Web Proper Names: Naming Referents on the Web

Harry Halpin
ICCS, School of Informatics
University of Edinburgh
h.halpin@ed.ac.uk

Henry S. Thompson
HCRC, School of Informatics
University of Edinburgh
and
World Wide Web Consortium
ht@inf.ed.ac.uk

1 Introduction

1.1 The Web is about things

The value of the World Wide Web stems in large part from the fact that the varied constituents of the Web are *about* things—they describe things or picture things or discuss things. Often, although not always, these things are not themselves on the Web, rather they exist in the physical world. The ability to understand something as being about something, as being oriented towards something else without any direct connection to it, is crucial to human intelligence. Any effort to make the Web more intelligent, for example by automating the exploitation of resources on the Web, will have to somehow reproduce the human ability to understand what things are about.

This is an issue of immense practical importance: when someone searches the Web, they are looking for information *about* something. At present no automatic processes exist to index, organise, share, or even decide what web resources are about—all searches have to work with is text. The effort to provide machine-readable metadata through standards such as RDF and description logics as embodied in OWL are efforts to improve this situation. Although such efforts do allow a human to express what they believe a web-page is about in a standard way, they still beg the question of how to *interoperably* identify real-world things in such metadata.

Unfortunately, no-one from professional logicians to philosophers of consciousness have a solid idea about how we determine whether or not a thing is actually about something else. On the surface this *aboutness* seems physically spooky: I can think about the Eiffel Tower in Paris without being in Paris, or even having ever set foot in France. I can imagine what the Eiffel Tower would look like if it was painted blue. I can even think of a situation where the Eiffel Tower wasn't called the Eiffel Tower. Most importantly for our purposes, I can view a web page, either by typing a URL such as `http:://www.tour-eiffel.fr/` into a browser or by typing Eiffel into a search engine and following one of the links it provides. Having done this, I know at a glance if the page is actually about the Eiffel Tower, or a hotel near the Eiffel Tower, as opposed to the object-oriented programming language Eiffel, or the film The Lavender Hill Mob, and so on. Yet this knowledge depends on fundamental aspects of human intelligence such as language understanding, scene recognition and so forth, which have proved distressingly resistant to automation.

1.2 Names for things

As presently constituted, the effort to automatically exploit the content of the Web is a broad movement, ranging from information retrieval performed by term-based search engines to the Semantic Web and Topic Map standardisation efforts. Some of these approaches use URIs as the primary terms in the languages they use to express ***metadata***, that is, information intended for machine processing. Metadata is composed of logical ***sentences*** which in turn use URIs to stand for things, for example:

```
http://www.tour-eiffel.fr/
 http:.../architect
http://www.vitruvio.ch/arc/masters/eiffel.htm
```

or

```
http://www.greatbuildings.com/buildings/
      Eiffel_Tower.html
 http:.../architect
http://studentwebs.coloradocollege.edu/~a_macindoe/
      biography.htm
```

All the metadata sentences we use for examples in this paper have this form, that is, three URIs, to be understood as subject, relation, object.

These two metadata sentences in fact say the same thing—the first URIs of each triple stand for the Eiffel Tower, the third URIs of each stand for Gustave Eiffel, its architect. However there is no obvious way for an automatic process to detect this fact. In the absence of any agreed central authority which decides what URIs should be used to stand for what things, there is a real risk that the Semantic Web will consist of a vast number

of self-consistent but mutually incommensurable collections of metadata. Strictly speaking, the URIs in the above examples just address web pages (we're still speaking relatively informally—see 2.1 for careful definitions of the terminology we use thereafter). When a software agent fetches a web page from a URI, it's the web page addressed by the URI, as rendered by the agent on the basis of the encoding (such as HTML) returned by a server, that is actually about the Eiffel Tower or Gustave Eiffel. We'll come back to this distinction below.

The first challenge for the project of automating the exploitation of the Web is thus not to know what web pages are about—that's too hard for the time being. Just knowing when two pages describe the *same* thing would be a huge step forward. See (Guha, 2004) for an example of another effort to address this problem in the context of the Semantic Web.

We believe the Web needs a solution to this problem which

1. Provides a distributed approach to creating and sharing Web names for things;

2. Allows ***use*** of Web names *as* names to be easily distinguished from the ***use*** of URIs to address web pages;

3. Allows for efficient and reliable determination of whether two URIs address web pages which describe the same thing;

4. Does not require a single canonical name, while still achieving interoperability of names.

In this paper we propose a solution to this problem which exploits the pervasive availability of search engines with substantial coverage by using them to find sets of pages that human users judge to describe certain things. Just as phrases such as the Eiffel Tower are called proper names, we call our approach ***Web Proper Names***, and use `wpn:` in our examples as a candidate URI scheme for Web Proper Names. Although the concept can be refined further, a ***Web Proper Name (WPN)*** for something is usually composed of a set of search terms known to return primarily URIs for descriptions which describe that thing. It's at least initially plausible that such an approach to naming things for the Web should satisfy the requirements listed above—the rest of this paper is devoted to spelling out the details and demonstrating that in fact this is the case.

Note that we do *not* require that it be possible given a description to automatically determine the Web Proper Name of whatever it describes. This would set the bar too high—even names in the real world don't have this property.

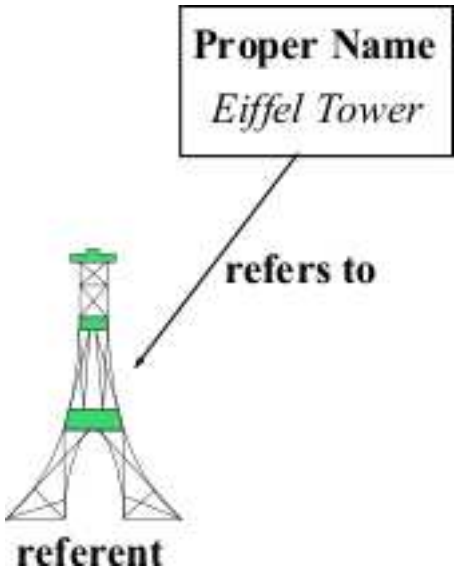

Figure 1: The reference relation

2 Philosophical Underpinnings

2.1 Terminology

Although terminology in this field often is confused, the underlying phenomena are reasonably clear: Generally, when something is understood to be about something else we talk about ***reference***, and the thing referred to is called the ***referent***. The reference relation is considered ***semantic*** or ***intentional***. One kind of reference is that which starts from ***names***—a class of linguistic expressions that are about something else. ***Proper names*** are names that refer uniquely to one referent, at least in an ideal situation. Figure 1 illustrates an example of this relationship.

On the Web this relationship becomes more complex. The draft recommendations of the W3C's Technical Architecture Group (2004) say that a URI ***identifies*** a ***resource***, and that browsers can ***retrieve representations*** which ***represent*** that resource.

We are uncomfortable with the status of the term resource, and will avoid its use in this paper, particularly because its precise meaning is evidently still under development by the Web community. Resource is currently defined by the TAG as "an item of interest in the information space known as the World Wide Web." (Jacobs, 2004) This idea of resource at first to be close to the concept of referent. Yet the URIs by which resources are identified do not seem to be connected to them in the way that names are to their referents (see §3 below). Henry S. Thompson refers to an actual person, while the URI of his web-page may refer to him as a rigid designator, or it may refer to itself, as explored in §2.2

Our take on the ordinary understanding of URIs is that

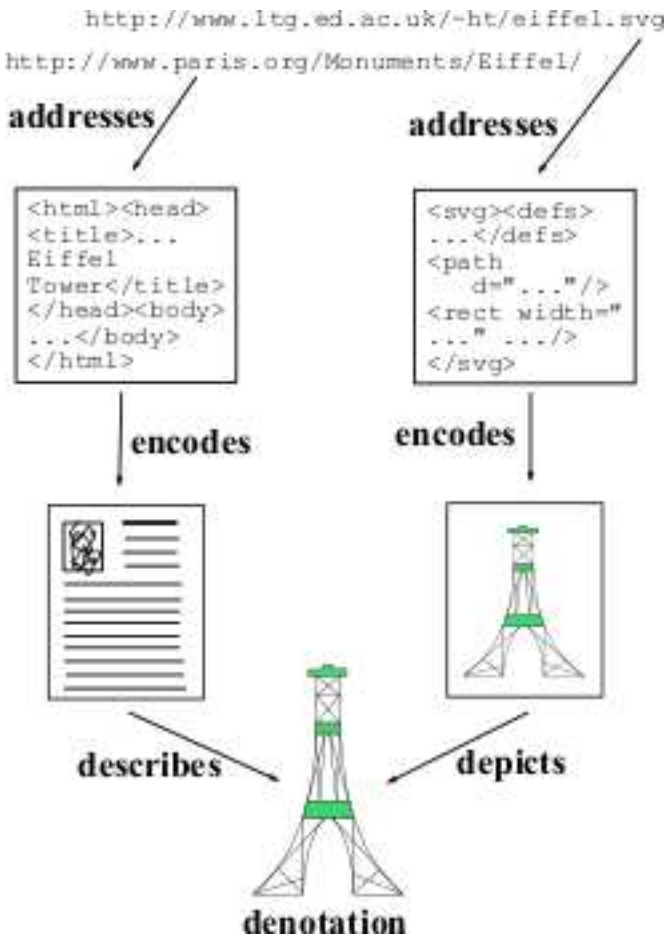

Figure 2: Description and denotation on the Web

a URI ***addresses*** a Web-based ***encoding*** of a ***description*** or ***depiction*** of a ***denotation***. An ***encoding*** is the character sequence that is actually retrieved, along with a specification of its media type, e.g. HTML or SVG. Informally it is the source for a web page, although the term is intended to be broad enough to cover non-web standards that encode their data more directly, such as JPG for images or MP3 for sound.

A linguistic ***description*** or pictoral ***depiction*** is the rendered output of a program given an encoding. Henceforth we will use expression as a cover term for the whole range of humanly-perceivable forms whose standardised encoding is addressed and retrievable by URIs—in other words ***expression*** is a cover term for HTML pages, SVG and JPG images, MP3 audio streams, as presented to humans by software. Also ***web pages*** will be used informally to cover both encodings and expressions in one term, and so will both cover the everyday language use of the term (as for HTML pages) but also refer to a wider set of phenomena (such as a URI addressing an audio stream). Finally, as in Goodman (1976), we use ***denotation*** for that which is depicted or described by an expression, where the philosophical treatment of reference would use ***referent***. This point is explored further in §2.3.

Subject to connectivity, the encoding addressed by a URI can be fetched, rendered as an expression by a software agent and seen or heard by a human, who can then determine what if anything the expression denotes. Figure 2 illustrates this. To summarize: the URI `http://www.tour-eiffel.fr` addresses an encoding in HTML, which can be retrieved by a web-browser, which renders the encoding as an expression composed of text and pictures, and these text and pictures will be recognised by a human being as denoting the actual Eiffel Tower in Paris

We can now be more precise about what's going on with respect to Web searches. When searching, a user typically wants to fetch ***expressions*** constituting ***descriptions*** (such as HTML or XML pages) or ***depictions*** (such as JPG or SVG images) that actually describe or depict some thing they are interested in. When searching the Web, many expressions can be found are not about the item of interest, and distinguishing those that denote the item of interest from those that do not is not straightforward. The human ability to do this, as remarked above, is evidently based on a wide range of linguistic and cognitive abilities, which machines have so far proved unable to reproduce.

In so far as determining the denotation of a web page may take us beyond the Web and into the rest of the real world, it is evidently beyond the reach of automation. What we *can* imagine being within reach is the slightly different problem of determining whether two expressions have the *same* denotation, in which case we say they are ***intentionally equivalent***. That determination is not in principle out of reach of automation if the denotations of the two expressions have been explicitly named, and it is the goal of the WPN effort described here to achieve this by providing a Web-appropriate naming mechanism.

2.2 The Use-Mention Distinction

A note of caution, in the guise of introducing one more bit of terminology. The connection from URI to expression and from expression to denotation encourages a confusion which is analogous to the ***use-mention*** confusion familiar to philosophers of language. Consider the difference between "Rice is tasty" and "rice is a one-syllable word." The first sentence ***uses*** the word rice to refer to a foodstuff in the world. The second sentence ***mentions*** the word rice in order to discuss a property *of the word itself*. There is a parallel problem on the Web when a URI occurs in metadata. In practice we observe that such metadata may *either* be understood as saying something about the expression whose encoding is addressed by the URI (a ***mention***), *or* as saying something about the denotation of said expression (a *use*).

For example, in order to understand the following as saying that Henry Thompson's W3C home page was created by Henry Thompson, we have to interpret the first URI as a ***mention*** but the second as a ***use***:

```
http://www.w3.org/People/thompson/
 http://purl.org/dc/elements/1.1/creator
http://www.ltg.ed.ac.uk/~ht/
```

For a range of reasons, which we will return to below, we think it's a mistake to ***use*** URIs in general in metadata—most URIs should be understood as being ***mentioned*** in metadata, that is, as referring to the expressions they address.

For ***use***, that is to refer to things in the world in metadata, we offer Web Proper Names as a particular kind of URI (that is, a URI using a particular URI scheme) intended for this purpose. This makes the understanding of URIs in metadata the opposite of the understanding of words in ordinary sentences. In ordinary sentences, words that are being ***used*** are unmarked, while words that are being ***mentioned*** are usually marked, either by quotes, italics, or underlining as in our previous example using rice. Whereas the unmarked case for words is ***use*** and the marked case is ***mention***, we are proposing that the unmarked case for URIs should be ***mention*** and the marked case (marked by the `wpn:` URI scheme) should be ***use***.

This problem is worth exemplifying in more detail. In our Eiffel Tower example (See §1.2 above) we alleged that the two metadata sentences said the exact same thing, because we had established by inspection that the relevant URIs shared the same denotation.

Consider now the example below, where on a purely syntactic basis, that being all that is available for automatic processing, the two sentences might appear to contradict one another, by asserting two different creators for the Eiffel Tower. But it is evident to a *human* who examines the four expressions addressed by the four URIs involved that the two are not contradictory: Gustave Eiffel is indeed the creator of the actual Eiffel Tower, while Gary Feuerstein created a web-page about the Eiffel Tower. That is, we must understand the first sentence as being about the ***denotation*** of the expression addressed by the URI `http://www.paris.org/Monuments/Eiffel` (a ***use***), while the second is about the ***expression*** addressed by that URI (a ***mention***).

```
 http://www.paris.org/Monuments/Eiffel
   http://purl.org/dc/elements/1.1/creator
 http://www.gustaveeiffel.com/

http://www.paris.org/Monuments/Eiffel
   http://purl.org/dc/elements/1.1/creator
http://www.endex.com/gf/
```

While the term `creator` may have a human-readable definition that can be found via the URI of `creator`, and so could specify only one of the above as a correct usage of `creator`, the natural language definition as written by a human could be ambiguous and a machine would not be able to understand that definition, especially if proper sub-categorisation is not provided. Regardless, an explicit distinction between denotation and expression would help make sense of such statements.

2.3 Search engines and descriptions

Although the philosophical story and the Web story (see Figure 1 and Figure 2 above respectively) appear to be different, in that in the one case reference is unmediated, but in the other mediated by an expression, in fact the parallel is much stronger.

The classic approach of Frege posits *three* elements to any reference: the ***name***, the ***sense***, and the ***referent*** (Frege, 1892). The actual thing in the world is still the referent, and a name is a symbol that has a referent. The ***sense*** is the mode of presentation, a type of public, objective knowledge about the item. Frege himself would likely judge this to be Platonic in nature. Russell and others analysed proper names as "abbreviated" descriptions. Their ***descriptivist*** theory of names analyses a name as identifying a set definite descriptive terms (Russell, 1905). These descriptive terms could be logical or linguistic in form. On the descriptivist account a name maps in the head of its user to a private concept of what the referent is. ***Sense*** is the public projection of that private concept among a shared community. The third party of ***sense*** mediates the reference relationship.

In Frege's classical example, Hesperus has a sense ("the morning star") different from that of Phosphorus ("the evening star"), yet both have the same referent, the planet Venus.

The descriptivist notion of sense is evidently parallel to the place of search terms in the Web story. An expression addressed by a URI can thereby be fetched and shared among the community of Web users. The notion of a sense as composed of definite descriptive terms also has an intriguing connection to the contemporary use of search engines. Typing descriptive terms such as Eiffel, Tower and Paris into a search engine returns URIs that address descriptions of the actual Eiffel Tower. In the context of the Web, there is clearly a non-arbitrary, although not strictly necessary, relationship between the descriptive terms and whatever the recovered web pages denote. Insofar as we've hinted that a Web Proper Name is a collection of search terms, this analogy is encouraging, particularly because the first step, from search terms to URIs, is automated and distributed.

It is important to note, however, that there are problems treating ***sense*** as a set of descriptive terms. It is in practice very difficult to come up with a set of descriptions that identifies exactly one referent. The Eiffel Tower is "a large metal monument." To distinguish it from the multitude of other large metal monuments in the world, the Eiffel Tower is "a large metal monument in Paris." There are other large metal monuments in Paris, and the Eiffel Tower would still be the Eiffel Tower if it were moved to Lake Havasu City. Searle addresses this issue in his ***cluster theory*** of names (1958), in which he suggests that only some or most of the

terms intended to identify a referent need do so. Furthermore, many of the descriptive terms, or indeed all of them, may also describe things which are *not* the intended referent.

Analogously, when using Google, typing in search terms for the Eiffel Tower such as `Eiffel Tower Paris` results in *some* web pages about the actual Eiffel Tower in Paris, but not all of them, and also web pages of things only marginally connected to the Eiffel Tower, such as hotels with views of the Eiffel Tower, or worse, something as inappropriate as an Eiffel programming language conference in Paris. The size of the retrieved set will also be *quite* large ("about 379,000" according to Google on the day of writing).

This suggests a refinement not usually found in philosophical accounts: the use of negative search terms. For example, the fact that the Eiffel Tower is not a hotel can be reflected by using `Eiffel Tower Paris -hotel` as the set of search terms. This has a dramatic effect—at the time of writing the size of the set Google returns for these terms is "about 166,000".

The analogy we are developing looks like this—a Web Proper Name should function like a natural language ***name***, identifying a referent. It consists of a set of search terms, including negative ones. Courtesy of a search engine, it determines a set of URIs that address web pages. At least a subset of those in turn denote the ***referent*** of interest.

When someone uses a search engine, if the majority of the descriptions retrieved for a given set of search terms, particular the high-ranking ones, do in fact describe the desired referent, then the search is generally considered successful. Analogously, a set of search terms is a good candidate for a Web Proper Name if the majority of the URIs retrieved for those terms, particular the high-ranking ones, do in fact address web pages with the same denotation, the intended referent of the Web Proper Name.

3 Names, descriptions and fixing the referent

It's important not to confuse a ***name*** with ***descriptions*** of its referent. In the real world, we use the *name* Eiffel Tower to uniquely determine the Eiffel Tower referent. We use names, not descriptions, to identify people. For example, the ***name*** Tim Berners-Lee identifies a certain man in Boston who is the Director of the W3C and wrote the book called *Weaving the Web* about his part in the creation of the World Wide Web. Moreover, when we want to refer to Tim Berners-Lee, we don't have to redescribe him using his title or the book he's written. A name *alone* determines its referent, at least where all parties involved attach the name to the same referent. Furthermore, this is achieved without appeal to descriptions.

In *Naming and Necessity*, Kripke says that names function to *fix a referent* without being a shorthand for sets of descriptive terms (1972). This is in tension with both the descriptivist and cluster theories of names discussed above. Descriptions aren't entirely out of the picture on Kripke's account—they are necessary for disambiguation when the context of use allows more than one interpretation of a name, and they may figure in the process by which things actually *get* their names. In Kripke's account an agent or agents fix a name to a referent by a process called ***baptism***, in which a thing and a word are directly associated. Afterwards one can use a name by virtue of being in a causal chain with the baptism. If the agent, the thing being named, and the listener are all co-present, the thing being named can be directly identified, otherwise careful use of descriptive terms will be required in order to adequately identify the object.

Sometimes proper names include ordinary words which themselves contribute to our understanding, for example Prime Minister, Crystal Palace, Big Island. The use of search terms in Web Proper Names parallels this to some extent. Using these terms a search engine can select from the vast number of web pages available on the Web a set which may describe the referent one is interested in. Note that other forms of information such as Semantic Web metadata, or the use of more sophisticated heuristics from information retrieval, may contribute to the selection of this set.

The lessons here for naming on the Web are that names and search terms are not the same, but that search terms can be used to create names for the Web, via web pages, in a productive and interoperable way. Baptism on the Web can be achieved by an appeal to a set of search terms which recover appropriate expressions, which in turn denote the intended referent. The baptizing agent of a Web Proper Name is the owner of the Web Proper Name. The referent is whatever thing the owner is interested in. A Web Proper Name is composed of search terms that given to a search engine will recover a set of URIs which address expressions which can in turn be verified by the baptizing agent as denoting the referent. We can now effectively merge our two earlier pictures, as shown in Figure 3.

It would be difficult if not impossible to select a set of search terms that uniquely determine a referent, that is, terms which recover a set of URIs such that *all* the web pages addressed thereby denote the intended referent. That's why the role of the baptising agent is crucial: It's their job to determine whether the denotation of each web page is *really* the intended referent.

Bar the creation of genuine artificial intelligence, currently only human inspection can check whether or not a given web page denotes a particular referent. A human agent with a referent to baptise must refine a set of search terms until an appropriate subset of the expressions addressed by the URIs recovered by a search engine from those terms denote that referent. They can then promulgate a Web Proper Name. For this story to be truly successful, it is crucial that the baptising agent need not continue to be involved in the process beyond the initial creation of a Web Proper Name—how this can be achieved is discussed below. Details of just what appropriate means above, what a Web Proper Name looks like in detail and how it removes the baptizing agent from ongoing uses, will be given below.

This process is actually what many users of the Web do everyday—using a search engine to find web pages about something, getting a list of URIs back and manually checking the descriptions they address to see if they are really about the referent, then changing the search terms if required to improve the result. Web Proper Names are a formalization of this everyday phenomena that allows the results to be packaged via a URI scheme (as detailed in §4) or a file (as detailed

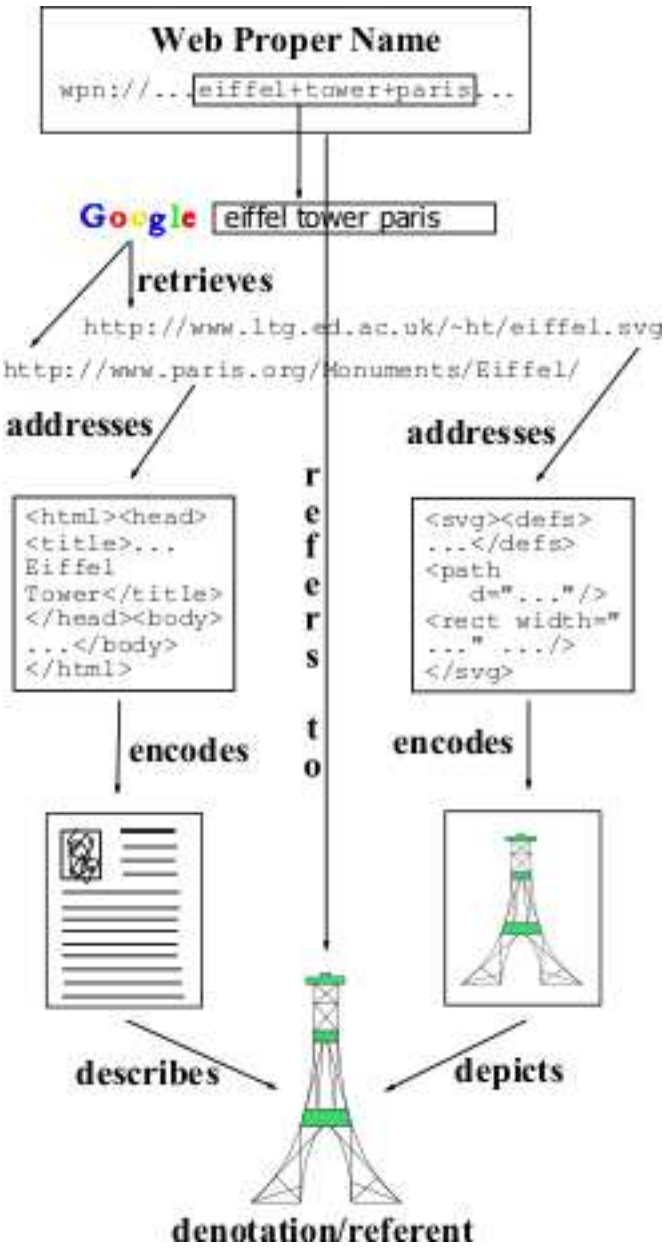

Figure 3: Web Proper Names

in §5), and so shared and used as the foundation for further information gathering on the Web.

4 Specification of Web Proper Names

A ***Web Proper Name*** is a Web-usable name for a referent, based on a set of search terms which recover a set of web pages that denote that referent. A Web Proper Name not only determines many web pages, but a single web page may participate in many Web Proper Names. A Web Proper Name should not be confused either with the set of search terms, the referent itself, the set of descriptions, or the additional information needed to situate the context of its baptism. A Web Proper Name is unique by virtue of the conjunction of all these properties. WPNs are not limited to naming things that already have ordinary proper names, such as the Eiffel Tower or Tim Berners-Lee, but can be constructed to name virtually anything, such as my eldest sister-in-law and lambda calculus, as well as fictional referents such as unicorns.

We define a Web Proper Name formally as a nine-tuple, as follows, with abbreviations for the components in parentheses:

Owner:	The baptising authority
Short name:	A short mnemonic for the WPN
Terms:	The positive and negative search terms used
Engine (se):	The search engine used
Date (dt):	The date the search was done
Language (ln):	The language of the terms
Result Sequence Size (rs):	Number of URIs returned by the search engine as a binary order of magnitude
Checked Sequence Size (cs):	Number of URIs in the result sequence that have been checked.
Percent Correct (pc):	The percent of correct URIs (actually about the referent) of those checked

For use in metadata, a Web Proper Name must be recognisable as such. Accordingly we package the constituents defined above into a URI using the hypothetical `wpn:` scheme as follows:

```
wpn://owner/shortName?terms=terms&
  se=engine&dt=date&rs=resultSequenceSize&
  cs=checkedSequenceSize&pc=percentCorrect
```

Note that since the size of some of the sequences, particularly the Result Sequence, might be quite large, the size of the sequence is expressed as a binary order of magnitude in integer form. The main advantage of the binary order of magnitude encoding is that it allows a fine-grained grasp of the size of small sequence sizes while a coarse-grained grasp of the size of large sequences. Since it allows very large sequence sizes to be estimated by small integers, it is an excellent choice for human readability. The binary order of magnitude is defined by $x = 2^y$, where x is the number of web-pages and y is the binary order of magnitude. When finding the integer binary order of magnitude, the closest integer to the actual number of the binary magnitude should be used.

For example a Web Proper Name with the following composition:

Owner:	www.ltg.ed.ac.uk/~ht/WPN
Short name:	EiffelTower
Engine:	www.google.com
Date:	2004-04-29
Terms:	eiffel+tower+paris+ -hotel+-webcam
Language:	en
Result Sequence Size:	17
Checked Sequence Size:	5
Percent Correct:	84

is expressed in a `wpn:` URI like this:

```
wpn://www.ltg.ed.ac.uk/~ht/WPN/EiffelTower?
 terms=eiffel+tower+paris+-hotel+-webcam&
 ln=en&se=www.google.com&dt=2004-05-21&
 rs=17&cs=5&pc=84
```

In more detail, the constituents of a Web Proper Name are as follows:

Owner: Identification of the baptising authority, in a form usable as an `http:` URI.

Short name: A short mnemonic for the WPN, in a form allowing it to be combined with the Owner to give a valid `http:` URI.

Engine: The domain name of the search engine used.

Date: The date the search was done, in YYYY-MM-DD form.

Terms: The positive and negative search terms used, combined with plus signs, phrases surrounded by double-quotes, spaces in phrases escaped as %20, negative terms marked with minus signs.

Language: The natural language of the terms—for inclusion in the query if the search engine supports language-filtering.

Result Sequence Size: The binary order of magnitude of the cardinality of the sequence of URIs retrieved.

Checked Sequence Size: The binary order of magnitude of the cardinality of the subsequence of the Result Sequence that have been checked to determine whether they describe the referent.

Percent Correct: The percentage of the Checked Sequence found to actually describe the referent.

Note that the Checked Sequence is always a subsequence or sequence of subsequences of the Result Sequence, preserving the search-engine-determined ordering of the Result Sequence. The Checked Sequence can only be constructed by human inspection, by fetching the description addressed by each URI in turn and inspecting it.

Is a new URI scheme really required for Web Proper Names? As stated in the (Jacobs, 2004), "When a software agent dereferences [a non-http:] URI, if what really happens is that HTTP GET is invoked to retrieve a representation of the resource, then an "http" URI would have sufficed." The primary intended use of Web Proper Names is to identify the referents in metadata sentences, while the primary use of `http:` URIs is to address a web page or group of web pages. In practice (see §5 below), a WPN may be dereferenced to retrieve a usable web page with additional details concerning the baptism of the WPN, but it is necessary for its primary role as a *name* that it be intrinsically (i.e. notationally) distinguishable for normal `http:` URIs, so in fact it *cannot* use the `http:` URI scheme.

4.1 Requirements and design choices

Each of the constituents of a Web Proper Name is intended to help achieve one or more of the goals we set out initially. The key to interoperability is a form of reproducibility: the Engine, Terms and Language enable anyone to repeat the original query and examine the Result Sequence. This reproducibility is not perfect due to the dynamic nature of the Web, and will change over time. To help deal with this, the time elapsed since the Date, and the Result Sequence Size, allow a user to judge how far things may have changed since the original query. However, one distinct advantage of WPNs is that they can be easily updated by running the search again and inspecting the changes in the results. The identity of the Owner provides a concrete basis for judging the reliability of the Web Proper Name as a whole, and Percent Correct gives an estimate of the precision of the search terms with respect to the intended referent.

How do we stand then with respect to the four goals stated in §1.2?

Provides a distributed approach to creating and sharing Web names for things

Anyone can create a Web Proper Name, and the components described above can be either published using the `wpn://` scheme or in an expanded form described below in §5. The fact that anyone can create a Web Proper Name does not distinguish it from URIs in general. What makes Web Proper Names as defined here independently creatable and sharable for the purpose of naming things on the Web in a way that arbitrary URIs are not is that it is easy for independently created Web Proper Names to be compared. This is discussed further in §4.1 below.

Allows *use* of Web names to be easily distinguished from *mention* of URIs

Web Proper Names evidently satisfy this by definition—the use of the `wpn:` URI scheme ensures this, and this use is the primary justification for the creation of the `wpn:` scheme. By definition a WPN always denotes its referent, not any Web-situated representation of that referent. In current usage, a URI may or may not be intended to denote a referent, and this intention is not typically determinable by non-human agents.

Allows for efficient and reliable determination of whether two URIs identify resources which are about the same thing

The design given here for Web Proper Names satisfies this goal, at least for URIs known to be intended as names, that is, `wpn:` URIs, at three levels:

1. by including the ***Short Name*** constituent, which can be used to signal the baptizer's intent;
2. by including the ***Terms*** constituent, which specifies the baptizer's intent much more explicitly;
3. by allowing for much more detailed information about the ***Checked Sequence***, including a partition of its member URIs into correct and incorrect, to be fetched using a URI (see §5 below). Significant overlap between the membership of the correct Checked results of two WPNs gives a strong presumption of identity of intended referent.

Does not require a single canonical name, while still achieving interoperability of names

The implicit contrast here is with an approach to naming on the Web that requires or assumes some form of centralisation, either of names themselves, or of assertions of equivalence of names.Web Proper Names are interoperable without such centralisation, because two Web Proper Names can be compared on the basis of their constituents, in particular the ***Terms*** constituent. Identical positive terms give a strong presumption of significant relationship between two Web Proper Names, identical positive *and* negative terms strongly suggest identity of intended referent. For many purposes we expect this level of comparison to be adequate. For greater precision and reliability, comparison of Expanded Web Proper Names, as discussed in §5 may be required, but this is achieved by appeal to the ***Owner***, not to a universal central authority.

4.2 The relative strength of Web Proper Names

The higher the ***Percent Correct***, the stronger the Web Proper Name. A ***Percent Correct*** of 100 means that *all* the URIs retrieved by its search terms (contextualized by the other parameters), are about the referent; ***weak*** says if it's *not* in the set retrieved by the terms, there are some URIs about the referent. Getting directly to ***strong*** is hard in most cases, and requires either explicit negatives or incidental positives, which both risk throwing out true positives. ***Weak*** is much easier to achieve, although users should aim to create the strongest WPNs possible given the constraints on their time, the search engine, and so on. Strong WPNs are easily identified as their percent correct is 100%. Currently, in general the Web is too large for any search engine to have perfect retrieval of every web page that fulfills the search terms and other parameters, and also the authors of a WPN have a finite amount of time. Given these two facts and the growth of the Web, WPNs are always incomplete. However, incompleteness does not imply that WPNs have no or limited uses. §6 discusses examples of their use that address important aspects of Web architecture.

4.3 Creating WPNs without a Search Engine

WPNs do not *require* search engines, and so the search engine and descriptions parameters are optional. URIs may be gathered from many places; they can be e-mailed directly, seen on the sides of cars, written in ads in magazines, found by casually poking around some web-pages. As long as a group of URIs are about the same referent, they can be added to a WPN. This also allows one to make WPNs whose size is only one. For example, Pat Hayes may e-mail me a web-page he has made as a rigid designator for himself, `http://www.ihmc.us/users/phayes/PatHayes.html`. This can be made into a WPN of size one for Pat Hayes. Creating WPNs in this manner has some but not all of the advantages of search-term-based WPNs, as at least interoperability and the use of search terms to find more expressions about the referent in question lost. An encoding of this information into a `wpn://` scheme URI is possible, but perhaps misleading—it is recommended that these manually created WPNs be packaged as Expanded WPNs as detailed in §5.

In §2, it was assumed that a URI addressed some encoding, yet it is possible that a URI does not actually address anything, such as in the use of URIs to identify namespaces. It may be useful to have a URI that is about a referent without addressing any explicit encoding, such as if `http://www.ibiblio.org/hhalpin/myParisnamespace` is only supposed to be used in relation to the actual Eiffel Tower. In the case of URIs which don't address an encoding, such as namespace names, the expression of the URI is actually in the text of the URI itself. That is, in this example the expression is not missing, rather the text of the URI itself serves as a description of the referent. Since search engines do not at this time recover such non-referring URIs, they should be a problem for the standard creation of WPNs. Yet, these non-referring URIS can be included in a manually built WPN, although the inspecting of such a URI is simply a human reading the URI itself.

4.4 Context Dependence and WPNs

With regards to exactly what a referent is, WPNs and this proposal are agnostic. A WPN should not be confused with its denotation. It is simply a distinguisable URI type for use when reference is required. A referent can be anything that can be referred to by an agent, not necessarily something as concrete and particular as the Eiffel Tower. Web Proper Names do not restrict referents to only those things that have proper names, and Web Proper Names makes no claims regarding a theory of natural kinds. Referents should be allowed to have various levels of abstraction, just as The University of Edinburgh is very concrete, but the class of things known as universities is less concrete. Some referents exist in the world only as ideas, and many can be fictional. Some of these, such as the idea of Web Standards, may have their physical existence primarily on the Web itself. It is not the task of WPN to define what a referent is, it is the task of the human who is creating the WPN and adding URIs to the WPN. WPNs do not claim to create a universal and centralised ontology as Cyc does (Lenat and Feigenbaum, 1987), but rather aims to enable a distributed and cooperating ontology fragments. The class of referents is as diverse as the possible interests of humans and world itself (Smith, 1991).

While the WPN specification makes no claim about what a referent is in general, it does imply that the owner decides this in each particular case. The context-dependent nature of WPNs is stored in the required parameters—the date of creation and owner URI—although the general principle of incorporating context permeates the whole design of Web Proper Names. The very judgement about whether or not a particular expression is about a referent is a matter of perspective on the part of the owner of the WPN. If one was searching for information about the Eiffel Tower, would pictures of the Eiffel Tower count, or the mention of the Eiffel Tower in lists? What about expressions in multiple languages? All this depends on the decisions of the owner, and owners will make different decisions. Some owners may not want to include photos of the Eiffel Tower, others will draw the line at *discussions* of pictures of the Eiffel Tower, but there is no reason anyone should have to pre-emptively restrict themselves to any particular type of expression. Yet, we expect that there will be some measure of overlap for popular and concrete referents. There are also practical advantages

to context dependence. Grounding a WPN in context such as the `owner` property allows circles of trust to be implemented with WPNs. Lastly, in their Expanded form (as detailed in §5) a WPN also allows its URIs to be ranked with a **relevance** parameter that allows a owner to rank how relevant they think a constituent URI of a WPN is to its denotation.

5 Expanded Web Proper Names

While Web Proper Names are not *universal* in the sense that a WPN uniquely identifies its referent over the Web for everyone, its format should be *uniform*, so WPNs may be exchanged and processed in a uniform manner by everyone. While WPNs can be given the form of the `wpn://` URI scheme, the same information can be structured and expanded on in a web page. The information contained in this web page can then be transformed into other formats, such as XML documents, RDF metadata, and OWL classes. Allowing the information to be packaged in a way that is not restricted to being easily displayed within a URI allows far greater detail to be provided for the WPN. We use the name ***Expanded Web Proper Name (EWPN)*** for this packaging and expansion of WPN information.

Expanded Web Proper Names expand upon WPNs both on a conceptual and a practical level. Using the term representation from the TAG, a WPN does not allow access to a representation since it is supposed to be about a referent, not a representation of a referent (Jacobs, 2004). However, if a web page that stores the information of a WPN is not allowed, then WPNs become difficult to exploit and integrate with a range of applications. Clearly the information in a WPN needs to be stored as a web page for some uses, and Expanded Web Proper Names exist to fill this role. In fact, an EWPN is independently useful as a *web page that is guaranteed by its owner to be about a certain referent.*

The EWPN allows more information to be stored than in the original WPN. The WPN as defined above is meant to be human-readable and concise. A number of design decisions were made to further these goals, such as using binary order of magnitude to represent URI counts. Due to this, the original concise ten-tuple used by the `wpn://` format leaves out information. In particular, it leaves out:

1. The URIs of the members of the Checked Sequence.
2. Whether the URIs that have been checked by the owner are about or not about the intended referent of the WPN.
3. Further optional data about the referent that could be useful. This could include information about further tools that were used in refining the search results, further options used to control the search engine, further information about the author of the WPN or the referent of the WPN.

For many purposes, such as re-checking a WPN or comparing WPNs, the exact URIs recovered from its search terms are crucial. If two EWPNs have a majority of recovered URIs in common, then there is a strong presumption that they are about the same thing or closely related things. However, this can not be determined unless the actual URIs are available. For fine-grained comparison of WPNs or statistics about WPNs, the exact size of the retrieved sequence is needed.

The original specification of WPNs is accordingly modified with the additional information detailed above to make the Expanded Web Proper Name specification. Also, many of the previous optional features are now required. Three new items of information have been added at the end of the specification, and instead of representing cardinalities of URI sequences as binary magnitudes the information is now exact:

Owner: Identification of the baptising authority, in a form usable as an `http:` URI.

Short name: A short mnemonic for the WPN, in a form allowing it to be combined with the Owner to give a valid `http:` URI.

Engine: The domain name of the search engine used. (Optional)

Date: The date the search was done, in YYYY-MM-DD form.

Terms: The positive and negative search terms used, combined with plus signs, phrases surrounded by double-quotes, spaces in phrases escaped as %20, negative terms marked with minus signs. (Optional)

Language: The natural language of the terms—for inclusion in the query if the search engine supports language-filtering. (Optional)

Result Sequence Size: The exact cardinality of the sequence of URIs retrieved.

Checked Sequence Size: The total number of URIs in the Result Sequence that have been checked to be about the referent, even if they were not about the referent.

Correct Checked Sequence Size: Number of URIs in the Checked Sequence that has been checked and verified by an agent, such as the owner, to actually be about the referent. This means that they have been verified by some investigation of the expression addressed by the URI (or in the case of non-referring URIs, the URI itself).

Percent Correct: The Correct Checked Sequence Size divided by the Checked Sequence Size.

Correct Checked Sequence: A list of URIs in the Result Sequence that have been checked and *are* about the referent. The number of URIs in this list will be equal to the Correct Checked Sequence Size.

Incorrect Checked Sequence: A list of URIs in the Result Sequence that have been checked and are *not* about the referent. The number of in this list will be equal to the Checked Sequence Size minus the Correct Checked Sequence Size.

Further Information: Any further potentially useful information. (Optional)

The entries in the two lists of URIs may also include optional **relevance** ratings to rate a URI on an ordinal scale as to how relevant to the WPN they are, as well as an optional **comment** for any additional potentially relevant information on the URI.

The **Further Information** parameter of an Expanded WPN is for additional metadata about the WPN itself over and above the minimum data normally included in an EWPN. Metadata could give version history, such as how often the WPN is updated. More metadata would be crucial if one were merging WPNs, such as one would want to do when building multilingual WPNs. One could include in **Further Information** information about the set of tools used to automatically prune the result sequence, such as Semantic Web technologies or information retrieval heuristics.

Search engines return the URIs in a sequence, and so it is recommended that the order of the URI lists be the same order that the search engine returned. Also, we imagine that many of these numbers, such as the sequence sizes and date parameters, can be easily derived and filled in automatically by WPN authoring tools.

An Expanded WPN may be stored anywhere. We encourage people to store them so they are addressed by an `http` URI formed by adding the shortname to their owner identification. For example if the owner's identification was `http://www.inf.ed.ac.uk/~ht/` and the shortname *EiffelTower*, the encoding for the EWPN should be addressed by `http://www.inf.ed.ac.uk/~ht/EiffelTower`.

The information content of an Expanded WPN should be encoded as an XHTML RDDL file (Borden and Bray, 2002). However, as long as this information is encoded in some form, the encoding is an EWPN. A few of the more obvious non-canonical encodings (XML, RDF, OWL) are explored as examples of WPN use in §6. Schemas for the RDDL encoding are available at `http://www.webpropernames.org/`, along with schemas for the non-canonical encodings.

6 Uses of Web Proper Names

There are many applications crying out for Web Proper Names, not surprisingly given the central status of reference throughout the Web. This section introduces a few such uses, presenting in order of increasing complexity.

6.1 Distinguishing Use from Mention

If taken to refer to an encoding, then all `http://` URIs are cases of ***mention***, which makes it difficult to have any metadata about a URI that is about not the referent but the representation in an unambiguous manner. Using the wpn:// URI scheme can solve this problem, so that the following triple asserts that the person Henry S. Thompson created his W3C web page:

```
http://www.w3.org/People/thompson/
 :creator
wpn://www.ltg.ed.ac.uk/~ht/WPN/HenrySThompson?
terms=Henry\%20S.\%20Thompson\&
ln=en\&se=www.google.com\&dt=2004-05-21\&
rc=17\&tc=5\&cc=5\&pc=87
```

6.2 As Authoritative Web Pages

Because a canonical Expanded WPN is an RDDL document, it is also XHTML and can usefully be displayed in a web browser for human perusal. Other human-readable information relevant to the referent could be added. An example WPN screenshot for the Eiffel Tower is included in 4. For example, an EWPN Henry S. Thompson created about Pat Hayes might include his birth date, the names of his parents, his Social Security Number, and a link to his picture. This information could be found via the Further Information part of the Expanded WPN.

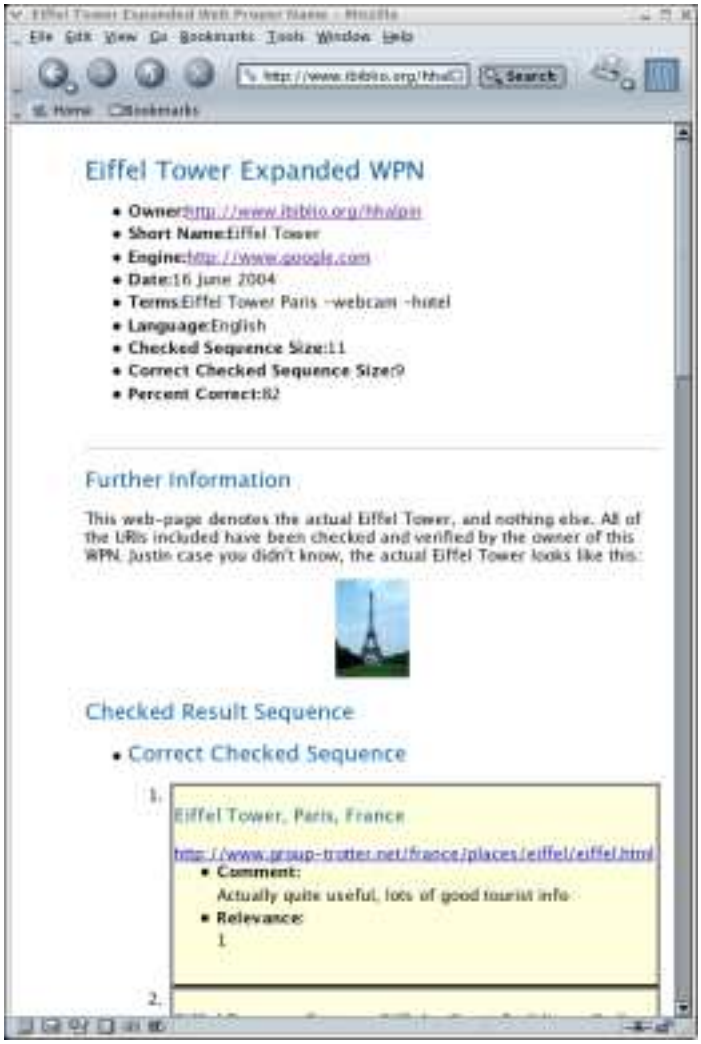

Figure 4: Screen Shot of an Expanded WPN

6.3 As Improved Bookmarks

The storing and collection of EWPNs can then be easily integrated into web browsers in the same fashion as bookmarks. There are already several established XML-based bookmark schemes like XBEL (XML Bookmark Exchange Language), and a transformation from a RDDL EWPN format to a more barebones XML EWPN format suitable for use with web browsers has been created (Drake, 1998). Current XML bookmark schemes and WPNs share the same authority metadata, namely the time and owner information. Yet

while a bookmark is shorthand for a single URI, a WPN is a shorthand for a group of URIs about a particular referent. The coincides with the informal practice of Web users to group ordinary bookmarked URI together by subject. An EWPN offers a crucial advantage over ordinary bookmarks: a set of search terms. Instead of manually marking a bookmark, a user often will remember a set of search terms and type these in their favorite search engine, returning a set of Web-pages, one of which usually is the page containing the information for which they were searching. Due to rate of change of URIs, the use of a search engine can often return not only more a more up-to-date URI than a possibly outdated or broken bookmark, but a cluster of web pages that contain more information about the subject of interest as an added benefit. By preserving the set of search terms, one keeps both the benefit of static bookmarks and search engines.

We also allow the creation of EWPNs in the same manner bookmarks. A user can type a set of search terms about a particular referent into a search engine, which are saved with relevant authority data, and then browse through the search results. When the user finds a search result that is pertinent to their referent, they can add that URI to the checked reference set in the exact same manner one could create a bookmark, by adding it with a few clicks of a mouse. Since the search terms are preserved, it is easy to later repeat the search for more resources about a referent without being forced to remember a set of search terms.

6.4 The Semantic Web from the Bottom-Up

There is movement to store bookmarks as RDF as exemplified by Annotea's bookmark scheme (Koivunen et al., 2003). Since a bookmark can be stored as a metadata about in a particular web page, and in a similar manner an EWPN can be stored as RDF; the canonical RDDL format easily transformed into RDF, since the base component of a WPN are URIs. This would allow the expressive power of OWL to be used in the management of EWPNs. For example, *unionOf* and *intersectionOf* can then be used automatically merge EWPNs and find difference sets of EWPNs. From the viewpoint of ontology development, this also provides a very attractive methodology for building web ontologies. First, many referents are things in the world that are amendable to being part of an ontology. For example, you may have a WPN about The Eiffel Tower, which could have a sub-class relationship to Tourist Destinations in France WPN, which itself could have a sub-class relationship to Tourist Destinations in Europe. This allows the hierarchical structure of many WPNs to be adequately captured. Additional assertions can be made about the referent itself through making metadata statements about the EWPN. The Eiffel Tower EWPN could have an `architect` property that mentions a Gustave Eiffel EWPN. This allows generalized information about the referent (such as the Eiffel Tower being in France) to be stored in the EWPN, while connecting that information to the URIs that support it, in a manner similar to the Trellis system (Gil, 2003). Lastly, the members URIs of the checked sequence of a EWPN could automatically have a *sameAs* OWL property attached to them. WPNs provide a natural way for everyday users of the Web to build ontologies in a an analogous way that they currently build hierarchies of Web bookmarks.

This use of WPNs provides an alternative methodology for the development of the Semantic Web other than the top-down methodology that hopes large organizations will come to agreement on standard ontologies for various domains. In contrast, the bottom-up methodology notes users are already creating rough and ready ontologies at home through their web searches, and storing them as bookmark hierarchies. The Semantic Web effort should not fail to capitalize on this behavior, and the WPN effort captures this behavior in a principled way compatible with ontology development.

7 Conclusion

There is much work to be done. Since WPNs have yet to be tested on a large scale, the exact form of the `wpn:` URI scheme, as well as the inventory of information included therein, cannot be confidently said to be optimal. Likewise the shape and contents of EWPNs will probably be in need of extensions and revisions.

To begin to gain practical experience with WPNs and EWPNs, a number of browsers need to have working WPN implementations, and a WPN support for Mozilla is currently under development. A web resource, *www.webpropernames.org*, currently exists for the further development of Web Proper Names, including community feedback on the conceptual apparatus. Lastly, while WPNs currently allow users to bootstrap ontologies from their WPN usages, much further work could be used on how these ontologies can be developed and merged, and how the bottom-up strategy of ontology creation can best work with larger-scale top-down ontology developments.

WPNs are one proposal for addressing the problem of reference for the Web. This problem is fundamental for the Web, involving the crucial aspects of co-reference and identity. The Web Proper Name proposal, by making a clear distinction between a referent and URIs for web pages about that referent, adds to the conceptual apparatus needed to tackle this problem. By offering a series of concrete XML-based formats and implementations, applications that exploit this distinction can be built. In is in all our best interest, from the everyday user to professional ontologists, to put semantics, in all of its mystery and power, back into the Web.

Acknowledgments

The ideas in this exposition were influenced of through conversations with Brian Cantwell Smith, Fred Dretske, and Paul Schweizer.

References

Borden, J. and Bray, T. (2002). Resource Directory Description Language. Technical report, RDDL Group. `http://www.rddl.org/`.

Drake, F. (1998). The XML Bookmark Exchange Language. Technical report, XBEL. `http://pyxml.sourceforge.net/topics/xbel//`.

Frege, G. (1892). Uber sinn und bedeutung. *Zeitshrift fur Philosophie and philosophie Kritic*, 100:25–50.

Gil, Y. (2003). Knowledge Mobility: Semantics for the Web as a White Knight for Knowledge-Based Systems. In D. Fensel, J. Hendler, H. L. and Wahlster, W., editors, *Spinning the Semantic Web*. MIT Press, Cambridge, Massachusetts.

Goodman, N. (1976). *Languages of Art: An approach to a theory of symbols*. Hackett Publishing.

Guha, R. (2004). Semantic Negotiation: Co-identifiying Objects across Data Sources. In *In Proceedings of Semantic Web Services Symposium*, Palo Alto, CA.

Jacobs, I. (2004). Architecture of the World Wide Web. Technical report, W3C. `http://www.w3.org/TR/webarch/`.

Koivunen, M.-R., Swick, R., Kahan, J., and Prudhommeaux, E. (2003). An Annotea Bookmark Schema. Technical report, W3C. `http://www.w3.org/2003/07/Annotea/BookmarkSchema-20030707`.

Kripke, S. (1972). *Naming and Necessity*. Harvard University Press.

Lenat, D. and Feigenbaum, E. (1987). On the thresholds of knowledge. In *In Proceedings of International Joint Conference on Artificial Intelligence*.

Russell, B. (1905). On Denoting. *Mind*, 14:479–493.

Searle, J. R. (1958). Proper Names. *Mind*, 67:166–173.

Smith, B. C. (1991). The Owl and the Electric Encyclopedia. *Artificial Intelligence*, 47:251–288.