TRAC Audit

This page will be used to build up our case as part of a TRAC audit.

Governance and organizational viability

We will skip this section for now, though it's important, it doesn't relate to to the talk scheduled for OR2011.

B. Digital Object Management

For much of this, the process will be to identify the informal way we comply, formalize it and move it closer to the repository when possible. The "trusted repository" consists of pre-ingest routines, HPSS, backed-up spreadsheets and other workflow documents on elm, wiki documentation of collections, xubmit, fedora, and our purl resolution service.

B1. Ingest: acquisition of content

B1.1 Repository identifies properties it will preserve for digital objects.

This is largely specified on the wiki, in historical ingest routines in SVN, and increasingly by using better content models.

B1.2 Repository clearly specifies information that needs to be associated with digital material at the time of its deposit (i.e., SIP)

This is largely specified on the wiki, in historical ingest routines in SVN, and increasingly by using better content models.

B1.3 Repository has mechanisms to authenticate the source of all materials

Passwords?

B1.4 Repository's ingest process verifies each submitted object (i.e., SIP) for completeness and correctness as specified in B1.2

ImageProc verifies image integrity.
The ingest routine chokes requiring manual intervention if a critical file for a given collection is missing, misnamed or invalid.
Before any ingest files are removed from disk, they are compared with the data pulled off of fedora.

B1.5 Repository obtains sufficient physical control over the digital objects to preserve them.

Yes because the file system is owned by us. Passwords, not public.

B1.6 Repository provides producer/depository with appropriate responses at predefined points during the ingest processes.

Image processing/HPSS processing sends an e-mail for success and for failure?
Automatic ingest sends out a report.
Users from the lilly, routinely verify their ingested objects by requesting the PDF as a PURL.

B1.7 Repository can demonstrate when preservation responsibility is formally accepted for the contents of the submitted data objects

The final ingest of a collection, indicated by the removal of the ingest files from the to-fedora directory indicates this, as well as when a file is put into the Archiver database.

B1.8 Repository has contemporaneous records of actions and administration processes that are relevant to preservation (Ingest: content acquisition).

B2 Ingest: creation of the archivable package

B2.1 Repository has an identifiable, written definition for each AIP or class of information preserved by the repository.

B2.2 Repository has a definition of each AIP (or class) that is adequate to fit long-term preservation needs.

B2.3 Repository has a description of how AIPs are constructed from SIPs.

B2.4 Repository can demonstrate that all submitted objects (i.e., SIPs) are either accepted as whole or part of an eventual archival object (i.e., AIP), or otherwise disposed of in a recorded fashion.

B2.5 Repository has and uses a naming convention that generates visible, persistent, unique identifiers for all archived objects (i.e., AIPs).

B2.6 If unique identifiers are associated with SIPs before ingest, the repository preserves the identifiers in a way that maintains a persistent association with the resultant archived object (e.g., AIP).

B2.7 Repository demonstrates that it has access to necessary tools and resources to establish authoritative semantic or technical context of the digital objects it contains (i.e., access to appropriate international Representation Information and format registries).

B2.8 Repository records/registers Representation Information (including formats) ingested.

B2.9 Repository acquires preservation metadata (i.e., PDI) for its associated Content

Information.

B2.10 Repository has a documented process for testing understandability of the information content and bringing the information content up to the agreed level of understandability.

B2.11 Repository verifies each AIP for completeness and correctness at the point it is generated.

B2.12 Repository provides an independent mechanism for audit of the integrity of the repository collection/content.

B2.13 Repository has contemporaneous records of actions and administration processes that are relevant to preservation (AIP creation).

B3 Preservation Planning

B3.1 Repository has documented preservation strategies.

B3.2 Repository has mechanisms in place for monitoring and notification when Representation Information (including formats) approaches obsolescence or is no longer viable.

B3.3 Repository has mechanisms to change its preservation plans as a result of its monitoring activities.

B3.4 Repository can provide evidence of the effectiveness of its preservation planning.

B4 Archival storage & preservation/maintenance of AIPs

B4.1 Repository employs documented preservation strategies.

B4.2 Repository implements/responds to strategies for archival object (i.e., AIP) storage and migration.

B4.3 Repository preserves the Content Information of archival objects (i.e., AIPs).

Perhaps this is demonstrated by our decision to retain the METS/MODS metadata on objects when we changed the metadata format?

B4.4 Repository actively monitors integrity of archival objects (i.e., AIPs).

B4.5 Repository has contemporaneous records of actions and administration processes that are relevant to preservation (Archival Storage).

B5. Information management

B5.1 Repository articulates minimum metadata requirements to enable the designated community(ies) to discover and identify material of interest.

B5.2 Repository captures or creates minimum descriptive metadata and ensures that it is

associated with the archived object (i.e., AIP).

B5.3 Repository can demonstrate that referential integrity is created between all archived objects (i.e., AIPs) and associated descriptive information.

B5.4 Repository can demonstrate that referential integrity is maintained between all archived objects (i. e., AIPs) and associated descriptive information.

B6. Access management

B6.1 Repository documents and communicates to its designated community(ies) what access and delivery options are available.

B6.2 Repository has implemented a policy for recording all access actions (includes requests, orders etc.) that meet the requirements of the repository and information producers/depositors.

B6.3 Repository ensures that agreements applicable to access conditions are adhered to.

B6.4 Repository has documented and implemented access policies (authorization rules, authentication requirements) consistent with deposit agreements for stored objects.

B6.5 Repository access management system fully implements access policy.

B6.6 Repository logs all access management failures, and staff review inappropriate “access denial” incidents.

B6.7 Repository can demonstrate that the process that generates the requested digital object(s) (i.e., DIP) is completed in relation to the request.

B6.8 Repository can demonstrate that the process that generates the requested digital object(s) (i.e., DIP) is correct in relation to the request.

B6.9 Repository demonstrates that all access requests result in a response of acceptance or rejection.

B6.10 Repository enables the dissemination of authentic copies of the original or objects traceable to originals.

C1. System Infrastructure

C1.1 Repository functions on well-supported operating systems and other core infrastructural software.

C1.2 Repository ensures that it has adequate hardware and software support for backup functionality sufficient for the repository’s services and for the data held, e.g., metadata associated with access controls, repository main content.

yes

C1.3 Repository manages the number and location of copies of all digital objects.

They assure us there's a copy in Indianapolis, we can't verify this. When we had corrupted files, we were assured that a second copy was pulled. Right now we have assurances from a trusted party, but we'd like to move towards written copy in SLA (service level agreement). There's no way with the given architecture that we could independently verify the second copy.

They've documented internal procedures and policies, as part of HIPAA compliance, but haven't made it available to us yet.

C1.4 Repository has mechanisms in place to ensure any/multiple copies of digital objects are synchronized.

Handled by HPSS.

C1.5 Repository has effective mechanisms to detect bit corruption or loss.

Inherent in the tape hardware there's some checksumming and error checking, there's also support at network transport layers. We likely need routine checking at the file level.

There also exists pieces of hardware that can monitor read-errors on tape drives and do reporting on tapes going bad, but none is yet used by HPSS.

C1.6 Repository reports to its administration all incidents of data corruption or loss, and steps taken to repair/replace corrupt or lost data.

No formal process. We have technical solutions, but they aren't in place.

C1.7 Repository has defined processes for storage media and/or hardware change (e.g., refreshing, migration).

Yes, for HPSS but we haven't seen them.
For intelligent infrastructure there's regular funding and scheduled equipment replacement on a <4 year timeframe.

C1.8 Repository has a documented change management process that identifies changes to critical processes that potentially affect the repository’s ability to comply with its mandatory responsibilities.

There is a change management process for IU, but we don't use it. We need to adopt an approach to prevent unexpected data loss or outages.

C1.9 Repository has a process for testing the effect of critical changes to the system.

We lack a formal deployment process internally, we don't know about HPSS.

C1.10 Repository has a process to react to the availability of new software security updates based on a risk-benefit assessment.

Yes. Brian looks at patches and security updates. UITS sends notifications occasionally.

C2 Appropriate technologies

C2.1 Repository has hardware technologies appropriate to the services it provides to its designated community(ies) and has procedures in place to receive and monitor notifications, and evaluate when hardware technology changes are needed.

Probably. We get end-of-life messages.

C2.2 Repository has software technologies appropriate to the services it provides to its designated community(ies) and has procedures in place to receive and monitor notifications, and evaluate when software technology changes are needed.

Yes, we're probably fine.

C3. Security

C3.1 Repository maintains a systematic analysis of such factors as data, systems, personnel, physical plant, and security needs.

C3.2 Repository has implemented controls to adequately address each of the defined security needs.

Our data is as secure as our password, and HPSS manager can delete data.

We'd like to ask Kurt if there's a way to undelete items.

C3.3 Repository staff have delineated roles, responsibilities, and authorizations related to implementing changes within the system.

We probably need to be better. We need to distribute responsibility so no one role has the power to destroy everything.

C3.4 Repository has suitable written disaster preparedness and recovery plan(s), including at least one off-site backup of all preserved information together with an off-site copy of the recovery plan(s).

We have a document.