

Proposal for a new URI Scheme in MIRIAM

Camille LAIBE
camille.laibe@ebi.ac.uk
European Bioinformatics Institute (UK)

January 28, 2008

Contents

1	Introduction	2
2	Background	2
2.1	MIRIAM URIs	2
2.2	Overview of URIs used	2
3	Problems induced	2
3.1	Usage of URLs	3
3.2	Registered authority	4
3.3	MIRIAM notification	4
3.4	Consistency	4
4	Possible solutions	5
4.1	Create a new URI scheme	5
4.2	Create a new URN namespace	5
4.3	Use the HTTP scheme	6
4.4	Other possible solutions	6
5	Implementation	7
5.1	Obsolete framework	7
5.2	Domain name	7
5.3	Naming	7
5.4	Mirrors	7
5.5	Semantics	7
6	Semantic Web	8
6.1	Architecture	8
6.2	URI dereferencement	8
6.3	Usage of fragments	9
7	Discussion	9
8	Conclusion	10
9	Acknowledgements	10
10	Glossary	10

1 Introduction

With **MIRIAM Standard**, an identifier scheme was introduced to identify unambiguously components of computational models. This scheme is based on *Uniform Resource Identifier* (URI), a standard from the World Wide Web (W3C).

In this document we will describe the issues induced by the usage of these URIs.

Several alternative (and potentially better) solutions will be presented in detail, including the pro and cons of each of them.

This document aims at opening a discussion during the *Super-hackathon* about "standards and ontologies for Systems Biology" (Jan 28-Feb 2, Okinawa, Japan) in order to finally agree on a perennial URI Scheme for **MIRIAM**.

2 Background

The *Minimal Information Requested In the Annotation of biochemical Models* (**MIRIAM**)[1] is a set of guidelines for the annotation and curation processes of computational models, in order to facilitate their exchange and reuse. An important part of the standard consists in the controlled annotation of model components, based on *Uniform Resource Identifiers*[2].

In order to enable interoperability of this annotation, the community has to agree on a set of standard URIs, corresponding to recognised data types. **MIRIAM Resources**[3] are being developed to support the use of those URIs.

MIRIAM Resources as well as information about the standard are available at:
<http://www.ebi.ac.uk/miriam/>.

2.1 MIRIAM URIs

MIRIAM URIs follow the general scheme: {data type, identifier}. It is important to notice that the identifier makes only sense within the context of the data type.

To date, each data type in **MIRIAM Resources** is associated with (at least) one URI. As two syntaxes are supported: *Uniform Resource Name* (URN)[4] and *Uniform Resource Locator* (URL)[5], it is therefore possible to give one of each.

For example, it is possible to have these two URIs associated with one data type:
<http://www.dataType.ext/#entity> and <urn:lsid:dataType:entity>.

2.2 Overview of URIs used

Here are some examples of URIs stored in **MIRIAM Resources**. The great diversity of identifiers is obvious.

- <http://biomodels.net/MIRIAM/>
- <http://www.ebi.ac.uk/IntEnz/> (obsolete)
- <http://www.ec-code.org/>
- <urn:lsid:ec-code.org>
- <http://www.ebi.ac.uk/chebi/>
- <urn:lsid:uniprot.org:uniprot>
- <http://www.taxonomy.org/>

3 Problems induced

The usage of the current URIs generates several problems.

3.1 Usage of URLs

Currently the extreme majority of our URIs are actually URLs, and this imply several issues:

3.1.1 Confusion

There is a danger of confusion between the identifier of an *abstract* concept and the *physical* address of a Web page.

Here are some examples: for each data type there is the **MIRIAM URI** and the physical address used to access a piece of information using a specific resource:

1. *KEGG Compound*

- <http://www.genome.jp/kegg/compound/>
- [http://www.genome.jp/dbget-bin/www_bget?cpd:\\\$id](http://www.genome.jp/dbget-bin/www_bget?cpd:\$id)

2. *ChEBI*

- <http://www.ebi.ac.uk/chebi/>
- [http://www.ebi.ac.uk/chebi/searchFreeText.do?searchString=\\\$id](http://www.ebi.ac.uk/chebi/searchFreeText.do?searchString=\$id)

3.1.2 Ownership

Usage (and creation) of URLs that don't belong to us: for which we are not the owner of the domain name (example: **taxonomy.org**). There could be several cases: we didn't registered the domain name and it is not registered (yet) or the domain name has already been registered, but not by us.

According to the W3C, organisations or individuals which have the authority to create URIs can be thought of as the owners of those URIs. Therefore we are currently using something which doesn't belong to us.

Nevertheless, the situation is fuzzy: domain names are not trademarks. At least in most of the cases: that can be possible if the domain name is actually used to identify the registrant's goods or services to the public, rather than simply being the location of the Website (**Amazon.com** is an example of this situation)[6][7].

The fact is: we don't use these URLs as physical locations. Therefore, even if they are registered, that should not be prejudicial to anybody.

But even if the situation is not that bad, the main issue is that our current URIs are not protected at all! Collisions could occur in the future: if another organisation provides identification of different datasets with the same URIs.

3.1.3 Semantic Web concerns

Some people are confused by (or didn't understand the purpose of) **MIRIAM URIs**: why aren't they "valid"? Actually they mean that dereference is not possible.

We could argue that the scheme or protocol part ("http:") is a nonsense in our context, cf. <http://www.geneontology.org/>, where there is no direct dereference possible.

However, with the advent of *Semantic Web technologies*, the Web is extending so that **http:** URIs can identify not just Web documents but also objects and other abstract concepts.

For example, *Gene Ontology* uses that kind of URIs in their RDF format, cf. figure 1 (page 4).

Actually, the W3C recommends to be on and use the Web, an extremely robust and scalable information publishing system[8]. Whenever a URI is mentioned, be able to look it up to retrieve a description containing relevant information and links to related data would be interesting.

```

<go:term rdf:about="http://www.geneontology.org/go#GO:0000002" >
  <go:accession>GO:0000002</go:accession>
  <go:name>mitochondrial genome maintenance</go:name>
  <go:definition>The maintenance of the structure and integrity of the mitochondrial genome;
  <go:is_a rdf:resource="http://www.geneontology.org/go#GO:0007005" />
  <go:dbxref rdf:parseType="Resource">
    <go:database_symbol>InterPro</go:database_symbol>
    <go:reference>IPR009446</go:reference>
  </go:dbxref>
  <go:dbxref rdf:parseType="Resource">
    <go:database_symbol>Pfam</go:database_symbol>
    <go:reference>PF06420</go:reference>
  </go:dbxref>
</go:term>

```

Figure 1: Extract of the RDF-XML version of Gene Ontology

3.2 Registered authority

Most of the URNs used are actually LSIDs. They follow the pattern:

```
urn:lsid:<authority>:<namespace>:<objectId>[:<version>]
```

LSID specifications state that "it is recommended that [the domain name in the authority section] be owned by the organization that assigns an LSID in question". But we created (and use) `urn:lsid:ec-code.org` and we are not "ec-code.org" ourselves.

According to that, we should only create LSID-based URNs which start by something like: `urn:lsid:biomodels.net:dataType:entity` or `urn:lsid:miriam-uri.org:dataType:entity`.

3.3 MIRIAM notification

None of these URIs contains an element which "specify" their creator, the project they come from or their purpose. Nobody can link them with **MIRIAM**.

Therefore it is not possible for a "user" (in the general meaning), without any prior knowledge of **MIRIAM**, to understand what these URIs are used for, what they mean and where to get information about them.

3.4 Consistency

The syntax of the whole set of URIs is very diverse.

Some data types are identified by a neutral name, other linked to an organisation:

- `http://www.taxonomy.org/`
- `http://www.ebi.ac.uk/chebi/`

The best would be to have names for data types as neutral as possible (not directly linked to the name or domain name of any organisation).

Some URLs contain the "www." pattern, others not:

- `http://www.eccode.org/`
- `http://biomodels.net/MIRIAM/`

Inconsistencies exist as well in URNs:

- `urn:lsid:uniprot.org:uniprot`
- `urn:oai:arXiv.org`

A unified scheme would be interesting, although not mandatory.

4 Possible solutions

In this section, we will try to enumerate the available alternative methods and give their advantages and disadvantages (according to relevant specifications) of their usage in our case. Some other elements will be taken into account: simplicity, stability, manageability and perennality are the main ones.

4.1 Create a new URI scheme

A URI scheme is the top level of the URI naming structure: the first part before the colon character. Sometimes it is referred to as "protocol", since most URIs were originally designed to be used with a particular protocol. Today, it is not the case anymore, for example "http:" URIs are used for other purposes which are not related to the HTTP protocol, such as RDF resource identifiers and XML namespaces.

Each URI scheme has a specification that explains the scheme-specific details of how scheme identifiers are allocated and become associated with a resource. Some widely used URI schemes are: "http:", "mailto:", "ftp:", "urn", ...

Therefore, a solution could be to create our own URI scheme. For example, a URI could look like: `mir://dataType/entity` or `miriam:dataType:entity`.

The advantages of such a method are multiple:

- consistency of the URIs
- no confusion possible with a physical address
- reference to **MIRIAM** included in the URIs

The disadvantages are:

- this new scheme would need to be registered for perennality purposes
- these URIs would not be on the Web (in the sense of the W3C).

Introducing a new URI scheme is always costly, mainly because it is mandatory to register it. Without that procedure, anybody could use the same URIs but to identify different objects.

The URI ownership is delegated from the Internet Assigned Numbers Authority (IANA)[9]. This entity oversees the registration of new URI schemes and maintains a registry of mappings between URI scheme names and scheme specifications[10]. The IANA is itself operated by the Internet Corporation for Assigned Names and Numbers (ICANN).

The registration with the IANA includes a process of peer review. Usually that also requires the development and deployment not only of client software to handle the scheme, but also of elements such as gateways, proxies, and caches. The overall process can be quite lengthy[11][12].

4.2 Create a new URN namespace

The URN Syntax Scheme delegates ownership of portions of URN space to URN namespace specifications[13] which themselves are registered in an IANA-maintained registry of URN Namespace Identifiers.

For example "oasis", as used in `urn:oasis:names:tc:ubl:schema:xsd:Invoice-1.0`, is a URN namespace. It is obvious that this does not allow the possibility for the dereferencing the URI to retrieve the information, and there is not protocol associated with it.

URN namespace is much easier to register than a URI scheme[14]. The process takes about three weeks and involves a review by a technical list. It is often an opportunity to receive some feedback on the scheme which may enhance it.

Using this solution, we should be able to use URIs like: `urn:miriam:dataType:entity`.

4.3 Use the HTTP scheme

The "http:" scheme is a standard combining two elements: a universal naming scheme and a communication protocol.

Even if on the traditional Web, URIs were used primarily for Web documents (create links and access to them via a browser), it is now an obsolete idea. Several projects use URLs to identify elements which don't have a "representation" on the Web (cf. Gene Ontology).

Therefore, we could use something like: `http://www.miriam.org/dataType/entity` or `http://www.miriam-uri.org/id/dataType/entity`.

The advantages of using such a solution are huge[15]: base our work on an existing, robust and widely used scheme, consistency of the URIs generated, no need for a heavy registration procedure and the possibility to be "on the Web", if we register a domain name (but this last point is not mandatory).

Although many URI schemes are named after protocols, this does not imply that use of such a URI will necessarily result in access to the resource via the named protocol[16]. So, the dereference possibility is really only an option.

Moreover, the fact that if one scheme already exists, works well and covers our needs: why create a new one?

The disadvantage is the creation of confusion[17] about the element identified. Is it a name, a concept, a Web location or a document instance?

4.3.1 Going further... towards a Semantic Web

Even if our original needs only require an identifier scheme, the fact to be able to retrieve information could be interesting too. With this method, we could even imagine (and setup) a service where one can look up a description of the identified resource. This kind of mechanism could be important to easily and widely establish a good understanding of what **MIRIAM URIs** identify. For more details about this aspect, cf. section 5.5, page 7.

Of course there are still some perennality issues, but they are not about the perennality of the URIs as identifiers, but as physical locations. Fortunately it is not a real issue because a not valid URL (in the context of physical location) is still a valid URL (in the sense of URI).

4.4 Other possible solutions

These are not very attractive for our purposes, but worth mentioning.

4.4.1 PURL

A PURL is a Persistent Uniform Resource Locator[18]. A PURL is a URL. However, instead of pointing directly to the location of a resource, a PURL points to an intermediate resolution service which acts like a standard HTTP redirect.

Some organisations use PURL to identify their objects, such as the Dublin Core Metadata Initiative[19]. For example: `http://purl.org/dc/elements/1.1/identifier`.

4.4.2 eXtensible Resource Identifier

eXtensible Resource Identifier (XRI) is a scheme and resolution protocol for abstract identifiers compatible with Uniform Resource Identifiers and Internationalized Resource Identifiers. The goal of XRI is to provide a universal format for abstract, structured identifiers that are domain-, location-, application-, and transport-independent.

The W3C Technical Architecture Group considers that the same aims can be achieved with a better interoperability using "http:" scheme URIs.

4.4.3 TAG URIs

Another possible alternative is the TAG URI scheme[20].

This is a simple way to create persistent identifiers very quickly and cheaply. Everything needed is a domain name or an email address. By combining that with a date, a persistent base identifier is created under which one can generate URIs.

For example, the owner of the domain name "miriam-uri.org" on the 1st January 2008 could use: `tag:miriam-uri.org,2008-01-01:dataType:entity`.

The main advantage is the non registration requirement.

5 Implementation

5.1 Obsolete framework

The fact to setup a new URI scheme will generate a lot of deprecated URIs. In order to cope with that, a strong bridge between new and old URIs will be mandatory.

A deprecated system already exists and the Web Services provided have several methods for this purpose: checks if a URI is deprecated or not, retrieves the official URI from a deprecated one, ...

5.2 Domain name

In the case of a solution involving URLs available on the Web, a domain name will need to be used.

The choice to use the EBI one (`ebi.ac.uk`) is not an option: **MIRIAM** needs to have an international representation not tight to any institution.

The use of `biomodels.net` domain could be possible, but it would not be recommended: **MIRIAM** probably needs something even more neutral: it is used outside the BioModels.net project.

Therefore the registration of a new domain name is necessary. It would probably include a reference to **MIRIAM**.

A quick "whois" search on 19 TLDs (`.com`, `.org`, ...) gives the following results: all domains based on "miriam" are already registered, except one `miriam.re`. But some others, for example based on "miriam-uri" or "miriam-uris" are still available with the main generic Top-Level Domain (gTLD), such as `.org` or `.net`.

5.3 Naming

It is necessary to be cautious about the fact that DNS servers are *case-insensitive*, which means created URIs different from already existing ones regardless of the case.

5.4 Mirrors

MIRIAM Resources are the only resolution service for those URIs: a failure in this service would render the system unusable: that's a Single Point of Failure. What can be done to avoid that? By creating mirrors. Once this step achieve, we would have contributed to built a standards-based, non-fragmented, decentralized Semantic Web...

5.5 Semantics

Add more semantics in MIRIAM Resources. This can be achieved by different ways. For example, include links to different formats of the same document, like below:

```
[...]
<head>
  <link rel="alternate"
        type="application/rdf+xml"
        title="RDF Version"
        href="http://www.ebi.ac.uk/miriam/data/dataType" />
  [...]
</head>
[...]
```

6 Semantic Web

As explained previously, by using the "http:" scheme, we could bring more semantics to our infrastructure. This would be done by allowing the dereference of the URIs.

We could go further and provide different formats of content regarding the nature of the agent. Humans could get a readable representation, such as (X)HTML, and machines could get RDF data. All this using the standard Web transfer protocol HTTP[21].

With this resolving service, there is no need to have knowledge about **MIRIAM Resources** to understand and have access to the data identified by **MIRIAM URIs**.

Of course such a single point of failure (if no existing mirrors) is against the Web's decentralised nature, but would be highly useful. And, in case of a failure, the URLs would still stay valid URIs.

6.1 Architecture

The W3C Technical Architecture group gives a lot of good advices for achieving that. It states that in order to solve the architectural problem of the ambiguity of URLs[22], a specific dereferencing system must be used: the HTTP response must be a "200" code (OK, successful retrieval) in the case of the URL used as a physical location. In the case of a URL identifying an abstract document, a "See Other" response code (303) must be used.

The "303" code indicates that there is no representation available for the resource being accessed. However, it also indicates that a response may be found using a different URI and provide this other URI. It is important to notice that this other URI is not an alias of the original URI (like the one given by a "301" or "302" code). The URI provided is related to the original one, but is not a substitute for it[23].

6.2 URI dereferencement

Dereference a URI is the fact to use a URI as a physical address.

If URLs, with an availability on the Web, are chosen, here is an example of what information could be provided and how it could be provided (cf. figure 2, page 9).

6.2.1 Human interaction

A simple (X)HTML page containing the same information as the one on MIRIAM Resources about the same data type. Some links could be generated and presented to the user, which would be the actual physical locations where they will be able to have more information about the identified entity.

6.2.2 Machine interaction

The same pieces of information as the ones given to human could be provided in an machine understandable format, such as XML, or better: RDF.

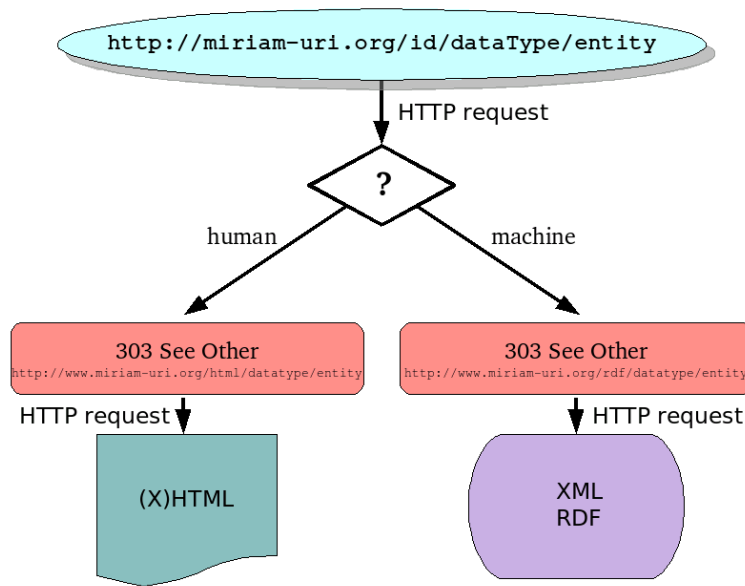


Figure 2: URI dereference system

6.3 Usage of fragments

The URL solution presented previously could be modified a little by using "fragments". URIs composed with a fragment are formed from a part concerning the primary resource with the addition of a fragment identifier after a hash character (#). An example could be: `http://www.miriam-uri.org/id/dataType#entity`.

In our case, the primary resource would be the data type and the secondary one, the entity (its identifier within the data type context). This would be quite appropriate because the "entity" has no meaning outside the context of the data type.

7 Discussion

Several questions now need to be answered:

1. Is the current situation that bad that we need to change the URIs?
2. Are you (users of MIRIAM URIs) ready for such a change?
3. Do you have any other idea/scheme which could be used in our case?
4. Which scheme do we choose for MIRIAM?

Concerning the first question, everybody should now be convinced that there is actually no other suitable solution if we want to provide a robust annotation infrastructure.

To help us address the last question, we can first think more about other questions:

1. Do we need to have URIs that are explicitly not Web addresses and therefore avoid any confusion?
2. Is it mandatory to have URIs that are valid Web addresses?,
3. Do we need to setup two schemes? one for URLs and another one for URNs?

I don't think we need to have URIs that are explicitly not Web locations. This annotation is designed to be done via a software, not by hand anyway.

Concerning the URIs that are valid locations: it is definitively not mandatory, but without decreasing the perennality of our identifier, that could bring some simplicity in their usage.

Once we have a good scheme, I don't think it is still necessary to carry two syntaxes: one identifier to rule them all!

8 Conclusion

First of all, we (authors of **MIRIAM Standard** and creators of **MIRIAM Resources**) would like to apologise to the community for releasing a great tool for annotation of computational models, but based on awkward URIs.

It is obvious that the current situation can't be kept any longer and that a real URI scheme must be chosen and used.

According to all the presented elements in this document, I would recommend the usage of "http:" URIs in **MIRIAM**, as well as a dereferencing system.

The "http:" scheme already exists and cover our requirements (it can even provide more). Reuse it is the easiest way to do as well as providing the best advantages.

Now, with such URIs (and the **MIRIAM Resources** architecture), we have all the elements to make other identifier schemes obsolete: LSID, PURL and LSRN to name but a few.

9 Acknowledgements

This proposal uses several ideas from the *W3C* (<http://www.w3c.org>) and its endless (by their number and size) specifications.

10 Glossary

Agent A person or a piece of software acting on the Web on behalf of a person, an entity, or a process.

Dereference a URI When an agent uses a URI to access a representation of the referenced resource: it is called dereferencing a URI.

Fragment identifier The part of a URI that allows identification of a secondary resource.

Physical location It is a string which can be put in the address bar of a Web browser, be retrieved by it and displayed to the user. This term is equivalent to "Web address" and "Web location".

Resource Anything that can be identified by a URI.

References

- [1] Nicolas Le Novère, Andrew Finney, Michael Hucka, Upinder S Bhalla, Fabien Campagne, Julio Collado-Vides, Edmun J Crampin, Matt Halstead, Edda Klipp, Pedro Mendes, Poul Nielsen, Herbert Sauro, Bruce E Shapiro, Jacky L Snoep, Hugh D Spence, and Barry L Wanner. Minimum information requested in the annotation of biochemical models (MIRIAM). *Nature Biotechnology*, 23(12):1509–1515, 2005.
- [2] T Berners-Lee, R Fielding, and L Masinter. Uniform Resource Identifier (URI): Generic syntax. <http://www.ietf.org/rfc/rfc3986.txt>, 2005. Internet Engineering Task Force (IETF).

- [3] Camille Laibe and Nicolas Le Novère. MIRIAM Resources: tools to generate and resolve robust cross-references in Systems Biology. *BMC Systems Biology*, 1(58), 2007.
- [4] R Moats. URN syntax. <http://www.ietf.org/rfc/rfc2141.txt>, 1997. Internet Engineering Task Force (IETF).
- [5] T Berners-Lee, L Masinter, and M McCahill. Uniform Resource Locator (URL). <http://www.ietf.org/rfc/rfc1738.txt>, 1994. Internet Engineering Task Force (IETF).
- [6] Trademark article on Wikipedia. <http://en.wikipedia.org/wiki/Trademark>. Wikipedia.
- [7] C Waelde. Domain names and Trade marks: What's in a name? http://www.law.ed.ac.uk/it&law/ch4_main.htm.
- [8] L Sauermann and R Cyganiak. Cool URIs for the Semantic Web. <http://www.w3.org/TR/cooluris>, 2007. World Wide Web Consortium (W3C).
- [9] B Carpenter, F Baker, and M Roberts. Memorandum of Understanding Concerning the Technical Work of the Internet Assigned Numbers Authority. <http://tools.ietf.org/rfc/rfc2860.txt>, 2000. Internet Engineering Task Force (IETF).
- [10] IANA's online registry of URI Schemes. <http://www.iana.org/assignments/uri-schemes>. Internet Assigned Numbers Authority (IANA).
- [11] L Masinter, H Alvestrand, D Zigmond, and R Petke. Guidelines for new URL Schemes. <http://www.ietf.org/rfc/rfc2718.txt>, 1999. Internet Engineering Task Force (IETF).
- [12] R Petke and I King. Registration Procedures for URL Scheme Names. <http://www.ietf.org/rfc/rfc2717.txt>, 1999. Internet Engineering Task Force (IETF).
- [13] L Daigle, D van Gulik, R Iannella, and P Faltstrom. URN Namespace Definition Mechanisms. <http://www.ietf.org/rfc/rfc2611.txt>, 1999. Internet Engineering Task Force (IETF).
- [14] K Sollins and L Masinter. Functional Requirements for Uniform Resource Names. <http://tools.ietf.org/rfc/rfc1737.txt>, 1994. Internet Engineering Task Force (IETF).
- [15] HS Thompson and D Orchard. URNs, Namespaces and Registries. <http://www.w3.org/2001/tag/doc/URNsAndRegistries-50>, 2006. World Wide Web Consortium (W3C).
- [16] I Jacobs. Architecture of the World Wide Web, Volume One. <http://www.w3.org/TR/webarch/>, 2004. World Wide Web Consortium (W3C) Recommendation.
- [17] D Booth. Four Uses of a URL: Name, Concept, Web Location and Document Instance. http://www.w3.org/2002/11/dbooth-names/dbooth-names_clean.htm, 2003. World Wide Web Consortium (W3C).
- [18] Persistent Uniform Resource Locator (PURL). <http://purl.org/>.
- [19] Dublin Core Metadata Initiative. <http://dublincore.org/>.
- [20] T Kindberg and S Hawke. The 'tag' URI Scheme. <http://www.faqs.org/rfcs/rfc4151.html>, 2005.
- [21] R Fielding, J Gettys, J Mogul, H Frystyk, L Masinter, P Leach, and T Berners-Lee. Hypertext Transfer Protocol – HTTP/1.1. <http://www.ietf.org/rfc/rfc2616.txt>, 1999. Internet Engineering Task Force (IETF).
- [22] T Berners-Lee. What HTTP URIs Identify? <http://www.w3.org/DesignIssues/HTTP-URI2.html>, 2005. World Wide Web Consortium (W3C).
- [23] R Lewis. Dereferencing HTTP URIs. <http://www.w3.org/2001/tag/doc/httpRange-14/HttpRange-14.html>, 2007. World Wide Web Consortium (W3C).