Building Personal Collections and Networks of Digital Objects in a Fedora Repository Using VUE

Anoop Kumar and Nikolai Schwertner
UIT-Academic Technology
Tufts University
16 Dearborn Road, Somerville, MA 02145

Introduction

Digital repository technology has evolved over the past decades enabling streamlined creation and sharing of digital content. Current implementations make it feasible to setup repository infrastructures not only for large institutions but also for smaller groups and individual users. Yet, one of the difficulties in the adoption of the technology remains the relative scarcity of powerful tools for publishing content to digital repositories. With this in mind, we developed a Fedora publishing module allowing instant ingestion of VUE-generated content maps in a Fedora repository. We take advantage of the relationship facilities within Fedora to build a knowledge base of interrelated VUE maps and resources though user contributions.

Fedora is a flexible repository system with powerful digital object model for managing institutional repository needs [1]. Fedora has become one of the popular repository systems and Tufts University has adopted it as the framework for its digital repository [2]. Some of the features, which make Fedora appealing to us include versioning, policy model, object extensibility, support for rdf, and objects relationship management.

Often times publishing content into a repository involves several stages and complex work flows, therefore requiring a lengthy content creation to publishing cycle. At Tufts, for instance, publishing content to the Tufts Digital library is a multiple-stage process involving coordination between many university groups. This process works reasonably well for archival content which requires rigorous review, but carries too high of an overhead for personal and temporary collections. Also, revising and updating the content is time consuming, which may lead to out-of-synch objects, which have fallen through the cracks. The fedora administrative client, in our opinion, has been designed for users with relatively good understanding of Fedora and is not appropriate for general content contributors, who don't necessarily possess required technical skills.

To simplify the workflow of getting objects into repository, applications such as Fez [3] and Elated [4] have been developed. Fez is a configurable digital repository and workflow management system based on Fedora. Elated is a general purpose web-based client for Fedora. While these systems are good for authoring smaller collections, they can become tedious for creating collections with many objects, building relationships, and dynamically updating them.

The Visual Understanding Environment (VUE) extends the functionality of traditional concept mapping applications by allowing users to attach content to the *nodes* or *concepts* in the concept maps [5].Prior to the VUE integration with Fedora, VUE users used to create concept maps in isolation, although the content in these maps may have

been related to that in others' maps. The value of these maps would be much higher if they could be combined with maps created by all users in a shared repository. Thus we identified the need to integrate VUE with Fedora and enable automated publishing of concept maps directly into a Fedora repository.

Currently, VUE is integrated with many digital repository systems including Fedora and allows users to search, browse and deposit content back into the digital repository systems. In VUE II, the latest version, nodes and links can be associated with ontologies, thus making the maps more meaningful not just for users but also applications that can understand of those ontologies. By taking advantage of these features in VUE, we have a developed a system to publish objects from a local VUE instance into a Fedora Repository. Objects in a concept map are published along with its metadata and relationship information.

Implementation in VUE

A concept map in VUE is considered analogous to a network of objects in a repository. Every *concept* or *object* in VUE can be assigned metadata terms using metadata schemes such as Dublin Core or VRA [6]. In addition, users can load additional metadata schemes or create their own vocabularies in order to assign terms to an object. This metadata is stored along with the objects when it is ingested into a repository. Relationships among objects are defined by Fedora's Ontology. When a link of fedora ontology type is created it gets deposited in the repository along with the object in the RELS-EXT datastream. Thus a rich network of objects along with relationships is persisted when the concept map is published to the repository.

Searching Repository

VUE implements an Open Knowledge Initiative (OKI) based framework to discover objects in digital repositories. Users may perform simple or advanced search on a specific repository or a set of repositories to retrieve desired content. The results can be dragged in to create nodes and build concept maps. The metadata associated with the digital objects is also pulled into the map and persists with the map. Once these objects are drawn into VUE, additional metadata can be associated with the objects.

Unique Identifiers for Objects

A unique identifier (id) is generated for every object published to a Fedora repository through VUE. The id is 32 bytes long and based on date, time and the physical address of the computer. It is of the following format*: vue:a40711e982405ee401a80747f97a3e30.* The chances for collision of ids among objects are therefore minimized.

Persistence to Repository

Once users create concept maps they can publish them to repository in two modes: *Publish Map* **or** *Publish All.* While *Publish Map* only publishes the map to repository, *Publish All* mode publishes each object in the map, modifies the map to point to the object in the repository and then publishes the map to the repository. Thus the map with full functionality and all related objects exists in the repository and can be shared with

other users. When the map or a resource in the map is changed and re-published the objects in the repository are assigned a new version.

Metadata

Assigning appropriate metadata to objects is an essential feature of a system that publishes to repository. VUE natively supports adding Dublin Core and VRA metadata to nodes and links. Additional metadata schemas can be loaded into VUE by changing its configuration file. Alternatively users can create custom metadata terms or just tag the objects which get transferred to the repository.

Ontology for Relationships

Every object in a concept map when published to Fedora gets relationship metadata that connects it to the map it belongs to. VUE also supports Fedora relationship ontology terms (RELS-EXT). Relationships can be created among objects within a concept map and they get persisted in a Fedora Repository thus creating a true network of objects connecting an object to all its relatives.

Implementation/Customization in Fedora

Content Models

In order to accommodate the VUE-generated objects within the Fedora framework, we designed three content models – one for VUE maps (tufts/vue/map/generic), one for managed objects (tufts/vue/other), and one for remote objects (tufts/vue/remote). All three models rely on an rdf datastream (RELS-EXT) to define their relationships to each other, one payload data stream (RESOURCE), and Dublin Core metadata (DC). Each object is assigned a unique PID generated in VUE .

Security Model

The goal of the project was to create a Fedora-based collaborative environment that would allow the VUE user base to contribute and link to a growing content map of resources. It was essential that users be allowed to add their relationships to existing objects created by other users without having privileges to perform any other modification on objects that don't belong to them. Thus, we designed a set of XACML policies that restrict all API-M Fedora operations to the owner of the object, while allowing modification of the RELS-EXT datastream by all users. We reasoned that a Wikipedia-like collaborative model with some restrictions and versioning would meet our requirements.

It is worth noting that in order to allow VUE to create objects using Fedora's upload servlet, we had to allow object creation by all users and also provide access to the upload servlet to any user. VUE creates new objects in one transaction using FOXML after uploading the managed content to Fedora and obtaining the requisite Fedora-assigned URNs for the managed datastream. Once the object is ingested, only the owner has full control over the object.

Collection model
We represent a VUE map in Fedora as a collection of objects corresponding to the resources within the map. Each object containing a resource encapsulates relationship metadata representing its links to other nodes of the content map. In addition, we ingest one object containing the entire map in the native VUE format. The map object's relationship metadata points to all the objects representing its nodes. If multiple maps are ingested, which share common objects, VUE does not create separate resource objects for each content map, but instead reuses the already existing resource object within Fedora and adds relationship data to it so that it also becomes part of the new map.
In essence, each content map generated by a VUE user becomes a subset within a super-map containing all user-generated map. This super-map of resources and relations grows organically through each new ingested VUE map, which contains links to existing resources.

Versioning of objects
Fedora provides automatic versioning of the objects within the repository. This, combined with the policy allowing only modifications to the RELS-EXT data stream by non-owners, creates a wiki-like versioned collaborative environment which offers the ability to roll back changes to the objects (for example, after intentional or unintentional vandalism).

Discussion
We have successfully demonstrated content and collection creation in Fedora through user friendly workflow in VUE. Users can use VUE to ingest large sets of objects along with metadata. In addition relationships among the objects are captured. The Fedora community can use this approach to share and collaboratively build a rich network of digital resources.

References
[1] Payette S. and Lagoze C.: Flexible and Extensible Digital Object Repository Architecture, in Christos Nikolau and Constantine Stephanidis, eds., *Research and Advanced Technologies for Digital Libraries: Proceedings of Second European Conference, ECDL '98, Heraklion, Crete,Greece, September 21-23, 1998.*

[2] Kumar A, Saigal R., Chavez R and Schwertner S.: Architecting an extensible digital repository. JCDL 2004

[3] Fez: http://dev-repo.library.uq.edu.au/wiki/index.php

[4] Elated: http://elated.sourceforge.net/

[5] Kumar A. and Kahle D. VUE: A concept mapping tool for digital content. Paper presented at the *Proc. of the Second Int. Conference on Concept Mapping,* San José, Costa Rica, CMC 2006

[6] VRA: http://www.vraweb.org/projects/vracore4/index.html