RDF2RDFa: Turning RDF into Snippets for Copy-and-Paste

Martin Hepp
E-Business & Web Science Research Group, Universität der Bundeswehr
Werner-Heisenberg-Weg 39
D-85579 Neubiberg, Germany
+49-89-6004-4217
mhepp@computer.org

Roberto García
Computer Science and Engineering Department
Universitat de Lleida
Jaume II, 69, E-25001 Lleida, Spain
+34-973-702-742
roberto@rhizomik.net

Andreas Radinger
E-Business & Web Science Research Group, Universität der Bundeswehr
Werner-Heisenberg-Weg 39
D-85579 Neubiberg, Germany
+49-89-6004-4218
andreas.radinger@unibw.de

ABSTRACT

In this demo and poster, we show a conceptual approach and an on-line tool[1] that allows the use of RDFa for embedding non-trivial RDF models in the form of invisible div/span elements into existing Web content. This simplifies the publication of sophisticated RDF data, i.e. such that goes beyond simple property-value pairs, by broad audiences. Also, it empowers users with access limited to inserting XHTML snippets within Web-based authoring systems to add fully-fledged RDF and even OWL. Such is a frequent limitation for users of CMS systems or Wikis.

Categories and Subject Descriptors

I.2.4 **[Artificial Intelligence]:** Knowledge Representation Formalisms and Methods

General Terms

Design, Human Factors, Standardization, Languages

Keywords

RDFa, RDF/XML, Resource Description Framework, CMS, Wiki, HTML, XHTML

1. INTRODUCTION

For many potential users of Semantic Web technology, publishing rich RDF meta-data is a serious technical difficulty. For example, the current best practices of publishing RDF require at least sufficient server privileges to upload files via FTP and, in the case of "slash" URIs for non-information resources, access to the server configuration at the level of .htaccess (on Apache servers). Such is not available to a large share of the individuals who contribute most of the content of today's Web. Many inexpensive hosting services do not grant access to the .htaccess file, and users of content management systems (CMS), Wikis, or of proprietary platforms like eBay auction or vendor pages are at most able to insert HTML/XHTML snippets into Web resources.

Now, with the RDFa syntax for embedding RDF data in XHTML attributes being a W3C Recommendation, there is a standard way of adding RDF to Web content by inserting additional mark-up [1]. However, the current usage of RDFa in the community is dominated by (1) using simple property-value pairs rather than complex graph structures and (2) a close coupling between page content for rendering and the literals attached to properties. For example, a typical recipe would be to augment a phone number in a page by making it the literal attached to the vcard:tel property:

```
<body>
 <div property="vcard:tel" datatype="xsd:string">+49-89-6004-0
 </div>
</body>
```

The key reason for the popularity of this approach is that there is no data redundancy, i.e. what is shown in a browser is always identical to what an RDF-aware application will extract.

While this is appropriate for very lightweight annotations, it becomes very complicated if (1) more sophisticated RDF models are to be embedded or (2) the content or organization of the information for humans on one hand and for machines on the other hand differ. Also, the interweaving of existing Web content for humans with non-trivial RDF models requires a lot of expertise, in particular if many nodes in the RDF model have no visual counterparts. In those cases, the initial goal of avoiding data redundancy clashes with the goal of the separation of concerns, and the XHTML+RDFa markup gets hard to read and difficult to maintain because it closely couples presentation and data. For examples, see http://www.ebusiness-unibw.org/wiki/Rdfa4google. Most of all, it is not possible to provide users with XHTML snippets to be simply inserted into Web resources, without the need to manually integrate them with existing XHTML markup.

Now, RDFa by design allows differences between the literals used as property values and the literals being displayed using the "content" attribute [1], e.g.

```
<body>
<div property="vcard:tel" datatype="xsd:string" content="+49-89-6004-0">
+49-89-6004 ext. 0
</div>
</body>
```

This is particularly useful if the formatting of the data for humans and machines differs, e.g. in the case of date and time information ("2009-04-24T00:00:00+01:00"). It is possible to exploit this to create XHTML snippets that just contain the meta-data and insert it for instance at the bottom of the page:

```
<body>
<!-- Content for humans -->
<div>+49-89-6004-0</div>
<!-- RDFa rich meta-data -->
<div property="vcard:tel" datatype="xsd:string" content="+49-89-6004-0">
</div>
</body>
```

The potential advantages of this approach are that (1) we disentangle the markup and that (2) respective snippets for simple copy-and-paste can be provided by form-based tools like FOAF-a-Matic [2]. As compared to publishing a separate RDF/XML file on the server, the advantages are that (1) RDFa data is considered by Yahoo! SearchMonkey and other services, (2) one still has to

[1] http://www.ebusiness-unibw.org/tools/rdf2rdfa/

maintain a single file only (reducing the likelihood of outdated, forgotten meta-data files), (3) the content creator does not require access beyond being able to edit the page. Also, note that literal values will often have to be encoded in RDFa “content” attributes anyway, because the string for the presentation is not suitable as meta-data content (e.g. dates or country codes).

In a nutshell, the proposed approach can be a powerful way of publishing non-trivial RDF meta-data suitable for broad audiences. Imagine e.g. if eBay sellers were able to put detailed GoodRelations [3] meta-data directly into the free markup part of their product description in the system.

2. RDF2RDFa TOOL

In many cases, RDF data for a given resource is either available or can be created easily in the RDF/XML or N3 syntaxes. For that purpose we developed the RDF2RDFa tool which translates RDF/XML files into XHTML/RDFa snippets for simple copy-and-paste into existing Web content. The tool is available both as a Web application and as a REST-style Web service. In the demo, we will show the tool and its usage for augmenting CMS and Wiki pages with sophisticated RDF data, and the invocation of the service from Python.

Internally, the RDF2RDFa tool uses an XSLT transformation, which was derived from the one for the RDF2HTML service from the ReDeFer project (htt://rhizomik.net/redefer/). The original transformation (RDF2HTML) converts the input RDF into XHTML+RDFa with both human-readable and machine-readable mark-up. For RDF2RDFa, the transformation was modified so that the whole RDF model provided will be properly represented in the attributes of one XHTML snippet of otherwise empty div/span elements. The first "div" element is the binding clip for the whole snippet and holds all the namespace declarations.

The details of the transformation are outside the scope of this demo and poster abstract, but the full XSLT is available at http://rhizomik.net/redefer-services/xsl/rdf2rdfa.xsl.

2.1 Web-based Tool

The Web-based tool is available from http://www.ebusiness-unibw.org/tools/rdf2rdfa/ and offers the specification of a URI or direct input of RDF/XML content. It returns an equivalent XHTML/RDFa snippet suitable for copy-and-paste plus instructions on how to add that to an existing page.

2.2 REST-style Web Service

The RDF2RDFa tool can also be directly accessed from applications in a REST style by sending an HTTP GET request to the URI

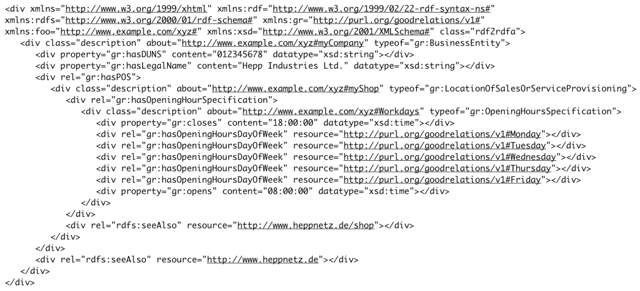
http://rhizomik.net/redefer-services/rdf2rdfa?rdf=RDFURL

with "RDFURL" being the escaped URI of a RDF/XML file to be converted. This directly returns a RDFa snippet, representing the submitted RDF content as a block of div/span elements without visible content.

3. EXAMPLE

In the following we give an example of the proposed approach. Assume that we want to add the RDF data shown below to a company’s Web page (namespace prefixed omitted). The respective RDFa output created by the RDF2RDFa tools is then as in Figure 1.

```
foo:myCompany
 a gr:BusinessEntity ;
 gr:hasLegalName "Hepp Industries Ltd."^^xsd:string ;
 gr:hasDUNS "012345678"^^xsd:string ;
 gr:hasPOS foo:myShop ;
 rdfs:seeAlso <http://www.heppnetz.de>.
foo:myShop
 a gr:LocationOfSalesOrServiceProvisioning ;
 rdfs:seeAlso <http://www.heppnetz.de/shop> ;
 gr:hasOpeningHourSpecification foo:Workdays.
foo:Workdays
 a gr:OpeningHoursSpecification ;
 gr:opens "08:00:00"^^xsd:time ;
 gr:closes "18:00:00"^^xsd:time ;
 gr:hasOpeningHoursDayOfWeek
   gr:Monday, gr:Tuesday, gr:Wednesday, gr:Thursday, gr:Friday .
```



```
<div xmlns="http://www.w3.org/1999/xhtml" xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#" xmlns:gr="http://purl.org/goodrelations/v1#"
xmlns:foo="http://www.example.com/xyz#" xmlns:xsd="http://www.w3.org/2001/XMLSchema#" class="rdf2rdfa">
   <div class="description" about="http://www.example.com/xyz#myCompany" typeof="gr:BusinessEntity">
      <div property="gr:hasDUNS" content="012345678" datatype="xsd:string"></div>
      <div property="gr:hasLegalName" content="Hepp Industries Ltd." datatype="xsd:string"></div>
      <div rel="gr:hasPOS">
         <div class="description" about="http://www.example.com/xyz#myShop" typeof="gr:LocationOfSalesOrServiceProvisioning">
            <div rel="gr:hasOpeningHourSpecification">
               <div class="description" about="http://www.example.com/xyz#Workdays" typeof="gr:OpeningHoursSpecification">
                  <div property="gr:closes" content="18:00:00" datatype="xsd:time"></div>
                  <div rel="gr:hasOpeningHoursDayOfWeek" resource="http://purl.org/goodrelations/v1#Monday"></div>
                  <div rel="gr:hasOpeningHoursDayOfWeek" resource="http://purl.org/goodrelations/v1#Tuesday"></div>
                  <div rel="gr:hasOpeningHoursDayOfWeek" resource="http://purl.org/goodrelations/v1#Wednesday"></div>
                  <div rel="gr:hasOpeningHoursDayOfWeek" resource="http://purl.org/goodrelations/v1#Thursday"></div>
                  <div rel="gr:hasOpeningHoursDayOfWeek" resource="http://purl.org/goodrelations/v1#Friday"></div>
                  <div property="gr:opens" content="08:00:00" datatype="xsd:time"></div>
               </div>
            </div>
            <div rel="rdfs:seeAlso" resource="http://www.heppnetz.de/shop"></div>
         </div>
      </div>
      <div rel="rdfs:seeAlso" resource="http://www.heppnetz.de"></div>
   </div>
</div>
```

Figure 1. RDFa snippet of the example

4. EVALUATION AND CONCLUSION

The resulting snippets can be properly extracted from Web pages (often even if the page itself is no proper XHTML), e.g. using the RDFa Distiller [4]. We were also able to insert respective snippets into a standard MediaWiki with just RDFa permissions activated via a simple patch. In a nutshell, the approach presented promises to empower broader audiences to publish sophisticated RDF data as part of their Web presence. It will also allow the creation of novel form-based tools that produce useful snippets for many application areas. As a first success of our initiative, Google changed the wording of their rich snippets recipe at http://knol.google.com/k/google-rich-snippets/google-rich-snippets/32la2chf8l79m/1#, removing the previous statement that hiding meta-data was against Google’s content policies.

Acknowledgments: The authors thank Mark Birbeck, Toby Inkster, Kavi Goel, Othar Hansson, Kingsley Idehen, and everybody who contributed to the original discussion on semantic-web@w3.org, for useful feedback on the initial idea, and Andreas Wechselberger for testing our snippets in MediaWiki.

5. REFERENCES

[1] RDFa in XHTML: Syntax and Processing. A collection of attributes and processing rules for extending XHTML to support RDF. W3C Recommendation 14 October 2008, available at http://www.w3.org/TR/2008/REC-rdfa-syntax-20081014.

[2] FOAF-a-Matic, available at http://www.ldodds.com/foaf/foaf-a-matic.

[3] Hepp, M.: GoodRelations: An Ontology for Describing Products and Services Offers on the Web. 16th International Conference on Knowledge Engineering and Knowledge Management (EKAW2008), Acitrezza, Italy, Springer LNCS Vol. 5268, 2008, pp. 332-347.

[4] RDFa Distiller, available at http://www.w3.org/2007/08/pyRdfa