Report of the CUL Working Group

April 21, 2006

Executive Summary

A group of technical people in the Cornell University Library (CUL) system evaluated several solutions for managing persistent identifiers for the digital resources in three of CUL's digital collection projects. After having compared those solutions' capabilities with the group's list of requirements, the members of the group created a list of recommendations for a CUL strategy, including the adoption of the CNRI Handle System for a test implementation.

The group and its charge

Bill Kehoe and Rick Silterra conceived of the need for this group while discussing their desire for a consistent strategy for global and persistent identifiers (PIDs) for digital objects. They recognized the integral role that PIDs would play in the Integrated Framework, CUL OAIS, and MathArc projects. They believed the best way to achieve consensus for a strategy would be through a lightweight, focused, and efficient evaluation process. They hoped to assemble a small group of people designing systems that use identifiers, who could vet the different approaches to PIDs with the goal of finding the most appropriate PID strategy for use on these projects. Marcy Rosenkrantz, Nancy McGovern and Oya Reiger proceeded to help organize a group of people to carry out the evalutation process. It included (in alphabetical order):

- Adam Chandler
- John Fereira
- Bill Kehoe
- George Kozak
- Rick Silterra
- Adam Smith

During its initial meeting in early February, 2006, the group, along with Marcy and Oya, clarified its charge and the approach to be taken. The group was to meet weekly for a planned period of four to six weeks to review PID approaches, ending in a "retreat" in which recommendations would be formulated and ultimately presented to stakeholders within the library, particularly the Digital Content Delivery Platforms Forum, and the technical team working on a digital preservation system for CUL.

Problem statement

- Digital collection identifiers
    - A long standing issue in the presentation of our digital collections is the lack of a reliable identifier
      for the collections themselves. Collections move to new machines, collections change their delivery platform,
      and collections change their default behaviors. We need to be able to locate collections reliably over a long period of time and discover aspects of collections for interoperating with other collections.

- Persistent IDs for digital preservation
    - We are working on preserving the digital objects in two large collections, the Euclid journals and the arXiv.org preprints. In general, we can view the objects as having at least one content file and an object descriptor file containing metadata about the object. Most digital objects contain multiple content and metadata files. We need to be able to identify and locate the files for a long time, regardless of where they are located. Rather than changing the metadata in the descriptor file every time a file is moved-we would like to create persistent identifiers that can be mapped to the files' current locations.
    - The number of component files to be preserved will be several times larger than the number of digital objects. With processing efficiency in mind, we would prefer a solution that will allow us to resolve the identifiers locally, without going out over the internet for each request for resolution.
    - The digital objects' component files in our preservation system will not be directly accessible to the public; access will occur through a gated interface. The persistent identifier mechanism we use should be able to be produce identifiers that are private and not discoverable.

Requirements for an implementation

- We don't want to break any system currently used at CUL that uses persistent identifieres, such as the PURL server. It should be backward compatible.
- Every PID should be globally unique.
- PIDs must be globally resolvable, resolvable from anywhere.

- On the other hand, some PIDs must be optionally resolvable only within a constrained environment. In other words, the system must have the ability to make PIDs private, not discoverable. For example, an archiving system should be able to restrict access to low-level components of complex digital objects.

- The system should be able to ensure confidentiality. It should define a mechanism for client authentication and authorization to ensure data integrity and authority control.
- The system should not have dependencies on external systems in order to resolve local PIDs. Reliance on the network outside CUL for every resolution request would introduce potential latency problems, slowing down all our systems.
- PID should be free of location semantics. For example, a file originally stored on a server as "/some/graphic/collection/file1234" should not be mapped to a PID like "cul:graphics_collection_00000001234." If ownership of the file changes, or location changes, or if the file is reassigned to another collection, the PID would have misleading and incorrect meaning. Any relationships among files, collections, and owners should be documented in metadata.

- PIDs must be able to be assigned to OAI-PMH records. Such records must be accessible through persistent links, else they lose their value.

- PID must be able to refer to multiple aspects, attributes, or behaviors of the digital object, but with a default aspect that conforms with convential use. We want the ability to store mappings to various facets of an object. For example, using the PID as a URL would return an html page, perhaps, whereas using the PID in a REST-style request would return a pdf or object metadata.

- The system must allow fine-grained management of PIDs, so that groups can maintain their own sets, without having to maintain multiple PID resolvers/servers.

Methodology

The group began a Wiki that gathered previous research into PID solutions by members of CUL, and outlined a number of those approaches as well as the relevant standards and other documentation for each. In the first meeting, the group surveyed the landscape of existing PID strategies and other approaches to identifying print and electronic resources including the general issues and challenges underlying these strategies. Then, in each subsequent week, the group chose to look at a particular PID strategy, reading the relevant documentation and discussing the advantages and disadvantages of each approach. The group also examined the experiences of any existing implementations of each PID strategy at CUL and elsewhere. Throughout the group's explorations, the original scope and charge were reevaluated in light of the discussions, and the impact that the group's eventual recommendations would have on existing and future CUL projects were considered.
A rough list of topics discussed includes:

- an overview of identifier strategies and issues,
- PURL,
- ARK,
- Handl
- OpenURL, and
- OAI-PMH requirements.

By the end of these discussions, the group began to reach a consensus around Handles based on that approach's general maturity and installed base, its fit for CUL's projects, and the existing knowledge about this approach within CUL. The group then began to plan and write this report on the group Wiki that recommends further exploration of Handles.

Recommendations

- We recommend that the Library use the CNRI Handles System for generating and maintaining persistent identifiers at Cornell.
    - The Corporation for National Research Initiatives (CNRI) undertakes, fosters, and promotes research in the public interest. One of its many projects is the Handle System.
    - The Handle System is a comprehensive system for assigning, managing, and resolving persistent identifiers, known as "handles," for digital objects and other resources on the Internet. Handles can be used as Uniform Resource Names (URNs).
    - The Handle System includes an open set of protocols, a namespace, and an implementation of the protocols. The protocols enable a distributed computer system to store handles of digital resources and resolve those handles into the information necessary to locate and access the resources. This associated information can be changed as needed to reflect the current state of the identified resource without changing the handle, allowing the name of the item to persist over changes of location and other state information. Each handle may have it own administrator(s), and administration can be done in a distributed environment. The name-to-value bindings may also be secured, allowing handles to be used in trust management applications.
    - The system operates by assigning a unique identifier, or handle, to every file sought to be retrieved by the system. The handle and the file's location are then registered with a handle server. The handle server can either be private, on a local network, or public, as on the Internet. When a user requests a document by submitting its handle, the handle server reports the location of the file, which is then retrieved and presented to the user.
    - The Handles System naming convention is sufficiently flexible to allow truly unique names for digital objects. Handles take the form: naming authority/name. Naming authorities are organized in a hierarchy, with each being able to create sub-naming authorities. Document names need only be unique with reference to the issuing naming authority.
- We recommend that CUL undertake a proof-of-concept implementation of the CNRI Handle System. We would hope to test the system's ability to meet all of our requirements and gain insight and skill in the technical and organizational demands of maintaining a persistent identifier system. CNRI provides free handle server software and documentation. We envision a local system that can be administered in a distributed manner, so caretakers of different collections will be able to make autonomous decisions about the identifiers their collections use. We also recognize that a proof-of-concept system might prove to be of limited or no use in the long run. During the period of experimentation we would ensure that the digital objects we give identifiers to would be able to be remapped to their original URLs or some other useful identifer system, in case we decide to discontinue using the Handle System.

- We considered recommending that specific resources be allocated for an pilot implementation, but we could only guess at the requirements. We assume there will be some development cost, a server (perhaps shared-use) with an open-source database, some disk space (much less than collection-size) and some maintenance cost.

- We recommend that the system and its use be evaluated at some time in the future, including the following minimum criteria:
    - Does the system solve the problem that it was designed to solve?
    - Can it be invoked in a batch mode to create and update large numbers of objects for preservation purposes?
    - Does it meet performance goals? Does the handle system impose an acceptable amount of overhead?
    - Is the system actually being used for its designed purpose? Is the administration so difficult that few digital collection curators will accept its use?

- We recommend that, after a period of testing, a Usage Document be written to explain how the system can be integrated into the CUL collection lifecycle. Monitoring how the handles are used will assist CUL to optimize the way in which we make our collections available.

- Finally, we recommend that, embedded in the metadata mapped to each PID, there be an explicit statement of the estimated lifespan of the PID and of the object it represents. This practice will add value to the identifier and help enforce a best practice in the lifecycle management of CUL's digital objects.