Use Scenarios

These use scenarios are provided to guide development of more detailed requirements for the ingest tool.

Use scenarios, in expected order of importance

Batch ingest

Ingesting a batch of files that come from an external system where media and metadata are complete.

Amy (a DLP developer) receives a set of files for the new "Insects in the Wells Library" photograph collection. These files include master TIFF images and item-level MODS records. Amy runs a manually-created script to generate derivative images. She creates a collection configuration file based on the collection template, and submits a small subset of the collection to the ingest tool. When she is satisfied that items are being ingested correctly, she ingests the rest of the collection.

Notes:

- While this process happens with the majority of our collections, it is relatively infrequent, and requires some technical expertise. While the ingest tool must support it, there is little benefit to be gained from automating the steps.

Single-item (or incremental batch) ingest

Bob (a special collections librarian) digitizes a new bug for the existing "Insects in the Wells Library" collection. He submits his TIFF to imageproc, and when processing is complete, opens the cataloging tool to add some minimal metadata. When he clicks "Save", he opens the bug in his Web browser to ensure it appears correctly.

Notes:

- IN Harmony and other "active" collections will rely on this model.
- When we build a "general library holdings" collection, many objects will rely on this model.

Cataloging

Chantel (a cataloger) is improving the metadata of photographs in the Cushman collection. She searches for "utility poles", and notes the items that should have this subject heading removed. She opens the cataloging tool, types in the ID of a record, updates the relevant metadata, and saves the record.

Notes:

- While the record will be updated "live", the search index will be updated less frequently (every 5 minutes?).
- This case doesn't apply to EAD-based collections (see below).

SIP ingest

Darren (a DLP developer) receives a SIP (packaged METS and media files) from another institution. He creates scripts to break the METS file up into the metadata files he needs, and then proceeds as if he were performing a batch ingest.

Notes:

- Ideally, the ingest tool could handle a SIP intelligently, but that would require more standardization in SIP formats. If the SIP format used by Fedora's Directory Ingest Service gains support, we will want to support that format natively.

Batch metadata update

Frank (a metadata librarian) wants to update all objects in the Hohenberger collection to fit a new metadata schema. He checks out from the source control system:

- the original EAD file
- XSL tranforms to create MODS and DC records from the EAD file.
- CollectionConfiguration.xml file used to ingest the object.

Frank modifies the XSL transformation to add the new field, modifies the CollectionConfiguration.xml to remove the sections that won't be reingested except the MODS specification (e.g., removes Master file specifications, Derivative file specifications, and Technical metadata specification). He then runs the bulk Ingest Tool (servlet) to update the collection.

Updating a single media file

Helene (a QC specialist) discovers poor contrast in a photograph in the Hoagy collection. She logs and reports the problem and explains how it should be corrected. Jonas (a digitizer) receives the report and corrects the problem using the Image Cataloging Application. When he submits the new image, derivatives are created automatically by imageproc. The Ingest Tool uploads the new derivatives and regenerates any technical metadata associated with them.

Updating the structure of a hierarchical object

George (a digitizer) notices that the online copy of "Gullver's Travels" is missing a page. Using the cataloging tool, he finds the object that needs to be changed. Using the cataloging tool, attaches a new media file to the object and saves the updated object. The cataloging tool uploads the new media file, modifies the metadata structures that contain references to the media files. It also:

- Adds a PURL link to the MODS in the main METS metadata
- Generates the technical metadata using Jhove and adds it to the appropriate METS section
- Updates the struct-map records in METS
- Modifies the object on Fedora

EAD collection update

In a collection based on EAD (but with derived item-level MODS records), new item-level metadata has been created and/or new files have been digitized. Ellen (a DLP developer) generates a new set of MODS records, and ingests all the media files, along with their associated MODS records. She runs a manually-created script to compare the new records to the existing records. She updated the MODS records for objects that have changed since the last update of the collection.

Notes:

- This is rare, but necessary.
- Once it actually happends, we should document the process for comparing new MODS records with existing records.