Minutes of 22-Feb-2006 meeting

These minutes may be fragmentary as the discussion was moving right along, and
I am not sure I captured everyone's thoughts. Please add anything I have missed,
and correct anything I have misstated.
(rick)
Present George K. (GK) Adam S. (AS) Adam C. (AC) Rick S. (RS) John Fereira (JF)

Our assigned readings were not covered in any coherent way, but served to
start discussion. JF noted that 2 readings, both pretty fat, was an awful lot
to absorb. [rs says: This was my fault - wanting to bring in the webarch reading as
background to see how just plain web best practices might help us.]

1) Adam S. brought up the scope issue. What will we issue when we are done?
The general feeling seems to be that we will issue recommendation(s), but we don't want
them to turn into mandates. Generally, it seems that mandates just do not work due to
the nature of the projects and the nature of our disparate organizations.
RS thought that most resources we are interested in are entered into the catalog, and
JF disagreed.

2) As John, and George were not present at last meeting, we tried to
continue with use cases.

Here are some cases noted by John.
2 main cases:
give a description of a resource to the pid system, and get back an identifier.
give an identifier to a the pid system and get back a description.

John noted several characteristics of our target PID system:
ease of use, standard.

GK discussed how the current PURL server came into existence: mostly needed to point
to databases that move - solve a problem for the catalog. strong points:
easy to use, administrative GUI, a standard, batch tools.

weak point(s): old software, not supported anymore, no API for programmatic use.

GK brought up that we seem to be using most of the systems available:
PURLs, Handles in DSPACE, DOI's in ProjectEuclid.
Not represented is the new kid - ARKs.

Another use case (we brought this up last time).
AC brought up the need to offer persistent identifiers in the OAI context - people
who harvest us want to be able to refresh their information without worrying about what
has moved, or been inserted into a different delivery system.

We talked about the granularity of the PID - given the example of KMODDL -
we point to the collection, then to an object, then to a part of an object -
how does the pid system go about describing the relationships between objects -
given a page image of a book in the KMODDL system for instance, how does it point to
the 'book' it is part of? and likewise how does the book point to the parts that compose it:
pdfs, images and so forth. Fedora stores and can describe these relationships - at some
point GK brought up the fact that Dspace assigns handles to "items" in dspace; those are resolvable
using the handle - bit streams that compose these items are assigned identifiers (handles?) but those
handles seem to not be resolvable by the global handle server (I hope I got this right).
We discussed the distributed nature of handles - the global authority, which assigns each
local node (like cornell) an identifier. Each local node then assigns identifiers.
RS suggested that the distributed nature of handle resolution might make administration of a system
simpler, more scalable for folks that have lots (like millions) of identifiers to manage.
JF brought up the issue of delivering the right page to a user: if they have an persistent
identifier pointing to data that changes daily - then how does the PID system deliver the right resource,
but if it is necessary to get to a particular version, or version as of a particular date - how
does the PID system handle that.

AC brought up the use of 'buckets' in CUGIR to gather together different file formats under one
identifier, and offer the user the choice of format.

AS talked about the priority 6, 'Get It', and their goal of unifying the access to items that we
can deliver - one button that gets the user to a resource, if possible, or requests a resource for them
from the local system, or as inter library loan. AC talked about how the 'direct connect' feature of
web bridge enables this sort of one-click access to resources or requests for resources.
OpenURL was broached as a sort of bridge or translator that might serve to allow a user to fill
in necessary information to refine a persistent identifier, or that might transport a request from a user
context to the appropriate delivery context (I made up these phrases).
There was general feeling/hope/perception that some combination of PID and OpenURL might meet our requirements
for unambigously identifying items, but also tailoring the delivery of the right item for a particular user.
On that note - I was pretty hot, so I suggested we wrap it up.

we decided to do our next reading on handles. AC will have a demo/discussion of web bridge. and our reading after this will be on openURL.

These are my notes as best as I could interpret them. Please correct the notes if I have forgotten or misstated or misquoted anyone.

Rick