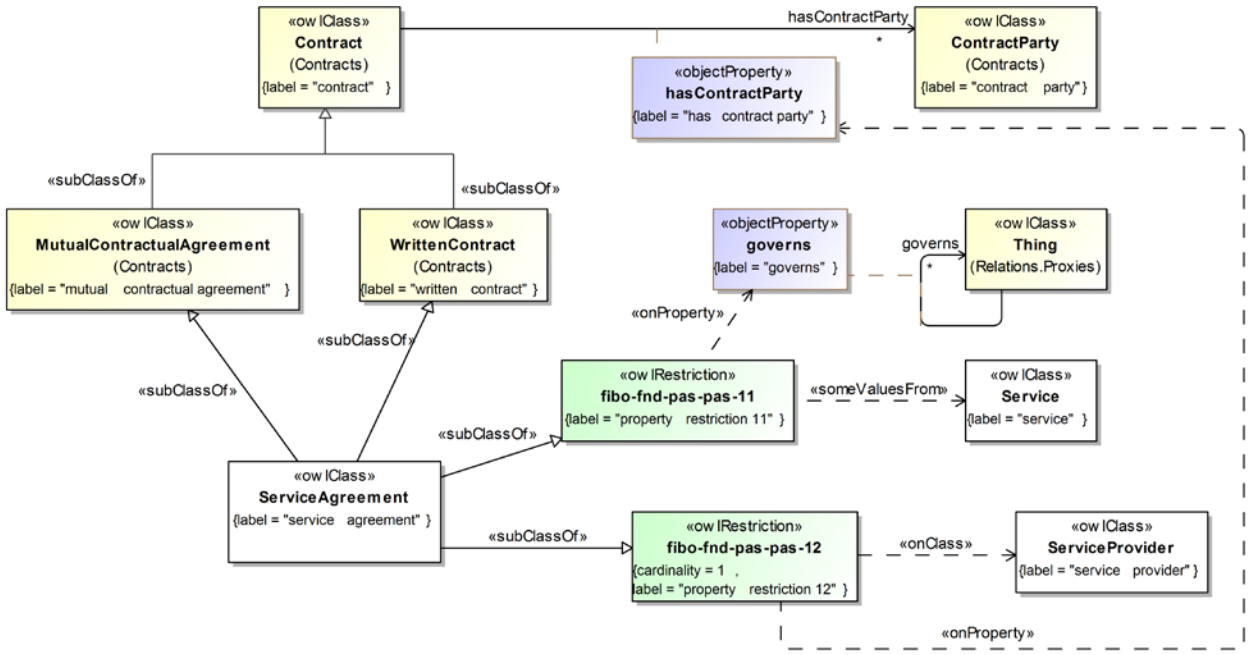

Figure 10.97 Definition of Service Agreement

Figure 10.98 shows the definition of a customer.

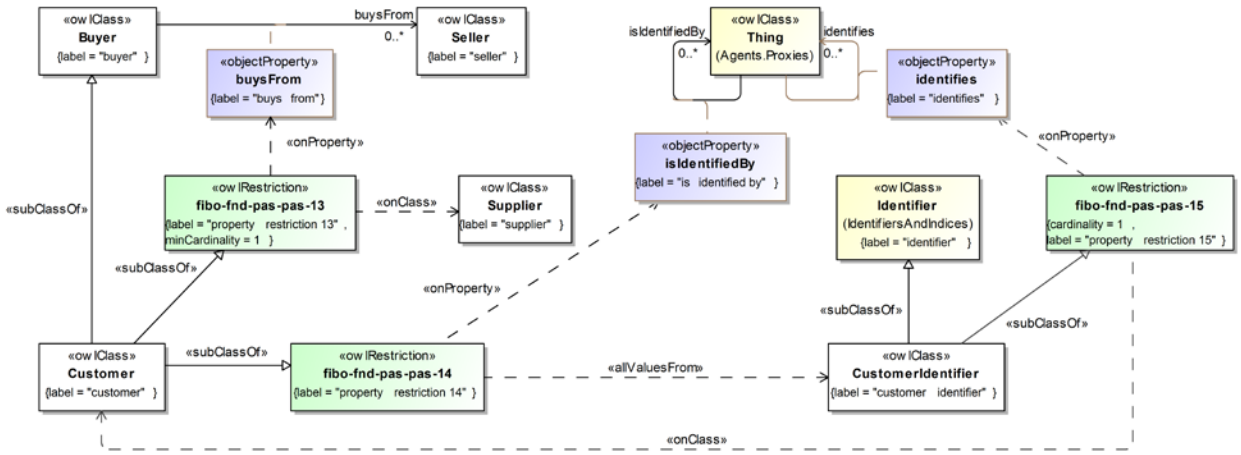

Figure 10.98 Definition of Customer

Figure 10.99 shows the definition of a client. Note that some parties may be both customers and clients. In the context of many banking products, such as various kinds of accounts, for example, account holders are both clients and customers. The identifiers assigned to clients (and/or customers) may or may not be the same as an account number in such a context. These identifiers are essential to addressing use cases including, but not limited to, "Know Your Customer (KYC)" and, in conjunction with legal entity identifiers (defined in the FIBO Business Entities Specification [FIBO BE]), counterparty identification and risk management.

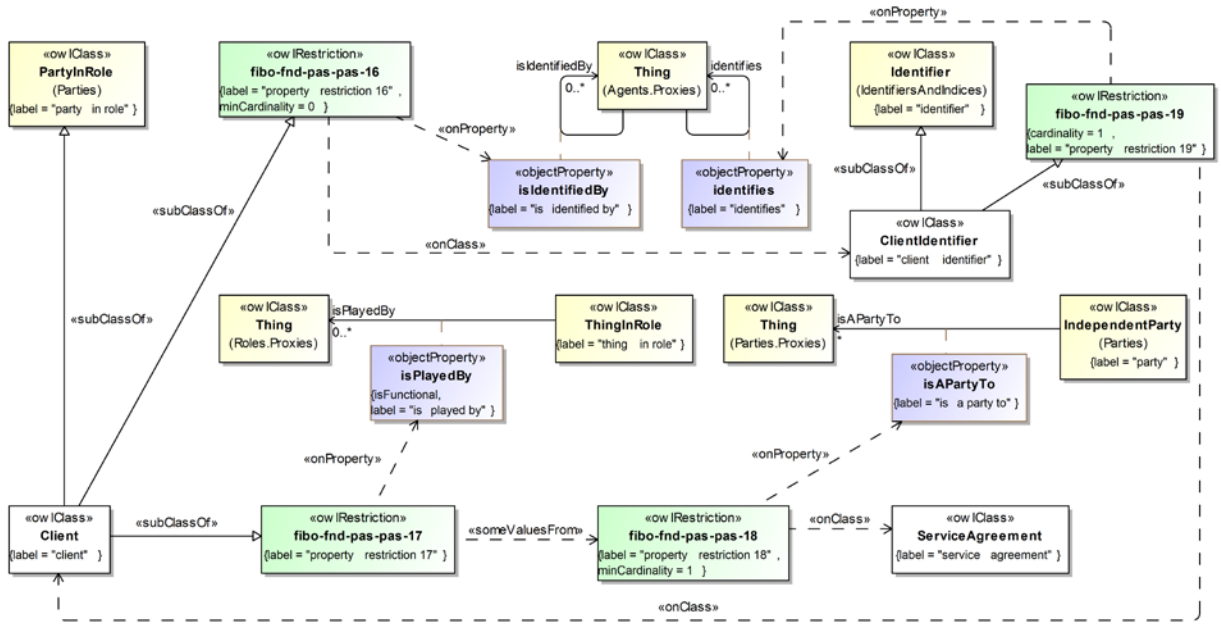

Figure 10.99 Definition of Client

Figure 10.100 provides definitions of transaction events and transaction confirmations.

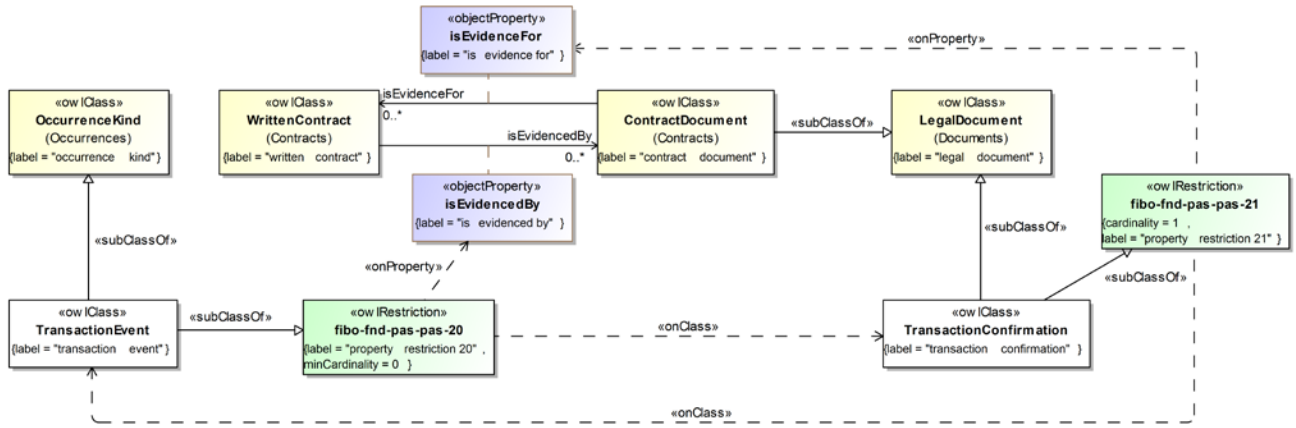

Figure 10.100 Transaction Events and Confirmations

Finally, Figure 10.101 refines the definition of service.

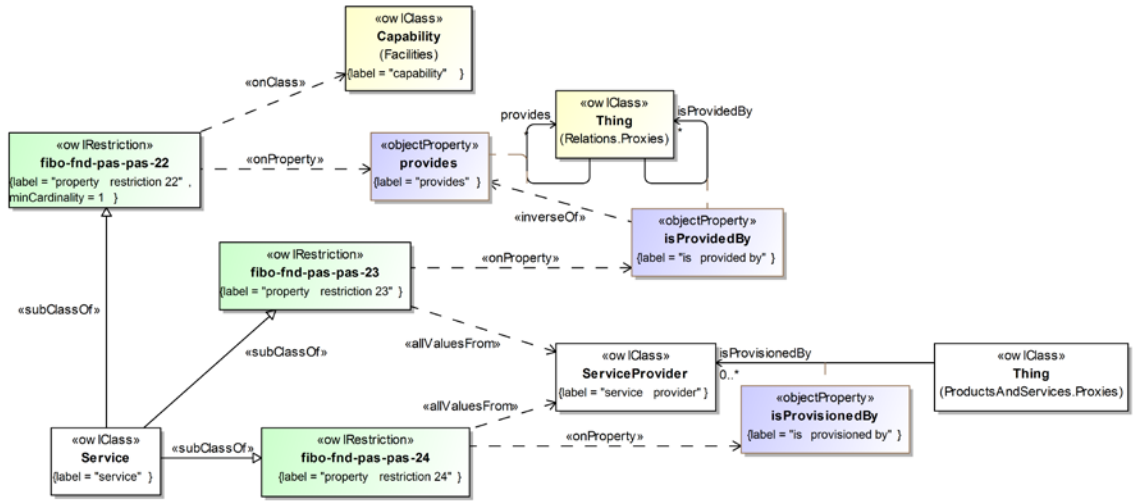

Figure 10.101 Definition of Service

Detailed content for the Products and Services ontology is provided in Table 10-92, below.

Table 10-92 Products and Services Ontology Details

Classes

Name	Annotations	Class Expressions
Buyer (buyer, purchaser)	Definition: a party that acquires, or agrees to acquire, ownership (in case of goods), or benefit or usage (in case of services), in exchange for money or other consideration under a contract of sale Adapted from: http://www.businessdictionary.com/definition/buyer.html	Parent Class: PartyInRole Property Restriction: ≥ 1 **buys**.**Product** (fibo-fnd-pas-pas-06)
Client (client)	Definition: a party that acquires, or agrees to acquire, ownership (in case of goods), or benefit or usage (in case of services), in exchange for money or other consideration under a contract of sale Adapted from: http://www.businessdictionary.com/definition/buyer.html	Parent Class: PartyInRole Property Restriction: ≥ 0 **isIdentifiedBy**.**ClientIdentifier** (fibo-fnd-pas-pas-16) Property Restriction: ∃ **isPlayedBy** (≥ 1 **isAPartyTo**.**ServiceAgreement**) (fibo-fnd-pas-pas-17, fibo-fnd-pas-pas-18)
ClientIdentifier (client identifier)	Definition: an identifier for a client	Parent Class: Identifier Property Restriction: = 1 **identifies**.**Client** (fibo-fnd-pas-pas-19)
Commodity (commodity)	Definition: a basic good used in commerce that is interchangeable with other commodities of the same type Explanatory note: Commodities are most often used as inputs in the production of other goods or services. The quality of a given commodity may differ slightly, but it is essentially uniform across producers. Adapted from: http://www.investopedia.com/terms/c/commodity.asp	Parent Class: Good
Customer (customer)	Definition: a buyer that receives or consumes products (goods or services) and has the ability to choose between different products and suppliers	Parent Class: Buyer Property Restriction: ≥ 1

Name	Annotations	Class Expressions
	Adapted from: http://www.businessdictionary.com/definition/customer.html	**buysFrom**.**Supplier** (fibo-fnd-pas-pas-13) Property Restriction: ≥ 0 **isIdentifiedBy**.**CustomerIdentifier** (fibo-fnd-pas-pas-14)
CustomerIdentifier (customer identifier)	Definition: an identifier for a customer	Parent Class: Identifier Property Restriction: = 1 **identifies**.**Customer** (fibo-fnd-pas-pas-15)
Good (good)	Definition: any tangible thing that is not money or real estate Explanatory note: An inherently useful and relatively scarce tangible item produced from agricultural, construction, manufacturing, or mining activities. According to the UN Convention On Contract For The International Sale Of Goods, the term 'good' does not include (1) items bought for personal use, (2) items bought at an auction or foreclosure sale, (3) aircraft or oceangoing vessels. Adapted from: http://www.businessdictionary.com/definition/good.html	Class Axiom: ¬ AmountOfMoney Class Axiom: ¬ RealEstate
Producer (producer)	Definition: the manufacturer of a product, also called maker Adapted from: http://www.investorwords.com/3872/producer.html	Parent Class: PartyInRole Property Restriction: ≥ 1 **produces**.**Product** (fibo-fnd-pas-pas-04)
Product (product)	Definition: A commercially distributed good that is (1) tangible property, (2) the output or result of a fabrication, manufacturing, or production process, or (3) something that passes through a distribution channel before being consumed or used. Adapted from: http://www.businessdictionary.com/definition/product.html	Parent Class: Good ∩ Service (fibo-fnd-pas-pas-25) Property Restriction: ∀ **isIdentifiedBy**.**ProductIdentifier** (fibo-fnd-pas-pas-01) Property Restriction: ≥ 1 **isProducedBy**.**Producer** (fibo-fnd-pas-pas-02)
ProductIdentifier (product identifier)	Definition: an identifier for a product	Parent Class: Identifier Property Restriction: = 1 **identifies**.**Product** (fibo-fnd-pas-pas-05)
Seller (seller, vendor)	Definition: a party that makes, offers or contracts to make a sale to an actual or potential buyer Adapted from: http://www.businessdictionary.com/definition/seller.html	Parent Class: PartyInRole Property Restriction: ≥ 1 **sells**.**Product** (fibo-fnd-pas-pas-07)
Service (service)	Definition: a type of economic activity that is intangible, is not stored and does not result in ownership; a service is consumed at the point of sale Example: Services include intangible products, such as accounting, banking, cleaning, consultancy, education, insurance, expertise, medical treatment, or transportation services. Adapted from: http://www.investorwords.com/6664/service.html Adapted from: http://www.businessdictionary.com/definition/services.html	Property Restriction: ≥ 1 **provides**.**Capability** (fibo-fnd-pas-pas-22) Property Restriction: ∀ **isProvidedBy**.**ServiceProvider** (fibo-fnd-pas-pas-23) Property Restriction: ∀ **isProvisionedBy**.**ServiceProvider** (fibo-fnd-pas-pas-24)
ServiceAgreement (service agreement,	Definition: a written contract between a client and service provider whereby the service provider supplies some service in the form of time,	Parent Class: MutualContractualAgreement

Name	Annotations	Class Expressions
service contract)	effort, and/or expertise in exchange for compensation Adapted from: http://www.businessdictionary.com/definition/service-contract.html	Parent Class: WrittenContract Property Restriction: ∀ **governs**.**Service** (fibo-fnd-pas-pas-11) Property Restriction: = 1 **hasContractParty**.**ServiceProvider** (fibo-fnd-pas-pas-12)
ServiceProvider (service provider)	Definition: a party that provides and typically provisions professional services, such as consulting, financial, legal, real estate, education, communications, storage, or processing services, to other parties, typically defined in a service agreement Adapted from: http://en.wikipedia.org/wiki/Service_provider	Parent Class: PartyInRole Property Restriction: ≥ 1 **provides**.**Service** (fibo-fnd-pas-pas-08) Property Restriction: ≥ 1 **provisions**.**Service** (fibo-fnd-pas-pas-09) Property Restriction: ≥ 0 **iprovisions**.**Facility** (fibo-fnd-pas-pas-10)
Supplier (supplier)	Definition: a party that supplies goods or services Explanatory note: A supplier may be distinguished from a contractor or subcontractor, who commonly adds specialized input to deliverables. Adapted from: http://www.businessdictionary.com/definition/supplier.html	Parent Class: PartyInRole Property Restriction: ≥ 1 **supplies**.**Product** (fibo-fnd-pas-pas-03)
TransactionConfirmation (transaction confirmation)	Definition: written communication from a seller or service provider reciting the relevant details of a transaction Adapted from: Barron's Dictionary of Banking Terms, Sixth Edition, 2012	Parent Class: LegalDocument Property Restriction: = 1 **isEvidenceFor**.**TransactionEvent** (fibo-fnd-pas-pas-21)
TransactionEvent (transaction event)	Definition: any sale, assignment, lease, license, loan, advance, contribution, or other transfer of any interest in or right to use any property (tangible or intangible) or money, however that transaction is effected, and regardless of whether the terms of the transaction are formally documented Adapted from: Barron's Dictionary of Banking Terms, Sixth Edition, 2012	Parent Class: OccurrenceKind Property Restriction: ≥ 0 **isEvidencedBy**.**TransactionConfirmation** (fibo-fnd-pas-pas-21)

Properties

Name	Annotations	Property Axioms
buys (buys)	Definition:	Domain: Buyer Range: Product
buysFrom (buys from)	Definition:	Domain: Buyer Range: Seller
isProducedBy (is produced by)	Definition: identifies the producer that fabricates, manufactures or creates the product through some production process	Parent Property: isProvidedBy Domain: Product Range: Producer

Name	Annotations	Property Axioms
		Inverse: produces
isProvisionedBy (is provisioned by)	Definition: identifies the service provider that provisions the service or facility	Parent Property: isProvidedBy Range: ServiceProvider Inverse: provisions
isSuppliedBy (is supplied by)	Definition:	Parent Property: provides Domain: Product Range: Supplier Inverse: supplies
produces (produces)	Definition: identifies the product that is fabricated, manufactured or created via some production process	Parent Property: provides Domain: Producer Range: Product
provisions (provisions)	Definition: customizes, provides, or outfits something required for use in delivering a service	Parent Property: provides Domain: ServiceProvider
sells (sells)	Definition:	Domain: Seller Range: Product
sellsTo (sells to)	Definition:	Domain: Seller Range: Buyer
supplies (supplies)	Definition:	Parent Property: provides Domain: Supplier Range: Product

10.15.2 Ontology: Payments and Schedules

This ontology defines basic concepts such as payment, payee, payer, and payment schedule, extending the scheduling concepts from the Dates and Times module, among others.

Metadata defining the primary metadata elements for the PaymentsAndSchedules ontology are given in Table 10-93, below.

Table 10-93 PaymentsAndSchedules Ontology Metadata

Metadata Term	Value
sm:filename	PaymentsAndSchedules.rdf
sm:fileAbbreviation	fibo-fnd-pas-psch
OntologyIRI	http://www.omg.org/spec/EDMC-FIBO/FND/ProductsAndServices/PaymentsAndSchedules/
owl:versionIRI	http://www.omg.org/spec/EDMC-FIBO/FND/20160201/ProductsAndServices/PaymentsAndSchedules/
sm:dependsOn	http://www.omg.org/spec/EDMC-FIBO/FND/Utilities/AnnotationVocabulary/ http://www.omg.org/spec/EDMC-FIBO/FND/Relations/Relations/ http://www.omg.org/spec/EDMC-

Metadata Term	Value
	FIBO/FND/DatesAndTimes/FinancialDates/ http://www.omg.org/spec/EDMC-FIBO/FND/DatesAndTimes/Occurrences/ http://www.omg.org/spec/EDMC-FIBO/FND/Parties/Roles/ http://www.omg.org/spec/EDMC-FIBO/FND/Parties/Parties/ http://www.omg.org/spec/EDMC-FIBO/FND/Accounting/CurrencyAmount/ http://www.omg.org/spec/EDMC-FIBO/FND/Agreements/Agreements/ http://www.omg.org/spec/EDMC-FIBO/FND/Agreements/Contracts/ http://www.omg.org/spec/EDMC-FIBO/FND/Law/LegalCapacity/ http://www.omg.org/spec/EDMC-FIBO/FND/ProductsAndServices/ProductsAndServices/

Figure 10.102 provides an overview of the concepts in the PaymentsAndSchedules ontology. Subsequent diagrams provide more detailed views on several of the concepts, with a focus on the logical restrictions that further define them.

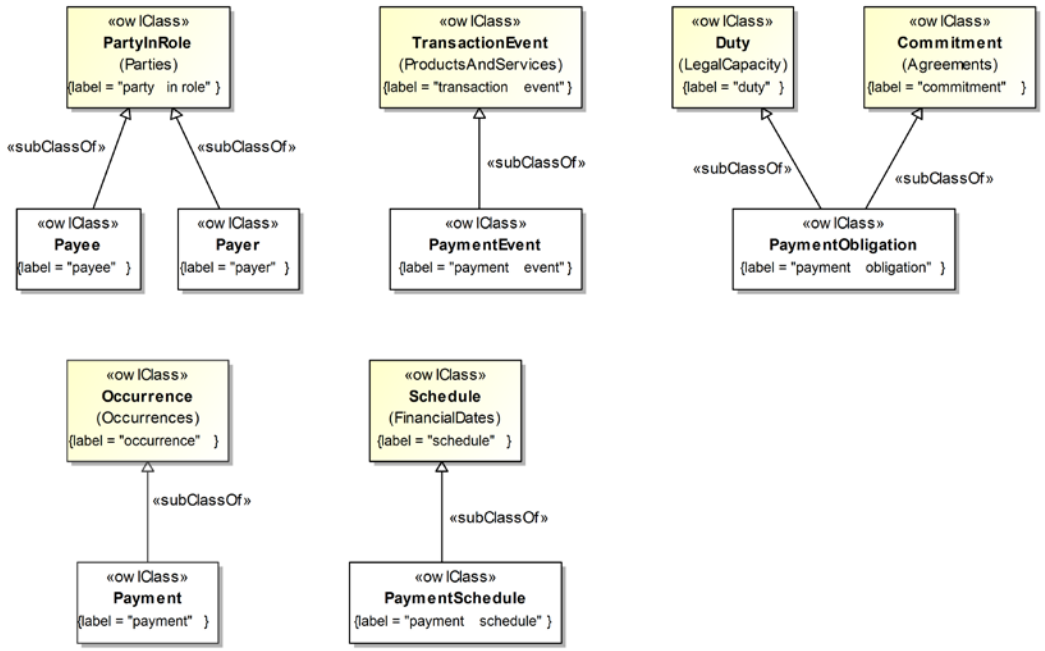

Figure 10.102 Payments and Schedules Class Hierarchy

Figure 10.103 shows the definition of a Payment.

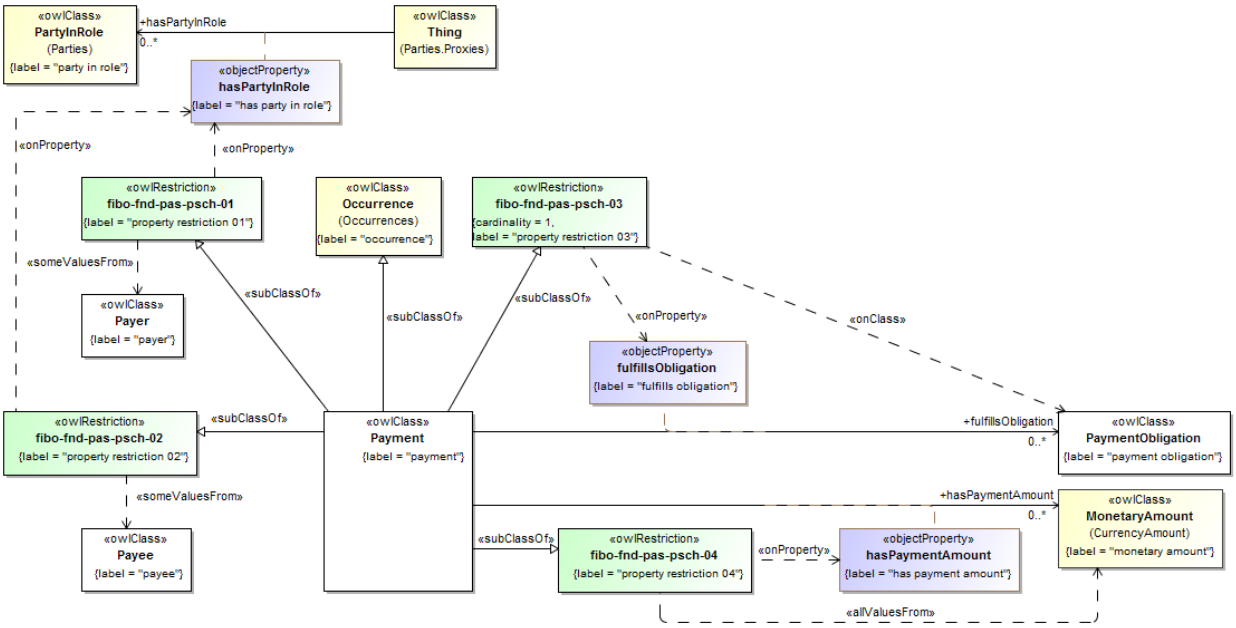

Figure 10.103 Definition of Payment

Figure 10.104 provides the definition of a Payee.

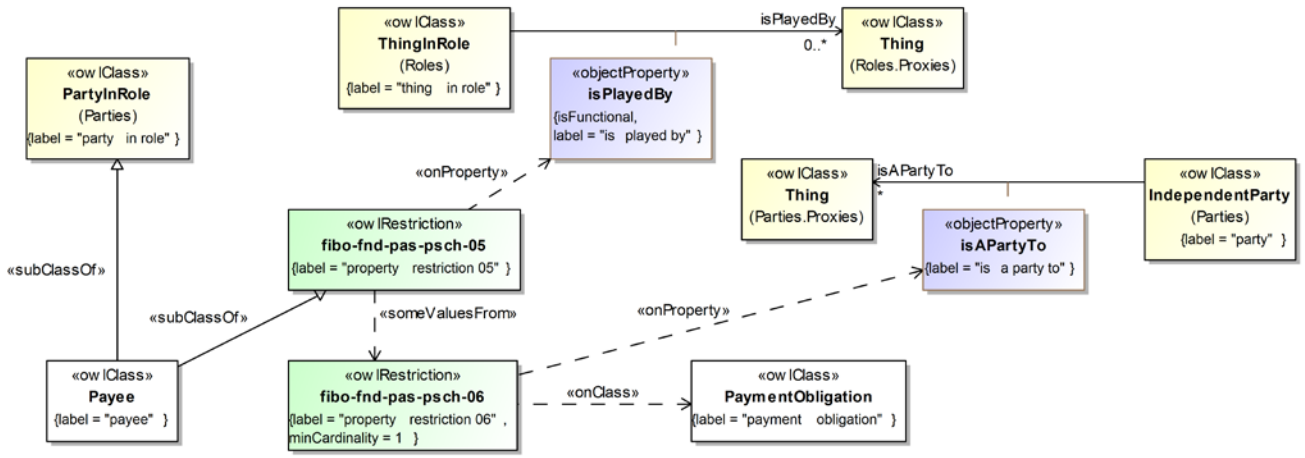

Figure 10.104 Definition of Payee

Figure 10.105 provides the definition of a Payer.

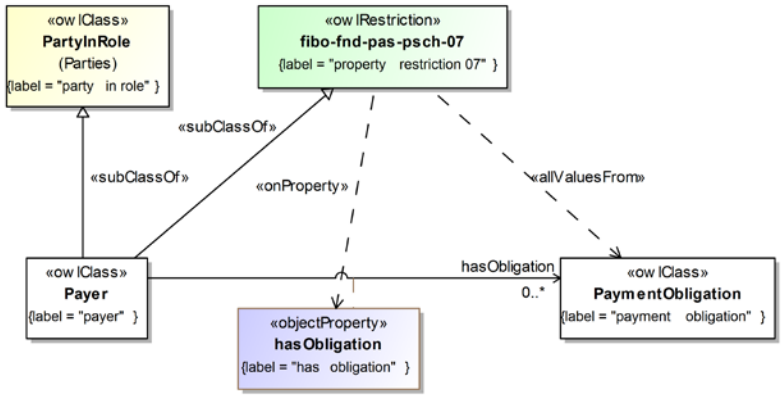

Figure 10.105 Definition of Payer

Figure 10.106 depicts the definition of a PaymentEvent.

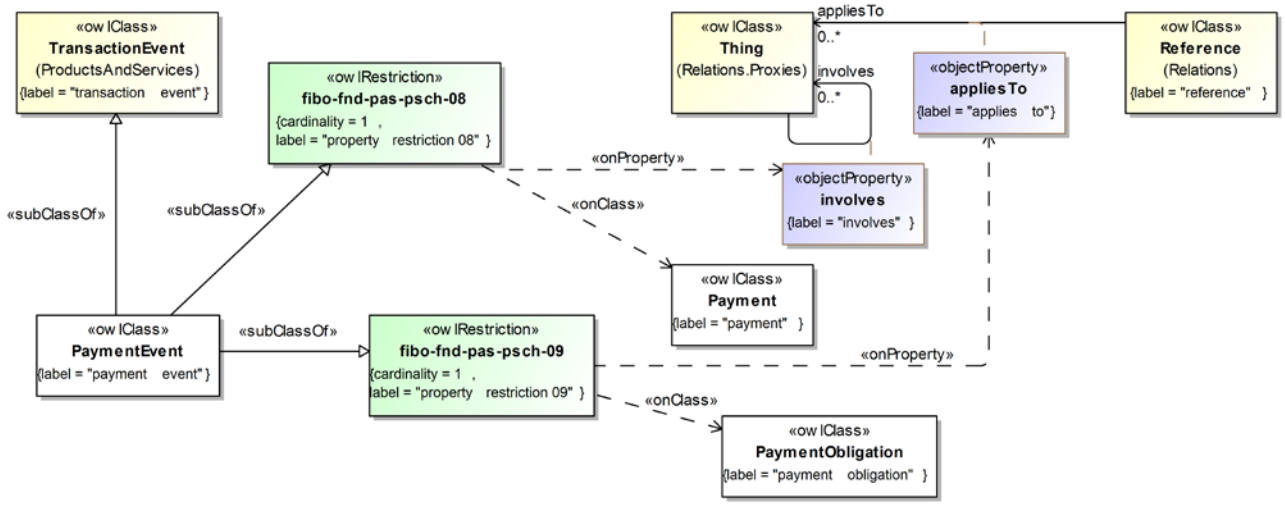

Figure 10.106 Definition of Payment Event

Figure 10.107 depicts the definition of a PaymentObligation.

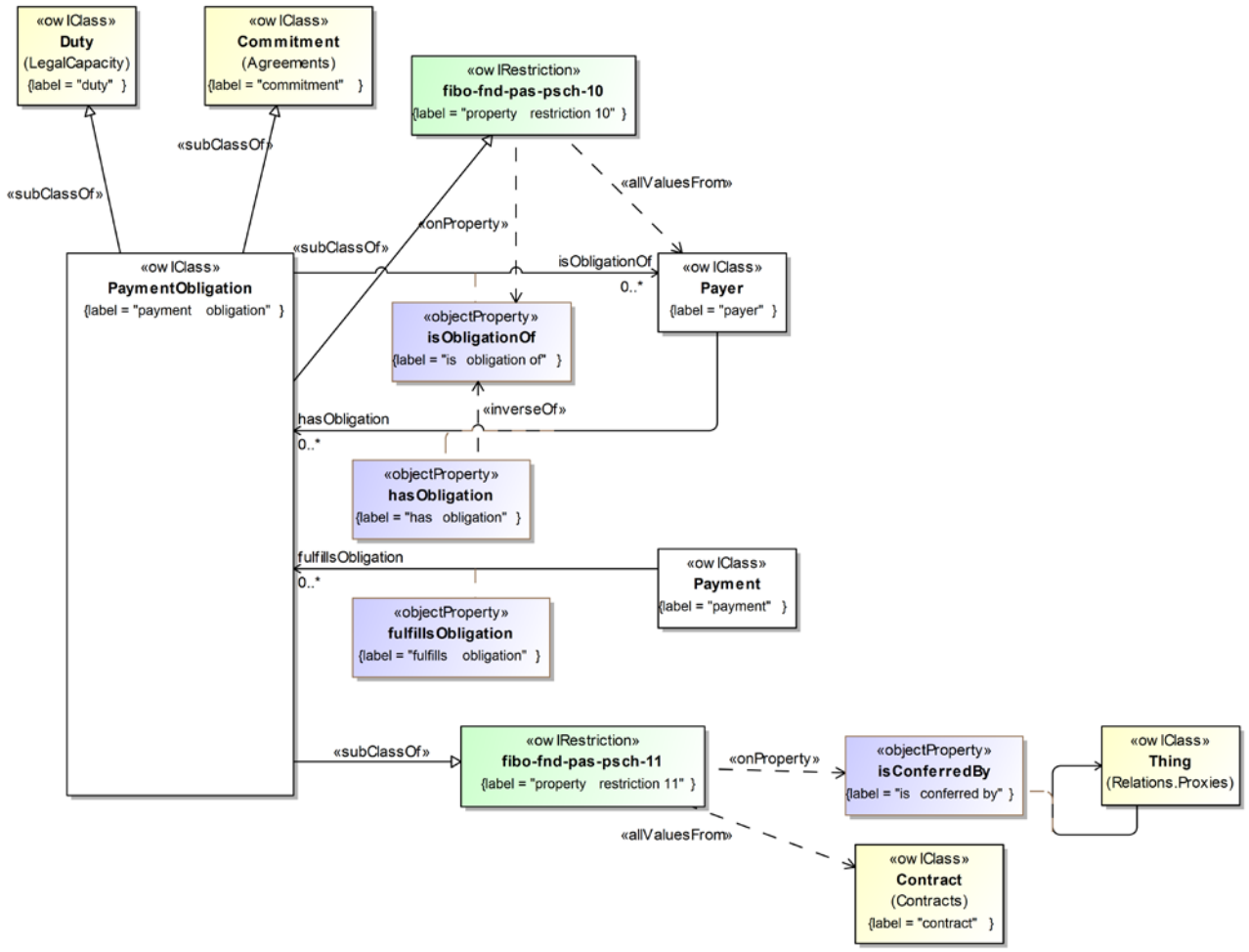

Figure 10.107 Definition of Payment Obligation

Finally, Figure 10.108 depicts the definition of a PaymentSchedule.

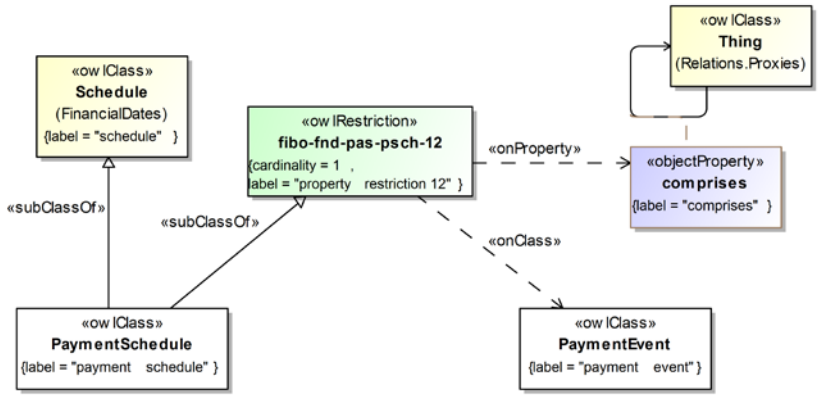

Figure 10.108 Definition of Payment Schedule

Detailed content of the Payments and Schedules ontology is provided in Table 10-94, below.

Table 10-94 Payments and Schedules Ontology Details

Classes

Name	Annotations	Class Expressions
Payee (payee)	Definition: a party to whom a debt should be paid, or to whose order a bill of exchange, note, or check is made payable Adapted from: Barron's Dictionary of Business and Economics Terms, Fifth Edition, 2012	Parent Class: PartyInRole Property Restriction: ∃ **isPlayedBy** (≥ 1 **isAPartyTo**.**PaymentObligation**) (fibo-fnd-pas-psch-05, fibo-fnd-pas-psch-06)
Payer (payer)	Definition: a party who pays a bill or fees Adapted from: Barron's Dictionary of Business and Economics Terms, Fifth Edition, 2012	Parent Class: PartyInRole Property Restriction: ∀ **hasObligation**.**PaymentObligation** (fibo-fnd-pas-psch-07)
Payment (payment)	Definition: delivery of money in fulfillment of an obligation, such as to satisfy a claim or debt Adapted from: Barron's Dictionary of Business and Economics Terms, Fifth Edition, 2012	Parent Class: Occurrence Property Restriction: = 1 **fulfillsObligation**.**PaymentObligation** (fibo-fnd-pas-psch-03) Property Restriction: ∃ **hasPartyInRole**.**Payee** (fibo-fnd-pas-psch-02) Property Restriction: ∃ **hasPartyInRole**.**Payer** (fibo-fnd-pas-psch-01) Property Restriction: ∀ **hasPaymentAmount**. **MonetaryAmount** (fibo-fnd-pas-psch-04)
PaymentEvent (payment event)	Definition: an event that involves delivery of money in fulfillment of an obligation Adapted from: Barron's Dictionary of Business and Economics Terms, Fifth Edition, 2012	Parent Class: TransactionEvent Property Restriction: = 1 **appliesTo**.**PaymentObligation** (fibo-fnd-pas-psch-09) Property Restriction: = 1 **involves**.**Payment** (fibo-fnd-pas-psch-08)
PaymentObligation (payment obligation)	Definition: a legally enforceable duty to pay a sum of money, or agree to do something (or not to do something), according to the terms stated in a contract Example: the duty of a borrower to repay a loan, and the legal right of a lender to enforce payment Adapted from: Barron's Dictionary of Business and Economics Terms, Fifth Edition, 2012	Parent Class: Commitment Parent Class: Duty Property Restriction: ∀ **isConferredBy**.**Contract** (fibo-fnd-pas-psch-11) Property Restriction: ∀ **isObligationOf**.**Payer** (fibo-fnd-pas-psch-10)
PaymentSchedule (payment schedule)	Definition: schedule for delivery of money in fulfillment of an obligation, such as a coupon payment schedule, loan payment schedule, interest payment schedule Adapted from: Barron's Dictionary of Business and Economics Terms, Fifth Edition, 2012	Parent Class: Schedule Property Restriction: = 1 **comprises**.**PaymentEvent** (fibo-fnd-pas-psch-12)

Properties

Name	Annotations	Property Axioms
fulfillsObligation (fulfills obligation)	Definition: satisfies a requirement for payment of some claim, debt, or other obligation	Domain: Payment Range: PaymentObligation
hasObligation (has obligation)	Definition: identifies a duty or obligation that a given party has taken on	Parent Property: has Domain: Payer Range: PaymentObligation Inverse: isObligationOf
hasPaymentAmount (has payment amount)	Definition: specifies the amount of money involved in a payment	Parent Property: has Domain: Payment Range: MonetaryAmount
isObligationOf (is obligation of)	Definition: identifies a party that has incurred a given obligation	Parent Property: isConferredOn Domain: PaymentObligation Range: Payer

Annex A: Machine Readable Files Part of This Specification

(normative)

The FIBO ontologies are delivered as (1) RDF/XML serialized OWL (normative and definitive), (2) UML XMI, serialized from UML with the ODM profiles for RDF and OWL applied (normative), (3) ODM XMI, serialized based on the ODM MOF metamodels for RDF and OWL (normative), and (4) Visual Ontology Modeler (VOM) model files, based on the VOM plug-in to MagicDraw (informative). If there are differences between the OWL files, ODM XMI, and UML XMI, the OWL files take precedence, followed by the UML XMI, and finally the ODM XMI.

Regardless of their form, each of the ontologies included in Foundations makes normative reference to the DCMI Dublin Core Metadata Terms[4], W3C Simple Knowledge Organization System (SKOS) Recommendation[5], and the OMG Architecture Board's Specification Metadata Recommendation[6], which are not part of this specification.

The individual RDF/XML files are organized by module (directory), and within a given module, alphabetically by name, as shown in the URI structure for each individual OWL file. These files are UTF-8 conformant XML Schema files that are also OWL 2 compliant, and may be examined using any text editor, XML editor, or RDF or OWL editor. They have been verified for syntactic correctness via the W3C RDF Validator and University of Manchester OWL 2 Validator. They have also been checked for logical consistency using the Pellet OWL 2 reasoner from Clark & Parsia as well as the HermiT OWL 2 reasoner from Oxford University. It is anticipated that the OWL ontologies will be dereference-able, together with technical documentation (HTML) from the OMG site once the specification is adopted.

Note that the ontologies use features of the OWL 2 language and other ODM revisions that will not be available in the Ontology Definition Metamodel (ODM) until the ODM 1.1 specification is published. The ODM RTF has published a convenience document, available to OMG members, that incorporates specification changes required for FIBO that have already been resolved by the working group, and which we anticipate will be available later this year once the report and related specification is published.

[4] http://www.dublincore.org/documents/dcmi-terms/

[5] http://www.w3.org/TR/2009/REC-skos-reference-20090818/

[6] http://www.omg.org/techprocess/ab/SpecificationMetadata/

Annex B: Shared Semantics Treatments

(normative)

B.1 Introduction

Intended Audiences: *Semantic Modelers; Technical architects*

The model content is grounded in terms which come from outside the realm of business entities of financial services. These are maintained in the Foundations ontology. Wherever possible, terms in this annex are cross referenced to terms set out by suitable standards bodies and academic bodies, so that the meanings of these terms are grounded in a broader community of semantics modeling.

Some of these external standards are in the form of formal ontologies, modeled typically but not necessarily in the Web Ontology Language (OWL) and in any case grounded in formal first order logic. In addition, some terms are derived from models which are not formally grounded in first order logic but which in some way or another are identified as meaningful concepts, either by explicit mark-up of the model content, by some separate theory of meaning, or by some statement at the level of the model identifying it as a semantic model. Such models are typically in the Unified Modeling Language (UML) or some other formalism such as that of the eXtensible Business Reporting Language (XBRL).

Some of the models are only referred to in part, for example because the scope of the standard, as identified by its business requirement, is very different to the scope of the concepts in the Foundations ontologies, or because the ontology contains formal axioms or facts which are at odds with Foundations.

This annex describes the range of treatments by which such external standards are cross referenced in the Foundations ontologies. A number of such treatments have been identified, depending on the nature of the standard or vocabulary referred to in FIBO Foundations, the language in which it is framed or the extent to which we are confident of making direct formal reference to it. For example, for some ontologies we wish to make direct, explicit reference, whereas for others we may have less visibility or confidence in the maintenance arrangements of that model's content and so have elected to create a local 'snapshot' of that ontology with its own namespace.

B.2 Shared Semantics Treatments

Case 1: Complete, stable OWL Ontologies

Treatment: If an ODM representation does not already exist as part of the standard, create a surrogate of the ontology using ODM.

Because this is in ODM, it shall have the actual URIs of the external standard. The material in FIBO represents a direct use of that ontology with its original namespace.

Case 2: Ontology Snapshot

If the external ontology is in OWL but we want to make a snapshot of it at a point in time

Treatment:

- Create clone copy of the ontology in our repository
- Allocate a URI which identifies this as a clone (to include the elements of the original URI plus "/fiboclone/")
- Use OWL equivalentClass, to point from an element in the FIBO clone to the corresponding element in that ontology.

When to use snapshot

This is used when for any reason we don't want to reference changes to the external ontology.

Case 3: Partial Snapshot

This treatment is for when the external ontology has a broader or different business requirement and range of concepts, such that we may not wish to refer to or replicate them all.

Treatment: Create a clone of only those the parts of the ontology we wish to refer to.

Otherwise the treatment is the same as for Case 2, except that in place of the URI fragment “/fiboclone”, the fragment “/fibopartialclone” should be used.

Annex C: Logical versus Conceptual Models comparison

(informative)

Intended Audiences: *Technology Management*

C.1 Comparison Table

The principal differences between a logical data model and a semantic model are shown in Table C1.1.

Table C1.1 Model Comparisons

Logical Data Model	Semantic Model
Represents elements in a database design	Should not include design information but is a model of business concepts
Represents data model design components (Classes in OO design; tables in relational database design)	Represents "Things" using set theory concepts
Combines common data structures for reuse and efficiency	No efficiency considerations because it is not a design; reiterates concepts as they apply
Single inheritance hierarchy	Multiple inheritance
May define a number of optional properties of a class, such that the application developer would know whether these apply or not	Defines what facts are applicable to a given type of thing.
Uses enumerations to quality classes	Enumerates classes ("Things")
Closed World Assumption (CWA)	Open World Assumption (OWA)

These are explained further in the sub clauses which follow.

C.2 Detailed Models Comparison

Design Elements versus Business Concepts

A logical data model represents the design of some data structure such as a database or a message design. This differs from a physical data model in that it is not specific to any one implementation or platform. That is, a logical data model is a kind of "Platform Independent Model" or PIM, as distinct from a "Platform Specific Model" or PSM.

While a logical data model is not specific to any one physical implementation, it does represent some design. That is, the logical data model, like any logical design, represents the results of some design effort by some designer.

A semantic model does not represent any design of any solution, but explicitly represents facts about the problem domain.

If a designer sets out to design something, there should normally be something that they are working from. In the design of software, designers work from formal business requirements statements, such as "Use Case" models or a requirements specification document. For data, the equivalent is a semantic model. That is to say, a designer of a data model should be expected to work from some source of knowledge of the items which are to be catered for in the database or messages for which they are carrying out the design.

Components that are Represented (Classes, Tables or Things)

In order to create a model which represents the logical design of some database or message scheme, the modeler will create a model which represents components of that design. For example, in a relational database they will create a model of database tables, along with relationships between those tables, public and private keys and so on. A logical representation of the design is therefore a representation of database constructs, namely tables, relationships, keys and so forth. The logical data model design is therefore couched in a notation which has formal representations of those elements. This may take the form of an Entity Relationship Model (ERM) or an object oriented model in the form of a Class Model in the UML design notation.

Depending on the model notation chosen by the developer therefore, the model may be an ERM model of data entities and relationships, or a UML class model of classes, associations, composition relationships and so on. These are the items to which elements of the model refer.

By contrast, a semantic model does not represent a logical design, and the things in the semantic model represent instead the real world entities in the business domain itself.

For example, a logical data model for securities may contain a representation of data tables for data about shares, bonds and so on, whereas a semantic model of the securities domain will contain representations of shares and bonds themselves, as kinds of "Thing".

The relationship between a semantic model element and the things it represents is made explicit in the Semantic Web "Web Ontology Language" or OWL notation. In an OWL model, every kind of "Thing" in the model (also known as "Classes") is a set theory construct which defines membership of the set in terms of the properties of its members. All classes in an OWL ontology model are sub-classes of a class known as the "Universal" set, commonly labeled as "Thing". This is the set of which everything is a member. In this way it is made explicit that everything in the model is some thing.

Reuse

It is sensible when carrying out data model design, to identify similar sets of terms and combine these into reusable sets. A semantic model may end up combining common concepts if the concept can be described as a more general, more abstract variant of the kind of thing. However, this is not a requirement for model design - things may be combined according to similarity in the data structures without reference to their meaning.

This is really another aspect of the basic fact that, since a semantic model is not a design, it has no design constraints (note this may not the case for an individual semantic technology application, where constraints are rightly applied but are very different to those for relational database or message design).

Single versus Multiple Inheritance

A limitation of some (though not all) relational design environments and notations is that the classes would be arranged in a hierarchy of classes. These would be in a single inheritance "tree" i.e. each class has only one parent class of which it is a specialization (ignoring polymorphism for now).

Semantic models more closely reflect the real world dispensation of taxonomies of kinds of thing, namely that a set of classes may defined according to more than one property. For example, a whale is both a marine animal and a mammal according to two different kinds of classification hierarchy, and an individual whale, being a member of the class of things which are a whale, is classified as both kinds of thing.

This is particularly valuable in modeling of kinds of security for different applications. For example risk management and securities trading performance analysis have different requirements, based on asset types, cash flow behaviors and so on. One application would need to classify things according to one set of requirements. Regulators have different requirements to traders, and even different regulators or different areas of regulatory analysis and systemic risk analysis may dictate different ways in which the universe of instruments may be "sliced" for analysis.

Optionality

In standards, particularly message standards, it is good practice to have a number of properties that may or may not apply to a given category of data element (for example, for a data element for a debt security), and make all of these optional. This is practical: for any debt instrument, not all the properties necessarily apply, but someone wanting to send a message

from one point to another will be able to populate the message with those properties that exist for that security.

This, by definition, does not represent the knowledge that business practitioners may have about what facts necessarily must apply for a given instrument of a given type. In order to provide a message which is complete and correct, the sending party needs to apply knowledge from outside the model, about what facts necessarily apply to a given instrument. This intelligence would typically need to be built into the application that builds the message which is sent according to that schema. The knowledge is not represented in the schema.

At base this is simply another way of saying that the logical design of the message is not a representation of the knowledge about the instrument. Needless to say, this is not a criticism of such a message, it is simply a statement of why the message schema is not a record of the knowledge about the instruments.

Enumerations

A valid and good design approach to different kinds of thing is to provide a single data element which is an enumeration, containing entries for each of a number of entries that distinguish these things.

In a semantic model, each thing in the enumeration is a separate class of "Thing". The presence of enumerations in a model indicates that this is a logical model.

Note that for simplicity is it sometimes the practice to provide an enumeration (of textual strings, or 'literals') in a semantic model. However this is usually a pointer to the need to develop the semantics of the model further.

Open versus Closed World Assumption

FIBO specifications are expressed in OWL, which uses the Open World Assumption.

- Open World Assumption: Absence of evidence is not evidence of absence
- Closed World Assumption: Absence of evidence is evidence of absence

What this means in practice is that facts can be asserted about a thing in a semantic model without consideration to whether these facts are represented by actual data. For example, a fact about any event is that it has a cause, however causes of events need not be known or represented.

On a more detailed level, a semantic model can describe and represent facts about things without those facts being represented as data. Very often the facts, which define the nature of a thing, may not correspond directly to data. For example, many financial instrument types are defined in terms of the legal rights and obligations that they represent to one or other party to the contract. These rights and obligations may correspond indirectly to data elements, but the legal facts themselves may be more abstract, i.e. a fact stated in terms of "has right to" or "commits to" may refer to the abstract concept of a right, while the data may contain details of those rights and obligations, which may be regarded as a sort of signature revealing the existence of those rights and obligations.

This would be true of anything which is defined and classified according to facts which are themselves abstract. This would include most legal concepts.

C.3 Model Partitioning

The FIBO Foundations concepts are partitioned into several non-mutually exclusive categories, in the sense in which the term “partition” is used in the semantic modeling community. These are:

- Independent, Relative and Mediating things
- Concrete and Abstract things
- Continuant and Occurrent things.

Each partition is represented as a class of OWL Thing and as a sub-type of the OWL Thing class, without additional archetype indications.

Terms defined in the model in this specification, and any terms defined in future additions to this specification or in local ontologies derived by extension of this specification, may not have a direct parent class of 'OWL Thing'. All classes of

thing in the model described in this specification are given a parent which is either an archetype class of Thing or has an archetype as an ancestor, and all archetypes are given a parent from each of the three partitions listed above, with the exception of temporal terms which exist in a separate partition to the above.

Users of parts of this model may optionally ignore the above partitions in order to dispose model content under separate partitions of their own.

C.3.1 Independent, Relative and Mediating Things

This set of partitions provides a division into the model according to categories which have been arrived at through a considerable body of philosophical literature, notably that of C. S. Peirce. This partitioning relies on the claim in that literature that all things which can be named and classified fall into one and only one of these categories. This principle is reflected in the model described in this specification.

An independent thing is something which is defined in its own right and without reference to any context. For example, a business entity is an independent thing.

A relative thing is something the definition and meaning of which is specific to some specific context. That which is defined in that context is itself identified as some independent thing, or in some cases some other kind of relative thing, which stands in the role or relationship defined as the relative thing. For example a party to a contract is a relative thing, being itself some independent thing, in this case some business entity.

A mediating thing is the context in which some thing is defined as being some relative thing. For example, the context of contractual relationships, or of the context in which some specific kind of contract is entered into, is the mediating thing in which the business entity is identified as being some contract party. The term 'Mediating Thing' is synonymous with 'context' in the broadest sense of that term.

Relative things always have a relationship of 'identity' with some thing which may stand in the role identified by the relative thing. This is usually but not always some independent thing. In some cases the identity relationship may refer to some other relative thing, for example a securities issuer may be a 'Special Purpose Vehicle' which itself is defined as a kind of relative entity, the identity of which may be a company incorporated by the issue of shares, a limited liability partnership or some other form of legal entity. For this reason, while relative things should normally have an identity relationship to some independent thing, the most general application of this relationship is to the universal class 'Thing'.

C.3.2 Concrete and Abstract Things

This partition simply identifies whether something is a concrete item with weight and mass, or an abstract construct. Many of the concepts formally identified in the financial services industry are by their nature abstract.

Archetypes may only be identified as concrete or abstract if this is necessarily the case for all things of that archetype.

Note that things which have legal standing and which may be either provided on paper or in a dematerialized form are identified in this model as concrete. The intention of the Abstract partition is to define things which by their very nature are abstractions, such as goals.

One important class of abstract things is those things that are made up of information. According to the modeling principals, only things which are real may be represented in this model. This necessarily excludes things like database keys and locally defined identifiers. A common sense test needs to be applied to any kind of information before it is considered to be real and therefore able to be modeled here. Public information constructs such as security identifiers, business entity identifiers, credit ratings and the like pass this test because they are published by some party. In addition, documents and messages and the like which are passed between entities or parties in the course of carrying out some business process are equally real even though they are not published. The test for their reality is passed because information constructs such as documents have some real business, legal or financial import, that is some impact on something which is itself modeled as being part of the real world and not part of the technical design of some data or application.

C.3.3 Continuant and Occurrent Things

This partition segregates things which by their nature have some existence of a period of time, with a beginning and an end to their existence, and things which by their nature occur at a point in time. The precise timescales on which a thing may be said to occur or to have an ongoing existence is itself dependent on the domain being modeled, in this case all concepts relating to business entities and more broadly to the carrying out of business activities in the human world. So for example a human being would be considered on an astronomical scale as an occurrent thing, the difference in granularity in the time scales being determined according to the context in which the ontology is to be used. More precisely, a human being could still be considered as a Continuant Thing, with a human life being the corresponding Occurrent Thing, so in many cases it is reasonable to try to frame definitions of things which are clearly either continuant or occurrent.

For the avoidance of doubt, the partitioning of continuant from occurrent things is not formally represented by any axioms, and is definitional only. This means that terms in this model may be cross referenced to terms in models which use different formal ways of distinguishing continuant from occurrent things, for example what are called four dimensional, three dimensional, and similar modeling arrangements. The partitioning given in the model described in this specification contains no such assertions and is provided to enable the problem domain to be partitioned according to the basic nature of what is defined. This enables the model to contain concepts to do with events, processes, states and the like, though these are not utilized in the business entities semantic model.

Annex D: How to extend FIBO ontologies

(informative)

Intended Audiences: *The intended audience for this Annex is semantic modelers, who are expected to have some familiarity with the basic principles of semantic modeling but not necessarily with the principles specific to FIBO. Basic OWL principles are also reiterated here. This annex is not intended for purely business audiences or purely technical audiences.*

This Annex should be read in conjunction with the clause on Conformance (2).

D.1 Terminology used in this Annex

There are several sets of terminology in use throughout this specification, and the meanings of some terms (such as 'thing') may be different in different specialized usages. Here the intended sense of these words, unless otherwise stated, is the sense used for business communication of the ontology content, and not the sense used in technical modeling or conventional Semantic Web terminology. If a formal definition of a term is not given or referred to via the "Definitions" clause of this specification (4), the normal, English language sense of a word should be assumed, and not that of any technical body of knowledge or community of practice.

The model described in this specification follows the principles of the Web Ontology Language (OWL). This defines the concept of a 'Class' as a set theory construct and is not to be confused with the usage of the word ‘Class’ in the UML modeling paradigm. In descriptions aimed as business audiences, we usually use the word ‘Thing’ in place of this, and on the basis that the OWL library class “Thing” is the ultimate parent of all classes in an OWL model (so they are all things). This also precludes having to explain to a business audience the very nuanced distinctions between UML and OWL Classes. The specialized technical usage of the word 'Thing' to refer to an OWL individual is not the sense used in this Annex.

In this Annex, the term 'class' and 'thing' will be used interchangeably to describe the OWL classes as set theory constructs, that is in the natural language (dictionary) sense in which one speaks of classes of thing (for example in the sentence "what class of locomotive is this?" or "what class of animal is a fish?"). This corresponds to the OWL usage of the term but not (or not without some qualification) to the UML usage of the term.

D.2 Overview

D.2.1 Classes of Thing

In OWL and therefore in FIBO models, membership of a class may be defined intensionally by way of properties which define the membership (the extension) of that class, or extensionally by way of listing the members of the set which makes up that class.

In the model described in this specification, all classes are defined intensionally except where extensional models are unavoidable. The modeling notation employed here supports the definition of extensional classes but this is discouraged except for the definition of classes which are necessarily extensional such as days of the week.

D.2.2 Model relationship to Subject Matter

The formal statement by which everything in the model has an ultimate super-class which is the universal set of 'Thing' is the means by which this model is formally identified as being a business conceptual model and not a data model representation.

In order to preserve the integrity of the model as a model of business concepts, all classes which are added to the model must:

1. Be given a superclass (a class with which the new class has a sub-class relationship) from one of the existing classes in the model;
2. Represent something in the business domain itself, and
3. Represent a set of possible members which in all cases would also be members of the set defined by the super-class in (1)

D.2.3 How to Model New Classes

In modeling semantics, it is a requirement to model each new kind of "Thing" (hereafter referred to as 'classes') in the model according to the following two criteria:

- What kind of thing is this?
- What facts distinguish it from other things?

The consequence of addressing these questions is that for each kind (or class) of thing in the domain of discourse (in this case business entities and legal entities), this will be defined in terms of the following question:

> "What is the simplest kind of thing that this is one of?"

By defining classes in terms of simpler kinds of thing, future changes will be additive. This benefit only applies if each class in the model is adequately generalized into some more abstract concept.

Failure to adequately generalize classes of "Thing" in the taxonomic hierarchy will have the result that future additions to that part of the taxonomy may prove to be disruptive. When the model is extended in the future to cover additional concepts, if the model components are not adequately abstracted then it will become necessary to break the existing chain of generalization to interpose new terms to support these new concepts. It is therefore important that modelers exercise imagination in this regard.

D.2.4 Declaring Class Disjointness

A disjointness relationship indicates that two classes of thing are mutually exclusive, that is that members of one may not also be members of the other.

Class disjointness refers to the situation whereby the members of one class may not also be members of another class when there is a disjoint relationship between the two. In OWL this relationship uses the 'isDisjoint' construct.

New 'isDisjoint' relationships should be labeled with the natural language label of "mutually exclusive"

Classes may have several separate sets of sub-classes which are mutually disjoint.

Note that disjointness is inherited through sub-class relationships. If a disjoint is misapplied this may cause inconsistencies. Conversely, if there is an inconsistency and disjointness has been correctly applied, then somewhere in the model there is an incorrect statement which would assert that some individual may be a member of more than one mutually disjoint class. The application of disjoint relationships therefore provides a useful diagnostic for subsequent extensions to the model, provided it is implemented correctly.

D.2.5 How to Model New Facts about Things

There are two kinds of "fact" in the model (in formal modeling terms, two kinds of "Property"):

1. Relationship Properties (known in OWL as Object Properties);
2. Simple Properties (known in OWL as Datatype Properties)

These are similar in their intent, in that they assert something about the class of which they are a property, but are shown differently in model diagrams.

Facts (properties) should be presented in the model only at the level of the class to which they apply. If a fact is not always applicable or relevant to the meaning of some concept, it should be applied to one or more sub-types of that class

where it would be applicable. Similarly a property should not be applied to sub-classes where they would not always be true.

As an example, vertebrates are a class of things which are an animal and which have a backbone. It would not be appropriate to model the term "has backbone" as an optional property of all animals. Nor would it be sensible to say, for each class of things which is a vertebrate, that this class of vertebrates also has a backbone.

Note that there is a difference here from data modeling. In a data model it may be more efficient to assign a property to a class, make it optional, and then have some sub-classes which use that property and some which do not. This is appropriate for a data model because such a model is not intended to convey the meanings of those classes; rather, the user of the model has to know which sub-classes would have data for that property and which of them would not. In contrast, the semantic model in FIBO is intended to convey the knowledge that such a user would need to have. For this reason, considerations of efficiency which would be brought to bear on a data model design exercise, should not be considered when extending FIBO models.

Impact on Sub-classes

When adding a new Relationship Property or Simple Property to an existing class, ensure that this fact would be true of all the classes that are sub-classes of this class, and that are sub-classes of their classes and so on. If the meaning asserted by the addition of the new property is not necessarily true of all the descendent classes of thing, then it would not be correct to add it to this class. Instead it should be added to those of the sub-classes to which it does apply (that is, those to which it contributes something of the meaning of what it is to be a member of that class).

If there is a clearly identifiable group of those sub-classes for which the property is applicable, then it is possible that these could be grouped together as a new sub-class with that property. However, the addition of such a class, being as it would be interposed into an existing class hierarchy, should be handled with care - this constitutes a disruptive rather than an additive change, and will have different and more stringent change management requirements.

Adding a Relationship Property

Wherever possible, a Relationship Property should be a specialization of another Relationship Property which is already in the model. When adding the Relationship Property, the RDF construct "subPropertyOf" should be used to assert what is the parent property.

The new property should extend or refine the meaning of the parent property in some way.

It is also allowable to have more than one parent property. This is appropriate in cases where the meaning of one Relationship Property is recognizably derivable from the meanings of two or more other Relationship Properties. This construction should be used sparingly and with care.

Types of Relationship Property

In terms of the OWL language, there are a number of distinctions between kinds of relationship which may be asserted in this model. For example, it is possible to assert that a relationship is symmetric, or that it is 'functional'. Functional relationships are relationships where only one individual of the type that's shown as the range of the property, may be that thing.

In the UML modeling environment, the information about what kind of relationship a given relationship is, is provided by means of tagged values.

At present the terms distinguishing different types of relationship are not widely used in the model. If in doubt, relationships should be added without attempting to populate this information.

When adding a new relationship and making it a sub-property of some existing relationship, modelers should check the parent relationship and any of its parents, to verify whether these are defined as being one of these specialized types of OWL object property. If they are, then the new relationship will also take on this type, so modelers must ensure that this would be correct for the relationship being added.

Adding a Simple Property

Simple Properties may only have a range (the object of the predicate) which is a simple information type or an enumerated data range.

The simple information types may be found in the model sub clause "Business Types". These include concepts such as text, numbers, dates and yes/no answers.

Simple Properties should not have ranges which are technical datatypes (the XML primitive datatype set or the datatypes made available within a UML modeling framework). XML primitive datatypes are allowable in RDF/XML based OWL ontologies, and would be used in an operational ontology derived from these models, but for the purposes of business understanding of the model these are all either given aliases (like 'yes/no' for boolean), or have more detailed types derived from them such as the various kinds of number.

There are no "Complex Types" in FIBO. For presentation purposes in different UML editing environments it is possible to consider rendering certain Relationship Properties (OWL object properties) as if they were simple types, i.e. using the UML "attribute" construct, but this is not formally supported in the sub-set of ODM defined in this specification. If this technique is used, such properties must be formally identified as OWL object properties; datatypes properties may not refer to classes which themselves have properties, such as monetary amounts or dated values.

D.2.6 Inverse Relationships

Whenever two relationships are in an inverse pair, this must be indicated by adding a relationship between those relationships, using the OWL construct 'inverseOf'. This should be labeled with the natural language label of 'inverse'.

Many Relationship Properties about things in the real world come in pairs, where one is the inverse of the other. For example "Account held by Account Holder" and "Account Holder holds Account" are two ways of saying the same thing, from the two perspectives of the Account and the Account Holder.

All relationships in the semantic notation used here and in the Semantic Web are unidirectional, that is they are 'triples' of the form Sub verb Object.

This is different to the way relationships are treated in data modeling. The 'ends' of a relationship in a data modeling format may be considered as being analogous to the separate relationships in a semantic model.

When to add these: Where it is considered relevant in defining the meanings of concepts, Relationship Properties (other than symmetric ones - see 'Types of Relationship Property') may also be given an inverse. It is not a formal requirement to indicate all the inverses that may possibly exist. Such relationships should be present in the model and extensions to the model if the two senses are in common use, if they correspond to a named term for which there is a formal definition in use in the financial industry, or if Relationship Properties that are commonly defined for sub-types of the class that they are a fact about, are commonly specified or referred to in the opposite direction to the one which has already been specified.

For this reason, the addition of new classes of thing in the model, given that these specialize existing things, may sometimes require the addition of the inverse of some existing Relationship Property, which was previously implied but not present as a property in the model.

D.2.7 How and When to Use Enumerations

There are two kinds of enumeration in the modeling notation:

- Enumerated Data Range
- Enumerated Class

Enumerated data ranges look a lot like enumerated datatypes in data models. However, these are used differently and will not usually correspond.

The 'Enumerated Data Range' construct should be used to enumerate possible data literals, that is pieces of text, numbers and so on, any one and only one of which may be the literal value of that datatype property for one instance of that class.

Where a data model enumerations may enumerate types of real thing and are frequently used to "flag" some class to say what kind of thing this is, this arrangement cannot be used in the FIBO semantic model. If a class of thing may be of several types, then these should be modeled as distinct classes, each of them a sub-class of the class of thing that they are all types of.

Where a class is to be defined by enumerating its members (extensional definition of the class), then the class itself should be modeled not as an OWL Class but as an OWL Enumeration Class.

D.2.8 Foundations Concepts Usage

Because it was a requirement that classes of thing be abstracted to their simplest possible types, the modeling already carried out in FIBO necessarily required the creation of a set of classes which, by their nature, are not unique to business entities or financial services terms and definitions.

There is a second scenario in which terms are required which are not unique to financial services. This is when a relationships fact (OWL object property) about some business entity has a relationship to something which is not itself a concept unique to the context of the financial services sector.

The terms which are not unique to the financial services sector are maintained in a separate part of the model repository and are given a separate namespace. These are packaged as the FIBO Foundations ontologies. Use of the appropriate terms in these ontologies is normative for this specification, but in many cases these ontologies are being evolved, improved upon and better aligned with other publicly available standard ontologies and with relevant academic work.

In Semantic Web terms, these are mid level ontologies. These are additionally supplemented by the inclusion of an "Upper Ontology" consisting of three sets of underspecified, high level partitions into which all model content is divided.

When adding new classes or Relationship Properties, modelers should seek out and select concepts from within the Foundations ontologies which represent the terms they need to specialize or refer to. They should also recognize and adequately respect the 'Archetype' of that term, if available. In particular, the ontology partitions under which the required archetype term resides should be inspected and understood, in order not to give rise to inconsistencies in the resultant ontology.

New general terms should not be added without first seeking the appropriate terms in these Foundations ontologies or in some recognized external ontology, which must itself be cross referenced using one of the methods described in Annex C (Shared Semantics Treatments), in order to create the necessary relationships.

D.2.9 Content Creation Summary

In summary, there are two scenarios where classes of thing are needed in any ontology for business entities, for financial securities, loans, derivatives and so on:

- The kind of "Thing" which something is;
- Things which are referred to in facts about things.

The first question will lead the modeler to find a more general class of thing of which to make the new class a sub-class. This should be sought initially in the ontology which is being extended, and after exhausting this, in the appropriate Foundations ontology, which must be inspected and fully understood before implementing the new sub-class ('is a') relationship.

The second question will lead the modeler to seek out the appropriate class of thing to which they need to refer. Often, but not necessarily, this will require the creation of some new class of thing. For example, a new class of 'Interest Payment Terms' might be appropriate in order to define a property of a new class of interest-bearing instrument which is defined by way of unique interest payment terms.

Modelers should look in the first instance for some class of thing which is exactly appropriate to the new relationship. For example, concepts like "Monetary Amount" or "Dated Monetary Amount" may be appropriate targets ("Ranges" in Semantic Web parlance) for more than one Relationship Property about more than one class of thing.

In the absence of such a class, modelers should add a suitable sub-class of some existing class of thing which is broader

in meaning but otherwise identical to the class to which the new Relationship Property is to refer. In the interest payment terms example above, they would add a new sub-type of the class which is 'Interest Payment Terms Set' or perhaps 'Fixed Interest Payment Terms Set' or 'Bond Fixed Interest Payment Terms Set' as appropriate. This should be labeled with a suitably business-facing label which uniquely describes it within that ontology and which as far as possible reflects what is unique about its meaning (note that meanings do not follow from these labels, but that business comprehension of the model follows from their allocation).

Where a term is not available for specialization within the ontology which the modeler is extending, these are to be found in the FIBO Foundations ontologies, which have been created for the purpose of providing such terms. These are ontologies of things which are not specific to financial services. These include legal concepts like contracts, business concepts such as service provision, as well as an extensive set of concepts for times, dates, mathematical constructs, events and activities, and so on.

If a suitable general term cannot be found then it may be necessary to extend one of the FIBO Foundations ontologies. This should be undertaken as a collaborative effort since this term will almost certainly be needed again in the future and by others. Such terms should be defined with formal reference to other, publicly available ontologies (these being defined either in Semantic Web formats or in some presentation, notation of theoretical grounding which makes it unambiguously clear that the terms in question are not part of a data model or other logical design).

D.3 Presentation Considerations

The presentation conformance requirements described in this specification are mainly a consideration for those creating or setting up editing environments in different modeling tools, and are not covered in this Annex. However, in the course of creating extensions to the model content there are a number of considerations which the modeler should keep in mind, as described in this sub clause.

D.3.1 Labeling

All classes, Relationship Properties and Simple Properties should be given natural language labels. These should be rendered with spaces just as normal text is written.

These labels should conform to the following style requirements:

- Classes: Names should be in Upper Sentence Case
    - Abbreviations (if used) should be in their normal upper case rendition e.g. ABC.
    - Small words (of, and etc.) should also be capitalized (this is to enable technical users to compress the names without loss of sense)
- Relationship Properties: Names should take the form Subject predicate Object with the casing as shown
    - Subject and Object to have the full name of the classes themselves except where this is cumbersome
    - The predicate (verb part) of the relationship name should be in all lower case, with spaces
        - If possible, relationship lines (which are displayed in 'simple' diagrams that don't have the boxes that come with the Relationship Properties), should be labeled with only the predicate.
- Simple Properties: Names should be in Upper Sentence Case
- Other types of "Thing" construct (OWL Union Classes, Intersection Classes, Enumerated Classes and Enumerated Data Ranges) should follow the same naming convention as classes.

In addition to the above constructs, which define the terms in the business domain, there are a number of built in constructs which make additional statements, in set theory terms, about the classes and properties. These should be labeled as follows:

- Logical Union relationships: these are rendered using the UML construct of a generalization set (UML "GeneralizationSet"). Such sets have one name. This name should be a natural language label, with spaces and

in lower case. The label should make clear the sense that it is a union relationship defining the logical union of the classes which participate in the generalization set, for example by ending the label with the word 'union'.

- Disjoints (OWL disjointWith): should always have the label "mutually exclusive"
- Inverses of relationships (OWL inverseOf): should always have the label "inverse"

D.3.2 Ontologies

These are implemented using the UML base class of 'Package'. Names for these should be in Upper Sentence Case. Wherever possible short or one word names should be considered.

D.3.3 UML Considerations

UML Diagrams

Diagrams are not transferred from any modeling environment into or out of the model repository. Diagrams are to be created by the modeler for presentation to business domain experts in the area in which they are working, or in the case of new submissions of the model content for future updates, to the wider community, and must be designed to be readable by business domain experts.

UML Notation

No explicitly UML notation should be present on any diagram.

The guiding principle here is one of language: any diagram which includes anything which belongs in or looks as though it belongs in some technical notation, will signal to the business reviewer that this diagram is in a language for which they have had no formal training. No matter how obvious the meaning of a diagram appears to be, the appearance of any technical notation means that it will appear to be something that requires some technical training to parse its meaning.

This means that

- no repurposed punctuation marks may be present on the diagrams. For example:
    - no curly braces and therefore no OCL
    - no guillemets - so stereotype indications must be disabled
    - no plus signs at the ends of relationships or next to attribute names
- UML class partitions that are unused (such as the operations partition) must be made invisible - either by manually resizing the class box until the extra line disappears, or by some other means;
- Exceptions may be made for relationship multiplicities, but the implications of these must be clearly explained to business domain experts who are expected to review the model content
- The Generalization arrowhead is an exception to the above: although this represents a technical notation (Generalization in UML), its meaning is more universal and can be explained to business domain experts ahead of any review. Such explanations must either reference Aristotelian syllogisms or be described in terms of the "is a" relationship with examples from natural taxonomy, depending on the knowledge of the business audience, but should not make reference to UML or words like Generalization or transitivity.
- Namespace indications: in some tools these are indicated with a double colon, which breaks the first rule above. Diagrams with these on may be created and maintained so that maintainers of the content can keep track of what is in what ontology, but these diagrams should not be considered as suitable for general business domain distribution.

Diagram Layout

Modelers should take care to lay out these in a clear and consistent way.

Generalization relationships should be laid out with the "arrowhead" pointing vertically upwards, in either the vertical tree style or direct style of routing. This is because this relationship, while technology neutral (it represents a basic Aristotelian syllogism), has to be explained to business domain experts and should therefore be presented in the same visual layout in which it has been explained, namely to represent taxonomic hierarchies with the most general terms at the top and the most specific at the bottom. These generalization relationships should never be drawn or found pointing downwards or sideways.

Where possible, the physical arrangement of the concepts in a diagram should try to follow the layout of the corresponding concepts in the archetype diagrams for those concepts.

Where large numbers of concepts are found in the same ontology, modelers should try to create separate diagrams which emphasize separate aspects of the subject matter (for example segregating contractual terms from legal obligations, or events from parties).

The relationship sub-property relationships are a particular hazard to creating clear, clean diagrams. However, these should rarely be shown to business domain experts. Where practicable, modelers are encouraged to create, for each separate thematic diagram, a set of three diagrams: one with all the material that needed to be modeled, one without the class component of the Relationship Properties, and one without the Simple Properties (compressing the class glyph as needed to remove the appearance of the attributes partition boundary).

Diagram Notes

Diagrams may also be decorated with informative notes. However, nothing of substance to the model content should be included in these, since these will not be retained when the model is transferred into the model repository or into other modeling environments.

UML Diagram Boundaries

As with notes, these may be included in business diagrams to aid in readability, but these UML boundaries do not form part of the model content and are not retained when the model content is transferred between environments.

UML Packages

UML Packages do not form part of the model, unless the package is stereotyped as an OWL Ontology.

OWL ontology packages may not be nested within other OWL ontology packages.

Modelers may arrange packages as appropriate for the usage to which they intend to put the model, and as part of this they may elect to make hierarchical structures of packages. Packages which are not stereotypes as OWL ontologies may be used for the purposes of such organization. Such packages may only contain other such packages or OWL ontology packages (that is, they should contain no loose classes or other constructs). Such packages do not form part of the model content, and will not be retained when the model content is transferred between environments.

No relationships between packages should be interpreted as, or created to imply, any relationship between ontologies.

All ontology imports must be explicitly modeled using the ODM "owlImports" construct. Each ontology should contain a diagram showing the full set of OWL imports required for that ontology, up to and including the "Lattice" ontology.

Annex E: Creating Applications with FIBO (Informative)

E1. Introduction

This annex contains guidelines on the production of operational applications that take the various FIBO Business Conceptual Ontologies as a point of reference. Such applications include operational OWL ontologies and applications based on conventional data models. The sub clauses below set out the overarching principles for creating such applications, and itemize the things to consider when deriving operational ontologies or logical data models from the content in those FIBO specifications.

E.1.1 Principles

These are the basic principles in order to avoid making assertions which contradict those assertions already made in FIBO:

1. It is not necessary to include all the ancestor classes but disjoints asserted between those ancestor classes must be respected
2. Two classes cannot be introduced into the same logical class hierarchy which have ancestors which are disjoint in FIBO. This is because otherwise it becomes possible to introduce contradictions or data structures which correspond to contradictory or untrue (or absurd) facts about the world.
3. Relationships which have restrictions defined for them (for example functional object properties) may not be extended to have looser multiplicity in logical data models but they may be further restricted.
4. New facts or relationships should not be introduced which directly contradict some fact in the FIBO terms which are used, or in any FIBO terms which are not directly used but which *have a bearing* on the terms which are used.

E.1.2 Operational Ontologies

The following questions are to be considered when creating an operational OWL ontology using terms set out in one or more of the FIBO Business Conceptual Ontologies:

- When to replace an object property with a Boolean
- Shortening the inheritance hierarchy
- Using independent things without relative things
- Redefining Relative Things as Independent Things
    - This is valid when the context of the application matches the “Mediating Thing” that is the context in which the Relative Thing is defined
    - Example: Legal Entity is a relative thing but for an application whose scope is constrained to one jurisdiction or LEI issuer, it can be treated as an Independent Thing
- Use of property chains
- Extraction of single-inheritance (monohierarchical) taxonomy
    - May also be conformant, as a sub-set of the FIBO material
- OWL Restrictions versus rdfsSubPropertyOf relations between multiple object properties.

E.1.3 Conventional Applications

The following questions are to be considered when creating a logical data model using terms set out in one or more of the FIBO Business Conceptual Ontologies:

- Possible architectures
    - Use of semantically under-specified classes, with enumerations to identify semantics
    - Other styles –e.g. a direct rendition of the ontology with addition of database keys
- General
    - Enumerations – don't have mixed semantics in one enumerated datatype (causes combinational explosions)
    - Text: when to collapse a chain of properties that end in a text field, with just an attribute that has text as a datatype
    - Combining pairs of object properties into one association – with the object property names as the labels of the ends of the association
- UML considerations
    - When to render object properties with a specific archetype, as UML Associations or Generalizations
    - Multiplicity
- Relative Things
    - These may be treated as independent classes when the context of the application matches the "Mediating Thing" that is the context in which the Relative Thing is defined
    - Example: Legal Entity is a relative thing but for an application whose scope is constrained to one jurisdiction or LEI issuer, it can be treated as an Independent Thing
- Localization within a part of the taxonomy
    - Patterns for taking a starting point within the hierarchy (e.g. MBS versus Bond versus Security), and navigating each of the object properties that apply at that level, navigating downwards (but not upwards) in the taxonomy of things that are the range of the object property, and defining these as the full possible scope of the model
- Extraction via Context
    - From a given "Mediating Thing", navigate to each of the "Relative Things" defined in that context, and each of the "Independent Things" that may take on the "identity" property of those relative things – this should result in a set of all and only those things needed for the application