# FTF Report

**of the**

## Ontology Definition Metamodel Finalization Task Force 2

**to the**

## Platform Technical Committee

**of the**

## Object Management Group

## 25 August 2008

**Document Number: ptc/2008-08-04**

**Accompanied by:**

- **ptc/2008-08-05, revised specification**
- **ptc/2008-08-06, CMOF XMI file**
- **ptc/2008-08-07, EMF compatible UML2 XMI file**
- **ptc/2008-08-08, Issue status spreadsheet**

**Template: omg/04-02-02**

# Table of Contents


**Summary of ODM FTF2 Activities** 1
*Formation* ... *1*
*Revision / Finalization Task Force Membership* ... *1*
*Issue Disposition:* ... *2*
*Voting Record:* ... *3*
*Summary of Changes Made* ... *4*
**Disposition: Resolved** **5**
*OMG Issue No: 10845* ... *5*
Title: Annex D.4 typo 5
**Disposition: Resolved** **6**
*OMG Issue No: 10848* ... *6*
Title: Metalevels 6
**Disposition: Resolved** **7**
*OMG Issue No: 10851* ... *7*
Title: Classes and properties 7
**Disposition: Resolved** **8**
*OMG Issue No: 10859* ... *8*
Title: Name as instance 8
**Disposition: Resolved** **9**
*OMG Issue No: 10869* ... *9*
Title: N-aries 9
**Disposition: Resolved** **10**
*OMG Issue No: 10878* ... *10*
Title: Restriction 10
**Disposition: Resolved** **11**
*OMG Issue No: 10880 & 10882* ... *11*
Title: Constants 11
**Disposition: Resolved** **12**
*OMG Issue No: 10881* ... *12*
Title: Herbrand Semantics 12
**Disposition: Resolved** **14**
*OMG Issue No: 10883* ... *14*
Title: All / SomeValuesFrom 14
**Disposition: Resolved** **15**
*OMG Issue No: 10896* ... *15*
Title: Behavioral Features 15
**Disposition: Resolved** **17**
*OMG Issue No: 10903* ... *17*
Title: Symmetric 17
**Disposition: Resolved** **18**
*OMG Issue No: 12386* ... *18*
Title: RDFSDatatype uriRef Property Multiplicity Issue 18
**Disposition: Resolved** **20**
*OMG Issue No: 12387* ... *20*
Title: RDFSisDefinedBy text issue 20
**Disposition: Resolved** **21**
*OMG Issue No: 12388* ... *21*
Title: RDFSseeAlso text issue 21
**Disposition: Resolved** **22**
*OMG Issue No: 12389* ... *22*
Title: RDFType text issue 22

**Disposition: Resolved** **23**
*OMG Issue No: 12391* ..... *23*
Title: OWL Annotation properties are impoverished in the OWL profile 23
**Disposition: Resolved** **24**
*OMG Issue No: 12392* ..... *24*
Title: URIReferenceNode definition is misplaced in the specification 24
**Disposition: Resolved** **25**
*OMG Issue No: 12393* ..... *25*
Title: URIReferenceNode constraint text 25
**Disposition: Resolved** **26**
*OMG Issue No: 12395* ..... *26*
Title: OWL Cardinality Constraints text revision 26
**Disposition: Resolved** **27**
*OMG Issue No: 12396* ..... *27*
Title: OWL Properties have conflicting definitions 27
**Disposition: Resolved** **28**
*OMG Issue No: 12397* ..... *28*
Title: No stereotype for OWL individuals 28
**Disposition: Resolved** **30**
*OMG Issue No: 12398* ..... *30*
Title: Definition of RDF Property and its use in the figures is inconsistent 30
**Disposition: Resolved** **40**
*OMG Issue No: 12793* ..... *40*
Title: Design of RDF metamodel for rdf:Statement, triple, and graph controversial in semantic web community 40
**10.4 RDFBase Package, RDF Statements** **44**
**10.5 RDFBase Package, RDF Graphs** **46**
**Disposition: Deferred** **52**
*OMG Issue No: 10844* ..... *52*
Title: Figure D.3 notation 52
**Disposition: Deferred** **53**
*OMG Issue No: 10846* ..... *53*
Title: Annex D.4 sets 53
**Disposition: Deferred** **54**
*OMG Issue No: 10849* ..... *54*
Title: Figure 16.1 incomplete 54
**Disposition: Deferred** **55**
*OMG Issue No: 10850* ..... *55*
Title: Formal structure 55
**Disposition: Deferred** **56**
*OMG Issue No: 10853* ..... *56*
Title: Associations 56
**Disposition: Deferred** **57**
*OMG Issue No: 10863* ..... *57*
Title: Distinct associations, ownedAttribute associations 57
**Disposition: Deferred** **58**
*OMG Issue No: 10864* ..... *58*
Title: Distinct associations, restrictions 58
**Disposition: Deferred** **59**
*OMG Issue No: 10865* ..... *59*
Title: Identifiers 59
**Disposition: Deferred** **60**
*OMG Issue No: 10866* ..... *60*
Title: Associations 60
**Disposition: Deferred** **61**

*OMG Issue No: 10867* ..... *61*
Title: Subproperties and redefinition 61
**Disposition: Deferred** **62**
*OMG Issue No: 10871* ..... *62*
Title: Association member ends 62
**Disposition: Deferred** **63**
*OMG Issue No: 10872* ..... *63*
Title: Table 16.9 and Naries 63
**Disposition: Deferred** **64**
*OMG Issue No: 10873* ..... *64*
Title: Translation of binary associations 64
**Disposition: Deferred** **65**
*OMG Issue No: 10874* ..... *65*
Title: UML Thing 2 65
**Disposition: Deferred** **66**
*OMG Issue No: 10875* ..... *66*
Title: Individuals 66
**Disposition: Deferred** **67**
*OMG Issue No: 10876* ..... *67*
Title: Disjoint 67
**Disposition: Deferred** **68**
*OMG Issue No: 10877* ..... *68*
Title: Classes of classes 68
**Disposition: Deferred** **69**
*OMG Issue No: 10879* ..... *69*
Title: Mandatory properties 69
**Disposition: Deferred** **70**
*OMG Issue No: 10884* ..... *70*
Title: Derivation 70
**Disposition: Deferred** **71**
*OMG Issue No: 10885* ..... *71*
Title: Table 16.10 71
**Disposition: Deferred** **72**
*OMG Issue No: 10886* ..... *72*
Title: Table 16.11 72
**Disposition: Deferred** **73**
*OMG Issue No: 10887* ..... *73*
Title: Table 16.12, Thing 73
**Disposition: Deferred** **74**
*OMG Issue No: 10888* ..... *74*
Title: Table 16.12, AllValuesFrom 74
**Disposition: Deferred** **75**
*OMG Issue No: 10889* ..... *75*
Title: Table 16.12, classes as instances 75
**Disposition: Deferred** **76**
*OMG Issue No: 10890* ..... *76*
Title: Table 16.12, disjoint 76
**Disposition: Deferred** **77**
*OMG Issue No: 10891* ..... *77*
Title: Inferring subsumption 77
**Disposition: Deferred** **78**
*OMG Issue No: 10892* ..... *78*
Title: Boolean combination 78
**Disposition: Deferred** **79**
*OMG Issue No: 10893* ..... *79*

Title: Names, unique names 79
**Disposition: Deferred** **80**
*OMG Issue No: 10894* ..... *80*
Title: Names, UML namespaces 80
**Disposition: Deferred** **81**
*OMG Issue No: 10895* ..... *81*
Title: Other OWL 81
**Disposition: Deferred** **82**
*OMG Issue No: 10897* ..... *82*
Title: Complex Objects 82
**Disposition: Deferred** **83**
*OMG Issue No: 10898* ..... *83*
Title: Keywords 83
**Disposition: Deferred** **84**
*OMG Issue No: 10899* ..... *84*
Title: Profiles 84
**Disposition: Deferred** **85**
*OMG Issue No: 10900* ..... *85*
Title: UML to OWL, OWL-DL 85
**Disposition: Deferred** **86**
*OMG Issue No: 10901* ..... *86*
Title: UML to OWL, Table 16.10 86
**Disposition: Deferred** **87**
*OMG Issue No: 10902* ..... *87*
Title: Object identification in UML 87
**Disposition: Deferred** **88**
*OMG Issue No: 10904* ..... *88*
Title: N-aries. Section 16.3.6 88
**Disposition: Deferred** **89**
*OMG Issue No: 10905* ..... *89*
Title: Multiplicity 89
**Disposition: Deferred** **90**
*OMG Issue No: 10906* ..... *90*
Title: navigableOwnedEnd 90
**Disposition: Deferred** **91**
*OMG Issue No: 10907* ..... *91*
Title: Enumeration literals 91
**Disposition: Deferred** **92**
*OMG Issue No: 10908* ..... *92*
Title: Individuals, mapping 92
**Disposition: Deferred** **93**
*OMG Issue No: 10909* ..... *93*
Title: complementOf and disjointWith 93
**Disposition: Deferred** **94**
*OMG Issue No: 10910* ..... *94*
Title: Multiple Domains or Ranges for Properties 94
**Disposition: Deferred** **95**
*OMG Issue No: 10911* ..... *95*
Title: Ontology Properties 95
**Disposition: Deferred** **96**
*OMG Issue No: 10912* ..... *96*
Title: Anonymous Classes 96
**Disposition: Deferred** **97**
*OMG Issue No: 10913* ..... *97*
Title: Universal Superclass 97

**Disposition: Deferred** **98**
*OMG Issue No: 10914* ..... *98*
Title: Constructed Classes 98
**Disposition: Deferred** **99**
*OMG Issue No: 10915* ..... *99*
Title: Range Restriction Restriction Classes 99
**Disposition: Deferred** **100**
*OMG Issue No: 10916* ..... *100*
Title: Range Restriction Restriction Classes 100
**Disposition: Deferred** **101**
*OMG Issue No: 10917* ..... *101*
Title: Properties in OWL 101
**Disposition: Deferred** **102**
*OMG Issue No: 11099* ..... *102*
*Title: Constraints in the RDF Metamodel Chapter (10) should be specified in OCL* ..... *102*
**Disposition: Deferred** **103**
*OMG Issue No: 11100* ..... *103*
*Title: Constraints in the OWL Metamodel Chapter (10) should be specified in OCL* ..... *103*
**Disposition: Deferred** **104**
*OMG Issue No: 11102* ..... *104*
*Title: Mapping from Common Logic to OWL should be revised* ..... *104*
**Disposition: Deferred** **105**
*OMG Issue No: 11320* ..... *105*
*Title: Thing in the Profile* ..... *105*
**Disposition: Deferred** **106**
*OMG Issue No: 11321* ..... *106*
*Title: RDFSContainer-MembershipProperty* ..... *106*
**Disposition: Closed, no change** **107**
*OMG Issue No: 10843* ..... *107*
Title: Annex D.2, OWL Full 107
**Disposition: Duplicate/merged** **108**
*OMG Issue No: 10882* ..... *108*
Title: Transitive Closure 108
**Disposition: Duplicate/merged** **109**
*OMG Issue No: 11107* ..... *109*
Title: Set value for annotation property in UML profile for OWL 109

# Summary of ODM FTF2 Activities

## Formation

- Chartered By: Platform Technical Committee
- On: September 28, 2007 in Jacksonville, Florida
- Comments Due Date: February 22, 2008
- Report Due Date: July 3, 2008, revised to September 29, 2008

## Revision / Finalization Task Force Membership

| Member | Organization | Status |
|---|---|---|
| Pete Rivett | Adaptive | charter |
| Roger Burkhart | Deere & Company | charter |
| Peter Haase | Forschungszentrum Informatik | charter, but removed ~1 May 2008 |
| Jishnu Mukerji | Hewlett-Packard | charter |
| Chris Welty | International Business Machines | charter, but removed ~1 May 2008 |
| James Odell | Kabira Technologies | charter, but removed ~25 April 2008 |
| Evan Wallace | NIST | charter (coChair) |
| Ho Kit Robert Ong | No Magic | charter |
| Roy Bell | Raytheon | charter |
| Elisa Kendall | Sandpiper Software | charter (coChair) |
| Ravi Sharma | Vangent | Added 14 March 2008, removed ~11 August 2008 |
| | | |

## Issue Disposition:

| Disposition | Number of Occurrences | Meaning of Disposition |
|---|---|---|
| Resolved | 24 | The RTF/FTF agreed that there is a problem that needs fixing, and has proposed a resolution (which may or may not agree with any resolution the issue submitter proposed) |
| Deferred | 55 | The RTF/FTF agrees that there is a problem that needs fixing, but could not agree on a resolution and deferred its resolution to a future RTF/FTF. |
| Transferred | 0 | The RTF/FTF decided that the issue report relates to another specification, and recommends that it be transferred to the relevant RTF. |
| Closed, no change | 1 | The RTF/FTF decided that the issue report does not, in fact, identify a problem with this (or any other) OMG specification. |
| Duplicate or merged | 2 | This issue is either an exact duplicate of another issue, or very closely related to another issue: see that issue for disposition. |

## Voting Record:

| Poll No. | Closing date | Issues included |
|---|---|---|
| 1 | 14 March 2008 | 10848, 10851, 10859, and 10878 |
| 2 | 30 April 2008 | 10869, 10880-10883, 10896, and 10903 |
| 3 | 19 May 2008 | 12386-12389, 12391-12393, and 12395-12397 |
| 4 | 11 August 2008 | 12398 |
| 5 | 25 August 2008 | 12793 |
| 6 | 25 August 2008 | 10843-10846, 10849, 10850, 10853, 10863-10867, 10871-10877, 10879, 10884-10895, 10897-10902, 10904-10917, 11099, 11100, 11102, 11107, 11320, 11321 |

| Voter | Vote in poll 1 | Vote in poll 2 | Vote in poll 3 | Vote in poll 4 | Vote in poll 5 | Vote in poll 6 |
|---|---|---|---|---|---|---|
| Pete Rivett, Adaptive | Yes | Did not vote | Abstain | Yes | Did not vote | Yes |
| Roger Burkhart, Deere & Co. | Did not vote | Yes | Yes | Yes | Yes | Yes |
| Peter Haase, FZI | Did not vote | Did not vote | n/a | n/a | n/a | n/a |
| Jishnu Mukerji, HP | Yes | Did not vote | Yes | Yes | Yes | Yes |
| Chris Welty, IBM | Did not vote | Did not vote | n/a | n/a | n/a | n/a |
| Jim Odell, Kabira | Did not vote | Did not vote | n/a | n/a | n/a | n/a |
| Evan Wallace, NIST | Yes | Yes | Yes | Abstain | Yes | Yes |
| Robert Ong, No Magic | Yes | Yes | Yes | Did not vote | Yes | Did not vote |
| Roy Bell, Raytheon | Yes | Yes | Yes | Yes | Yes | Yes |
| Elisa Kendall, Sandpiper Software | Yes | Yes | Yes | Yes | Yes | Yes |
| Ravi Sharma, Vangent | Yes | Yes | Yes | Yes | n/a | n/a |

## Summary of Changes Made

The ODM FTF2 made changes that:

- Added features that improved the usability of the profiles and improved their suitability to original purpose;
- Corrected features that impeded implementation and/or did not properly match the language being modeled;
- Increased readability, clarity, correctness, and precision of the text.

The following is a table that categorizes the issues as to the degree of changes that were made in resolving them.

| Extent of Change | Number of Issues | OMG Issue Numbers |
|---|---|---|
| **Significant** - Fixed problems with normative parts of the specification that raised concern about implementability | **4** | **12391 and 11107, 12397, 12398** |
| **Minor** - Fixed minor problems with normative parts of the specification | **8** | **12386, 12387, 12388, 12389, 12393, 12395, 12396, 12793**[1] |
| **Support Text** -Changes to descriptive, explanatory, or supporting material. | **13** | **10845, 10848, 10851, 10859, 10878, 10869, 10880, 10881, 10882, 10883, 10896, 10803, 12392** |

[1] The problem described by 12793 was minor with respect to implementation of the metamodel, but appeared a major departure from RDF and OWL to certain semantic web language experts.

# Disposition: Resolved

## OMG Issue No: 10845

### Title: Annex D.4 typo

**Source:**

NIST (Mr. Conrad Bock, conrad.bock@nist.gov conradb@cme.nist.gov)

**Summary:**

Annex D.4 typo. Annex D.4, second sentence, "OntoClear" should be "OntoClean".

**Resolution:**

Replace text as requested.

**Revised Text:**

Change portion of second sentence in D.4 that reads:

> OntoClear system [OntoClean].

with

> OntoClean system [OntoClean].

**Disposition: Resolved**

# Disposition: Resolved

## OMG Issue No: 10848

### Title: Metalevels

#### Source:

NIST (Mr. Conrad Bock, conrad.bock@nist.gov conradb@cme.nist.gov)

#### Summary:

Metalevels. In Chapter 16 (Mapping UML to OWL), third paragraph, first sentence, before the bullets, should refer to "models", rather than UML models. It can also refer the reader to more examples and explanation in Sections 7.9 through 7.12 of [UML Infrastructure, http://doc.omg.org/formal/07-02-06

#### Resolution:

Replace text as described below.

#### Revised Text:

Replace first sentence of third paragraph of chapter 16 which currently reads:

> UML models are organized in a series of metalevels: M3, M2, M1 and M0, as follows:

with

> UML models are defined within a larger meta-modeling framework standardized by OMG. This framework defines a series of metalevels for UML:

**Disposition: Resolved**

# Disposition: Resolved

## OMG Issue No: 10851

### Title: Classes and properties

#### Source:

NIST (Mr. Conrad Bock, conrad.bock@nist.gov conradb@cme.nist.gov)

#### Summary:

Classes and properties. In Section 16.2.1 (UML Kernel), Under Figure 16.1, fifth bullet, properties do not implement classes (the "implementation" usually refers to how the model is translated to a platform). UML properties have the same semantics as OWL properties. Classes do not necessarily have properties. See multiplicity from Class to Property in the UML spec.

#### Resolution:

Replace text as described below.

#### Revised Text:

Replace fifth bullet following figure in section 16.2.1 which currently reads:

> A class can have a property which is the structural feature that implements the class.

with

> A class can have properties which characterize instances of the class.

#### Disposition: Resolved

# Disposition: Resolved

## OMG Issue No: 10859

### Title: Name as instance

**Source:**

NIST (Mr. Conrad Bock, conrad.bock@nist.gov conradb@cme.nist.gov)

**Summary:**

In Section 16.2.2 (Class and Property - Basics), paragraph underneath Table 16.5, the second sentence says a name can be an instance, but a name is usually a property or a string, not an instance. The third sentence says if name is the identifier, then "the remainder of the slots could be filled dynamically from other properties of the class". What does dynamically mean? It appears this is going into relational modeling, like the previous section does.

**Resolution:**

Delete paragraph underneath table 16.5.

**Revised Text:**

None

**Disposition:** **Resolved**

# Disposition: Resolved

## OMG Issue No: 10869

### Title: N-aries

#### Source:

NIST (Mr. Conrad Bock, conrad.bock@nist.gov conradb@cme.nist.gov)

#### Summary:

In 16.2.2 (Class and Property - Basics), second paragraph under Figure 16.3, association classes are not the same as naries. The translation given to N-ary associations is incomplete, because n-ary associations have multiplicities. These will not translate to cardinalities of binaries, at least not without a constraint to ensure there is only one instances of the association class in OWL for each link in UML.

#### Resolution:

Replace text as described below.

#### Revised Text:

The following is the current contents of this paragraph:

> This specification takes advantage of the fact that an N-ary relation among types $T_1$ ... $T_N$, or an association class with attributes, is formally equivalent to a set R of identifiers together with N projection functions $P_1$, ..., $P_N$, where $P_i$:R -> $T_i$. Thereby N-ary UML associations are translated to OWL classes with bundles of binary functional properties.

Replace this paragraph with the following:

> This specification takes advantage of the fact that both an N-ary relation among types $T_1$ ... $T_N$ and an association class with attributes are formally equivalent to a set R of identifiers together with N projection functions $P_1$, ..., $P_N$, where $P_i$:R -> $T_i$. Thereby both association classes and N-ary UML associations are translated to OWL classes with bundles of binary functional properties.

**Disposition: Resolved**

# Disposition: Resolved

## OMG Issue No: 10878

### Title: Restriction

#### Source:

NIST (Mr. Conrad Bock, conrad.bock@nist.gov conradb@cme.nist.gov)

#### Summary:

In 16.2.3 (More Advanced Concepts), ninth paragraph, first sentence should clarify that the restriction is recorded in the domain in the XML format, but is a restriction on the range. In particular, "relation" should be expanded to clarify that the restriction applies to each domain individual, not the relation as a whole (i.e., not all tuples).

#### Resolution:

Edit the text in the ninth paragraph of section 16.2.3 as described below.

#### Revised Text:

Replace the first sentence which currently reads:

> In OWL, a property when applied to a class can be constrained by cardinality restrictions on the domain giving the minimum (**minCardinality**) and maximum (**maxCardinality**) number of instances which can participate in the relation.

with

> In OWL, a property restriction applied to a class can impose a cardinality constraint giving the minimum (**minCardinality**), maximum (**maxCardinality**), or exact number of instances that can participate in a relation (of the specified type) with an instance of that class.

#### Disposition: Resolved

# Disposition: Resolved

# OMG Issue No: 10880 & 10882

## Title: Constants

### Source:

NIST (Mr. Conrad Bock, conrad.bock@nist.gov conradb@cme.nist.gov)

### Summary 10880:

In 16.2.3 (More Advanced Concepts), under the XML example, paragraph starting "It is not required", the same is true in UML.

### Summary 10882:

In 16.2.3 (More Advanced Concepts), under the XML example, the paragraph starting "Note that a consequence of", seems to have lost its context. It doesn't appear related to the paragraphs around it. If it is, this should be clarified.

### Resolution:

Replace and append text as described below.

### Revised Text:

The following is the current contents of the first paragraph:

> It is not required in OWL that there be a constant C such that X = C. All horses have color, but we may not know what color a particular horse has.

The current document then has two more paragraphs (not shown) followed by the following paragraph:

> Note that a consequence of this possible indeterminacy, it may not be possible to compute a transitive closure for a property across several ontologies, even if they share individuals

Replace the above paragraphs with the following:

> UML and OWL do not require that there is a constant C such that X = C, e.g., all horses have color, but we may not know what color a particular horse has. As consequence of this possible indeterminacy, it may not be possible to compute a transitive closure for a property across several ontologies, even if they share individuals.

### Disposition: Resolved

# Disposition: Resolved

# OMG Issue No: 10881

## Title: Herbrand Semantics

### Source:

NIST (Mr. Conrad Bock, conrad.bock@nist.gov conradb@cme.nist.gov)

### Summary:

Herbrand semantics. In 16.2.3 (More Advanced Concepts), under the XML example, the paragraph starting "In UML, there is a strict separation" is incorrect. The M0 level of UML can be real world individuals, not just software implementations (this is called an "analysis" application). Even when they are software implementations, they do not need to be specific ones, such as an SQL database manager. The last sentence is fine because of the qualification. The previous one makes it seem like the qualification is always the case. The entire next paragraph seems to also to omit the qualification, and I think can be dropped, since the presence of particular kinds of nulls in databases not relate to UML as generally applied. The last sentence of that paragraph can be used as a summary of the discussion.

### Resolution:

Replace text as described below.

### Revised Text:

The following is the current contents of these two paragraphs:

> In UML, there is a strict separation between the M1 and M0 levels. At the M1 level, that an association is mandatory (minimum cardinality greater than 0) is exactly the predicate [M]. Any difference between UML and OWL must come from the treatment of the model of the M1 theory at the M0 level. In practice, M0 models in UML applications tend to be ground Herbrand models implemented by something like an SQL database manager. For these cases, if we know a horse has a color, then we know what color it has. To the extent that UML tools and modeling build this expectation into products, conflict can occur when interoperating with an OWL ontology.
>
> But UML does not mandate M0 models to be Herbrand models. In particular SQL-92 supports the Null value construct, which has multiple interpretations, including “value exists but is not known.” Some years ago, CJ Date proposed a zoo of nulls with specific meanings, including “value exists but is not known,” and there have been proposals by Ray Reiter and others for databases with either existentially quantified variables in the data or which reason with the M1 theory for existentially quantified queries. It is possible for a particular application to introduce a special constant “unknown” into a class, which is treated specially by the programs. UML does not forbid an implementation of a class model in one of these ways. So there is no difference in principle between UML and OWL for properties which are declared to have minCardinality greater than 0 (and maxCardinality >= minCardinality) for a class.

Replace these two paragraphs with the following:

In UML, there is a strict separation between the M1 and M0 levels. If an association is mandatory (minimum cardinality greater than 0) it exactly matches the predicate [M]. Any difference between UML and OWL must come from the treatment of the model of the M1 theory at the M0 level. In practice, implementations derived from UML models tend to be ground Herbrand models implemented by something like an SQL database manager. For these cases, if we know a horse has a color, then we know what color it has. To the extent that UML tools and modeling build this expectation into products, conflict can occur when interoperating with an OWL ontology.

But UML does not mandate Herbrand compliance. It is possible for a particular application to introduce a special constant "unknown" into a class, and define special treatment by the programs. UML does not forbid an implementation of a class model in one of these ways. There is no difference in principle between UML and OWL for properties which are declared to have minCardinality greater than 0 (and maxCardinality >= minCardinality) for a class.

**Disposition: Resolved**

# Disposition: Resolved

## OMG Issue No: 10883

### Title: All / SomeValuesFrom

#### Source:

NIST (Mr. Conrad Bock, conrad.bock@nist.gov conradb@cme.nist.gov)

#### Summary:

In 16.2.3 (More Advanced Concepts), under the XML example, the paragraph starting "An OWL property can have", the translation to UML for allValuesFrom restrictions is property subsetting. There is no translation for someValuesFrom unless using the UML Profile for OWL.

#### Resolution:

Replace text as described below.

#### Revised Text:

The following is the current contents of this paragraph:

> An OWL property can have its range restricted when applied to a particular class, either that the range is limited to a class (subclass of *range* if declared) (**allValuesFrom**) or that the range must intersect a class (**someValuesFrom**). UML permits these and other restrictions using the facilities **specializes** or **refines**.

Replace this paragraph with the following:

> An OWL property can have its range restricted when applied to a particular class such that the range is limited (subtype of the property's *range* if declared) (**allValuesFrom**). UML permits these and other restrictions using the facilities: **subsets**, **specializes** or **redefines**. Often the specific restriction is defined in OCL.

#### Disposition: Resolved

# Disposition: Resolved

# OMG Issue No: 10896

## Title: Behavioral Features

### Source:

NIST (Mr. Conrad Bock, conrad.bock@nist.gov conradb@cme.nist.gov)

### Summary:

In Secton 16.6.1 (Behavioral Features), first paragraph is about a number of things other than behavioral features, and much of it is incorrect or uses incorrect terminology. Behavioral features only declare capabilities or services, not resources. They aren't the "program" that implements the service (called Behavior in UML). Behavioral features can be used in OCL that defines a derivation of a property, but the behavioral feature isn't directly related to the derived property. Operations include the parameters (including return value). A "method" in UML is the behavior that implements the operation on a particular class. Responsibility in UML is only a standard stereotype of a usage dependency. It isn't a well-developed part of UML class modeling. Qualified associations are more accurately described as a special kind of ternary relation. An abstract can have operations and methods like any other class. Abstract classes cannot have direct instances. Interfaces specify features of classes, including operation features. They aren't interfaces of operations themselves.

### Resolution:

Replace text as described below.

### Revised Text:

The following is the current contents of this paragraph:

> UML allows the specification of behavioral features, which declare capabilities or resources. One use of behavioral features is to calculate property values. Behavioral features can be used in the OCL that derives properties. Facilities of UML supporting programs include **operations**, which describe the parameters of methods; **static operations**, which are operations attached to a class like static attributes; **interface classes**, which specify among other things operation features; **qualified associations**, which are a special kind of ternary relation; and **active classes**, which are classes each instance of which controls its own thread of execution control.

Replace this paragraph with the following:

> UML allows the specification of behavioral features, which declare capabilities and dynamic aspects of the system. OCL can be used to restrict derived properties. Facilities of UML that can be used to describe application programs include **operations**, which describe the parameters of methods; **static operations**, which have a shared implementation for all subclasses; **interface classes**, which specify an interface to a set of attributes and operations that could be implemented by one or more classes; and **active classes**, which are classes that have a separate thread of execution control for each instance.

**Disposition:** **Resolved**

# Disposition: Resolved

# OMG Issue No: 10903

## Title: Symmetric

### Source:

NIST (Mr. Conrad Bock, conrad.bock@nist.gov conradb@cme.nist.gov)

### Summary:

Section 16.3.5 (Binary Association To Object Property), second paragraph says that binary associations with the same type on both ends translate to symmetric properties in OWL. This isn't correct. For example, an association that has Animal on both ends, with ends named "chases" and "chased by", doesn't mean that if animal A chases animal B, that animal B chases animal A. It means that animal B is chased by animal A.

### Resolution:

Replace text as described below.

### Revised Text:

The following is the current contents of the first and second paragraphs in 16.3.5:

> An association specifies a semantic relationship that can occur between typed instances. It has at least two ends represented by properties, each of which is connected to the type of the end. More than one end of the association may have the same type. In this section, only binary association is discussed. In Section 16.3.4, instances of OWLObjectProperty have been created. However, the possible OWLinverseOf relationship between two navigableOwnedEnd of an association has not been created. AssociationToObjectProperty relation is used to set OWLinverseOf relationships among related properties.
>
> Further, associations both of whose ends are properties with the same type will be mapped to symmetric properties in OWL.

Replace these paragraphs with the following:

> A binary association specifies a relationship that can occur between typed instances. It has exactly two ends represented by properties, each of which is connected to the type of the end. The AssociationToObjectProperty relation is used to set OWLinverseOf relationships between inverse properties.

### Disposition: Resolved

# Disposition: Resolved

## OMG Issue No: 12386

### Title: RDFSDatatype uriRef Property Multiplicity Issue

**Source:**

**Sandpiper Software, Inc. (Mrs. Elisa F. Kendall, ekendall@sandsoft.com)**

**Summary:**

The uriRef property for RDFSDatatype is defined in the specification as a LiteralString [0..*], which is incorrect. The uriRef property is required to have value, and thus the multiplicity should be [1..*]

**Resolution:**

Revise multiplicity as suggested and remove first constraint.

**Revised Text:**

Currently, the text in section 14.1.6.2 that describes properties and constraints for RDFSDatatype reads as follows:

**Properties**

• uriRef: LiteralString [0..*] – the URI reference(s) associated with the datatype.

**Constraints**

[1] A class stereotyped by «rdfsDatatype» must have a value for the uriRef property.

[2] The value of the uriRef property must be a UML::LiteralString that is stereotyped by «uriReference».

[3] For built-in datatypes (i.e., those that are not user-defined), the string value of the uriRef must be that of an XML Schema Datatype as defined in [XML Schema Datatypes], and as given in Annex A.

[4] (Semantic) The value of the uriRef property must link the «rdfsDatatype» to either an XML Schema Datatype or user-defined type corresponding to a datatype definition of the appropriate type.

Revise it to read:

**Properties**

• uriRef: LiteralString [1..*] – the URI reference(s) associated with the datatype.

**Constraints**

[1] The value of the uriRef property must be a UML::LiteralString that is stereotyped by «uriReference».

[2] For built-in datatypes (i.e., those that are not user-defined), the string value of the uriRef must be that of an XML Schema Datatype as defined in [XML Schema Datatypes], and as given in Annex A.

[3] (Semantic) The value of the uriRef property must link the «rdfsDatatype» to either an XML Schema Datatype or user-defined type corresponding to a datatype definition of the appropriate type.

**Disposition: Resolved**

# Disposition: Resolved

## OMG Issue No: 12387

### Title: RDFSisDefinedBy text issue

**Source:**

Sandpiper Software, Inc. (Mrs. Elisa F. Kendall, ekendall@sandsoft.com)

**Summary:**

The text of the Constraints section for RDFSisDefinedBy requires revision (per ODM FTF2 meeting minutes of Feb 6, 2008): delete (a) "i.e., the classifier owning the dependency", (b) "i.e., the type of the dependency", and (c) the end of the last sentence, starting with ", or, at a minimum ..." to the end

**Resolution:**

Revise text as suggested.

**Revised Text:**

Currently, the text in section 14.1.6.3 RDFSisDefinedBy describing Constraints reads as follows:

[1] (Semantic) The «rdfsIsDefinedBy» stereotype is used to state that a particular resource (the subject of the RDF statement, i.e., the classifier owning the dependency) is defined by another resource (the object of the RDF statement, i.e., the type of the dependency). In theory, this stereotype can be applied to a dependency between any two “generic” resources, but in practice, we recommend that it is applied to a UML::Dependency that links two UML::InstanceSpecifications that are stereotyped by «rdfsResource», or, at a minimum, that the type of the dependency should be a UML::InstanceSpecifications stereotyped by «rdfsResource».

Revise it to read:

[1] (Semantic) The «rdfsIsDefinedBy» stereotype is used to state that a particular resource (the subject of the RDF statement) is defined by another resource (the object of the RDF statement). In theory, this stereotype can be applied to a dependency between any two “generic” resources, but in practice, we recommend that it is applied to a UML::Dependency that links two UML::InstanceSpecifications that are stereotyped by «rdfsResource».

**Disposition:** **Resolved**

# Disposition: Resolved

## OMG Issue No: 12388

### Title: RDFSseeAlso text issue

**Source:**

**Sandpiper Software, Inc. (Mrs. Elisa F. Kendall, ekendall@sandsoft.com)**

**Summary:**

The text of the Constraints section for RDFSseeAlso requires revision (per ODM FTF2 meeting minutes of Feb 6, 2008): delete (a) "i.e., the classifier owning the dependency", (b) "i.e., the type of the dependency", and (c) the end of the last sentence, starting with ", or, at a minimum ..." to the end

**Resolution:**

Revise text as suggested.

**Revised Text:**

Currently, the text of section 14.1.6.4 RDFSseeAlso describing Constraints reads as follows:

[1] (Semantic) The «rdfsSeeAlso» stereotype is used to state that additional information about a particular resource (the subject of the RDF statement, i.e., the classifier owning the dependency) is given by another resource (the object of the RDF statement, i.e., the type of the dependency). As with «rdfsIsDefinedBy», this stereotype can be applied to a dependency between any two “generic” resources, but in practice, we recommend that it is applied to a UML::Dependency that links two UML::InstanceSpecifications that are stereotyped by «rdfsResource», or, at a minimum, that the type of the dependency should be a UML::InstanceSpecifications stereotyped by «rdfsResource».

Revise it to read:

[1] (Semantic) The «rdfsSeeAlso» stereotype is used to state that additional information about a particular resource (the subject of the RDF statement) is given by another resource (the object of the RDF statement). As with «rdfsIsDefinedBy», this stereotype can be applied to a dependency between any two “generic” resources, but in practice, we recommend that it is applied to a UML::Dependency that links two UML::InstanceSpecifications that are stereotyped by «rdfsResource».

**Disposition: Resolved**

# Disposition: Resolved

## OMG Issue No: 12389

### Title: RDFType text issue

**Source:**

**Sandpiper Software, Inc. (Mrs. Elisa F. Kendall, ekendall@sandsoft.com)**

**Summary:**

The text describing RDFType in the UML Profile for RDF requires revision (per ODM FTF2 meeting minutes of Feb 6, 2008): "This can be represented in a UML model by the relation between an instance specification and its classifiers" (fix second sentence)

**Resolution:**

Revise text as suggested.

**Revised Text:**

Currently, the Description in section 14.1.6.6 for RDFType reads as follows:

rdf:type maps to the relation between instance and classifier in UML. This is equivalent in UML to the relation from an element in a model to an element in the UML metamodel, or between an instance specification and its classifiers. Note that resources in RDF can be multiply classified. No stereotype is needed.

Revise it to read:

rdf:type maps to the relation between instance and classifier in UML. This can be represented in a UML model by the relation between an instance specification and its classifiers. For M1, rdf:type maps to the classifier association between an instanceSpecification and classifiers in UML. Note that resources in RDF can be multiply classified. No stereotype is needed.

**Disposition:** **Resolved**

# Disposition: Resolved

## OMG Issue No: 12391

### Title: OWL Annotation properties are impoverished in the OWL profile

**Source:**

**Sandpiper Software, Inc. (Mrs. Elisa F. Kendall, ekendall@sandsoft.com)**

**Summary:**

(Per ODM FTF2 meeting minutes of February 20), OWLAnnotationProperty is defined in the profile as having a base class of UML::Property, whereas RDF properties have base classes including UML::Property and UML::AssociationClass. Add AssociationClass to the set of base classes for the <<owlAnnotation>> stereotype.

**Resolution:**

In section 14.2.3.1, revise the Stereotype and Base Class for OWLAnnotationProperty as suggested.

**Revised Text:**

Currently, the Stereotype and Base Class for OWLAnnotationProperty is defined as:

«owlAnnotation» with base class of UML::Property

Revise it to be:

«owlAnnotation» with base class of UML::AssociationClass and UML::Property

**Disposition:** **Resolved**

# Disposition: Resolved

## OMG Issue No: 12392

### Title: URIReferenceNode definition is misplaced in the specification

**Source:**

**Sandpiper Software, Inc. (Mrs. Elisa F. Kendall, ekendall@sandsoft.com)**

**Summary:**

Section 14.1.4.1 URIReferenceNode should be moved to 14.1.3.7 -- it is "buried" under section 14.1.4 ReificationKind for some unknown reason.

**Resolution:**

Renumber paragraph 14.1.4.1 to 14.1.3.7 and move it to the end of section 14.1.3, where it belongs.

**Revised Text:**

None.

**Disposition:** **Resolved**

# Disposition: Resolved

## OMG Issue No: 12393

### Title: URIReferenceNode constraint text

**Source:**

Sandpiper Software, Inc. (Mrs. Elisa F. Kendall, ekendall@sandsoft.com)

**Summary:**

Modify the text for the first constraint in section 14.1.4.1 URIReferenceNode from "The uriRef:String property, inherited from ..." to "The uriRef property, inherited from ..." to remove the superfluous and incorrect type information.

**Resolution:**

Revise text as suggested.

**Revised Text:**

Currently, the text for the constraint in question reads:

[1] The uriRef: String property, inherited from «rdfsResource», must have a value.

Revise it to be as follows:

[1] The uriRef property, inherited from «rdfsResource», must have a value.

**Disposition:** **Resolved**

# Disposition: Resolved

## OMG Issue No: 12395

### Title: OWL Cardinality Constraints text revision

**Source:**

**Sandpiper Software, Inc. (Mrs. Elisa F. Kendall, ekendall@sandsoft.com)**

**Summary:**

From the ODM FTF2 meeting minutes of February 20th: Cardinality constraints -- the second paragraph of the description states: "Additionally, isOrdered = false on OWL properties, and isUnique = true on owl properties...". These "constraints" are not specified in the owl properties section (14.2.6) - should they apply to all owl properties, or only to cardinality constraints? According to Conrad, this text should apply to all RDF and OWL properties, but it comes from the text on multiplicities in UML, which is why it landed in this section of the document. Note that it does occur under the rdfProperty section of the profile specification, so it does apply to all properties ... thus we might remove the "OWL" from the sentence for further clarification.

**Resolution:**

In section 14.2.5.4, clarify the text for OWL cardinality constraints as suggested.

**Revised Text:**

Currently, the second paragraph under the 14.2.5.4 heading reads as follows:

Value specifications for multiplicities in OWL must be non-negative integer literals. Additionally, isOrdered = false on OWL properties, and isUnique = true on OWL properties, meaning that the values are a set, not a bag.

Revise the second sentence to be:

Additionally, isOrdered = false and isUnique = true on all RDF and OWL properties, meaning that the values are a set, not a bag.

**Disposition:** **Resolved**

# Disposition: Resolved

## OMG Issue No: 12396

### Title: OWL Properties have conflicting definitions

**Source:**

**Sandpiper Software, Inc. (Mrs. Elisa F. Kendall, ekendall@sandsoft.com)**

**Summary:**

From the ODM FTF2 meeting, Burlingame meeting: In 14.2.6.1 owl:DatatypeProperty specifies base class of AssociationClass AND Property. 14.2.6.2 and 14.2.6.3 specify a base class of AssociationClass OR Property. What is it, AND or OR? The text for section 14.2.6.2 owl:ObjectProperty and 14.2.6.3 owl:Property, defining the base class and stereotype is wrong - it should be AND.

**Resolution:**

Correct the Stereotype and Base Class text for owl:ObjectProperty (14.2.6.2) and owl:Property (14.2.6.3) as suggested.

**Revised Text:**

1. Currently, the Stereotype and Base Class text for owl:ObjectProperty (14.2.6.2) are defined as follows:

«objectProperty» with base class of UML::AssociationClass or UML::Property.

Revise the definition to be:

«objectProperty» with base class of UML::AssociationClass and UML::Property.

2. Currently, the Stereotype and Base Class text for owl:Property (14.2.6.3) are defined as follows:

«owlProperty» with base class of UML::AssociationClass or UML::Property.

Revise the definition to be:

«owlProperty» with base class of UML::AssociationClass and UML::Property.

**Disposition: Resolved**

# Disposition: Resolved

## OMG Issue No: 12397

### Title: No stereotype for OWL individuals

**Source:**

**Sandpiper Software, Inc. (Mrs. Elisa F. Kendall, ekendall@sandsoft.com)**

**Summary:**

From the ODM FTF2 meeting minutes, February 20th: There is currently no stereotype available for individuals in the OWL Profile. Participants agreed that we would like to have a stereotype <<owlIndividual>> on UML instance specifications for individual, and would want an additional constraint saying that such an instance specification would only have one value, i.e., that it specifies a single individual; also, what if we have a singleton, but not the corresponding individual? We would still translate the singleton class to an OWL individual in this case ... see issue 10908 (related issue in the mapping chapter).

**Resolution:**

Add a stereotyped instance specification for <<owlIndividual>> and the corresponding constraint, as suggested.

**Revised Text:**

Currently, the Stereotype and Base Class text for section 14.2.7.1 **Class Membership and Property Values of Individuals** reads as follows:

No stereotype, use UML::InstanceSpecification typed by a class having the properties desired for the individual. The class may be stereotyped by «singleton» to indicate it is for a specific individual[5]. Classes stereotyped by «singleton» are not translated to OWL, and their properties appear in OWL as properties of the individual.

1. Revise the first sentence to be:

«owlIndividual» with a base class of UML::InstanceSpecification, typed by a class having the properties desired for the individual.

2. Add the following constraint:

[2] The instance specification stereotyped by «owlIndividual» has only one value, i.e., it specifies a single individual.

**Disposition: Resolved**

# Disposition: Resolved

## OMG Issue No: 12398

**Title: Definition of RDF Property and its use in the figures is inconsistent**

**Source:**

**Sandpiper Software, Inc. (Mrs. Elisa F. Kendall, ekendall@sandsoft.com)**

**Summary:**

From email dated 3/12/2008 from SRI, and as discussed (and documented in the minutes from the ODM FTF2 F2F DC meeting: Section 14.1.7, In Figure 4.3 the line is an "association". However, the specified base classes are "AssociationClass" and "Property". To duplicate this example in Enterprise Architect, it was necessary to define a profile in which the rdfProperty stereotype extends the "Association" metaclass.

**Resolution:**

This issue and ensuing discussions have led the FTF working group to recognize the need for an additional way to represent an RDF property in UML, that is, reified as a UML::Class, in addition to adding UML::Association to the set of base classes available for modeling RDF properties in the profile.

Tool implementation of AssociationClass (tested in IBM Rational Software Architect, No Magic MagicDraw, and Sparx Enterprise Architect) is unsatisfactory from an RDF/OWL perspective, as one cannot draw free-standing RDF properties (as association classes with an «rdfProperty» stereotype applied) without dragging the default relationships to rdfs:Resource or owl:Thing onto a diagram). In many cases when defining RDF vocabularies, it is desirable to define a property on its own, without specifying any domain or range, but sometimes showing property inheritance. Downstream vocabularies may then further refine these definitions as appropriate.

The proposed resolution is to

- add UML::Association as a base class for all cases where RDF properties are used in the RDF and OWL profiles
- add UML::Class as a base class for all cases where RDF properties are used in the RDF and OWL profiles

- introduce two new stereotypes, «rdfsDomain» and «rdfsRange», to support domain and range property restrictions when UML classes are used to represent RDF properties,

**UML Profile for RDF Revised Text:**

1. Section 14.1.7.1 RDFProperty, Description

Currently,the second paragraph in the description section states:

"Properties in RDF and OWL are defined globally, that is, they are available to all classes in all ontologies – not only to classes in the ontology they are defined in, but to classes in ontologies that are imported. For RDF properties that are defined without specifying a domain or range, the profile uses an anonymous class (analogous to owl:Thing in OWL ontologies) for the "missing" end class."

Revise this section to read:

Properties in RDF and OWL are defined globally, that is, they are available to all classes in all vocabularies and ontologies – not only to classes in the vocabulary or ontology they are defined in, but to classes in other vocabularies and ontologies, including those that are imported. For RDF properties that are defined without specifying a domain or range, the profile uses an anonymous class (a singleton class representing rdfs:Resource) for the "missing" end class, as needed.

2. Section 14.1.7.1 RDFProperty, Stereotype and Base Class

Modify this section to be:

**«rdfProperty» with base class of UML::Class, UML::AssociationClass, UML::Property, and UML::Association**

3. Section 14.1.7.1 RDFProperty, Graphical Notation

(1) At the top of this section, insert the following:

Properties without a specified domain or range may be represented using a stereotyped class, as shown in Figure 14.x


.

**Figure 14.x Property `hasColor` – Class Notation Without Specified Domain or Range**

(2) Revise the text above Figure 14.4 from:

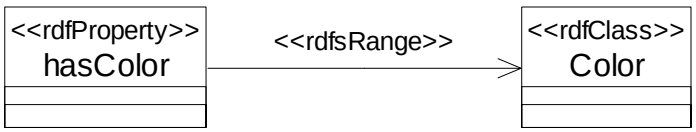
Associations can be classes, as shown in Figure 14.4:

to

RDF properties may be represented as AssociationClasses, as shown in Figure 14.4:

(3) Modify the caption under Figure 14.4, to add "Without Specified Domain," after hasColor.

(4) Insert the following between the paragraph following Figure 14.4 and section B:




with the caption:

**Figure 14.x Property `hasColor` Without Specified Domain, Class Notation**

Insert the following text below the caption for Figure 14.x:

A stereotype indicating that the association between the hasColor property and Color class representing the RDF range restriction is introduced here, defined in paragraph 14.1.7.4.

(5) Modify the text in C, below Figure 14.5 from:

C. Properties with a defined range have the range class as their type in UML. Properties with no range have an anonymous class as their type in UML, as shown in Figure 14.5.

to

C. Properties with a defined range have the range class as their type in UML. Properties with no range may have an anonymous class as their type in UML, as shown in Figure 14.5.

(8) Insert the text and figure below in E, below Figure 14.5, and renumber the paragraph currently defined as 'E' to 'F':

E. Properties with a defined domain and range may also be specified using the class notation as shown in Figure 14.xx.

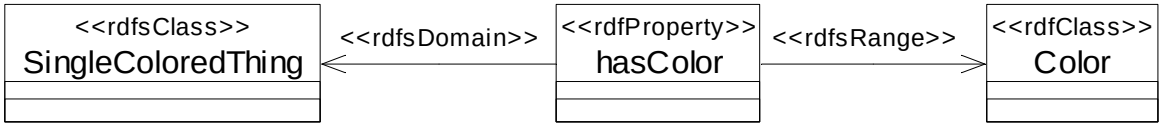



**Figure 14.x Property `hasColor` With Specified Domain and Range, Class Notation**

The «rdfsDomain» stereotype introduced above is defined in section 14.1.7.3, below.

(6) Revise the text in E (renumbered 'F', as indicated above), below Figure 14.5, from:

Two ways of representing RDF/S and OWL property subtyping (*i.e.*, rdfs:subPropertyOf) use UML property/unidirectional association subsetting or association class subtyping. The UML semantics for both is that all links (instances, tuples) of the subtype properties or associations are links (instances, tuples) of all the supertypes properties or associations.

to

The notation for RDF/S and OWL property subtyping (*i.e.*, rdfs:subPropertyOf) uses UML property/unidirectional association subsetting, association class subtyping, or generalization/specialization.

(7) Insert the following text and figure below Figure 14.8:

Figure 14.xxx provides another example of RDFS property subtyping, in this case using UML classes to represent the relevant RDF properties.

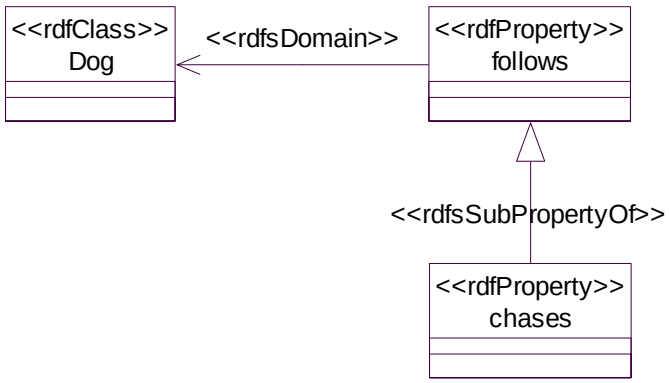


**Figure 14.xxx Property Subsetting – Class Notation**

4. Revise section 14.1.7.2 RDFGlobalProperty, Description, from:

An optional stereotype on a unidirectional association class or property with «rdfProperty» applied, indicating the association/property is defined globally, i.e., that class having the property, or on the non-navigable end of the association, is the class on which the property/association is introduced, i.e., the class does not inherit the property or association from a superclass.

to

An optional stereotype on a unidirectional association class, class, or property with «rdfProperty» applied, indicating the association/class property is defined globally, i.e., that the class having the property, or the non-navigable end of the association, is the class on which the class property/association is introduced, i.e., the property is not inherited from a superclass.

5. Revise section 14.1.7.2 RDFGlobalProperty, Stereotype and Base Class, to:

**«rdfGlobal» with base class of UML::Class, UML::AssociationClass, UML::Property, and UML::Association**

6. Insert new section 14.1.7.3 RDFSDomain, as follows:

### Description

`rdfs:domain` indicates that RDF resources denoted by the subjects of triples whose predicate is a given property P are instances of the class denoted by the domain. When a property has multiple domains, then the resources denoted by the subjects of triples whose predicate is P are instances of *all of the classes* stated by the `rdfs:domain` properties (In OWL, this is typically thought of as an anonymous class representing the intersection of all of the classes said to be in the domain of the property).

### Stereotype and Base Class

**«rdfsDomain» with base class of UML::Association**

### Parent

None.

### Properties

None.

### Constraints

[1] Applies only to associations between a class with an «rdfsClass» stereotype applied (or any of its children, *e.g.*, «owlClass»), and a class with «rdfProperty» applied.

[2] Associations with «rdfsDomain» applied are binary.

[3] Associations with «rdfsDomain» applied have unidirectional navigation (from the class with the «rdfProperty» stereotype applied to the class with the «rdfsClass» stereotype applied)

### Graphical Notation


**Figure 14.x «rdfsDomain» Stereotype Notation – Class Notation for RDF Property**

7. Insert new section 14.1.7.4 RDFSRange, as follows:

### Description

`rdfs:range` indicates that the values of a given property P are instances of the class denoted by the range. When a property has multiple `rdfs:range` properties, then the resources denoted by the objects of triples whose predicate is P are instances of *all of the classes* stated by the `rdfs:range` properties (In OWL, this is typically thought of as an anonymous class representing the intersection of all of the range classes).

### Stereotype and Base Class

**«rdfsRange» with base class of UML::Association**

### Parent

None.

### Properties

None.

### Constraints

[1] Applies only to associations between a class with an «rdfsClass» stereotype applied (or any of its children, *e.g.*, «owlClass»), and a class with «rdfProperty» applied.

[2] Associations with «rdfsRange» applied are binary.

[3] Associations with «rdfsRange» applied have unidirectional navigation (from the class with the «rdfProperty» stereotype applied to the class with the «rdfsClass» stereotype applied)

### Graphical Notation

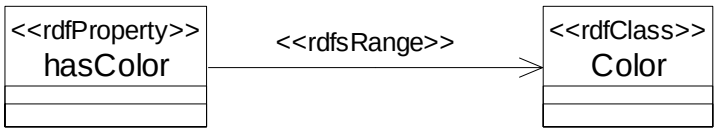



**Figure 14.x «rdfsRange» Stereotype Notation – Class Notation for RDF Property**

**UML Profile for OWL Revised Text:**

1. In section 14.2.3.1, OWL Annotation Property, revise the Stereotype and Base Class to be:

**«owlAnnotation» with base class of UML::Class, UML::AssociationClass, UML::Property, and UML::Association**

Note that this correction supercedes the resolution defined for issue 12391.

2. In section 14.2.5.4, Cardinality Constraints, under Graphical Notation:

Insert the following text and figure after Figure 14.16:

When class notation for properties is desirable, cardinality constraints can be represented as shown in Figure 14.y.

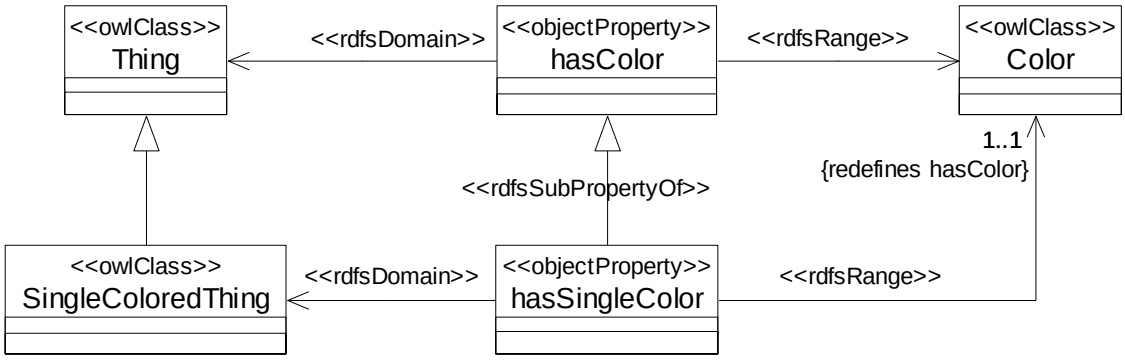



**Figure 14.y owl:Cardinality – Restricted Multiplicity in Subtype**

3. In section 14.2.5.5, owl:allValuesFrom Constraint:

Insert the following after Figure 4.19:

In addition to the «rdfsSubPropertyOf» stereotype, in order to facilitate disambiguation, vendors may optionally apply the «owlValue» stereotype (defined in paragraph 14.2.5.6, below), to the association redefining the property (*i.e.*, the association representing the restriction), by setting the allValuesFrom property to the class filling the value restriction with a multiplicity of 1.

In cases where UML classes are used to represent OWL properties, the option shown in Figure 14.z, may be used. Again, vendors may optionally apply the «owlValue» stereotype (defined in paragraph 14.2.5.6, below), to the association redefining the property (*i.e.*, the association representing the restriction).

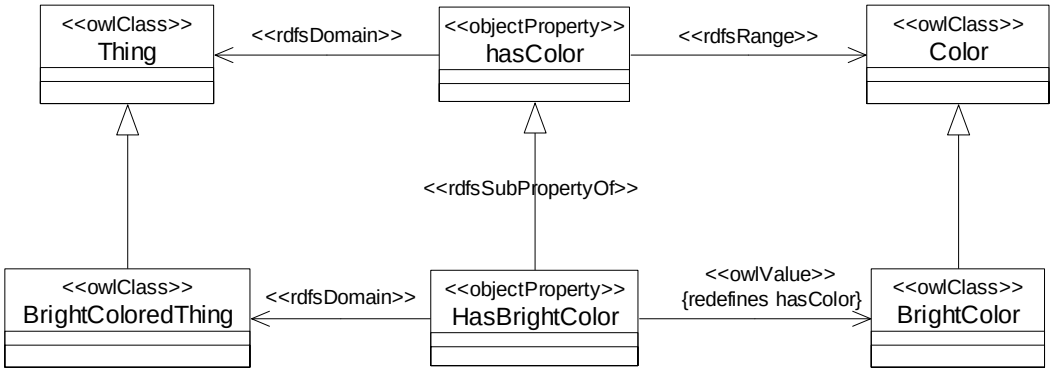



**Figure 14.z Property Redefinition for owl:allValuesFrom Using Classes**

4. Section 14.2.5.6, owl:someValuesFrom and owl:hasValue Constraints

(1) In the Properties section, add the following property:

- allValuesFrom: Class [0..1] – identifies a class stereotyped by «owlClass» or «owlDataRange»

(2) Add the following constraints to the Constraints section:

[3] The value of the allValuesFrom property must be stereotyped «owlClass» or «owlDataRange».

(3) Under the Graphical Notation, delete the text and Figure 14.20, replacing it with the following:

The graphical notation for `owl:someValuesFrom` is the same as the notation given in section 14.2.5.5 for `owl:allValuesFrom`, with the distinction being the choice of property selected (someValuesFrom in this case). For `owl:hasValue`, one option using the class notation is shown in Figure 14.zz, below. Again in this case, use of the «owlValue» stereotype, and property hasValue whose value would be the instance specification that is a member of the singleton class, Red, is optional.

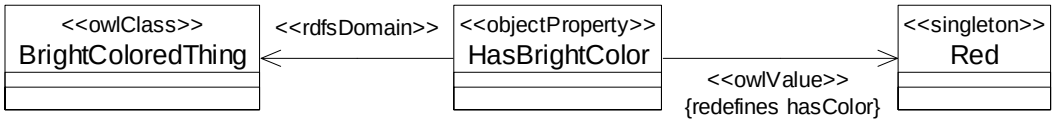



**Figure 14.zz Property Redefinition for owl:hasValue Using Classes**

5. Section 14.2.6.1, owl:DatatypeProperty

Modify the Stereotype and Base Class to be:

**«datatypeProperty» with base class of UML::Class, UML::AssociationClass, UML::Property, and UML::Association**

6. Section 14.2.6.2, owl:ObjectProperty

Modify the Stereotype and Base Class to be:

**«objectProperty» with base class of UML::Class, UML::AssociationClass, UML::Property, and UML::Association**

Note: this supercedes the resolution to issue 12396.

7. Section 14.2.6.3, owl:Property

Modify the Stereotype and Base Class to be:

**«owlProperty» with base class of UML::Class, UML::AssociationClass, UML::Property, and UML::Association**

Note: this supercedes the resolution to issue 12396.

**Disposition: Resolved**

# Disposition: Resolved

## OMG Issue No: 12793

**Title: Design of RDF metamodel for rdf:Statement, triple, and graph controversial in semantic web community**

**Source:**

**National Institute of Standards and Technology (Evan Wallace, ewallace@nist.gov)**

**Summary:**

In November-December 2007, there were discussions of the ODM metamodels for RDF and OWL within the OWL working group of the W3C. It became clear from these discussions that key OWL and RDF experts were surprised and not particularly happy with how ODM modeled the RDF data model and reification vocabulary. These portions of the RDF specification describe the fundamental data model for RDF assertions: Triple with subject, predicate and object properties; Graphs for collecting triples as sets of assertions; and a reification vocabulary enabling assertions about triples themselves. Pragmatic design decisions were made for the ODM metamodel which merged support for triples and the reification vocabulary into a single class, Statement, and merged support for a non-standard extension for RDF, named graphs, with graph. Unfortunately, the reification vocabulary for RDF has proved problematic and controversial, and because these aspects are key to the semantics of RDF, some are very sensitive about how they are modeled.

To encourage better acceptance of ODM in the semantic web community the RDF metamodel should be changed to correspond with expectations of SemWeb experts. Triples, Statements, Graphs and Named Graphs should all be modeled with separate constructs with non-normative and non-standard elements noted. The OWL Ontology model which uses these constructs should be modified to use the fundamental rdf forms: triple and graph, and should do this in a way consistent with the RDF specifications, e.g., RDF triples in a graph are considered unordered (a set).

**Resolution:**

Factor the Statements diagram in Figure 10.2 in RDFBase into 3 diagrams: Graph Data Model, Reification, and Graphs. This separates rdf triples from statements about them and creates a separate construct for Named Graph since it is not a part of the current rdf specification. The diagram for rdf:statements is now called Reification (rdfs calls this the reification vocabulary), the Graph Data Model diagram depicts triples, and the Graphs diagram depicts RDF graphs and named graphs. New subsections in RDFBase will be added for RDF Reification and Graphs: Reification after the Literals section and Graphs after Reification.

Places in odm metamodels which extended or otherwise referred to statements are changed to refer to triples. This includes the Documents model in the RDFWeb package and the Ontology model in OWLBase.

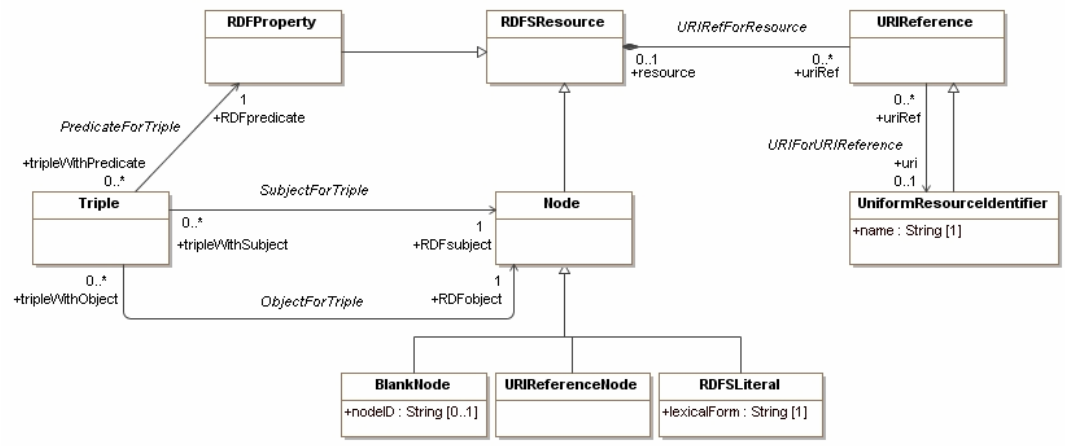
Changes include:

- Changing section 10.2 from describing RDF Statements to describing Triples.
- Revising figure 10.2 to describe triples, moving RDF Graph to 10.5, eliminating Reification kind, and introducing a supertype which is a complete and disjoint covering for URIReferenceNode, BlankNode, and RDFLiteral called Node for RDFSubject and RDFObject roles from Triple.
- Adding a new section 10.4 called RDF Statements to describe the RDF reification vocabulary. Including a diagram describing Statements and its relationship to Triples.
- Adding a new section 10.5 called Graphs describing rdf graphs and named triples. This includes a Graphs diagram depicting RDF Graphs and Named Graphs as separate classes and including associations on named graphs to for equivalentGraphs and subGraphs per the seminal named graphs reference.
- Revising Documents and Namespaces in the RDFWeb package to refer to triples instead of statements and revising the Documents diagram accordingly.
- Revising OWL Ontology section and diagram in the OWLBase package to refer to triples instead of statements and to eliminate {ordered} attribute for sets of triples.

**Revised Text:**

1. Revise current section 10.2 RDFBase Package, RDF Statements as follows:

Change the title of "10.2 RDFBase Package, RDF Statements" to "10.2 RDFBase Package, RDF Triples".
Replace current figure 10.2 – RDFBase Package, The Statements Diagram with RDFBase Package, The Graph Data Model (see below)




Change the text in first sentence of section 10.2 from "depicts the RDF base statements diagram" to "depicts the RDF base graph data model".
Replace the second paragraph with the text: "RDF provides a reification vocabulary for making statements about triples. This is described in section 10.4 RDFBase Package, RDF Statements."
Delete third paragraph.
Add new subsection 10.2.2 Node after BlankNode subsection. The Description section should begin "The subject and object of a Triple of of type Node. URIReferenceNode, BlankNode, and RDFSLiteral form a complete and disjoint covering of Node."
Move the constraints from RDFSResource to Node.
Add a Semantics section for Node which states: "This type represents the nodes in RDF Graphs."
Add a sentence into the Semantics section for RDFProperty which states: "This type represents the arc in RDF graphs."
Add a constraint for RDFSLiteral that a literal may not be an RDFobject or RDFpredicate.
Change the title of 10.2.6 RDFStatement to RDF Triple
In the Triple subsection do the following:

- Delete reification attribute bullet and replace with "None" unindented.
- In first association bullet change StatementForGraph to TripleForGraph and statement to triple.
- Move NameForReification association to new RDFStatement in Reification section.

- Change the text for SubjectForStatement association to read: “RDFsubject: Node [1] in association SubjectForTriple – links a triple to the node that is the subject of the triple.”
- Change the text for PredicateForStatement association to read: “RDFpredicate: RDFProperty [1] in association PredicateForTriple – links a triple to the property that is the predicate of the triple.”
- Change the text for ObjectForStatement association to read: “RDFobject: Node [1] in association SubjectForTriple – links a triple to the node that is the subject of the triple.”
- Move last association (specialize RDFSResource) to RDFStatement in Reification section.
- Slightly modify the first constraint to remove “resource” and the parenthesis around node.
- Remove the last sentence of Semantics section.

  Delete ReificationKind entirely since reified triples are now always represented by RDFStatements.

  Move RDF Graph to new Graphs section.

2. Add a new RDFBase, RDF Statements section (10.4) following the RDF literals section with text and figure as show below.

# 10.4 RDFBase Package, RDF Statements

RDFS provides a reification vocabulary with no formal semantics.

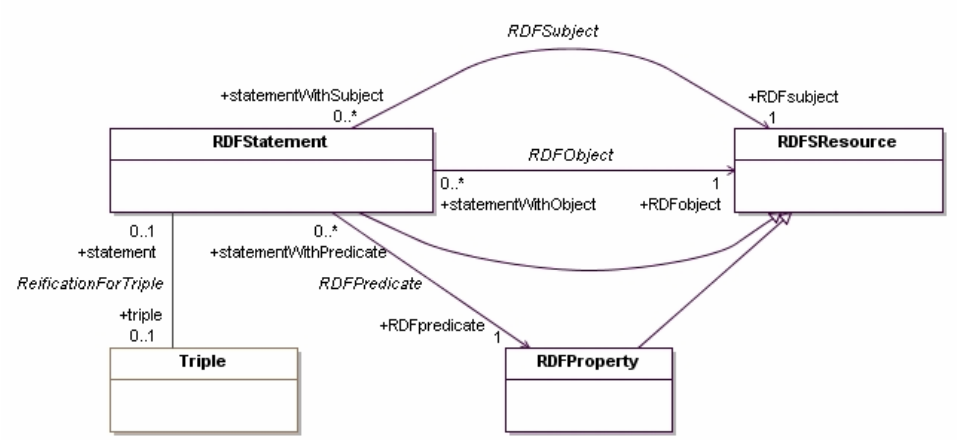



**Figure 10.? – RDFBase Package, The Reification Diagram**

## 10.4.1 RDFProperty (Augmented Definition)

### Associations

- statementWithPredicate:RDFStatement [1] in association RDFPredicate – links a statement to its predicate.

## 10.4.2 RDFResource (Augmented Definition)

### Associations

- statementWithObject:RDFStatement [0..*] in association RDFObject – a resource represents an object of zero or more statements
- statementWithSubject:RDFStatement [0..*] in association RDFSubject – a resource represents an subject of zero or more statements

## 10.4.3 RDFStatement

### Description

RDF Statement provides a way to make statements about triples or describe statements without asserting them.

### Attributes

None

### Associations

- RDFobject :RDFSResource [1] in association RDFObject – links a statement to the resource that is its object.
- RDFpredicate: RDFSProperty [1] in association RDFPredicate – links a statement to a property that is its predicate.
- RDFsubject: RDFSResource [1] in association RDFSubject – links a statement to a resource that is its subject.
- Triple: Triple [0..1] in association ReificationForTriple – links a statement to the triple it reifies, if such a triple exists.
- Specialize Class RDFSResource.

## 10.4.4 Triple (Augmented Definition)

### Associations

- statement:RDFStatement in association ReificationForTriple – links a triple with a statements that reifies it, if the triple is reified.

3. Add a new RDFBase Package, RDF Graphs section 10.5 using the text and figure below.

# 10.5 RDFBase Package, RDF Graphs

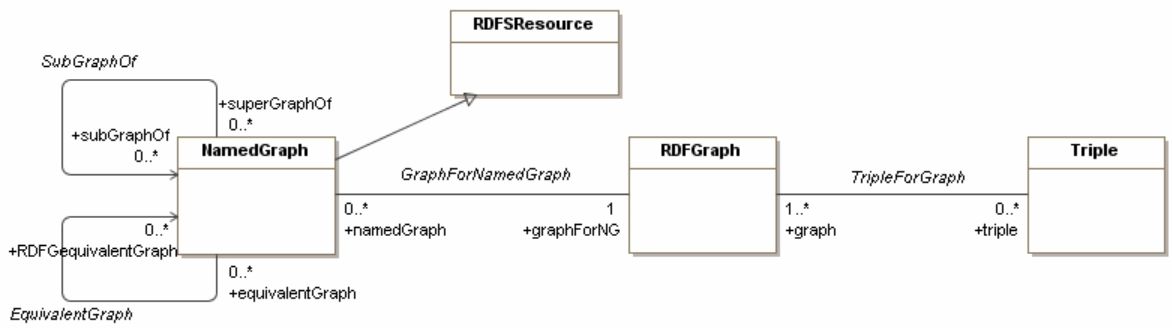



**Figure 10.? – RDFBase Package, The Graphs Diagram**

## 10.5.1 NamedGraph

### Description

A named graph is a uri reference and RDF graph pair. It effectively provides a way to name an RDF graph and thus refer to the graph in a graph.

At the time of this writing, NamedGraphs are not a part of RDF, but have been proposed as a way of associating metadata with semantic web content that can be used to handle issues of trust and access, among other things. A Named Graph construct is included here because of the importance of this feature and the expectation that it will eventually be incorporated into the semantic web infrastructure. However, ODM tools are not required to support this element, and it may change in future revisions if the W3C standardizes this in a form that differs from that described in *Named Graphs, Provenance and Trust* [2]

### Attributes

None

### Associations

- graphForNG:RDFGraph [1] in association GraphForNamedGraph – a named graph is associated with exactly one RDF graph.

[2] http://www2005.org/cdrom/docs/p613.pdf

- subGraphOf:NamedGraph [1..*] in association SubGraphOf – links a named graph with named graphs for which it is a subgraph.
- RDFGequivalentGraph:NamedGraph[0..*] in association EquivalentGraph – links a named graph with named graphs that are equivalent.
- Specialize class RDFResource.

### Constraints

[1] The multiplicity on the derived URIRefForResource association on the uriRef role must be 1 for NamedGraphs.

### Semantics

A named graph is a first class object that represents an RDF graph. It is named with a URIReference. Two relationship types are predefined for relationships among named graphs. These are EquivalentGraph and SubGraphOf. These assert equivalence and subset relationships respectively among the RDF graphs (in the graphForNG role) that correspond to the named graphs linked by these relationships.

## 10.5.2 RDFGraph

### Description

An RDF Graph is a set of RDF triples. The set of nodes of an RDF graph is the set of subjects and objects of triples in the graph.

A number of classes in the metamodel, including RDFGraph, RDFStatement, Triple, Document, etc., are included (1) for the sake of completeness, and (2) are provided for vendors to use, as needed from an application perspective. They may not be necessary for all tools, and may not necessarily be accessible to end users, again, depending on the application requirements

### Attributes

None

### Associations

- namedGraph:NamedGraph [0..*] in association GraphForNamedGraph – links an RDF graph with named graphs which may represent it.
- triple:Triple [0..*] in association TripleForGraph – links an RDF graph with the triples it contains.

### Constraints

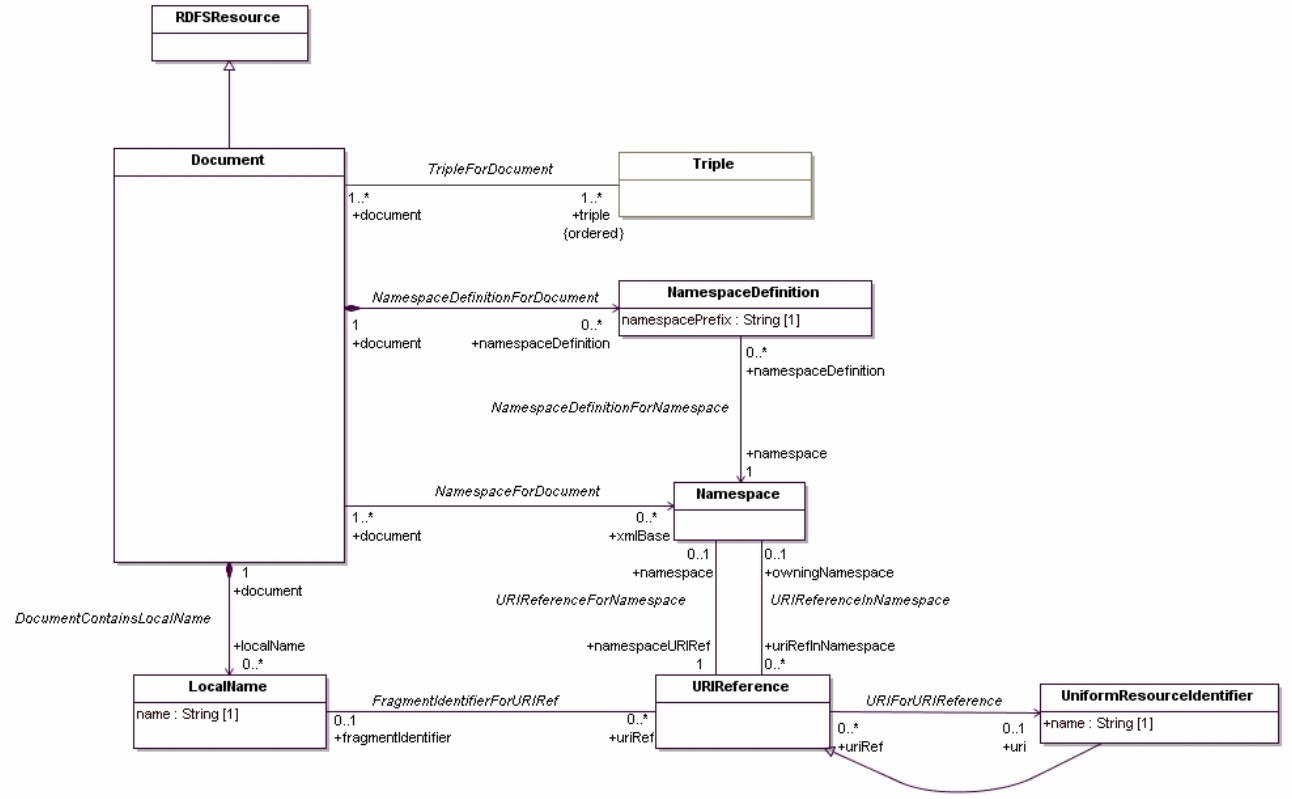
None

## Semantics

As described in [RDF Semantics], RDF is an assertional language, intended for use in defining formal vocabularies and using them to state facts and axioms about some domain.

An RDF graph is defined as a set of RDF triples. A subgraph of an RDF graph is a subset of the triples in the graph. A triple is identified with the singleton set containing it, so that each triple in a graph is considered to be a subgraph. A proper subgraph is a proper subset of the triples in the graph. A ground RDF graph is one with no blank nodes. The assertion of an RDF triple says that some relationship, indicated by the predicate, holds between the things denoted by subject and object of the triple. The assertion of an RDF graph amounts to asserting all the triples in it, so the meaning of an RDF graph is the conjunction (logical AND) of the statements corresponding to all the triples it contains.

4. Make the following changes in Documents and Namespaces:
4.1 In the paragraph entitled “Multiple graphs in the same document.” Change the beginning of the sentence which reads “Thus, the optional name attribute on the Graph class supports...” with “Thus, the NamedGraph class can be used to support...”
4.2 Replace the current documents diagram with the one below which replaces the Statement class with Triple and changes the association and role names for Triple accordingly:



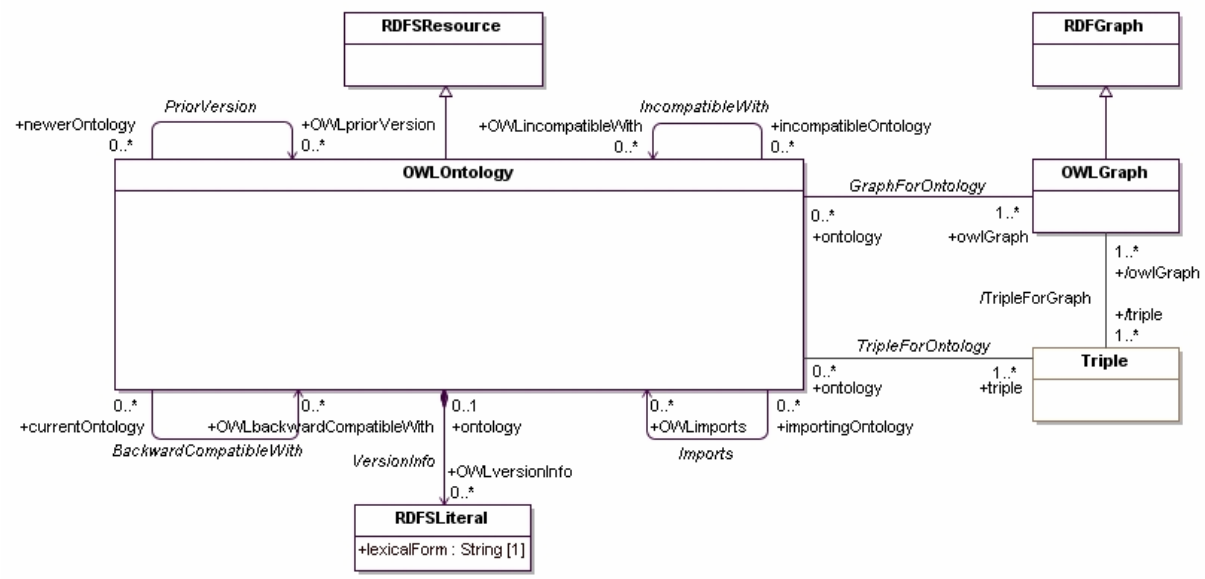

4.3 In the Associations list for Document replace in the third bullet which reads “statement:RDFStatement [1..*] in association StatementForDocument – links a document to the set of triples (statements) in contains (ordered).” with “triple:Triple [1..*] in association TripleForDocument – links a document to the set of triples in contains.”
4.4 Replace the entire10.?.5 RDFStatement (Augemented Definition) subsection with the text below.

## 10.?.5 Triple (Augumented Definition)

### Associations

- document:Document [1..*] in association TripleForDocument – the document(s) containing the triple.

5. In OWLBase Package – OWL Ontology change all occurrences of statement to triple and remove ordering attribute on association end linking to triples.
5.1 At the end of the first sentence of the intro replace "statement" with "triple".
5.2 Replace the Ontology Diagram with the one below which replaces the OWLStatement class with Triple.




5.3 Make the following changes to the OWLGraph subsection
5.3.1 Change the second association for OWLGraph from "owlStatement:OWLStatement [1..*] in association StatementForGraph (deverived – links an OWL graph to the ordered set of triples it contains." to "triple:Triple [1..*] in association TripleForGraph (derived) – links an Owl graph to the set of triples it contains."
5.3.2 Replace constraint "[1] If an OWLStatement s of OWLOntology o, identified through StatementForOntology, is linked through /StatementForGraph to an OWLGraph g, then that OWLGraph g must be linked with OWL Ontology o." with "[1] If an OWLTriple t of OWLOntology o, identified through TripleForOntology, is linked through /TripleForGraph to an OWLGraph g, then that OWLGraph g must be linked with OWL Ontology o."
5.3.3 In constraint "[2] If OWLGraph g is linked with an OWLOntology o, then they must have a statement in common.", change "statement in common" to "triple in common".
5.4 In OWLOntology subsection make the following changes:
5.4.1 In 1st association reading "owlGraph:OWLGraph [1..*] in association GraphForOntology – links an ontology to one or more graphs containing statements that define it." change "statements that" to "triples that".
5.4.2 Replace penultimate association reading "owlStatement:OWLStatement [1..*] in association StatementForOntology – links an ontology to one or more ordered statements it contains." with "owlTriple:OWLTriple [1..*] in association TripleForOntology – links an ontology to one or more triples it contains."

5.4.3 Replace constraint 2 reading "[2] If an OWLStatement s of OWLOntology o, identified through StatementForOntology, is linked through /StatementForGraph to an OWLGraph g, then that OWLGraph g must be linked with OWLOntology o." with "[2] If an OWLTriple s of OWLOntology o, identified through TripleForOntology, is linked through /TripleForGraph to an OWLGraph g, then that OWLGraph g must be linked with OWLOntology o."
5.4.4 In constraint 3 reading "[3] If an OWLGraph g is linked with an OWLOntology o, then they must have a statement in common." replace "statement in common" with "triple in common".
5.4.5 Replace constraint 4 reading "[4] If an OWLStatement is linked through /StatementForGraph to an OWLGraph g of an OWLOntology o (identified through GraphForOntology), then that OWLStatement s must be in OWLOntology o" with "[4] If an OWLTriple is linked through /TripleForGraph to an OWLGraph g of an OWLOntology o (identified through GraphForOntology), then that OWLTriple t must be in OWLOntology o".
5.5 Replace entire subsection 11.2.3 OWL Statement with a subsection for Triple (Augmented Definition) using the following text:

## 11.2.3 Triple (Augumented Definition)

### Associations

- ontology:OWLOntology [0..*] in association TripleForOntology – relates zero or more ontologies to the statements they contain.
- owlGraph:OWLGraph [1..*] in association TripleForGraph (derived) – links an OWL graph to the set of triples it contains.

### Contraints

[1] If an OWLTriple is linked through /TriplesForGraph to an OWLGraph g of an OWLOntology o (identified through GraphForOntology), then that OWLTriple t must be in OWLOntology o.

**Disposition:** **Resolved**

# Disposition: Deferred

# OMG Issue No: 10844

## Title: Figure D.3 notation

### Source:

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

### Summary:

In Annex D, in Figure D.2, the instance names should be underlined. Some of the association end names are so far from the ends of the lines that it's hard to tell which they are referring to.

### Discussion:

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

### Disposition: Deferred to RTF

# Disposition: Deferred

## OMG Issue No: 10846

### Title: Annex D.4 sets

**Source:**

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

**Summary:**

Annex D.4, under Figure D.4 should have another constraint that prevents two instances of NAryProperty from having the same values for the properties of the Nary. Otherwise, it could represent a bag of property values, which OWL properties cannot

**Discussion:**

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

**Disposition: Deferred to RTF**

# Disposition: Deferred

# OMG Issue No: 10849

## Title: Figure 16.1 incomplete

### Source:

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

### Summary:

Figure 16.1 (Key Aspects of UML Class Diagram) is missing the multiplicities on general/specific, and the subsetting between ownedEnd and memberEnd.

### Discussion:

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

### Disposition: Deferred to RTF

# Disposition: Deferred

## OMG Issue No: 10850

### Title: Formal structure

**Source:**

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

**Summary:**

Under Figure 16.1, the first sentence refers to "formal structure". Should explain what this is. Is it the metamodel?

**Discussion:**

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

**Disposition: Deferred to RTF**

# Disposition: Deferred

## OMG Issue No: 10853

### Title: Associations

**Source:**

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

**Summary:**

In Section 16.2.1 (UML Kernel), the discussion around Tables 16.2 through 16.4 seems to be about relational implementations, rather than UML modeling in the sense that is important to OWL. My suggestion is to replace Tables 16.3 and 16.4 with the tabular forms of the metamodel, as in 16.2. The paragraph above Table 16.3, first sentence, modeling associations does not depend on the implementation of classes (the "implementation" usually refers to how the model is translated to a platform). Same comment on the second sentence, which says Table 16.2 is an implementation, when it is only a tabular form of the metamodel. The second sentence refers to the disjoint union of attributes, but there's nothing like this in UML.

**Discussion:**

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

**Disposition:** **Deferred to RTF**

# Disposition: Deferred

# OMG Issue No: 10863

## Title: Distinct associations, ownedAttribute associations

### Source:

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

### Summary:

In 16.2.2 (Class and Property - Basics), in the paragraph below Table 16.6, there is the sentence " Note that UML ownedAttribute M2 associations are distinct, even if ownedAttributes have the same name associated with different classes." What are "M2 owned attribute associations"? In the case of M1 properties, properties with the same name may be on different classes, but if they inherit from the same base class where a property of that name is introduced, then they are the same property from OWL's point of view. There is usually no need to translate to unique OWL properties, just restrictions. See next issue.

### Discussion:

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

**Disposition:** **Deferred to RTF**

# Disposition: Deferred

## OMG Issue No: 10864

### Title: Distinct associations, restrictions

**Source:**

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

**Summary:**

In 16.2.2 (Class and Property - Basics), in the paragraph above Table 16.7, says the OWL properties "arising" (I assume due to translation) from a UML model are distinct, that OWL restrictions aren't in the translation. UML can redefine properties in subtypes of the classes where the property is introduced, which is equivalent to restriction. The method employed in the chapter is not adequate.

**Discussion:**

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

**Disposition:** **Deferred to RTF**

# Disposition: Deferred

## OMG Issue No: 10865

### Title: Identifiers

**Source:**

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

**Summary:**

In 16.2.2 (Class and Property - Basics), in the paragraph below Table 16.7, the first sentence says the translation assumes that a single name identifies each instance of the class. It isn't necessary to assume this, since UML does not assume a relational semantics. The notion of identity is primitive in UML and applies even to instances of classes that have no attributes or attribute values. The rest of the paragraph may apply to relational implementations, but is not a general solution. It also assumes that the property names of classes are always different, but distinct classes can have the same properties in UML. (BTW, fourth sentence, "values" -> "names")

**Discussion:**

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

**Disposition:** **Deferred to RTF**

# Disposition: Deferred

## OMG Issue No: 10866

### Title: Associations

**Source:**

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

**Summary:**

In 16.2.2 (Class and Property - Basics), in the paragraph below Table 16.7, gives the wrong translation to OWL for UML associations. UML associations have properties at end, and these are often navigable. Binary associations in UML translate to two inverse properties, using these property names, not the association name. See the UML profile for OWL for the translation options for associations, and the third paragraph in 16.2.3.

**Discussion:**

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

**Disposition:** **Deferred to RTF**

# Disposition: Deferred

## OMG Issue No: 10867

### Title: Subproperties and redefinition

**Source:**

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

**Summary:**

In 16.2.2 (Class and Property - Basics), in the paragraph below Table 16.8, the second sentence, in parentheses, says that subproperties translate to redefinition. The translation is only to subsetting. Also the wording in parenthetical remark conflates association generalization with property subsetting. Same comment about the last sentence of this paragraph, which omits property subsetting. Same comment about the translation given in the next paragraph. UML associations, even binary ones, can have more than one property, and each property can be subsetted if the association as a whole is specialized, but they don't all need to be.

**Discussion:**

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

**Disposition:** **Deferred to RTF**

# Disposition: Deferred

# OMG Issue No: 10871

## Title: Association member ends

### Source:

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

### Summary:

In 16.2.2 (Class and Property - Basics), third paragraph under Figure 16.3 describes UML member ends incorrectly. The second sentence says that the classes Staff and Enrolled are member ends, but member ends are classes, not properties.

### Discussion:

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

**Disposition:** **Deferred to RTF**

# Disposition: Deferred

# OMG Issue No: 10872

## Title: Table 16.9 and Naries

### Source:

NIST (Mr. Conrad Bock, conrad.bock@nist.gov conradb@cme.nist.gov)

### Summary:

In 16.2.2 (Class and Property - Basics), Table 16.9 replace the "Parts" header with "Properties". The Reification property isn't necessary, because AssociationClass is both a class and association, there is no separate reification of the association (this is necessary in OWL DL, however, and even in OWL Full, some extension is needed for a subclass of Property and Class to correspond to a UML Association Class). The text below the table uses the term "implements" which doesn't apply (these are platform-dependent models), and introduces the reified association, which doesn't exist in UML.

### Discussion:

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

**Disposition:** **Deferred to RTF**

# Disposition: Deferred

# OMG Issue No: 10873

## Title: Translation of binary associations

### Source:

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

### Summary:

In 16.2.3 (More Advanced Concepts), third paragraph, next to last sentence, the domain of the OWL property is the class at the non-navigable end. This is because the ends of associations in UML are placed opposite the class they navigate from.

### Discussion:

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

**Disposition:** **Deferred to RTF**

# Disposition: Deferred

## OMG Issue No: 10874

### Title: UML Thing 2

**Source:**

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

**Summary:**

In 16.2.3 (More Advanced Concepts), fourth paragraph, last sentence, it's clear what the tool sets would do with it: provide Thing for modelers to explicitly assign as the end of a class, and use it as the default end class when none is given.

**Discussion:**

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

**Disposition:** **Deferred to RTF**

# Disposition: Deferred

## OMG Issue No: 10875

### Title: Individuals

**Source:**

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

**Summary:**

In 16.2.3 (More Advanced Concepts), fifth paragraph, the first sentence draws a conclusion ("therefore") without any justification. Individuals in OWL are all classified by Thing, whether or not this is explicitly recorded. It's just syntactic sugar to omit it. In UML, instance specifications can be classified by Thing in the model library and have the same semantics as OWL individual.

**Discussion:**

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

**Disposition:** **Deferred to RTF**

# Disposition: Deferred

## OMG Issue No: 10876

### Title: Disjoint

**Source:**

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

**Summary:**

In 16.2.3 (More Advanced Concepts), sixth paragraph, parenthetical remark should note that with UML Thing the same is true in UML).

**Discussion:**

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

**Disposition:** **Deferred to RTF**

# Disposition: Deferred

## OMG Issue No: 10877

### Title: Classes of classes

**Source:**

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

**Summary:**

In 16.2.3 (More Advanced Concepts), seventh paragraph, the second sentence implies classes are not instances in OWL DL, but even in DL, OWL Class is a class of classes, by definition. For example, an ontology of animals might have the class Dog, which is an instance (of OWL Class) and a class (of Fido, Rover, and other individual dogs). The third sentence should be moved to be the second, and start with "however"|, because it is an exception to the first sentence. After "declaration" should be replaced with "a common superclass".

**Discussion:**

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

**Disposition:** **Deferred to RTF**

# Disposition: Deferred

## OMG Issue No: 10879

### Title: Mandatory properties

**Source:**

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

**Summary:**

In 16.2.3 (More Advanced Concepts), under the XML example, third paragraph, I assume "may not" should be "must". The property must have values for every individual

**Discussion:**

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

**Disposition:** **Deferred to RTF**

# Disposition: Deferred

## OMG Issue No: 10884

### Title: Derivation

**Source:**

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

**Summary:**

In 16.2.3 (More Advanced Concepts), under the XML example, the paragraph starting "UML allows a property", UML derivation means derivation from values of properties, not from generalizations of the classes that are the domain of those properties.

**Discussion:**

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

**Disposition:** **Deferred to RTF**

# Disposition: Deferred

## OMG Issue No: 10885

### Title: Table 16.10

**Source:**

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

**Summary:**

In 16.2.3 (More Advanced Concepts), Table 16.10, the names of classes are capitalized in UML. The UML element corresponding to OWL subproperty is property subsetting. N-aries and association classes are not well-supported in OWL, so don't belong in a table of common features (see other issues on n-aries and association classes).

**Discussion:**

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

**Disposition:** **Deferred to RTF**

# Disposition: Deferred

## OMG Issue No: 10886

### Title: Table 16.11

**Source:**

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

**Summary:**

In 16.2.3 (More Advanced Concepts), Table 16.11, the last row (classes as instances), is supported in OWL Full, and even in DL (OWL Class is a class of classes, by definition). For example, an ontology of animals might have the class Dog, which is an instance (of OWL Class) and a class (of Fido, Rover, and other individual dogs). This table should be in Section 16.6 (In UML but not OWL).

**Discussion:**

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

**Disposition:** **Deferred to RTF**

# Disposition: Deferred

## OMG Issue No: 10887

### Title: Table 16.12, Thing

**Source:**

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

**Summary:**

In 16.2.3 (More Advanced Concepts), Table 16.12, the first row (Thing) should be qualified by the fact that OWL is using syntactic sugar for global properties and autonomous individuals, and that the standard UML model library given in ODM enables UML to support these features. This table should be in Section 16.5 (OWL but not UML).

**Discussion:**

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

**Disposition:** **Deferred to RTF**

# Disposition: Deferred

# OMG Issue No: 10888

## Title: Table 16.12, AllValuesFrom

### Source:

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

### Summary:

In 16.2.3 (More Advanced Concepts), Table 16.12, the second row (AllValuesFrom), AllValuesFrom is directly supported in UML as property subsetting.

### Discussion:

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

**Disposition:** **Deferred to RTF**

# Disposition: Deferred

# OMG Issue No: 10889

## Title: Table 16.12, classes as instances

### Source:

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

### Summary:

In 16.2.3 (More Advanced Concepts), Table 16.12, class as instances appears in both this table and Table 16.11.

### Discussion:

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

### Resolution:

**Defer to RTF**

# Disposition: Deferred

## OMG Issue No: 10890

### Title: Table 16.12, disjoint

**Source:**

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

**Summary:**

In 16.2.3 (More Advanced Concepts), Table 16.12, last row. UML supports declaring disjoint classes.

**Discussion:**

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

**Disposition:** **Deferred to RTF**

# Disposition: Deferred

## OMG Issue No: 10891

### Title: Inferring subsumption

**Source:**

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

**Summary:**

In Section 16.5.1 (Predicate Definition Language), first sentence, UML can support subsumption reasoning also, see http://www.inf.unibz.it/~calvanese/papers-html/AIJ-2005.html

**Discussion:**

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

**Disposition:** **Deferred to RTF**

# Disposition: Deferred

## OMG Issue No: 10892

### Title: Boolean combination

**Source:**

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

**Summary:**

In Section 16.5.1 (Predicate Definition Language), third sentence, UML supports the equivalent of unionOf.

**Discussion:**

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

**Disposition:** **Deferred to RTF**

# Disposition: Deferred

## OMG Issue No: 10893

### Title: Names, unique names

**Source:**

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

**Summary:**

In Section 16.5.2 (Names), the first two paragraphs imply that UML assumes unique names. M1 instance specifications in UML can have different names, but refer to the same M0 individual. They can also have the same name and refer to different M0 individuals. The third paragraph implies UML does not have name management (given the title of Section 16.5), which of course it does in namespaces.

**Discussion:**

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

**Disposition:** **Deferred to RTF**

# Disposition: Deferred

## OMG Issue No: 10894

### Title: Names, UML namespaces

**Source:**

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

**Summary:**

In Section 16.5.2 (Names), next to last paragraph, namespaces are supported at all metalevels in UML/MOF.

**Discussion:**

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

**Disposition:** **Deferred to RTF**

# Disposition: Deferred

## OMG Issue No: 10895

### Title: Other OWL

**Source:**

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

**Summary:**

In Section 16.5.3 (Other OWL Developments), should refer to OWL 1.1.

**Discussion:**

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

**Disposition:** **Deferred to RTF**

# Disposition: Deferred

# OMG Issue No: 10897

## Title: Complex Objects

### Source:

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

### Summary:

In Section 16.6.2 (Complex Objects), the first two paragraphs and the last omit the critical aspect of connectors, that they provide a model of the interconnections of objects that are all related to the same other object. For example, the engine in a car powers the wheels and is controlled by the driver. See http://www.jot.fm/issues/issue_2004_11/column5

### Discussion:

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

### Discussion:

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

### Resolution:

**Defer to RTF**

# Disposition: Deferred

## OMG Issue No: 10898

### Title: Keywords

**Source:**

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

**Summary:**

In Section 16.6.4 (Keywords) keywords are confused with stereotypes. Keywords don't extend, stereotypes do. Keywords are just an element of notation.

**Discussion:**

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

**Disposition:** **Deferred to RTF**

# Disposition: Deferred

## OMG Issue No: 10899

### Title: Profiles

**Source:**

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

**Summary:**

In Section 16.6.5 (Profiles), third paragraph says that profiles not necessary because of metalevel separation. They are used as an alternative way to extend M2 classes with subclasses, in particular, where the subclasses are defined at M1, even though they have the effect of being at M2.

**Discussion:**

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

**Disposition:** **Deferred to RTF**

# Disposition: Deferred

## OMG Issue No: 10900

### Title: UML to OWL, OWL-DL

**Source:**

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

**Summary:**

Section 16.3 (UML to OWL), third sentence, says the mapping is only to OWL-DL. Why not OWL Full?

**Discussion:**

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

**Disposition:** **Deferred to RTF**

# Disposition: Deferred

# OMG Issue No: 10901

## Title: UML to OWL, Table 16.10

### Source:

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

### Summary:

Section 16.3 (UML to OWL), third paragraph, first sentence, says the mapping is based on Table 16.10. The section containing that table has a lot of errors about UML. It would be better to base the mapping on the profile (Chapter 14), which has had much more review from the UML perspective

### Discussion:

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

### Resolution:

**Defer to RTF**

# Disposition: Deferred

## OMG Issue No: 10902

### Title: Object identification in UML

**Source:**

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

**Summary:**

Section 16.3.1 (Naming Issues), second paragraph says UML (packageable) elements are identified by name. UML packageable elements can be anonymous, and they still have identity. The notion of identity is primitive in UML and applies even when no names are used.

**Discussion:**

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

**Disposition:** **Deferred to RTF**

# Disposition: Deferred

## OMG Issue No: 10904

### Title: N-aries. Section 16.3.6

**Source:**

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

**Summary:**

Section 16.3.6 (Association Classes and N-ary Associations), second paragraph, says the translation treats association classes and naries the same way. Association classes are not the same as n-aries, see issues filed on n-ries in 16.2.2 (Class and Property - Basics).

**Discussion:**

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

**Disposition:** **Deferred to RTF**

# Disposition: Deferred

## OMG Issue No: 10905

### Title: Multiplicity

**Source:**

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

**Summary:**

Section 16.3.7 (Multiplicity), the translation can also be to OWL FunctionalProperty or InverseFunctionalProperty if the multiplicity is 1.

**Discussion:**

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

**Disposition:** **Deferred to RTF**

# Disposition: Deferred

# OMG Issue No: 10906

## Title: navigableOwnedEnd

### Source:

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

### Summary:

The introduction to Section 16.3.5 (Binary Association To Object Property) accounts for navigableOwnedEnd, but the introduction to Section 16.3.8 () Association Generalization) does not.

### Discussion:

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

**Disposition:** **Deferred to RTF**

# Disposition: Deferred

# OMG Issue No: 10907

## Title: Enumeration literals

### Source:

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

### Summary:

The introduction to Section 16.3.4 (Attribute to Property) accounts for enumeration literals that are instances of classifiers, but the introduction to Section 16.3.9 (Enumeration) does not.

### Discussion:

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

**Disposition:** **Deferred to RTF**

# Disposition: Deferred

# OMG Issue No: 10908

## Title: Individuals, mapping

### Source:

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

### Summary:

Section 16.4.1.1 (Mapping for Individuals), first sentence says the profile (Chapter 14) represents individuals as a singleton class. This is incorrect. The profile models individuals as instance specifications. To give property values to the individual, the profile uses a singleton class. Section 16.4.1.1 incorrectly concludes that individuals should not be mapped, which affects 16.4.1.2 (Mapping for Enumerated Classes) and Section 16.4.13 (Annotation Properties to Comments).

### Discussion:

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

**Disposition:** **Deferred to RTF**

# Disposition: Deferred

## OMG Issue No: 10909

### Title: complementOf and disjointWith

**Source:**

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

**Summary:**

Section 16.4.1.3 (Mapping for complementOf and disjointWith) says UML has constructions for complementOf and disjointWith in the PowerTypes pacakge. It actually has constructs for unionOf and disjointWith. Section 16.4.1.3 says no mapping is given because the OWL constructs are pairwise, but OWL unionOf and disjointWith are not pairwise, they can apply to any number of classes.

**Discussion:**

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

**Disposition:** **Deferred to RTF**

# Disposition: Deferred

# OMG Issue No: 10910

## Title: Multiple Domains or Ranges for Properties

### Source:

NIST (Mr. Conrad Bock, conrad.bock@nist.gov conradb@cme.nist.gov)

### Summary:

Section 16.4.1.4 (Multiple Domains or Ranges for Properties) says that multiple domains or ranges for properties is equivalent to the intersection of the domains and ranges. UML properties have at most one type, and intersection can't be represented in UML without the profile (Chapter 14). How is this translated?

### Discussion:

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

### Resolution:

**Defer to RTF**

# Disposition: Deferred

# OMG Issue No: 10911

## Title: Ontology Properties

### Source:

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

### Summary:

Section 16.4.3.2 (Ontology Properties to Comments) should use dependencies for some of the translations. See the profile (Chapter 14).

### Discussion:

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

**Disposition:** **Deferred r to RTF**

# Disposition: Deferred

## OMG Issue No: 10912

### Title: Anonymous Classes

**Source:**

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

**Summary:**

Section 16.4.4.3 (Anonymous Class to Class) can translate blank nodes to anonymous classes in UML.

**Discussion:**

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

**Disposition:** **Deferred to RTF**

# Disposition: Deferred

## OMG Issue No: 10913

### Title: Universal Superclass

**Source:**

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

**Summary:**

Section 16.4.5.2 (Universal Superclass) should also refer to Annex A.

**Discussion:**

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

**Disposition:** **Deferred to RTF**

# Disposition: Deferred

## OMG Issue No: 10914

### Title: Constructed Classes

**Source:**

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

**Summary:**

The introduction to Section 16.4.6 (Constructed Classes) refers to OWL "difference". I assume this is supposed to be complementOf. The introduction to the section says intersection can be mapped to subclass relationships, but this isn't true, at least not without the profile, see intersection in Chapter 14. It also says union can be translated to subclass relationships, but doesn't mention UML generalization sets and isCovering, see Section 16.3.10 (Powertypes).

**Discussion:**

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

**Disposition:** **Deferred to RTF**

# Disposition: Deferred

# OMG Issue No: 10915

## Title: Range Restriction Restriction Classes

### Source:

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

### Summary:

The introduction to Section 16.4.8 (Range Restriction Restriction Classes) refers to properties "behaving". Properties are static, they don't "behave".

### Discussion:

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

**Disposition:** **Deferred to RTF**

# Disposition: Deferred

## OMG Issue No: 10916

### Title: Range Restriction Restriction Classes

**Source:**

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

**Summary:**

The introduction to Section 16.4.8 (Range Restriction Restriction Classes) says the translation is to a comments. But AllValuesFrom translates directly to redefinition of property types, see the profile (Chapter 14).

**Discussion:**

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

**Disposition:** **Deferred to RTF**

# Disposition: Deferred

# OMG Issue No: 10917

## Title: Properties in OWL

### Source:

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

### Summary:

The end of Section 16.4.9 (Properties in OWL) refers to multiple domains being equivalent to the domain being an intersection. This does not translate to UML, see issue on Constructed Classes

### Discussion:

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved.

**Disposition:** **Deferred to RTF**

# Disposition: Deferred

# OMG Issue No: 11099

## Title: Constraints in the RDF Metamodel Chapter (10) should be specified in OCL

### Source:

Sandpiper Software, Inc. (Mrs. Elisa F. Kendall, ekendall@sandsoft.com)

### Summary:

Text based descriptions of constraints provided in chapter 10 with the RDF metamodel should be specified in OCL.

### Discussion:

The FTF expected resolutions to outstanding concerns to revise the RDF and OWL metamodel. The precise metamodel changes were determined at the end of the life of the FTF (in poll 5). Thus it made sense to defer the enhancements to the spec. (formalizing the constraints in OCL) to a later revision.

**Disposition:** **Deferred**

# Disposition: Deferred

# OMG Issue No: 11100

## Title: Constraints in the OWL Metamodel Chapter (10) should be specified in OCL

### Source:

Sandpiper Software, Inc. (Mrs. Elisa F. Kendall, ekendall@sandsoft.com)

### Summary:

Text based descriptions of constraints provided in chapter 11 with the OWL metamodel should be specified in OCL.

### Discussion:

The FTF expected resolutions to outstanding concerns to revise the RDF and OWL metamodel. The precise metamodel changes were determined at the end of the life of the FTF (in poll 5). Thus it made sense to defer the enhancements to the spec. (formalizing the constraints in OCL) to a later revision.

**Disposition:** **Deferred**

# Disposition: Deferred

# OMG Issue No: 11102

## Title: Mapping from Common Logic to OWL should be revised

### Source:

Sandpiper Software, Inc. (Mrs. Elisa F. Kendall, ekendall@sandsoft.com)

### Summary:

The mapping from RDFS and OWL to CL should be revised to reflect metamodel changes in CL due to finalization of ISO 24707.

### Discussion:

FTF resources were scarce and priority was given to issues against normative sections, hence many issues such as this were left unresolved. As with the OCL constraint issues, it was also felt best to delay this until after the metamodel revision.

**Disposition:** **Defer to RTF**

# Disposition: Deferred

# OMG Issue No: 11320

## Title: Thing in the Profile

### Source:

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

### Summary:

Thing in the Profile. The UML Profile (Chapter 16) should use Annex A Thing instead of an anonymous class to model owl:Thing. Search on "Thing" (case sensitive) in the profile.

### Discussion:

We addressed the most urgent profile issues. We would expect this to be among those addressed first in RTF.

**Disposition:** **Deferred to RTF**

# Disposition: Deferred

# OMG Issue No: 11321

## Title: RDFSContainer-MembershipProperty

### Source:

NIST (Mr. Conrad Bock, conrad.bock@nist.gov)

### Summary:

In Annex A, RDFSContainer-MembershipProperty should be moved to the UML Profile chapter as a stereotype based on UML:Property.

### Discussion:

We addressed the most urgent profile issues. We would expect this to be among those addressed first in RTF.

**Disposition:** **Deferred to RTF**

# Disposition: Closed, no change

## OMG Issue No: 10843

### Title: Annex D.2, OWL Full

**Source:**

NIST (Mr. Conrad Bock, conrad.bock@nist.gov conradb@cme.nist.gov)

**Summary:**

In Annex D.2, under Figure D1, the second sentence says that OWL Full must be used to subclass OWL class. Why can't OWL class be subclasses in OWL Lite or DL? All instances of subclasses of OWL:Class are also OWL classes, and presumably wouldn't violate the constraints of OWL Lite or DL just because of the subclassing.

**Discussion:**

OWL DL and OWL Lite prohibit “messing with the vocabulary”. This is intended to prevent users from changing the semantics of the language. OWL Class is an element of this vocabulary. When we asked Description Logic experts about this issue, they reported that the OWL DL semantics prohibits subclassing OWL Class. This prohibition also holds for OWL Lite which is a syntactic subset of OWL DL. Thus the text at issue is correct, subclassing OWL Class puts one into OWL Full.

**Disposition:** **Closed, no change**

# Disposition: Duplicate/merged

## OMG Issue No: 10882

### Title: Transitive Closure

**Source:**

NIST (Mr. Conrad Bock, conrad.bock@nist.gov conradb@cme.nist.gov)

**Summary:**

In 16.2.3 (More Advanced Concepts), under the XML example, the paragraph starting "Note that a consequence of", seems to have lost its context. It doesn't appear related to the paragraphs around it. If it is, this should be clarified.

**Disposition:** **See issue 10880 for disposition**

# Disposition: Duplicate/merged

## OMG Issue No: 11107

### Title: Set value for annotation property in UML profile for OWL

**Source:**

International Business Machines (Mr. Guo Tong Xie, xieguot@cn.ibm.com)

**Summary:**

In section 14.2.3.1, it is not clear on how to set values for annotation property

**Disposition:** **See issue 12391 for disposition**