Software Architecture Working Group

Minutes of April 17, 2008 Meeting

Agenda

1. Intro and process items
   - brief comments on OR2008 conference
   - creation of an open source management team
2. Review of Fedora 3.0 migration requirements (Triggs, et al)
3. Storage architecture
4. Exporting of ETDs to ProQuest (Ellis, et al)
5. Requirements up next for review
6. Additional items from last meeting
   - open source headers
   - signature failures, handling of
   - admin space specification

Migration to Fedora 3.0

Fedora 3.0 has been installed on lefty64 (several times). Basic API functions are working and disseminators are working through the Fedora generated content models. The following are working assumptions that will be verified through additional testing and prototyping:

CMDA

- All our object architectures will have a content model. There are (maybe) 15 object architectures in total. We can probably use the generated content models as a starting point to develop a content model for each object architecture (book, article, photograph, etc). (Jeffery will develop a book content model as a first test).
- Although Fedora does not require each object to have a content model, we decided that the best working approach is to assume each object has a content model. This simplifies the WMS approach and is a good strategy for preservation of objects.
- The object can point to its content model by using Fedora relationship services (rels-ext) with the relationship of “hasContentModel”.
- The major steps in the migration scenario are as follows: a) To use the migration utility, we will need to migrate from Fedora 2.1 to Fedora 3.0, b) therefore, in the final testing scenario, we install Fedora 2.1 on lefty64, c) pull all objects from mss3 and ingest, d) migrate to Fedora 3.0 where the analyzer output is processed with XSLT to point to proper content models, e) ingest into Fedora 3.0., and f) begin testing of R5.0/Fedora 3.0 on lefty64.

WMS Impact

- For all objects, WMS will need to create the relationship to point to the appropriate content model. This will require the creation of the rels-ext datastream and should be standard "boiler-plate" for each object architecture.
- XACML policies should be external to the object. We should have a small number of policies to which many objects will point. Including the policies as inline xml will likely lead to cumbersome editing processes. WMS will need to have a capability for the user to select an access policy. The xacml will be a datastream in another Fedora object which the WMS references in the resource object as an external datastream with ID equal to "POLICY".
- WMS will need to point to archival masters that use the "external" file management mode in Fedora. See next section.

Storage Architecture

The following are major assumptions regarding the storage architecture and R5.0:

- We will need to treat archival masters for large files using the Fedora "external" mode. In particular, the NJVid release will need this capability for video archival masters. Note that WMS has already been updated to allow any archival master to be external.
- For R5.0, the "external" file system will be local to the Fedora server. In addition to changes for WMS, the signature verifying code will need to find the archival master as a managed datastream or an external datastream. (Note: in a subsequent investigation, it was verified that the file pointer must be a url for http access – a Fedora requirement).
- For external archival masters, the archival master file along with the checksum will need to be positioned in a permanent external file system. The presentation files (e.g. QT and FLV) will continue to be located in the workarea and processed as usual. WMS will upload the presentation files, point to the archival master, and include the checksum in technical metadata. This process requires that the archival master and the checksum be pre-loaded in the external file system.
- The question of whether we need a backend storage server will need further discussion. In particular, we need to be part of the discussion with NJVid as to what storage system they will purchase.

Purchase of an Additional Server

The storage architecture led to a continuing discussion of whether or not we should purchase another server. We concluded that purchase of an additional server should be postponed and the budget invested in updating facilities at TSB and the SCC in order to accommodate additional servers. One specific need for an additional server is for a "read-only" capability that we can switch to when the production server is

out of service. We also agreed that we need a process improvement to be better prepared for how to use year-end budget.

Google Indexing

We concluded that we should go ahead and push the Google sitemaps out even though the urls will have "mss3" in them. We need to take on the task of coming up with a single url for pointing to RUcore objects that does not have the server name embedded in the url.

Possible Agenda Items for Next Meeting

- Requirements Schedule for R5.0 and R5.1
- Export of ETDs to ProQuest (Shaun)
- Continuation of Migration Requirements (Jeffery)
- XACML Requirements for R5.0 (Jie)
- Sitemap url references and associated issues (e.g. read-only system)
- open source headers (Chad)
- signature failures, handling of (Kalaivani, Jeffery)
- admin space specification (Mary Beth)