Implementing Persistent Identifiers

Hans-Werner Hilse and Jochen Kothe

Implementing Persistent Identifiers

This report was written by the Research and Development Department of the Goettingen State and University Library (Niedersächsische Staats- und Universitätsbibliothek Göttingen) at the request of the Advisory Task Group of the Consortium of European Research Libraries.

Published by
Consortium of European Research Libraries, London, www.cerl.org
European Commission on Preservation and Access, Amsterdam, www.knaw.nl/ecpa
November 2006

ISBN 90-6984-508-3
Available as PDF at www.cerl.org and www.knaw.nl/ecpa
Identifier urn:nbn:de:gbv:7-isbn-90-6984-508-3-8
(resolving service http://nbn-resolving.de)
(http://nbn-resolving.de/urn:nbn:de:gbv:7-isbn-90-6984-508-3-8)

Typesetting and design: Edita, Royal Netherlands Academy of Arts and Sciences

The paper in this publication meets the requirements of the ∞ ISO-standard 9706 (1994) for permanence.

Implementing Persistent Identifiers

Overview of concepts, guidelines and recommendations

Hans-Werner Hilse and Jochen Kothe

Consortium of European Research Libraries
European Commission on Preservation and Access
2006

About the authors

Hans-Werner Hilse has been working at the Research and Development Department at the Niedersächsische Staats- und Universitätsbibliothek (Lower Saxony State and University Library, SUB) Göttingen, Germany, since 2002. He also works at the same institution for the University Press. He is a specialist in computer programming as well as e-publishing and participates in working groups of the German Initiative for Networked Information (DINI), electronic publications. One of his main interests is the analysis of complex systems, which also explains why he is still engaged in the study of law.

Jochen Kothe has been working at the SUB as senior programmer and computer administrator since 2003. In recent years, he has been organizing information systems for the digitization facility located in the SUB (Göttinger Digitalisierungszentrum, GDZ), where he has to deal with persistent identification of resources of many kinds. Jochen Kothe is also involved in various publicly funded projects for preservation of cultural heritage.

Consortium of European Research Libraries

CERL aims to share resources and expertise between research libraries to record, preserve and exploit the materials for the history of Europe's written and printed cultural heritage. The Hand Press Book Database is a collaborative catalogue for European printed books before c. 1830. The CERL Thesaurus records variant forms, in Latin and European languages, of names of places, printers and authors found in books of the hand press period. The CERL Portal gives integrated access to online manuscript catalogues with the option of combining searches with the Thesaurus and the HPB.

Consortium of European Research Libraries (CERL)
40 Bowling Green Lane , Clerkenwell
London EC1R 0NE UK
secretariat@cerl.org
www.cerl.org

European Commission on Preservation and Access

The European Commission on Preservation and Access (ECPA) promotes activities aimed at keeping collections in European archives and libraries accessible over time. The ECPA acts as a European platform for discussion and cooperation of heritage organizations in areas of preservation and access. To promote the exchange of experience and expertise, the ECPA organizes conferences, meetings and workshops.

European Commission on Preservation and Access (ECPA)
Royal Netherlands Academy of Arts and Sciences
P.O. Box 19121, 1000 GC Amsterdam, The Netherlands
ecpa@bureau.knaw.nl
www.knaw.nl/ecpa

Executive Summary

Traditionally, organisations have relied on URL hyperlinks to provide interested parties with access to their digitised content via the internet. However, over time, more and more of these hyperlinks are ‘broken’. The URL relies on providing the specific location details for a document. When, for example, an organisation’s website is re-organised and its directories are renamed, the URL no longer provides a correct location path, thus rendering the documents effectively inaccessible to the end-user.

In the mid 1990s, a number of schemes were developed that, rather than relying on the precise address of a document, introduced the idea of name spaces for recording the names and locations of documents. The identifiers for documents are registered centrally. When an end-user wishes to access a certain document, the identifier in his request is ‘resolved’, i.e. the correct document is retrieved, without the end-user needing to know the exact location of the document. This report describes a number of such schemes in detail.

Key concepts introduced include Handles, Digital Object Identifiers (DOIs), Archival Resource Keys (ARKs), Persistent Uniform Resource Locators (PURLs), Uniform Resource Names (URNs), National Bibliography Numbers (NBNs), and the OpenURL. These schemes are described with examples and extensive references.

The report emphasises that supporting persistent identification requires administrative effort and commitment. The systems presented support these administrative tasks but do not render them obsolete. All changes in location, ownership or metadata must be reflected in the name-space system – causing the organisations that run an identification system to incur costs. To assist organisations that wish to implement a persistent identification scheme, the report details questions that need to be addressed and offers possible strategies to tackle a number of scenarios. Organisations are strongly recommended to investigate collaboration with partners with existing schemes that have similar problems to solve and to choose the syntax for their persistent identifiers in such a way that they can be integrated into any of the schemes introduced in this report.

History

The Advisory Task Group (ATG) of the Consortium of European Research Libraries (CERL) commissioned this report from the Research and Development Department of the Niedersächsische Staats- und Universitätsbibliothek in March 2005. The ATG had found that URL links in records in the Hand Press Book Database (HPB) – which is compiled by CERL – frequently after a certain period of time no longer provided access to the digital content they were originally intended to link to. The ATG’s intention with this report was to offer CERL members and HPB data providers a structured overview of the schemes that have been developed to support persistent identification as well as pointers for establishing such a scheme in their own libraries. However, the topic of this report is of importance to organisations beyond the CERL membership. CERL was, therefore, delighted to find the European Commission on Preservation and Access (ECPA) willing to act as a co-publisher of this report, as this will ensure wider dissemination of its contents.

Acknowledgements

The authors would like to thank Thomas Fischer, of the Goettingen State and University Library, and especially John Kunze, California Digital Library, for their helpful comments and suggestions (and even a few corrections). We are also grateful to David Shaw, CERL Secretary, and Anthony G. Curwen, CERL consultant, for their help, to Marian Lefferts, CERL Executive Manager, for her invaluable efforts in managing the publication process, and to Yola de Lusenet, Executive Secretary of the ECPA, and the ECPA staff.

Contents

1 Introduction 1
1.1 The problem: links ‘break’ 1
1.2 Current concepts: WWW and URLs from a technical point of view 2
1.3 What is wrong with URLs? 6
1.4 What next 7

2 URN-based mechanisms 8
2.1 History 8
2.2 Functionality 8
2.3 Implementations 9
2.4 Current applications, long-term perspective 13
2.5 Participation 13
2.6 Summary 13

3 National Bibliographic Numbers (NBN) 14
3.1 History 14
3.2 Functionality 14
3.3 Implementation 15
3.4 Current applications 16
3.5 Long-term perspective 16
3.6 Participation 16
3.7 Summary 16

4 Handles 17
4.1 History 17
4.2 Functionality 17
4.3 Implementation 18
4.4 Current applications 19
4.5 Long-term perspective 19
4.6 Participation 20
4.7 Summary 20

5 Digital Object Identifiers (DOI) 21
5.1 History 21
5.2 Functionality 21
5.3 Implementation 22
5.4 Current applications 24
5.5 Participation 24
5.6 Summary 25

6 Archival Resource Keys (ARK) 26
6.1 History 26
6.2 Functionality 26
6.3 Implementation 26
6.4 Participation 30
6.5 Summary 31

7 Persistent URLs (PURL) 32
7.1 History 32
7.2 Functionality 32
7.3 Implementation 33
7.4 Current applications 34
7.5 Long-term perspective 34
7.6 Participation 34
7.7 Summary 35

8 OpenURLs 36
8.1 History 36
8.2 Functionality 36
8.3 Implementation 36
8.4 Current applications 39
8.5 Long-term perspective 39
8.6 Participation 39
8.7 Summary 39

9 Guidelines and recommendations 40
9.1 Determining the status quo 40
9.2 Strategies to choose 41
9.3 Long-term perspective of software, protocols and concepts 45
9.4 Support through use: identifier politics 45
9.5 Joining the discussion 46
9.6 Why not recommend a specific implementation 47
9.7 Checklist 47

Appendices

A. Timeline 49
B. Glossary 50
C. Further information 53

References 54

1 Introduction

In the past two decades, the importance of the Internet as a platform for academic resource materials has continuously increased. When first introduced, it was a medium mainly for scholarly communication and remote control of computers, but with the general acceptance of the World Wide Web (WWW) it has become a medium for publication and access to scientific information. Not only commercial publishers, but a wide variety of academic and cultural institutions offer different types of electronic publications on the Web. Tens of thousands of serials are published in electronic form, universities allow their PhD students to publish their theses electronically, and in digitization projects huge amounts of paper materials are converted to electronic documents.

The advantages of electronic publication for instant access and easy, low-cost distribution and duplication are obvious. One of the main concerns for the development of a stable information infrastructure, however, is the long-term management of the digital environment.[1] Digital preservation not only deals with migration to new carriers and new formats and with maintenance of functionality, it starts with a very basic requirement: namely, that documents can be identified unambiguously and located by those who need them.

However, this is not always the case. Over the years, the phenomenon of broken links or 'link rot' has become widespread. This undermines the value of the electronic environment as a publication medium, as the tradition of publishing and research strongly depends on reliable referencing.

1.1 The problem: links 'break'

Documents on the WWW are commonly referenced by a Unified Resource Locator (URL). These URLs are used to create hyperlinks on the web (and for other protocols and purposes). Because the access method – viewing or requesting documents via the Internet – required the use of a URL as addressing mechanism, the URL also became the preferred method of referencing documents, e.g. in a citation.

Over time the risk grows that the document is no longer accessible at the location given as reference. Web servers that follow the HTTP protocol then give the notorious reply: '404 not found'. This resembles the situation of a book in a – very large – library that is not on the shelf at the position indicated in the catalogue. How is it to be found?

It would be even worse if no such error message appeared: a URL may also be

[1] See e.g. the 'Preserving Access to Digital Information' (PADI) website for more information on current projects: http://www.nla.gov.au/padi/

unstable in that it now points to a different resource that has replaced the earlier one. Here, the link would be ‘broken’ too, but users may not recognize this, as there is no error message to alert them and the ‘wrong’ document is presented instead.

Breaking of links is mostly due to administrative changes at the referenced Internet node. Broken links have a negative impact on all documents referencing these URLs. The only solution for the end user is to use additional metadata (such as author or title of the publication) and start a search for the object, using available search technology (‘google it’) or specialised information, e.g. on the institution (university, library, publisher) supposedly holding stewardship or ownership of the object. This has an adverse influence on several things:

- Citations are more complex to track down and may even become invalid. So a broken link will even compromise other documents that themselves may still be accessible but do use the broken link in a citation.
- A lot of work is needed to retrieve the ‘lost’ resource, if this is possible at all. There may just not be any information about accessibility left although the resource may still be accessible somewhere.
- Visibility of the document itself is compromised as visibility mainly depends on citations and stored references (links) from other documents and databases.
- Scientific results – often funded by the public – are lost.

To understand the problem in detail, we will have a short look at the techniques that lie behind such acronyms as URL and HTTP and briefly outline the concept of the WWW.

1.2 Current concepts: WWW and URLs from a technical point of view

1.2.1 The Internet Protocol

IP/IPv4

The first address mechanism of the Internet was based on the Internet Protocol (IP).[2] Data packets exchanged over the Internet always carry source and destination information, the IP address. This is a combination of four octets (of which each carries 8bit of information, i.e. an IP address has 32bit in total), commonly written as dot separated numbers between 0 and 255. As there are several reserved combinations with special meanings, not all 16.7 million possible numbers are available for addressing.

Ranges of these numbers were assigned to institutions. These institutions manage the outgoing routing for those addresses outside of their range and on the other hand manage the distribution of incoming traffic to the target identified by an address in the pool of managed addresses. In fact, there is an additional infrastructure to manage the routing of Internet packets to their destination. This

[2] RFC 791.

routing layer is commonly invisible to the end user and often referred to as the 'backbone' infrastructure of the Internet. The layer is used for routing traffic between the institutions that manage address ranges.

IPv6

A new protocol for the Internet has already undergone standardisation and is ready to be used: IPv6.[3] Designed with compatibility to the original IP protocol (referred as 'IPv4') in mind, it allows a much bigger address space that makes the concept of routing and private subnets easier to achieve. Most software currently capable of IPv4 should be able to adopt the new IP protocol without great difficulty.

1.2.2 The Domain Name System

To ease the use of the Internet for humans the Domain Name System (DNS)[4] was established to build a layer based on an alphabet and a systematic syntax to access an IP address. The DNS protocol allows a client to ask a name server for the IP address that belongs to a domain name. However, there were means to use lexical descriptions to substitute IPs earlier. Computers used to have a local list (managed by the administrator) which carried host names as aliases for IPs. This did not scale well, nor was it an easy undertaking to ensure that all hosts on the Internet were aware of all the latest additions and corrections to the list.

To remedy this, the DNS is now hierarchically structured in both its naming convention as well as the service it provides. Each level of authority for assigning the names is expressed in the full domain name. The topmost authority starts at the right and is separated from the next level authority by a dot (e.g. *www.cerl.org*).

There is a fixed set of *root name servers* which always provide the first level of authority that is expressed in the last element of a domain name (i.e. at the far right). In most cases, this is a country domain (.uk, .fr, .de) or some general 'category name' describing purpose or community (.com, .org, .net). These authorities manage all prefixes to this 'top level' domain. These days, the second authority is usually the most important part of a domain name and is usually administered by an institution or person. There are exceptions to this general rule: some countries have set up an additional structure between the national and the 'private' level, e.g. the UK uses .co.uk for commercial providers, .ac.uk for the academic community etc.

Further subdivision of the domain name by creating subdomains is the prerogative of the 'owner of a domain'. The concept of 'ownership' of a domain can be better described as having leased it from the next upper node in the domain name system hierarchy.

DNS was designed as a distributed service that allows local caching. Although its design took place in a time when no one would have thought about millions of

[3] RFC 2460.

[4] RFC 1034, RFC 1035.

users, it has proven to scale well. Virtually all communication between nodes on the Internet makes use of the DNS in some way; its availability is actually crucial for most services on the Internet.

1.2.3 The World Wide Web

Today the most commonly used access mechanism to documents on the Internet is the World Wide Web (WWW) and its Hypertext Transport Protocol (HTTP). It uses TCP/IP as its underlying network protocol, which is a connection protocol layer on top of the above described IP layer. The World Wide Web uses the hypertext mark-up language (HTML) to describe documents and hyperlinks – identifier strings that are embedded in hypertext documents and are used to assemble the final documents (e.g. images) and provide links as entry points to new documents or views. Web browser software allows users to interact with these documents. It retrieves them from a Web server and renders them to the user.

1.2.4 The URI

The Uniform Resource Identifier (URI) concept was first published by Tim Berners -Lee in 1994 as Universal Resource Identifier.[5] The term URI had been used previously in discussions about the implementation of the Web. The URI can be classified as either (a) a name, or (b) a locator, or (c) both a name and a locator. Both concepts, names and locators, have historically gone different ways, as URNs and URLs.

a. URLs

The HTTP protocol uses Uniform Resource Locators (URLs) for addressing documents. URLs are a certain kind of a URI and are only made for locating resources. The concept of URLs was developed for the technical implementation of the first Web software in 1991 and onwards. It even predates the larger concept of URIs although issues like distinguishing between location and name were already discussed at that point.[6] URLs were introduced to the public with the early HTTP protocol specifications although they were only referred to as ‘addresses’.[7] The term URI was used in subsequent development of HTTP for both URLs and URNs.[8]

The term URL is, however, commonly used to specify those URIs that are used for addressing rather than for naming.

There are multiple types of information encoded in an URL:

- a scheme that distinguishes the namespace from other kinds of URIs and indicates – in case of an URL – the access mechanism that is registered for that scheme (e.g. ‘http’ for HTTP-based access),

[5] RFC 1630.

[6] Tim Berners-Lee: Design Issues for the Web / Naming.

[7] HTTP 0.9 Addressing.

[8] HTTP 1.0. Request.

- a network path that includes the domain name of the queried host (or alternatively its IP) and optionally a port that differs from the default,
- optionally a path name resolved by the target host's server software,
- optionally additional parameters, query specifications and a fragment specifier for an anchor in an html document.

Examples are:

http://www.knaw.nl/cfdata/epic/announcements.cfm#227
http://www.knaw.nl/cfdata/grip/output/gripresults.cfm?descriptor_id=102

It is important to note that URLs are

- dependent upon the DNS information for the domain name which they contain, for which there is a path of authority from the local DNS registry through the national DNS registrar and up to the DNS root servers,
- dependent upon the correct resolution of the additional information in the file system path and/or query string by the Internet server identified through the DNS.

b. URNs

The Uniform Resource Name (URN) was born out of the idea of providing a means for naming resources instead of addressing them. The concept was published in 1994 as a Request for Comments (RFC).[9] It only specified the requirements for URNs, not their syntax. From the RFC:

> The purpose or function of a URN is to provide a globally unique, persistent identifier used for recognition, for access to characteristics of the resource or for access to the resource itself.

The syntax of URNs was formally described in 1997.[10] URNs carry a namespace identifier (NID) from a defined list that is currently maintained by the Internet Assigned Numbers Authority (IANA).[11] These namespaces allow the integration of other naming schemes (e.g. ISBN, ISSN, SICI) as subsets of the URN namespace without conflicts. URNs are designed to fit the definition of URIs. They are described in more detail in chapter 2.

c. Consolidation of both concepts

The current URI specification[12] suggests that future specifications and documentation should use the general term 'URI' rather than the more restrictive 'URN' and 'URL' terms.[13] The first part of an URI always identifies a certain scheme. Although this scheme has different meanings for URNs and URLs, the scheme

[9] RFC 1737.

[10] RFC 2141.

[11] http://iana.org/ .

[12] RFC 3986.

[13] RFC 3986, 1.1.3.

'urn' is reserved for URNs and thus integrates URNs as a subset clearly separable from URLs into the URI namespace. So a given URI can be identified as either an URN or an URL by looking at its scheme.

1.3 What is wrong with URLs?

So what is wrong with the currently used URL-approach? We already mentioned that from a user's point of view links appear to be 'broken'. This is mostly due to one of the following:

(a) the document is no longer available, or

the document is available, but was

(b) relocated so that it is now accessible using a different domain name,
(c) relocated on the same Internet server so that it is now accessible with an other file system path or query specification.

One could argue that these problems are simply due to inadequate administration. However, scenario (b) above, where domain names are used in the network path of URLs, illustrates an additional difficulty. Domain names are not stable. They represent trademarks, company names or names of departments, which may change. Such names changes will then need to be reflected in the domain names. While it is actually possible to create DNS names that are pure numbers or otherwise devoid of meaning (e.g., creating time coded domains[14]), both public and private name assigners have mostly ignored this possibility of the DNS and have turned to other schemes like ARK or Handle that explicitly require numeric institutional designations.

Scenario (c) above, can certainly be attributed to a lack of administration. Today's Web servers allow a high level of indirection when it comes to mapping the URL path name to a real file on the server's file system, to a file on another server or to the path to a Web application running on the local Web server. This means, an administrator could set up rewriting rules and a redirection (relocation) table on the Web server so that 'old' URLs are kept functional.

A problem here is that in many cases a new system (e.g. repository software on a Web server) introduces its own URLs for the documents. After migration from a system used before, the old URL would be mapped to the new one. If this document is from then on cited with the new URL, too, it would be hard to tell if a document citing the old URL and a document citing the new URL refer to the same document without comparing them. The original document would end up with two URL-based identifiers. Of course, software could overcome this by integrating old, legacy identifiers so that the document is not only accessible with its old URL but also still displays the old URL as its address. Most of today's document management systems, however, do not have such a feature. So in the current situation, migration of URL-identified documents would probably lead to

[14] Tim Berners-Lee, Axioms of the Web architecture 2, Section 'Naming: A social and contractual Issue'.

additional URLs being assigned to each document migrated.

This illustrates the problem with URLs: they are a crucial technical element of the Web, as shown above. The positive aspect is that they are technically very well integrated into the available software. However, there is no way to express levels of commitment regarding the persistence or time span of validity in an URL. Some URLs may only work once for the session of one user only, some may be maintained for persistent access. To let users know about its policy on URLs, an institution would have to write a description for the users. If, on the other hand, a user encounters an URL without such a description, it would not be clear how durable the URL is. This compromises the acceptance of URLs for durable identification.

The systems introduced in this report all provide a means of clarifying the policy regarding the persistence of the identifiers. The ARK system introduced here even offers a standardised way to access a commitment statement from the institution in charge of the administration of an encountered identifier.[15] Generally speaking, the usage of a system for persistent identification is in itself a statement by the institution introducing the system about their commitment regarding their identifiers' persistence.

1.4 What next

An important step for institutions dealing with electronic documents should be to create awareness of this problem among their users. They should make administrative rule sets for the identification of their objects according to a deliberate strategy for persistent naming.

In order to evaluate different strategies, we will provide an overview of frameworks, protocols and services which are currently being discussed or are already in use – also outlining under what circumstances their application may be feasible. We will then add comments about criteria that should be kept in mind when choosing a strategy and finally give some advice on taking part in the current discussion of the systems.

[15] J. Kunze: Towards electronic persistence using ARK identifiers. This report explains that persistence needs a commitment statement as a crucial element rather than another level of indirection.

2 URN-based mechanisms

This chapter gives a short overview of different kinds of Uniform Resource Names (URNs) and related technology. There have been several suggestions for the implementation of identifier schemes based on the URN concept. Most of them were not designed with persistence as their primary design goal, although since the beginning of the URN discussion persistence was recognized as an important part of a naming strategy.

2.1 History

As mentioned in the introduction to this report, the URN concept dates back to 1994. The concept is based on a generalized functional specification, 'Functional Requirements for Uniform Resource Names'.[16]

The syntax of URNs was fully specified in 1997 in another RFC, 'URN Syntax'.[17] Since then, many URN namespaces have been assigned to various identification concepts. For a list of those concepts and the assigned namespaces see section 2.3.1 below.

2.2 Functionality

The basic functionality of a URN is resource naming. The requirements that led to further syntax and scheme definitions are outlined in the 'Functional Requirements for Uniform Resource Names' RFC:

- Global scope of the names: they have the same meaning everywhere.
- Global uniqueness: different resources cannot have the same URN.
- Persistence: in the URN context, the name's lifespan is permanent, regardless of the lifespan of the named resource.
- Scalability: room to accommodate the number of names required in the next centuries.
- Legacy support: should allow the integration of other naming schemes.
- Extensibility: future extensions to the URN scheme are possible.
- Independence: determining the conditions for issuing a name is the sole the responsibility of the name-issuing authority.
- Resolution: if a URN corresponds to a URL, there must be some feasible mechanism to resolve it.

These functional requirements are met by all URN-based implementations. The general syntax of URNs is defined as follows:[18]

[16] RFC 1737.

[17] RFC 2141.

[18] RFC 2141.

```
"urn:" <NID> ":" <NSS>
```

Every URN begins with the 'urn:' character string, followed by the Namespace Identifier (NID). The NID can consist of letters, numbers and hyphens. When comparing URNs, the NID is considered to be case-insensitive. As such, identifiers like 'urn:isbn:...' and 'urn:ISBN:...' both refer to the same namespace.

Separated by a single colon, the remainder of the URN consists of a Namespace Specific String (NSS). As its name implies, the NSS's syntax depends on the namespace identified by the NID. The NSS can consist of any possible characters which may have to be encoded using the same encoding method as URLs.

Example: *urn:isbn:3-938616-59-8* (isbn is the Namespace Identifier, and the actual ISBN number is the Namespace Specific String).

2.3 Implementations

2.3.1 URN namespaces

Of all identifier concepts that aim at persistent resource identification, the most noteworthy approach using URNs is the NBN namespace – which will be introduced in more detail in the next chapter. However, many more URN namespaces have already been registered. They are all listed below, with short comments on their scope. Each of these namespaces has its own NSS syntax. The current list is accessible on the IANA website.[19] General considerations on integrating other bibliographic identification schemes as namespaces into the URN concept were published in a separate RFC.[20]

CLEI[21] The CLEI namespace allows for the integration of CLEI Codes into the URN concept. CLEIs are part numbers for global telecommunication network parts. CLEIs are managed by Telcordia Technologies. CLEIs are standardised by ANSI.

fdc[22] This namespace has been assigned for identifying federated content. This is defined as content that is not managed or centrally administered. It uses a domain name and a timestamp in its NSS element. As such, it is an implementation of the time-coded URL concept. At the time of writing this report no detailed information was available, which is why it is not covered in greater detail here.

fipa[23] Allows assigning of URNs in order to identify standard components published by the Foundation for Intelligent Physical Agents (FIPA). Those components can be publications and specifications, as well as parts thereof.

[19] http://www.iana.org/assignments/urn-namespaces
[20] RFC 2288.
[21] http://www.rfc-editor.org/internet-drafts/draft-tesink-urn-clei-00.txt
[22] http://www.rfc-editor.org/internet-drafts/draft-dtessman-urn-namespace-federated-content-03.txt
[23] RFC 3616.

IETF[24] The urn:ietf namespace was assigned for the RFC family of documents developed by the Internet Engineering Task Force (IETF) as well as the minutes of working groups (WG) and birds of a feather (BOF) meetings which occur during IETF meetings.

IPTC[25] Namespace for documents of the International Press Telecommunications Council (IPTC) and other resources created by the IPTC that need to be persistently identified. The need for another namespace arose because the scope of the 'NEWSML' namespace had been defined too narrowly, and related documents could not be identified through identifiers of the 'NEWSML' namespace.

ISAN[26] Allows the integration of International Standard Audiovisual Number (ISAN) into the URN concept. The ISAN is an ISO-approved standard. The issuing of ISANs is administered by the ISAN International Agency.

ISSN[27] Allows integration of International Standard Serial Numbers (ISSNs) into the URN concept.

ISBN[28] Allows expression of International Standard Book Numbers (ISBNs) as URNs.

liberty[29] The Liberty Alliance uses this namespace for persistent identification of various objects that are part of the Liberty Architecture. The Liberty Alliance aims at providing a federated network identity for use in e-commerce, personalized services and other network based services.

mace[30] This namespace has been assigned to the Middleware Architecture Committee for Education (MACE) for assigning URNs to publications as well as identifying directory attributes and controlled vocabularies of those attributes.

MPEG[31] The MPEG namespace has been reserved for naming persistent resources that are parts of published standards by the Motion Picture Experts Group (MPEG).

NBN[32] This namespace has been designed to allow national libraries to integrate their identification concepts into a common URN namespace. This concept is introduced in more detail in the next chapter of this report.

[24] RFC 2648.
[25] RFC 3937.
[26] http://www.rfc-editor.org/internet-drafts/draft-dolan-urn-isan-01.txt
[27] RFC 3044.
[28] RFC 3187.
[29] RFC 3622.
[30] RFC 3613.
[31] RFC 3614.
[32] RFC 3188.

NEWSML[33] This namespace allows identification of NewsML NewsItems using URNs. NewsML is an XML-based multimedia news resource format developed by the International Press Telecommunications Council (IPTC).

OASIS[34] Similar to the 'IETF' namespace, this allows the persistent identification of publications by the Organisation for the Advancement of Structured Information Standards (OASIS) through the use of URNs.

OID[35] This allows Object Identifiers (OIDs) as specified in the Abstract Syntax Notation One (ASN.1) specification to be expressed as URNs. More details on the ASN.1 concepts are beyond the scope of this report.

PIN[36] This namespace has been engineered by Network Solutions, Inc., for naming people and organisations.

publicid[37] The XML standard defines two identifiers for external entities: the system identifier, which was defined as being a URI, and the public identifier for which no syntax was defined. This namespace allows the integration of those public identifier character strings into the URN concept.

swift[38] This namespace has been reserved for SWIFT, one of the principal standardization bodies for financial messages and services. The namespace allows SWIFT to persistently name the standards and items used for SWIFT messages.

tva[39] Namespace reserved for documents and specifications of the TV-Anytime forum, an international association of organisations which develops specifications for audio-visual and other services.

UCI[40] This URN namespace has been reserved for Uniform Content Identifiers, a new persistent identifier scheme being developed at the South Korean National Computerization Agency. At the time of writing this report, it was not possible to obtain further relevant information on the future of this identifier scheme, which is why it is not covered in greater detail in this report.

[33] RFC 3085.

[34] RFC 3121.

[35] RFC 3061.

[36] RFC 3043.

[37] RFC 3151.

[38] RFC 3615.

[39] http://www.rfc-editor.org/internet-drafts/draft-kameyama-tv-anytime-urn-02.txt

[40] http://www.rfc-editor.org/internet-drafts/draft-sangug-uci-urn-02.txt

UUID[41] This namespace allows expression of Universally Unique Identifiers (UUIDs) as URNs. UUIDs are also known as Globally Unique Identifiers (GUIDs) and are being used in distributed and networked software, e.g. the Microsoft Windows operating system.

WEB3D[42] Like other organisational namespaces, e.g. IETF, OASIS and XMLORG, this namespace has also been dedicated to an organisation. It is reserved for the identification of documents by the Web3D Consortium.

XMLORG[43] This namespace has been dedicated to the naming of resources originating from OASIS' XML.org repository.

2.3.2 URN resolution

There are many approaches to the implementation of resolution services for the different URN namespaces. There are a few general concepts which are worth mentioning:

NAPTR DNS records

Probably the most general concept is provided by the 'NAPTR' mechanism.[44] The Name Authority Pointer (NAPTR) is a standardized type of record in the Domain Name System (DNS). This entry points at resolution services in the Internet in a distributed manner. It was designed to work for all kinds of NIDs and to allow for further delegation of resolution services. Each level in a resolving path is expressed by the delimiting colon (':') in the URN. Unlike the resolving of domain names, the segments of a URN are looked up from the left to the right.

The NAPTR DNS record gives information on the type of the resolution service (e.g. HTTP-based, resolution of one URN to many URLs,...) and allows expression of textual replacements and modifications to be done to the URN in order to finally generate a usable address where the resolution service can be found. The whole standard is still considered to be experimental. There are not many implementations of this standard yet.

Trivial HTTP resolution protocol

In order to achieve easier implementation of a URN resolution service, a simple mechanism was defined for the resolution of URNs to URLs, URNs to resources and URLs to corresponding URNs, if any.

[41] RFC 4122.

[42] RFC 3541.

[43] RFC 3120.

[44] RFC 2168 which was further specified in RFC 2915 and "obsoleted" by RFC 3401-3405. Those RFCs specify a much wider application context, the Dynamic Delegation Discovery System (DDDS), which integrates the NAPTR mechanism in RFC 3403.

The Trivial HTTP resolution protocol[45] (THTTP) uses HTTP as the underlying protocol for network access to the resolution service. The protocol specifies the query and answer syntax. It is basically a textual concatenation of the resolution service's URL, the type of resolution query and the identifier that is to be resolved. The answer depends on the query type and may consist of one or many URLs or URNs as well as one or multiple resources or metadata on resources.

The method for discovering the resolution service's URL is not part of the protocol specification.

2.4 Current applications, long-term perspective

The list of NIDs in section 2.3.1 shows the heterogeneous community of URN users. Introducing all the implementations in more detail would be beyond the scope of this report. In the next chapter, one implementation is described in detail. The long-term perspective of the general URN concept is good: it was the first of its kind and has strong supporting institutions.

2.5 Participation

In order to use URNs as persistent identifiers one can choose between two different approaches. One could decide to get a URN NID assigned. The process is standardised and is outlined in a best-current-practice document.[46] Doing so would involve publishing a full definition of that namespace, its lexical conventions and a description of the available resolution services, as well as a statement on the persistence of the identifiers. This can only be recommended to institutions that develop a new, globally unique approach to issuing or resolving identifiers. Alternatively, one could join an initiative that has already obtained a NID and meets the listed requirements. A prominent example is the URN-NBN namespace whose underlying concepts are introduced in the next chapter of this report.

2.6 Summary

The URN is a general concept that creates a common namespace for many different kinds of identifiers.

Important design principles are persistence of the identifiers and the possibility of resolving them.

[45] RFC 2169.

[46] RFC 3406.

3 National Bibliographic Numbers

The National Bibliographic Number (NBN) is an URN Namespace ID (NID) which was developed and registered by the National Library of Finland.

3.1 History

NBNs have been in production use in demonstration systems since summer of 1998; thousands of URNs within this namespace have already been assigned in Finland, Sweden, Norway, Germany, some Baltic states, Switzerland and other countries. The specification was subsequently further refined to satisfy the requirements for a registration proposal for a NBN namespace (January 2001). It was defined more broadly, although still in terms of a national bibliography.

3.2 Functionality

The bibliographic community uses several bibliographic identifiers functioning as names for objects that exist both in print and, increasingly, in electronic formats. Therefore, one of the main aims for the definition of the NBN was to demonstrate that the current URN syntax proposal can accommodate those existing identifiers, perhaps as *assigned NBN-strings* (see below).

NBN is a namespace which is exclusively assigned to national libraries. The global registry for the *URN:NBN* namespace is the Library of Congress.[47] All National Libraries are responsible for sub-namespaces that are expressed by the ISO 3166 country code (2-letter-code). All other non-ISO 2-letter codes are reserved for possible future country codes. All other non-ISO codes that may consist of three or more letters or digits must be registered. This results in the following general syntax[48] of NBN-URNs:

```
URN:NBN:<ISO country code>-<assigned NBN string>
URN:NBN:<ISO country code:sub-namespace>-<assigned NBN string>
URN:NBN:<non-ISO prefix>-<assigned NBN string>
```

Examples:

urn:nbn:de:kobv:11-10063181 (country code is *de* for Germany, *kobv:11* is the sub-namespace, *10063181* is the assigned NBN string),

urn:nbn:hu-3006 (country code is *hu* for Hungary, no sub-namespace, 3006 is the assigned NBN string).

Registration of sub-namespaces must be done by the national libraries of the

[47] Library of Congress: http://www.loc.gov/

[48] RFC 3188.

country that the code is assigned to. This registry must be available via the Web, it should be accessible via the global registry and, if at all possible, it should be accessible via other national registries.

The further design and syntax of the assigned NBN-URN string is under the authority of the national libraries. In addition, the task of resolving the identifiers is assigned to the national libraries.

3.3 Implementation

Several national libraries developed their own NBN-URN-based systems in the context of national and international research projects, and several implementations are already in practical use. An example is the DIVA[49] project at the Uppsala University Library[50] in Sweden, where documents published in the DIVA-Portal have an unique identifier. In cooperation with the Royal Library[51] of Sweden they implemented an URN:NBN-System. As the national library of Sweden, the Royal Library was assigned the URN:NBN:SE namespace. It then assigned a sub-namespace to the DIVA project.

One can access every document registered to the DIVA system from the NBN resolver at the Royal Library, whether it may be located at its originating institution or at the Royal Library archive. In addition to resolving NBN-URNs, the Royal Library's resolver is also able to resolve other identifier schemes like Handles/DOIs and ARKs covered later in this report. In order to assign URN-NBNs, interested institutions can get their own sub-namespace assigned.

A similar example is the EPICUR[52]-Project at the Deutsche Nationalbibliothek.[53] The aim of the project is to enhance the existing URN:NBN:DE system for online theses. Cornerstones of both systems are:

– to generate and to distribute NBNs,
– to provide an NBN-resolver,
– to store standard metadata together with every NBN (document),
– to archive the documents.

At the Deutsche Nationalbibliothek one can get a sub-namespace for an assigned document server. Within this sub-namespace it is possible for an institution to generate and distribute its own NBN-URNs and to archive these documents along with the created identifier. Utilisation of NBN-URNs is a requirement for the German DINI[54]-Zertifikat, a certification for document servers to assure they comply with certain quality standards.

[49] DIVA: http://www.diva-portal.org/about.xsql

[50] Uppsala University Library: http://www.ub.uu.se/eindex.cfm

[51] The Royal Library, Stockholm: http://www.kb.se/ENG/kbstart.htm

[52] EPICUR: http://www.persistent-identifier.de/?link=330

[53] Deutsche Nationalbibliothek (German national library): http://www.ddb.de/

[54] DINI – Deutsche Initiative für Netzwerkinformation: http://www.dini.de/

3.4 Current applications

Because the authority for the national NBN sub-namespaces is delegated to the national libraries and each national library may have its own philosophy, necessity and basic local parameters, there are no applications that can be used for all different national sub-namespaces. In order to see examples of existing applications, one has to take a look at the website of national libraries that already have established NBN systems.[55] There usually will be descriptions of the policy, a facility for resolving the NBN-URNs and often downloadable tools, e.g. to generate NBNs, for the implementation of resolving systems and helpers to store metadata.

3.5 Long-term perspective

The NBN-URN scheme is a well known, established, international standard, so it should be consistent. It can be expected that there will be broader recognition if international projects evolve to assure interoperability among the different (national) implementations.

3.6 Participation

The NBN namespace is exclusively assigned to the national libraries; only they are participants and it is their authority to decide about the group of participants they manage themselves.

Other institutions should check with the local national library in their respective countries for participation policies.

3.7 Summary

NBN is a URN namespace assigned to national libraries.

NBNs are focussed on the naming of resources, both in print and in electronic format.

NBNs are designed to accommodate existing identifier schemes.

Resolving of NBNs is done by the national libraries for their respective namespace and is not well defined when it comes to individual implementation of resolving services.

Some national libraries have established data exchange between each other to facilitate resolving of other national libraries' NBN-URNs. However, there is no central resolver for all possible NBN-URNs.

The NBN namespace has no commercial background, but it is the sovereign territory of national libraries.

[55] Also see the presentations of some systems at the Erpanet Seminar on Persistent Identifiers: http://www.erpanet.org/events/2004/cork/

4 Handles

One of the proposed mechanisms for implementing persistent identifiers is the Handle System.

4.1 History

The Handle System[56] was developed by the Corporation for National Research Initiatives (CNRI)[57] for the Computer Science Technical Reports (CSTR) project.[58] Its development was funded by the Defense Advanced Research Projects Agency (DARPA).[59] One of the project's aims was to develop a framework for digital libraries.[60] The Handle System was designed to be the naming component of a system for accessing digital objects and was designed with uniqueness as its primary intention.[61] The first implementation of the Handle System was made available in the autumn of 1994.[62]

4.2 Functionality

The Handle System was designed to provide naming services. The Handle System is composed of different elements: a set of protocols, a name space and a reference software implementation. Here, when we talk about the Handle System, we are in fact discussing all of these elements.

Handle commonly refers to an identifier created by the Handle System that complies with the Handle System namespace definition.

The protocol suite defines a protocol suitable for resolving the authority for such a Handle and exchanging authentication information for different tasks regarding the management of the data assigned to a Handle. It defines a mechanism to allow locating the authority that is in charge of the information pertaining to the named item while being independent of the DNS by not making use of its services in the Handle Protocol.

The main design goals for the Handle System are summarised here:[63]

- Uniqueness of the Handles
- Persistence: this means that there is a policy that an operational connection between the Handle and the identified entity is maintained within the Handle

[56] http://handle.net/

[57] http://www.cnri.reston.va.us/

[58] http://www.cnri.reston.va.us/cstr.html

[59] http://www.darpa.mil/

[60] Lannom, Laurence: 'Handle System Overview'. RFC 3650, Ch. 8.

[61] Kahn/Wilenski: A Framework for Distributed Digital Object Services.

[62] See DOI Handbook, A2.1.

[63] Lannom, Laurence: 'Handle System Overview'. RFC 3650.

System. The design recognizes that persistence itself is a function of administrative care.

- Multiple instances: Handles have the ability to refer to multiple instances of the named resource.
- Extensible namespace: for Naming Authorities, it is possible to introduce their own (possibly pre-existing) namespace into the Handle System. There is no full integration in the sense that such a namespace can be used as is, but integration through a sub-namespace is possible.
- International support: Handle is based on Unicode 2.0, which the current protocol encodes as UTF-8. This allows expression of virtually all known printable characters.
- Distributed service model: one global service can delegate Handle queries to the local service of a Naming Authority but can also answer the query itself. The local service can also dispatch the query internally to allow mirroring and clustering to facilitate high availability.
- Secured name service: operations on the Handle databases have to be authorised. The authorization mechanism is fine-grained for the different operations on the Handle database.
- Distributed administration service: the above mentioned authorization mechanism applies individually for each Handle so that administration can be distributed individually.
- Efficient resolution service: this is accomplished by separating the administration protocol from the resolving protocol to allow easier distribution of needed computational resources.

4.3 Implementation

4.3.1 The Handle Namespace

The namespace (naming scheme) of the Handle System is defined as

```
<Handle> ::=        <Handle Naming Authority> "/"
                    <Handle Local Name>
```

Example (fictional): *145.76/jan2005-rk324942199*

The Handle Naming Authority (NA) is a suffix assigned by the Global Handle Service (e.g 145.76 in the example above). The Handle Local Name can be composed according to the NA's policies and thus be used to carry local sub-namespaces (e.g. j*an2005-rk324942199* above). There is no limitation to the Handle Local Name except that it must be expressed through printable characters from Unicode's UCS-2 character set. The NA itself is currently decimal and is assigned in a sequential fashion. For historical reasons, there are also some alphanumerical NAs.

The dot ('.') character is used to express a path in a hierarchy of NAs in the NA string. The path is to be read from the left to the right (the reversed order compared to the DNS domain names). This hierarchy is not supposed to have technical implications: technically, a NA '5.6.7' could be independent of an NA '5.6'. The hierarchy is mainly implemented at the administrative level. The protocol just ensures that initial creation of new nodes is only possible with authorization by the higher node in the hierarchy. There are no further dependencies between an upper level NA and the subordinated NA.

4.3.2 The Handle architecture and protocol

The Handle System is designed to 'resolve' the Handles independently of the Domain Name System. So instead of being based on the DNS root servers, it has its own root server, the Global Handle System hosted by the CNRI. This system knows about all Naming Authorities and delegates queries to these. Each Naming Authority can establish its own infrastructure. The Handle System allows mirroring and delegation of resolving services to other Handle servers.

Further, the Handle System does not rely on URLs. It can make use of them and there is a HTTP proxy server that can redirect URLs according to stored URLs for Handles. Equally, arbitrary data can be stored in the Handle database and is categorised by either its index (integer value) or type (hierarchically constructed string)[64] and as such can indicate a certain scheme (e.g. URL). So the Handle System would be ready to store locators for access mechanisms other than HTTP as well as other additional metadata.

4.4 Current applications

The Handle System is currently being used by a number of different institutions and projects:

- The Defense Virtual Library[65] is a project of the Defense Technical Information Center,[66] the DARPA and CNRI to develop a pilot digital library implementation.
- The DOI (introduced in the next section) design uses it as the naming component.
- DSpace,[67] an open source repository software, uses it to name the document containers and to provide access to those document containers.

4.5 Long-term perspective

As the Handle System is currently being used for a number of other projects (also see the next chapter, on 'DOI'), which themselves are establishing a permanent identification mechanism, it is not likely that the Handle System will cease to exist in the near future.

[64] For an in depth look see the Handle Protocol Specification, RFC 3652, 3.2.1.

[65] http://dvl.dtic.mil/

[66] http://www.dtic.mil/

[67] http://www.dspace.org/

4.6 Participation

To take part in the Handle System, one has to register and establish a Naming Authority. While the software is made freely available by the CNRI, the registration of a new Naming Authority requires contact with the CNRI and signing a licence. But because the software is freely available, it is possible to do local tests with Handles and a temporary arbitrary Naming Authority first, before registering a ‘real’ Naming Authority.

The CNRI recently published policy documents for participating in the Handle System. As of June 2006, there is now a registration fee for a Handle Naming Authority number of $50 and an annual service fee of $50.[68] Programming libraries for developing clients that make use of the Handle System are freely available, as well. There is a standardised workflow for implementers of the open source DSpace document repository system to support registering a new Naming Authority.[69]

4.7 Summary

Handle is a concept for a DNS-independent naming and resolving mechanism. The Handle Protocol in some ways resembles the DNS. It can be used to resolve the names, ‘Handles’, to URLs but also allows them to be resolved to other identifier formats and arbitrary data.

Technically, Handle defines a two-level hierarchy that can and is being extended to more levels by utilizing administrative naming policies. It gives the freedom to use any printable character from the Unicode UCS-2 character set for the names.

Handle allows integration of other naming conventions at the second level of its hierarchy.

The resolving infrastructure exists. It works and scales well.

The software is freely available and can be tested extensively before registration with CNRI and production use.

[68] http://hdl.handle.net/4263537/5029

[69] See the Handle System homepage: http://www.handle.net/

5 Digital Object Identifiers

The Digital Object Identifier (DOI) is managed and controlled by the International DOI Foundation (IDF). IDF membership is 'open to all organisations with an interest in electronic publishing'.[70] The DOI provides administrative schemes and workflows for the management and persistent identification of digital objects. From a technical perspective, the DOI builds on the Handle System with a set of additional schemes and interoperability standards and provides a root service for interactions conforming to the Handle standard as well as its own additional standards both from a technical and administrative point of view.

5.1 History

The origin of the DOI lies in a project of the Association of American Publishers[71] in 1996. In partnership with the CNRI, DOI was designed 'to link customers with publishers, facilitate electronic commerce, and enable copyright management systems'.[72] It was decided early on that the Handle System should be the underlying 'communication' technology for managing and resolving DOIs.

After a brief time of piloting beginning in July 1997 with a closed set of other interested publishers, the DOI was introduced to the public at the Frankfurt Book Fair in October 1998. It was opened at that point to other publishers with an interest in participation. The International DOI Foundation was established and interested parties were invited to acquire membership.

5.2 Functionality

The DOI's functionality is difficult to compare against the other identification systems introduced in this report. It goes far beyond the technical level which mainly consists of the underlying Handle System. While the Handle System gives the opportunity to introduce further specifications within the boundaries of a Naming Authority, the DOI actually does just that. It has created a high-level hierarchical Application Profile on top of the Handle architecture to ensure interoperability among all DOI references as well as applications and services built on top of the DOI specifications. The DOI was created to be[73]

- Persistent: in the DOI context this is defined as the DOI having a (fixed) relation to the named resource, whereas its location or stewardship is only expressed in the metadata that belong to the DOI.[74]

[70] DOI Handbook, 7.13.

[71] http://www.publishers.org/

[72] DOI homepage as of 15 December 1997, accessible via the Wayback machine: http://web.archive.org/

[73] DOI Handbook, 1.4.

[74] DOI Handbook, 1.4.1.

- Actionable: that means that metadata can be accessed by the user of a DOI (that can consist of location information but is not restricted to this kind of information).[75]
- Interoperable: the DOI is meant to integrate other ‘legacy’ identifier schemes, to be technically independent of current access mechanisms (HTTP, URL) through use of the Handle System and to integrate efforts of other initiatives at the metadata level.[76]
- An identifier to be suitable for even wider scopes of object identification in the digital world: the usage of a DOI is not restricted to digital objects. The aim is to provide an identification mechanism for all trading transactions concerning rights management.[77]

5.3 Implementation

5.3.1 Usage of the Handle System

DOI introduces both a technical and administrative layer on top of the facilities of the Handle System.

The IDF was assigned a Handle Naming Authority, ‘10’. This is the general prefix used for further subdivision of this namespace and creation of new Sub Naming Authorities. The assignment of such a new Naming Authority is controlled by the IDF and technically enforced by the mechanisms of the Handle System. Because the Handle System allows storage of arbitrary data this facility is used by the DOI framework to store metadata about the object referenced by the DOI.

5.3.2 Data management

The DOI model delegates the assignment of DOIs to ‘Registration Agencies’ (RAs). Those must fulfil certain quality standards and are free to choose any business model. They are in charge of the data stored for the Handles and, in some cases, the Handle System infrastructure for a Handle Naming Authority. RAs have to be members of the IDF.

5.3.3 Metadata policies

A DOI Data Model was created to achieve the aim of promoting interoperability through the use of common standards and ensuring a certain level of quality for the data management of DOIs. There is a ‘DOI Kernel Declaration’ that gives basic information which must be implemented by all RAs for all Application Profiles and Services (see below for these terms) – possibly by mapping other metadata schemes.

[75] DOI Handbook, 1.4.2.

[76] DOI Handbook, 1.4.3.

[77] DOI Handbook, 1.4.4.

Occurrence/kernel metadata element	Description
1 DOI	DOI/Handle assigned to the identified resource.
1 structuralType	One of physical, digital, performance, abstraction
1-3mode	Intended modes of perception: hear, view, feel, etc.
1+ resourceType	Categorization of the described resource: audio file, journal article, etc.
0+ resourceIdentifier	References to another identifiers that have to be unique within their domains
0+ resourceName	Names of the document without commitment of their uniqueness
1+ principalAgent	Holds information about stewardship, publication, etc. of the resource. The specific detail of this information is at the discretion of the RA issuing the Application Profile.
Further information about the given metadata declaration	
1 registrationAgency	Reference to the Registration Agency that issued the Metadata Declaration
1 issueDate	Date of issue of the Declaration
1 issueNumber	Version of the Declaration, counted sequentially from 1.

The DOI utilizes the indecs Data Dictionary (iDD) and provides the framework for applying it. The iDD is built around the <indecs> (interoperability of data in e-commerce) framework[78] and was designed with interoperability with other metadata schemes in mind. It allows mapping of existing metadata schemes into its common dictionary for metadata descriptions to achieve easier comparison between metadata schemes and eases mapping between them.[79]

5.3.4 Applications and services

Still under development but nearly completed, are the DOI specifications for implementing services on top of the basic resolving mechanisms. Through use of the DOI metadata framework a specification of applications and services is possible. The Application Profiles and services must be registered with the IDF to be approved[80] and are then assigned a DOI for further reference.

Subsequently, the Application Profile can be assigned to a group of ‘Content’ DOIs by referencing the Application Profile in the Content DOIs metadata. In this way, the Application Profiles put a mark on the Content DOIs and can group and categorize them. A Service DOI can be registered within the data assigned to an Application Profile DOI to express the applicability of that service for the DOIs that are themselves grouped under that Application Profile.

That way, DOIs can express their compliance with certain applications and thus services.

[78] <indecs> project’s homepage:http://www.indecs.org/, more information on the framework is linked from http://www.indecs.org/project.htm/

[79] DOI Factsheet ‘DOI and Data Dictionaries’.

[80] DOI Handbook, 5.5.1.

5.4 Current applications

The DOI is heavily used in the commercial scientific publishing area and by some publicly funded projects as well. DOIs are even used to provide entry points to scientific data that cannot be categorised as ‘documents’, such as scientific measurement data or similar information.[81]

There are already tens of millions of registered DOIs and the resolving mechanism is being used a few million times each month.

Noteworthy is the CrossRef service[82] that aims at providing a single entry point for the linking of citations for all scientific literature. It utilizes the DOI concept and also makes use of the OpenURL concept.[83]

5.5 Participation

5.5.1 Participation strategies

Participation is possible at different levels:

- For registration of Content DOIs, one has to find a Registration Agency that fits the purpose.[84] This Registration Agency manages the data flow to the DOI (and thus to the Handle) System.
- An organisation that wishes to participate in the work of the ongoing standardization of metadata, applications and services within the DOI framework, must become a member of the IDF. As such it can take part in the IDF's working groups.
- As a member of the IDF, one can establish a new Registration Agency in cooperation with the IDF.

Participation at a higher level than just registering some DOIs at an intermediate RA therefore requires interaction with the IDF and taking part in its organisational structures.

5.5.2 Participation costs

The IDF has fixed policies for the costs for different levels of participation:

- For registration of DOIs only, the costs are according to the chosen Registration Agency's own policy.
- The Membership fee for the IDF is meant to cover the actual costs of a continued (‘persistent’) organisation model so the fees have varied and may continue to do so in the coming years.[85] There are various forms of membership:[86]

[81] See e.g. http://www.std-doi.de/

[82] http://crossref.org/.

[83] Introduced in Section 8 ‘OpenURL’.

[84] A list of those RAs is always accessible at the DOI homepage: http://www.doi.org/registration_agencies.html

[85] DOI Handbook, 7.2.1.

[86] DOI Handbook, 7.13.2f.

- General Membership for principal participation within the development of the DOI system for organisations with relations to electronic publishing. The annual fee is $35,000[87] but can be reduced at the sole discretion of the IDF's Board, under certain circumstances.[88]
- Charter Membership for organisations with a primary interest in the creation or production of intellectual property. The annual fee is $70,000, which can also be reduced under certain circumstances.[89]
- Registration Agency Membership is only available after signing a Letter of Intent with the IDF. The annual fee is $35,000,[90] with an additional fee of $0.04 per DOI issued.[91]
- Affiliate Membership offers no voting rights or other full membership rights but allows an institution to have a nominated representative in a DOI Working Group. The current annual fee is $5,000 per working group.

5.6 Summary

DOI is an administrative framework for assuring common standards and practices.

DOI utilizes the Handle System as naming and resolving component.

DOI introduces new metadata concepts for items, applications and services and provides a framework for interaction between them.

[87] All mentioned Dollar prices in this chapter are given in US Dollar.

[88] DOI Handbook, 7.13.3.

[89] Ibid.

[90] Ibid.

[91] Erpanet Cork Final Report, DOI Overview, Pg. 15.

6 Archival Resource Keys

A relatively young, but promising new approach towards the implementation of persistent identifiers is the Archival Resource Key (ARK).[92]

6.1 History

The Archival Resource Key (ARK) is a concept developed by John Kunze and R.P.C. Rogers at the conclusion of a study of persistent identifier systems for the US National Library of Medicine (NLM). It was issued as an Internet draft in February 2001, and the current draft was issued in August 2006. The ARK scheme is maintained at the California Digital Library (CDL)[93] within the University of California.

The ARK not only includes a concept for persistent identification but also focuses on a complete protocol and software suite to provide a starting point for a full framework for persistent identification and resolving of the identifiers to different kinds of information resources.

6.2 Functionality

There are three postulates for the functionality of ARKs[94] that are based on the idea that persistence is purely a matter of service rather than the naming syntax:

- An identifier should allow users to access a ‘promise of stewardship’ for the identified object.
- An identifier should allow access to a description of the identified object (metadata).
- An identifier should – if at all possible – link to the identified object itself.

The ARK scheme does not assert that the identifiers are persistent since that depends solely on the service(s) behind them. Thus the best an ARK can do is to create linkages to the services from which the holder of an ARK can make an informed judgement. The ARK further allows for multiple dimensions of a persistence commitment to be expressed.[95]

6.3 Implementation

As mentioned before, the ARK concept is a complete and open framework for persistent identification that consists of an administrative model for the whole

[92] http://www.cdlib.org/inside/diglib/ark/

[93] http://www.cdlib.org/

[94] ARK Identifier Scheme, Internet Draft, 1.1.

[95] Byrnes, Margaret: Defining NLM’s Commitment to the Permanence of Electronic Information. 2000. This report explains the different dimensions of persistence defined for the NLM.

system, naming conventions for expressing hierarchy and versions, and network protocols. Elements making up an ARK are discussed in more detail below.

6.3.1 Administration of ARKs

In order to assign ARKs, one must either become a Name Assigning Authority (NAA) or be authorised by an NAA to assign names (as a sub-authority). This NAA is assigned a decimal number, the Name Assigning Authority Number (NAAN). The NAA is in charge of assigning or delegating the assignment of names to objects.

Additionally, there is the resolution part. Each NAAN has one or more associated Name Mapping Authority Hostports (NMAHs). This is the location of a current provider of services (e.g. hosting, access, forwarding) for the identified objects. So the NMAH makes the ARK actionable without disturbing its fundamental identity which is defined to follow the NMAH. In a sense this is similar to how HTTP proxies work for the other identification schemes introduced here that are not URLs themselves (Handles/DOIs, URNs). However, as this is fully integrated into the ARK scheme, there is a defined way for ARKs to cut off the NMAH part to allow for determination whether two ARKs identify the same object. The other mentioned systems do not have such a standardised way to strip off ‘proxy information’.

The NMA is in charge of offering resolution and information services for one or more NAAs. The NMAH is a technical location for the service and may deal with objects assigned by more than one NAA. The current list of NAANs and the corresponding NMAHs are published in a publicly known location. This is the fixed root to allow the resolving of ARKs. It is constructed as a simple lookup table.

6.3.2 The ARK Namespace

The ARK Namespace is defined as follows[96]:

```
[“http://” <NMAH> “/”]“ark:/” <NAAN> “/” <Name> [ <Qualifier> ]
```

where the specification of the Name Mapping Authority Hostport (NMAH), the protocol ‘http://’ and the specification of a Qualifier part are all optional. So the ARK itself basically consists of the string ‘ark:/’, the Name Assigning Authority Number (NAAN) and the Name chosen by the NAAN.

To identify ARKs that are encoded in other identifiers, e.g. URIs (URLs, URNs) or similar, they are prefixed with the ‘ark:/’ label string. It makes them more easily recognizable when encoded in those other schemes. That way, the occurrence of that label in such ‘foreign’ identifiers indicates a certain probability that a valid ARK can be extracted from the other identifier or locator. As in the case of any identifier scheme incorporated into another, there is no absolute certainty that an extracted string is a valid ARK (or URN or Handle) unless the

[96] ARK Identifier Scheme, Internet Draft, 2.

proxy sites are known to belong to a scheme specific service provider. In any case, ARKs, like Handles or URNs, await the day when client technology (e.g. Web browsers) will implement their respective resolving mechanisms so that the bare identifiers will be universally resolvable without being incorporated into other identifiers or locators.[97]

The NAAN registry is currently maintained by the CDL and mirrored at the NLM. Currently, NAANs have five digits. When this namespace for NAANs is filled up, it will then start up again with numbers of 9 digits each.[98]

The Name part consists of visible ASCII characters. Importantly, the ARK scheme discourages the use of semantics inside term identifiers. There are some reserved characters:

‘%’ ‘-’ ‘.’ ‘/’

These have special meanings: ‘%’ is the general escape char for encoding bytes. There is no explicit character set, it is possible to encode ‘characters’ on the byte level. This way it is possible to include other identification schemes.

Hyphens (‘-’) are to be ignored. They might be integrated in ARKs to improve readability. This way, the ARK scheme is aware that a publishing process of a document containing ARKs may introduce hyphens.

Software using ARKs is expected simply to ignore hyphens. The normalization done by the software is described in the Internet draft.[99] The slash (‘/’) and dot (‘.’) characters are reserved for the ARK’s way of expressing hierarchies and variants.

6.3.3 Trees, paths and nodes: hierarchy and variants (optional)

The ARK Identifier Scheme specifies ways to express hierarchies in the Identifier. They are described by a path of slash (‘/’) separated nodes. The suggested way to use this facility is to allow the resolution of upper nodes in order to acquire information about the higher hierarchy levels. An example of this would be to assign identifiers to some serial’s articles like

```
ark:/<NAAN>/2341-xth-3242/rbgs
ark:/<NAAN>/2341-xth-3242/oiis
```

and to an information record for the serial as well:

```
ark:/<NAAN>/2341-xth-3242
```

This is only a two-level hierarchy where a hierarchy of arbitrary depth is possible. The mode of utilizing the possibility of expressing such hierarchies is up to the NAA. It can also decide to not disclose hierarchy information at all.

Furthermore, the ARK Internet draft standardizes a way of expressing variants in ARKs. It is up to the NAA or NMA to determine what difference level be-

[97] ARK Identifier Scheme, Internet Draft, 2.2.

[98] ARK Identifier Scheme, Internet Draft, 2.3.

[99] ARK Identifier Scheme, Internet Draft, 2.7.

tween two manifestations or objects is necessary to establish that one is a variant of the other instead of both being identical. The draft defines, however, the way to express the fact that a variant is referenced by an ARK. If the ARK contains a dot ('.') after its name part, the following part is interpreted to identify a variant. There can be multiple levels of variance expressed by separating them by more dots. If the dot character is used in an ARK, this implies that at least one variant is identified by the ARK without this dot-separated postfix.

Examples of the ability to express object variants:

```
ark:/<NAAN>/sld-213-zt.tiff.g4
ark:/<NAAN>/sld-213-zt.tiff.lzw
```

The existence of those two ARKs implies the existence of these other two ARKs:

```
ark:/<NAAN>/sld-213-zt.tiff (must resolve)
ark:/<NAAN>/sld-213-zt (must resolve as well)
```

By avoiding the use of dots, assigners choose to not disclose variant information at all.

6.3.4 Locating the resolver

To prepare to make an ARK actionable, it needs to include an indication of where to ask for those services. A mechanism is specified for looking up the responsible Name Mapping Authority Hostport (NMAH).

First of all, a working NMAH could be already encoded in the prefix part of the ARK. If this NMAH does not work (any more), maybe because responsibilities have changed, a mechanism to locate the new NMAH is needed. Currently, the preferred way is to look up the responsible NMAH in the global list.[100]

Another proposal introduced in the Internet draft is to use the DNS/NAPTR mechanism as a service discovery facility. This mechanism was introduced to allow location of services for certain URN schemes by utilizing the capabilities of the Domain Name System (DNS).[101] The NAPTR records allow specification of such an NMAH, but it is up to the client to use that information. Common clients for the WWW (Web browsers) currently support NAPTR – if at all – only via plugins. But the NAPTR mechanism would allow the implementation of a service discovery facility for an arbitrary number of identification systems, not just ARKs. A reference implementation for client-side usage is introduced in the ARK Internet draft. Compared to the algorithm proposed for URN resolution, the ARK implementation is a streamlined, simplified approach to use NAPTR.

6.3.5 Making ARKs actionable: THUMP

The Internet draft also specifies a simple convention for using HTTP to deliver

[100] See below, 6.4, for the current list.

[101] See RFC 2168 for additional information.

the three ARK services. The convention is itself a simple protocol that can be implemented with a few server rewrite rules and an email-header based output. It is called the Tiny HTTP URL Mapping Protocol (THUMP). It is based on issuing a HTTP GET plus a URL sent to a THUMP-enabled Web server (essentially a Web server calling a CGI script) and provides different services:

- When called with just the ARK appended, it redirects to the identified object or to a sensible substitute (e.g. a table of contents, a description of how to access physical resources etc.).
- When called with a question mark ('?') appended to the ARK, it sends metadata about the identified item.
- When called with a double question mark ('??') appended to the ARK, it sends a formalised minimal permanence statement.

The THUMP server sends its results in 'text/plain' format with a line-based syntax. It uses the Electronic Resource Citation (ERC) syntax[102] to express the metadata and the permanence policy. The ERC syntax is a simple, yet powerful metadata format that shares certain structures with the Dublin Core metadata set and thus is in large parts mappable from and to Dublin Core metadata. In addition, it also allows specification of metadata with finer granularity. There are four basic questions answered by ERC metadata

ERC element	Description
who	Responsible person or party
what	A name or identifier that should be human readable
when	Some point in time relevant for the described object
where	A location or identifier that allows for location of the object

These questions can answered for different contexts, called 'segments':

ERC segment	Described context
erc	ERC segment describing the expression of the object.
erc-about	Segment describing what the object's content is about.
erc-support	Segment describing the support commitment made to the object.
erc-from	Segment describing the provenance of the ERC metadata

6.4 Participation

The ARK Identifier scheme is more lightweight and much less 'packaged' than to the DOI and Handle systems regarding the model for participation. This may be due to its roots being in the area of publicly funded academic research. The ARK user community may be deduced from the current list of Name Assigning Authorities.[103]

[102] Further Information on ERC and how it compares to Dublin Core: J. Kunze: A metadata kernel for electronic permanence.

[103] Taken from http://www.cdlib.org/inside/diglib/ark/natab

NAAN	Name Assigning Authority
12025	National Library of Medicine
12026	Library of Congress
12027	National Agriculture Library
12148	Bibliothèque nationale de France/ National Library of France
13030	California Digital Library
13038	World Intellectual Property Organization
13960	Internet Archive
15230	Rutgers University Libraries
20775	University of California San Diego
25593	Emory University
27927	Ithaka Electronic-Archiving Initiative
28722	University of California Berkeley
29114	University of California San Francisco
64269	Digital Curation Centre
62624	New York University Libraries
67531	University of North Texas Libraries
78428	University of Washington
80444	Northwest Digital Archives
88435	Princeton University
89901	Archives of Region of Västra Götaland and City of Gothenburg, Sweden

Among these institutions there are several known for their knowledge in the field of management of digital resources so this table gives a good overview regarding the backers of this relatively new concept.

Several of the developments described are still under development or in a pilot phase. It should not be difficult to join the ongoing discussion.

If one wants to issue ARKs, the institution must be assigned a Name Assigning Authority Number (NAAN). Examples for such institutions according to the Internet draft are national libraries, national archives and publishers. In order to have one assigned, contact the project's email address.[104]

6.5 Summary

ARKs introduce a concept combining the features that a persistent identifier should have and building a technical and administrative framework on that concept.

The ARK is focussed on resolving and delivering metadata.

The concept of the ARK has a two-level hierarchical namespace. Below the root, there are the Name Assigning Authorities that have their own namespace to assign Names.

The ARK concept is designed both to allow integration of other identifier schemes as well as being integrated into other identifier schemes itself.

The ARK concept has no commercially motivated background.

The technical requirements are fairly low (DNS, Web server and a Web browser on client side). Thus future maintenance will probably be easier than it would be for complex specialised software.

7 Persistent URLs

The Persistent URL (PURL) is one of the first implementations of Persistent Identifiers based on the URN specification.

7.1 History

PURLs were developed by OCLC[105] as a naming and resolution service for Internet resources in order to aid creating acceptance for the URN technology. In 1996 they were implemented for the Internet Cataloguing Project, a U.S. Department of Education-funded project to advance cataloguing practices for Internet resources.[106]

PURLs are designed to specify a resource in printed documents, Web pages or cataloguing systems. They are also locators by pointing to an intermediate resolution service.

7.2 Functionality

A PURL is a URL – but instead of directly pointing to an Internet resource, there is an intermediate resolution service. This resolution service associates the PURL with the actual URL pointing to the identified resource and returns that URL to the client. In order to do so, the PURL resolution service uses the standardised HyperText Transfer Protocol (HTTP) redirect. Further access to the resource itself is accomplished by the client (Browser) automatically accessing the server providing the Internet resource.

The redirection used by OCLC's PURL is a standard HTTP feature; that way the PURL Resolution Service keeps the resolution server load light. After resolving the PURL to the URL, all further network traffic happens within the communication between the client (the user's browser) and the server providing the Internet resource.

The main design goals for OCLC's PURL are:

- Separation of locators from names of Internet resources.
- Use of standard (already implemented) services and protocols.
- Persistence, which is implemented in the OCLC software to maintain PURLs as follows: one can change what a PURL resolves to, but one cannot change or delete the PURL itself. This means that PURLs persist eternally. When the associated URL of a PURL becomes outdated, the resolution may fail, but the PURL and its full history will be available as long the PURL Service itself is

[105] OCLC – Online Computer Library Center, Inc.: http://purl.oclc.org

[106] Keith Shafer, Stuart Weibel, Erik Jul, Jon Fausey: *Introduction to Persistent Uniform Resource Locators*.

maintained. This serves to underline that persistence is a function of organisation and administration, not of technology.

7.3 Implementation

PURLs are URLs consisting of three parts: protocol, resolver address and name for the resource.

```
PURL ::= <protocol><resolver address><name>
```

Example: *http://purl.oclc.org/NET/DIGIZEIT.PPN*

Note that PURLs use well-established services, in this case:

– The HyperText Transfer Protocol,
– Domain Name Services (DNS) to get the IP-address assigned to the resolver 'purl.oclc.org'.
– The name 'NET/DIGIZEIT.PPN' is user-assigned and is resolved by the associated PURL resolver that is identified by the resolver address.

Usually the path part of an URL is case sensitive, but this is not the case with the name part of PURLs. For example, *http://purl.oclc.org/NET/DIGIZEIT.PPN* and *http://purl.oclc.org/NET/DigiZeit.ppn* are considered as being equal and are consequently the same PURL.

The PURL server software developed by OCLC is freely available at the PURL Website.[107] It contains all that is needed to maintain PURLs in a distributed environment, including user and group management for maintainers and tools to create and to maintain PURLs or partial redirects.

The example is a PURL for the PPN[108]-Resolver from DigiZeitschriften.[109] Try *http://purl.oclc.org/net/DigiZeit.ppn?PPN=PPN345571991_1856.*

7.3.1 The PURL namespace

Each PURL-Resolver is responsible for resolving PURLs of its own namespace, so worldwide uniqueness of PURLs depends on the resolver address. The separate namespaces are organised as a tree of domains. The first part of a name is the top-level domain, all other parts of the path are subdomains. So the PURL <*http://purl.fake.com/A/B/C/document*> has three domains encoded, *A* as top-level domain, *B* and *C* as subdomains. The *document* resides in domain C. This scheme allows partial redirection: if the PURL

<*http://purl.fake.com/bar/*>

is associated with the URL

<*http://your.web.site/*>,

[107] http://purl.oclc.org

[108] PPN – Pica Production Number: http://www.oclcpica.org

[109] DigiZeitschriften: http://www.digizeitschriften.de

then a PURL like

<http://purl.fake.com/bar/some/stuff.html>

would be automatically associated with the URL

<http://your.web.site/some/stuff.html>

7.4 Current applications

The OCLC PURL server is still up and running at http://purl.oclc.org/ and everyone is invited to establish their own sub-domain on this server and maintain one's own PURLs. The OCLC PURL software was downloaded over 600 times by different domains, so there may be many installations around the world. For more information about current usage, please visit the PURL Website.

7.5 Long-term perspective

PURLs are a direct result of OCLC's work in the Uniform Resource Name (URN) standards and library cataloguing communities. The assignment of PURLs is an intermediate step towards a time when URNs are an integral part of the Internet information architecture. The eventual syntax of URNs is clear enough at this time to afford confidence that the syntax of PURLs can be inexpensively and mechanically translated to the eventual URN form.[110] For instance, the PURL

http://purl.fake.com/foo/bar

could be written as follows using the URN syntax:

URN:[PURL-NID]:/com/fake/purl/foo/bar

So if URN has a long-term perspective, so has PURL.

7.6 Participation

The PURL Specification and the PURL Server Software are freely distributed for establishing a new PURL server and so automatically creating a new namespace is easy to implement.

Additionally, it is possible to reuse the OCLC PURL Server and to establish sub-domains and to assign PURLs that reside in these sub-domains.

[110] Keith Shafer, Stuart Weibel, Erik Jul, Jon Fausey: Introduction to Persistent Uniform Resource Locators.

7.7 Summary

A PURL focuses on the location of an electronic resource in a persistent fashion.

If a PURL service is properly maintained and administered, it offers persistent identification facilities.

There are PURL implementations that offer easy participation and cooperative use of a central service.

PURLs offer relocation services and access to the history of known locations of the identified resource.

8 OpenURLs

OpenURL differs from a persistent identifier system in that it has a different focus. It is introduced here because it is often discussed along with identification systems.

The OpenURL is basically a metadata transport protocol. Using OpenURLs, it is possible to establish value-added services based on the information encoded in the OpenURL. Examples of such information items are the context of the user (institution, access policies, authentification data) and metadata describing the linked object.

8.1 History

The OpenURL concepts were developed by Herbert Van de Sompel at the University of Ghent (now at Los Alamos National Laboratory) in 1999 and are now a NISO[111] standard (Z39.88-2004). It is a protocol for interoperability between an information resource and a service component, referred to as a link server, which offers localised services.

The initial development of the OpenURL standard – published as version 0.1– was targeted at the electronic delivery of scholarly journal articles. In version 1.0, the framework was generalised to enable communities beyond the original audience of scholarly information users to adopt extended linking services and to lower the entry barrier for new implementers.

8.2 Functionality

The underlying concept of the OpenURL standard is that links should lead a user to appropriate resources. The OpenURL standard enables a user to obtain immediate access to the 'most appropriate' copy of an object through the implementation of extended linking services. This selection occurs without interaction by the user; it is made possible by the transport of metadata together with the OpenURL link from the source citation to a 'resolver' (the link server), which stores the preference information and the links to the appropriate material.

8.3 Implementation

The OpenURL Framework is a standard, not an implementation of the standard. Several commercial and non-commercial link servers that are based on OpenURL exist. The main goals of those services are:

- access management based on user context, perhaps IP address, cookies or user/ password combinations saved earlier;

[111] NISO: National Information Standards Organization: http://www.niso.org/

– presentation of additional metadata of the resource;
– value-added services like related links to search engines, library catalogues, order services or other repositories based on given metadata,
– and obviously the link to the online resource itself.

The ‘OpenURL demonstrator’[112] at the UKOLN[113] website shows the achievement of several link resolvers based on OpenURL.

8.3.1 The OpenURL syntax[114]

The OpenURL syntax is described here as a HTTP GET request with the following syntax:

```
OpenURL ::= <BASE-URL>”?”<QUERY>
QUERY ::= <DESCRIPTION>(“&&”<DESCRIPTION>)
```

Base-URL is the locator for the service component that accepts an OpenURL as input. *Query* describes the origin of the transported metadata object as well as the metadata-object itself. If multiple objects are transported via the OpenURL, their *description* must be delimited by two ampersands.

```
DESCRIPTION ::=
   (<ORIGIN-DESCRIPTION>”&”)?<OBJECT-DESCRIPTION> |
    <OBJECT-DESCRIPTION>(“&”ORIGIN-DESCRIPTION>)?
```

Object-description contains information about the metadata-object transported in the OpenURL. *Origin-description* contains information about the information system where the transported metadata object originates. It describes the system that inserts the OpenURL. The OpenURL must transport at least one object. As such the OpenURL must contain at least one *object-description*. The order in which *object-description* and *origin-description* are provided is not significant.

```
ORIGIN-DESCRIPTION ::=
   sid “=” <VendorID> “:” <DatabaseID>
```

The *origin-description* consists of the sid tag-name (service identifier) and a corresponding tag-value. This tag-value consists of two parts that are separated by a colon. Example: *sid=EBSCO:MFA*

```
OBJECT-DESCRIPTION ::= <ZONE>(“&”<ZONE>)*
ZONE ::=
   (<GLOBAL-IDENTIFIER-ZONE> | <OBJECT-METADATA-ZONE> |
   <LOCAL-IDENTIFIER-ZONE>)
```

All zone(s) are optional, but at least one of them must be provided.

[112] OpenURL demonstrator: http://www.ukoln.ac.uk/distributed-systems/openurl/

[113] UKOLN - UK Office for Library Networking

[114] Herbert Van de Sompel; Patrick Hochstenbach; Oren Beit-Arie - OpenURL Syntax Description

```
GLOBAL-IDENTIFIER-ZONE ::=
   ‶id=″<GLOBAL-NAMESPACE>″:″<GLOBAL-IDENTIFIER>
   (‶&id=″<GLOBAL-   NAMESPACE>″:″<GLOBAL-IDENTIFIER>)*

GLOBAL-NAMESPACE :==
(‶doi[115]″ | ‶pmid[116]″ | ‶bibcode[117]″ | ‶oai[118]″)
```

The *global-identifier* must by globally unique in its corresponding namespace. Example (*global-identifier-zone* consisting of two identifiers):
id=doi:123/345678&id=pmid:202123

```
OBJECT-METADATA-ZONE ::=
   <META-TAG> ‶=″ <META-VALUE>
   ( ‶&″ <META-TAG> ‶=″ <META-VALUE> )*

META-TAG ::=
   ( ‶genre″ | ‶aulast″ | ‶aufirst″ | ‶auinit″ | ‶auinit1″ |
   ‶auinitm″ | ‶coden″ | ‶issn″ | ‶eissn″ | ‶isbn″ | ‶title″ |
   ‶stitle″ | ‶atitle″ | ‶volume″ | ‶part″ | ‶issue″ | ‶spage″ |
   ‶epage″ | ‶pages″ | ‶artnum″ | ‶sici″ | ‶bici″ | ‶ssn″ |
   ‶quarter″ | ‶date″ )
```

The *object-metadata-zone* is used for metadata elements of the transported metadata object in a format that is shared among all OpenURLs. If for some reason metadata elements cannot be described in this common format, they can still be included in the *private-identifier-zone*.

```
LOCAL-IDENTIFIER-ZONE ::= ‶pid=″ VCHAR+
```

The *local-identifier-zone* allows the transport of metadata in formats that are specific to the originating information system and that cannot be expressed in the standardised syntax proposed for the *object-metadata-zone*. Example:
pid=<author>Hilse, Hans-Werner; Kothe, Jochen</author>&<yr>05</yr>

OpenURLs are URLs which means that all characters not allowed must be encoded, for example, the OpenURL:

http://sfxserver.uni.edu/sfxmenu?sid=EBSCO:MFA&id=pmid:203456&pid=<author>Smith, Paul ; Klein, Calvin</author>&<yr>98</yr>

and the corresponding correctly encoded OpenURL:

http://sfxserver.uni.edu/sfxmenu?sid=EBSCO:MFA&id=pmid:203456&pid=%3Cauthor%3ESmith%2C%20Paul%20%3B%20Klein%2C%20Calvin%3C%2Fauthor%3E&%3Cyr%3E98%2F1%3C%2Fyr%3E.

An OpenURL using an HTTP GET request format longer than 255 characters may not function successfully in all circumstances. Some older Web clients (browsers) or proxy servers might not properly support URLs consisting of more than 255 characters. *A priori* the length of an OpenURL is not limited and all

[115] DOI: Document Object Identifier

[116] pmid: PubMed identifier

[117] bibcode: Identifier used in Astrophysics Data System

[118] oai: Identifier used in the Open Archives initiative

modern software should accept longer OpenURLs.
However, very long OpenURLS should be sent encoded in a HTTP POST query instead of a HTTP GET query as POST queries are not limited in early implementations of the HTTP standard.

8.4 Current applications

Until its approval by the American National Standards Institute (ANSI) on 15 April 2005, the standard had been in trial use since June 2003. There are many services basing on OpenURL in the Web; a list of services based on OpenURL is available at the 'Ex Libris'[119] website.

8.5 Long-term perspective

Because the OpenURL Framework is now a NISO standard and is being used in several commercial and non-commercial services, it is not probable that OpenURL will cease to exist.

8.6 Participation

Institutions can host their own linking servers (such as SFX,[120] LinkFinderPlus,[121] Openly Informatics[122]) and configure their own localised linking environment.

As OpenURL does not rely on principles such as being unique themselves but rather integrates various concepts of persistent identification, it is only an additional option for institutions that plan to implement a persistent identifier strategy.

Institutions that focus on specialised services for objects that are identified by one of OpenURL's supported identification schemes may have an interest in integrating the OpenURL concept. It allows for offering a context-sensitive service to users.

8.7 Summary

OpenURL is not a scheme for persistent identification but itself makes use of (persistent) identifiers.

OpenURL is a metadata transport protocol.

OpenURLs are mainly used for cross linking and citation.

The OpenURL concept is designed to integrate other identifier schemes.

OpenURL is able to anticipate access management based on user context.

OpenURL is designed to enable value-added services.

OpenURL is a NISO standard; there are commercial and non-commercial implementations using OpenURL.

[119] Ex Libris – OpenURL Enabled Resources - http://www.exlibrisgroup.com/sfx_sources.htm
[120] SFX:http://www.exlibrisgroup.com/sfx.htm
[121] LinkFinderPlus http://www.endinfosys.com/prods/linkfinderplus.htm
[122] Openly Informatics http://www.openly.com/

9 Guidelines and recommendations

It should be noted that among all the concepts which have been introduced there is no ‘one size fits all’ strategy for implementing persistent identifiers. Although the basic problems to be solved are the same, each of the systems addresses them in its own way on different administrative and technical levels.

Depending on your individual strategy, (a) one persistent identifier concept introduced here may exactly fit the purpose, (b) multiple systems do, or (c) it could be best not to choose any existing system at all and roll out your own specifications.

It is therefore not possible to formulate one single recommendation for all readers of this report. Instead, we will introduce some important questions organisations need to consider, followed by some possible strategies to address various problems. This should give directions for deciding which approach to take.

9.1 Determining the status quo

First of all, an institution must carefully analyse its current use of identifiers in general. In most cases, where data is collected, it is identified in some way. If it is data about other data or objects – metadata – it will often contain an identifier for the referred item. In every data collection the following should be analysed:

- Is persistence needed for this kind of data? How would persistence be defined?
- What features should the implemented system offer? (e.g. resolving, metadata exchange, etc.)

Organisations need to decide whether one of the systems described suits all needs or whether it would be preferable to implement more than one system.

In most cases, identification and naming systems are already in use. These systems were mostly not designed with persistence as a first consideration. But nevertheless it may be worth thinking about integrating the old naming scheme in the new one – leading to another important question:

- Are there identification mechanisms in use that should be incorporated into the new strategy, e.g. as a sub-namespace?

Often, an institution already cooperates with other institutions that deal with a similar environment. Before deciding to opt for a new implementation, organisations should therefore consider the following:

- Are there strategies for persistent identification already in active use at partner institutions?
- Do you require interoperability with those institutions?

9.2 Strategies to choose

9.2.1 Creating awareness

It has already been noted that persistence is always a matter of administration. A system for persistent identification is built just to ease the administration, not to make it obsolete. Computers cannot easily take account of changes in real life if these have not been anticipated, so there is no automated solution for persistence: changes must be reflected by administrative work.

Because of this, the most important task for all institutions that are developing their document persistence strategy is to create awareness of the problems of persistence among all individuals concerned with handling the documents both technically and administratively. An institution should issue internal policies for its use of persistent identifiers to avoid a mixture of incompatible implementations.

Institutions should recognize that the implementation of persistent identifiers always comes with some costs. All changes in location, ownership or other metadata must be reflected in the persistent identifier system. Consequently, each migration of the document base, e.g. to a new document server, involves some work in maintaining the identification system.

If an institution offers services to external users, it should clarify its own policy regarding the persistence of the identifiers and explain the practical use of the identifiers. This includes directions for resolving or citing these identifiers.

9.2.2 Implementing a system starting only locally

When, after considering the questions posed in the previous section, an organisation has reached the conclusion that cooperation with other institutions is not an option, it may choose to implement persistent identifiers according to an individual policy.

In such a situation the focus will mainly be on the end users, making it necessary to evaluate what features they expect and what systems may fit these expectations. A strategy is probably better than none, so one may start with a system that is simple to implement – and may lack a few features of the competing systems. Very careful checks should be made whether systems that are already existent (database identifiers, sequential numbering schemes, etc.) should be reused in the definition of the naming policy. It is important to emphasize that if such schemes are to be integrated, these systems must either ensure persistent uniqueness themselves, or are only used at one point in time and checked for conflicts with identifiers already issued.

For example, where a resolver extracts a certain portion of the identifier within its sub-namespace and uses this extracted string as database identifier, it must be ensured that the database identifier is itself persistent in order not to compromise the full identifier, too. Typical cases are deletion in the database and re-issuing of the database identifier.

As noted before, efforts being made for offering services using persistent identifiers should be publicised to the users and if at all possible to the public, too.

Identification is a crucial element in communication. It should therefore be recognised that identifiers are in most cases used regardless of closed domains. Whenever an object needs to be referred to and there is some kind of identifier of a scheme known to the communicating individuals, it will be used. In most cases it is best to cooperate among institutions that use one specific identification scheme or are actively taking part in the community (see below). There are probably very few reasons to try to implement an identifier scheme with the intent to use it only in a closed domain. In most cases it is better to prepare for a broader use of the identifiers, even if there does not seem to be current need.

9.2.3 Establishing a common infrastructure among multiple institutions

If an institution is in a position to organise the process of implementing a strategy for persistent identification together with other institutions (e.g. national libraries, consortia), it may decide to suggest the adoption of one particular system for all associated institutions, as this will greatly enhance interoperability. There are a few important factors to consider:

Technical interoperability among institutions?

In most cases, a common infrastructure is established for interoperability reasons. An example would be if one wants to make use of the 'extended' features of some of the persistent identification systems, e.g. the delivery of object metadata as defined by the ARK. In that case, it would be important that the specific protocol for this task (THUMP) is supported by all implementations.

If the group of institutions intend to use the identifiers only for linking, cataloguing and simple URL resolution and redirection, *and* the organisations have heterogeneous requirements, it may not be necessary to restrict the use of different systems. The group should, however, develop a common naming scheme to be implemented as a sub-namespace for each system in use.

Synergy effects

It should be noted, however, that if one specific solution is implemented at all institutions, it will be easier to share knowledge and administrative tasks within the group. In addition, virtually all systems introduced here allow for later consolidation of the infrastructure.

Before implementing a system, organisations should investigate whether there are possible cooperation partners that have similar problems to solve.

Open towards other interested parties

A restricted group of implementers without the possibility for others to join does not really work for most persistent identifier strategies. To assure the biggest acceptance among its users, a system for persistent identification should not be kept separate unnecessarily, as such a system would lose some of its possible visibility due to its proprietary nature. An institution should generally publicize its efforts in this field and list all responsible contact persons.

9.2.4 Common naming guidelines among multiple institutions

It has been noted that the lowest common denominator of an institution's strategy should be a common naming scheme. This scheme should

- explain a common syntax for names independent of the identifier system used,
- explain what characters should be allowed to construct names, and
- explain how to encode them for systems that do not implement characters that the common scheme allows.

Of the systems introduced, the Archival Resource Key has the most restrictive syntax. However, it allows encoding of characters that are not directly allowed. The Handle system and therefore the DOI system both use UTF-8 character encoding, resulting in some bit combinations having special meaning and being reserved. Some care should therefore be exercised to find a naming policy that fits all possible identifier systems.

An institution should consider restricting the use of ‘speaking’ names. While these may make sense for date strings, wording that tries to abstract the object description should be avoided. Words can be expected to change their recognised meaning over time whereas the identifiers should never be subject to change. So a word that may perfectly describe the identified object today may be misleading in the future.

There are situations where it is not even sensible to integrate date and time strings into the name: dynamic objects may be subject to change or the string in the identifier may be more misleading than helpful.

Always remember that your identifiers are intended to be used both by machines and humans: they should permit being spoken out (or spelled, respectively), written down and being parsed by machines. Introducing the full Unicode character set as offered by Handles and DOIs is therefore probably impractical: some people may not have even seen the more exotic characters that are possible here, and other characters introduce the risk of being mistaken for others (e.g. accented characters).

9.2.5 Using persistent identifiers

Persistent identifiers can attract parties not directly involved in the scientific communication process, as long as they provide a clear definition on which applications and services can rely.

Projects that offer linking services for electronic documents that are based on persistent identifiers have already been established, and other projects use persistent identifiers to track down and analyse citations in electronically published resources.

There are many more concepts for the use of persistent identifiers: it would be possible to offer services based on documents of an arbitrary nature, e.g. print-on-demand of electronic documents, incorporation of metadata into search machines, reselling access to electronic documents, etc. It is beyond the scope of this report to give a full description of how persistent identifiers may be used.

9.2.6 Using identifiers without issuing them

For individuals and institutions that do not deal with the issuing of persistent identifiers or the offering of services that make use of the infrastructure that goes along with PI systems, it is nevertheless important to be aware of some basic principles of those systems.

Cataloguers

If an institution aims at the cataloguing of resources that are identified in a persistent way, the institution should try to follow the development of the systems described here and those systems that may be developed in the future. There is no need to implement specialised technology, but the different identification schemes should be understood and well separated. E.g., when cataloguing resources that have heterogeneous kinds of persistent identifiers, the corresponding identifier scheme should be recorded together with the persistent identifier. At the time of writing, it is possible to recognize most identifiers' schemes by just looking at them, but as schemes evolve this may become more difficult. Cataloguers should adopt a fixed syntax to describe the exact identification scheme in use.

Authors

An author of a new document should also be aware of persistent identifiers. If electronic documents are cited within the new document, they should be cited by their persistent identifier. This way it is possible to ensure the continuous accessibility of the cited resource by use of the citation. This also ensures the visibility of the document itself when the citations are tracked back to it at a later point. Citations are very important today as they permit an evaluation of the structure of scientific communication. Analyses of various kinds are made on the basis of citation databases.

Today, persistent identifiers are not widely recognised and are not well integrated into current software for the Web, e.g. browsers. So it seems to be good advice to cite them as URLs. For those identifiers that are themselves URLs (ARKs, PURLs), they can be cited verbatim. Other identifiers (Handles, DOIs, URNs) mostly have a central resolving service and can be appended to its URL. Depending on the new document's medium, special care should be taken not to let identifiers be hyphenated automatically, as only the ARK concept is aware of this problem at the moment.

Finally, the author should try to get a persistent identifier assigned to his own work. The importance of continuous accessibility has already been mentioned. It should be noted that it is quite possible for individuals to get a persistent identifier assigned, e.g. at DOI Registration Agencies or as a PURL at OCLC or even at a self-maintained PURL service (which we do not recommend for individuals).

9.3 Long-term perspective of software, protocols and concepts

It has been noted that, at its core, persistence is an administrative task that cannot be replaced by technology. It can, however, be assisted by technology. Technology is also crucial for the automation of services using persistent identifiers. The identifier never loses its identification function. But the underlying infrastructure may possibly cease.

9.3.1 Durability of software and protocols due to broad adoption

This leads to an important question: how durable are the underlying techniques like software and protocols? This question cannot be answered absolutely, but it can be noted that

- broad adoption of a software or protocol causes further adoption, and as such increases the probability of continued availability;
- if the software or protocol is easily extensible by e.g. accommodating a version mechanism, the overall concept is more likely to be reused rather than declared obsolete;
- the simpler the software or protocol is to implement, the bigger is the probability of broad adoption.

9.3.2 Future tasks

Regardless of what software or protocol will be implemented, the time will come when it or a crucial part of the infrastructure needed for its proper working becomes obsolete. It is therefore inevitable that the technical infrastructure of a persistent identifier system will need to be migrated at some point in the future.

9.3.3 Less technology, more persistence

This shows that a system for persistent identification should rely on technology as little as possible. It is therefore imperative to ensure that no technically meaningful semantics creeps into the syntax of persistent identifiers. This may be covered by a proper definition of the syntax.

In this context, the integration of persistent identifiers into URL strings is risky: it introduces problems if those URLs are not clearly marked as containing a certain encoded identifier. In this case, the URL cannot easily be converted at a later point in time because it is hard to determine which element is, in fact, an encoded identifier.

9.4 Support through use: identifier politics

Acceptance by users will be an important measure for future implementations of one of the PI systems. At some point in time, more users will become aware of the problem of persistent naming of documents. Today, most users still expect electronic documents available on the Internet to be referred by a URL. Over time, more and more documents cease to be accessible by their old URL, which should induce users to recognise the importance of a persistent naming layer. This is when we can expect a large-scale adoption of the new technology, such as full integration in end user software (e.g., Web browsers).

The systems introduced in this report have many features in common. In fact, because all are addressing one basically identical problem, it can be argued that they are mostly exchangeable or can be made subsidiary to one another (e.g., by including one system's namespace in a specified subset of another system's namespace). This means there is a kind of competition between the systems. They are still being discussed controversially and it is still not quite clear if there is a future for all of them.

Implementing a system for persistent identification always implies supporting its architecture. The reputation of the institutions backing a certain technology has strong influence on the reputation of the technology itself. The usage count of a persistent identification system is an important measure for further implementations, and an institution's decision will therefore influence further discussions.

It is important to realise that most institutions are in a situation where various architectures would fit. An institution should be careful to choose a strategy that allows the use of every system that could possibly do the job, in order to keep open the option for a later change of the system, if needed. If an organisation could implement multiple systems, it should start with the easiest implementation because it may need to keep the infrastructure for that system running when persistent identifiers according to that system have been issued to ensure their continued functioning. For easier management of workflows, it is advisable to have one single naming scheme in sub-namespaces of each identification system to be used.

9.5 Joining the discussion

All the systems described here are under development and standardization work is still in progress. For most systems there are communities, meetings and conferences. Institutions which need to develop this expertise should consider taking part.

There are also some mailing lists where discussion takes place. This would be a good starting point to ask questions regarding the implementation of persistent identifiers.

An institution should carefully examine possible cooperation with other organisations. Most of the persistent identifier systems' technical infrastructure can be shared among multiple institutions and can be centrally administered.

Some pointers to Internet home pages of initiatives dealing with persistent identifiers have been provided in the reference section of this report.

9.6 Why not recommend a specific implementation

The authors of this report hesitate to recommend the implementation of one specific technical implementation. This is due to many reasons:First of all, in most cases the actual implementation is secondary to the commitment to identifier persistence. However, none of these systems ensure persistence: persistence can only achieved by administrative commitment.

This also means that the actual system to be implemented should be chosen after contacting all parties involved with assigning, maintenance and usage of the identifiers. Existing technology, knowledge and experience are important factors that heavily influence the costs of the technical implementations.

Another important factor is whether you need to support a specific scheme in order to enable third party services to identify and access your documents (e.g. national long term storage initiatives, print-on-demand services, centralized search and browsing).

Based on their experience, the authors of this report note that, generally speaking, end-users are not aware of the variety in possible implementations and do not even have a great interest unless something is not resolvable. Unsatisfactory usability of the chosen system will obviously affect end-users. All implementations introduced in this report have their own philosophy of how to meet users' expectations.

Which system is optimal in your environment depends on many factors. The following section introduces a list of questions that should help you to clarify the important points and finally decide for a specific implementation.

With the exception of DOIs, all implementations are likely to induce similar costs. DOIs are special in that the costs depend on the Registration Agency and its business model and service coverage.

If an institution is not yet sure of which specific scheme to adopt, there is nothing wrong with starting to issue persistent identifiers locally, i.e. only valid in the institution itself. By carefully deciding on the syntax of those identifiers, it is easy to integrate them into a namespace issued to the institution for any of the schemes at a future point. E.g., if an institution chooses a simple, short internal identifier scheme (while avoiding complex character sets) it is easy to integrate them into any of the schemes introduced in this report at a later date.

9.7 Checklist

The following questions and comments are meant to help you to implement persistent identifiers and select a specific scheme. The list is not exhaustive, but may help you to consider all important points.

Administrative commitment

- Is the administration aware of the future costs for maintenance of issued identifiers?
- Is the administration willing to contract other institutions for the maintenance of identifier data (e.g. DOI registration agencies)?

- An institution should issue a central policy for all departments that deal with the identifiers explaining how identifiers are to be issued, maintained and resolved.
- If the identifiers are to be used by end users, a policy regarding the stewardship for maintenance of the identifiers should be made available to them, as well.

Existing identifier schemes

- Does your organisation already use identifiers for resources?
- If so, are those identifiers unique and stable?
- Can those identifiers be integrated into one of the larger scale schemes introduced in this report (e.g. character set issues, practical issues such as length)?

Available technology and knowledge

- Which identifier schemes are already supported by existing technology (e.g. cataloguing software, databases)?
- Are there schemes that can not be integrated into existing technology (e.g. character set issues, resolving technology)?

Co-operations, third parties

- Are there workgroups and discussion facilities that deal with specific schemes?
- Should this be required by your administration: is commercial support available (i.e. contractors)?
- Are there initiatives on a regional level that support specific implementations (e.g. national libraries, URN-NBN scheme)?

Users' demands

- What identifier scheme is most intuitive to use and supports your users' needs? Do the users actually care and if that is the case, do they prefer specific schemes?

Appendix A. Timeline

	System	Organisation
1991	URI/URL	CERN
	..	
1994	URN	IETF
1995	Handle	CNRI
1996	PURL	OCLC
1997	DOI	AAP
1998	NBN	National Library of Finland
1999	OpenURL	
	..	
2001	ARK	National Library of Medicine (US)
	URN:NBN	Helsinki University Library

Appendix B. Glossary

AAP	Association of American Publishers http://www.publishers.org/
ACS	American Chemical Society http://pubs.acs.org/
AIP	American Institute of Physics http://www.aip.org/
ALCS	Authors Licensing and Collecting Society http://www.alcs.co.uk/
ANSI	American National Standards Institute http://www.ansi.org/
APS	American Physical Society http://www.aps.org/
ARK	Archival Resource Key http://www.cdlib.org/inside/diglib/ark/
BICI	Book Item and Component Identifier
CCC	Copyright Clearance Centre http://www.copyright.com/
CNRI	Corporation for National Research Initiatives http://www.cnri.net/
CORDS	Copyright Office Electronic Registration, Recordation, and Deposit System http://www.copyright.gov/cords/
CSIRO	Commonwealth Scientific and Industrial Research Organisation http://www.csiro.au/
DARPA	Defense Advanced Research Projects Agency http://www.darpa.mil/
DDDS	Dynamic Delegation Discovery System RFC 3404.
DNS	Domain Name System RFCs 1034, 1035.
DOI	Digital Object Identifier http://www.doi.org/
IANA	Internet Assigned Numbers Authority htpp://www.iana.org
IDF	International DOI Foundation http://www.doi.org/
IEEE	Institute of Electrical and Electronics Engineers – the IEEE is a global technical professional society serving the public interest and

	members in electrical, electronics, computer, information & other technologies. http://www.ieee.org/
IESG	The Internet Engineering Steering Group http://www.ietf.org/iesg.html
IETF	The Internet Engineering Task Force http://www.ietf.org/
INDECS	Interoperability of Data in E-Commerce Systems http://www.indecs.org/
IRI	International Resource Identifier RFC 3987.
ISBN	International Standard Book Number
ISCW	International Standard Musical Work Code
ISO	International Organisation for Standardization http://www.iso.org
ISSN	International Standard Serial Number
JISC	Joint Information Systems Committee http://www.jisc.ac.uk/
NAA	Name Assigning Authority (ARK)
NAAN	Name Assigning Authority Number (ARK)
NAPTR	Naming Authority Pointer RFC 2915.
NBN	National Bibliographic Number– a URN Namespace Identifier developed and registered by the National Library of Finland
NCSTRL	Networked Computer Science Technical Reports Library http://www.cnri.net/cstr.html
NISO	National Information Standards Organization http://www.niso.org/
NLA	National Library of Australia http://www.nla.gov.au/
NMA	Name Mapping Authority (ARK)
NMAH	Name Mapping Authority Hostport (ARK)
OCLC	An international not-for-profit cooperative of libraries and other institutions that share a common database (WorldCat) to identify and share resources and to share research into libraries and information science. Originally, OCLC stood for Ohio College Library Center. Today, the full legal name is OCLC Online Computer Library Center, Inc.
PURL	Persistent Uniform Resource Locator
PI	Persistent Identifier
PICS	Platform for Internet Content Selection http://www.w3.org/PICS/
PII	A Publisher Item Identifier (PII) is a standard agreed by ACS, AIP, APS, Elsevier, and IEEE. It provides a unique identification of individual published documents.

RDF	Resource Description Framework http://www.w3.org/RDF/
RFC	The Requests for Comments (RFC) document series is a set of technical and organizational notes about the Internet (originally the ARPANET), beginning in 1969. http://www.rfc-editor.org/
SICI	Serial Item and Contribution Identifier – human-readable PI for elements of periodicals (like articles)
URC	Uniform Resource Characteristics
URI	Uniform Resource Identifier http://www.ietf.org/rfc/rfc3986.txt
URL	Uniform Resource Locator
URN	Uniform Resource Name
W3C	The World Wide Web Consortium (W3C) is an international consortium where Member organisations, a full-time staff, and the public work together to develop Web standards. http://www.w3.org/

Appendix C. Further information

Other reports and comparisons

Erpanet: Seminar on Persistent Identifiers. Cork, UK, 2004.
With various presentations and a final report.
http://www.erpanet.org/events/2004/cork/

National Library of Australia: Report on a consultancy conducted by Diana Dack for the NLA. May, 2001.
http://www.nla.gov.au/initiatives/persistence/PIcontents.html

Høgås, Hilde; van der Werf, Titia; Powell, Andy: BIBLINK – LB 4034.
D2.1 Identification. A study for the European Commission. May, 1997.
http://hosted.ukoln.ac.uk/biblink/wp2/d2.1/

Mailing lists, forums

Dublin Core Metadata Initiative (DCMI) Persistent Identifiers Working Group:
Mailing list information:
http://www.jiscmail.ac.uk/lists/DC-PERSISTENT-IDENTIFIERS.html

PURL mailing list: Mailing list information:
http://purl.oclc.org/docs/subscribe.html

International DOI Foundation: Pointers to mailing lists and working groups on their Web page:
http://doi.org/maillist-info1.html

Link lists

Preserving Access to Digital Information (PADI): Persistent Identifiers. Kept up to date.
http://www.nla.gov.au/padi/topics/36.html

References

URLs given here are working as of October 2006. For the resolution of Handles and DOIs given here, it is suggested to use the resolver service at http://www.crossref.org/

Internet RFCs:

Postel, J.: RFC 791 – Internet Protocol – DARPA Internet Program – Protocol Specification. 1981. http://www.ietf.org/rfc/rfc791.txt

Mockapetris, P.: RFC 1034 – Domain names – concepts and facilities. 1987. http://www.ietf.org/rfc/rfc1034.txt

Mockapetris, P.: RFC 1035 – Domain names – implementation and specification. 1987. http://www.ietf.org/rfc/rfc1035.txt

Berners-Lee, T.: RFC 1630 – 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. 1994. http://www.ietf.org/rfc/rfc1630.txt

Sollins, K. and Masinter, L.: RFC 1737 – Functional Requirements for Uniform Resource Names. 1994. http://www.ietf.org/rfc/rfc1737.txt

Moats, R.: RFC 2141 - URN Syntax. 1997. http://www.ietf.org/rfc/rfc2141.txt

Daniel, R. and Mealling, M.: RFC 2168 – Resolution of Uniform Resource Identifiers using the Domain Name System. 1997. http://www.ietf.org/rfc/rfc2168.txt

Daniel, R.: RFC 2169 – A Trivial Convention for using HTTP in URN Resolution. 1997. http://www.ietf.org/rfc/rfc2169.txt

Lynch, C., Preston, C., and R. Daniel: RFC 2288 – Using Existing Bibliographic Identifiers as Uniform Resource Names. 1998. http://www.ietf.org/rfc/rfc2288.txt

Berners-Lee, T.; Fielding, R.; Irvine, U.C.; Masinter, L.: RFC 2396 – Uniform Resource Identifiers (URI): Generic Syntax. 1998. http://www.ietf.org/rfc/rfc2396.txt

Deering, S. and Hinden, R.: RFC 2460 – Internet Protocol, Version 6 (IPv6) Specification. 1998.http://www.ietf.org/rfc/rfc2460.txt

Moats, R.: RFC 2648 – A URN Namespace for IETF Documents. 1999. http://www.ietf.org/rfc/rfc2648.txt

Mealling, M.: RFC 3043 – The Network Solutions Personal Internet Name (PIN): A URN Namespace for People and Organizations. 2001. http://www.ietf.org/rfc/rfc3043.txt

Rozenfeld, S.: RFC 3044 – Using The ISSN (International Serial Standard

Number) as URN (Uniform Resource Names) within an ISSN-URN Namespace. 2001. http://www.ietf.org/rfc/rfc3044.txt

Mealling, M.: RFC 3061 – A URN Namespace of Object Identifiers. 2001. http://www.ietf.org/rfc/rfc3061.txt

Coates, A., Allen, D., and D. Rivers-Moore: RFC 3085 – URN Namespace for NewsML Resources. 2001. http://www.ietf.org/rfc/rfc3085.txt

Best, K. and N. Walsh: RFC 3120 – A URN Namespace for XML.org. 2001. http://www.ietf.org/rfc/rfc3120.txt

Best, K. and N. Walsh: RFC 3121 – A URN Namespace for OASIS. 2001. http://www.ietf.org/rfc/rfc3121.txt

Walsh, N., Cowan, J., and P. Grosso: RFC 3151 – A URN Namespace for Public Identifiers. 2001. http://www.ietf.org/rfc/rfc3151.txt

Hakala, J. and H. Walravens: RFC 3187 – Using International Standard Book Numbers as Uniform Resource Names. 2001. http://www.ietf.org/rfc/rfc3187.txt

Hakala, Juha: RFC 3188 - Using National Bibliography Numbers as Uniform Resource Names. 2001. http://www.ietf.org/rfc/rfc3188.txt

Mealling, M.: RFC 3401-3405 – Dynamic Delegation Discovery System (DDDS) Part One – Part Five. 2002. http://www.ietf.org/rfc/rfc3401.txt http://www.ietf.org/rfc/rfc3405.txt

Daigle, L., van Gulik, D., Iannella, R., and P. Faltstrom: RFC 3406 – Uniform Resource Names (URN) Namespace Definition Mechanisms. BCP 66. 2002. http://www.ietf.org/rfc/rfc3406.txt

Walsh, A.: RFC 3541 – A Uniform Resource Name (URN) Namespace for the Web3D Consortium (Web3D). 2003. http://www.ietf.org/rfc/rfc3541.txt

Morgan, R. and K. Hazelton: RFC 3613 – Definition of a Uniform Resource Name (URN) Namespace for the Middleware Architecture Committee for Education (MACE). 2003. http://www.ietf.org/rfc/rfc3613.txt

Smith, J.: RFC 3614 – A Uniform Resource Name (URN) Namespace for the Motion Picture Experts Group (MPEG). 2003. http://www.ietf.org/rfc/rfc3614.txt

Gustin, J. and A. Goyens: RFC3615 – A Uniform Resource Name (URN) Namespace for SWIFT Financial Messaging. 2003. http://www.ietf.org/rfc/rfc3615.txt

Bellifemine, F., Constantinescu, I., and S. Willmott: RFC 3616 – A Uniform Resource Name (URN) Namespace for Foundation for Intelligent Physical Agents (FIPA). 2003. http://www.ietf.org/rfc/rfc3616.txt

Mealling, M.: RFC 3622 – A Uniform Resource Name (URN) Namespace for the Liberty Alliance Project. 2004. http://www.ietf.org/rfc/rfc3622.txt

Sun, S.; Lannom, L.; Boesch, B.: RFC 3650 – Handle System Overview. 2003. http://www.ietf.org/rfc/rfc3650.txt

Sun, S.; Reilly, S.; Lannom, L.: RFC 3651 – Handle System Namespace and Service Definition. 2003. http://www.ietf.org/rfc/rfc3651.txt

Sun, S.; Reilly, S.; Lannom, L.; Petrone, J.: RFC 3652 – Handle System Protocol (ver 2.1) Specification. 2003. http://www.ietf.org/rfc/rfc3652.txt

Steidl, M.: A Uniform Resource Name (URN) Namespace for the International Press Telecommunications Council (IPTC). 2004. http://www.ietf.org/rfc/rfc3937.txt

Berners-Lee, T.; Fielding, R.; Masinter, L.: RFC 3986 – Uniform Resource Identifier (URI): Generic Syntax. 2005. http://www.ietf.org/rfc/rfc3986.txt

Duerst, M. and Suignard, M.: RFC 3987 - Internationalized Resource Identifiers (IRIs). 2005. http://www.ietf.org/rfc/rfc3987.txt

Leach, P., Mealling, M., and R. Salz: RFC 4122 – A Universally Unique IDentifier (UUID) URN Namespace. 2005. http://www.ietf.org/rfc/rfc4122.txt

W3C material

Berners-Lee, T.: Axioms of the Web architecture: 2. The Myth of Names and Addresses. 1996. http://www.w3.org/DesignIssues/NameMyth.html

Berners-Lee, T.: Design Issues for the Web. Naming. 1991. http://www.w3.org/DesignIssues/Naming.html

Berners-Lee, T.: HTTP Addressing. http://www.w3.org/Addressing/HTTPAddressing.html

Unspecified author, probably Berners-Lee, T.: HTTP Request. 1992. http://www.w3.org/Protocols/HTTP/Request.html

Further resources

International DOI Foundation (ed.): DOI Factsheet ‘DOI and Data Dictionaries’. Version 2.1. November 2004. http://www.doi.org/factsheets/DOIDataDictionaries.html

International DOI Foundation (ed.): The DOI Handbook. Version 4.2. February 2005. HTML: Handle/DOI: 10.1000/182 PDF: Handle/DOI: 10.1000/186

Byrnes, Margaret M.: Defining NLM’s Commitment to the Permanence of Electronic Information. October 2000. http://www.arl.org/newsltr/212/nlm.html

Erpanet: Seminar on Persistent Identifiers. Final Report. Cork, UK, 2004.
http://www.erpanet.org/events/2004/cork/Cork%20Report.pdf

Kahn, Robert and Wilensky, Robert: A Framework for Distributed Digital Object Services. 1995. Handle: cnri.dlib/tn95-01

Kunze, John A.: Towards Electronic Persistence Using ARK Identifiers. July 2003. http://www.cdlib.org/inside/diglib/ark/arkcdl.pdf

Kunze, John A.: The ARK Persistent Identifier Scheme. Internet Draft. February 2005. http://www.cdlib.org/inside/diglib/ark/arkspec.pdf (PDF) http://www.cdlib.org/inside/diglib/ark/arkspec.txt (TXT)

Kunze, John A.: A Metadata Kernel for Electronic Permanence. In: Journal of Digital Information, Vol 2, Issue 2, Article 84. November 2002. http://jodi.ecs.soton.ac.uk/Articles/v02/i02/Kunze/ http://dot.ucop.edu/home/jak/mdkernel.nr.pdf (archived)

OpenURL Framework Standard. Approved NISO standard Z39.88.

http://www.niso.org/standards/standard_detail.cfm?std_id=783

Shafer, Keith; Weibel, Stuart; Jul, Erik; Fausey, Jon: Introduction to Persistent Uniform Resource Locators. 1996. http://purl.oclc.org/docs/inet96.html

Van de Sompel, Herbert; Hochstenbach, Patrick; Beit-Arie, Oren: OpenURL Syntax Description. 2000. http://www.exlibrisgroup.com/sfx_openurl_syntax.htm

Traditionally, references to web content have been made by using URL hyperlinks. However, as links are 'broken' when content is moved to another location, a reference system based on URLs is inherently unstable and poses risks for continued access to web resources.

To create a more reliable system for referring to published material on the web, from the mid-1990s a number of schemes have been developed that use name spaces to identify resources, enabling retrieval even if the location on the web is unknown.

This report was written to explain the principle of persistent identifiers and help institutions decide which scheme would best fit their needs. It discusses Handles, Digital Object Identifiers (DOIs), Archival Resource Keys (ARKs), Persistent Uniform Resource Locators (PURLs), Uniform Resource Names (URNs), National Bibliography Numbers (NBNs), and the OpenURL, providing examples and extensive references for each.

Consortium of European Research Libraries
European Commission on Preservation and Access