Persistent Identifiers

Hennie Brugman
Technical coordinator CATCH plus project[1]
Max-Planck-Institute for Psycholinguistics, Nijmegen, Netherlands
Institute for Sound and Vision, Hilversum, Netherland

2009-06-03

Definition of the problem

Archive systems and other information systems at heritage institutions contain many objects that require reference, which is why these objects have a name or number: an *identifier*. This may concern physical or digital objects. Catalogues contain metadata descriptions, which refer to objects using identifiers. Metadata descriptions often contain references to concepts in a vocabulary ('thesaurus terms'). These concepts also have an identifier. It may also be desirable to provide the metadata descriptions themselves with an identifier. Other objects which require an identifier handled by Digital Libraries are annotations and user profiles.

In practice, many different identifier systems are used within heritage institutions. These identifiers are usually only *unique* within the context of one specific information system, and sometimes this is not even the case. In an increasing number of cases, identifiers are unique within the context of an institution. Also, identifiers of objects tend to change quite often, for instance if data are exported or migrated to a new database management system. This means that, in many cases, identifiers are not *persistent*, which becomes a problem if references are made from another location using such an identifier. Within the daily practice of data management and archiving, solving these problems requires large amounts of time and money.

The increasing importance of the Internet, as well as collaborative projects such as Europeana, intensify the problem: if objects are referred to from outside the institutions, how can the institution guarantee that the identifiers used are unique and persistent? If institutions start to develop and provide data and services in collaborative projects or as a consortium, how can these external data be referred to from within your own institution in a reliable and sustainable manner without endangering your own company processes or services? And how are objects that are relocated from one institution to the other to be handled?

First of all, proper agreements must be made about guidelines and responsibilities relating to the attribution and management of identifiers. Secondly, the execution of such agreements can be made as simple, reliable and efficient as possible using supporting technology.

This document is an initial start to formulate guidelines with reference to the heritage domain and applicable in Digital libraries, and suggests matching technological solutions.

General solutions

First and foremost, the introduction and use of persistent identifiers is an organisational task. In the second place, it is a technical problem. Every solution involves continuous administrative tasks and therefore costs. If these administrative tasks are not managed properly, no technological solution will work.

Choice of policy

The introduction of persistent identifiers starts with making agreements about a number of subjects in mutual consultation. These agreements must be documented and made transparently available to all parties involved. Questions to be answered include:

What objects must be associable with an identifier?

[1] Catch plus: continuous access to cultural heritage plus http://www.catchplus.nl/

Is there a distinction to be made between physical objects and their digitalised representations? Are metadata records to have their own identifier? Do we want to be able to identify composite objects? Queries? Users? Concepts and alignments? Should we be able to refer to segments of digital objects? To annotations? And vice versa, which objects do not require the use of persistent identifiers?

What is the format of an identifier?
What characters are permitted? Are they also intended for human inspection? Are they permitted to contain meaningful information? If so, which information and which not? For instance, may the name of the organisation of origin be included? Are there any identifiers within participating organisations that can or should be included? Will we be generating identifiers automatically?

What persons and organisations are responsible for the attribution and management of identifiers?
Who are permitted to publish new identifiers? Who are permitted to change these identifiers (associate them with a new URL)? Who are permitted to delegate these rights? Who deals with the management of users and groups? This typically involves tasks which are part of (digital) collection management.

When is an object to be designated a new identifier?
This involves management of versions and identifying variants of a digital object (such as HTML, XML and PDF versions of the same object).

How to handle the relocation of objects or collections between institutions?
Do these objects receive new identifiers (preferably not)? What procedures can be followed?

Persistence
Persistence has a number of aspects: firstly, identifiers are to remain unchanged over time. Secondly, identifiers must remain resolvable (corresponding locations must be known). Thirdly, it must be possible to find identified objects at the locations indicated by the resolver. And finally, an identifier must permanently identify the same object through time. For all of these aspects, the participating organisations have to issue guarantees with regard to persistence.

Hosting
Will we be hosting identifiers ourselves, and if so, who takes responsibility with regard to the management and availability of corresponding services? Will one institution handle the hosting, or several? If not, what central authority will do this? What extent of supervision of our own identifiers do we require? What guarantees can this authority provide? An alternative is the periodical harvesting of identifiers. (Note: hosting identifiers is not the same as managing identifiers.)

Technological support

Currently there are a number of more or less widespread solutions available for supporting persistent identifiers technologically. The most relevant and best known are URN-NBN[2], Handles[3], DOI[4], PURL[5], and ARK[6].
With the exception of URN-NBN, all of these solutions have an accompanying *resolver* architecture (there are resolver solutions for URN-NBN, but these differ from country to country). The basic idea behind a resolver is that all identifiers for a certain 'Naming Authority' (NA) are stored together in one (virtual) repository under the management of the NA. Additional information is stored for every identifier, at least the location(s) where the object belonging to an identifier can be found. Such a location has the typical shape of an URL. The major advantage of such a solution is that identification and location are separated from one another: if the location(s) of an object is/are changed, the only thing that has to be changed is the table in the resolver, all references to the object use the identifier and thus automatically remain valid.

---

[2] http://www.ietf.org/rfc/rfc3188.txt

[3] http://www.handle.net

[4] http://www.doi.org/

[5] http://purl.org/

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

Services a system like this can fulfil for persistent identifiers and other possible properties include:

1. The automatic generation and/or validating of (worldwide) unique identifiers
2. 'Resolving' identifiers
3. A 'Naming Authority' can be explicitly associated with a certain domain. The responsibility for the attribution and management of identifiers is explicitly filed at an institution or consortium.
   The NA
   a. will enter into commitments regarding the guarantee of persistence of identifiers
   b. determines the management of the PI-URL associations
      i. What do PIs look like?
      ii. What do they identify?
      iii. Who owns rights to change them?
4. Persistent identifiers can be associated with metadata (for instance, the PI manager, a description, an e-mail address or a statement with regard to guarantees for persistence)
5. Updating location(s) associated with a PI by authorised persons
6. Finely-woven solutions for organising redundancy (by mirroring, for example)
7. Identifiers continue to exist even after identified objects have disappeared
8. The history of a persistent identifier can be recorded.

Demands and requirements concerning heritage domain and digital libraries

Following, a number of requirements for persistent identifier solutions within the context of Digital libraries are stated and briefly explained.

General

Preferably associate with existing initiatives for distribution and management of persistent identifiers

Currently, two running initiatives in the Netherlands can serve as exemplary: national libraries in Europe (including the Koninklijke Bibliotheek (Dutch Royal Library)) are working together in the implementation of persistent identifiers on the basis of URN-NBN. Within the Dutch context (and in consultation with other European parties) DANS[7] has implemented a resolver solution. Currently, there is commitment only with regard to the provision of resolver services within the context of SurfShare. At this stage, this commitment cannot yet be fulfilled because of missing (redundant) server capacity.

Within Europe, people are currently working on an independent and redundant identifier solution for eScience based on Handles. Two organisations have committed themselves to this already (German Max-Planck-Gesellschaft[8] through the GWDG[9], and the Finnish CSC[10]). A few more partners are being sought, including one in the Netherlands.

Making use of existing and tested technological solutions(s)

Currently, the most common solutions are URN-NBN, Handles, DOI, ARK and PURL.

Preferably a homogenous solution for the European Digital libraries

As far as known, at this moment a solution for persistent identifiers in the Netherlands is used indoors at the Koninklijke Bibliotheek only. This provides a good opportunity to implement a homogenous solution jointly, which is beneficial to interoperability and makes sharing knowledge and experience simple.

Including with regard to other and already existing persistent identifiers

Institutions have their own demands and requirements and also participate in other collaborative projects. Eventually, the normal situation will probably be that there are various solutions for PIs next to one another. A

[7] http://www.dans.knaw.nl/
[8] http://www.mpg.de/
[9] http://www.gwdg.de
[10] http://www.csc.fi/

(homogenous) solution, which will be chosen by the heritage and/or digital library community, must therefore be compatible with other PI systems.

Commitment with regard to persistence by long-term existing institution(s)
PIs are persistent only if an organisation guarantees this persistence. The organisation must thus at least be durable (large heritage institutions are pre-eminently suited for this and may even have a natural task here).

Suitable for application by museums, libraries and archives
There may possibly be additional demands for general PI systems from the use within the heritage and/or digital library sector.

Compatible with Semantic Web, with Linked Open Data in particular
Semantic Web applications are becoming more and more important for heritage domain and digital libraries. Within the SW community, objects are identified by means of HTTP URIs (Uniform Resource Identifiers). In addition, there are guidelines with regard to resolving these URIs within the Linked Open Data[11] initiative. It is important to remain compatible with LOD. This also means that it must be possible to represent persistent identifiers as resolvable HTTP URIs. Selected text representations for identifiers must be 'URL safe', because it must be possible to use them as (part of) URLs. (Comment: URN-NBN does not comply with this, but the DANS URN-NBN resolver does). The 'host' part of the HTTP URI must be a stable name, because it is considered part of the identifier within the SW community. In addition, it is an advantage if a PI system allows for the implementation of 'content negotiation': depending on the desired content type required by the user, the resolver can return another URL location (for example: URLs for html/skos/xml/json representations of a concept from a vocabulary)

Organisational
Reliable and redundant
Services surrounding persistent identifiers (resolving in particular) are essential for localising data, so it is of extreme importance for these services to be reliable and quick. A PI resolver may not become a 'single point of failure', so redundancy is essential, using PI of mirror sites, for instance.

Limited costs of use
It is not the intention to become dependant on a specific supplier or to be confronted with periodic licence fees. All current PI technologies are based on open source software, so there are no licence fees. Substantial costs for servicing or costs per identifier attributed might jeopardise the use of PI's. For substantial numbers of identifiers, as is the case in heritage collections, DOI is an alternative that is not free of charge as DOI charges per identifier.

Division of responsibility and costs for management
This is a joint venture in which arrangements will be made about the division of management efforts and costs involved. The selected solution is to support this, for instance by the explicit recording of users/administrators, user groups and their rights regarding certain management tasks.

Identifier management separated from Webserver management and hosting of identifiers
Management of identifiers, providing storage and services surrounding identifiers and managing web servers are very different tasks so in principle, various persons, and possibly various organisations, must be able to execute these. It must be possible to manage identifiers without the intervention of a system or web server manager, the person who manages these services around identifiers does not have to be the owner of these identifiers.

Consortium-wide pool of persistent identifiers
The ideal is to reach one shared pool of persistent identifiers for the entire heritage and Digital Library community. In principle, this pool (provided it is redundant) may be hosted at various physical locations.

Minimum management effort required
It goes without saying that management instruments must be as efficient and user-friendly as possible.

[11] http://linkeddata.org/

Agreements about policy with regard to persistence must be recorded explicitly wherever possible
Sometimes, PI systems even provide built-in options for this, as is the case with ARK, in which a 'promise of stewardship' can be associated with every identifier.
Open for other heritage and digital library institutions
In principle, the infrastructure developed for PI is open for use by heritage and Digital Library institutions. It must be clear under which conditions this is possible and what is required, technically and organisationally.

Institutional independence
Heritage and digital library institutions have many collaborative projects. Implementation of persistent identifiers in one consortium should not interfere with collaborations within other connections.

Technical
Granularity
Within heritage and digital library collections, identifiers are required for various types of objects. Sometimes large numbers are involved, such as in the case of annotated text documents, where (semantic) annotations can be linked to segments in the text. Possibly, each segment must be identified. This can involve hundreds of segments per document. In other words, this makes great demands on the capability to resolve large numbers of identifiers (1 identifier per annotation), or this requires that the resolver can handle fragment identifiers (1 identifier per document) transparently.

Scalability
In view of the fact that the future use and magnitude of the persistent identifier services is not as yet known, the solution has to be scalable. This scalability has two aspects: scalability with regard to the number of identifiers, and performance.
Means to make solutions scalable include:

- Hashing: method that allows identifiers to be divided among various servers and to efficiently specify the server where a certain identifier is located.
- Caching: temporary local storage of previously 'resolved' identifiers.
- Replication: makes it possible to use various sites for the same pool of identifiers.

Reliability
In a technical sense, this can be achieved by applying replication (mirroring). One variant is the resolver solution as realised by DANS: persistent identifiers are offered by the providers of collections by means of OAI-PMH[12] and then periodically harvested by the central resolver.

Centrally registered Naming Authority
For various reasons, it is important to register the Naming Authority for a certain collection of identifiers at a central authority. Resolver(s) are thus traceable worldwide for a certain identifier, it guarantees that the identifier (combined with a unique ID for the NA itself) is unique worldwide, and ensures that redundancy and caching can be dealt with.

Metadata
In principle, metadata belongs in a separate catalogue and therefore requires minimal support from the PI system. The following may be worthwhile: administrator/owner of the identifier, contact information, description of the identified object or a statement relating to persistence.

Requirements from application scenarios?
Further analysis is yet to be carried out, which is to provide clarity about the numbers of PIs expected and types of objects to be identified (user profiles, docs, concepts, annotations, physical objects…). There may also be requirements with regard to performance.

It must be possible for persistent identifiers to be modified per group

[12] http://www.openarchives.org/OAI/openarchivesprotocol.html

Collections and other data collections are sometimes relocated as a whole, in which case it must be possible to update all accompanying persistent identifiers in one single transaction. This applies to relocations both within and between heritage institutions/digital libraries.

Demands to client tools and software libraries
There are various ways to make use of the services of the PI system: via tools (applications), via web front-ends or applets, via client software libraries, via web services, on a protocol level (private protocol, such as for Handles, or HTTP), from standard web browsers. For management by heritage employees, access by means of applications and/or web access to services is required (for resolver management and use). For software developers working for Digital Libraries access via software libraries or web services is of importance (mainly for resolver use).
Finally, it must be possible to resolve persistent identifiers by means of a standard web browser. (This implies that a HTML representation of the associated URL(s) or of the data attributed by this/these URL(s) must be available.)

Safety and encryption
In some cases, a PI framework provides support for encryption, digital signatures and server authentication (such as Handles). It has to become evident from the heritage and Digital library community that there is a need for this.

Solution proposals

Evaluation of technical solutions

Possible alternatives for services regarding identifiers are: making use of standard web technologies only (HTTP, URI, DNS), Handles, DOI, PURL, URN-NBN and ARK. With the exception of URN-NBN, all of these alternative persistent identifiers can be resolved by means of a (HTTP) URI. In each of these cases, this URI follows the same pattern of a 'service request': it consists of a host-id, a service-id, and the persistent part, the actual identifier.
Examples of these service requests:

HTTP URI:	http://www.beeldengeluid.nl/gtaa#Subject_aalscholvers
PURL:	http://identifiers.erfgoed.nl/purl/vocabularies/iconclass/concept1821 or http://purl.org/vocabularies/iconclass/concept1821
Handle:	http://identifiers.erfgoed.nl/hdl/1280.14/local_id_1821
ARK:	http://identifiers.erfgoed.nl/ark:/128014/local_id_1821
URN-NBN:	urn:nbn:nl-local_id_1821, plus resolver: www.persistent-identifier.nl

The accompanying subdivisions in host-id, service-id and identifier:

Host ID	*Service ID of the person registered*	*Naming Authority*	*Actual identifier*
http://www.beeldengeluid.nl	-	www.BeeldenGeluid.nl	gtaa#Subject_aalscholvers
http://purl.org http://identifiers.erfgoed.nl	 Purl	Vocabularies -	iconclass/concept1821 iconclass/concept1821
http://identifiers.erfgoed.nl	hdl	1280.14	local_id_1821
http://identifiers.erfgoed.nl	ark:	128014	local_id_1821
	-	Nbn:nl-	local_id_1821

In principle, it is possible to resolve various types of identifiers on one host using HTTP.

DOI is frequently used in the publishing world. Technically, it is a specific application of Handles, and DOIs can be resolved via Handle resolvers. DOI requires payment based on the number of identifiers attributed. In view of the fact that, within Digital Libraries community, large numbers of identifiers may be involved, we will leave DOI aside based on cost considerations.

In principle, standard URIs can be used as persistent identifiers, in which case resolving is realised by a combination of domain name resolution (via DNS) and 'HTTP redirection'. The major advantage is that tested, standard Web technology is exclusively used. However, this solution does assume considerable discipline in making agreements

and applying guidelines that are to guarantee persistence[13]. In addition, there are no necessary services for managing identifiers. Web server managers can solve some management tasks at the most[14]. In addition, this solution is not flexible because identifier management, hosting of identifiers and web server management cannot be disconnected. We thus conclude that the use of standard web technology alone is not sufficient. However, the requirement remains that it must be possible to use chosen solutions for persistent identifier services via standard web technology (via HTTP and DNS, represented using HTTP URIs, according to cool URI and Linked Open Data best practices, URL-safe encoded).

Despite the fact that in the Netherlands URN-NBN is used as a basis for collaboration in infrastructure for persistent identifiers (also by SURFShare, DANS and the KB), it does not comply with a number of our requirements: URN-NBN identifiers have no URI representation and are not URL safe. By definition, Naming Authorities correspond to national libraries, which is not flexible enough for our profiles. Furthermore, there is no software included to realise services surrounding URN-NBNs. These services are a responsibility for each separate national library. For the Dutch implementation of URN-NBN services, it seems that, as of yet, it is difficult to guarantee reliability, performance and redundancy. For now, we will leave URN-NBN aside as a solution.

Remaining technical solutions which more or less comply with our demands and requirements are PURL, Handle and ARK. PURL and Handle are tried and tested platforms which are being applied on a large scale for over a decade. Of these two, Handle is absolutely best where performance, scalability, distributed set-up, authentication and authorisation, security and available software tools and libraries are concerned. Our proposition is therefore to support both PURL and Handle identifiers, with Handle as the basic solution. Where URN-NBN and ARK are concerned, a resolver may 'redirect' to the external resolvers concerned.

Handles

The Handle System (http://www.handle.net, by CNRI) is a worldwide and commonly applied distributed system for identifiers and resolving. Since the mid 1990s, it has been used by a number of universities, national libraries, government institutions, computer centres and companies. The largest and best-known user is the International DOI Foundation, which handles identifiers for the international publishing sector[15].

Handles are identifiers with a very simple syntax: 'prefix/suffix', an example: '10.1045/april2006-paskin'. The Handle System consists of two layers: a global service called Global Handle Registry and Local Handle Services. Each LHS corresponds with a registered Naming Authority and has a unique prefix (€ 50.00 per year is charged for the registration of a prefix). The suffix part can be used according to one's own judgement of this NA. The Global Handle Registry contains special Handles which associate prefixes with the locations of Local Handle Services. Local Handle Services manage and resolve their own identifiers. Every LHS has access to all Handles worldwide via the Global Handle Registry in no more than two steps.

The Handle System is especially designed with a view to performance and scalability. It has built-in options for Local Handle Services to configure mirrors, to divide identifiers over various servers and to temporarily store identifier information locally (caching).

Handles can be resolved via a private handle protocol, and in addition, all Handles can be resolved via an HTTP proxy server (such as via http://hdl.handle.net/4263537/5555). The Handle software also supports resolving via HTTP by local Handle services (such as http://identifiers.erfgoed.nl/hdl/4263537/5555).

Finally, Handles will soon provide support for fragment identifiers also (part of a URL behind a # character, which is used to indicate part of a resource). The Handle System first strips the fragment identifier from the URL, resolves the remainder and pastes the fragment ID behind the results. This is an important feature for the use of identifiers for annotations, which are linked to parts of digital objects.

---

[13] http://www.w3.org/TR/cooluris/

[14] http://www.w3.org/TR/swbp-vocab-pub/

[15] http://www.doi.org/

Profiles

Four 'management profiles' with regard to the management of persistent identifiers are introduced in the (very readable) documents of the Australian PILIN project[16]. A distinction between these profiles is made along the 'hosting of identifiers' dimensions (central versus private hosting) and 'management of identifiers' (private management versus joint management). A quotation:

> 'The default profile is *devolved management*, where each party manages their own identifiers and identifier systems (own hosting, exclusive management).
> *Centralised systems* introduce an economy of scale, and relieve parties of administrative burdens; but they also take away much of the ownership of the identifiers being pooled into the centralised system (central hosting, shared management).
> *Autonomous systems* address this problem by uncoupling hosting from identifier management (central hosting, exclusive management).
> *Federation* addresses the problem in a different way, restoring ownership through a shared consortium—but this introduces its own administrative burden at the consortium level (own –federated- hosting, shared management).'

The most suitable profile for Digital libraries largely depends on the preferences and preconditions of the participating heritage institutions..

Three alternative proposals are presented below which follow the 'autonomous systems', 'federation' and 'centralised systems' profiles.

Alternative 1: autonomous systems

In this model, the storage of identifiers and the provision of services take place centrally. In principle however, management of identifiers is executed by each of the participating heritage institutions themselves. Concretely, this means that there will be more then one 'naming authorities', each with their own registered Handle prefix. A project bureau deals with central hosting and identifier services for the three Local Handle Systems.
Redundancy, reliability, performance and persistence of the services have to be provided. In order to achieve this, suitable partners are being sought. In principle, there are three resolvers (which we call http://identifiers.musea.erfgoed.nl, http://identifiers.bibliotheken.erfgoed.nl, and http://identifiers.archieven.erfgoed.nl). Each of these three can resolve Handles (such as http://identifiers.musea.erfgoed.nl/hdl/4263537/5555) and PURL identifiers, if required. Each of the naming authorities attributes an 'identifier manager' who is responsible for the correct management of their own persistent identifiers. In principle, each institution pursues its own policy regarding naming and management.

Alternative 2: federation

In this model, participants act more as a consortium: a joint pool of identifiers is set up according a shared policy with regard to management and naming. This total pool of identifiers is hosted by various partners: each partner hosts their own identifiers as well as those of the others, resulting in redundancy.
In this case, the consortium corresponds to one naming authority with one registered Handle prefix. Only one Local Handle System is set up, which, for example, consists of a number of Local Handle sites, which form mirrors of each other. One shared resolver is thus constructed with the name http://identifiers.erfgoed.nl, for instance. Again, Handles and PURL identifiers can be resolved.

As in the above alternative, three 'identifier managers' are appointed, but in this case there is a mutually agreed policy. As a part of the shared policies, it is still possible to make distinctions according to heritage institution in the Handle suffix, though this may be less desire for this. (http://identifiers.erfgoed.nl/hdl/12345/rma-678)

Alternative 3: centralised systems

In this alternative, hosting and servicing and part of the management policies are put out to a central authority, where the interests of the participating parties are served by representation in a coordinating administrating authority. In concrete, currently a consortium of large academic computer centres is being developed (at this time with intended partners in Germany, Finland and the Netherlands) for the benefit of eScience. This consortium will provide Handle

[16] https://www.pilin.net.au/Project_Documents/Community_Guidelines/Guidelines.htm

services for Europe in an autonomous and redundant manner, and will run mirror for the Global Handle Repository. Agreements in principle have already been made with American CNRI.

This solution is comparable to DOI, where the publishing world provides additional services on top of the Handle System, on the understanding that a different business model will be used, in which charging should not take place per identifier attributed.

In this case, heritage identifiers would be resolved by something like http://escience.eu/hdl or a Dutch equivalent of http://handle.gwdg.de. An identifier could look as follows: http://escience.eu/hdl/[vaste-prefix]/CH-NL-[identifier]. In this case, agreements would have to be made with the central authority concerning the registration of identifier managers of the heritage institutions.

Additional choices

Naming
With regard to choosing text strings for identifiers, it is recommended not to include elements which may change over the course of time; this means the exclusion of semantics and technology dependencies.
In addition, the real local identifier must be preceded by a unique ID of a registered Naming Authority (Handle prefix, ARK Name Assigning Authority Number, PURL path name, URN-NBN ISO country code).

PURL resolving
Two manners of support are possible for this. First, a private domain can be arranged on purl.org, and below that, centrally, PI-URL associations can be managed. The second option is to install a private local PURL resolver for our PI-URL associations, and to globally 'register' it at purl.org (by means of a 'partial redirect'). In this latter case, http://purl.org/erfgoed-nl/XXX can first be resolved to http://identifiers.erfgoed.nl/purl/XXX, and then to the URLs associated with this. The last option can be implemented if this becomes desirable because of magnitude and performance.

URN-NBN support
It should be possible to redirect everything beginning with http://identifiers.erfgoed.nl/urn:nbn:nl to the DANS resolver http://www.persistent-identifier.nl, for example. It must, however, be guaranteed that everything takes place in a 'URL-safe' manner.

Version management
A general identifier for the object refers to the latest version; separate identifiers may be used for every version for which it must be possible to be localised separately.

Metadata
The aim is to associate minimal metadata with identifiers: only those metadata that serve to identify and localise objects. Descriptive metadata belong in catalogues or other metadata repositories.

Possible Services
Additional services can be developed

- Automatic generation and/or validation of new identifiers
- Content negotiation: service requests for Handle (and possibly PURL) content negotiation supported. It must be possible for various associated URLs to exist for a certain identifier, which can be distinguished by type. Handles have the technical provisions for this, but for PURLs, this can possibly only be realised by using a modified local PURL resolver.
- Updating identifiers by group.