

Of Hats and Islandora (the Tuque story)

Jonathan Green, CTO, DiscoveryGarden Inc., jonathan@discoverygarden.ca

Co-presenter TBD

Abstract

This session will describe in general how Tuque was constructed to be a repository agnostic API to allow Islandora to connect to repositories using the Object Oriented approaches that programmers would expect. We will dive into the basics of Tuque, how it is laid out, and where all the major features are. We will then delve into some specific examples where Tuque has been used to implement features that would have been hard or impossible without it in the Islandora project.

Session Proposal

Background

As the Islandora project grew we decided that we needed an abstraction layer between Islandora and the backing repository and the Tuque project was born. Tuque is at its core a PHP library to connect to repositories. It is intentionally Drupal agnostic to allow it to be used in as many PHP applications as possible. It contains a large number of unit tests, with new tests being added for every new feature.

Overview

Tuque breaks the repository down into three basic constructs: Repository, Objects, and Datastreams. Each of these constructs has an abstract base class, which can be implemented for any different repository.

The repository class has methods to do basic manipulation of the repository, you can add new objects, remove objects, and find objects. Adding repository objects in Tuque is a two step process. First you create a new object, which creates an object in php, which you can use to fully describe the repository object, then when you are done, you commit the object to the repository by ingesting it.

The object and datastream classes allow the user to manipulate repository objects and datastreams. You can change the contents, and change any properties of the objects and datastreams. In the session we will present some simple code snippets and examples to illustrate how to use Tuque.

Islandora Integration

When integrating Tuque into Islandora we took advantage of Tuques heavily object oriented approach and extended all of the Tuque base classes for use in Islandora. The session we will walk through some code examples. We did an integration at this level so that we could bring some Drupal and Islandora specific features into Tuque. Specifically we call a number of hooks before any action is taken in Tuque, so that other Drupal modules can tap in and take action based on what is happening in the repository.

Duracloud Integration

Using the framework provided by Tuque UPEI has added some extensions to be able to communicate with Duracloud. This project was undertaken as part of an effort to build a Duracloud module into Islandora, however some pieces of the Tuque framework proved useful to this effort. We will show some examples of the Duracloud integration, and show how it is using Tuque.

Fedora Futures

The Fedora Futures project is aimed at creating the next generation of repository. As one of the partners in the project Islandora is being used during development of Fedora as a test platform to validate the ideas that are being put into the new repository platform. To keep up with Fedora development, and allow Islandora to run on top of both the Fedora 3 platform and Fedora Futures at the same time, we have been using Tuque's base classes. We override some of the base classes, passing in what version of Fedora we want to use and to the rest of Islandora everything else is seamless.

Conclusion

This presentation will look at where Tuque is currently, how it was designed and how to use it. Then we'll look at some of the creative uses of Tuque that are being developed today, and take a look into the future at new features in future versions of Tuque.