**Resource Description Framework, Extensible Markup Language and Ontology**

**Resource Description Framework (RDF)**

The Resource Description Framework (RDF) is a framework for representing information in the Web. RDF is a general framework for describing website metadata, or "information about the information" on the website. It provides interoperability between applications that exchange machine-understandable information on the Web. RDF details information such as a site's sitemap, the dates of when updates were made, keywords that search engines look for and the Web page's intellectual property rights.

The Resource Description Framework (RDF) is an infrastructure that enables the encoding, exchange and reuse of structured metadata. RDF is an application of XML that imposes needed structural constraints to provide unambiguous methods of expressing semantics. RDF additionally provides a means for publishing both human-readable and machine-processable vocabularies designed to encourage the reuse and extension of metadata semantics among disparate information communities. The structural constraints RDF imposes to support the consistent encoding and exchange of standardized metadata provides for the interchangeability of separate packages of metadata defined by different resource description communities.

RDF defines a simple, yet powerful model for describing resources. A syntax representing this model is required to store instances of this model into machine-readable files and to communicate these instances among applications. XML is this syntax. RDF imposes formal structure on XML to support the consistent representation of semantics.

The underlying structure of any expression in RDF is a collection of triples, each consisting of a subject, a predicate and an object. A set of such triples is called an RDF graph. Each triple represents a statement of a relationship between the things denoted by the nodes that it links. Each triple has three parts:

1. Subject,
2. Object, and
3. Predicate (also called a property) that denotes a relationship.



The direction of the arc is significant: it always points toward the object. The nodes of an RDF graph are its subjects and objects. The assertion of an RDF triple says that some relationship, indicated by the predicate, holds between the things denoted by subject and object of the triple.

**Extensible Markup Language (XML)**

XML is a markup language for documents containing structured information. A markup language is a mechanism to identify structures in a document. The XML specification defines a standard way to add markup to documents.

Structured information contains both content (words, pictures, etc.) and some indication of what role that content plays (for example, content in a section heading has a different meaning from content in a footnote, which means something different than content in a figure caption or content in a database table, etc.). Almost all documents have some structure.

XML specifies neither semantics nor a tag set like HTML. In fact XML is really a meta-language for describing markup languages. In other words, XML provides a facility to define tags and the structural relationships between them. Since there's no predefined tag set, there can't be any preconceived semantics. All of the semantics of an XML document will either be defined by the applications that process them or by stylesheets.

***Extensible***, XML is extensible. It lets you define your own tags, the order in which they occur, and how they should be processed or displayed. Another way to think about extensibility is to consider that XML allows all of us to extend our notion of what a document is: it can be a file that lives on a file server, or it can be a transient piece of data that flows between two computer systems (as in the case of Web Services).

***Markup****,* The most recognizable feature of XML is its tags, or elements (to be more accurate). In fact, the elements you’ll create in XML will be very similar to the elements you’ve already been creating in your HTML documents. However, XML allows you to define your own set of tags.

***Language***, XML is a language that’s very similar to HTML. It’s much more flexible than HTML because it allows you to create your own custom tags. However, it’s important to realize that XML is not just a language. XML is a meta-language: a language that allows us to create or define other languages. For example, with XML we can create other languages, such as RSS, MathML (a mathematical markup language), and even tools like XSLT.

**Ontology**

An ontology is a formal description of knowledge as a set of concepts within a domain and the relationships that hold between them. To enable such a description, we need to formally specify components such as individuals (instances of objects), classes, attributes and relations as well as restrictions, rules and axioms. As a result, ontologies do not only introduce a sharable and reusable knowledge representation but can also add new knowledge about the domain.

There are, of course, other methods that use formal specifications for knowledge representation such as vocabularies, taxonomies, thesauri, topic maps and logical models. However, unlike taxonomies or relational database schemas, for example, ontologies express relationships and enable users to link multiple concepts to other concepts in a variety of ways.

In computer science and information science, an ontology encompasses a representation, formal naming and definition of the categories, properties and relations between the concepts, data and entities that substantiate one, many or all domains of discourse.

Every field creates ontologies to limit complexity and organize data into information and knowledge. As new ontologies are made, their use hopefully improves problem solving within that domain. Translating research papers within every field is a problem made easier when experts from different countries maintain a controlled vocabulary of jargon between each of their languages.

**Ontology Web Language (OWL)**

The Ontology Web Language (OWL) was developed—or, more accurately, was derived from several earlier language initiatives—in order to provide a standardized way of representing ontologies on the semantic web.

OWL is a semantic web computational logic-based language, designed to represent rich and complex knowledge about things and the relations between them. It also provides detailed, consistent and meaningful distinctions between classes, properties and relationships. By specifying both object classes and relationship properties as well as their hierarchical order, OWL enriches ontology modelling in semantic graph databases, also known as RDF triples. Also, OWL comes equipped with means for defining equivalence and difference between instances, classes and properties. These relationships help users match concepts even if various data sources describe these concepts somewhat differently. They also ensure the disambiguation between different instances that share the same names or descriptions.

**RDF vs XML**

RDF and XML both attempt to address the problem of enabling different programs and computers to communicate effectively with each other. XML is a syntax and its data model is a tree. RDF is a data model based on a graph, that uses URIs and has several different syntax, including an XML syntax. Nevertheless, both XML and RDF can be used to represent structured data on the web and move data around between applications.

There are different ways of representing data in XML. In a way, the tree data model of XML is too flexible: you can say what you want to say in so many different ways. In order to understand what the XML document is about, an application needs the DTD (Document Type Definition) or the schema. RDF also has a schema, namely an ontology, represented in RDFS (RDF Schema) or OWL (Web Ontology Language). The ontology adds the semantics to your data and even allows you to infer new information from your current data (XML can’t do this).

The complexity of querying the XML tree is because there are in general a large number of ways in which the XML maps onto the logical tree, and the query you write has to be independent of the choice of them. So much of the query is an attempt to basically convert the set of all possible representations of a fact into one statement. This is just what RDF does. It gives you some standard ways of writing statements so that however it occurs in a document, they produce the same effect in RDF terms. The same RDF tree results from many XML trees.

**RDF vs RDFS vs OWL**

RDF: Straightforward representation, focused on the instances and on the mapping to their types (rdf:type). It is possible to define custom properties to link data and creating triples. RDF data are queried with SPARQL.

RDFS: Some situations are not easily modelled by RDF alone, it is sometimes interesting to represent more complex relations like subclasses (*the type of a type*) for example. RDFS provides special means to represent such cases, with constructs like rdfs:subClassOf, rdfs:range or rdfs:domain.For instance if you have the triples John a Man and Man rdfs:subClassOf Human then you should generate as well the triple John a Human. Note that this is not possible to do with RDF alone.

OWL: The highest level of expressivity. Relation between classes can be formally modelled based on description logics (mathematical theory). it is possible to express complex constructs such as chained properties for instance or restriction between classes. OWL serves to build ontologies or schema on the top of RDF datasets. OWL adds ontological capability to RDF (which on its own only provides extremely limited capability for formal knowledge representation), by providing the apparatus to define the components of your triple using formal computable first order description logic. In effect, this means that you can use an OWL ontology as a schema for RDF