

	<p>web: http://resolver.net.au/hdl/102.100.272/0N8J991QH email: policy@pilin.net.au</p>
---	--

Version History

Version	Date	Status & changes	Expression identifiers
V1.0	2008-05-19	"Identifier Service Guidelines"	PILIN/NQNDLPDQH hdl:102.100.272/NQNDLPDQH
V1.1	2008-05-19	Counter draft for public discussion.	
V1.2	2008-05-24		PILIN/R8PN4LTRH hdl:102.100.272/R8PN4LTRH

Draft PILIN Guideline

Resolution Service Guidelines

To cite the *latest* version of this work use <http://resolver.net.au/hdl/102.100.272/1KKBLPDQH>

To cite *this* version of this work, use <http://resolver.net.au/hdl/102.100.272/R8PN4LTRH>

1 Purpose

On the internet, there is a longstanding confusion between identifiers and service requests, since URLs can be considered as both. This becomes a problem if an identifier is to be used with more than one type of service. This document makes best practice recommendations on:

- distinguishing identifiers from service requests on those identifiers;
- distinguishing the identifier from the rest of the service request;
- ensuring that identifiers are seen to be actionable by more than one service.

2 Resolution services

An identifier resolution service is a service that returns information about the thing identified, which distinguishes the thing from all other things. A resolution service can return information about how to retrieve the thing; but it is not limited to that function.

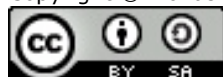
An identifier is actionable if it has some action associated with it.¹ A service on an identifier is an action operating on an identifier through a defined protocol for requests and responses. So an identifier that can be resolved through a service is actionable—though other types of action can also be defined.

For computer processes, identifiers are only meaningful if they are actioned through some defined service. So the identifier will not appear outside the context of those services, and will only be presented to processes through service requests.

At a higher level, the identifier coordinates the various services that can act on it, and can be presented as an abstraction outside of any services that make it

¹The PILIN Ontology restricts actionability further: the action must relate to the association between name and thing (so not merely citing the identifier name), and it must not be a curatorial action (e.g. updating the identifier).

Copyright © Monash University



This work is licensed under the Creative Commons Attribution-Share Alike 2.5 Australia License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-sa/2.5/au/>

This work was created as part of the PILIN project. The PILIN project is funded by the Australian Commonwealth Department of Education, Science and Training, (DEST) under the Systemic Infrastructure Initiative (SII) as part of the Commonwealth Government's Backing Australia's Ability – An Innovation Action Plan for the Future (BAA) under the ARROW Project.

actionable. So for information management, the identifier is more important than any service requests.

2.1 Service Request

The service request uses a request protocol to encode the service host, the service and the identifier to be actioned. Other parameters may also be encoded.

For example, the URL `http://resolver.net.au/hdl/102.100.272/0N8J991QH` is a service request for identifier resolution:

- `http://` is the request protocol.
- `resolver.net.au` is the identifier for the service host.
- `hdl/` is the identifier for the resolution service.
- `102.100.272/0N8J991QH` is the identifier to be resolved (the resource targeted by the service request).

Note that in this example the fact that the identifier is a Handle is inferred by the resolution service and not specifically encoded.²

The service request binds the identifier to one particular action but the same identifier can be associated with more than one action. Other service requests may also be constructed, with different service instances and different actions, for example:

- `http://hdl.handle.net/102.100.272/0N8J991QH`
- `http://openurl.example.com?url_ver=Z39.88-2004&rft_id=info%3Ahdl%3A102.100.272%2F0N8J991QH`
- `http://oai-pmh.example.com?verb=GetRecord&metadataPrefix=oai_dc&identifier=info%3Ahdl%3A102.100.272%2F0N8J991QH`

Note that in these three examples the identifier for the host and the resolution service are combined. In the first example, the same action is requested by invoking the global Handles Registry and the Australian equivalent – resolution through a local Handles service to the identified object and/or conditions of access. In the second example, an OpenURL service is being requested to resolve to the identified object and to negotiate transparent access for authorised users. In the third example, an OAI-PMH service is being requested to return metadata describing the object encoded using DCMI Metadata Terms and transferred using the OAI-PMH protocol. The protocol is implied by the service. Additional parameters specify what is to be returned and the required data syntax.

2.2 Service response

The resolution service actions the request according to instructions implied by or explicitly specified through parameters in the request protocol.

The response as we have defined it could be any information distinctive to the thing identified: Resolve is deliberately a cover term for a range of behaviours. This can include:

- A unique description of the thing.
- Information about how to access the thing (e.g. network location)

² `http://`, `resolver.net.au`, and `hdl/` can each be regarded as identifiers themselves; for each, the context of the name is implicit in the service. So the service request is actually a serialisation of several names for components of the request.

- A representation of the thing itself (which is properly Retrieval)

Generally, when a person actions a cited identifier, the default behaviour will be to resolve to a network location. Other systems such as rendering services will use resolution services to get an appropriate dissemination information package for the task at hand.

3 Implicit and Explicit Services

3.1 Default resolution behaviours

The differentiation between an identifier and a resolution service request on an identifier is unfamiliar to contemporary users of the Web, as URLs have historically conflated the two, using the locator as a non-persistent identifier. The expectation on the Web is therefore that any presentation of an identifier (as a URL) is actionable and that there is only one action possible: retrieving the thing identified, whether a downloadable file or a rendering service.

This is already not the case for http: URIs (which are most strongly associated with retrieval):

- They can be used without any resolution implied, e.g. as labels for XML namespace declarations, as in `<?xml:namespace ns="http://purl.org/dublin_core/schema/" prefix="DC"?>`). But users still assume that they can get to an informative file at the end of such URLs.
- They can be actioned through more than one service; for example, OpenURL could take an http URI: (and not just a Handle), to resolve to an appropriate copy of the resource, or an abstract, or an interlibrary loan request, instead of resolving to the network location specified in the URI.

Nonetheless, the expectation of users and software that URIs online will resolve is too powerful to ignore. It can be addressed in three ways

- Have a default resolution behaviour defined for any presentation of the identifier. This is the approach of PURL and ARK. The default can be overridden with an explicit service call, as is the case for ARK (the "?" and "??" operators).
- Have a context-dependent resolution behaviour defined for any presentation of the identifier. This is the policy now recommended for URI: URIs can be presented as resolvable through HTTP, but need not be, and the behaviour can be defined on a case-by-case basis.
- Differentiate actionable presentations of the identifier from presentations of the identifier in isolation, making clear to the user which presentation they can actually click on. This is the policy PILIN has chosen to deal with Handles, presenting both an identifier `hdl:102.100.272/1KKBLPDQH` and an actionable identifier, binding it to a resolution service, in `http://resolver.net.au/hdl/102.100.272/1KKBLPDQH`. This approach can lead to complexity, and PILIN allows for only one of the two presentations to be used in particular contexts.

Likewise, the expectation of users and software that there is only one possible resolution service for an identifier can be addressed by:

- Presenting the various available resolution services explicitly to the user. (e.g. present an Abstract, ILL, etc. link for the resource without privileging the Resolve to Resource link as the only "real" resolution of the identifier.)
- Avoiding presenting identifiers as strongly bound to one resolution behaviour. (An example of that presenting identifiers exclusively as http:

URIs, as opposed to labels which may be embedded in an http: URI. A Handle does not suffer as much from this strong binding.)

3.2 Encoding additional parameters

A generic syntax has been defined for URI queries (RFC 3986 §3.4) that enables additional parameters to be encoded if explicit behaviours need to be requested, and delimits them from the identifier for the resource. The OpenURL and OAI-PMH protocols both use this syntax to support explicit resolution service requests. For example:

- `http://openurl.example.com?url_ver=Z39.88-2004&rft_id=info%3Ahdl%3A102.100.272%2F0N8J991QH&svc_id=http:%3A%3Atlf.edu.au%3Asvc%3Aabstract : "Give me the (tlf.edu.au) Abstract of info:hdl:102.100.272/0N8J991QH, under the Z39.88-2004 version of OpenURL."3`

Some resolution services encode action parameters as extensions to the identifier. For example, in ARK the "?" operand, appended to a resolution service request, presents metadata describing the thing identified, instead of retrieving the thing itself:

- `http://ark.nlm.nih.gov/ark:/12025/psbbantu : "Retrieve the thing identified by ark:102025/psbbantu, via the ARK resolver http://ark.nlm.nih.gov"`
- `http://ark.nlm.nih.gov/ark:/12025/psbbantu? : "Get metadata about the thing identified by ark:102025/psbbantu, via the ARK resolver http://ark.nlm.nih.gov"`

But this makes it difficult to separate the identifier from the required behaviour, when there is a need to action an identifier through other services that may not understand the conventions used. Any system dealing with ARK identifiers, for example, needs to know the ARK convention that "?" cannot be part of an identifier name, and is instead a service parameter: there is no such identifier as `ark:102025/psbbantu?`.⁴

The kinds of additional parameters that might be encoded in a resolution service request will depend on the objects being identified and the workflows needing to be supported. Examples include:

- Return metadata describing the identified object in a specified data schema.
- Return a dissemination information package representing the object in a specified packaging schema.
- Return a dissemination information package representing the object at a specified date.

³ The extra parameters may themselves be identifiers, and will not necessarily be presented in the same way as they would in isolation. In this example, the Abstracting service has a URL, <http://www.tlf.edu.au/svc/abstract>. But as a parameter in an OpenURL service request, the URL is treated as an identifier, and is URL-encoded: the user is not supposed to be able to resolve it separately.

⁴ If we were migrating ARK to another system, we might exploit this potential for ambiguity, and create a new identifier `ark:102025/psbbantu?` to deal with metadata requests, as a digital object distinct from `ark:102025/psbbantu`. But this is certainly not what the original identifier scheme intended.

- Limit the dissemination information package only to files of a given use (for example, master, thumbnail, print).
- Limit the dissemination information package only to files needed to render the object for delivery.

Having well-understood encoding conventions for common workflows will facilitate machine-machine resolution service requests when more than the default behaviour is required.

It will also enable different representations of an object to be requested without each representation needing to have its own published identifier. The identifier combined with the action parameter serves to identify the required representation through an exposed, well-defined service. This increases the value of the identifier for the object as a whole, allowing it to drive a range of business processes.

4 Persistence of Services

If an identifier is persistent, users expect that all presentations and qualities of the identifier are also persistent. So a service request for resolution of a persistent identifier should itself be persistent, using persistent identifiers for the service and the service host. But the service may be managed outside the identifier management system; so the guarantee made that the identifier is persistent may not apply to the resolution services acting on it. This is a problem: if the identifier is cited in a service request (e.g. a hyperlink to a resolution service), and the service goes away, the persistence of the identifier has not helped much. The separate guarantee of persistence of resolution is strongly emphasised in the design of the ARK identifier scheme; it is also a concern in Handle, as branded, institution-specific resolution services have emerged.

So identifier managers whose persistent identifiers are resolvable should arrange for the resolution services to be persistent themselves: they should make the persistence of the services accountable the same way that the persistence of identifiers is made accountable. The same general principles for persisting a service apply as for persisting an identifier: meaningful and branded names should be avoided; there should be handover arrangements in case the current host of the service becomes unavailable; a large institution or consortium of institutions provides a better guarantee of longevity than an individual or small institution; and so forth.

Any identifier can be acted on by a third-party resolver out of the control of the identifier management system. The identifier management system cannot vouch for such resolvers, and certainly cannot guarantee their persistence or accuracy. So the identifier system should also expose the fact that its resolution service is canonical, and allow users access to mechanisms to verify accurate resolution (such as validation services).

5 Appendix

- RFC 3986: Berners-Lee, T., Fielding, R., Masinter, L. 2005. *Uniform Resource Identifier (URI): Generic Syntax*. RFC 3986. <http://www.ietf.org/rfc/rfc3986.txt>
- Kunze, J. 2007. *The ARK Persistent Identifier Scheme*. <http://www.ietf.org/internet-drafts/draft-kunze-ark-14.txt>