ISSN 22773061

Mapping between XML and RDF via DTD

Harvinder

CSE/IT Department, Jaypee Institute of Information Technology
Noida, India

harvinder@jiit.ac.in

ABSTRACT

XML has gained massive popularity for representation of data in last decades. Both HTML and XML are used as primary data formats since the existence of World Wide Web. As the amount of data is growing, need has emerged to represent the data in such a manner that it can be recognized by machines rather than only humans. The next generation web, Semantic web aims at representing large amount of data available in www in a machine understandable format. However, XML represent the structure of data and cannot serve the purpose of machine understandability and HTML is too inexpressive to be used by machines for interpretation of data. To bridge this gap between syntax and semantics, current data needs to be converted into Semantic web compatible formats and XML DTDs can directly be converted into RDF. This paper focuses on conversion of data available in XML DTDs into RDF using classes, subclasses and properties.

Indexing terms/Keywords

XML, DTD, RDF, Semantic interoperability

Academic Discipline And Sub-Disciplines

Semantic web;Data Mining

SUBJECT CLASSIFICATION

Computer Science

TYPE (METHOD/APPROACH)

Quasi-Experimental; Literary Analysis

Council for Innovative Research

Peer Review Research Publishing System

Journal: INTERNATIONAL JOURNAL OF COMPUTERS & TECHNOLOGY

Vol 10, No 6

editor@cirworld.com
www.cirworld.com, member.cirworld.com

INTRODUCTION

The WWW has drastically changed the world into “One home for all”. Global connectivity is now extended to each and every corner of the world ,changing the way data used to be in past decades .Billions of computers are now connected to each other to share, consume, connect, publish and access ample amount of data.

Web page was initially only a simple media to carry data which a user can use to access data to grasp the required information. Now-a-days, Search engines have emerged impressively finding content from these web pages, making the life even more easier for a surfer. Despite of all these, a big shortcoming of current web is that the machines themselves cannot churn out the required information from a given set of data .To make data machine understandable and to make machines more capable, Tim Berner Lee proposed the concept of Semantic Web. 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." It is a source to retrieve information from the web (using the web spiders from RDF files) and access the data through Semantic Web Agents or Semantic Web Services. [1]

Most of the current web applications use XML and HTML formats to represent the data. While HTML can only be used for displaying and is too inefficient to be related to meaning of data, XML represents the structure of the data and can further be used for semantic purposes. The foremost reason for triumph of XML is its flexibility. Users are free to define their tags to describe entities in an XML document. However, these formats cannot be used for the required machine interpretation. For example an XML document might contain

< sports product > bat </sports product>

And another

< animal > bat < /animal>

A machine cannot distinguish between the two bats .There is a need to convert data in a machine understandable format such that machine is able to derive the difference between two statements and one such format is RDF. In addition, Semantic Web, not only requires the structured data but also the semantic content. Therefore, we cannot directly use XML data for the Semantic Web, and need another language to interpret a given document. RDF is clearly defined using URIs in such a hierarchical manner that it will be able to distinguish amongst the two words. [2] The remainder of the paper is organized as follows. In the next section various technologies related to the work are explained. Section III discusses the comparison of RDF and XML, along with the need of conversion of XML files into RDF files and why it is the need of hour to get the data in RDF format. Section IV concentrates on the method used to convert a DTD document into a RDF triplet. Finally, Section V concludes the paper.

RELATED TECHNOLOGIES

XML

XML is a tree like hierarchical structure used to represent data on web. XML is designed to transport and store the data keeping in focus that what the data is all about. It allows to build own new tags. And it’s a widely used format used to represent structure of data for information interchange on web. The reason for its world wide use is its ability to present data as a structure given by designer. The structure of a XML document is defined by XML schemas and DTDs.XML documents validated by XML schemas & DTDs are called “Well –formed documents. Though XML plays a crucial role in structuring the document, it has shortcoming when coming to the semantic interoperability. XML primarily focuses on the grammar, but there is no way to explain the semantics of the document.The XML file is represented as opening and closing tag. All the data is specified between both these tags. For example: <note>....</note>.

DTD

A Document Type Definition (DTD) defines the legal building blocks of an XML document. It defines the document structure with a list of legal elements and attributes. The two building blocks of DTD are element and attributes. An element specifies the structure of the XML document and an attribute specifies the extra information related to elements. An XML document, obeying the XML syntaxes, is called well-formed XML document. If a well-formed XML document is created based on the construction in a DTD or XML schema, it is called a valid XML document. Usually, XML schema and DTD are used as a standard mechanism to exchange information on the web. For example, in the electronic commerce, when the associates are unanimous in a common DTD, they will produce valid XML documents and carry out their communication. This provides us a large number of valid XML documents. Alternatively, users can draw DTD from a well-formed XML document by following its construction and labels. Otherwise, there is a tool helping to draw DTD from XML documents, such as DTDMaker. [3]

RDF

The Resource Description Framework (RDF) is a W3C standard for describing Web resources, such as the title, author, modification date, content, and copyright information of a Web page. RDF was designed to provide a common way to describe information so it can be read and understood by computer applications.RDF descriptions are not designed to be displayed on the web .[16] It is a tool to describe web resources .It decomposes the knowledge into small pieces defined as Subject-Predicate-Object triplet. Subject and Object are entities referred by URIs and Predicate shows the relationship between the two entities in real world.RDF can refer to anything represented in the world of data, be it s book or a coffee shop or a state of living somewhere or cooking something. RDF is a common method to decompose knowledge into minute pieces, with some rules about the semantics, or meaning, of those pieces. The point is to have a method so simple that it can express any fact, and yet so structured that computer applications can do useful things with knowledge expressed in RDF. [4]

RDF VS XML

In some ways, RDF can be compared to XML. XML represents data in a simple manner and is applicable to any type of data. XML represents the data in a hierarchical format which in turn become self-contained documents. What sets RDF apart from XML is that RDF is designed to represent knowledge in a distributed world. That RDF is designed for knowledge, and not data, means RDF is a particularly concerned with meaning. [4] To compare XML with RDF firstly, XML itself is not concerned with meaning of the nodes. XML nodes are not required to be associated with particular concepts, and the XML standard cannot be used to derive some new facts based on the current facts.RDF in contrast gives some basic meaning to a given set of data which may be used by any user throughout the world to derive certain information. Secondly, RDF very much supports distributed environment and decentralized nature of data. By linking the common vocabularies created by two different users, RDF can be used at every level. Drawing of information is thus not limited to a central application and diversified to whole internet. Contrarily, XML code is a kind of agreement between some parties and applications related to the same data server & a central application. This means that some specified applications connected to a central application can access the XML data. An outside application will not be able to understand the data specified by other applications in an application specific database.[13,14,16]

Need for XML to RDF conversion

1) **Semantic Interoperability** – As mentioned above, XML tags define the structure of the data and not the meaning associated with it. RDF supports the semantic structure of data as well as decentralized nature of data and RDF is a better candidate for semantic web. Therefore, to switch to a better & more meaningful web, conversion is necessary.

2) **XML schemas/DTDs are intricate towards any change** - Generally Validation of XML documents are done by either XML schemas or DTDs which implies that a minor change in an XML file has to be reflected in its validating document. It is very difficult to every time change the content of respective XML schema/DTDs to reflect the changes of XML file making it a tedious and lengthy task. In contrast, RDF has only one syntax which can be specified using URI to form a valid document.

3) **Ambiguity resolving** - RDF removes ambiguity as it is based on the concept of URIs. All the entities in RDF are URIs. A uniform resource identifier (URI) is a string of characters used to identify a name or a web resource.[5]A URI uniquely identifies any entity on web, all the entities represented as RDF-Triplet are expressed as URIs leading to a clear and crisp specification of any entity.XML however, lacks this ability as the data is application specific.

4) **Data management made easy**- Data on web is becoming voluminous day by day. Mankind cannot decipher such a huge amount of data efficiently after a certain limit. Data has to be organized in such a way that machines almost replace us for all kinds of data manipulation works. RDFs assist this vision by supporting meanings associated with data. Semantic implementation will make this dream real.[6]

5) **Difference in data standards**- Use of XML causes interoperability problems amongst different data standards.(Tags with same name in XML may have different meanings in various applications and vice versa).RDF has a common format all over www which makes it the best candidate to be used for semantic interoperability.

6) **Query accomplishment of RDF-** RDF is designed such that queries can be run using SPARQL (an RDF query language) which further add up to semantic interoperability. There is no such provision which entitles a user to query an XML file. The reason is simple, XML is designed keeping in mind the structure and valid syntax of data and not the semantics related with that data.

Above rules are applied on cheseserecipe.xml to illustrate how DTD can be used to form an RDF statement. The following table illustrates the RDF triplets. Once these triplets are designed, the meaning of all the above tags will remain same, no matter which application uses the RDF. The meanings can further be extracted easily by a machine omitting the human interference. This property of RDF supports very well the Semantic Web and its realization.

Converting DTD to RDF with example

A number of steps are followed for the necessary conversion. All the data given by DTD is classified according to the type it is representing in DTD. The algorithm first traces classes and subclasses and subsequently looks for the properties, attributes and values of the tags.

Repeat the following steps till closing tag of DOCTYPE is not encountered then RDF will have entries as follows:

1. If the given class is a root class then:

   Subject - URI of Class

   Predicate - rdf: hasResource

   Object - root-Class

This step will make RDF triplet for the root class.

There is only one root class for an XML file so the rdf:rootclass appears only once in the RDF file.

2. If any class is inside root class or any other Class then
   Subject - class name
   Predicate -rdf:hasClass OR hasSubClass
   (hasSubClass is used when a given class is parent other than root)
   Object - class (parent class)

For example-

<! ELEMENT cheeserecipe (description, recipe, title)>

Here, cheeserecipe is root class and description, recipe, title are its subclasses. 3. Any attribute is denoted as rdf:property and its corresponding value is given as rdf: value.If the data type, domain, range of any attribute is to be specified then it is given as rdf:datatype, rdf:domain, rdf: range respectively.The example used here is cheeserecipe.xml and its DTD was taken from “http://cs.au.dk/~amoeller/XML/schemas/dtd-example.html” and modified according to the requirements.

```
<? xml version="1.0"   encoding="UTF-8"?>

< cheeserecipe >

<title>Cottage cheese</title>

  <description>

     Recipes used for the XML tutorial

  </description>

  <recipe>

    <ingredient name="cottage cheese" amount="0.5"   unit="pound"/>

    ...

    <preparation>

      <step>

        Preheat oven to 350 degrees F (175 degrees C).

      </step>

      ...

    </preparation>

    <nutrition calories="1167" fat="23" carbohydrates="45" protein="32"/>

  </recipe>

  ...

</ cheeserecipe >
```

The above file is cheeserecipe.xml defining the recipe of cheese dish.

The file starts with tag cheeserecipe and ends with the closing tag. The root tag is used to create the root class of RDF file. The URI of the file will be given by the namespace path where this file will be located on web. The DTD given below is used to validate the above DTD. The dtd file confirms to the structure of XML file given.

Using this DTD file and algorithm above, the data in the file can be converted into a well defined RDF file.

```
<! DOCTYPE cheeserecipe [
<! ELEMENT cheeserecipe (description, recipe, title)>
<! ELEMENT title (#PCDATA)>
<! ELEMENT description ANY>
<! ELEMENT recipe (ingredient*, preparation, comment?, nutrition)>
<! ELEMENT ingredient (ingredient*, preparation)>
<! ATTLIST ingredient name CDATA #REQUIRED
                  amount CDATA #IMPLIED
                  unit CDATA #IMPLIED>
<! ELEMENT preparation (step*)>
<! ELEMENT step (#PCDATA)>
<! ELEMENT nutrition EMPTY>
<! ATTLIST nutrition protein CDATA #REQUIRED
                 carbohydrates CDATA #REQUIRED
                 fat CDATA #REQUIRED
                 calories CDATA #REQUIRED    ]>
```

Above rules are applied on cheseserecipe.xml to illustrate how DTD can be used to form an RDF statement. The following table illustrates the RDF triplets. Once these triplets are designed, the meaning of all the above tags will remain same, no matter which application uses the RDF. The meanings can further be extracted easily by a machine omitting the human interference. This property of RDF supports very well the Semantic Web and its realization.

TABLE I. ILLUSTRATING THE RDF FILE CORRESPONDING TO DTD

Subject	**Predicate**	**Object**
URI: http://...	Rdfs:Resource	cheeserecipe
cheeserecipe	rdf:hasclass	description
cheeserecipe	rdf:hasclass	recipe
cheeserecipe	rdf:hasclass	title
description	rdf:value	Recipe used for XML
recipe	rdf:hasSubclass	ingredient
ingredient	rdf:property	name
Name	rdf:value	Cottage cheese
ingredient	rdf:property	amount
amount	rdf:value	0.5
ingredient	rdf:property	unit
unit	rdf:value	pound
recipe	rdf:hasSubclass	preparation
preparation	rdf:property	steps
steps	rdf:domain	string
recipe	rdf:hasSubclass	nutrition
nutrition	rdf: property	calories
calories	rdf:value	1167
calories	rdf:domain	integer
nutrition	rdf: property	fats
fats	rdf:value	23
fats	rdf:domain	integer
nutrition	rdf: property	carbohydrates
carbohydrates	rdf:value	45
carbohydrates	rdf:domain	integer
nutrition	rdf: property	protein
protein	rdf:value	32
protein	rdf:domain	integer

CONCLUSION AND FUTURE WORK

An XML format is a traditional format which must be matched with the modern RDF format to make semantic web a reality. There are various approaches proposed for the conversion of XML data to RDFs. All schemes have their own advantages and disadvantages. Depending upon the kind of data, required conversion can be done. The above scheme focuses on XML DTD to RDF conversion. Later a technique may be designed so that a machine itself can parse and decide if the file used at the client side is XML schema or DTD and according to that convert the required data into RDF. This approach will create a bridge between DTDs and Semantic web. A large amount of RDF triples can be produced automatically using this approach. Those RDFs can further be used in many Semantic Web applications. In future, these conversion techniques may further be extended to onotology designing and Schema designing.

REFERENCES

[1] Tim Berners-Lee, “Semantic Web – A guide to the future XML web services and knowledge management”, Weaving the Web, Harper San Francisco, 1999.

[2] Michael Klein, “Interpreting XML documents via an RDF Schema ontology”, Proceedings of the 13th International Workshop on Database and Expert Systems Applications, 2002, pp. 889-893 Michael Erdmann, Rudi Studer, “How to structure and access XML documents with Ontologies”, April 2000.

[3] Frank Manola, Eric Miller, “RDF Primer”, W3C Recommendation,February 2004, available at:http://www.w3c.org/TR/REC-rdf-syntax/

[4] James Hendler, and Ora Lassila, "The Semantic Web",Scientific American, 2001

[5] Pham Thi Thu Thuy, Young-koo lee, and SungYoung Lee, “Transforming Valid XML Documents into RDF via RDF Schema”, Third International Conference on Next Generation Web Services Practices, Seoul, 2007, pp. 35-40.

[6] Bert Bos, “The XML data model”, http://www.w3.org/XML/ Datamodel html, 2005.

[7] Tim Berners_Lee, “A strawman Unstriped syntax for RDF in XML”, http://www.w3.org/DesignIssues/Syntax.html, W3C, 2007.

[8] Stefan Decker, Sergey Melnik, Frank Van Harmelen, Dieter fensel, Michel Klein, Jeen Broekstra, Michael Erdmann, and Ian Horrocks, “The Semantic Web: The roles of XML and RDF”, IEEE, 2000.

[9] Sergey Melnik, “Bridging the gap between RDF and XML”, Dec 1999.

[10] Jonathan Boden, “Simplified XML syntax for RDF”, June 2001, available at:http://www.openhealth.org/RDF/RDFSurfaceSyntax.html

[11] Dan Brickley, R.V. Guha, and Brian McBride, “RDF Vocabulary description language 1.0: RDF schema”, W3C, 10 Feb 2004, available at http://www.w3.org/TR/2004/REC-rdf-schema-20040210

[12] Tim Berners_Lee, “Why RDF model is different from the XML model”, Sep 1998, available at: http://www.w3.org/DesignIssues/RDF-XML.html.

[13] G. Klyne and J. J. Carroll, editors. Resource Description Framework (RDF): Concepts and Abstract Syntax. W3C Recommendation. World Wide Web Consortium, February 2004.

[14] Tim Berners-Lee, James Handler, and Ora Lassila, “The Semantic Web: A new form of Web content that is meaningful to computers will unleash a revolution of new possibilities”, Scientific American, 2001

[15] Pronab Ganguly, Fethi A. Rabhi and Pradeep K.Ray, “Bridging Semantic Gap”, Third Asian Pacific Conference on Pattern languages of Program, 2002

[16] Peter Patel-Schneider and Jérôme Siméon, “The Yin/Yang Web: XML syntax and RDF Semantics”, The Eleventh International World Wide Web conference, Honolulu, Hawaii, May 2002.

[17] Tim BERNERS-LEE, W. HALL, J. HENDLER, N. SHADBOLT, et al. Creating a science of the web Science, 313(5788):769–771, 2006a.