

# Image Cataloging Application

## Current Version

Known as "PhotoCat".

Implemented for the [Liberian Collections Project](#)

[Production application](#)

[Test application](#)

Config files - production:

Feta - /usr/local/tomcat-production/webapps/photocat/WEB-INF/conf/collection/(collection-name)/field-config.xml

Config files - development:

Cheddar - /usr/local/tomcat/webapps/photocat/WEB-INF/conf/collection/(collection-name)/field-config.xml

## Background

We want to create a tool for cataloging items and placing them into the repository. Initially, this tool will work with photograph collections. We will then generalize it, or use it as the basis for other types of cataloging tools. The photo cataloging tool will be available to departments on campus that want to put content into our repository.

Non-MARC cataloging systems are still fairly primitive. No cataloging system has gained a lot of attention. Most systems are very collection-specific.

Harvard has created a general-purpose cataloging interface driven by XML Schema. However, this tool is not yet available to the general public, search for "schema-driven" [on this page](#) for updates.

Is something like this worth using? Should we use/build something that is specific to each collection? Or can we build a middle-of-the road system that takes plugins for various metadata schemas?

## Current Feature Priorities

- [Older spreadsheet](#) - out of date ⚠
- [Jira task list](#)

## Tasks/timeline

For version 1.0, due end of summer 2007:

- Images will be loaded in batches using a processing step similar to InHarmony - using ImageProc system.
- Local identifiers assigned by individual partners will be stored but also a new ID will be assigned to these items by us.
- Some more work needs to be done on metadata fields from individual partner's fields to MODS fields
  - A DC transformation for each of these will also be needed for generating DC metadata automatically.
- For the pilot test, only a limited basic search functionality will be implemented. In the future, search interface might be extended and customized for each institution.
- Simple validation such as required fields, date field compliance will be performed on the input fields in this version.
- Simple role based access control will be sufficient for this version. Authentication will be through IU CAS.
- A basic configurable interface to enter fields will be implemented. More complicated features (such as controlled vocabulary) will be added in future revisions. See [Field Configuration](#) page for a discussion on this.

LCP Pilot test plan

- Define (a few) users and roles
- Design a user interface
- Design a workflow for catalogers
- Set up imageproc for incoming images
  - Define (a relaxed) set of image validation steps to be used during the test
- Set up a pilot repository and collection
- Make accommodations for importing existing data from spread sheets
  - (We need to find out their file naming scheme for image files)

## CV integration planning

- [CV integration methods for image cataloging application](#)
- [Inventory of approaches taken by other tools](#)

## Tools we may use as a basis:

- Cushman Access Database \\bl-ldlp-euterpe\dlp\Projects\Cushman\DB\cushman.mdb

- IN Harmony Cataloging Tool \\bl-ldlp-euterpe\dlp\Projects\IN Harmony\client\inharm\_client\_1.11.exe
- Existing IU Archives Photo Cataloging Database
- [SnapDragon](#)
- VireoCat
- [Iris](#) (this link is dead)
- [UCAI](#)
- Eviada's Annotator's Workbench
- Eviada's METS-to-FOXML converter
- Eviada's Thesaurus tool looks very promising
- Other photo tools, like Picasa?
- [Intern's report on functionality of various image cataloging tools](#)
- [NISO Z39.87 Data Dictionary--Technical Metadata for Digital Still Images](#)
- [Intern's report on MIX](#)

## Some Open Questions

- How to notify users on UI about the results of lengthy processing steps (such as validation, derivative creation or saving to repository)?
- Should the items that passed the automated validation step go into production directly or should there be an approval step? Or in other words, what should the general workflow be like?
  - If so, who performs approval?