Framing the URI Resource Identity Problem: The Fundamental Use Case of the Semantic Web

David Booth, Ph.D.
david@dbooth.org
Draft 2012-12-05 - Comments invited

Latest version of this document:
http://dbooth.org/2012/fyn/Booth-fyn.pdf (PDF)

Abstract. In the Semantic Web, URIs are used as names for "resources" – things in the universe of discourse. But what resource does a given URI denote? How is its identity determined? The issue of URI resource identity has plagued the Semantic Web community for over a decade. Discussions drift into philosophical debate because they are insufficiently grounded in appropriate use cases. This paper describes the fundamental use case of the Semantic Web: that a semi-autonomous agent should be able to sensibly merge two RDF datasets that were authored independently, using common URIs to join related information. It then examines the issue of URI resource identity from the engineering perspective of addressing this fundamental use case, and explains why this use case is more appropriate in framing the resource identity problem than a use case of sending and receiving a message. It also explains key requirements for standardizing the "follow your nose" convention for locating a URI definition – a URI definition discovery protocol – and their relevance to URI owners, RDF authors and RDF consumers.

Keywords: Semantic web architecture, resource identity, URI definition, RDF semantics, URI Definition Discovery Protocol, UDDP, follow-your-nose

1 Introduction

The Architecture of the World Wide Web (AWWW) states that "By design, a URI identifies one resource"[1]. As in the RDF Semantics, the word *resource* is used herein to mean "anything in the universe of discourse"[2]. The issue of *URI resource identity* – the association between a URI (as a name) and a particular resource – is the question of establishing and determining what resource a given URI identifies. This issue has plagued the Semantic Web community since at least the beginning of the great httpRange-14 debate[3] over a decade ago – a debate that is closely related to the issue of URI resource identity.

A major reason this issue has been so hard to resolve is that discussions are often insufficiently grounded in appropriate, concrete use cases, and thus technical analyses

drift into subjective philosophical debate. One key purpose of this paper is therefore to define what is the fundamental use case of the Semantic Web. The paper then analyzes the issue of URI resource identity from the architectural engineering perspective of addressing this use case. It explains key requirements for standardizing the "follow your nose" convention for locating a URI definition – a URI definition discovery protocol – and their relevance to URI owners, RDF authors and RDF consumers. Although details of such a protocol must be worked out by a group process, this paper provides the rationale for such standardization.

For historical reasons this paper uses the term "URI" (Uniform Resource Identifier[4]) instead of the more general term "IRI" (Internationalized Resource Identifier[6]), but the principles discussed apply equally to IRIs.

1.1 The Fundamental Use Case of the Semantic Web

The purpose of the Semantic Web is to enable machines to usefully combine and process web information.[6] Since the Semantic Web uses RDF[7] to represent machine processable information, RDF datasets may be authored independently, and RDF uses URIs as names for resources, the fundamental use case of the Semantic Web can be summarized as follows:

> *A semi-autonomous agent should be able to sensibly merge two RDF datasets that were authored independently, using common URIs to join related information.*

By "semi-autonomous agent" we mean a software agent or application that is acting on a user's behalf, at the user's direction. By "sensibly merge" we mean that the RDF merge[8] is logically consistent according to its RDF semantics, and (loosely speaking) a URI that was used in both RDF datasets was used in each dataset to identify essentially the same resource, so such a merge is meaningful. By "authored independently" we mean that the authors do not communicate with each other and have no knowledge of each other. By "common URIs" we mean that some URIs are used in common between the two datasets, i.e., some of the same URIs are used in both datasets. This paper is not concerned with the question of *which* URIs are used in common or how such URIs are obtained – the OKKAM project[27] is one effort to address that question – but merely assumes that somehow such URIs *are* used in common by independent authors.

2 Framing the Problem

Within W3C Technical Architecture Group (TAG) discussions, the URI resource identity problem has often been framed as a problem of URI *meaning* and *communication*. For example, in his "HTTP Use Cases" document[9], TAG member Jonathan Rees writes: "There is a 'sender' writing a 'message'", and the problem is

how to ensure “that the receiver can discover the sender's meaning”, and that meaning is based on the meaning of each URI in the message. However, as the next two subsections explain, from a web architectural perspective it is better to frame the problem in terms of URI *definitions* and *merging data*.

2.1 URI Definitions – Not Meaning

It is intuitively enticing to frame the problem in terms of URI meaning, because the merge of two RDF datasets would not make sense if the meaning of a URI were completely different in each dataset. For example, if one dataset used a URI to denote the *tall building* known as the Eiffel Tower, and we were interested in that building, we would not want the same URI to denote the Eiffel Tower *metro stop* in the other data set, because RDF statements about the metro stop would get confused with statements about the tall building when we merged the two datasets. In spite of this intuitive appeal, from an engineering perspective it is better to frame the problem in terms of URI *definitions*, as we will explain.

By *URI definition* we mean a sequence of characters that indicates what the URI means (according to that URI definition). For the purposes of this paper, the language, format and effectiveness of the definition are unimportant, nor does it matter whether the definition is expressed in a formal or informal language. In the Semantic Web a URI definition is commonly an RDF document containing RDF assertions involving the URI whose meaning is being defined. Such a document might also be called a *description* – indeed, it is a kind of description – but *definition* is more specific and helps convey the document's intent. Furthermore, just as a term in English can have different definitions according to different people, so too a URI can have different definitions. Thus, a key concern in this paper is the question of how independent parties can obtain the *same* definition for a URI.

There are several important reasons why it is more advantageous in web architecture to frame the URI resource identity problem in terms of URI definitions rather than URI meaning.

Meaning is a philosophical tar pit. First, framing the problem in terms of meaning causes the analysis of the engineering problem to be far deeper, more subtle and more daunting than it otherwise needs to be. It also causes discussions to be murkier and less grounded than they otherwise need to be, thus leading both to more emphasis on subjective opinion and to more miscommunication. The great httpRange-14 debate[10] raged for thousands of email messages, largely because the concrete engineering requirements were so unclear. 'Nuff said?

The myth of unique reference. Second, when the problem is framed in terms of the meaning of a URI, there is an implicit assumption that a URI has, or should have, only one meaning. Presumably this assumption is rooted in the W3C Architecture of the World Wide Web (AWWW) assertion that “By design, a URI identifies one resource”.[11] However, this assumption is misleading in a practical sense, a theoretical sense and a technical sense:

- In a practical sense, we have no way of knowing or determining what meaning an application or RDF consumer may give to a particular URI. As an extreme example, a drug smuggler might use http://example/shirt as a code word in an RDF statement to mean *heroin* even though other RDF authors may use that same URI to mean *shirt*. Even in cases that are not intentionally misleading, it is clear that a URI is not always interpreted with the same meaning. For example, one application may use mailto:david@dbooth.org to identify an email destination, for email delivery. But another application may that same URI to identify the *person* who owns that email address. Some may claim that this is an example of indirect identification[12], but when we consider the fact that the application uses that URI in exactly the same way that it uses the URI http://t-d-b.org/?http://dbooth.org/2005/dbooth/ , which was specifically minted to *directly* identify that same person, it is clear that the mailto:david@dbooth.org is actually being used by that application to directly identify a person rather than a mailbox. As another example, Jeni Tennison notes how a Flickr URI is sometimes used to refer to a photograph, and other times used to refer to a web page describing the photograph, and suggests that we treat this as a form of punning[13].
- In a theoretical sense, it is generally impossible to be completely unambiguous about the referent of a name, whether that name is a URI or any other kind of name. The theory of reference – the relation between a name and the thing to which it refers – has been deeply studied and debated in philosophy[28], and the only web-scalable candidate we have for establishing a referent's identity is description. But "description is inherently ambiguous"[29], essentially because one can always create or discover ever finer distinctions than a description anticipated.
- In a technical sense, RDF does not assign a unique meaning to a URI. RDF deals only with sets of assertions and constraints on the ways they might be interpreted. As the RDF Semantics document notes: "It is usually impossible to assert enough in any language to completely constrain the interpretations to a single possible world, so there is no such thing as 'the' unique interpretation of an RDF graph."[14]

Meaning is untestable. A third difficulty with framing the problem in terms of meaning is that we have no objective means of verifying whether the message recipient actually did obtain the sender's intended meaning. In contrast, it is trivially easy to compare two URI definitions character by character to see whether they are the same.

Meaning is irrelevant to web architecture. Finally, there is no *need* to frame this architectural problem in terms of meaning, since the architecture can separate the problem of *obtaining* a URI definition from the problem of *interpreting* that definition – a clean separation of concerns. This is not to say that meaning of a URI definition does not matter at all, but simply that it does not matter *to web architecture*. Just as the HTTP specification[15] has no need to talk about the meaning of the content conveyed in an HTTP message, our fundamental Semantic Web use case can be

framed in terms of URI definitions without delving into their meanings. An architectural solution to address this use case merely needs to ensure that cooperating parties can obtain the same URI definitions without communicating with each other. The format, language and interpretation of those definitions can be left to other application-level specifications to address if they choose, just as HTTP merely needs to provide the protocol hooks such as Content-Type and Content-Encoding headers to allow other layers to specify how those HTTP messages should be interpreted. In short, the engineering problem is much clearer and simpler if we frame it in terms of URI definitions instead of URI meaning.

2.2 Merging Independently Authored Data – Not Communication

Obviously our fundamental Semantic Web use case involves some communication, but framing the URI resource identity problem in terms of a communication between two parties is misleading, as it fails to account for a critical element of this fundamental use case. If the problem were merely one of communication between an RDF author and an RDF consumer then the RDF author could simply choose his/her desired URI definition, use some common convention to tell the RDF consumer what definition was used (perhaps as message metadata), and – *voila* – the problem would be solved. Indeed, there would be no need for the role of URI owner in web architecture! But such a solution would be insufficient to address our fundamental Semantic Web use case, because this use case requires that two RDF authors, *acting independently*, should be able to use URIs according to the same URI definition, i.e., *without communicating with each other*. Thus, one author cannot simply tell the other author what definition was used. The two authors *must* use a common convention to obtain the same URI definition, and this involves (and is the purpose of) the role of URI owner, as explained below.

3 Problem Scenario: RDF Authors, RDF Consumer and URI Owner

This section expands and illustrates our fundamental Semantic Web use case in terms of a hypothetical scenario involving RDF authors (Arthur and Aster), an RDF consumer (Connie) and a URI owner (Owen):

> *Arthur and Aster are RDF authors. Arthur publishes RDF data about tall buildings, including the Eiffel Tower. Aster publishes RDF data on the number of tourists who visit famous landmarks each year, including the Eiffel Tower. Arthur and Astor work completely independently and know nothing of each other's work. Nonetheless, they wish when possible to use common URIs according to the same definitions, so that other parties can sensibly merge the RDF data that they publish, without Arthur's or Aster's assistance or*

knowledge. For example, instead of each minting their own URIs for the Eiffel Tower, Arthur and Aster wish to use a common URI that has already been minted for this purpose, and a common definition for this common URI. Furthermore, they wish to automatically obtain and inspect the definition to ensure that the URI identifies the tall building known as the Eiffel Tower as opposed to the metro stop of that same name, so that they can use the URI in a manner that is consistent with that definition.

Connie is an RDF consumer who wants to show the correlation between the heights of tall buildings and the number of tourists who visit them. She discovers both Arthur's RDF data and Aster's RDF data and wants her application to merge that data. Connie's application should also automatically obtain the definition of the Eiffel Tower's URI, so that Connie can verify that her application is displaying information on the correct notion of the Eiffel Tower: the tall building, not the metro stop.

A key requirement of this use case is that Arthur, Aster and Connie all want to use the same definition for this Eiffel Tower URI, even though Arthur and Aster have no knowledge of each other or of Connie. Since the parties do not coordinate directly with each other, this clearly requires the use of a common convention. **This need for a common convention is the key reason why such a convention should be standardized.**

One possibility for such a convention might be for the parties to use a central, world-wide URI dictionary with globally agreed definitions. But on the scale of the world-wide web, such a centralized approach would be both impractical and undesirable for social reasons. Thus, to address this need in a more decentralized way, the web architecture introduces a fourth role in this scenario – the role of *URI owner*[16]:

Owen is a URI owner who has minted a URI (within his URI space) that identifies the Eiffel Tower – the tall building – and has written a URI definition for this URI. Owen does not know who might use his URI or his URI definition, but he wants them to be useful to others who wish to make RDF statements about the Eiffel Tower, so he wishes to publish his URI definition in a way that allows others to retrieve it automatically, given only his Eiffel Tower URI.

4 Follow-Your-Nose and the Need for a URI Definition Discovery Protocol

One way the above scenario might be addressed would be for every RDF author to explicitly indicate the location of the URI definition that was used for each URI, perhaps by use of an owl:import or rdfs:isDefinedBy statement. However, as RDF data is mixed, selected and remixed, such statements can easily get disassociated from the the data to which they were attached. For this reason, and for brevity, there is

therefore still a community desire for conventions that allow a URI's definition to be located given only that URI.

Historically, the conventions used to address the above scenario have been based on the widely used but informal practice known as *follow your nose*. ***Follow your nose (FYN)*** [17] means dereferencing a URI (after stripping off any #fragment identifier to obtain its *stem*) to locate information about the resource identified by that URI. This practice is one of the defining principles of *Linked Data*[18] and has been in use since at least 2002[19]. Although it has been widely used to locate information *about* a URI's resource, users do not always view such information as *defining* the URI's resource. *URI Declaration in Semantic Web Architecture*[20] attempted to document and explain the practice of using FYN to obtain a URI definition, and *The URI Lifecycle in Semantic Web Architecture* [21] proposed a set of roles and responsibilities associated with URI owners and RDF statement authors. *Cool URIs for the Semantic Web*[22] provided publishing guidance to URI owners, based in part on the httpRange-14 resolution[23], but the exact mechanism for obtaining a URI definition, given only the URI, remained informal.

However in early 2012 TAG member Jonathan Rees solicited proposals to formalize what can be termed a *URI definition discovery protocol* to supersede the httpRange-14 resolution and provide clearer, more standardized conventions for obtaining a URI definition given only the URI.[24] Several proposals were received, [25]. How should such proposals be evaluated? This is discussed next.

5 Requirements for a URI Definition Discovery Protocol

First and foremost, a URI definition discovery protocol (or formalization of *follow your nose*) must adequately address the fundamental use case of the Semantic Web, as elaborated in the above scenario involving Arthur, Aster, Connie and Owen. Since the scenario involves cooperation between three roles – URI owner, RDF author and RDF consumer – the responsibilities of all three roles must be clearly specified:

- A URI owner needs to know what conventions to follow in minting a URI and hosting its definition.
- RDF authors need to know what conventions to follow in using URIs the RDF data that they publish, e.g., to ensure that they are using each URI consistently with the URI owner's definition.
- RDF consumers need to know what conventions to follow to obtain a URI's definition.

In short, a standard URI definition discovery protocol must meet the following criterion: ***Given an RDF graph, an agent should be able to algorithmically locate the URI definition that the RDF author used when authoring that graph, provided that the URI owner and the RDF author followed all best practices specified by the URI definition discovery protocol.*** However, this does *not* mean that every attempt to obtain the URI's definition of a URI must succeed, as the next section explains.

6 Using Market and Social Forces to Accommodate Failure

It is easy to see that if Owen, Arthur or Aster fails to follow the URI definition discovery protocol, or if Owen serves different URI definitions to Arthur, Aster and/or Connie – perhaps because of changing the definition over time – then Connie may end up with garbage when attempting to merge or interpret Arthur's and Aster's RDF data. For example:

- Owen may fail to follow the protocol in minting his Eiffel Tower URI or in hosting its definition, either deliberately or accidentally, thus causing Arthur, Aster and/or Connie to misinterpret the definition.
- Arthur, Aster and/or Connie may fail to follow the protocol in obtaining Owen's URI definition, thus causing them to misinterpret the definition.
- Arthur and/or Aster may use Owen's URI in a way that is inconsistent with Owen's URI definition, thus causing their data to mean something different than Connie thought it meant.

One may be tempted to assume that such a broad potential for failure would render a URI definition discovery protocol useless, but it does not. To understand why not, consider the net effect on Connie (the RDF consumer). From Connie's perspective, the result of these failures is indistinguishable from what Connie would see if Owen, Arthur and/or Aster had published bad or useless data. Connie may be disappointed that one or more of these other parties had published garbage (or so it appeared to Connie), but it does not break the web or cause any architectural difficulties.

The web is designed so that “anybody can say anything about anything”[26]. This is a feature, not a bug. The web architecture allows people to publish garbage, and it is up to the marketplace to ignore the garbage and reward the good stuff. In the Semantic Web this translates into rewarding URI owners and RDF authors who follow standard conventions and publish stable, quality URI definitions and data, and ignoring the noise introduced by those who either do not follow the conventions or who publish junk.

URI owners and RDF authors who play nicely together by following established protocols and publishing quality data will become more popular, and those who don't will be shunned. If protocols are standardized, whether de jure or de facto, there will be social pressure to conform to them.

This reliance on the market and social pressure -- instead of trying to solve the problem of bad publishers or bad data at the architectural level -- is one of the brilliant aspects of the web's architecture.

7 Conclusions

Key points:

- The fundamental use case of the Semantic Web involves merging two RDF datasets that were authored independently.
- The architectural problem that this use case poses should be framed in terms of URI definitions instead of URI meaning, and in terms of merging independently authored data instead of communication.
- Standardizing a URI definition discovery protocol (UDDP) based on the widely used follow your nose convention is important in enabling the fundamental use case of the Semantic Web.
- A URI definition discovery protocol does not have to work all the time to be useful to those who follow it.
- The effect of violating the URI definition discovery protocol is equivalent to the effect of publishing bad data, and the web is designed to be resilient to bad data.
- The marketplace will help sort out those who "play nicely" – by following standard protocols and publishing quality data – from those who don't.

Acknowledgments. Thanks to Paolo Bouquet for his comments on this paper. Thanks also to anonymous reviewers who provided very useful feedback, which I am still working to incorporate.

References

1. Jacobs, I., Walsh, N. (editors): Architecture of the World Wide Web, Volume One. (2004) W3C. http://www.w3.org/TR/webarch/#id-resources Retrieved 2012-06-08.

2. Hayes, P. (editor): RDF Semantics. W3C (2004). http://www.w3.org/TR/rdf-mt/#urisandlit Retrieved 2012-06-08.

3. W3C: httpRange-14: What is the range of the HTTP dereference function? W3C. http://www.w3.org/2001/tag/issues.html#httpRange-14 Retrieved 2012-06-08.

4. Berners-Lee, T., Fielding, R., Masinter, L.: Uniform Resource Identifier (URI): Generic Syntax. The Internet Society (2005). http://www.ietf.org/rfc/rfc3986.txt Retrieved 2012-06-08.

5. Duerst, M., Suignard, M.: Internationalized Resource Identifiers (IRIs). The Internet Society (2005). http://www.ietf.org/rfc/rfc3987.txt Retrieved 2012-06-08.

6. Berners-Lee, T., Hendler, J., Lassila, O.: The Semantic Web. Scientific American (2001). http://www.scientificamerican.com/article.cfm?id=the-semantic-web Retrieved 2012-06-08.

7. W3C: Resource Description Framework (RDF). http://www.w3.org/RDF/ Retrieved 2012-06-08.

8. Hayes, P. (editor): RDF Semantics. W3C (2004). http://www.w3.org/TR/rdf-mt/#defmerge Retrieved 2012-06-08.

9. Rees, J. (on wiki): HTTP URI Use Cases. http://www.w3.org/wiki/HTTPURIUseCases Retrieved 2012-06-08.

10. W3C: httpRange-14: What is the range of the HTTP dereference function? W3C. http://www.w3.org/2001/tag/issues.html#httpRange-14 Retrieved 2012-06-08.

11. Jacobs, I., Walsh, N. (editors): Architecture of the World Wide Web, Volume One. (2004) W3C. http://www.w3.org/TR/webarch/#URI-collision Retrieved 2012-06-08.

12. Jacobs, I., Walsh, N. (editors): Architecture of the World Wide Web, Volume One. (2004) W3C. http://www.w3.org/TR/webarch/#indirect-identification Retrieved 2012-06-08.

13. W3C OWL wiki: Punning. http://www.w3.org/2007/OWL/wiki/Punning Retrieved 2012-06-08.

14. Hayes, P. (editor): RDF Semantics. W3C (2004). http://www.w3.org/TR/rdf-mt/#sinterp Retrieved 2012-06-08.

15. Fielding, R., Gettys, J., et al: Hypertext Transfer Protocol -- HTTP/1.1. The Internet Society (1999). http://www.ietf.org/rfc/rfc2616.txt Retrieved 2012-06-08.

16. Jacobs, I., Walsh, N. (editors): Architecture of the World Wide Web, Volume One. (2004) W3C. http://www.w3.org/TR/webarch/#uri-ownership Retrieved 2012-06-08.

17. W3C Wiki: Follow Your Nose. http://www.w3.org/wiki/FollowYourNose Retrieved 2012-06-08.

18. Berners-Lee, T.: Linked Data. W3C (2006). http://www.w3.org/DesignIssues/LinkedData.html Retrieved 2012-06-08.

19. Connolly, D.: on media types for OWL (5.13). http://lists.w3.org/Archives/Public/www-webont-wg/2002Oct/0162.html Retrieved 2012-06-08.

20. Booth, D.: URI Declaration in Semantic Web Architecture. http://dbooth.org/2007/uri-decl/ Retrieved 2012-06-08.

21. Booth, D.: The URI Lifecycle in Semantic Web Architecture. http://dbooth.org/2009/lifecycle/ Retrieved 2012-06-08.

22. Sauermann, L., Cyganiak, R.: Cool URIs for the Semantic Web. W3C (2008). http://www.w3.org/TR/cooluris/ Retrieved 2012-06-08.

23. Fielding, R.: httpRange-14 Resolved. W3C (Email 2005-06-18) http://lists.w3.org/Archives/Public/www-tag/2005Jun/0039 Retrieved 2012-06-08.

24. Rees, J.: Call for proposals to amend the "httpRange-14 resolution". W3C. http://www.w3.org/2001/tag/doc/uddp/change-proposal-call.html Retrieved 2012-06-08.

25. Rees, J. (W3C wiki): TAG Issue 57 Responses. http://www.w3.org/wiki/TagIssue57Responses Retrieved 2012-06-08.

26. Russell, S.: rdf as a base for other languages. http://lists.w3.org/Archives/Public/www-rdf-logic/2001Jun/0075.html Retrieved 2012-06-08.

27. Stoermer, H.: Welcome to OKKAM – Enabling the Web of Entities. (OKKAM project home page.) http://www.okkam.org/ Retrieved 2012-09-29.

28. Reimer, M.: "Reference", The Stanford Encyclopedia of Philosophy (Spring 2010 Edition), Edward N. Zalta, editor. http://plato.stanford.edu/archives/spr2010/entries/reference/ Retrieved 2012-09-30.

29. Hayes, P., Halpin, H.: In Defense of Ambiguity. http://www.ibiblio.org/hhalpin/homepage/publications/indefenseofambiguity.html Retrieved 2012-09-30.