Naming in OHP

Sigi Reich, Dave M. Millard
{sr, dem97r}@ecs.soton.ac.uk
Multimedia Research Group
University of Southampton
SO17 1BJ, UK

Hugh C. Davis
hcd@multicosm.com
Multicosm Ltd
Chilworth Science Park, Southampton
SO16 7NS, UK

Abstract

Naming is a key issue in any distributed system. In particular, with the Open Hypermedia Systems Working Group's efforts towards openness and interoperability in Open Hypermedia Systems (OHS) resulting in the need for (globally) valid names for all types of resources the issue of naming has become increasingly important. In this position paper we examine the issues involved in naming and a present a proposal for naming to be used within the Open Hypermedia Protocol (OHP).

Keywords: Naming, Distributed System, Open Hypermedia System (OHS), Open Hypermedia Protocol (OHP), World-wide Web

1 Introduction

Naming plays a major role in any distributed system. The World-wide Web for instance has benefitted from clear naming via the use of URLs (Uniform Resource Locators): these have become an easy to use and thus popular means for exchanging information about a document's location. At the same time however, URLs also demonstrate the importance of naming in cases where moved or renamed documents result in invalid names that lead to unaddressable resources ('404 document not found').

Naming as an issue in Open Hypermedia Systems, in particular those 'talking' OHP, has been raised several times (e.g. [16,17,12]). In this paper we investigate issues and requirements related to naming in Open Hypermedia Systems and present an approach for a naming scheme to be used within OHP.

From an OHP's point of view it is not so much an issue to develop our own naming system. We basically can rely on existing systems (see Section 3 below) and agree on naming conventions. However, some aspects of the proposed naming scheme might influence the architecture and/or operation of OHP. For instance, we might need a standardised 'lookup service'; or, there will be cases where unique object IDs might not be suitable as parameters of messages because the requesting agent is interested in *any* instance rather than a particular instance (e.g. when requesting a service that is provided by more than one components, it is usually not important which particular component actually fulfils the service).

This position paper is structured as follows. Section 2 describes issues of naming in open hypermedia systems. Next, Section 3 briefly summarises existing approaches. Furthermore, we present a naming scheme to be used within the Open Hypermedia Protocol (see Section 4). We finish with a summary in Section 5.

2 Hypertext Resources and Issues

There are a number of hypertext objects (or resources) that are subject to naming. The OHSWG has already agreed on a common data model for OHP that comprises these objects (see http://www.ohswg.org/) and therefore we want to briefly recapitulate them here:

- **Node**: a node is a wrapper object which holds the meta-data about some content data including the information about where to find that content data. With many OHSs not dealing with document storage any naming conventions can only be applied to the (wrapping) node object but not to the document itself.
- **Anchor**: anchors define the node and the point within that node at which applications should indicate some kind of persistent selection (a 'hotspot').
- **Link**: a link is an object which represents a connection between zero or many endpoints.
- **Linkbase**: a linkbase is a collection of links. In OHP it is represented as a 'context' object.
- **Computations**: many linkservers provide extra services (called computations in OHP) which assist the user

in navigation such as finding similar nodes, etc.

As with naming in any distributed system there are a number of issues to be addressed by OHSs. These include

- **Scope**: names should be globally valid, their meaning should not depend on the location [18], i.e. a resource's name should not give any hint as to where the resource is (physically) located (location transparency, see [19]).
- **Uniqueness**: names should be globally unique, i.e. the same name will never be assigned to two or more different resources. Furthermore, the lifetime of a resource is potentially unlimited so names have to be persistent [18].
- **Readability**: names should be readable and interpretable by humans and they should be easily transcribable [1,10]. E.g., the World-wide Web's URLs have better readability in general than Hyper-G's URLs.
- **Scalability**: there are different aspects to scalability such as number of accesses, number of resources, etc. With respect to the number of accesses it can be said that a central broker dealing with name resolution will scale badly even when replicated; with respect to the number of resources it has to be said that resources might conceivably be available on the network forever, i.e. the number will be very high.
- **Mobility**: resources might be subject to re-location at other places [11]. For instance, a set of links might be imported from a different linkbase (and naming clashes will have to be resolved) or also services might be mobile and have different locations over time. Naming should be location independent, i.e. resources should be relocatable without affecting their names [19].
- **Querying**: often users will not know the exact name of a resource so that it can be looked up. However, they will know some attributes such as author, type, etc. In this case it should be possible to provide the user with a mechanism to look up a name given a set of attributes.

3 Existing Approaches

Due to the importance of naming there are a number of existing naming systems. In this section we will briefly mention the most relevant ones and describe their main characteristics.

DNS (Domain Name System) is defined in the Internet RFCs 1024 and 1035 [13,14]. DNS basically defines a hierarchical naming system that maps ASCII encoded host names to binary network addresses (represented as resource records), which then are used by programs. Generic (high-level) domains such as com, edu, mil, org, net, int, etc. as well as country specific domains (such as uk, us, at, de, etc.) provide a scalable mechanism for assigning and managing names. DNS also builds the basis for the Web's URLs.

X.500 [4] is a service comparable to a telephone directory service. With a network growing in size these services become more and more necessary: users require not only lookup but also search facilities. Hence, X.500 is also referred to as an attribute-based name service [6]. It is based on a tree structure (Directory Information Tree, DIT) and hence supports hierarchical naming. It is standardised by the CCITT and ISO organisations.

Naming and Addressing on the **World-wide Web** are specified by URIs (Uniform Resource Identifiers), URLs (Uniform Resource Locators), URNs (Uniform Resource Names) and URCs (Uniform Resource Characteristics) [5,18,1]). Figure 1 gives an overview.

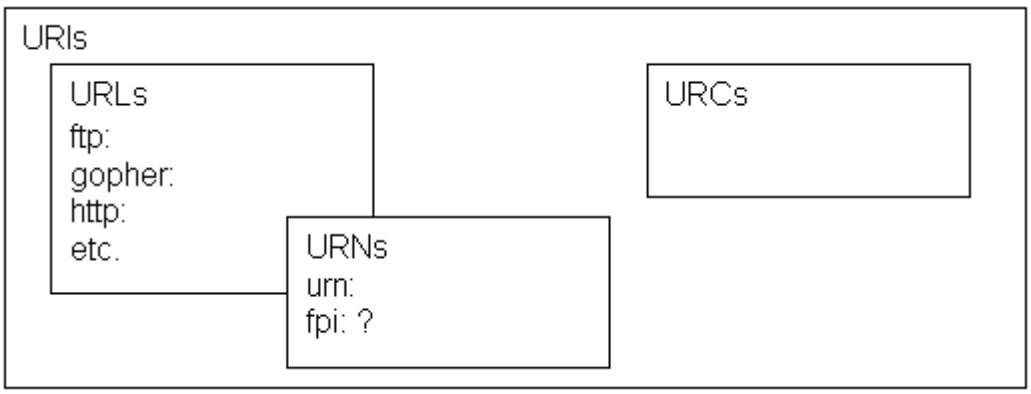

Figure 1: URIs, URLs, URNs, URCs [5]

The figure shows that URIs (Uniform Resource Identifiers) are the 'umbrella' for a number of naming concepts. A

URI defines the way how content on the Internet is identified. The most common form of URI is the Web page address, which is a particular form or subset of URI called a Uniform Resource Locator (URL).

Another kind of URI is the Uniform Resource Name (URN). A URN is a form of URI that has 'institutional persistence,' which means that its exact location (i.e. its URL) may change from time to time, but some agency will be able to find it. The term 'fpi' in the figure denotes Formal Public Identifiers as known from SGML (see e.g. [7]).

URCs define the syntax for carrying meta data about a resource such as owner, encoding, access restrictions, etc.

Persistent URLs (PURLs, [20]) are basically URLs. However, instead of pointing directly to the location of a resource, they point to an intermediate resolution service, which resolves the PURL to the URL, e.g. via a standard HTTP redirect.

Other systems dealing with naming and supporting similar features as the ones mentioned include Grapevine [3], a distributed system developed in the early eighties at Xerox, or the lookup services whois and whois++ [9].

4 A Proposal for Naming in OHP

The Open Hypermedia Protocol has been around for quite some time and has been used and demonstrated at various occassions. It is interesting to see that naming in OHP has been raised as an open issue several times (e.g. [16,17,12]) but that there is still no agreed scheme. In particular, the simple prototypes which were done using CORBA and RMI [12] clearly showed the need for a general naming concept. In this section we will briefly outline a proposal for naming.

We believe that with respect to scalability and autonomy of any involved organisations a hierarchical naming concept would be suitable. Based on existing, well agreed systems such as DNS this would also ease the setup. Domain names could hence serve well as one part of names in OHP.

We also think that there should be a two-level naming, i.e. symbolic names should be used to refer to physical names (similar to PURLs or URNs on the Internet, see Section 3). This would mean that we would need a lookup message in OHP that takes a symbolic name as input and delivers (a list of) physical names as output. This would also ease migration of resources to other locations.

With respect to transcribability [2] we suggest to let the organisations and users themselves decide for the names, i.e. by default an OHP compliant system does not assign IDs (though some systems might want to support such a feature). In any case, new names have to be registered with the system and the system has to check for name clashes. This could e.g. be done when hypertext objects are created.

Looking up of objects could be done by using OHP's query features as suggested by [15]. For a network of cooperating OHS systems we might also want to use an approach as suggested by [8] which would support re-routing of queries.

A new OHP session would be started by connecting to the default OHP naming service whose location will differ for the protocol spoken. An instance of this 'location service' (also called 'Server Information Manager (SIM)' [16]) is expected to exist per OHS domain. As mentioned, OHS domains are like DNS domains, whether an OHS service exists for a domain is decided by the organisation (hence, there might be an OHS location service for 'ecs.soton.ac.uk' but not for 'soton.ac.uk').

With respect to merging and mobility we believe that the proposed approach will allow transparent moving of objects *within* a domain. However, moving objects between domains might result in name clashes; these could e.g. be resolved by introducing additional hierarchies. It is expected that an object's (previous) domain supports a 'forwarding' service for lookups (such that names stay always valid).

Thus, we propose the following naming for OHP which follows the known syntax of URLs on the World-wide Web. An OHP name basically consists of two parts, a domain name (global part, see [16]) which equals the organisation's domain name and an (optional) OHP object name which is assigned locally, i.e by the organisation itself. A "/" is used as a separator; "OHP://" is used to show that the protocol spoken is OHP:

```
"OHP://" <domain name> "/" <OHP object name>
```

e.g. we would have

`OHP://mmrg.ecs.soton.ac.uk/Dave's Home Page` or

`OHP://daimi.au.uk/Pete's OHP Linkbase Set or`

`OHP://multicosm.com/`

It is expected that organisations will introduce their own structures (e.g. hierarchies, user level spaces, etc.) as appropriate. In case where the local part is left empty (see the last example above) the request (i.e. lookup) will refer to the default context provided by an OHP compliant server. This is very much comparable to the root hierarchy of a Web server. The main purpose is to provide by default a starting point for a navigational browsing session.

5 Summary

In this position paper we have shown the issues involved in naming in Open Hypermedia Systems. We have presented existing approaches and a proposal for a two-level naming which is an extension of the familiar notion of URLs as known from the Web. We believe that this proposal will serve as a starting point for building globally distributed open hypermedia systems that use OHP as their common protocol.

References

[1]
Berners-Lee, T. Universal resource identifiers in WWW. a unifying syntax for the expression of names and addresses of objects on the network as used in the world-wide web. Tech. rep., Internet RFC 1630, June 1994.

[2]
Berners-Lee, T., and Fielding, R. Universal resource identifiers in WWW. a unifying syntax for the expression of names and addresses of objects on the network as used in the world-wide web. Tech. rep., Internet RFC 2396, Aug. 1998.

[3]
Birrell, A. D., Levin, R., Needham, R. M., and Schroeder, M. D. Grapevine: an exercies in distributed computing. *Communications of the ACM 25* (1982), 39-59.

[4]
CCITT - International Telecommunications Union, Geneva, Switzerland. *Recommendation X.500: The Directory - Overview of Concepts, Models and Service*, 1988.

[5]
Conolly, D. Naming and addressing: URIs. Tech. rep., World-wide Web Consortium, Available as http://www.w3.org/Addressing/, 1999.

[6]
Coulouris, G., Dollimoore, J., and Kindberg, T. *Distributed Systems. Concepts and Design*, 2 ed. Addison-Wesley Publishing Company, 1994.

[7]
Daniel, R. Iso formal public identifiers as a urn namespace. Tech. rep., Los Alamos National Laboratory. Available as http://www.acl.lanl.gov/URN/FPI-URN.html, Apr. 1996.

[8]
DeRoure, D., El-Beltagy, S., Carr, L., and Hall, W. A distributed link service using query routing. Tech. rep., Multimedia Research Group, Department of Electronics and Computer Science. University of Southampton. Available as http://www.ecs.soton.ac.uk/~dder/qdls/, Dec. 1998.

[9]
Deutsch, P., Schoultz, R., Faltstrom, P., and Weider, C. Architecture of the whois++ service. Tech. rep., Internet RFC 1835, Aug. 1995.

[10]
Engelbart, D. C. Knowledge-domain interoperability and an open hyperdocument system. In *CSCW 90. Proceedings of the Conference on Computer-Supported Cooperative Work* (Oct. 1990), pp. 143-156.

[11]
Guenther, R. Naming conventions for digital resources. Tech. rep., Library of Congress. Available as http://lcweb.loc.gov/marc/naming.html, 1999.

[12]
Millard, D. E., Reich, S., and Davis, H. C. Reworking OHP: the road to OHP-Nav. In *4th Workshop on Open Hypermedia Systems (OHSs) at Hypertext '98, Pittsburgh* (June 1998), pp. 48-53. Available as Technical Report No. CS-98-01 from the Dept. of Computer Science Aalborg University Esbjerg, Denmark.

[13]

Mockapetris, P. Domain names - concepts and facilities. Tech. rep., Internet RFC 1034, Nov. 1987.

[14] Mockapetris, P. Domain names - implementation and specification. Tech. rep., Internet RFC 1035, Nov. 1987.

[15] Nürnberg, P. J., Grønbæk, K., Sloth, L., Bouvin, N.-O., and Wiil, U. K. Notes on modification of the ohswg navigation standard and related interfaces. Tech. rep., Open Hypermedia Systems Working Group (OHSWG), Sept. 1998.

[16] Nürnberg, P. J., and Leggett, J. J. A vision for open hypermedia systems. *Journal of Digital Information (JoDI). Special Issue on Open Hypermedia Systems 1*, 2 (1997).

[17] Nürnberg, P. J., Leggett, J. J., and Wiil, U. K. An agenda for open hypermedia research. In *Proceedings of Hypertext '98, Pittsburgh, PA* (1998), pp. 198-206.

[18] Sollins, K., and Masinter, L. Functional requirements for uniform resource names. Tech. rep., Internet RFC 1737, Dec. 1994.

[19] Tanenbaum, A. S. *Distributed Operating Systems*. Prentice-Hall International, Englewood Cliffs, New Jersey, 1995.

[20] Weibel, S., Jul, E., and Shafer, K. PURLs: Persistent uniform resource locators. Tech. rep., Online Computer Library Center (OCLC), Available as http://purl.oclc.org/OCLC/PURL/SUMMARY, 1999.

File translated from TEX by TTH, version 1.98.
On 11 Jan 1999, 11:14.