W3C Recommendation

OWL 2 Web Ontology Language XML Serialization (Second Edition)

W3C Recommendation 11 December 2012

This version:
http://www.w3.org/TR/2012/REC-owl2-xml-serialization-20121211/
Latest version (series 2):
http://www.w3.org/TR/owl2-xml-serialization/
Latest Recommendation:
http://www.w3.org/TR/owl-xml-serialization
Previous version:
http://www.w3.org/TR/2012/PER-owl2-xml-serialization-20121018/
Editors:
Boris Motik, University of Oxford
Bijan Parsia, University of Manchester
Peter F. Patel-Schneider, Nuance Communications
Contributors: (in alphabetical order)
Sean Bechhofer, University of Manchester
Bernardo Cuenca Grau, University of Oxford
Achille Fokoue, IBM Corporation
Rinke Hoekstra, University of Amsterdam

Please refer to the **errata** for this document, which may include some normative corrections.

A color-coded version of this document showing changes made since the previous version is also available.

This document is also available in these non-normative formats: PDF version.

See also translations.

Abstract

The OWL 2 Web Ontology Language, informally OWL 2, is an ontology language for the Semantic Web with formally defined meaning. OWL 2 ontologies provide classes, properties, individuals, and data values and are stored as Semantic Web documents. OWL 2 ontologies can be used along with information written in RDF, and OWL 2 ontologies themselves are primarily exchanged as RDF documents. The OWL 2 Document Overview describes the overall state of OWL 2, and should be read before other OWL 2 documents.

This document specifies an XML serialization for OWL 2 that mirrors its structural specification. An XML schema defines this syntax and is available as a separate document, as well as being included here.

Status of this Document

May Be Superseded

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/.

Summary of Changes

There have been no substantive changes since the previous version. For details on the minor changes see the change log and color-coded diff.

Please Send Comments

Please send any comments to public-owl-comments@w3.org (public archive). Although work on this document by the OWL Working Group is complete, comments may be addressed in the errata or in future revisions. Open discussion among developers is welcome at public-owl-dev@w3.org (public archive).

Endorsed By W3C

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.

Patents

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.

Table of Contents

1 Overview

This document defines the XML serialization for OWL 2, an alternative exchange syntax for OWL 2 designed for use by XML tools (e.g., tools using, for example, XQuery [XQuery]). Although the XML serialization is designed as an exchange syntax for OWL 2, RDF/XML is the only required exchange syntax for OWL—use of the XML serialization by OWL 2 tools is optional.

The italicized keywords *must*, *must not*, *should*, *should not*, and *may* are used to specify normative features of OWL 2 documents and tools, and are interpreted as specified in RFC 2119 [*RFC 2119*].

The XML serialization mirrors the structural specification of OWL 2 [*OWL 2 Specification*] and is defined by means of an XML schema [*XML Schema*] plus some additional constraints in prose.

The elements in the XML Schema belong to the *<http://www.w3.org/2002/07/owl#>* namespace, and the attributes belong to no namespace. The local parts of the names used in the XML Schema are the same as the names of their corresponding elements from the structural specification. Thus, the XML serialization can be seen as a notational variant of the functional syntax.

As a notational variant of the functional syntax, every OWL 2 ontology serialized according to this specification can also be serialized as an RDF document in a suitable concrete syntax such as RDF/XML. A suitable XSLT stylesheet, along with GRDDL, can allow GRDDL aware software to treat documents serialized as OWL/XML as if they were serialized as RDF/XML. See OWL-XGX for an informative example of how this works.

2 Example Ontology (Informative)

Example:

The following is an example of an OWL 2 ontology written in the XML serialization.

More examples can be found in the OWL 2 Primer [*OWL 2 Primer*].

```
<?xml version="1.0" encoding="UTF-8"?>
<Ontology xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
     xsi:schemaLocation="http://www.w3.org/2002/07/owl# http://www.w3.org/2009/09/owl2-xml.xsd"
     xmlns="http://www.w3.org/2002/07/owl#"
     xml:base="http://example.com/myOntology"
     ontologyIRI="http://example.com/myOntology">

     <Prefix name="myOnt" IRI="http://example.com/myOntology#"/>
     <Import>http://example.com/someOtherOntology</Import>

     <Declaration>
          <Class IRI="#Animal"/>
     </Declaration>
     <Declaration>
          <Class abbreviatedIRI="myOnt:Tabloid"/>
     </Declaration>
     <Declaration>
          <ObjectProperty IRI="#eats"/>
     </Declaration>
     <Declaration>
          <ObjectProperty IRI="#reads"/>
     </Declaration>

     <SubClassOf>
          <Class abbreviatedIRI="myOnt:Animal"/>
          <ObjectAllValuesFrom>
               <ObjectProperty IRI="#reads"/>
               <Class IRI="#Tabloid"/>
          </ObjectAllValuesFrom>
     </SubClassOf>

</Ontology>
```

3 The Serialization Syntax

3.1 IRIs

During parsing of ontology documents written in the XML serialization of OWL 2, all values that are declared in the schema given below as being of type *xsd:anyURI must* be resolved against the respective *base IRI* as specified in the XML Base specification [*XML Base*].

In contrast, OWL 2 literals of the *xsd:anyURI* datatype *must not* be resolved against the base IRI: all literals of OWL 2 are treated as opaque values whose value is fully defined by their lexical representation (as described in Section 4.6 of the *OWL 2 Specification*]).

Ontology documents written in the XML serialization of OWL 2 may make use of abbreviated IRIs as described in Section 2.4 of the *OWL 2 Specification*]. Such ontology documents *must* declare all prefixes used in the values of *abbreviatedIRI* attributes using a *Prefix* element in that document. In any particular file, a prefix may be defined by only one Prefix element and prefix declarations are scoped to the file in which they lexically appear. Thus, prefix declarations are not imported.

On any element, one, and exactly one, of an *IRI* attribute or an *abbreviatedIRI* attribute *must* appear. This constraint is not expressed in the Schema for technical reasons.

During parsing of ontology documents written in the XML serialization of OWL 2, every *abbreviatedIRI* attribute *must* be replaced with a corresponding *IRI* attribute. The value of the *abbreviatedIRI* attribute *must* be expanded into a full IRI as described in Section 2.4 of the [*OWL 2 Specification*].

Note: The structural specification does not handle either relative IRIs or abbreviated IRIs and their attendant syntax. Thus, an API which exactly conforms to the structural syntax can handle only absolute IRIs as the identifier for OWL entities. However, implementations are free to use whatever internal representation they see fit. An implementation based on the XML DOM [*Document Object Model*] could sensibly choose to maintain the abbreviated IRI machinery so long as it also exposed an API which presented all corresponding expanded IRIs.

3.2 Imports and Global Conditions

OWL imports are not handled at the XML level, but must be handled separately.

An OWL 2 ontology written in the XML serialization of OWL 2 *must* satisfy the conditions on OWL 2 ontologies from Section 3 of the OWL 2 Specification [*OWL 2 Specification*].

An OWL 2 DL ontology written in the XML serialization of OWL 2 *must* satisfy the conditions on OWL 2 DL ontologies from Section 3 of the OWL 2 Specification [*OWL 2 Specification*]. Some of these conditions involve imported ontologies, thus it is possible for an OWL 2 DL ontology written in the XML serialization of OWL 2 to satisfy the conditions to be an OWL 2 DL ontology in a manner invisible to XML Schema checking tools since they are not sensitive to OWL imports.

3.3 Profiles

The XML schema presented here covers the entire OWL 2 structural specification, and thus includes all the features available in OWL 2 profiles [*OWL 2 Profiles*].

3.4 The XML Schema

This schema may also be downloaded directly.

```
<!DOCTYPE xsd:schema [
    <!ENTITY PN_CHARS_BASE "[A-Z]|[a-z]|[&#x00C0;-&#x00D6;]|[&#x00D8;-&#x00F6;]|[&#x00F8;-&#x02FF;]|[&#x0370;-&#x037D;]|[&#x037F;-&#x1FFF;]|[&#x200C;-&#x200D;]|[&#x2070;-&#x218F;]|[&#x2C00;-&#x2FEF;]|[&#x3001;-&#xD7FF;]|[&#xF900;-&#xFDCF;]|[&#xFDF0;-&#xFFFD;]|[&#x10000;-&#xEFFFF;]">
    <!ENTITY PN_CHARS_U    "&PN_CHARS_BASE;|_">
    <!ENTITY PN_CHARS      "&PN_CHARS_U;|\-|[0-9]|&#x00B7;|[&#x0300;-&#x036F;]|[&#x203F;-&#x2040;]">
    <!ENTITY PN_PREFIX     "(&PN_CHARS_BASE;)((&PN_CHARS;|\.)*(&PN_CHARS;  ))?">
    <!ENTITY PN_LOCAL      "(&PN_CHARS_U;|[0-9])((&PN_CHARS;|\.)*(&PN_CHARS;))?">
    <!ENTITY PNAME_NS      "(&PN_PREFIX;)?:">
    <!ENTITY PNAME_LN      "(&PNAME_NS;)(&PN_LOCAL;)">
    <!ENTITY PrefixedName  "(&PNAME_LN;)|(&PNAME_NS;)">
    ]>
<!--  From: http://www.w3.org/TR/rdf-sparql-query/#grammar
  The entities implement productions [95] (PN_CHARS_BASE), [96] (PN_CHARS_U), [98] (PN_CHARS), [99] (PN_PREFIX),
  [100] (PN_LOCAL), [71] (PNAME_NS), [72] (PNAME_LN) and  [68] (PrefixedName)

  PN_PREFIX is roughly equivalent to NCName.-->

<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:owl="http://www.w3.org/2002/07/owl#"
  targetNamespace="http://www.w3.org/2002/07/owl#" elementFormDefault="qualified"
  attributeFormDefault="unqualified">

  <xsd:import namespace="http://www.w3.org/XML/1998/namespace"
    schemaLocation="http://www.w3.org/2001/xml.xsd"/>

  <!-- The ontology -->

  <xsd:complexType name="Prefix">
    <xsd:attribute name="name" use="required">
      <xsd:simpleType>
        <xsd:restriction base="xsd:string">
          <xsd:pattern value="&PN_PREFIX;|"/>
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:attribute>
    <xsd:attribute name="IRI" type="xsd:anyURI" use="required"/>
  </xsd:complexType>
  <xsd:element name="Prefix" type="owl:Prefix"/>

  <xsd:complexType name="Import">
    <xsd:simpleContent>
      <xsd:extension base="xsd:anyURI">
        <xsd:attributeGroup ref="xml:specialAttrs"/>
      </xsd:extension>
    </xsd:simpleContent>
  </xsd:complexType>
  <xsd:element name="Import" type="owl:Import"/>

  <xsd:complexType name="Ontology">
    <xsd:sequence>
      <xsd:element ref="owl:Prefix" minOccurs="0" maxOccurs="unbounded"/>
      <xsd:element ref="owl:Import" minOccurs="0" maxOccurs="unbounded"/>
      <xsd:group ref="owl:ontologyAnnotations"/>
      <xsd:group ref="owl:Axiom" minOccurs="0" maxOccurs="unbounded"/>
    </xsd:sequence>
    <xsd:attribute name="ontologyIRI" type="xsd:anyURI" use="optional"/>
    <xsd:attribute name="versionIRI" type="xsd:anyURI" use="optional"/>
    <xsd:attributeGroup ref="xml:specialAttrs"/>
  </xsd:complexType>
  <xsd:element name="Ontology" type="owl:Ontology">
    <xsd:unique name="prefix">
      <xsd:selector xpath="owl:Prefix"/>
      <xsd:field xpath="@name"/>
    </xsd:unique>
  </xsd:element>

  <!-- Entities, anonymous individuals, and literals -->

  <!-- Note that the "Entity" group does not have a corresponding abstract type.
       This is due to the fact that XML Schema does not support multiple inheritence.
       "owl:Class" is both an entity and a class expression. The authors of this schema
       determined it was more useful to be able to retrieve "owl:Class" in such queries
       as schema(*, owl:ClassExpression).
    -->
  <xsd:group name="Entity">
    <xsd:choice>
      <xsd:element ref="owl:Class"/>
      <xsd:element ref="owl:Datatype"/>
      <xsd:element ref="owl:ObjectProperty"/>
      <xsd:element ref="owl:DataProperty"/>
      <xsd:element ref="owl:AnnotationProperty"/>
      <xsd:element ref="owl:NamedIndividual"/>
    </xsd:choice>
  </xsd:group>

  <!-- This is the type for the attribute. The complex type for the element is capitalized. -->
  <xsd:simpleType name="abbreviatedIRI">
    <xsd:restriction base="xsd:string">
      <xsd:pattern value="&PrefixedName;"/>
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:complexType name="Class">
    <xsd:complexContent>
      <xsd:extension base="owl:ClassExpression">
        <xsd:attribute name="IRI" type="xsd:anyURI" use="optional"/>
        <xsd:attribute name="abbreviatedIRI" type="owl:abbreviatedIRI" use="optional"/>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="Class" type="owl:Class"/>

  <xsd:complexType name="Datatype">
    <xsd:complexContent>
      <xsd:extension base="owl:DataRange">
        <xsd:attribute name="IRI" type="xsd:anyURI" use="optional"/>
        <xsd:attribute name="abbreviatedIRI" type="owl:abbreviatedIRI" use="optional"/>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="Datatype" type="owl:Datatype"/>

  <xsd:complexType name="ObjectProperty">
    <xsd:complexContent>
      <xsd:extension base="owl:ObjectPropertyExpression">
        <xsd:attribute name="IRI" type="xsd:anyURI" use="optional"/>
        <xsd:attribute name="abbreviatedIRI" type="owl:abbreviatedIRI" use="optional"/>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="ObjectProperty" type="owl:ObjectProperty"/>

  <xsd:complexType name="DataProperty">
    <xsd:complexContent>
      <xsd:extension base="owl:DataPropertyExpression">
        <xsd:attribute name="IRI" type="xsd:anyURI" use="optional"/>
        <xsd:attribute name="abbreviatedIRI" type="owl:abbreviatedIRI" use="optional"/>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="DataProperty" type="owl:DataProperty"/>

  <xsd:complexType name="AnnotationProperty">
    <xsd:attribute name="IRI" type="xsd:anyURI" use="optional"/>
    <xsd:attribute name="abbreviatedIRI" type="owl:abbreviatedIRI" use="optional"/>
    <xsd:attributeGroup ref="xml:specialAttrs"/>
  </xsd:complexType>
  <xsd:element name="AnnotationProperty" type="owl:AnnotationProperty"/>

  <xsd:complexType name="Individual" abstract="true">
    <xsd:attributeGroup ref="xml:specialAttrs"/>
  </xsd:complexType>
  <xsd:group name="Individual">
    <xsd:choice>
      <xsd:element ref="owl:NamedIndividual"/>
      <xsd:element ref="owl:AnonymousIndividual"/>
    </xsd:choice>
  </xsd:group>

  <xsd:complexType name="NamedIndividual">
    <xsd:complexContent>
      <xsd:extension base="owl:Individual">
        <xsd:attribute name="IRI" type="xsd:anyURI" use="optional"/>
        <xsd:attribute name="abbreviatedIRI" type="owl:abbreviatedIRI" use="optional"/>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="NamedIndividual" type="owl:NamedIndividual"/>

  <xsd:complexType name="AnonymousIndividual">
    <xsd:complexContent>
      <xsd:extension base="owl:Individual">
        <xsd:attribute name="nodeID" type="xsd:NCName" use="required"/>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="AnonymousIndividual" type="owl:AnonymousIndividual"/>

  <xsd:complexType name="Literal">
    <xsd:simpleContent>
      <xsd:extension base="xsd:string">
        <xsd:attribute name="datatypeIRI" type="xsd:anyURI"/>
        <xsd:attributeGroup ref="xml:specialAttrs"/>
      </xsd:extension>
    </xsd:simpleContent>
  </xsd:complexType>
  <xsd:element name="Literal" type="owl:Literal"/>

  <!-- Declarations -->

  <xsd:complexType name="Declaration">
    <xsd:complexContent>
      <xsd:extension base="owl:Axiom">
        <xsd:sequence>
          <xsd:group ref="owl:Entity"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="Declaration" type="owl:Declaration"/>

  <!-- Object property expressions -->

  <xsd:complexType name="ObjectPropertyExpression" abstract="true">
    <xsd:attributeGroup ref="xml:specialAttrs"/>
  </xsd:complexType>
  <xsd:group name="ObjectPropertyExpression">
    <xsd:choice>
      <xsd:element ref="owl:ObjectProperty"/>
      <xsd:element ref="owl:ObjectInverseOf"/>
    </xsd:choice>
  </xsd:group>

  <xsd:complexType name="ObjectInverseOf">
    <xsd:complexContent>
      <xsd:extension base="owl:ObjectPropertyExpression">
        <xsd:sequence>
          <xsd:element ref="owl:ObjectProperty"/>
        </xsd:sequence>
      </xsd:extension>
```

```
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="ObjectInverseOf" type="owl:ObjectInverseOf"/>

  <!-- Data property expressions -->

  <xsd:complexType name="DataPropertyExpression" abstract="true">
    <xsd:attributeGroup ref="xml:specialAttrs"/>
  </xsd:complexType>
  <xsd:group name="DataPropertyExpression">
    <xsd:sequence>
      <xsd:element ref="owl:DataProperty"/>
    </xsd:sequence>
  </xsd:group>

  <!-- Data ranges -->

  <xsd:complexType name="DataRange" abstract="true">
    <xsd:attributeGroup ref="xml:specialAttrs"/>
  </xsd:complexType>
  <xsd:group name="DataRange">
    <xsd:choice>
      <xsd:element ref="owl:Datatype"/>
      <xsd:element ref="owl:DataIntersectionOf"/>
      <xsd:element ref="owl:DataUnionOf"/>
      <xsd:element ref="owl:DataComplementOf"/>
      <xsd:element ref="owl:DataOneOf"/>
      <xsd:element ref="owl:DatatypeRestriction"/>
    </xsd:choice>
  </xsd:group>

  <xsd:complexType name="DataIntersectionOf">
    <xsd:complexContent>
      <xsd:extension base="owl:DataRange">
        <xsd:sequence>
          <xsd:group ref="owl:DataRange" minOccurs="2" maxOccurs="unbounded"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="DataIntersectionOf" type="owl:DataIntersectionOf"/>

  <xsd:complexType name="DataUnionOf">
    <xsd:complexContent>
      <xsd:extension base="owl:DataRange">
        <xsd:sequence>
          <xsd:group ref="owl:DataRange" minOccurs="2" maxOccurs="unbounded"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="DataUnionOf" type="owl:DataUnionOf"/>

  <xsd:complexType name="DataComplementOf">
    <xsd:complexContent>
      <xsd:extension base="owl:DataRange">
        <xsd:sequence>
          <xsd:group ref="owl:DataRange"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="DataComplementOf" type="owl:DataComplementOf"/>

  <xsd:complexType name="DataOneOf">
    <xsd:complexContent>
      <xsd:extension base="owl:DataRange">
        <xsd:sequence>
          <xsd:element ref="owl:Literal" minOccurs="1" maxOccurs="unbounded"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="DataOneOf" type="owl:DataOneOf"/>

  <xsd:complexType name="DatatypeRestriction">
    <xsd:complexContent>
      <xsd:extension base="owl:DataRange">
        <xsd:sequence>
          <xsd:element ref="owl:Datatype"/>
          <xsd:element name="FacetRestriction" type="owl:FacetRestriction" minOccurs="1"
            maxOccurs="unbounded"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="DatatypeRestriction" type="owl:DatatypeRestriction"/>

  <xsd:complexType name="FacetRestriction">
    <xsd:sequence>
      <xsd:element ref="owl:Literal"/>
    </xsd:sequence>
    <xsd:attribute name="facet" type="xsd:anyURI" use="required"/>
    <xsd:attributeGroup ref="xml:specialAttrs"/>
  </xsd:complexType>

  <!-- Class expressions -->

  <xsd:complexType name="ClassExpression" abstract="true">
    <xsd:attributeGroup ref="xml:specialAttrs"/>
  </xsd:complexType>
  <xsd:group name="ClassExpression">
    <xsd:choice>
      <xsd:element ref="owl:Class"/>
      <xsd:element ref="owl:ObjectIntersectionOf"/>
      <xsd:element ref="owl:ObjectUnionOf"/>
      <xsd:element ref="owl:ObjectComplementOf"/>
      <xsd:element ref="owl:ObjectOneOf"/>
      <xsd:element ref="owl:ObjectSomeValuesFrom"/>
      <xsd:element ref="owl:ObjectAllValuesFrom"/>
      <xsd:element ref="owl:ObjectHasValue"/>
      <xsd:element ref="owl:ObjectHasSelf"/>
      <xsd:element ref="owl:ObjectMinCardinality"/>
      <xsd:element ref="owl:ObjectMaxCardinality"/>
      <xsd:element ref="owl:ObjectExactCardinality"/>
      <xsd:element ref="owl:DataSomeValuesFrom"/>
      <xsd:element ref="owl:DataAllValuesFrom"/>
      <xsd:element ref="owl:DataHasValue"/>
      <xsd:element ref="owl:DataMinCardinality"/>
      <xsd:element ref="owl:DataMaxCardinality"/>
      <xsd:element ref="owl:DataExactCardinality"/>
    </xsd:choice>
  </xsd:group>

  <xsd:complexType name="ObjectIntersectionOf">
    <xsd:complexContent>
      <xsd:extension base="owl:ClassExpression">
        <xsd:sequence>
          <xsd:group ref="owl:ClassExpression" minOccurs="2" maxOccurs="unbounded"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="ObjectIntersectionOf" type="owl:ObjectIntersectionOf"/>

  <xsd:complexType name="ObjectUnionOf">
    <xsd:complexContent>
      <xsd:extension base="owl:ClassExpression">
        <xsd:sequence>
          <xsd:group ref="owl:ClassExpression" minOccurs="2" maxOccurs="unbounded"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="ObjectUnionOf" type="owl:ObjectUnionOf"/>

  <xsd:complexType name="ObjectComplementOf">
    <xsd:complexContent>
      <xsd:extension base="owl:ClassExpression">
        <xsd:sequence>
          <xsd:group ref="owl:ClassExpression"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="ObjectComplementOf" type="owl:ObjectComplementOf"/>

  <xsd:complexType name="ObjectOneOf">
    <xsd:complexContent>
      <xsd:extension base="owl:ClassExpression">
        <xsd:sequence>
          <xsd:group ref="owl:Individual" minOccurs="1" maxOccurs="unbounded"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="ObjectOneOf" type="owl:ObjectOneOf"/>

  <xsd:complexType name="ObjectSomeValuesFrom">
    <xsd:complexContent>
      <xsd:extension base="owl:ClassExpression">
        <xsd:sequence>
          <xsd:group ref="owl:ObjectPropertyExpression"/>
          <xsd:group ref="owl:ClassExpression"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="ObjectSomeValuesFrom" type="owl:ObjectSomeValuesFrom"/>

  <xsd:complexType name="ObjectAllValuesFrom">
    <xsd:complexContent>
      <xsd:extension base="owl:ClassExpression">
        <xsd:sequence>
          <xsd:group ref="owl:ObjectPropertyExpression"/>
          <xsd:group ref="owl:ClassExpression"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="ObjectAllValuesFrom" type="owl:ObjectAllValuesFrom"/>

  <xsd:complexType name="ObjectHasValue">
    <xsd:complexContent>
      <xsd:extension base="owl:ClassExpression">
        <xsd:sequence>
          <xsd:group ref="owl:ObjectPropertyExpression"/>
          <xsd:group ref="owl:Individual"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="ObjectHasValue" type="owl:ObjectHasValue"/>

  <xsd:complexType name="ObjectHasSelf">
    <xsd:complexContent>
      <xsd:extension base="owl:ClassExpression">
        <xsd:sequence>
          <xsd:group ref="owl:ObjectPropertyExpression"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
```

```
  <xsd:element name="ObjectHasSelf" type="owl:ObjectHasSelf"/>

  <xsd:complexType name="ObjectMinCardinality">
    <xsd:complexContent>
      <xsd:extension base="owl:ClassExpression">
        <xsd:sequence>
          <xsd:group ref="owl:ObjectPropertyExpression"/>
          <xsd:group ref="owl:ClassExpression" minOccurs="0" maxOccurs="1"/>
        </xsd:sequence>
        <xsd:attribute name="cardinality" type="xsd:nonNegativeInteger" use="required"/>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="ObjectMinCardinality" type="owl:ObjectMinCardinality"/>

  <xsd:complexType name="ObjectMaxCardinality">
    <xsd:complexContent>
      <xsd:extension base="owl:ClassExpression">
        <xsd:sequence>
          <xsd:group ref="owl:ObjectPropertyExpression"/>
          <xsd:group ref="owl:ClassExpression" minOccurs="0" maxOccurs="1"/>
        </xsd:sequence>
        <xsd:attribute name="cardinality" type="xsd:nonNegativeInteger" use="required"/>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="ObjectMaxCardinality" type="owl:ObjectMaxCardinality"/>

  <xsd:complexType name="ObjectExactCardinality">
    <xsd:complexContent>
      <xsd:extension base="owl:ClassExpression">
        <xsd:sequence>
          <xsd:group ref="owl:ObjectPropertyExpression"/>
          <xsd:group ref="owl:ClassExpression" minOccurs="0" maxOccurs="1"/>
        </xsd:sequence>
        <xsd:attribute name="cardinality" type="xsd:nonNegativeInteger" use="required"/>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="ObjectExactCardinality" type="owl:ObjectExactCardinality"/>

  <xsd:complexType name="DataSomeValuesFrom">
    <xsd:complexContent>
      <xsd:extension base="owl:ClassExpression">
        <xsd:sequence>
          <xsd:group ref="owl:DataPropertyExpression" minOccurs="1" maxOccurs="unbounded"/>
          <xsd:group ref="owl:DataRange"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="DataSomeValuesFrom" type="owl:DataSomeValuesFrom"/>

  <xsd:complexType name="DataAllValuesFrom">
    <xsd:complexContent>
      <xsd:extension base="owl:ClassExpression">
        <xsd:sequence>
          <xsd:group ref="owl:DataPropertyExpression" minOccurs="1" maxOccurs="unbounded"/>
          <xsd:group ref="owl:DataRange"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="DataAllValuesFrom" type="owl:DataAllValuesFrom"/>

  <xsd:complexType name="DataHasValue">
    <xsd:complexContent>
      <xsd:extension base="owl:ClassExpression">
        <xsd:sequence>
          <xsd:group ref="owl:DataPropertyExpression"/>
          <xsd:element ref="owl:Literal"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="DataHasValue" type="owl:DataHasValue"/>

  <xsd:complexType name="DataMinCardinality">
    <xsd:complexContent>
      <xsd:extension base="owl:ClassExpression">
        <xsd:sequence>
          <xsd:group ref="owl:DataPropertyExpression"/>
          <xsd:group ref="owl:DataRange" minOccurs="0" maxOccurs="1"/>
        </xsd:sequence>
        <xsd:attribute name="cardinality" type="xsd:nonNegativeInteger" use="required"/>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="DataMinCardinality" type="owl:DataMinCardinality"/>

  <xsd:complexType name="DataMaxCardinality">
    <xsd:complexContent>
      <xsd:extension base="owl:ClassExpression">
        <xsd:sequence>
          <xsd:group ref="owl:DataPropertyExpression"/>
          <xsd:group ref="owl:DataRange" minOccurs="0" maxOccurs="1"/>
        </xsd:sequence>
        <xsd:attribute name="cardinality" type="xsd:nonNegativeInteger" use="required"/>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="DataMaxCardinality" type="owl:DataMaxCardinality"/>

  <xsd:complexType name="DataExactCardinality">
    <xsd:complexContent>
      <xsd:extension base="owl:ClassExpression">
        <xsd:sequence>
          <xsd:group ref="owl:DataPropertyExpression"/>
          <xsd:group ref="owl:DataRange" minOccurs="0" maxOccurs="1"/>
        </xsd:sequence>
        <xsd:attribute name="cardinality" type="xsd:nonNegativeInteger" use="required"/>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="DataExactCardinality" type="owl:DataExactCardinality"/>

  <!-- Axioms -->

  <xsd:complexType name="Axiom" abstract="true">
    <xsd:sequence>
      <xsd:group ref="owl:axiomAnnotations"/>
    </xsd:sequence>
    <xsd:attributeGroup ref="xml:specialAttrs"/>
  </xsd:complexType>
  <xsd:group name="Axiom">
    <xsd:choice>
      <xsd:element ref="owl:Declaration"/>
      <xsd:group ref="owl:ClassAxiom"/>
      <xsd:group ref="owl:ObjectPropertyAxiom"/>
      <xsd:group ref="owl:DataPropertyAxiom"/>
      <xsd:element ref="owl:DatatypeDefinition"/>
      <xsd:element ref="owl:HasKey"/>
      <xsd:group ref="owl:Assertion"/>
      <xsd:group ref="owl:AnnotationAxiom"/>
    </xsd:choice>
  </xsd:group>

  <!-- Class expression axioms -->

  <xsd:complexType name="ClassAxiom" abstract="true">
    <xsd:complexContent>
      <xsd:extension base="owl:Axiom"/>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:group name="ClassAxiom">
    <xsd:choice>
      <xsd:element ref="owl:SubClassOf"/>
      <xsd:element ref="owl:EquivalentClasses"/>
      <xsd:element ref="owl:DisjointClasses"/>
      <xsd:element ref="owl:DisjointUnion"/>
    </xsd:choice>
  </xsd:group>

  <xsd:complexType name="SubClassOf">
    <xsd:complexContent>
      <xsd:extension base="owl:ClassAxiom">
        <xsd:sequence>
          <xsd:group ref="owl:ClassExpression"/>
          <!-- This is the subexpression -->
          <xsd:group ref="owl:ClassExpression"/>
          <!-- This is the superexpression -->
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="SubClassOf" type="owl:SubClassOf"/>

  <xsd:complexType name="EquivalentClasses">
    <xsd:complexContent>
      <xsd:extension base="owl:ClassAxiom">
        <xsd:sequence>
          <xsd:group ref="owl:ClassExpression" minOccurs="2" maxOccurs="unbounded"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="EquivalentClasses" type="owl:EquivalentClasses"/>

  <xsd:complexType name="DisjointClasses">
    <xsd:complexContent>
      <xsd:extension base="owl:ClassAxiom">
        <xsd:sequence>
          <xsd:group ref="owl:ClassExpression" minOccurs="2" maxOccurs="unbounded"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="DisjointClasses" type="owl:DisjointClasses"/>

  <xsd:complexType name="DisjointUnion">
    <xsd:complexContent>
      <xsd:extension base="owl:ClassAxiom">
        <xsd:sequence>
          <xsd:element ref="owl:Class"/>
          <xsd:group ref="owl:ClassExpression" minOccurs="2" maxOccurs="unbounded"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="DisjointUnion" type="owl:DisjointUnion"/>

  <!-- Object property axioms -->

  <xsd:complexType name="ObjectPropertyAxiom" abstract="true">
    <xsd:complexContent>
      <xsd:extension base="owl:Axiom"/>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:group name="ObjectPropertyAxiom">
    <xsd:choice>
      <xsd:element ref="owl:SubObjectPropertyOf"/>
      <xsd:element ref="owl:EquivalentObjectProperties"/>
      <xsd:element ref="owl:DisjointObjectProperties"/>
```

```
      <xsd:element ref="owl:InverseObjectProperties"/>
      <xsd:element ref="owl:ObjectPropertyDomain"/>
      <xsd:element ref="owl:ObjectPropertyRange"/>
      <xsd:element ref="owl:FunctionalObjectProperty"/>
      <xsd:element ref="owl:InverseFunctionalObjectProperty"/>
      <xsd:element ref="owl:ReflexiveObjectProperty"/>
      <xsd:element ref="owl:IrreflexiveObjectProperty"/>
      <xsd:element ref="owl:SymmetricObjectProperty"/>
      <xsd:element ref="owl:AsymmetricObjectProperty"/>
      <xsd:element ref="owl:TransitiveObjectProperty"/>
    </xsd:choice>
  </xsd:group>

  <xsd:complexType name="SubObjectPropertyOf">
    <xsd:complexContent>
      <xsd:extension base="owl:ObjectPropertyAxiom">
        <xsd:sequence>
          <xsd:choice>
            <!-- This is the subproperty expression or the property chain -->
            <xsd:group ref="owl:ObjectPropertyExpression"/>
            <xsd:element name="ObjectPropertyChain" type="owl:ObjectPropertyChain"/>
          </xsd:choice>
          <xsd:group ref="owl:ObjectPropertyExpression"/>
          <!-- This is the superproperty expression -->
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="SubObjectPropertyOf" type="owl:SubObjectPropertyOf"/>

  <xsd:complexType name="ObjectPropertyChain">
    <xsd:sequence>
      <xsd:group ref="owl:ObjectPropertyExpression" minOccurs="2" maxOccurs="unbounded"/>
    </xsd:sequence>
    <xsd:attributeGroup ref="xml:specialAttrs"/>
  </xsd:complexType>

  <xsd:complexType name="EquivalentObjectProperties">
    <xsd:complexContent>
      <xsd:extension base="owl:ObjectPropertyAxiom">
        <xsd:sequence>
          <xsd:group ref="owl:ObjectPropertyExpression" minOccurs="2" maxOccurs="unbounded"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="EquivalentObjectProperties" type="owl:EquivalentObjectProperties"/>

  <xsd:complexType name="DisjointObjectProperties">
    <xsd:complexContent>
      <xsd:extension base="owl:ObjectPropertyAxiom">
        <xsd:sequence>
          <xsd:group ref="owl:ObjectPropertyExpression" minOccurs="2" maxOccurs="unbounded"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="DisjointObjectProperties" type="owl:DisjointObjectProperties"/>

  <xsd:complexType name="ObjectPropertyDomain">
    <xsd:complexContent>
      <xsd:extension base="owl:ObjectPropertyAxiom">
        <xsd:sequence>
          <xsd:group ref="owl:ObjectPropertyExpression"/>
          <xsd:group ref="owl:ClassExpression"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="ObjectPropertyDomain" type="owl:ObjectPropertyDomain"/>

  <xsd:complexType name="ObjectPropertyRange">
    <xsd:complexContent>
      <xsd:extension base="owl:ObjectPropertyAxiom">
        <xsd:sequence>
          <xsd:group ref="owl:ObjectPropertyExpression"/>
          <xsd:group ref="owl:ClassExpression"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="ObjectPropertyRange" type="owl:ObjectPropertyRange"/>

  <xsd:complexType name="InverseObjectProperties">
    <xsd:complexContent>
      <xsd:extension base="owl:ObjectPropertyAxiom">
        <xsd:sequence>
          <xsd:group ref="owl:ObjectPropertyExpression" minOccurs="2" maxOccurs="2"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="InverseObjectProperties" type="owl:InverseObjectProperties"/>

  <xsd:complexType name="FunctionalObjectProperty">
    <xsd:complexContent>
      <xsd:extension base="owl:ObjectPropertyAxiom">
        <xsd:sequence>
          <xsd:group ref="owl:ObjectPropertyExpression"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="FunctionalObjectProperty" type="owl:FunctionalObjectProperty"/>

  <xsd:complexType name="InverseFunctionalObjectProperty">
    <xsd:complexContent>
      <xsd:extension base="owl:ObjectPropertyAxiom">
        <xsd:sequence>
          <xsd:group ref="owl:ObjectPropertyExpression"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="InverseFunctionalObjectProperty" type="owl:InverseFunctionalObjectProperty"/>

  <xsd:complexType name="ReflexiveObjectProperty">
    <xsd:complexContent>
      <xsd:extension base="owl:ObjectPropertyAxiom">
        <xsd:sequence>
          <xsd:group ref="owl:ObjectPropertyExpression"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="ReflexiveObjectProperty" type="owl:ReflexiveObjectProperty"/>

  <xsd:complexType name="IrreflexiveObjectProperty">
    <xsd:complexContent>
      <xsd:extension base="owl:ObjectPropertyAxiom">
        <xsd:sequence>
          <xsd:group ref="owl:ObjectPropertyExpression"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="IrreflexiveObjectProperty" type="owl:IrreflexiveObjectProperty"/>

  <xsd:complexType name="SymmetricObjectProperty">
    <xsd:complexContent>
      <xsd:extension base="owl:ObjectPropertyAxiom">
        <xsd:sequence>
          <xsd:group ref="owl:ObjectPropertyExpression"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="SymmetricObjectProperty" type="owl:SymmetricObjectProperty"/>

  <xsd:complexType name="AsymmetricObjectProperty">
    <xsd:complexContent>
      <xsd:extension base="owl:ObjectPropertyAxiom">
        <xsd:sequence>
          <xsd:group ref="owl:ObjectPropertyExpression"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="AsymmetricObjectProperty" type="owl:AsymmetricObjectProperty"/>

  <xsd:complexType name="TransitiveObjectProperty">
    <xsd:complexContent>
      <xsd:extension base="owl:ObjectPropertyAxiom">
        <xsd:sequence>
          <xsd:group ref="owl:ObjectPropertyExpression"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="TransitiveObjectProperty" type="owl:TransitiveObjectProperty"/>

  <!-- Data property axioms -->

  <xsd:complexType name="DataPropertyAxiom" abstract="true">
    <xsd:complexContent>
      <xsd:extension base="owl:Axiom"/>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:group name="DataPropertyAxiom">
    <xsd:choice>
      <xsd:element ref="owl:SubDataPropertyOf"/>
      <xsd:element ref="owl:EquivalentDataProperties"/>
      <xsd:element ref="owl:DisjointDataProperties"/>
      <xsd:element ref="owl:DataPropertyDomain"/>
      <xsd:element ref="owl:DataPropertyRange"/>
      <xsd:element ref="owl:FunctionalDataProperty"/>
    </xsd:choice>
  </xsd:group>

  <xsd:complexType name="SubDataPropertyOf">
    <xsd:complexContent>
      <xsd:extension base="owl:DataPropertyAxiom">
        <xsd:sequence>
          <xsd:group ref="owl:DataPropertyExpression"/>
          <!-- This is the subproperty expression -->
          <xsd:group ref="owl:DataPropertyExpression"/>
          <!-- This is the superproperty expression -->
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="SubDataPropertyOf" type="owl:SubDataPropertyOf"/>

  <xsd:complexType name="EquivalentDataProperties">
    <xsd:complexContent>
      <xsd:extension base="owl:DataPropertyAxiom">
        <xsd:sequence>
          <xsd:group ref="owl:DataPropertyExpression" minOccurs="2" maxOccurs="unbounded"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="EquivalentDataProperties" type="owl:EquivalentDataProperties"/>
```

```
<xsd:complexType name="DisjointDataProperties">
  <xsd:complexContent>
    <xsd:extension base="owl:DataPropertyAxiom">
      <xsd:sequence>
        <xsd:group ref="owl:DataPropertyExpression" minOccurs="2" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
<xsd:element name="DisjointDataProperties" type="owl:DisjointDataProperties"/>

<xsd:complexType name="DataPropertyDomain">
  <xsd:complexContent>
    <xsd:extension base="owl:DataPropertyAxiom">
      <xsd:sequence>
        <xsd:group ref="owl:DataPropertyExpression"/>
        <xsd:group ref="owl:ClassExpression"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
<xsd:element name="DataPropertyDomain" type="owl:DataPropertyDomain"/>

<xsd:complexType name="DataPropertyRange">
  <xsd:complexContent>
    <xsd:extension base="owl:DataPropertyAxiom">
      <xsd:sequence>
        <xsd:group ref="owl:DataPropertyExpression"/>
        <xsd:group ref="owl:DataRange"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
<xsd:element name="DataPropertyRange" type="owl:DataPropertyRange"/>

<xsd:complexType name="FunctionalDataProperty">
  <xsd:complexContent>
    <xsd:extension base="owl:DataPropertyAxiom">
      <xsd:sequence>
        <xsd:group ref="owl:DataPropertyExpression"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
<xsd:element name="FunctionalDataProperty" type="owl:FunctionalDataProperty"/>

<!-- Datatype definitions -->

<xsd:complexType name="DatatypeDefinition">
  <xsd:complexContent>
    <xsd:extension base="owl:Axiom">
      <xsd:sequence>
        <xsd:element ref="owl:Datatype"/>
        <xsd:group ref="owl:DataRange"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
<xsd:element name="DatatypeDefinition" type="owl:DatatypeDefinition"/>

<!-- Key axioms -->

<xsd:complexType name="HasKey">
  <xsd:complexContent>
    <xsd:extension base="owl:Axiom">
      <xsd:sequence>
        <xsd:group ref="owl:ClassExpression"/>
        <xsd:group ref="owl:ObjectPropertyExpression" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:group ref="owl:DataPropertyExpression" minOccurs="0" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
<xsd:element name="HasKey" type="owl:HasKey"/>

<!-- Assertions -->

<xsd:complexType name="Assertion" abstract="true">
  <xsd:complexContent>
    <xsd:extension base="owl:Axiom"/>
  </xsd:complexContent>
</xsd:complexType>
<xsd:group name="Assertion">
  <xsd:choice>
    <xsd:element ref="owl:SameIndividual"/>
    <xsd:element ref="owl:DifferentIndividuals"/>
    <xsd:element ref="owl:ClassAssertion"/>
    <xsd:element ref="owl:ObjectPropertyAssertion"/>
    <xsd:element ref="owl:NegativeObjectPropertyAssertion"/>
    <xsd:element ref="owl:DataPropertyAssertion"/>
    <xsd:element ref="owl:NegativeDataPropertyAssertion"/>
  </xsd:choice>
</xsd:group>

<xsd:complexType name="SameIndividual">
  <xsd:complexContent>
    <xsd:extension base="owl:Assertion">
      <xsd:sequence>
        <xsd:group ref="owl:Individual" minOccurs="2" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
<xsd:element name="SameIndividual" type="owl:SameIndividual"/>

<xsd:complexType name="DifferentIndividuals">
  <xsd:complexContent>
    <xsd:extension base="owl:Assertion">
      <xsd:sequence>
        <xsd:group ref="owl:Individual" minOccurs="2" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
<xsd:element name="DifferentIndividuals" type="owl:DifferentIndividuals"/>

<xsd:complexType name="ClassAssertion">
  <xsd:complexContent>
    <xsd:extension base="owl:Assertion">
      <xsd:sequence>
        <xsd:group ref="owl:ClassExpression"/>
        <xsd:group ref="owl:Individual"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
<xsd:element name="ClassAssertion" type="owl:ClassAssertion"/>

<xsd:complexType name="ObjectPropertyAssertion">
  <xsd:complexContent>
    <xsd:extension base="owl:Assertion">
      <xsd:sequence>
        <xsd:group ref="owl:ObjectPropertyExpression"/>
        <xsd:group ref="owl:Individual"/>
        <!-- This is the source invididual  -->
        <xsd:group ref="owl:Individual"/>
        <!-- This is the target individual -->
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
<xsd:element name="ObjectPropertyAssertion" type="owl:ObjectPropertyAssertion"/>

<xsd:complexType name="NegativeObjectPropertyAssertion">
  <xsd:complexContent>
    <xsd:extension base="owl:Assertion">
      <xsd:sequence>
        <xsd:group ref="owl:ObjectPropertyExpression"/>
        <xsd:group ref="owl:Individual"/>
        <!-- This is the source invididual  -->
        <xsd:group ref="owl:Individual"/>
        <!-- This is the target individual -->
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
<xsd:element name="NegativeObjectPropertyAssertion" type="owl:NegativeObjectPropertyAssertion"/>

<xsd:complexType name="DataPropertyAssertion">
  <xsd:complexContent>
    <xsd:extension base="owl:Assertion">
      <xsd:sequence>
        <xsd:group ref="owl:DataPropertyExpression"/>
        <xsd:group ref="owl:Individual"/>
        <!-- This is the source invididual  -->
        <xsd:element ref="owl:Literal"/>
        <!-- This is the target individual -->
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
<xsd:element name="DataPropertyAssertion" type="owl:DataPropertyAssertion"/>

<xsd:complexType name="NegativeDataPropertyAssertion">
  <xsd:complexContent>
    <xsd:extension base="owl:Assertion">
      <xsd:sequence>
        <xsd:group ref="owl:DataPropertyExpression"/>
        <xsd:group ref="owl:Individual"/>
        <!-- This is the source invididual  -->
        <xsd:element ref="owl:Literal"/>
        <!-- This is the target individual -->
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
<xsd:element name="NegativeDataPropertyAssertion" type="owl:NegativeDataPropertyAssertion"/>

<!-- Annotations  -->

<xsd:complexType name="IRI">
  <xsd:simpleContent>
    <xsd:extension base="xsd:anyURI">
      <xsd:attributeGroup ref="xml:specialAttrs"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
<xsd:element name="IRI" type="owl:IRI"/>

<xsd:complexType name="AbbreviatedIRI">
  <xsd:simpleContent>
    <xsd:extension base="owl:abbreviatedIRI">
      <xsd:attributeGroup ref="xml:specialAttrs"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
<xsd:element name="AbbreviatedIRI" type="owl:AbbreviatedIRI"/>

<xsd:group name="AnnotationSubject">
  <xsd:choice>
    <xsd:element ref="owl:IRI"/>
    <xsd:element ref="owl:AbbreviatedIRI"/>
```

```
      <xsd:element ref="owl:AnonymousIndividual"/>
    </xsd:choice>
  </xsd:group>

  <xsd:group name="AnnotationValue">
    <xsd:choice>
      <xsd:element ref="owl:IRI"/>
      <xsd:element ref="owl:AbbreviatedIRI"/>
      <xsd:element ref="owl:AnonymousIndividual"/>
      <xsd:element ref="owl:Literal"/>
    </xsd:choice>
  </xsd:group>

  <xsd:complexType name="Annotation">
    <xsd:sequence>
      <xsd:group ref="owl:annotationAnnotations"/>
      <xsd:element ref="owl:AnnotationProperty"/>
      <xsd:group ref="owl:AnnotationValue"/>
    </xsd:sequence>
    <xsd:attributeGroup ref="xml:specialAttrs"/>
  </xsd:complexType>
  <xsd:element name="Annotation" type="owl:Annotation"/>

  <xsd:group name="axiomAnnotations">
    <xsd:sequence>
      <xsd:element ref="owl:Annotation" minOccurs="0" maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:group>

  <xsd:group name="ontologyAnnotations">
    <xsd:sequence>
      <xsd:element ref="owl:Annotation" minOccurs="0" maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:group>

  <xsd:group name="annotationAnnotations">
    <xsd:sequence>
      <xsd:element ref="owl:Annotation" minOccurs="0" maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:group>

  <!-- Annotation axioms -->

  <xsd:complexType name="AnnotationAxiom" abstract="true">
    <xsd:complexContent>
      <xsd:extension base="owl:Axiom"/>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:group name="AnnotationAxiom">
    <xsd:choice>
      <xsd:element ref="owl:AnnotationAssertion"/>
      <xsd:element ref="owl:SubAnnotationPropertyOf"/>
      <xsd:element ref="owl:AnnotationPropertyDomain"/>
      <xsd:element ref="owl:AnnotationPropertyRange"/>
    </xsd:choice>
  </xsd:group>

  <xsd:complexType name="AnnotationAssertion">
    <xsd:complexContent>
      <xsd:extension base="owl:AnnotationAxiom">
        <xsd:sequence>
          <xsd:element ref="owl:AnnotationProperty"/>
          <xsd:group ref="owl:AnnotationSubject"/>
          <xsd:group ref="owl:AnnotationValue"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="AnnotationAssertion" type="owl:AnnotationAssertion"/>

  <xsd:complexType name="SubAnnotationPropertyOf">
    <xsd:complexContent>
      <xsd:extension base="owl:AnnotationAxiom">
        <xsd:sequence>
          <xsd:element ref="owl:AnnotationProperty"/>
          <!-- This is the subproperty -->
          <xsd:element ref="owl:AnnotationProperty"/>
          <!-- This is the superproperty -->
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="SubAnnotationPropertyOf" type="owl:SubAnnotationPropertyOf"/>

  <xsd:complexType name="AnnotationPropertyDomain">
    <xsd:complexContent>
      <xsd:extension base="owl:AnnotationAxiom">
        <xsd:sequence>
          <xsd:element ref="owl:AnnotationProperty"/>
          <xsd:choice>
            <xsd:element ref="owl:IRI"/>
            <xsd:element ref="owl:AbbreviatedIRI"/>
          </xsd:choice>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="AnnotationPropertyDomain" type="owl:AnnotationPropertyDomain"/>

  <xsd:complexType name="AnnotationPropertyRange">
    <xsd:complexContent>
      <xsd:extension base="owl:AnnotationAxiom">
        <xsd:sequence>
          <xsd:element ref="owl:AnnotationProperty"/>
          <xsd:choice>
            <xsd:element ref="owl:IRI"/>
            <xsd:element ref="owl:AbbreviatedIRI"/>
          </xsd:choice>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="AnnotationPropertyRange" type="owl:AnnotationPropertyRange"/>

</xsd:schema>
```

4 OWL 2 XML serialization ontology document

An *OWL 2 XML serialization ontology document* is a sequence of Unicode [*UNICODE*] characters accessible from some IRI by means of the standard protocols that can be parsed into an XML document that conforms to the XML schema defined in this document and adheres to the constraints described in Section 3 of this document.

5 Appendix: The Derivation from the Functional Syntax (Informative)

The XML schema has been obtained by a straightforward translation of the structural specification of the OWL 2 Specification [*OWL 2 Specification*] in the following way:

- Each UML class that is intended to be instantiated is mapped to a global element, whose elements and attributes correspond to the components of the UML class. Each such element has an XML Schema type with the same name.
- Each UML class that is not intended to be instantiated directly, but instead gathers together commonalities, is mapped to a global element group, whose choice members correspond to the children of the UML class. For all except a select few, there is a corresponding global, abstract XML Schema type with the same name. The particles in the content model of the group are mapped into the corresponding subtypes of the group type.

Since XML Schema's type system does not support multiple inheritance, some abstract UML classes cannot be directly mapped into an XML Schema type hierarchy with the intended result. In the case where there are multiple parent classes, only the most useful parents are mapped into the active type hierarchy. The excluded groups are:

- Entity
- AnnotationSubject
- AnnotationValue

Some groups are mere documentation in the schema, and therefore are not included as types:

- axiomAnnotations
- ontologyAnnotations
- annotationAnnotations

The XML schema thus captures the structure of OWL 2 entities, expressions, and axioms. Not all XML documents which are legal according to this schema correspond to structural correct OWL ontologies.

To get to the OWL Ontology an OWL 2 XML serialization ontology document describes one must:

- Resolve all the IRIs and expand the abbreviated IRIs in the above described way.
- Get the imports closure of the ontology.

To determine whether the OWL Ontology is structurally correct, one must:

- Check the global constraints on axioms.
- Check the typing constraints.

Each axiom in the XML syntax of OWL 2 contains complete information about the type of all the entities in it. Therefore the OWL 2 XML Syntax parsing process is simpler than the canonical parsing process from Section 3.6 of OWL 2 Specification [*OWL 2 Specification*].

6 Appendix: Internet Media Type, File Extension, and Macintosh File Type

Contact
Ivan Herman / Sandro Hawke
See also
How to Register a Media Type for a W3C Specification [*Register MIME*] and Internet Media Type registration, consistency of use [*MIME Consistency*].

The Internet Media Type / MIME Type for the OWL XML Serialization is `application/owl+xml`.

It is recommended that OWL XML Serialization files have the extension `.owx` (all lowercase) on all platforms.

It is recommended that OWL XML Serialization files stored on Macintosh HFS file systems be given a file type of TEXT.

The information that follows will be submitted to the IESG for review, approval, and registration with IANA.

Type name
application
Subtype name
owl+xml
Required parameters
None
Optional parameters
charset This parameter may be required when transfering non-ASCII data across some protocols.
Encoding considerations
The syntax of the OWL XML Serialization is expressed over code points in Unicode [*UNICODE*].
Security considerations
The OWL XML Serialization uses IRIs as term identifiers. Applications interpreting data expressed in the OWL XML Serialization should address the security issues of Internationalized Resource Identifiers (IRIs) [*RFC 3987*] Section 8, as well as Uniform Resource Identifiers (URI): Generic Syntax [*RFC 3986*] Section 7. Multiple IRIs may have the same appearance. Characters in different scripts may look similar (a Cyrillic "o" may appear similar to a Latin "o"). A character followed by combining characters may have the same visual representation as another character (LATIN SMALL LETTER E followed by COMBINING ACUTE ACCENT has the same visual representation as LATIN SMALL LETTER E WITH ACUTE). Any person or application that is writing or interpreting data in the OWL XML Serialization must take care to use the IRI that matches the intended semantics, and avoid IRIs that may look similar. Further information about matching of similar characters can be found in Unicode Security Considerations [*UNISEC*] and Internationalized Resource Identifiers (IRIs) [*RFC 3987*] Section 8.
Interoperability considerations
There are no known interoperability issues.
Published specification
This specification.
Applications which use this media type
None at current time.
Additional information
None.
Magic number(s)
OWL XML documents are XML documents and thus may have initial strings similar to any XML document.
File extension(s)
".owx"
Base URI
As in XML.
Macintosh file type code(s)
"TEXT"
Person & email address to contact for further information
Ivan Herman, ivan@w3.org / Sandro Hawke, sandro@w3.org. Please send technical comments and questions about OWL to public-owl-comments@w3.org, a mailing list with a public archive at http://lists.w3.org/Archives/Public/public-owl-comments/
Intended usage
COMMON

Restrictions on usage
None
Author/Change controller
The OWL XML Serialization is the product of the W3C OWL Working Group; W3C reserves change control over this specification.

7 Appendix: Change Log (Informative)

7.1 Changes Since Recommendation

This section summarizes the changes to this document since the Recommendation of 27 October, 2009.

- With the publication of the XML Schema Definition Language (XSD) 1.1 Part 2: Datatypes Recommendation of 5 April 2012, the elements of OWL 2 which are based on XSD 1.1 are now considered required, and the note detailing the optional dependency on the XSD 1.1 Candidate Recommendation of 30 April, 2009 has been removed from the "Status of this Document" section.
- Minor typographical errors were corrected as detailed on the OWL 2 Errata page.

7.2 Changes Since Proposed Recommendation

This section summarizes the changes to this document since the Proposed Recommendation of 22 September, 2009.

- A note on the use of GRDDL was added to the introduction.
- An editor's note on the future definition of a GRDDL transformation mechanism was removed.
- Some minor editorial changes were made.

7.3 Changes Since Candidate Recommendation

This section summarizes the changes to this document since the Candidate Recommendation of 11 June, 2009.

- There were a few bug fixes in the schema itself (see http://www.w3.org/2007/OWL/wiki/OWL_XML_Schema).
- Some minor editorial changes were made.

7.4 Changes Since Last Call

This section summarizes the changes to this document since the Last Call Working Draft of 21 April, 2009.

- Some minor editorial changes were made.

8 Acknowledgments

The starting point for the development of OWL 2 was the OWL1.1 member submission, itself a result of user and developer feedback, and in particular of information gathered during the OWL Experiences and Directions (OWLED) Workshop series. The working group also considered postponed issues from the WebOnt Working Group.

This document has been produced by the OWL Working Group (see below), and its contents reflect extensive discussions within the Working Group as a whole. The editors extend special thanks to Kendall Clark (Clark & Parsia), Achille Fokoue (IBM Corporation) and Michael Grove (Clark & Parsia), Rinke Hoekstra (University of Amsterdam) for their thorough reviews, to Liam Quim (W3C) for his advice on XML Schema, and to Dmitry Repchevsky (Barcelona Supercomputing Centre) for catching a bug in the schema.

The regular attendees at meetings of the OWL Working Group at the time of publication of this document were: Jie Bao (RPI), Diego Calvanese (Free University of Bozen-Bolzano), Bernardo Cuenca Grau (Oxford University Computing Laboratory), Martin Dzbor (Open University), Achille Fokoue (IBM Corporation), Christine Golbreich (Université de Versailles St-Quentin and LIRMM), Sandro Hawke (W3C/MIT), Ivan Herman (W3C/ERCIM), Rinke Hoekstra (University of Amsterdam), Ian Horrocks (Oxford University Computing Laboratory), Elisa Kendall (Sandpiper Software), Markus Krötzsch (FZI), Carsten Lutz (Universität Bremen), Deborah L. McGuinness (RPI), Boris Motik (Oxford University Computing Laboratory), Jeff Pan (University of Aberdeen), Bijan Parsia (University of Manchester), Peter F. Patel-Schneider (Bell Labs Research, Alcatel-Lucent), Sebastian Rudolph (FZI), Alan Ruttenberg (Science Commons), Uli Sattler (University of Manchester), Michael Schneider (FZI), Mike Smith (Clark & Parsia), Evan Wallace (NIST), Zhe Wu (Oracle Corporation), and Antoine Zimmermann (DERI Galway). We would also like to thank past members of the working group: Jeremy Carroll, Jim Hendler, and Vipul Kashyap.

9 References

9.1 Normative References

[GRDDL]
Gleaning Resource Descriptions from Dialects of Languages (GRDDL). Dan Connolly, ed. W3C Recommendation, 11 September 2007, http://www.w3.org/TR/2007/REC-grddl-20070911/. Latest version available as http://www.w3.org/TR/grddl/.
[OWL 2 Specification]
OWL 2 Web Ontology Language: Structural Specification and Functional-Style Syntax (Second Edition) Boris Motik, Peter F. Patel-Schneider, Bijan Parsia, eds. W3C Recommendation, 11 December 2012, http://www.w3.org/TR/2012/REC-owl2-syntax-20121211/. Latest version available at http://www.w3.org/TR/owl2-syntax/.
[RFC 2119]
RFC 2119: Key words for use in RFCs to Indicate Requirement Levels. Network Working Group, S. Bradner. IETF, March 1997, http://www.ietf.org/rfc/rfc2119.txt
[UNICODE]
The Unicode Standard. The Unicode Consortium, Version 5.1.0, ISBN 0-321-48091-0, as updated from time to time by the publication of new versions. (See http://www.unicode.org/unicode/standard/versions/ for the latest version and additional information on versions of the standard and of the Unicode Character Database).
[XML Base]
XML Base (Second Edition). Jonathan Marsh and Richard Tobin, eds. W3C Recommendation, 28 January 2009, http://www.w3.org/TR/2009/REC-xmlbase-20090128/. Latest version available as http://www.w3.org/TR/xmlbase/.
[XML Schema]
W3C XML Schema Definition Language (XSD) 1.1 Part 1: Structures. Shudi Gao, C. M. Sperberg-McQueen, and Henry S. Thompson, eds. W3C Candidate Recommendation, 30 April 2009, http://www.w3.org/TR/2009/CR-xmlschema11-1-20090430/. Latest version available as http://www.w3.org/TR/xmlschema11-1/.

9.2 Nonnormative References

[Document Object Model]
Document Object Model (DOM) Technical Reports
[MIME Consistency]
Internet Media Type registration, consistency of use. Tim Bray, ed. W3C TAG Finding, 30 April 2004.
[OWL 2 Primer]
OWL 2 Web Ontology Language: Primer (Second Edition) Pascal Hitzler, Markus Krötzsch, Bijan Parsia, Peter F. Patel-Schneider, Sebastian Rudolph, eds. W3C Recommendation, 11 December 2012, http://www.w3.org/TR/2012/REC-owl2-primer-20121211/. Latest version available at http://www.w3.org/TR/owl2-primer/.
[OWL 2 Profiles]
OWL 2 Web Ontology Language: Profiles (Second Edition) Boris Motik, Bernardo Cuenca Grau, Ian Horrocks, Zhe Wu, Achille Fokoue, Carsten Lutz, eds. W3C Recommendation, 11 December 2012, http://www.w3.org/TR/2012/REC-owl2-profiles-20121211/. Latest version available at http://www.w3.org/TR/owl2-profiles/.
[Register MIME]
Register an Internet Media Type for a W3C Spec. Philippe Le Hégaret, ed. W3C Guidebook.
[RFC 3986]
RFC 3986: Uniform Resource Identifier (URI): Generic Syntax. T. Berners-Lee, R. Fielding, and L. Masinter. IETF, January 2005, http://www.ietf.org/rfc/rfc3986.txt
[RFC 3987]
RFC 3987: Internationalized Resource Identifiers (IRIs). M. Duerst and M. Suignard. IETF, January 2005, http://www.ietf.org/rfc/rfc3987.txt
[UNISEC]
Unicode Security Considerations. Mark Davis and Michel Suignard. Unicode technical report 36, 23 July 2008, http://www.unicode.org/reports/tr36/tr36-7.html. Latest version available as http://www.unicode.org/reports/tr36/.