Outcomes - Preservation Worthiness

Scope

Different repositories have different preservation workflows, and the role that Fedora 4 plays in the preservation workflow determines which functionality is relevant to preservation. There are three broad categories of relationship between the repository and preservation:

1. The repository is not involved in preservation at all. The repository may be downstream from or parallel to the preservation system, or preservation may not be done at all. In this case, preservation is completely independent of Fedora 4, so no preservation functionality is relevant.
2. The repository is used to enable preservation. The repository is upstream from the preservation system, and preservation-enabling functionality is a part of this process.
3. The repository is used as a preservation system. Fedora 4 is not, by itself, a complete preservation system, but preservation-enabling functionality can be used as part of the preservation process.

Preservation-enabling functionality

In addition to providing a general-purpose REST API (which can be used by a preservation workflow to retrieve metadata and content), Fedora 4 provides functionality to make it easier to verify fixity and create backup copies of repository content:

- **Upload fixity-checking for detecting transmission errors:** When uploading a content file, a SHA-1 checksum can be included with the request. The integrity of the uploaded content will then be checked by comparing its checksum to the provided checksum, and an error will be raised if there is a mismatch.
- **In-place fixity-checking for detecting storage errors:** Fedora 4 automatically calculates a SHA-1 checksum for content files when they are uploaded, and the checksum is recorded. The Fixity REST API endpoint can be used to calculate the checksum of a content file and compare it to the stored checksum.
- **Automatic metadata serialization**: The message consumer can be configured to automatically serialize RDF to disk for objects that are created or updated. This can be used to create an automatic backup of repository metadata.
- **Backup and restore**: The Backup and Restore REST API endpoint can be used to create a consistent backup of all repository content, and to restore a backup. Backups include metadata stored in JSON files and content files named according to their SHA-1 checksums.

Disaster-recovery functionality

In addition to the functionality above, Fedora 4 can also be used to store content in a transparent or human-readable way. This functionality allows repository content to be retrieved directly from the filesystem in a worst-case scenario where repository software stack is not working:

- **Read-only filesystem federation**: Filesystem Federation may be used to make content from the filesystem available as read-only objects in the repository. One common pattern is to put content files in the federated filesystem, and create metadata-only objects in the main repository storage. The federated filesystem contents can be copied, backed up, and accessed using the filesystem directly without needing Fedora 4
- **ModeShape storage artifacts**: Using the default configuration, content files are stored as files on disk, named according to their SHA-1 checksums. So if you have metadata that includes the SHA-1 checksums, it is easy to find the corresponding file on disk.