

## **The "asset actions" experiment & portal as demonstrated at the DLF forum, Spring 2006**

The Asset Actions experiment was designed to demonstrate an architectural model that could extend the traditional union catalog-like services built on top of OAI-based aggregations in order to enable more direct manipulation of digital objects. The experiment built on previous work done at the University of Virginia, Tufts and Northwestern University and was carried out under the auspices of the DLF Aquifer Technology Working Group. In particular, the experiment demonstrated how advanced tools developed at one campus (in this case the University of Virginia's Collector Tool) could be made to work in an OAI-PMH context with content originating from multiple other campuses. For purposes of this demonstration, scope was limited to digital image objects.

### **The experiment**

The experiment gathered OAI records from 5 collections from 3 repositories. One repository was hosted at Indiana University and included MODS descriptive metadata. The other two repositories, encompassing content from Northwestern University and Tufts, included only simple Dublin Core metadata and were created for OAI-PMH harvesting purposes as OAI static repositories hosted by the University of Illinois at UC. New OAI Aquifer-specific metadata formats were defined for purposes of this experiment. Aquifer MODS records combine MODS records containing descriptive metadata with "asset action records" containing "actionable" URLs pointing to different views of the resource described. Aquifer DC records embed a DC record containing descriptive metadata and an asset action record. Two schemas were developed to define Aquifer MODS and Aquifer DC OAI-compatible metadata formats:  
[http://gita.granger.uiuc.edu/AquiferTechWg/aquifer\\_dc.xsd](http://gita.granger.uiuc.edu/AquiferTechWg/aquifer_dc.xsd) and  
[http://gita.granger.uiuc.edu/AquiferTechWg/aquifer\\_mods.xsd](http://gita.granger.uiuc.edu/AquiferTechWg/aquifer_mods.xsd)

At this point, only actions for images had been defined. The records were harvested, then aggregated in a database at UIUC. A portal was built to provide access to the objects and demonstrate the utility of the actionable URLs. It also integrated the Collector tool developed at the University of Virginia. This aimed to prove the added value of integrating tools and processes developed in various DLF institutions. The portal is available at: <http://rama.granger.uiuc.edu/assetactions/>. It contains a total of 3258 records.

### **Specifications of different views**

The assumption underlying the experiment is that more precise and standardized information about how to access different views of a resource can facilitate reuse of more advanced tools and the manipulation in aggregated environments of widely dispersed content. Related assumptions appear to be driving the development of features found in more complex metadata formats such as METS, MPEG-21, and even the newest draft of MODS. METS in particular has recently added the concept of "behavioral" metadata. All 3 formats provide some elements and/or attributes which allow catalogers to specify with greater precision actionable URLs. However, for simplicity and given the single and narrow focus of this experiment, rather than try to use any of these other options, we chose to implant in our "asset action packages" a minimalistic, single-purpose, structure solely designed to convey a specific set of actionable URLs. This lightweight approach

allowed quick prototyping and took advantage of work already underway at the Universities mentioned above.

All asset action packages harvested for inclusion in the experimental portal provided URLs which could be used for the following purposes:

Action	Content
Asset Definition	the asset action record
Preview	to include in snippets
Label	to display with the image
Description	XML metadata record (DC at this point)
Web View	the image integrated in a Webpage, including a navigation bar and contextual information
Thumbnail	reduced-size versions of pictures
Screen Size	screen size image (taking the reference of a 800*600 display(?))
Max Size	largest image available or image with the largest available resolution
Dynamic View	the image, in an environment allowing user to execute actions such as zoom.

Given the exclusive focus on images, there was some redundancy in the URLs provided (e.g., Preview and Thumbnail URLs were typically the same). Also, some participants used the same URL for multiple items (e.g., IU used the same "Label" for all items in its Cushman Collection). In many instances Dynamic View and Web View were the same or very similar. **This may create potential issues if a service provider wants to label different views, he will refer the user to the same view under different names.**

There were inconsistencies institution-by-institution. Thus the thumbnails collected for the experiment range from 56\*79 or 71\*54 (war posters) to 200\*139 or 135\*200 (Cushman collection). This compares to a range of possible thumbnail sizes suggested by Jens Finke: <http://jens.triq.net/thumbnail-spec-0.3/thumbsize.html>

**Dimension (width x height) Descriptive name**

48x48	very small
64x64	small
96x96	medium
128x128	large
144x144	extra large
160x160	super large
192x192	ridiculous large

**IPR issues**

A number of data providers may be willing to share asset actions but not allow service providers to change the dimension of an image or display a part of the image only. (Just as certain data providers already specify a limitations to the use of metadata records.) It may be important to think

of a way for the data provider to encode in a machine-readable format (similar to the Creative Commons) potential limitations to the actions that a service provider can enable on the object.

### **The actions enabled in the portal**

- all actions from the UVa collector tool: adjusting brightness etc, importing metadata records, creating slideshows of images selected from the portal.
- selecting an area of an image
- annotating a part of an image
- creating a "bookbag" of images
- emailing bookbag in HTML format including thumbnails
- exporting bookbag in PDF or RTF including metadata records & full size images
- allowing different views of results: traditional snippets or thumbnails view (with thumbnails displayed on results screens pulled real-time from distributed servers).
- resizing an image from a different location
- displaying different sizes of the same image
- downloading a DC record in XML format

### **The integration of the Collector tool**

The Collector tool developed at the University of Virginia was integrated successfully to the application and appears to work with no difficulty with all the content harvested. At this point, it allows specifying the images on the page with which it should be enabled (all images with an id starting with "IMG"). The integration is lightweight. (In order to facilitate use of the collector tool in a distributed environment, we should state the adequate server settings to allow the Java applet to run as well as client-side specifications for the collector tool.) The experience of integrating different systems developed in different universities to extend their benefit to a larger community proved very useful and effective.

### **Future**

#### **▪ Automatic generation of actionable URLs**

Generating asset actions automatically for service providers: The Thumbgrabber program developed at UIUC (Tom Habing, Muriel Foulonneau) parses URLs provided in OAI records, and collects either thumbnails of the largest image in the page or a thumbnail of a screenshot of the page. It also records additional information from which it may be possible to generate enough information to generate asset actions for data providers who do not provide their own asset actions, for example because of limitation of their data provider software. In concert with services that create views of content held elsewhere (e.g., UIUC is working on a service that would make available thumbnails for remote resources) it may be possible to think of asset action packages not only as something provided directly by data providers over views they alone create, but something that can be created for data providers and that can include views generated elsewhere.

#### **▪ Re-exporting the modified object**

Storing and exporting the modified object: SVG could be investigated to provide some additional functionalities to guarantee that the modified object can be exported and preserved. Additional functionalities for exporting the data could include embedding metadata, objects, and annotations in a METS record.

- **Asset actions for text**

Defining actions for text: The University of Virginia, Tufts and Northwestern have started defining specific actions for textual digital resources. One interesting possibility could be to demonstrate generalizable nature of the Mets Navigator developed by Indiana University (in much the same way as we demonstrated the generalizable nature of the UVa Collector Tool). It allows representing individual pages of a book from a METS file. The METS file does not have to be transferred to the service provider; although, if it is not transferred to the service provider, the service provider cannot perform any control on the validity of the URLs to each individual page. The METS Navigator also requires that METS files conform to a specific profile.

- **Persistence of URLs**

The whole experiment and the concept of actionable URLs relies on the persistence of those URLs and the reliability of the association between a URL and the object behind it. If a data provider modifies its URLs, for example if the screen size view points to the thumbnail, this would prevent quality value-added services from being built. A mechanism has to be associated to the asset actions in order to guarantee the persistence of URLs. The service provider can control the URLs before data publication and/or implement a control during the publication and/or rely on persistence identification mechanisms, considering each view as a distinct object.