An Overview of RDF Model Representation Formats

Vladimír Grlický

Slovak University of Technology
Faculty of Informatics and Information Technologies
Ilkovičova 3, 842 16 Bratislava, Slovakia
grlicky@fiit.stuba.sk

Abstract. The Resource Description Framework (RDF) is a way how to model ontologies (i.e. concepts and relationships among these concepts) in the Semantic Web. Generally, RDF is a collection of triples, each consisting of a subject, a predicate and an object. This set of triples is also known as RDF graph (or RDF graph data model). Besides its graphical representation RDF model can be also represented textually. In this paper we present an overview of several the most used formats of serialization of this model. We also mention advantages and disadvantages of using these formats.

1 Introduction

RDF – the Resource Description Framework – is a foundation for processing metadata; it provides interoperability between applications that exchange machine-understandable information on the web. Although World Wide Web Consortium (W3C) recommended as a normative representation of RDF model the RDF/XML format, this one is not the only one representation of RDF model. In this paper we briefly mention four currently mostly used formats for RDF model representation (namely RDF/XML, Notation 3, N-Triples, and TriX). To demonstrate some of features of these formats we will try to encode following two sentences: “Full name of a person identified by http://semweb.grlicky.info/vgrlicky is Vladimir Grlicky. The name of this person consists of given name Vladimir and family name Grlicky”.

On this simple example we show that the same model can be represented not only by RDF/XML, but by other alternative formats that have minimally the same
expressive capabilities as RDF/XML. But, before we start exploring RDF representation formats, we briefly describe basic features of the RDF.

2 Basics of RDF

RDF (Resource Description Framework) is a standardized way how to describe resources located on the web as well as real-world objects (e.g. a paper book, a car) that are not directly accessible via the services of the Internet. Until 2003 RDF was specified by only one document – Model and Syntax (M&S) Specification [10]. The RDF M&S specification defined an abstract data model (the “Model” part of the specification). The model is abstract because it is defined in terms of abstract mathematical structures such as triples and sets. This specification also defined how to represent data conforming to this data model in XML (the “Syntax” part of the specification). RDF/XML – the XML serialization recommended by W3C is a representation of the abstract model. However, it is important to distinguish between the abstract data model and its representations. The specifications define constraints which apply to the abstract data model as a whole. As the abstract model is infinite, representations of the abstract data model may be finite and incomplete. Since February 2004 the RDF specification (revised version) is defined by a “set of six”, what stands for the set of six documents composed of Primer [11], Concepts (and Abstract Syntax) [9], Syntax [2], Semantics [8], Vocabulary (RDF Schema) [5], and Test Cases [7].

RDF data model is based on linguistically inspired construction composed of a triple of the form subject – predicate – object, often called a statement or an assertion. Universal element of RDF is a resource that is unambiguously identified by its URI (Uniform Resource Identifier). A resource in a statement can appear in the role of both a subject and an object. Predicates correspond to properties of subjects, objects refer to values of these properties. Object can have also the form of a literal, i.e primitive data type (Fig. 1).

![Graph representation of a subject-predicate-object relationship](image)

Important feature of RDF model is reification. The notion of reification originates from latin “res”- “thing”, what means materialization, and it is used for transformation of statement to atomic object. Reification adds to RDF model the ability to take the entire statement as a resource and to include it in other statements. Reification can be applied when reasoning about creditibility of individual statements on the basis of trust of information source. Such “meta-statements” have the same quality (are on the same level of importance) as basic statements, and we need not create special syntactic
constructions. On the other side the reification can reasonably complicate formal features of RDF model.

Other capabilities of RDF (typed literals, support for containers, collections, entailment etc.) are not described here but can be found in the “set of six” that is mentioned in above paragraphs.

Fig. 2. RDF model representation using directed edge-labeled graph

Abstract data model of RDF does not specify particular representation. In case of its graphical form a data model can be depicted as a graph; speaking more precisely, it is directed edge-labeled graph where subjects and objects are nodes, properties are drawn as arcs between nodes. Fig. 2 depicts the set of free text sentences, introduced as the example in section 1, represented by this type of graph. When we need the model to be represented textually, we can use RDF/XML, or one of its alternative formats briefly described in following sections.

3 RDF/XML

An RDF graph can be considered a collection of paths of the form node-predicate arc-node which cover the entire graph. In order to encode the RDF graph in XML, the nodes and predicates have to be represented in XML terms – element names, attribute names, element contents and attribute values. Therefore, in RDF/XML these turn into sequences of elements inside elements which alternate between elements for nodes and predicate arcs. This has been called a series of node/arc stripes. The node at the start of the sequence turns into the outermost element; the next predicate arc turns into a child element, and so on. The stripes generally start at the top of an RDF/XML document and always begin with nodes.

The following code presents aforementioned features of RDF/XML when describing the RDF graph shown in Fig. 2.
4 Notation 3

Notation 3 [3, 4], more commonly known via its abbreviation “N3”, is a shorthand non-XML serialization of RDF, designed with human-readability in mind. According to the Notation 3 Specification [3], it was created as an experiment in optimizing the “expression of data and logic in the same language”. N3 has been well received for its “scribblability”, because it is much more compact and readable than RDF/XML, and because it forms a good introduction into many key principles of the Semantic Web. The code below represents RDF model, depicted in Fig. 2, encoded using N3.

@prefix : <#> .

<http://semweb.grlicky.info/vgrlicky>
  <http://www.w3.org/2001/vcard-rdf/3.0#FN> "Vladmir Grlicky";
  <http://www.w3.org/2001/vcard-rdf/3.0#N> [ <http://www.w3.org/2001/vcard-rdf/3.0#Family> "Grlicky";
    <http://www.w3.org/2001/vcard-rdf/3.0#Given> "Vladimir" ] .

As its current disadvantages we can mention the long term stability of N3, which is not known, there are some internationalization issues, and a notable lack of analogies for various features of N3 in RDF/XML.

5 N-Triples

As stated in [1], N-Triples is a line-based, plain text format for representing the correct answers-statements for RDF. This format was designed to be a fixed subset of N3 (briefly described in section 4), hence all tools, which currently work with N3, can seamlessly work with it, too. Each triple must be written on a separate line, and consists of a subject specifier, a predicate specifier, then an object specifier, followed
by a period. One or more spaces or tabs separate subject from predicate, and predicate from object.

```
_:BN01
  <http://www.w3.org/2001/vcard-rdf/3.0#Given>
  "Vladimir" .
  <http://semweb.grlicky.info/vgrlicky>
  <http://www.w3.org/2001/vcard-rdf/3.0#N>
  _:BN01 .
  _:BN01
  <http://www.w3.org/2001/vcard-rdf/3.0#Family>
  "Grlicky" .
  <http://semweb.grlicky.info/vgrlicky>
  <http://www.w3.org/2001/vcard-rdf/3.0#FN>
  "Vladimir Grlicky" .
```

The source code above represents exemplary RDF graph model (Fig. 2) using N-Triples format. Because of its plain text line-oriented nature N-Triples is simple and very easy machine-processable RDF representation format.

6 TriX

TriX (Triples in XML) [6] is a serialization for named graphs. Its creators tried to provide a highly normalized and consistent XML representation of RDF model (i.e. graphs), which allows the effective use of generic XML tools as XSLT, XQuery and others. The RDF model (Fig. 2) represented in TriX is shown below:

```
<trix xmlns="http://www.w3.org/2004/03/trix/trix-1/">
  <graph>
    <uri>http://semweb.grlicky.info/graph1</uri>
    <triple>
      <id>BN01</id>
      <uri>http://www.w3.org/2001/vcard-rdf/3.0#Given</uri>
      <plainLiteral>Vladimir</plainLiteral>
    </triple>
    <triple>
      <uri>http://semweb.grlicky.info/vgrlicky</uri>
      <uri>http://www.w3.org/2001/vcard-rdf/3.0#N</uri>
      <id>BN01</id>
    </triple>
    <triple>
      <id>BN01</id>
      <uri>http://www.w3.org/2001/vcard-rdf/3.0#Family</uri>
      <plainLiteral>Grlicky</plainLiteral>
    </triple>
  </graph>
</trix>
```
One of the main advantages in Trix over RDF/XML is the ability to name graphs, syntactic extensibility, direct processing with XML tools (XSLT, XQuery, XPath) while still satisfying the well-formedness and validity of XML documents.

7 Conclusions

In this paper we tried to show that a set of sentences, modeled using RDF, can be represented not only by RDF/XML but also by other RDF representation formats. We briefly described some of alternatives to RDF/XML (N3, N-Triples, and TriX). These formats can encode RDF model without any loss of information, when compared with RDF/XML. However, it is important to say that this overview provided no formal proof of assertions stated here. When choosing “the right” RDF representation format, one should pay attention especially to its application support and capabilities to share and exchange ontologies modeled by RDF and represented by this format.

Acknowledgement: This work was supported by Science and Technology Assistance Agency under the contract No. APVT-20-007104.

References


