RDF Schema 1.1

W3C Recommendation 25 February 2014

This version:
http://www.w3.org/TR/2014/REC-rdf-schema-20140225/
Latest published version:
http://www.w3.org/TR/rdf-schema/
Previous version:
http://www.w3.org/TR/2014/PER-rdf-schema-20140109/

Editors:
Dan Brickley, Google
R.V. Guha, Google
Previous Editors:
Brian McBride

Please check the **errata** for any errors or issues reported since publication.

This document is also available in this non-normative format: diff w.r.t. 2004 Recommendation

The English version of this specification is the only normative version. Non-normative translations may also be available.

Abstract

RDF Schema provides a data-modelling vocabulary for RDF data. RDF Schema is an extension of the basic RDF vocabulary.

Status of This Document

This section describes the status of this document at the time of its publication. Other documents may supersede this document. A list of current W3C publications and the latest revision of this technical report can be found in the W3C technical reports index at http://www.w3.org/TR/.

This document is an edited version of the 2004 RDF Schema Recommendation. The purpose of this revision is to make this document available as part of the RDF 1.1 document set. Changes are limited to errata, revised references, terminology updates, and adaptations to the introduction. The title of the document was changed from "RDF Vocabulary Description Language 1.0: RDF Schema" to "RDF Schema 1.1". The technical content of the document is unchanged. Details of the changes are listed in the Changes section. Since the edits to this document do not constitute a technical change the Director decided no new implementation report was required.

This document was published by the RDF Working Group as a Recommendation. If you wish to make comments regarding this document, please send them to public-rdf-comments@w3.org (subscribe, archives). All comments are welcome.

This document has been reviewed by W3C Members, by software developers, and by other W3C groups and interested parties, and is endorsed by the Director as a W3C Recommendation. It is a stable document and may be used as reference material or cited from another document. W3C's role in making the Recommendation is to draw attention to the specification and to promote its widespread deployment. This enhances the functionality and interoperability of the Web.

This document was produced by a group operating under the 5 February 2004 W3C Patent Policy. W3C maintains a public list of any patent disclosures made in connection with the deliverables of the group; that page also includes instructions for disclosing a patent. An individual who has actual knowledge of a patent which the individual believes contains Essential Claim(s) must disclose the information in accordance with section 6 of the W3C Patent Policy.

Table of Contents

1. Introduction

RDF Schema provides a data-modelling vocabulary for RDF data. It is complemented by several companion documents which describe the basic concepts and abstract syntax of RDF [RDF11-CONCEPTS], the formal semantics of RDF [RDF11-MT], and various concrete syntaxes for RDF, such as Turtle [TURTLE], TriG, [TRIG], and JSON-LD [JSON-LD]. The RDF Primer [RDF11-PRIMER] provides an informal introduction and examples of the use of the concepts specified in this document.

This document is intended to provide a clear specification of RDF Schema to those who find the formal semantics specification [RDF11-MT] daunting. Thus, this document duplicates material also specified in the RDF Semantics specification. Where there is disagreement between this document and the RDF Semantics specification, the RDF Semantics specification should be taken to be correct.

RDF Schema is a semantic extension of RDF. It provides mechanisms for describing groups of related resources and the relationships between these resources. RDF Schema is written in RDF using the terms described in this document. These resources are used to determine characteristics of other resources, such as the domains and ranges of properties.

The RDF Schema class and property system is similar to the type systems of object-oriented programming languages such as Java. RDF Schema differs from many such systems in that instead of defining a class in terms of the properties its instances may have, RDF Schema describes properties in terms of the classes of resource to which they apply. This is the role of the domain and range mechanisms described in this specification. For example, we could define the `eg:author` property to have a domain of `eg:Document` and a range of `eg:Person`, whereas a classical object oriented system might typically define a class `eg:Book` with an attribute called `eg:author` of type `eg:Person`. Using the RDF approach, it is easy for others to subsequently define additional properties with a domain of eg:`Document` or a range of `eg:Person`. This can be done without the need to re-define the original description of these classes. One benefit of the RDF property-centric approach is that it allows anyone to extend the description of existing resources, one of the architectural principles of the Web [BERNERS-LEE98].

This specification does not attempt to enumerate all the possible forms of representing the meaning of RDF classes and properties. Instead, the RDF Schema strategy is to acknowledge that there are many techniques through which the meaning of classes and properties can be described. Richer vocabulary or 'ontology' languages such as OWL [OWL2-OVERVIEW], inference rule languages and other formalisms (for example temporal logics) will each contribute to our ability to capture meaningful generalizations about data in the Web.

The language defined in this specification consists of a collection of RDF resources that can be used to describe other RDF resources in application-specific RDF vocabularies. The core vocabulary is defined in a namespace informally called `rdfs` here. That namespace is identified by the IRI

```
http://www.w3.org/2000/01/rdf-schema#
```

and is conventionally associated with the prefix `rdfs:`. This specification also uses the prefix `rdf:` to refer to the RDF namespace

```
http://www.w3.org/1999/02/22-rdf-syntax-ns#
```

For convenience and readability, this specification uses an abbreviated form to represent IRIs. A name of the form prefix:suffix should be interpreted as a IRI consisting of the IRI associated with the prefix concatenated with the suffix.

2. Classes

Resources may be divided into groups called classes. The members of a class are known as *instances* of the class. Classes are themselves resources. They are often identified by IRIs and may be described using RDF properties. The `rdf:type` property may be used to state that a resource is an instance of a class.

RDF distinguishes between a class and the set of its instances. Associated with each class is a set, called the class extension of the class, which is the set of the instances of the class. Two classes may have the same set of instances but be different classes. For example, the tax office may define the class of people living at the same address as the editor of this document. The Post Office may define the class of people whose address has the same zip code as the address of the author. It is possible for these classes to have exactly the same instances, yet to have different properties. Only one of the classes has the property that it was defined by the tax office, and only the other has the property that it was defined by the Post Office.

A class may be a member of its own class extension and may be an instance of itself.

The group of resources that are RDF Schema classes is itself a class called `rdfs:Class`.

If a class C is a *subclass* of a class C', then all instances of C will also be instances of C'. The `rdfs:subClassOf` property may be used to state that one class is a subclass of another. The term super-class is used as the inverse of subclass. If a class C' is a super-class of a class C, then all instances of C are also instances of C'.

The RDF Concepts and Abstract Syntax [RDF11-CONCEPTS] specification defines the RDF concept of an RDF datatype. All datatypes are classes. The instances of a class that is a datatype are the members of the value space of the datatype.

2.1 rdfs:Resource

All things described by RDF are called *resources*, and are instances of the class `rdfs:Resource`. This is the class of everything. All other classes are subclasses of this class. `rdfs:Resource` is an instance of `rdfs:Class`.

2.2 rdfs:Class

This is the class of resources that are RDF classes. `rdfs:Class` is an instance of `rdfs:Class.`

2.3 rdfs:Literal

The class `rdfs:Literal` is the class of literal values such as strings and integers. Property values such as textual strings are examples of RDF literals.

`rdfs:Literal` is an instance of `rdfs:Class`. rdfs:Literal is a subclass of rdfs:Resource.

2.4 rdfs:Datatype

`rdfs:Datatype` is the class of datatypes. All instances of `rdfs:Datatype` correspond to the RDF model of a datatype described in the RDF Concepts specification [RDF11-CONCEPTS]. `rdfs:Datatype` is both an instance of and a subclass of `rdfs:Class`. Each instance of `rdfs:Datatype` is a subclass of rdfs:Literal.

2.5 rdf:langString

The class `rdf:langString` is the class of language-tagged string values. `rdf:langString` is an instance of `rdfs:Datatype` and a subclass of `rdfs:Literal`.

2.6 rdf:HTML

This section is non-normative.

The class `rdf:HTML` is the class of HTML literal values. `rdf:HTML` is an instance of `rdfs:Datatype` and a subclass of `rdfs:Literal`.

2.7 rdf:XMLLiteral

This section is non-normative.

The class rdf:XMLLiteral is the class of XML literal values. rdf:XMLLiteral is an instance of rdfs:Datatype and a subclass of rdfs:Literal.

2.8 rdf:Property

rdf:Property is the class of RDF properties. rdf:Property is an instance of rdfs:Class.

3. Properties

The RDF Concepts and Abstract Syntax specification [RDF11-CONCEPTS] describes the concept of an RDF property as a relation between subject resources and object resources.

This specification defines the concept of subproperty. The rdfs:subPropertyOf property may be used to state that one property is a subproperty of another. If a property P is a subproperty of property P', then all pairs of resources which are related by P are also related by P'. The term super-property is often used as the inverse of subproperty. If a property P' is a super-property of a property P, then all pairs of resources which are related by P are also related by P'. This specification does not define a top property that is the super-property of all properties.

> NOTE
>
> The basic facilities provided by rdfs:domain and rdfs:range do not provide any direct way to indicate property restrictions that are local to a class. Although it is possible to combine use rdfs:domain and rdfs:range with sub-property hierarchies, direct support for such declarations are provided by richer Web Ontology languages such as OWL [OWL2-OVERVIEW].

3.1 rdfs:range

rdfs:range is an instance of rdf:Property that is used to state that the values of a property are instances of one or more classes.

The triple

```
P rdfs:range C
```

states that P is an instance of the class rdf:Property, that C is an instance of the class rdfs:Class and that the resources denoted by the objects of triples whose predicate is P are instances of the class C.

Where P has more than one rdfs:range property, then the resources denoted by the objects of triples with predicate P are instances of all the classes stated by the rdfs:range properties.

The rdfs:range property can be applied to itself. The rdfs:range of rdfs:range is the class rdfs:Class. This states that any resource that is the value of an rdfs:range property is an instance of rdfs:Class.

The rdfs:range property is applied to properties. This can be represented in RDF using the rdfs:domain property. The rdfs:domain of rdfs:range is the class rdf:Property. This states that any resource with an rdfs:range property is an instance of rdf:Property.

3.2 rdfs:domain

rdfs:domain is an instance of rdf:Property that is used to state that any resource that has a given property is an instance of one or more classes.

A triple of the form:

```
P rdfs:domain C
```

states that P is an instance of the class rdf:Property, that C is a instance of the class rdfs:Class and that the resources denoted by the subjects of triples whose predicate is P are instances of the class C.

Where a property P has more than one rdfs:domain property, then the resources denoted by subjects of triples with predicate P are instances of all the classes stated by the rdfs:domain properties.

The rdfs:domain property may be applied to itself. The rdfs:domain of rdfs:domain is the class rdf:Property. This states that any resource with an rdfs:domain property is an instance of rdf:Property.

The rdfs:range of rdfs:domain is the class rdfs:Class. This states that any resource that is the value of an rdfs:domain property is an instance of rdfs:Class.

3.3 rdf:type

rdf:type is an instance of rdf:Property that is used to state that a resource is an instance of a class.

A triple of the form:

```
R rdf:type C
```

states that C is an instance of rdfs:Class and R is an instance of C.

The rdfs:domain of rdf:type is rdfs:Resource. The rdfs:range of rdf:type is rdfs:Class.

3.4 rdfs:subClassOf

The property rdfs:subClassOf is an instance of rdf:Property that is used to state that all the instances of one class are instances of another.

A triple of the form:

```
C1 rdfs:subClassOf C2
```

states that C1 is an instance of rdfs:Class, C2 is an instance of rdfs:Class and C1 is a subclass of C2. The rdfs:subClassOf property is transitive.

The rdfs:domain of rdfs:subClassOf is rdfs:Class. The rdfs:range of rdfs:subClassOf is rdfs:Class.

3.5 rdfs:subPropertyOf

The property rdfs:subPropertyOf is an instance of rdf:Property that is used to state that all resources related by one property are also related by another.

A triple of the form:

```
P1 rdfs:subPropertyOf P2
```

states that P1 is an instance of rdf:Property, P2 is an instance of rdf:Property and P1 is a subproperty of P2. The rdfs:subPropertyOf property is transitive.

The rdfs:domain of rdfs:subPropertyOf is rdf:Property. The rdfs:range of rdfs:subPropertyOf is rdf:Property.

3.6 rdfs:label

rdfs:label is an instance of rdf:Property that may be used to provide a human-readable version of a resource's name.

A triple of the form:

```
R rdfs:label L
```

states that L is a human readable label for R.

The rdfs:domain of rdfs:label is rdfs:Resource. The rdfs:range of rdfs:label is rdfs:Literal.

Multilingual labels are supported using the language tagging facility of RDF literals.

3.7 rdfs:comment

rdfs:comment is an instance of rdf:Property that may be used to provide a human-readable description of a resource.

A triple of the form:

```
R rdfs:comment L
```

states that L is a human readable description of R.

The rdfs:domain of rdfs:comment is rdfs:Resource. The rdfs:range of rdfs:comment is rdfs:Literal.

A textual comment helps clarify the meaning of RDF classes and properties. Such in-line documentation complements the use of both formal techniques (Ontology and rule languages) and informal (prose documentation, examples, test cases). A variety of documentation forms can be combined to indicate the intended meaning of the classes and properties described in an RDF vocabulary. Since RDF vocabularies are expressed as RDF graphs, vocabularies defined in other namespaces may be used to provide richer documentation.

Multilingual documentation is supported through use of the language tagging facility of RDF literals.

4. Using the Domain and Range vocabulary

This section is non-normative.

This specification introduces an RDF vocabulary for describing the meaningful use of properties and classes in RDF data. For example, an RDF vocabulary might describe limitations on the types of values that are appropriate for some property, or on the classes to which it makes sense to ascribe such properties.

RDF Schema provides a mechanism for describing this information, but does not say whether or how an application should use it. For example, while an RDF vocabulary can assert that an author property is used to indicate resources that are instances of the class Person, it does not say whether or how an application should act in processing that range information. Different applications will use this information in different ways. For example, data checking tools might use this to help discover errors in some data set, an interactive editor might suggest appropriate values, and a reasoning application might use it to infer additional information from instance data.

RDF vocabularies can describe relationships between vocabulary items from multiple independently developed vocabularies. Since IRIs are used to identify classes and properties on the Web, it is possible to create new properties that have a domain or range whose value is a class defined in another namespace.

5. Other vocabulary

Additional classes and properties, including constructs for representing containers and RDF statements, and for deploying RDF vocabulary descriptions in the World Wide Web, are defined in this section.

5.1 Container Classes and Properties

This section is non-normative.

RDF containers are resources that are used to represent collections. The same resource may appear in a container more than once. Unlike containment in the physical world, a container may be contained in itself.

Three different kinds of container are defined. Whilst the formal semantics [RDF11-MT] of all three classes of container are identical, different classes may be used to indicate informally further information. An rdf:Bag is used to indicate that the container is intended to be unordered. An rdf:Seq is used to indicate that the order indicated by the numerical order of the container membership properties of the container is intended to be significant. An rdf:Alt container is used to indicate that typical processing of the container will be to select one of the members.

Just as a hen house may have the property that it is made of wood, that does not mean that all the hens it contains are made of wood, a property of a container is not necessarily a property of all of its members.

RDF containers are defined by the following classes and properties.

5.1.1 rdfs:Container

The `rdfs:Container` class is a super-class of the RDF Container classes, i.e. `rdf:Bag`, `rdf:Seq`, `rdf:Alt`.

5.1.2 rdf:Bag

The `rdf:Bag` class is the class of RDF 'Bag' containers. It is a subclass of `rdfs:Container`. Whilst formally it is no different from an `rdf:Seq` or an `rdf:Alt`, the `rdf:Bag` class is used conventionally to indicate to a human reader that the container is intended to be unordered.

5.1.3 rdf:Seq

The `rdf:Seq` class is the class of RDF 'Sequence' containers. It is a subclass of `rdfs:Container`. Whilst formally it is no different from an `rdf:Bag` or an `rdf:Alt`, the `rdf:Seq` class is used conventionally to indicate to a human reader that the numerical ordering of the container membership properties of the container is intended to be significant.

5.1.4 rdf:Alt

The `rdf:Alt` class is the class of RDF 'Alternative' containers. It is a subclass of `rdfs:Container`. Whilst formally it is no different from an `rdf:Seq` or an `rdf:Bag`, the `rdf:Alt` class is used conventionally to indicate to a human reader that typical processing will be to select one of the members of the container. The first member of the container, i.e. the value of the `rdf:_1` property, is the default choice.

5.1.5 rdfs:ContainerMembershipProperty

The `rdfs:ContainerMembershipProperty` class has as instances the properties `rdf:_1, rdf:_2, rdf:_3 ...` that are used to state that a resource is a member of a container. `rdfs:ContainerMembershipProperty` is a subclass of `rdf:Property`. Each instance of `rdfs:ContainerMembershipProperty` is an `rdfs:subPropertyOf` the `rdfs:member` property.

Given a container C, a triple of the form:

```
C rdf:_nnn O
```

where nnn is the decimal representation of an integer greater than 0 with no leading zeros, states that O is a member of the container C.

Container membership properties may be applied to resources other than containers.

5.1.6 rdfs:member

rdfs:member is an instance of rdf:Property that is a super-property of all the container membership properties i.e. each container membership property has an rdfs:subPropertyOf relationship to the property rdfs:member.

The rdfs:domain of rdfs:member is rdfs:Resource. The rdfs:range of rdfs:member is rdfs:Resource.

5.2 RDF Collections

This section is non-normative.

RDF containers are open in the sense that the core RDF specifications define no mechanism to state that there are no more members. The RDF Collection vocabulary of classes and properties can describe a closed collection, i.e. one that can have no more members.

A collection is represented as a list of items, a representation that will be familiar to those with experience of Lisp and similar programming languages. There is a shorthand notation in the Turtle syntax specification for representing collections.

> NOTE
>
> RDFS does not require that there be only one first element of a list-like structure, or even that a list-like structure have a first element.

5.2.1 rdf:List

rdf:List is an instance of rdfs:Class that can be used to build descriptions of lists and other list-like structures.

5.2.2 rdf:first

rdf:first is an instance of rdf:Property that can be used to build descriptions of lists and other list-like structures.

A triple of the form:

```
L rdf:first O
```

states that there is a first-element relationship between L and O.

The rdfs:domain of rdf:first is rdf:List. The rdfs:range of rdf:first is rdfs:Resource.

5.2.3 rdf:rest

rdf:rest is an instance of rdf:Property that can be used to build descriptions of lists and other list-like structures.

A triple of the form:

```
L rdf:rest O
```

states that there is a rest-of-list relationship between L and O.

The rdfs:domain of rdf:rest is rdf:List. The rdfs:range of rdf:rest is rdf:List.

5.2.4 rdf:nil

The resource rdf:nil is an instance of rdf:List that can be used to represent an empty list or other list-like structure.

A triple of the form:

```
L rdf:rest rdf:nil
```

states that L is an instance of rdf:List that has one item; that item can be indicated using the rdf:first property.

5.3 Reification Vocabulary

This section is non-normative.

5.3.1 rdf:Statement

rdf:Statement is an instance of rdfs:Class. It is intended to represent the class of RDF statements. An RDF statement is the statement made by a token of an RDF triple. The subject of an RDF statement is the instance of rdfs:Resource identified by the subject of the triple. The predicate of an RDF statement is the instance of rdf:Property identified by the predicate of the triple. The object of an RDF statement is the instance of rdfs:Resource identified by the object of the triple. rdf:Statement is in the domain of the properties rdf:predicate, rdf:subject and rdf:object. Different individual rdf:Statement instances may have the same values for their rdf:predicate, rdf:subject and rdf:object properties.

5.3.2 rdf:subject

rdf:subject is an instance of rdf:Property that is used to state the subject of a statement.

A triple of the form:

```
S rdf:subject R
```

states that S is an instance of rdf:Statement and that the subject of S is R.

The rdfs:domain of rdf:subject is rdf:Statement. The rdfs:range of rdf:subject is rdfs:Resource.

5.3.3 rdf:predicate

rdf:predicate is an instance of rdf:Property that is used to state the predicate of a statement.

A triple of the form:

```
S rdf:predicate P
```

states that S is an instance of rdf:Statement, that P is an instance of rdf:Property and that the predicate of S is P.

The rdfs:domain of rdf:predicate is rdf:Statement and the rdfs:range is rdfs:Resource.

5.3.4 rdf:object

rdf:object is an instance of rdf:Property that is used to state the object of a statement.

A triple of the form:

```
S rdf:object O
```

states that S is an instance of rdf:Statement and that the object of S is O.

The rdfs:domain of rdf:object is rdf:Statement. The rdfs:range of rdf:object is rdfs:Resource.

5.4 Utility Properties

The following utility classes and properties are defined in the RDF core namespaces.

5.4.1 rdfs:seeAlso

rdfs:seeAlso is an instance of rdf:Property that is used to indicate a resource that might provide additional information about the subject resource.

A triple of the form:

```
S rdfs:seeAlso O
```

states that the resource O may provide additional information about S. It may be possible to retrieve representations of O from the Web, but this is not required. When such representations may be retrieved, no constraints are placed on the format of those representations.

The rdfs:domain of rdfs:seeAlso is rdfs:Resource. The rdfs:range of rdfs:seeAlso is rdfs:Resource.

5.4.2 rdfs:isDefinedBy

rdfs:isDefinedBy is an instance of rdf:Property that is used to indicate a resource defining the subject resource. This property may be used to indicate an RDF vocabulary in which a resource is described.

A triple of the form:

```
S rdfs:isDefinedBy O
```

states that the resource O defines S. It may be possible to retrieve representations of O from the Web, but this is not required. When such representations may be retrieved, no constraints are placed on the format of those representations. rdfs:isDefinedBy is a subproperty of rdfs:seeAlso.

The rdfs:domain of rdfs:isDefinedBy is rdfs:Resource. The rdfs:range of rdfs:isDefinedBy is rdfs:Resource.

5.4.3 rdf:value

rdf:value is an instance of rdf:Property that may be used in describing structured values.

rdf:value has no meaning on its own. It is provided as a piece of vocabulary that may be used in idioms such as illustrated in example below:

EXAMPLE 1

```
<http://www.example.com/2002/04/products#item10245>
    <http://www.example.org/terms/weight> [
       rdf:value 2.4 ;
       <http://www.example.org/terms/units> <http://www.example.org/units/kilograms>
       ] .
```

Despite the lack of formal specification of the meaning of this property, there is value in defining it to encourage the use of a common idiom in examples of this kind.

The rdfs:domain of rdf:value is rdfs:Resource. The rdfs:range of rdf:value is rdfs:Resource.

6. RDF Schema summary

This section is non-normative.

The tables in this section provide an overview of the RDF Schema vocabulary.

6.1 RDF classes

Class name	comment
rdfs:Resource	The class resource, everything.
rdfs:Literal	The class of literal values, e.g. textual strings and integers.
rdf:langString	The class of language-tagged string literal values.
rdf:HTML	The class of HTML literal values.
rdf:XMLLiteral	The class of XML literal values.
rdfs:Class	The class of classes.
rdf:Property	The class of RDF properties.
rdfs:Datatype	The class of RDF datatypes.
rdf:Statement	The class of RDF statements.
rdf:Bag	The class of unordered containers.
rdf:Seq	The class of ordered containers.
rdf:Alt	The class of containers of alternatives.
rdfs:Container	The class of RDF containers.
rdfs:ContainerMembershipProperty	The class of container membership properties, rdf:_1, rdf:_2, ..., all of which are sub-properties of 'member'.
rdf:List	The class of RDF Lists.

6.2 RDF properties

Property name	comment	domain	range
rdf:type	The subject is an instance of a class.	rdfs:Resource	rdfs:Class
rdfs:subClassOf	The subject is a subclass of a class.	rdfs:Class	rdfs:Class
rdfs:subPropertyOf	The subject is a subproperty of a property.	rdf:Property	rdf:Property
rdfs:domain	A domain of the subject property.	rdf:Property	rdfs:Class
rdfs:range	A range of the subject property.	rdf:Property	rdfs:Class
rdfs:label	A human-readable name for the subject.	rdfs:Resource	rdfs:Literal
rdfs:comment	A description of the subject resource.	rdfs:Resource	rdfs:Literal
rdfs:member	A member of the subject resource.	rdfs:Resource	rdfs:Resource
rdf:first	The first item in the subject RDF list.	rdf:List	rdfs:Resource
rdf:rest	The rest of the subject RDF list after the first item.	rdf:List	rdf:List
rdfs:seeAlso	Further information about the subject resource.	rdfs:Resource	rdfs:Resource
rdfs:isDefinedBy	The definition of the subject resource.	rdfs:Resource	rdfs:Resource
rdf:value	Idiomatic property used for structured values.	rdfs:Resource	rdfs:Resource
rdf:subject	The subject of the subject RDF statement.	rdf:Statement	rdfs:Resource
rdf:predicate	The predicate of the subject RDF statement.	rdf:Statement	rdfs:Resource
rdf:object	The object of the subject RDF statement.	rdf:Statement	rdfs:Resource

In addition to these classes and properties, RDF also uses properties called `rdf:_1`, `rdf:_2`, `rdf:_3`... etc., each of which is both a sub-property of `rdfs:member` and an instance of the class `rdfs:ContainerMembershipProperty`. There is also an instance of `rdf:List` called `rdf:nil` that is an empty `rdf:List`.

A. Acknowledgments

This section is non-normative.

The RDF Schema design was originally produced by the RDF Schema Working Group (1997-2000). The current specification is largely an editorial clarification of that design, and has benefited greatly from the hard work of the RDF Core Working Group members, and from implementation feedback from many members of the RDF Interest Group. In 2013-2014 Guus Schreiber edited this document on behalf of the RDF Working Group to bring it in line with the RDF 1.1 specifications.

David Singer of IBM was the chair of the original RDF Schema group throughout most of the development of this specification; we thank David for his efforts and thank IBM for supporting him and us in this endeavor. Particular thanks are also due to Andrew Layman for his editorial work on early versions of this specification.

The original RDF Schema Working Group membership included:

Nick Arnett (Verity), Dan Brickley (ILRT / University of Bristol), Walter Chang (Adobe), Sailesh Chutani (Oracle), Ron Daniel (DATAFUSION), Charles Frankston (Microsoft), Joe Lapp (webMethods Inc.), Patrick Gannon (CommerceNet), RV Guha (Epinions, previously of Netscape Communications), Tom Hill (Apple Computer), Renato Iannella (DSTC), Sandeep Jain (Oracle), Kevin Jones, (InterMind), Emiko Kezuka (Digital Vision Laboratories), Ora Lassila (Nokia Research Center), Andrew Layman (Microsoft), John McCarthy (Lawrence Berkeley National Laboratory), Michael Mealling (Network Solutions), Norbert Mikula (DataChannel), Eric Miller (OCLC), Frank Olken (Lawrence Berkeley National Laboratory), Sri Raghavan (Digital/Compaq), Lisa Rein (webMethods Inc.), Tsuyoshi Sakata (Digital Vision Laboratories), Leon Shklar (Pencom Web Works), David Singer (IBM), Wei (William) Song (SISU), Neel Sundaresan (IBM), Ralph Swick (W3C), Naohiko Uramoto (IBM), Charles Wicksteed (Reuters Ltd.), Misha Wolf (Reuters Ltd.)

B. Change since 2004 Recommendation

This section is non-normative.

Changes for RDF 1.1 Recommendation

- No changes.

Changes for RDF 1.1 Proposed Edited Recommendation

- Conversion to ReSpec, including formatting of examples and notes.
- References to RDF 1.0 documents where appropriate replaced by references to RDF 1.1 documents.
- Replaced the term "URI Reference" with the term "IRI".
- Removed discussion about distinction between plain and typed literals, as this distinction is absent in RDF 1.1 and has no technical bearing on RDF Schema.
- Removed the introductory paragraph of Sec. "Reification Vocabulary", as this discussion is not related to the technical content and is irrelevant and confusing now.
- Update of affiliation of the editors.
- Added RDF WG to the Acknowledgements section.
- Renamed the document from "RDF Vocabulary Description Language 1.0: RDF Schema" to "RDF Schema 1.1", as the term Vocabulary Description Language has led to confusion.
- Three paragraphs of the Introduction were left out. These paragraphs described the things that RDF Schema does not do and are now much less relevant than in 2004.
- Added the datatypes `rdf:langString` and `rdf:HTML`.
- Removed Appendix "RDF Schema in RDF/XML". It was informative, but now out of date, in terms of content and in terms of syntax.
- Marked `rdf:HTML` and `rdf:XMLLiteral` as non-normative.
- Removed references to 2004 Primer from Secs. 5.1, 5.2 and 5.4.3. In the latter case the example referred to was moved into this document for readability purposes.

C. References

C.1 Normative references

[JSON-LD]

Manu Sporny, Gregg Kellogg, Markus Lanthaler, Editors. *JSON-LD 1.0*. 16 January 2014. W3C Recommendation. URL: http://www.w3.org/TR/json-ld/

[RDF11-CONCEPTS]
Richard Cyganiak, David Wood, Markus Lanthaler. *RDF 1.1 Concepts and Abstract Syntax.* W3C Recommendation, 25 February 2014. URL: http://www.w3.org/TR/2014/REC-rdf11-concepts-20140225/. The latest edition is available at http://www.w3.org/TR/rdf11-concepts/

[RDF11-MT]
Patrick J. Hayes, Peter F. Patel-Schneider. *RDF 1.1 Semantics.* W3C Recommendation, 25 February 2014. URL: http://www.w3.org/TR/2014/REC-rdf11-mt-20140225/. The latest edition is available at http://www.w3.org/TR/rdf11-mt/

[TRIG]
Gavin Carothers, Andy Seaborne. *TriG: RDF Dataset Language*. W3C Recommendation, 25 February 2014. URL: http://www.w3.org/TR/2014/REC-trig-20140225/. The latest edition is available at http://www.w3.org/TR/trig/

[TURTLE]
Eric Prud'hommeaux, Gavin Carothers. *RDF 1.1 Turtle: Terse RDF Triple Language.* W3C Recommendation, 25 February 2014. URL: http://www.w3.org/TR/2014/REC-turtle-20140225/. The latest edition is available at http://www.w3.org/TR/turtle/

C.2 Informative references

[BERNERS-LEE98]
Tim Berners-Lee. *What the Semantic Web can represent*. 1998. URI: http://www.w3.org/DesignIssues/RDFnot.html.

[OWL2-OVERVIEW]
W3C OWL Working Group. *OWL 2 Web Ontology Language Document Overview (Second Edition)*. 11 December 2012. W3C Recommendation. URL: http://www.w3.org/TR/owl2-overview/

[RDF11-PRIMER]
Guus Schreiber, Yves Raimond. *RDF 1.1 Primer*. W3C Working Group Note, 25 February 2014. The latest version is available at http://www.w3.org/TR/rdf11-primer/.