

# Making URIs published on Data Web RDF dereferencable

Jing Mei  
IBM China Research Lab  
Zhongguancun Software Park  
Beijing 100193, China  
meijing@cn.ibm.com

Shengping Liu  
IBM China Research Lab  
Zhongguancun Software Park  
Beijing 100193, China  
liusp@cn.ibm.com

Guotong Xie  
IBM China Research Lab  
Zhongguancun Software Park  
Beijing 100193, China  
xieguot@cn.ibm.com

Hanyu Li  
IBM China Research Lab  
Zhongguancun Software Park  
Beijing 100193, China  
lihanyu@cn.ibm.com

Yuan Ni  
IBM China Research Lab  
Zhongguancun Software Park  
Beijing 100193, China  
niyuan@cn.ibm.com

Yue Pan  
IBM China Research Lab  
Zhongguancun Software Park  
Beijing 100193, China  
panyue@cn.ibm.com

## ABSTRACT

Nowadays, more and more URIs reside on Data Web, as published for linked open data, dereferencing URIs challenges the current Web to embrace Semantic Web. Although, quite a few practical recipes for publishing URIs have been provided to make URIs dereferencable, we believe a fundamental investigation of publishing and dereferencing URIs would contribute a forward compatibility with the RDF and OWL upper layers in the Semantic Web architecture. In this paper, we propose to make URIs published on Data Web RDF dereferencable, and we formalize such a requirement in an RDF-compatible semantics. Also, the dereferencing operation is defined in an abstract URI syntax, such that URIs, as interpreted as described resources, would be RDF dereferencable by default. Accompanied by a live demonstration, the poster demo explanation would elaborately discuss and seriously address issues on Data Web URIs, which were or have been taken for granted. Additionally, for case study, Metadata Web, a Data Web of enterprise-wide models, is explored. The URIs on Metadata Web is published as RDF dereferencable. Such an implementation of universal metadata management across the enterprise enables the metadata federation such that global query, search and analysis could be conducted on top of the Metadata Web.

## 1. INTRODUCTION

RDF (Resource Description Framework) is intended to provide a simple way to make statements about Web resources. A typical example is, as shown below in an RDF triple, `http://www.example.org/index.html` has a creator whose value is John Smith as identified by a staff ID 85740.

```
<http://www.example.org/index.html>  
<http://purl.org/dc/elements/1.1/creator>  
<http://www.example.org/staffid/85740> .
```

To respect the Web architecture [1], a Web resource (a web page) is identified by URI `http://www.example.org/index.html`. However, in the Data web context, a big challenge that is encountered is how to guarantee the above RDF triples are always retrieved when dereferencing this URI. We call it as RDF dereferencable, and its formal definition would be given in the later section. Since, web pages are information resources which could be directly dereferenced, dereferencing the above URI generally retrieves the HTML web page, rather than any RDF triple. Practice recipes from [5] and [3] also Cool URIs [7] instructed us a bit, such as using 303 redirect and content negotiation for dereferencing a URI which identifies a non-information resource. In this way, GET `http://www.example.org/index.html` with an `Accept: application/rdf+xml` header would be redirected to another URI like `http://www.example.org/index.html/data`, and then to get `http://www.example.org/index.html/data` for the RDF triples.

Again, being a URI, `http://www.example.org/index.html/data` identifies a Web resource, and we are allowed to make statements about it. Below is an example, which is (most possibly) not an authoritative description, if such a statement made by others than the owner of the URI.

```
<http://www.example.org/index.html/data>  
<http://www.w3.org/2000/01/rdf-schema#comment>  
"Please publish me and try to dereferencing me".
```

Interestingly, an RDF dereferencing result of the first URI has triggered to dereferencing the second URI, which retrieves the first RDF triple. In other words, directly dereferencing the second URI does not retrieve anything about itself, and again practice recipes such as using 303 redirect and content negotiation have to be applied for retrieval of the second RDF triple.

As a consequence, the so-called RDF dereferencability not only needs to be well-defined, but also needs to guarantee the retrieval of RDF triples is what delivered on Data Web. Otherwise, the failure of consuming RDF triples hurts the RDF data providers, and vice versa, the failure of providing RDF triples dismisses the RDF data consumers. Nowadays, various URIs are residing on Data Web, as the W3C SWEO Linking Open Data community project <sup>1</sup> proudly an-

<sup>1</sup><http://esw.w3.org/topic/SweoIG/TaskForces/CommunityProjects/LinkingOpenData>

nounced, special for using the recipes in [5, 7] which would introduce at least three URIs to describe a resource. Since, not all published URIs are RDF dereferencable, it is quite the time to do a fundamental investigation of publishing and dereferencing URIs on Data Web. Below, we propose a necessary and sufficient condition for making URIs published on Data Web RDF dereferencable.

- Necessary Condition: If a URI published on Data Web is RDF dereferencable, then dereferencing this URI retrieves RDF triples with subject of this URI.
- Sufficient Condition: A URI published on Data Web is RDF dereferencable, only if publishing this URI delivers RDF triples with subject of this URI.

Recalling to the RDF Semantics, there was no assumption of any particular relationship between the denotation and use of a URI, and such a requirement could be added as a semantic extension [2]. To some extent, satisfiability of the above necessary and sufficient condition is likely a required relationship, where publishing a URI is the denotation and dereferencing a URI is the use. We believe, URI, being a cornerstone of the Semantic Web, needs a forward compatibility with the RDF and OWL upper layers. In this paper, we would contribute an RDF-compatible semantics for making URIs published on Data Web RDF dereferencable. Also, the dereferencing operation is defined in an abstract URI syntax, such that URIs, as interpreted as described resources, would be RDF dereferencable by default.

## 2. RDF DEREFERENCABLE

By convention in the RDF Semantics [2], a set of names is referred to as a vocabulary, and a name is a URI reference or a literal. As specified in the generic URI syntax [4], a URI reference is either a URI or a relative reference.

**Definition 1:** A des-interpretation of a vocabulary  $V$  is a simple interpretation<sup>2</sup>  $I$  of  $V$ , extending with: (1) A set  $IR_d \subseteq IR$ , described resources; (2) A mapping  $IDES : IR_d \rightarrow 2^{V \times V \times V}$ , the resource description mapping, s.t.,  $\langle s p o \rangle \in IDES(I(s))$ , for any  $s \in \{u \in V | I(u) \in IR_d\}$ ,  $p \in V$  and  $o \in V$ . A URI  $u \in V$  is defined as RDF dereferencable if  $I(u) \in IR_d$ , i.e., interpreting  $u$  by a described resource.

Similar to rdf-interpretations and rdfs-interpretations, every des-interpretation is also a simple interpretation. The ‘extra’ description structure does not prevent it acting in the simpler role.

Given a URI published on Data Web, if it is RDF dereferencable, then dereferencing this URI should retrieve RDF triples with subject of this URI. On the contrary, if publishing this URI has delivered RDF triples with subject of this URI, then it is RDF dereferencable. By definition, we call the former as a necessary condition and the latter as a sufficient condition, for making URIs published on Data Web RDF dereferencable.

Following up, we formalize resource representation by definition of the dereferencing operation. First, we recall the generic URI syntax [4] which defines a grammar that is a superset of all valid URIs, consisting of a hierarchical sequence of components referred to as the scheme, authority,

path, query, and fragment. As for Data Web, conventionally, only HTTP URIs are used, to avoid other URI schemes such as URNs and DOIs [5]. Below is the generic syntax of HTTP URI, and HTTP URIs are called query URIs if containing a “?” [6].

```
http_URL = "http:" "/" host [ ":" port ]
           [ abs_path [ "?" query ] ]
```

In abstract syntax, we define the set of all valid HTTP URIs by  $U_0$ . A query URI  $v \in U_0$  is defined in the form of  $u?q$ , where  $u$  is the non-query part of  $v$ , and  $q$  is the query part. Besides,  $q$  consists of parameters (key/value pairs) using & for separator, viz.  $k_1 = v_1 \& \dots \& k_m = v_m$ , where  $k_i$  is the parameter name and  $v_i$  is the parameter value,  $1 \leq i \leq m$ .

**Definition 2:** Let  $U_0$  be the set of URIs,  $G$  the set of RDF graphs,  $F$  the set of representation formats,  $S$  the set of byte streams and  $G \subseteq S$ . A dereferencing operation is defined by  $\lambda : U_0 \rightarrow S$ . As well, a format transformation is defined by  $\tau : G \times F \rightarrow S$ .

Taking advantage of parameters in query URIs, we propose to publish URIs for described resources with the non-query form, so that they would be RDF dereferencable by default, while other URIs with suffix of parameters would be dereferenced in a usual way. That is, resource description is formalized by *IDES* and resource representation by  $\lambda$ , such that, given a URI  $u \in U_0$ , if interpreting  $u$  by a described resource  $I(u) \in IR_d$ , then  $\lambda(u) = IDES(I(u))$ . Any other query URI would be dereferenced with format transformation, i.e.,  $\lambda(u?k = v) = \tau(IDES(I(u)), v)$ .

Besides, such a strategy would benefit the paging implementation. As noted in [5] and [8], retrieval of a huge stream of bytes challenges the bandwidth. Now, parameterized pages are configurable in URIs to retrieve a specified page like <http://mdw.com/resource/Beijing?format=html&page=3>.

## 3. REFERENCES

- [1] Architecture of the World Wide Web, December 2004. <http://www.w3.org/TR/webarch/>.
- [2] RDF Semantics, February 2004. <http://www.w3.org/TR/rdf-mt/>.
- [3] Best Practice Recipes for Publishing RDF Vocabularies, January 2008. <http://www.w3.org/TR/swbp-vocab-pub/>.
- [4] T. Berners-Lee, R. T. Fielding, and L. Masinter. Uniform Resource Identifier (URI): Generic Syntax, January 2005. <http://gbiv.com/protocols/uri/rev-2002/rfc2396bis.html>.
- [5] C. Bizer, R. Cyganiak, and T. Heath. How to Publish Linked Data on the Web, July 2007. <http://sites.wiwiw.fu-berlin.de/suhl/bizer/pub/LinkedDataTutorial/>.
- [6] R. T. Fielding, J. Gettys, J. Mogul, H. F. Nielsen, L. Masinter, P. Leach, and T. Berners-Lee. Hypertext Transfer Protocol – HTTP/1.1, June 1999. <http://www.ietf.org/rfc/rfc2616.txt>.
- [7] L. Sauer mann, R. Cyganiak, and M. Volkel. Cool URIs for the Semantic Web, August 2007. <http://www.dfki.uni-kl.de/sauer mann/2006/11/cooluris/>.
- [8] F.-P. Servant. Linking Enterprise Data. In *WWW 2008 Workshop: Linked Data on the Web*, April 2008.

<sup>2</sup>We direct readers to RDF Semantics [2] for definition of the simple interpretation