

# Resource Description Framework

The **Resource Description Framework (RDF)** is a family of **World Wide Web Consortium (W3C) specifications**<sup>[1]</sup> originally designed as a **metadata data model**. It has come to be used as a general method for conceptual description or modeling of information that is implemented in **web resources**, using a variety of syntax notations and **data serialization** formats. It is also used in **knowledge management** applications.

RDF was adopted as a W3C recommendation in 1999. The RDF 1.0 specification was published in 2004, the RDF 1.1 specification in 2014.

## 1 Overview

The RDF data model<sup>[2]</sup> is similar to classical conceptual modeling approaches (such as **entity–relationship** or **class diagrams**). It is based upon the idea of making **statements** about **resources** (in particular **web resources**) in the form of **subject–predicate–object** expressions, known as *triples*. The subject denotes the resource, and the predicate denotes traits or aspects of the resource, and expresses a relationship between the subject and the object.

For example, one way to represent the notion “The sky has the color blue” in RDF is as the triple: a **subject** denoting “the sky”, a **predicate** denoting “has the color”, and an **object** denoting “blue”. Therefore, RDF swaps object for subject in contrast to the typical approach of an **entity–attribute–value** model in **object-oriented design**: entity (sky), attribute (color), and value (blue).

RDF is an abstract model with several **serialization formats** (i.e. file formats), so the particular encoding for resources or triples varies from format to format.

This mechanism for describing resources is a major **component** in the W3C’s **Semantic Web** activity: an evolutionary stage of the **World Wide Web** in which automated software can store, exchange, and use machine-readable information distributed throughout the Web, in turn enabling users to deal with the information with greater efficiency and **certainty**. RDF’s simple data model and ability to model disparate, abstract concepts has also led to its increasing use in **knowledge management** applications unrelated to Semantic Web activity.

A collection of RDF statements intrinsically represents a **labeled, directed multi-graph**. This theoretically makes an RDF data model better suited to certain kinds of **knowledge representation** than other **relational** or **ontological** models. However, in practice, RDF data is

often persisted in **relational database** or native representations (also called **Triplestores**—or **Quad stores**, if context (i.e. the **named graph**) is also persisted for each RDF triple).<sup>[3]</sup>

As **RDFS** and **OWL** demonstrate, one can build additional **ontology languages** upon RDF.

## 2 History

The initial RDF design, intended to “build a vendor-neutral and operating system-independent system of metadata,”<sup>[4]</sup> derived from the W3C’s **Platform for Internet Content Selection (PICS)**, an early web content labelling system,<sup>[5]</sup> but the project was also shaped by ideas from **Dublin Core**, and from the **Meta Content Framework (MCF)**,<sup>[4]</sup> which had been developed during 1995–1997 by **Ramanathan V. Guha** at **Apple** and **Tim Bray** at **Netscape**.<sup>[6]</sup>

A first public draft of RDF appeared in October 1997,<sup>[7][8]</sup> issued by a W3C working group that included representatives from **IBM**, **Microsoft**, **Netscape**, **Nokia**, **Reuters**, **SoftQuad**, and the **University of Michigan**.<sup>[5]</sup>

The W3C published a specification of RDF’s data model and an **XML** serialization as a recommendation in February 1999.<sup>[9]</sup>

Two persistent misunderstandings developed around RDF at this time: firstly, from the MCF influence and the RDF “Resource Description” acronym, the idea that RDF was specifically for use in representing metadata. Secondly that RDF was an XML format, rather than RDF being a data model and only the RDF/XML serialisation being XML-based. RDF saw little take-up in this period, but there was significant work carried out in **Bristol**, around **ILRT** at **Bristol University** and **HP Labs**, and also in Boston at **MIT**. **RSS 1.0** and **FOAF** became exemplar applications for RDF in this period.

The recommendation of 1999 was replaced in 2004 by a set of six specifications: “The RDF Primer”,<sup>[10]</sup> “RDF Concepts and Abstract”,<sup>[11]</sup> “RDF/XML Syntax Specification (revised)”,<sup>[12]</sup> “RDF Semantics”,<sup>[13]</sup> “RDF Vocabulary Description Language 1.0”,<sup>[14]</sup> and “The RDF Test Cases”.<sup>[15]</sup>

This series was superseded in 2014 by the following six “RDF 1.1” documents: “RDF 1.1 Primer”,<sup>[16]</sup> “RDF 1.1 Concepts and Abstract Syntax”,<sup>[17]</sup> “RDF 1.1 XML Syntax”,<sup>[18]</sup> “RDF 1.1 Semantics”,<sup>[19]</sup> “RDF Schema

1.1,”<sup>[20]</sup> and “RDF 1.1 Test Cases”.<sup>[21]</sup>

## 3 RDF topics

### 3.1 RDF vocabulary

The vocabulary defined by the RDF specification is as follows:<sup>[22]</sup>

#### 3.1.1 Classes

##### rdf

- **rdf:XMLLiteral** – the class of XML literal values
- **rdf:Property** – the class of properties
- **rdf:Statement** – the class of RDF statements
- **rdf:Alt**, **rdf:Bag**, **rdf:Seq** – containers of alternatives, unordered containers, and ordered containers (rdfs:Container is a super-class of the three)
- **rdf:List** – the class of RDF Lists
- **rdf:nil** – an instance of rdf:List representing the empty list

##### rdfs

- **rdfs:Resource** – the class resource, everything
- **rdfs:Literal** – the class of literal values, e.g. **strings** and **integers**
- **rdfs:Class** – the class of classes
- **rdfs:Datatype** – the class of RDF datatypes
- **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 rdfs:member

#### 3.1.2 Properties

##### rdf

- **rdf:type** – an instance of rdf:Property used to state that a resource is an instance of a class
- **rdf:first** – the first item in the subject RDF list
- **rdf:rest** – the rest of the subject RDF list after rdf:first
- **rdf:value** – idiomatic property used for structured values

- **rdf:subject** – the subject of the subject RDF statement
- **rdf:predicate** – the predicate of the subject RDF statement
- **rdf:object** – the object of the subject RDF statement

rdf:Statement, rdf:subject, rdf:predicate, rdf:object are used for **reification** (see **below**).

##### rdfs

- **rdfs:subClassOf** – the subject is a subclass of a class
- **rdfs:subPropertyOf** – the subject is a subproperty of a property
- **rdfs:domain** – a domain of the subject property
- **rdfs:range** – a range of the subject property
- **rdfs:label** – a human-readable name for the subject
- **rdfs:comment** – a description of the subject resource
- **rdfs:member** – a member of the subject resource
- **rdfs:seeAlso** – further information about the subject resource
- **rdfs:isDefinedBy** – the definition of the subject resource

This vocabulary is used as a foundation for **RDF Schema** where it is extended.

### 3.2 Serialization formats

Several common **serialization formats** are in use, including:

- **Turtle**,<sup>[25]</sup> a compact, human-friendly format.
- **N-Triples**,<sup>[26]</sup> a very simple, easy-to-parse, line-based format that is not as compact as Turtle.
- **N-Quads**,<sup>[27][28]</sup> a superset of N-Triples, for serializing multiple RDF graphs.
- **JSON-LD**,<sup>[29]</sup> a JSON-based serialization.
- **N3** or **Notation3**, a non-standard serialization that is very similar to Turtle, but has some additional features, such as the ability to define inference rules.
- **RDF/XML**,<sup>[30]</sup> an XML-based syntax that was the first standard format for serializing RDF.

RDF/XML is sometimes misleadingly called simply RDF because it was introduced among the other W3C specifications defining RDF and it was historically the first W3C standard RDF serialization format. However, it is important to distinguish the RDF/XML format from the abstract RDF model itself. Although the RDF/XML format is still in use, other RDF serializations are now preferred by many RDF users, both because they are more human-friendly,<sup>[31]</sup> and because some RDF graphs are not representable in RDF/XML due to restrictions on the syntax of XML **QNames**.

With a little effort, virtually any arbitrary **XML** may also be interpreted as RDF using **GRDDL** (pronounced 'grid-dle'), Gleaning Resource Descriptions from Dialects of Languages.

RDF triples may be stored in a type of database called a **triplestore**.

### 3.3 Resource identification

The subject of an RDF statement is either a **uniform resource identifier** (URI) or a **blank node**, both of which denote **resources**. Resources indicated by **blank nodes** are called anonymous resources. They are not directly identifiable from the RDF statement. The predicate is a URI which also indicates a resource, representing a relationship. The object is a URI, blank node or a **Unicode string literal**. As of RDF 1.1 resources are identified by IRI's. IRI is a generalization of URI.<sup>[32]</sup>

In Semantic Web applications, and in relatively popular applications of RDF like **RSS** and **FOAF** (Friend of a Friend), resources tend to be represented by URIs that intentionally denote, and can be used to access, actual data on the World Wide Web. But RDF, in general, is not limited to the description of Internet-based resources. In fact, the URI that names a resource does not have to be dereferenceable at all. For example, a URI that begins with "http:" and is used as the subject of an RDF statement does not necessarily have to represent a resource that is accessible via **HTTP**, nor does it need to represent a tangible, network-accessible resource — such a URI could represent absolutely anything. However, there is broad agreement that a bare URI (without a # symbol) which returns a 300-level coded response when used in an HTTP GET request should be treated as denoting the internet resource that it succeeds in accessing.

Therefore, producers and consumers of RDF statements must agree on the semantics of resource identifiers. Such agreement is not inherent to RDF itself, although there are some controlled vocabularies in common use, such as **Dublin Core Metadata**, which is partially mapped to a URI space for use in RDF. The intent of publishing RDF-based ontologies on the Web is often to establish, or circumscribe, the intended meanings of the resource identifiers used to express data in RDF. For example, the URI:

<http://www.w3.org/TR/2004/REC-owl-guide-20040210/wine#Merlot>

is intended by its owners to refer to the class of all **Merlot** red wines by vintner (i.e., instances of the above URI each represent the class of all wine produced by a single vintner), a definition which is expressed by the OWL ontology — itself an RDF document — in which it occurs. Without careful analysis of the definition, one might erroneously conclude that an instance of the above URI was something physical, instead of a type of wine.

Note that this is not a 'bare' resource identifier, but is rather a **URI reference**, containing the '#' character and ending with a **fragment identifier**.

### 3.4 Statement reification and context

The body of knowledge modeled by a collection of statements may be subjected to **reification**, in which each *statement* (that is each triple *subject-predicate-object* altogether) is assigned a URI and treated as a resource about which additional statements can be made, as in "*Jane says that John is the author of document X*". Reification is sometimes important in order to deduce a level of confidence or degree of usefulness for each statement.

In a reified RDF database, each original statement, being a resource, itself, most likely has at least three additional statements made about it: one to assert that its subject is some resource, one to assert that its predicate is some resource, and one to assert that its object is some resource or literal. More statements about the original statement may also exist, depending on the application's needs.

Borrowing from concepts available in **logic** (and as illustrated in graphical notations such as **conceptual graphs** and **topic maps**), some RDF model implementations acknowledge that it is sometimes useful to group statements according to different criteria, called *situations*, *contexts*, or *scopes*, as discussed in articles by RDF specification co-editor **Graham Klyne**.<sup>[33][34]</sup> For example, a statement can be associated with a context, named by a URI, in order to assert an "is true in" relationship. As another example, it is sometimes convenient to group statements by their source, which can be identified by a URI, such as the URI of a particular RDF/XML document. Then, when updates are made to the source, corresponding statements can be changed in the model, as well.

Implementation of scopes does not necessarily require fully reified statements. Some implementations allow a single scope identifier to be associated with a statement that has not been assigned a URI, itself.<sup>[35][36]</sup> Likewise *named graphs* in which a set of triples is named by a URI can represent context without the need to reify the triples.<sup>[37]</sup>

### 3.5 Query and inference languages

Main article: [RDF query language](#)

The predominant query language for RDF graphs is **SPARQL**. SPARQL is an **SQL**-like language, and a recommendation of the **W3C** as of January 15, 2008.

An example of a SPARQL query to show country capitals in Africa, using a fictional ontology.

PREFIX ex: <http://example.com/exampleOntology#>  
 SELECT ?capital ?country WHERE { ?x ex:cityname ?capital ; ex:isCapitalOf ?y . ?y ex:countryname ?country ; ex:isInContinent ex:Africa . }

Other non-standard ways to query RDF graphs include:

- **RDQL**, precursor to **SPARQL**, SQL-like
- **Versa**, compact syntax (non-SQL-like), solely implemented in **4Suite** (Python)
- **RQL**, one of the first declarative languages for uniformly querying RDF schemas and resource descriptions, implemented in **RDFSuite**.<sup>[38]</sup>
- **SeRQL**, part of **Sesame**
- **XUL** has a template element in which to declare rules for matching data in RDF. XUL uses RDF extensively for databinding.

### 3.6 Validation and description

Main article: [RDF query language](#)

There are several proposals to validate and describe RDF:

- **SHACL** (Shapes Constraint Language)<sup>[39]</sup> is expresses constraints on RDF Graphs. SHACL is divided in two parts: SHACL Core and SHACL-SPARQL. SHACL Core consists of a list of built-in constraints such as cardinality, range of values and many others. SHACL-SPARQL consists of all features of SHACL Core plus the advanced features of SPARQL-based constraints and an extension mechanism to declare new constraint components.
- **ShEx** (Shape Expressions)<sup>[40]</sup> is a concise language for RDF validation and description.

## 4 Examples

### 4.1 Example 1: RDF Description of a person named Eric Miller<sup>[41]</sup>

The following example is taken from the W3C website<sup>[41]</sup> describing a resource with statements “there is a Person identified by <http://www.w3.org/People/EM/contact#me>, whose name is Eric Miller, whose email address is e.miller123(at)example (changed for security purposes), and whose title is Dr.



An RDF Graph Describing Eric Miller<sup>[41]</sup>

The resource "<http://www.w3.org/People/EM/contact#me>" is the subject.

The objects are:

- “Eric Miller” (with a predicate “whose name is”),
- [mailto:e.miller123\(at\)example](mailto:e.miller123(at)example) (with a predicate “whose email address is”), and
- “Dr.” (with a predicate “whose title is”).

The subject is a URI.

The predicates also have URIs. For example, the URI for each predicate:

- “whose name is” is <http://www.w3.org/2000/10/swap/pim/contact#fullName>,
- “whose email address is” is <http://www.w3.org/2000/10/swap/pim/contact#mailbox>,
- “whose title is” is <http://www.w3.org/2000/10/swap/pim/contact#personalTitle>.

In addition, the subject has a type (with URI <http://www.w3.org/1999/02/22-rdf-syntax-ns#type>), which is person (with URI <http://www.w3.org/2000/10/swap/pim/contact#Person>).

Therefore, the following “subject, predicate, object” RDF triples can be expressed:

- `http://www.w3.org/People/EM/contact#me`, `http://www.w3.org/2000/10/swap/pim/contact#fullName`, “Eric Miller”
- `http://www.w3.org/People/EM/contact#me`, `http://www.w3.org/2000/10/swap/pim/contact#mailbox`, `mailto:e.miller123(at)example`
- `http://www.w3.org/People/EM/contact#me`, `http://www.w3.org/2000/10/swap/pim/contact#personalTitle`, “Dr.”
- `http://www.w3.org/People/EM/contact#me`, `http://www.w3.org/1999/02/22-rdf-syntax-ns#type`, `http://www.w3.org/2000/10/swap/pim/contact#Person`

In standard N-Triples format, this RDF can be written as:

```
<http://www.w3.org/People/EM/contact#me> <http://www.w3.org/2000/10/swap/pim/contact#fullName>
“Eric Miller” . <http://www.w3.org/People/EM/contact#me> <http://www.w3.org/2000/10/swap/pim/contact#mailbox>
<mailto:e.miller123(at)example> . <http://www.w3.org/People/EM/contact#me>
<http://www.w3.org/2000/10/swap/pim/contact#personalTitle> “Dr.” . <http://www.w3.org/People/EM/contact#me>
<http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://www.w3.org/2000/10/swap/pim/contact#Person> .
```

Equivalently, it can be written in standard Turtle (syntax) format as:

```
@prefix eric: <http://www.w3.org/People/EM/contact#> . @prefix contact: <http://www.w3.org/2000/10/swap/pim/contact#> . @prefix rdf:
<http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
eric:me contact:fullName “Eric Miller” . eric:me contact:mailbox <mailto:e.miller123(at)example> .
eric:me contact:personalTitle “Dr.” . eric:me rdf:type contact:Person .
```

Or, it can be written in RDF/XML format as:

```
<?xml version="1.0" encoding="utf-8"?> <rdf:RDF
xmlns:contact=\char"0022relax{ }http://www.w3.org/2000/10/swap/pim/contact#"
xmlns:eric=\char"0022relax{ }http://www.w3.org/People/EM/contact#"
xmlns:rdf=\char"0022relax{ }http://www.w3.org/1999/02/22-rdf-syntax-ns#">
  <rdf:Description
rdf:about=\char"0022relax{ }http://www.w3.org/People/EM/contact#me">
    <contact:fullName>Eric Miller</contact:fullName>
  </rdf:Description>
  <rdf:Description
rdf:about=\char"0022relax{ }http://www.w3.org/People/EM/contact#me">
    <contact:mailbox
rdf:resource=\char"0022relax{ }mailto:e.miller123(at)example\char"0022relax{ }/>
  </rdf:Description>
  <rdf:Description
rdf:about=\char"0022relax{ }http://www.w3.org/People/EM/contact#me">
    <contact:personalTitle>Dr.</contact:personalTitle>
  </rdf:Description>
```

```
<Description> <rdf:Description
rdf:about=\char"0022relax{ }http://www.w3.org/People/EM/contact#me">
  <rdf:type
rdf:resource=\char"0022relax{ }http://www.w3.org/2000/10/swap/pim/contact#Person\char"0022relax{ }/>
</rdf:Description> </rdf:RDF>
```

## 4.2 Example 2: The postal abbreviation for New York

Certain concepts in RDF are taken from **logic** and **linguistics**, where subject-predicate and subject-predicate-object structures have meanings similar to, yet distinct from, the uses of those terms in RDF. This example demonstrates:

In the **English language** statement ‘*New York has the postal abbreviation NY*’, ‘*New York*’ would be the subject, ‘*has the postal abbreviation*’ the predicate and ‘*NY*’ the object.

Encoded as an RDF triple, the subject and predicate would have to be resources named by URIs. The object could be a resource or literal element. For example, in the N-Triples form of RDF, the statement might look like:

```
<urn:x-states:New%20York> <http://purl.org/dc/terms/alternative> “NY” .
```

In this example, “urn:x-states:New%20York” is the URI for a resource that denotes the US state **New York**, “http://purl.org/dc/terms/alternative” is the URI for a predicate (whose human-readable definition can be found at here <sup>[42]</sup>), and “NY” is a literal string. Note that the URIs chosen here are not standard, and don’t need to be, as long as their meaning is known to whatever is reading them.

## 4.3 Example 3: A Wikipedia article about Tony Benn

In a like manner, given that “http://en.wikipedia.org/wiki/Tony\_Benn” identifies a particular resource (regardless of whether that URI could be traversed as a hyperlink, or whether the resource is *actually* the **Wikipedia** article about **Tony Benn**), to say that the title of this resource is “Tony Benn” and its publisher is “Wikipedia” would be two assertions that could be expressed as valid RDF statements. In the N-Triples form of RDF, these statements might look like the following:

```
<http://en.wikipedia.org/wiki/Tony_Benn>
<http://purl.org/dc/elements/1.1/title> “Tony Benn” .
<http://en.wikipedia.org/wiki/Tony_Benn> <http://purl.org/dc/elements/1.1/publisher> “Wikipedia” .
```

To an English-speaking person, the same information could be represented simply as:



The title of this resource, which is published by Wikipedia, is 'Tony Benn'

However, RDF puts the information in a formal way that a machine can understand. The purpose of RDF is to provide an **encoding** and interpretation mechanism so that **resources** can be described in a way that particular **software** can understand it; in other words, so that software can access and use information that it otherwise couldn't use.

Both versions of the statements above are wordy because one requirement for an RDF resource (as a subject or a predicate) is that it be unique. The subject resource must be unique in an attempt to pinpoint the exact resource being described. The predicate needs to be unique in order to reduce the chance that the idea of **Title** or **Publisher** will be ambiguous to software working with the description. If the software recognizes *http://purl.org/dc/elements/1.1/title* (a specific **definition** for the **concept** of a title established by the **Dublin Core** Metadata Initiative), it will also know that this title is different from a land title or an honorary title or just the letters t-i-t-l-e put together.

The following example, written in Turtle, shows how such simple claims can be elaborated on, by combining multiple RDF vocabularies. Here, we note that the primary topic of the Wikipedia page is a "Person" whose name is "Tony Benn":

```
@prefix rdf:      <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix foaf:    <http://xmlns.com/foaf/0.1/> .
@prefix dc:      <http://purl.org/dc/elements/1.1/> .
<http://en.wikipedia.org/wiki/Tony_Benn>
dc:publisher "Wikipedia" ;
dc:title "Tony Benn" ;
foaf:primaryTopic [ a foaf:Person ; foaf:name "Tony Benn" ] .
```

## 5 Applications

- **DBpedia** – Extracts facts from Wikipedia articles and publishes them as RDF data.
- **Creative Commons** – Uses RDF to embed license information in web pages and mp3 files.
- **FOAF (Friend of a Friend)** – designed to describe people, their interests and interconnections.
- **Haystack client** – Semantic web browser from MIT CS & AI lab.<sup>[43]</sup>
- **IDEAS Group** – developing a formal **4D ontology** for **Enterprise Architecture** using RDF as the encoding.<sup>[44]</sup>
- Microsoft shipped a product, **Connected Services Framework**,<sup>[45]</sup> which provides RDF-based Profile Management capabilities.

- **MusicBrainz** – Publishes information about Music Albums.<sup>[46]</sup>
- **NEPOMUK**, an open-source software specification for a Social Semantic desktop uses RDF as a storage format for collected metadata. NEPOMUK is mostly known because of its integration into the **KDE SC 4** desktop environment.
- **Press Association** is a news agency in the UK. They use ontologies to dynamically identify and link their NoSQL data to do **semantic publishing** but in a dynamic, rules based way that creates custom content on the fly.<sup>[47]</sup>
- **RDF Site Summary** – one of several "**RSS**" languages for publishing information about updates made to a web page; it is often used for disseminating news article summaries and sharing **weblog** content.
- **Simple Knowledge Organization System (SKOS)** – a KR representation intended to support vocabulary/thesaurus applications
- **SIOC (Semantically-Interlinked Online Communities)** – designed to describe online communities and to create connections between Internet-based discussions from message boards, weblogs and mailing lists.<sup>[48]</sup>
- **Smart-M3** – provides an infrastructure for using RDF and specifically uses the ontology agnostic nature of RDF to enable heterogeneous mashing-up of information<sup>[49]</sup>

Some uses of RDF include research into social networking. It will also help people in business fields understand better their relationships with members of industries that could be of use for product placement.<sup>[50]</sup> It will also help scientists understand how people are connected to one another.

RDF is being used to have a better understanding of road traffic patterns. This is because the information regarding traffic patterns is on different websites, and RDF is used to integrate information from different sources on the web. Before, the common methodology was using keyword searching, but this method is problematic because it does not consider synonyms. This is why ontologies are useful in this situation. But one of the issues that comes up when trying to efficiently study traffic is that to fully understand traffic, concepts related to people, streets, and roads must be well understood. Since these are human concepts, they require the addition of **fuzzy logic**. This is because values that are useful when describing roads, like slipperiness, are not precise concepts and cannot be measured. This would imply that the best solution would incorporate both fuzzy logic and ontology.<sup>[51]</sup>

## 6 See also

### Notations for RDF

- [TRiG](#)
- [TRiX](#)
- [RDF/XML](#)
- [RDFa](#)
- [JSON-LD](#)

### Similar concepts

- [Entity-attribute-value model](#)
- [Graph theory](#) – An RDF model is a labeled, directed multi-graph.
- [Website Parse Template](#)
- [Tagging](#)
- [SciCrunch](#)
- [Semantic network](#)

### Other (unsorted)

- [Associative model of data](#)
- [Business Intelligence 2.0 \(BI 2.0\)](#)
- [DataPortability](#)
- [EU Open Data Portal](#)
- [Folksonomy](#)
- [Life Science Identifiers](#)
- [Swoogle](#)
- [Universal Networking Language \(UNL\)](#)

## 7 References

- [1] “XML and Semantic Web W3C Standards Timeline” (PDF). 2012-02-04.
- [2] <http://www.w3.org/TR/PR-rdf-syntax/> “Resource Description Framework (RDF) Model and Syntax Specification”
- [3] Optimized Index Structures for Querying RDF from the Web Andreas Harth, Stefan Decker, 3rd Latin American Web Congress, Buenos Aires, Argentina, October 31 to November 2, 2005, pp. 71–80
- [4] “World Wide Web Consortium Publishes Public Draft of Resource Description Framework”. W3C. Cambridge, MA. 1997-10-03.
- [5] Lash, Alex (1997-10-03). “W3C takes first step toward RDF spec”. *CNET News*. Archived from the original on June 16, 2011. Retrieved 2015-11-28.
- [6] Hammersley, Ben (2005). *Developing Feeds with RSS and Atom*. Sebastopol: O'Reilly. pp. 2–3. ISBN 0-596-00881-3.
- [7] Lassila, Ora; Swick, Ralph R. (1997-10-02). “Resource Description Framework (RDF): Model and Syntax”. W3C. Retrieved 2015-11-24.
- [8] Swick, Ralph (1997-12-11). “Resource Description Framework (RDF)”. W3C. Archived from the original on February 14, 1998. Retrieved 2015-11-24.
- [9] “Resource Description Framework (RDF) Model and Syntax Specification”. 22 Feb 1999. Retrieved 5 May 2014.
- [10] Manola, Frank; Miller, Eric (2004-02-10), *RDF Primer*, W3C, retrieved 2015-11-21
- [11] Klyne, Graham; Carroll, Jeremy J. (2004-02-10), *Resource Description Framework (RDF): Concepts and Abstract Syntax*, W3C, retrieved 2015-11-21
- [12] Beckett, Dave (2004-02-10), *RDF/XML Syntax Specification (Revised)*, W3C, retrieved 2015-11-21
- [13] Hayes, Patrick (2014-02-10), *RDF Semantics*, retrieved 2015-11-21
- [14] Brickley, Dan; Guha, R.V. (2004-02-10), *RDF Vocabulary Description Language 1.0: RDF Schema: W3C Recommendation 10 February 2004*, W3C, retrieved 2015-11-21
- [15] Grant, Jan; Beckett, Dave (2004-02-10), *RDF Test Cases*, W3C, retrieved 2015-11-21
- [16] Schreiber, Guus; Raimond, Yves (2014-06-24), *RDF 1.1 Primer*, W3C, retrieved 2015-11-22
- [17] Cyganiak, Richard; Wood, David; Lanthaler, Markus (2014-02-25), *RDF 1.1 Concepts and Abstract Syntax*, W3C, retrieved 2015-11-22
- [18] Gandon, Fabien; Schreiber, Guus (2014-02-25), *RDF 1.1 XML Syntax*, W3C, retrieved 2015-11-22
- [19] Hayes, Patrick J.; Patel-Schneider, Peter F. (2014-02-25), *RDF 1.1 Semantics*, W3C, retrieved 2015-11-22
- [20] Brickley, Dan; Guha, R.V. (2014-02-25), *RDF Schema 1.1*, W3C, retrieved 2015-11-22
- [21] Kellogg, Gregg; Lanthaler, Markus (2014-02-25), *RDF 1.1 Test Cases*, W3C, retrieved 2015-11-22
- [22] “RDF Vocabulary Description Language 1.0: RDF Schema”. W3C. 2004-02-10. Retrieved 2011-01-05.
- [23] “RDF 1.1 Turtle: Terse RDF Triple Language”. W3C. 9 Jan 2014. Retrieved 2014-02-22.
- [24] “application/rdf+xml Media Type Registration”. IETF. September 2004. p. 2. Retrieved 2011-01-08.

- [25] “RDF 1.1 Turtle: Terse RDF Triple Language”. W3C. 9 January 2014.
- [26] “RDF 1.1 N-Triples: A line-based syntax for an RDF graph”. W3C. 9 January 2014.
- [27] “N-Quads: Extending N-Triples with Context”. 2012-06-25.
- [28] “RDF 1.1 N-Quads”. W3C. January 2014.
- [29] “JSON-LD 1.0: A JSON-based Serialization for Linked Data”. W3C.
- [30] “RDF 1.1 XML Syntax”. W3C. 25 February 2014.
- [31] “Problems of the RDF syntax”. Vuk Miličić.
- [32] RDF 1.1 Concepts and Abstract Syntax <https://www.w3.org/TR/rdf11-concepts/>
- [33] Contexts for RDF Information Modelling
- [34] Circumstance, Provenance and Partial Knowledge
- [35] The Concept of 4Suite RDF Scopes
- [36] Redland RDF Library – Contexts
- [37] Named Graphs
- [38] “The RDF Query Language (RQL)”. *The ICS-FORTH RDFSuite*. ICS-FORTH.
- [39] SHACL Specification
- [40] ShEx Specification
- [41] “RDF Primer”. W3C. Retrieved 2009-03-13.
- [42] DCMI Metadata Terms. Dublincore.org. Retrieved on 2014-05-30.
- [43] Haystack
- [44] The IDEAS Group Website
- [45] Connected Services Framework
- [46] RDF on MusicBrainz Wiki
- [47]
- [48] SIOC (Semantically-Interlinked Online Communities)
- [49] Oliver Ian, Honkola Jukka, Ziegler Jurgen (2008). “Dynamic, Localized Space Based Semantic Webs”. IADIS WWW/Internet 2008. Proceedings, p.426, IADIS Press, ISBN 978-972-8924-68-3
- [50] An RDF Approach for Discovering the Relevant Semantic Associations in a Social Network By Thushar A.K, and P. Santhi Thilagam
- [51] Traffic Information Retrieval Based on Fuzzy Ontology and RDF on the Semantic Web By Jun Zhai, Yi Yu, Yiduo Liang, and Jiatao Jiang (2008)

## 8 Further reading

- W3C’s RDF at W3C: specifications, guides, and resources
- RDF Semantics: specification of semantics, and complete systems of inference rules for both RDF and RDFS

## 9 External links

- Resource Description Framework at DMOZ



## 10 Text and image sources, contributors, and licenses

### 10.1 Text

- **Resource Description Framework** *Source:* [https://en.wikipedia.org/wiki/Resource\\_Description\\_Framework?oldid=780636994](https://en.wikipedia.org/wiki/Resource_Description_Framework?oldid=780636994) *Contributors:* WojPob, Mav, Uriyan, The Anome, Mjb, Stevertigo, Danja, Michael Hardy, Kku, Liftarn, Stw, Haakon, Angela, Emperor, Scott, Kaihsu, Denny, Novum, Universimmedia, Sanxiyn, RaymondYee, SEWilco, Carbuncle, Robbot, Mccaffry, Fredrik, RedWolf, Securiger, Rursus, Bergie, Baojie, Beorhast, Lysy, David Gerard, ShaunMacPherson, Sstair, Fleminra, Jonathan O'Donnell, Niteowlneils, Carlo.lerna, Langed, Rchandra, AlistairMcMillan, Jackol, Kwan, DRE, Ojw, SamuelScarano, BeakerK44, Chmod007, Asqueella, Jwdietrich2, Rauyran, RossPatterson, Rich Farmbrough, Mecanismo, Bishonen, Xezbeth, BernhardBauer, Yitzhak, Trixter, Stesmo, Smalljim, John Vandenberg, Enric Naval, Joe Jarvis, Bawolff, Vanished user 19794758563875, Minghong, Mdd, Jumbuck, Diego Moya, Cjcollier, Fkbreitl, Dtcddthingy, Nkour, Tomlzz1, SteinbDJ, Roland2~enwiki, Onlyemarie, JFG, KingsleyIdehen, GregorB, Waldir, Toussaint, TNLNYC, Turnstep, Caoimhin, Mandarax, Stoni, Pete142, Graham87, Kbdank71, Dpv, Koavf, Amire80, Venullian, Dmccreary, Syced, CloCk-WeRX, FlaBot, Ultimatewisdom, Mathiastck, Schandi, Chobot, Bgwhite, Anrie Nord, FrankTobia, Roboto de Ajvol, YurikBot, Wavelength, Pi Delport, Dr Shorthair, Wiki alf, BobKeim, Deodar~enwiki, Jpbowen, Davemck, DanBri, Zwobot, Arthur Rubin, Donald Albury, Cedar101, SMcCandlish, Esprit15d, LeonardoRob0t, Back ache, Shepard, Finell, Tom Morris, Luk, SmackBot, Commander Keane bot, Chris the speller, ARK, Stevage, Jdthood, JonHarder, Rrburke, N3c, MichaelBillington, StephenReed, A5b, Sina2, Zearin, Ian Spackman, Soumyasch, Nux, Avé, Jhammerb, Mauro Bieg, Sibersandi, Hu12, Hetar, Arto B, Booles, DBooth, CmdrObot, Lokatzis, Alcalazar, N2e, Pagatiponon, Sdorrance, Harrigan, Tezza2k1, IanDBailey, Cydebot, JesseChisholm, Dancter, Blue.death, Barticus88, Sapoguapo, BackwardsBoy, Pemboid, EdJohnston, Blathnaid, Acaciz, Akinyemi, Cygri, TikaKino, AnAj, Clan-destine, JAnDbot, Inverse.chi, Shermanmonroe, Nikevich, Hiplibrarianship, LaMona, Lpswaggerty, Jodi.a.schneider, Gwern, Pvosta, DoSiDo, Yaron K., Vsdd-kjn, Daniele Gallezio, Yonidebot, Maduskis, Midnight Madness, Erick.Antezana, Funandtrvl, PatHayes, TXiKiBoT, Andy Dingley, SieBot, Knavesdied, Mkbergman, Int21h, Svick, ImageRemovalBot, ClueBot, WurmWoode, C1932, EoGuy, Grantbow, Niceguyedc, Wesleyneo, Pointillist, Excirial, CohesionBot, Alexbot, Northernhenge, Shinkolobwe, Iohannes Animosus, Triadic2000, Backoftheboat, Alexius08, Dfoxvog, Cedringen, Carmenutzadd, Addbot, GhettoBlaster, Kiranoush, LaaknorBot, Gyuri10, Legobot, Luckas-bot, Yobot, Bunnyhop11, Shallysharma13, Amirobot, Wonderfl, AnomieBOT, Lucideer, ArthurBot, The Evil IP address, GrouchoBot, January2009, EddyVanderlinden, Mark Renier, Ankitasdeveloper, Winterst, I dream of horses, Skyerise, Jandalhandler, Broosty, M3wiki1, Jonkerz, Kghbln, Oliver H, Bokorember, Ottomachin, EmausBot, Angrytoast, Khurrad~enwiki, RA0808, Tommy2010, K6ka, Chrcho, Alisha.4m, Ontheinternets, L0ngpar1sh, Erianna, It writer, Opoirel, Argrico, BioPupil, Karima Rafes, Yoavt, Gougnafier, DASHBotAV, ClueBot NG, Zviedris, Gubbins^100, Highfellow, BG19bot, Endobrendo, Leggattst, MusikAnimal, Compfreak7, Sieuthulin, Radiusdanu, BattyBot, Nikosbik, Cyberbot II, Esszet, Smagch, Mogism, AntoineDube, PetersonKevinJ, Zziccardi, Mark viking, SeMelmoth, FockeWulf FW 190, Amy van der Hiel, Evilbikes79, DavidBooth, Bmarcisz, WikiPhoenix999, Izkala, KasparBot, You better look out below!, Thichfich, Felipecast and Anonymous: 230

### 10.2 Images

- **File:Commons-logo.svg** *Source:* <https://upload.wikimedia.org/wikipedia/en/4/4a/Commons-logo.svg> *License:* PD *Contributors:* ? *Original artist:* ?
- **File:Rdf\_graph\_for\_Eric\_Miller.png** *Source:* [https://upload.wikimedia.org/wikipedia/commons/5/58/Rdf\\_graph\\_for\\_Eric\\_Miller.png](https://upload.wikimedia.org/wikipedia/commons/5/58/Rdf_graph_for_Eric_Miller.png) *License:* Attribution *Contributors:* <http://www.w3.org/TR/rdf-primer/fig1dec16.png> *Original artist:* W3C
- **File:XML.svg** *Source:* <https://upload.wikimedia.org/wikipedia/commons/6/68/XML.svg> *License:* CC BY-SA 2.5 *Contributors:* en:Image:XML2.svg *Original artist:* en>User:Dreftymac

### 10.3 Content license

- Creative Commons Attribution-Share Alike 3.0