

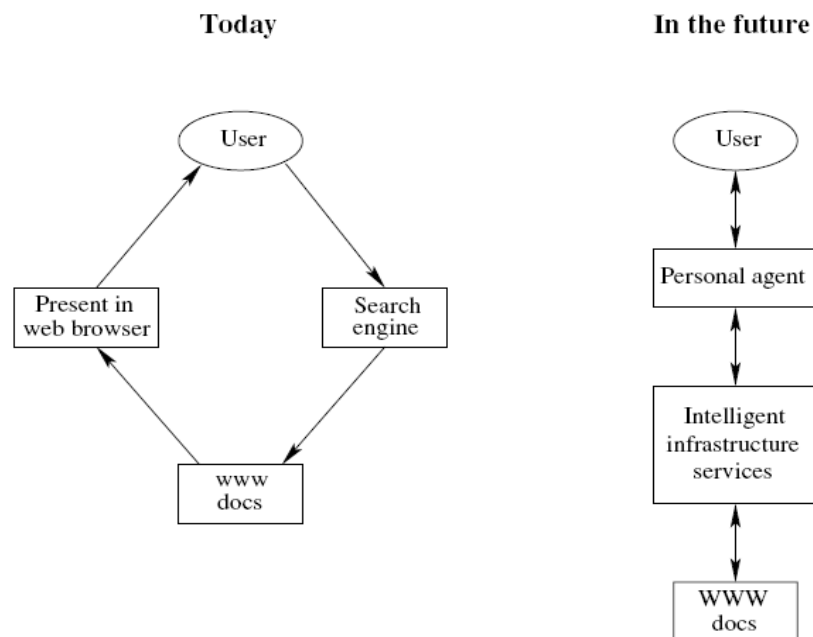
Unit V: Semantic web Technology, Layered Architecture, RDF and OWL representation

1. Semantic Web Technology : Needs, Evolution, Types of Data, Level of Semantics

The World Wide Web has changed the way people communicate with each other and the way business is conducted. It lies at the heart of a revolution that is currently transforming the developed world toward a knowledge economy and, more broadly speaking, to a knowledge society.

This development has also changed the way we think of computers. Originally they were used for computing numerical calculations. Currently their predominant use is for information processing, typical applications being databases, text processing and games. At present there is a transition focus towards the view of computers as entry points to the information high ways.

Most of today's web content is suitable for human consumption. Even web content that is generated automatically from databases is usually presented without the original structural information found in databases. Typical uses of the web today involve people's seeking and making use of information, searching for and getting in touch with other people, reviewing catalogs of online stores and ordering products by filling out forms, and viewing material.



These activities are not particularly well supported by software tools. Apart from the existence of links that establish connections between documents, the main valuable, indeed indispensable, tools are search engines. Keyword-based search engines, such as AltaVista, Yahoo, and Google, are the main tools for using today's web. It is clear that the web would not have been the huge success it was, were it not for search engines. However, there are serious problems associated with their use.

- High recall, low precision. There are many relevant and irrelevant pages retrieved together.
- Low or no recall. Often it happens that we do not get any answer for our request or that important and relevant pages are not retrieved. But it is a less frequent problem.
- Results are highly sensitive to vocabulary.
- Results are single web pages. It is in our hands to initiate several queries to collect the relevant document.

Despite the improvement in search engine technology, the difficulties remain essentially the same. The main obstacle to providing better support to web users is that, at present the meaning of web content is not machine-accessible.

Most information is currently available in a weakly structure form, for example, text, audio, and video. From the knowledge management perspective, the current technology suffers from limitations of the following areas

- Only keyword based searching is permitted
- More time and effort consuming to extract information
- Inconsistency and failure to remove outdated information
- New knowledge extraction is still difficult for distributed, weakly structured collection of documents
- Views and Hiding information from the database are hard to realize.

The aim of the Semantic Web is to allow much more advanced knowledge management systems:

- Knowledge will be organized in conceptual spaces according to its meaning.
- Automated tools will support maintenance by checking for inconsistencies and extracting new knowledge
- Keyword-based search will be replaced by query answering. Requested knowledge will be retrieved, extracted and presented in a human friendly way.
- Query answering over several documents will be supported
- Defining who may view certain parts of information (even parts of documents) will be possible.

The evolutions of Web:

The Semantic Web was made through incremental changes, by bringing machine-readable descriptions to the data and documents already on the Web. The following figure illustrates the various developed technologies that made the concept of the Semantic Web possible.

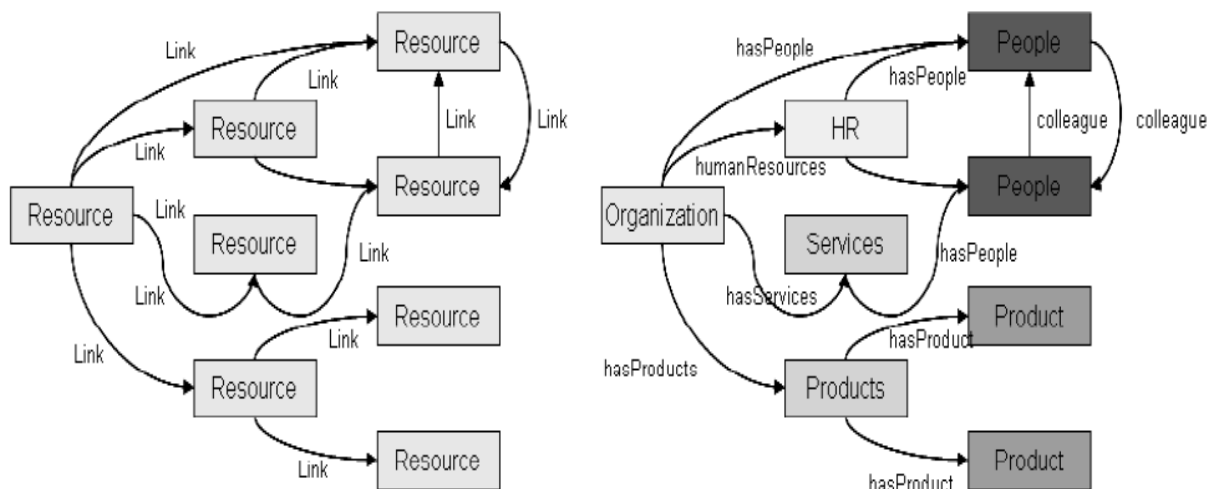
	Static	Dynamic	Syntax	Semantic
Encoding	HTML	+ RDBMS	+ XML	+ RDF/OWL
Creation	Manually	Generated by server-side applications	Generated by applications based on schema	Generated by applications based on models
Users	Humans	Humans	Humans and applications	Humans and applications
Paradigm	Browse	Create/Query/Update	Integrate	Interoperate
Applications	Browsers	Browsers	Process Integration, EAI, BPMS, Workflows	Intelligent agents, Semantic engines

1995
2000
2005

As already stated, the Web was originally a vast set of static Web pages linked together. Many organizations still use static HTML files to deliver their information on the Web. However, to answer to the inherent dynamic nature of businesses, organizations are using dynamic publishing methods which offer great advantages over Web sites constructed from static HTML pages.

The information on the Web can be defined in a way that can be used by computers not only for display purposes, but also for interoperability and integration between systems and applications. One way to enable machine-to-machine exchange and automated processing is to provide the information in such a way that computers can understand it. This is precisely the objective of the semantic Web—to make possible the processing of Web information by computers. The Semantic Web is not a separate Web but an extension of the current one, in which information is given well-defined meaning, better enabling computers and people to work in cooperation.

Currently the Web is undergoing evolution (as illustrated in Figure) and different approaches are being sought for solutions to adding semantics to Web resources. On the left side of Figure, a graph representation of the syntactic Web is given. To give meaning to resources and links, new standards and languages are being investigated and developed as shown in right side of Figure.



Unstructured, Semi-structured and structured data:

Unstructured data	Semi-structured data	Structured data																								
The university has 5600 students. John s ID is number 1, he is 18 years old and already holds a B.Sc. degree. David s ID is number 2, he is 31 years old and holds a Ph.D. degree. Robert s ID is number 3, he is 51 years old and also holds the same degree as David, a Ph.D. degree.	<pre><University> <Student ID= 1"> <Name>John</Name> <Age>18</Age> <Degree>B.Sc.</Degree> </Student> <Student ID= 2"> <Name>David</Name> <Age>31</Age> <Degree>Ph.D. </Degree> </Student> </University></pre>	<table> <tr> <th>ID</th><th>Name</th><th>Age</th><th>Degree</th></tr> <tr> <td>1</td><td>John</td><td>18</td><td>B.Sc.</td></tr> <tr> <td>2</td><td>David</td><td>31</td><td>Ph.D.</td></tr> <tr> <td>3</td><td>Robert</td><td>51</td><td>Ph.D.</td></tr> <tr> <td>4</td><td>Rick</td><td>26</td><td>M.Sc.</td></tr> <tr> <td>5</td><td>Michael</td><td>19</td><td>B.Sc.</td></tr> </table>	ID	Name	Age	Degree	1	John	18	B.Sc.	2	David	31	Ph.D.	3	Robert	51	Ph.D.	4	Rick	26	M.Sc.	5	Michael	19	B.Sc.
ID	Name	Age	Degree																							
1	John	18	B.Sc.																							
2	David	31	Ph.D.																							
3	Robert	51	Ph.D.																							
4	Rick	26	M.Sc.																							
5	Michael	19	B.Sc.																							

Data breaks down into three broad categories : unstructured, semistructured, and structured.

- A) Unstructured Data: Unstructured data is what we find in text, files, video, e-mails, reports, PowerPoint presentations, voice mail, office memos, and images. Data can be of any type and does not necessarily follow any format, rules, or sequence. For example, the data present on HTML Web pages is unstructured and irregular. Unstructured data does not readily fit into structured databases except as binary large objects (BLOBs-binary large objects). Although unstructured data can have some structure—for example, e-mails have addressees, subjects, bodies, and so forth, and HTML Web pages have a set of predefined tags—the information is not stored in such a way that it will allow for easy classification, as the data are entered in electronic form.
- B) Semi-structured data: Semi-structured data lie somewhere in between unstructured and structured data. Semistructured data are data that have some structure, but are not rigidly structured. This type of data includes unstructured components arranged according to some predetermined structure. Semistructured data can be specified in such a way that it can be queried using general-purpose mechanisms. Semistructured data are organized into entities. Similar entities are grouped together, but entities in the same group may not have the same attributes. The order of attributes is not necessarily important and not all attributes may be required.

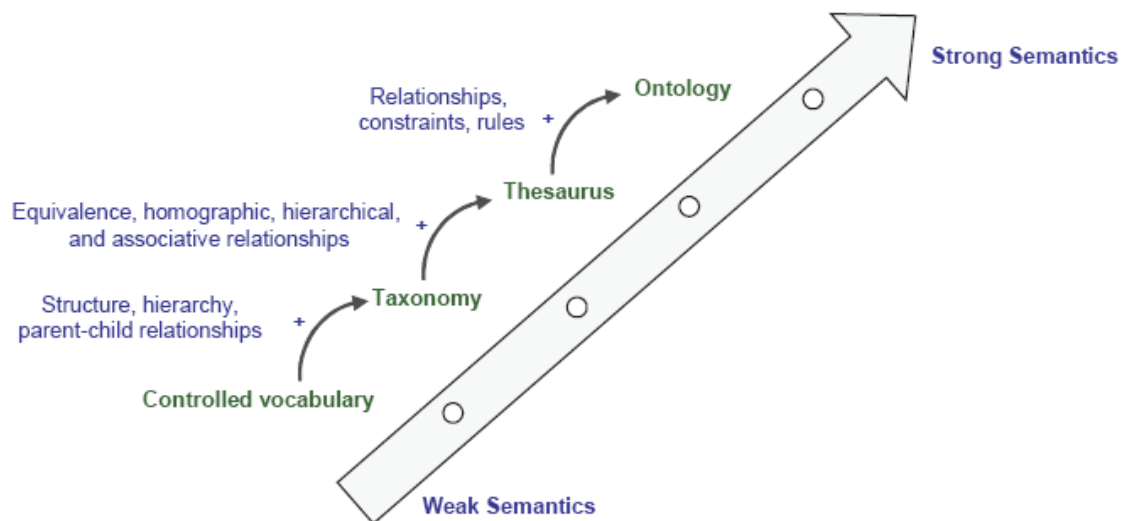
The size and type of same attributes in a group may differ. An example of semistructured data is a curriculum Vitae. One person may have a section of previous employments, another person may have a section on research experience, and another may have a section on teaching experience. We can also find a CV that contains two or more of these sections.

A very good example of a semistructured formalism is XML which is a de facto standard for describing documents that is becoming the universal data exchange model on the Web and is being used for business-to-business transactions. XML supports the development of semistructured documents that contain both metadata and formatted text. Metadata is specified using XML tags and defines the structure of documents. Without metadata, applications would not be able to understand and parse the content of XML documents. Compared to HTML, XML provides explicit data structuring. XML uses DTD or XSD as schema definitions for the semistructured data present in XML documents.

- C) Structured data: In contrast, structured data *are very rigid* and describe objects using strongly typed attributes, which are organized as records or tuples. *All records have the same fields*. Data are organized in entities and similar entities are grouped together using relations or classes. Entities in the same group have the same attributes. The descriptions for all the entities in a schema have the same defined format, predefined length, and follow the same order. Structured data have been very popular since the early days of computing and many organizations rely on relational databases to maintain very large structured repositories.

Levels of Semantics:

Semantics is the study of the meaning of signs, such as terms or words. Depending on the approaches, models, or methods used to add semantics to terms, different degrees of semantics can be achieved. The levels of semantics is shown in the following figure.



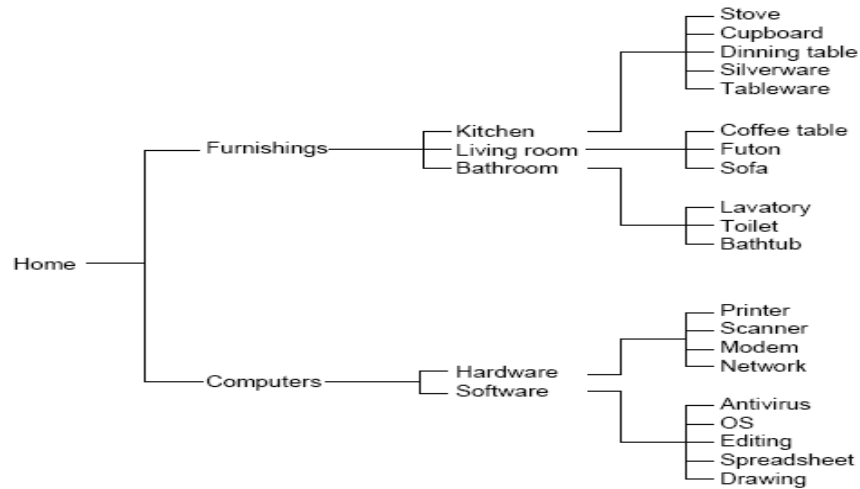
- A) Controlled vocabulary: A controlled vocabulary is a list of terms (e.g., words, phrases, or notations) that have been enumerated explicitly. All terms in a controlled vocabulary should have an unambiguous, non-redundant definition. For example, Amazon.com has the following (Table 1) controlled vocabulary which can be selected by the user to search for products.

Table 1. Controlled vocabulary used by Amazon.com

Books	Electronics	Travel
Popular Music	Camera & Photo	Cell Phones & Service
Music Downloads	Software	Outlet
Classical Music	Tools & Hardware	Auctions
DVD	Office Products	zShops
VHS	Magazines	Everything Else
Apparel	Sports & Outdoors	Scientific Supplies
Yellow Pages	Outdoor Living	Medical Supplies
Restaurants	Kitchen	Indust. Supplies
Movie Showtimes	Jewelry & Watches	Automotive
Toys	Beauty	Home Furnishings
Baby	Gourmet Food Beta	Lifestyle
Computers	Musical Instruments	Pet Toys
Video Games	Health/Personal Care	Arts & Hobbies

- B) Taxonomy: A *taxonomy* is a subject-based classification that arranges the terms in a controlled vocabulary into a hierarchy without doing anything further. The taxonomy arrangement in a home is shown in Figure.

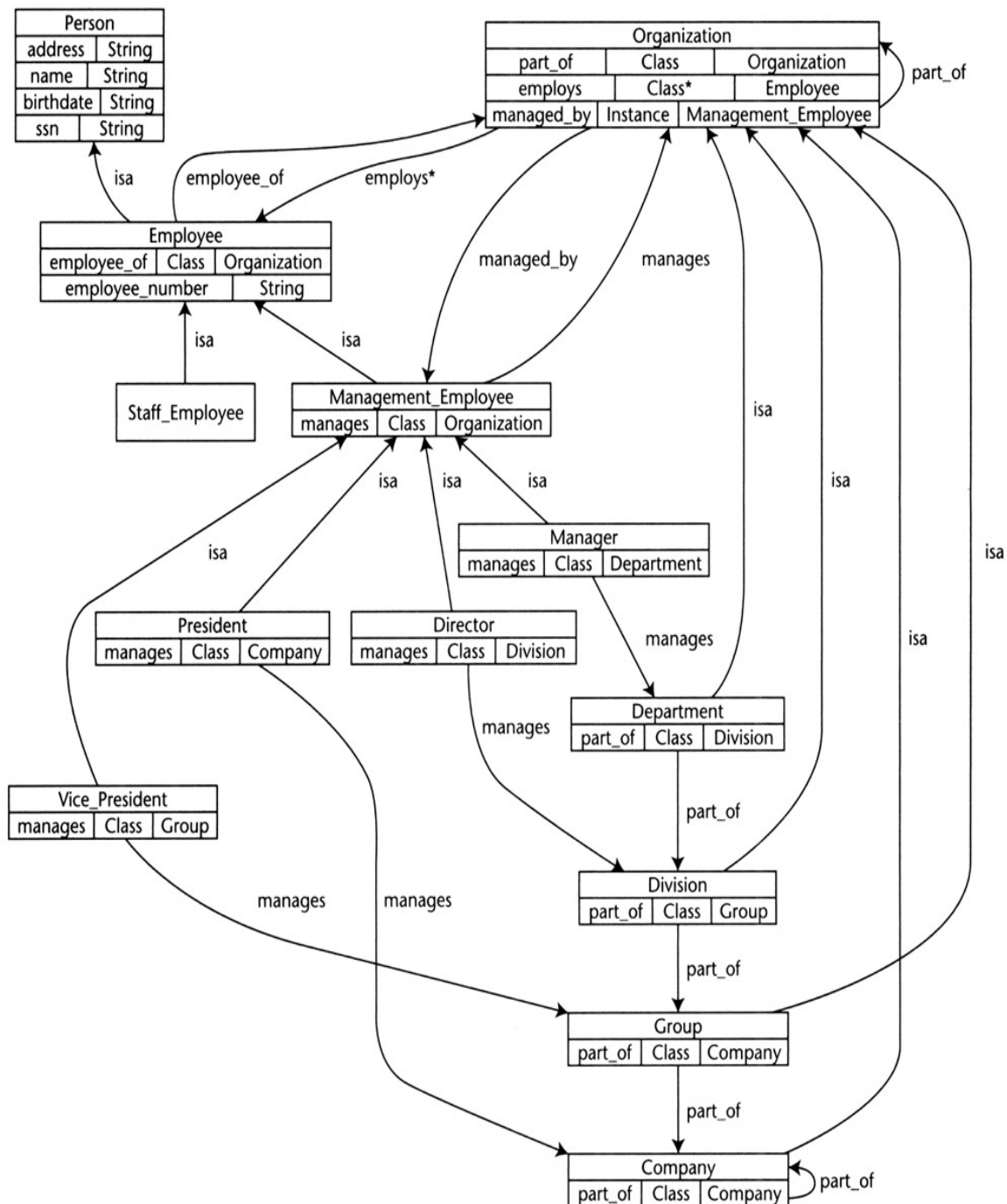
Figure 5. Example of a taxonomy



- C) Thesaurus : A thesaurus is a networked collection of controlled vocabulary terms with conceptual relationships between terms. A thesaurus is an extension of a taxonomy by allowing terms to be arranged in a hierarchy and also allowing other statements and relationships to be made about the terms. The following table shows the semantic relationships of a Thesaurus with suitable example.

SEMANTIC RELATION	DEFINITION	EXAMPLE
<i>Synonym</i> Similar to Equivalent Used for	A term <i>X</i> has nearly the same meaning as a term <i>Y</i> .	"Report" is a synonym for "document."
<i>Homonym</i> Spelled the same Homographic	A term <i>X</i> is spelled the same way as a term <i>Y</i> , which has a different meaning.	The "tank," which is a military vehicle, is a homonym for the "tank," which is a receptacle for holding liquids.
<i>Broader Than</i> (Hierarchic: parent of)	A term <i>X</i> is broader in meaning than a term <i>Y</i> .	"Organization" has a broader meaning than "financial institution."
<i>Narrower Than</i> (Hierarchic: child of)	A term <i>X</i> is narrower in meaning than a term <i>Y</i> .	"Financial institution" has a narrower meaning than "organization."
<i>Associated</i> Associative Related	A term <i>X</i> is associated with a term <i>Y</i> , i.e., there is some unspecified relationship between the two.	A "nail" is associated with a "hammer."

- D) Ontology: Ontologies are similar to taxonomies but use richer semantic relationships among terms and attributes, as well as strict rules about how to specify terms and relationships. Ontologies have generally been associated with logical inferencing and recently have begun to be applied to the semantic Web. An ontology is a shared conceptualization of the world. Ontologies consist of definitional aspects such as high-level schemas and assertional aspects such as entities, attributes, interrelationships between entities, domain vocabulary and factual knowledge—all connected in a semantic manner. The ontology example is given in Figure.

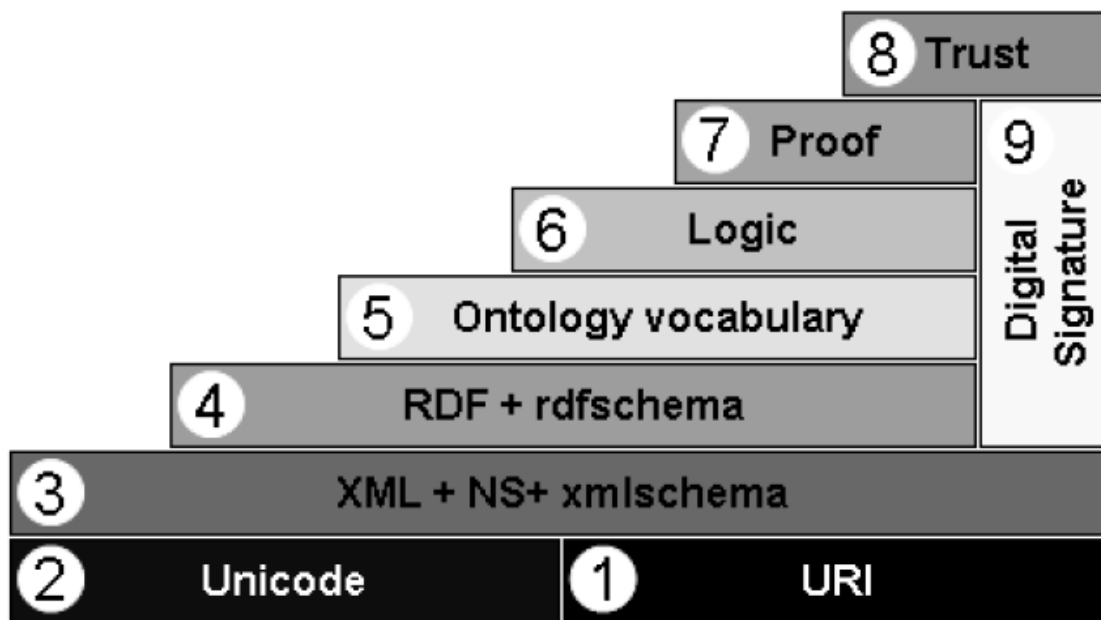


2. Semantic Web : Layered Architecture

Semantic Web is the new generation Web that tries to represent information such that it can be used by machines, not just for display purposes, but for automation, integration, and reuse across applications (Berners-Lee 2000). Furthermore, semantic Web is about explicitly declaring the knowledge embedded in many Web based applications, integrating information in an intelligent way, providing semantic based access to the Internet, and extracting information from texts.

Traditionally, HTML provides the standard of structured document published on the Internet. Though the simplicity of HTML promotes the growth of the Web, it seriously hampered advanced applications such as processing, understanding and semantic interoperability of information contained in several documents. Semantic Web is the new generation Web which makes possible to express information in precise, machine-interpretable form. It enables intelligent services such as information brokers, search agents and information filters, and also offers greater functionality and interoperability. Semantic Web promotes Web based applications with both semantic and syntactic interoperability. The explicit representation of meta-information, accompanied by domain theories (i.e. ontologies), will enable a Web to provide a qualitatively new level of service. This process may ultimately create extremely knowledgeable systems with various specialized reasoning services.

The architecture of semantic Web (W3C) is shown in Figure. The semantic Web technologies offer a new approach to managing information and processes, the fundamental principle of which is the creation and use of semantic metadata.



(i) URI

A universal resource identifier (URI) is a formatted string that serves as a means of identifying abstract or physical resource. A URI can be further classified as a locator, a name, or both. Uniform resource locator (URL) refers to the subset of URI that identifies resources via a representation of their primary access mechanism. An uniform resource name (URN) refers to the subset of URI that is required to remain globally unique and persistent even when the resource ceases to exist or becomes unavailable. For example:

- The URL [http://dme.uma.pt/jcardoso/index](http://dme.uma.pt/jcardoso/index.htm).htm identifies the location from where a Web page can be retrieved
- The URN urn:isbn:3-540-24328-3 identifies a book using its ISBN

(ii) Unicode

Unicode provides a unique number for every character, independently of the underlying platform, program, or language. Before the creation of Unicode, there were various different encoding systems. The diverse encoding made the manipulation of data complex. Any given computer needed to support many different encodings. There was always the risk of encoding conflict, since two encodings could use the same number for two different characters, or use different numbers for the same character. Examples of older and well known encoding systems include ASCII and EBCDIC.

(iii)XML and XML Namespace

XML (eXtensible markup language) with XML namespace and XML schema definitions makes sure that there is a common syntax used in the semantic Web. XML namespaces allow specifying different markup vocabularies in one XML document. XML schema serves for expressing schema definition of a particular XML document. When it comes to semantic interoperability, however, XML has disadvantages.

(iv)RDF and RDF Schema

On top of XML is the Resource Description Framework (RDF), for representing information about resources in a graph form. RDF is based on triples O-A-V that form a graph data with a relation among object (a resource), an attribute (a property), and a value (a resource).

RDF Schema (RDFS) defines the vocabulary of RDF model. It provides a mechanism to describe domain-specific properties and classes of resources to which those properties can be applied, using a set of basic modeling primitives (class, subclass-of, property, subproperty-of, domain, range, type). However, RDFS is rather simple and it still does not provide exact semantics of a domain.

(v) Ontology

Ontology comprises a set of knowledge terms, including the vocabulary, the semantic interconnections, simple rules of inference and logic for some particular topic. Ontologies applied to the Web are creating the semantic Web.

Ontologies facilitate knowledge sharing and provide reusable Web contents, Web services, and applications. Few of the ontology languages are DAML (DARPA Agent Markup Language), OIL (Ontology Interference Layer) and OWL (Web Ontology Language). OWL is developed starting from description logic and DAML+OIL. OWL is a set of XML elements and attributes, with well-defined meaning, that are used to define terms and their relationships (e.g. Class, equivalentProperty, intersectionOF, unionOF, etc.).

OWL elements extend the set of RDF and RDFS elements, and the OWL namespace is used to denote OWL encoding. OWL comes in three species – OWL Lite for taxonomies and simple constraints, OWL DL for full description logic support and OWL Full for maximum expressiveness and syntactic freedom of RDF. OWL DL is widely used for ontology representation.

In practice, ontologies are often developed using integrated, graphical, ontology authoring tools, such as Protégé, OILed and OntoEdit. Protégé facilitates extensible infrastructure and allows an easy construction of knowledge rich domain ontologies.

(vi) Logic, Proof, Trust and Digital Signature

The logic layer is used to enhance the ontology language further and to allow the writing of application-specific declarative knowledge.

The proof layer involves the actual deductive process as well as the representation of proofs in Web languages and proof validation.

Finally, the Trust layer will emerge through the use of digital signatures and other kinds of knowledge, based on recommendations by trusted agents or on rating and certification agencies and consumer bodies.

For the semantic Web to become more expressive enough to help in a wide range of situations, it will become necessary to construct a powerful logic language for making inferences. The next step in the architecture is ‘Trust’ and ‘Proof’. Very little is written about these layers though they will become important in future.

3. RDF Representation

At the top of XML, the World Wide Web Consortium (W3C) has developed the Resource Description Framework (RDF) language to standardize the definition and use of metadata. RDF is a simple general-purpose metadata language for representing information in the Web and provides a model for describing and creating relationships between resources. With RDF it is possible to add predefined modeling primitives for expressing semantics of data to a document without making any assumptions about the structure of the document.

The fundamental concepts of RDF are resources, properties and statements.

- Resources : A resource as an object, can be a thing such as person, a song, or a Web page. Every resource has URI to identify itself.
- Properties: They are a special kind of resources; they describe relations between resources, for example “written by”, “age”, “title” and so on.
- Statements: Statements assert the properties of resources. A statement is an object-attribute-value (O-A-V) tripe, consisting of a resource, a property and a value. Value can be either be resources or literals. It may also be represented as (Subject-Predicate-Object).

The basic structure of RDF is in the form of triples

Object/Resource/Thing $\leftrightarrow$ Property/Attribute $\leftrightarrow$ Value (is also a resource / literal)

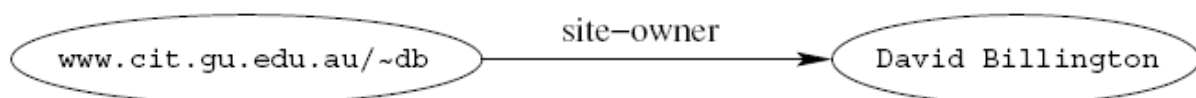
An example of a statement is

*David Billington is the owner of the Web page
<http://www.cit.gu.edu.au/~db>.*

The simplest way of interpreting this statement is to use the definition and consider the triple

(“David Billington”, <http://www.mydomain.org/site-owner>,
<http://www.cit.gu.edu.au/~db>).

The graphical RDF representation

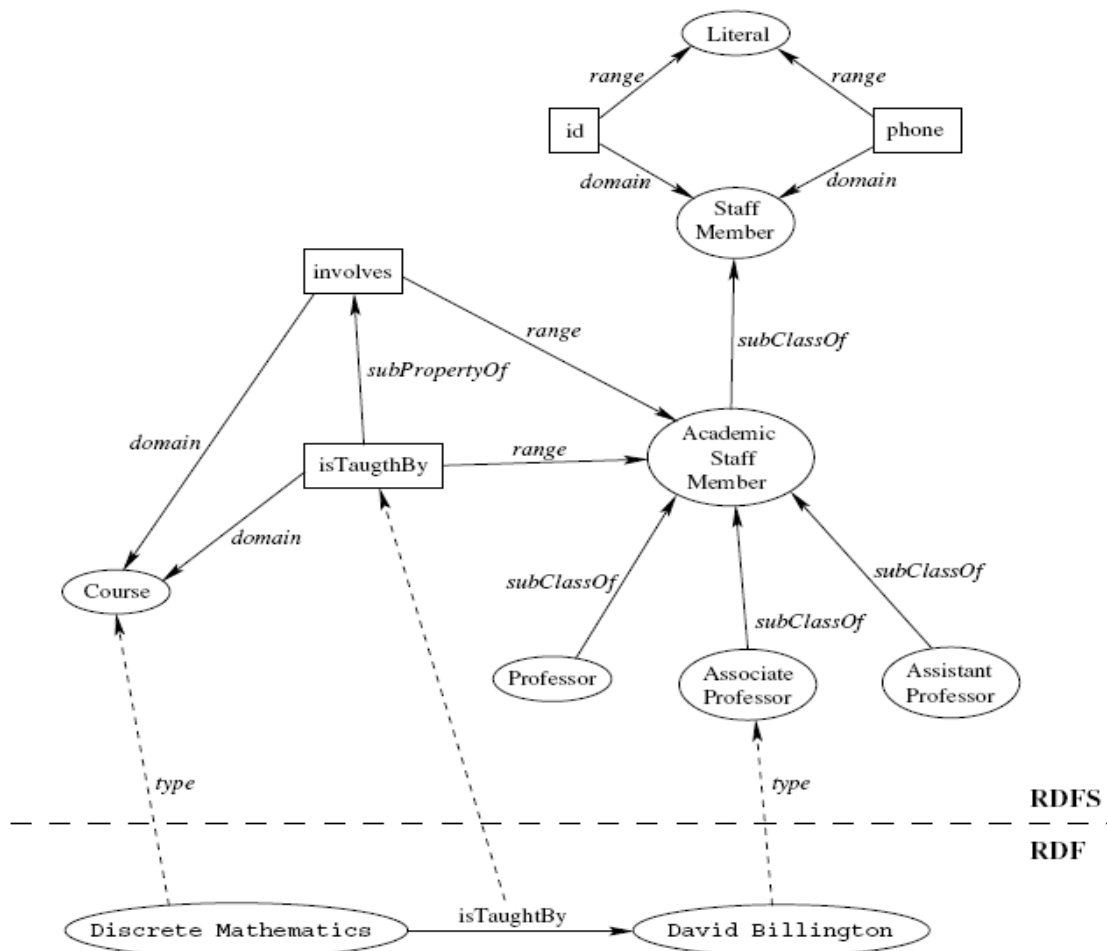


We can think of this triple (x,P,y) as a logical formula P(x,y), where the binary predicate P relates the object x to the object y. In fact, RDF offers only binary predicate (properties).

Consider the RDF statement:

Discrete Mathematics is taught by David Billington

The schema for this statement may contain classes such as lecturers, academic staff mebers, staff members, first-year courses, and properties such as is-taught by, involves, phone, employee id. In the given figure blocks are properties, ellipses above the dashed line are classes, and ellipses below the dashed line are instances.



Here is an example stating that all lecturers are staff members:

```
<rdfs:Class rdf:about="lecturer">
  <rdfs:subClassOf rdf:resource="staffMember"/>
</rdfs:Class>
```

Here is an example, stating that phone applies to staff members only and that its value is always a literal.

```
<rdf:Property rdf:ID="phone">
  <rdfs:domain rdf:resource="#staffMember"/>
  <rdfs:range rdf:resource="#&rdf;Literal"/>
</rdf:Property>
```

4. OWL Representation

The RDF and RDF schema expressions are limited to binary group predicates and limited to subclass hierarchy and property hierarchy with domain and range definitions of properties. The motivation of OWL (Web ontology language) is resulting nontrivial relation with RDF schema with a richer set of RDF and RDF schema.

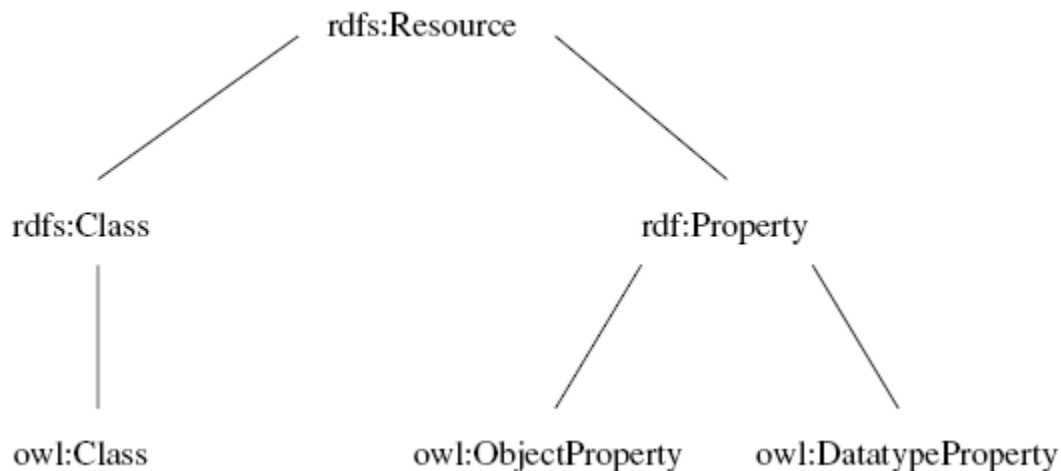
OWL ontology languages allow users to write explicit, formal conceptualization of domain models. They full fill the main requirements of ontology languages such as

- a well-defined syntax
- a formal semantics
- convenience of expression
- efficient reasoning support
- sufficient expressive power

The three species of OWL:

OWL comes in three species – OWL Lite for taxonomies and simple constraints, OWL DL for description logic support and OWL Full for maximum expressiveness and syntactic freedom of RDF. OWL DL is widely used for ontology representation.

The subclass relationship between OWL and RDF property:



OWL builds on RDF and RDF schema and uses RDF's XML based syntax. OWL can be defined with XML based syntax, Abstract syntax which makes use of RDF and Graphical Syntax.

Example 1:

Classes are defined using an `owl:Class` element.⁷ For example, we can define a class `associateProfessor` as follows:

```
<owl:Class rdf:ID="associateProfessor">
  <rdfs:subClassOf rdf:resource="#academicStaffMember"/>
</owl:Class>
```

We can also say that this class is disjoint from the `assistantProfessor` and `professor` classes using `owl:disjointWith` elements. These elements can be included in the preceding definition, or added by referring to the ID using `rdf:about`. This mechanism is inherited from RDF.

```
<owl:Class rdf:about="#associateProfessor">
  <owl:disjointWith rdf:resource="#professor"/>
  <owl:disjointWith rdf:resource="#assistantProfessor"/>
</owl:Class>
```

Equivalence of classes can be defined using an `owl:equivalentClass` element:

```
<owl:Class rdf:ID="faculty">
  <owl:equivalentClass rdf:resource="#academicStaffMember"/>
</owl:Class>
```

Here is an example of a datatype property:

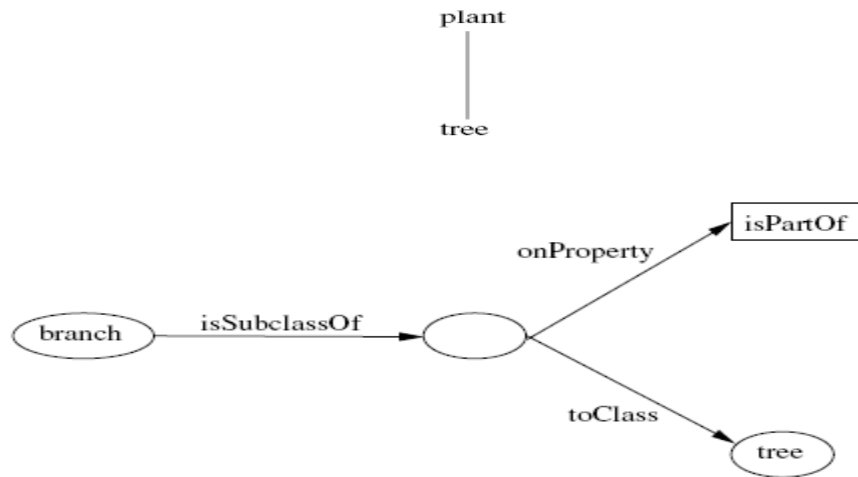
```
<owl:DatatypeProperty rdf:ID="age">
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema
    #nonNegativeInteger"/>
</owl:DatatypeProperty>
```

User-defined data types will usually be collected in an XML schema and then used in an OWL ontology.

Here is an example of an object property:

```
<owl:ObjectProperty rdf:ID="isTaughtBy">
  <rdfs:domain rdf:resource="#course"/>
  <rdfs:range rdf:resource="#academicStaffMember"/>
  <rdfs:subPropertyOf rdf:resource="#involves"/>
</owl:ObjectProperty>
```

Example 2:



```

<owl:Class rdf:ID="plant">
  <rdfs:comment>
    Plants form a class disjoint from animals.
  </rdfs:comment>
  <owl:disjointWith rdf:resource="#animal"/>
</owl:Class>

<owl:Class rdf:ID="tree">
  <rdfs:comment>Trees are a type of plant.</rdfs:comment>
  <rdfs:subClassOf rdf:resource="#plant"/>
</owl:Class>

<owl:Class rdf:ID="branch">
  <rdfs:comment>Branches are parts of trees.</rdfs:comment>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#is_part_of"/>
      <owl:allValuesFrom rdf:resource="#tree"/>
    </owl:Restriction>
  </rdfs:subClassOf>
</owl:Class>

<owl:Class rdf:ID="leaf">
  <rdfs:comment>Leaves are parts of branches.</rdfs:comment>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#is_part_of"/>
      <owl:allValuesFrom rdf:resource="#branch"/>
    </owl:Restriction>
  </rdfs:subClassOf>
</owl:Class>

```

Case Study

Develop XML Document, CSS, XSL, XSLT, XPATH for the following case studies:

1. Student Information System (personal and gradesheet XML document)
2. Tourist Information System (TourSeason, Sightseeing)