**Semantic Web**

What is semantic web?

The **Semantic Web** is an extension of the [World Wide Web](https://en.wikipedia.org/wiki/World_Wide_Web) through standards by the [World Wide Web Consortium](https://en.wikipedia.org/wiki/World_Wide_Web_Consortium) (W3C).[[1]](https://en.wikipedia.org/wiki/Semantic_Web#cite_note-1) The standards promote common data formats and exchange protocols on the Web, most fundamentally the [Resource Description Framework](https://en.wikipedia.org/wiki/Resource_Description_Framework) (RDF).

Many people don't have a clear idea of what the Semantic Web is. The best informal definition is maybe found in the May 2001 Scientific American article "The Semantic Web" (Berners-Lee et al.), that says "The Semantic Web is an extension of the current web in which information is given well-defined meaning, better enabling computers and people to work in cooperation." People who work on the Semantic Web have a more detailed idea of what this broad statement means, and base their work on the famous "semantic web tower", a product of Tim Berners-Lee's inspiring drawing on whiteboards. Even if you are not a geek, you might have read some article on the semantic web, or followed some presentation, and then you will surely have met the tower, which is a drawing like this



The Semantic Web is a web of data. There is lots of data we all use every day, and it is not part of the web. I can see my bank statements on the web, and my photographs, and I can see my appointments in a calendar. But can I see my photos in a calendar to see what I was doing when I took them? Can I see bank statement lines in a calendar?

Why not? Because we don't have a web of data. Because data is controlled by applications, and each application keeps it to itself.

The Semantic Web is about two things. It is about common formats for integration and combination of data drawn from diverse sources, where on the original Web mainly concentrated on the interchange of documents. It is also about language for recording how the data relates to real world objects. That allows a person, or a machine, to start off in one database, and then move through an unending set of databases which are connected not by wires but by being about the same thing.

The **Semantic Web** provides a common framework that allows **data** to be shared and reused across application, enterprise, and community boundaries. It is a collaborative effort led by W3C with participation from a large number of researchers and industrial partners. It is based on the Resource Description Framework ( [RDF](https://www.w3.org/RDF/)).

We arrive at the following three main topics that provide conceptual underpinnings for the Semantic Web:

• Building models: the quest for describing the world in abstract terms to allow for an easier understanding of a complex reality.

• Computing with knowledge: the endeavor of constructing reasoning machines that can draw meaningful conclusions from encoded knowledge.

• Exchanging information: the transmission of complex information resources among computers that allows us to distribute, interlink, and reconcile knowledge on a global scale.

**Resource Description Framework(RDF) Standard**

The Resource Description Framework RDF is a formal language for describing structured information. The goal of RDF is to enable applications to exchange data on the Web while still preserving their original meaning. As opposed to HTML and XML, the main intention now is not to display documents correctly, but rather to allow for further processing and re-combination of the information contained in them. RDF consequently is often viewed as the basic representation format for developing the Semantic Web.

As of today, numerous practical tools are available for dealing with RDF. Virtually every programming language offers libraries for reading and writing RDF documents. Various RDF stores – also called triple stores for reasons that shall become clear soon – are available for keeping and processing large amounts of RDF data, and even commercial database vendors are already providing suitable extensions for their products. RDF is also used to exchange (meta) data in specific application areas. RDF is based on a very simple ***graph-oriented data schema.***

**Names in RDF: URIs**

On the one hand, it may happen that the same resource is labeled with different identifiers, for instance, since there is no globally agreed identifier for the book “Foundations of Semantic Web Technologies.” On the other hand, it may occur that the same identifiers are used for different resources, e.g., “CRC” could refer to the publishing house as well as to the official currency of Puerto Rico. Such ambiguity would obviously be a major problem when trying to process and compose information automatically. To solve the latter problem, RDF uses so-called Uniform Resource Identifiers (URIs) as names to clearly distinguish resources from each other. URIs are a generalization of URLs (Uniform Resource Locators), i.e. of Web addresses as they are used for accessing online documents. Every URL is also a valid URI, and URLs can indeed be used as identifiers in RDF documents that talk about Web resources. In numerous other applications, however, the goal is not to exchange information about Web pages but about many different kinds of objects. In general this might be any object that has a clear identity in the context of the given application: books, places, people, publishing houses, events, relationships among such things, all kinds of abstract concepts, and many more. Such resources can obviously not be retrieved online and hence their URIs are used exclusively for unique identification. URIs that are not URLs are sometimes also called Uniform Resource Names (URNs).

**Data Values in RDF: Literals**

The general construction scheme of URIs is summarized below, where parts in brackets are optional: scheme :[//authority ]path [?query ][#fragment ]

The meaning of the various URI parts is as follows:

**scheme** The name of a URI scheme that classifies the type of URI. Schemes may also provide additional information on how to handle URIs in applications. Examples: http, ftp, mailto, file, irc **authority** URIs of some URI schemes refer to “authorities” for structuring the available identifiers further. On the Web, this is typically a domain name, possibly with additional user and port details. The authority part of a URI is optional and can be recognized by the preceding //. Examples: semantic-web-book.org, john@example.com, example.org:8080

**path** The path is the main part of many URIs, though it is possible to use empty paths, e.g., in email addresses. Paths can be organized hierarchically using / as separator. Examples: /etc/passwd, this/path/with/-:\_˜/is/../okay (paths without initial / are only allowed if no authority is given)

**query** The query is an optional part of the URI that provides additional non-hierarchical information. It can be recognized by its preceding ?. In URLs, queries are typically used for providing parameters, e.g., to a Web Service. Example: q=Semantic+Web+book

**fragment** The optional fragment part provides a second level of identifying resources, and its presence is recognized by the preceding #. In URLs, fragments are often used to address a sub-part of a retrieved resource, such as a section in an HTML file. URIs with different fragments are still different names for the purpose of RDF, even if they may lead to the same document being retrieved in a browser. Example: section1

**Web Ontology Language**

But we have also seen that RDF(S) provides only very limited expressive means and that it is not possible to represent more complex knowledge. For modeling such complex knowledge, expressive representation languages based on formal logic are commonly used. This also allows us to do logical reasoning on the knowledge, and thereby enables the access to knowledge which is only implicitly modeled. OWL is such a language. In order to give the user a choice between different degrees of expressivity, three sublanguages of OWL – called species of OWL – have been designed: OWL Full, OWL DL, and OWL Lite. OWL Full contains both OWL DL and OWL Lite, and OWL DL contains OWL Lite

**OWL Syntax and Intuitive Semantics**

OWL documents are used for modeling OWL ontologies. Two different syntaxes have been standardized in order to express these. One of them is based on RDF and is usually used for data exchange. It is also called OWL RDF syntax since OWL documents in RDF syntax are also valid RDF documents. The other syntax is called the OWL abstract syntax and is somewhat more readable for humans.

**Knowledge Graph and Linked Data**

The **Knowledge Graph** is a knowledge base used by [Google](https://en.wikipedia.org/wiki/Google) and its services to enhance its [search engine](https://en.wikipedia.org/wiki/Google_Search)'s results with information gathered from a variety of sources. This information is presented to users in a box to the right of search results.

**So, then, what exactly is the Knowledge Graph?** It’s Google’s systematic way of putting facts, people and places together, to create interconnected search results that are more accurate and relevant.

More specifically, the “[knowledge graph](http://searchengineland.com/library/google/google-knowledge-graph)” is a database that collects millions of pieces of data about keywords people frequently search for on the World wide web and the intent behind those keywords, based on the already available content.With the knowledge graph, users can get information about people, facts and places that are interconnected in one way or the other.

In [computing](https://en.wikipedia.org/wiki/Computing), **linked data** (often capitalized as **Linked Data**) is a method of publishing structured data so that it can be interlinked and become more useful through [semantic queries](https://en.wikipedia.org/wiki/Semantic_query). It builds upon standard [Web](https://en.wikipedia.org/wiki/World_Wide_Web) technologies such as [HTTP](https://en.wikipedia.org/wiki/Hypertext_Transfer_Protocol), [RDF](https://en.wikipedia.org/wiki/Resource_Description_Framework) and [URIs](https://en.wikipedia.org/wiki/Uniform_resource_identifier), but rather than using them to serve web pages for human readers, it extends them to share information in a way that can be read automatically by computers’

**Triplestore**

A **triplestore** or **RDF store** is a purpose-built [database](https://en.wikipedia.org/wiki/Database) for the storage and retrieval of [triples](https://en.wikipedia.org/wiki/Semantic_triple)[[1]](https://en.wikipedia.org/wiki/Triplestore#cite_note-1) through [semantic queries](https://en.wikipedia.org/wiki/Semantic_query). A triple is a data entity composed of subject-[predicate](https://en.wikipedia.org/wiki/Predicate_(grammar))-object, like "Bob is 35" or "Bob knows Fred".

Much like a [relational database](https://en.wikipedia.org/wiki/Relational_database), one stores information in a triplestore and retrieves it via a [query language](https://en.wikipedia.org/wiki/Query_language). Unlike a relational database, a triplestore is optimized for the storage and retrieval of triples. In addition to queries, triples can usually be imported/exported using [Resource Description Framework](https://en.wikipedia.org/wiki/Resource_Description_Framework) (RDF) and other formats.

RDF triplestore is a type of graph database that stores semantic facts. RDF, which stands for Resource Description Framework, is a model for data publishing and interchange on the Web [standardized by W3C](https://www.w3.org/RDF/).

Being a graph database, triplestore stores data as a network of objects with materialized links between them. This makes RDF triplestore a preferred choice for managing highly interconnected data. Triplestores are more flexible and less costly than a relational database, for example.

The RDF database, often called a semantic graph database, is also capable of handling powerful semantic queries and of using inference for uncovering new information out of the existing relations.

RDF triplestore databases are successfully used for managing Linked Open Data datasets, such as [DBPedia](http://dbpedia.org/" \t "_blank) and [GeoNames](http://www.geonames.org/" \t "_blank), which are published as RDFs and are interconnected with one another. Linked Open Data allows for querying and answering federated queries much faster and for obtaining highly relevant search results.

**LODmilla: a Linked Data browser for all**

LODmilla (<http://lodmilla.sztaki.hu/lodmilla/>) is an open effort to evolve a generic platform for exploring and editing Linked Open Data. Our aim is to enable the extraction and sharing of data associations (or information) hidden in Linked Open Data. LODmilla is an open web application supporting graph views, graph searching and many other commodity features for surfing over Linked Data. The main difference with other existing LOD browsers is that only selected triples are shown, and the prepared 'association graphs' can be easily shared with others. It is possible to make simple edits such as adding, changing or deleting triples also for non-experts, even without any knowledge of RDF. The service is open, the JavaScript source code is on GitHub.