Getting Started with Fedora

Getting Started with Fedora

The Flexible Extensible Digital Object Repository Architecture is a conceptual framework that uses a set of abstractions about digital information to provide the basis for software systems that can manage digital information. It provides the basis for ensuring long-term durability of the information, while making it directly available to be used in a variety of ways. It is very important to understand that Fedora provides a foundation upon which to build a variety of information management schemes for different use cases, not a full solution for a specific use case. The Fedora software that DuraSpace distributes has been designed to provide many different possibilities for a large array of applications.

Fedora has a very active developer community, both contributing to the core software development process and developing complete applications on top of Fedora that address particular use cases or application areas. This guide is designed to give you a basic understanding of the Fedora architecture and the core repository management software, and to give you some general ideas about how to use it. Whether you want to look at adopting one of the existing Fedora-based solutions or develop you own, this general introduction should be useful to you.

The Fedora Basics

In a Fedora repository, all content is managed as data objects, each of which is composed of components ("datastreams") that contain either the content or metadata about it. Each datastream can be either managed directly by the repository or left in an external, web-accessible location to be delivered through the repository as needed. A data object can have any number of data and metadata components, mixing the managed and external datastreams in any pattern desired.

Each object can assert relationships to any number of other objects, providing a way to represent complex information as a web of significant meaningful entities without restricting the parts to a single context.

Each data object is represented by an XML file that is managed in the file system, which contains information about how to find all of the components of the object, as well as important information needed to ensure its long-term durability. The system keeps an audit trail of actions that have affected the object, any formal policies may be asserted about the object and its use, and things like checksums, all within that XML file. As long as both the XML files and the content files that are managed by the repository are backed up properly, the entire running instance of the repository can be reconstructed from the XML files. There is no dependence upon any software to do so, no relational database that cannot be completely reconstructed from the files.

Fedora also provides a way to define any number of views of the digital object as a set of virtual datastreams or behaviors of the object, some of which can created on the fly. This allows the object to present a set of virtual data products on the front end that are derived from the actual data that is being managed on the backend.

While Fedora can easily be used to model digital collections of surrogates of traditional, catalog-based collections, it has been designed to be able to support durable web-like information architectures. Because each object completely contains all of the content, metadata and attributes of a unit of content, and can assert any number of relationships to any other object, it is easy to support schemes in which objects have multiple contexts with no dependencies upon each other. A Fedora repository does not have a particular catalog. Any number of indices, designed for specific purposes can be applied to any pattern of components of objects desired.

- The Fedora Abstractions - A description of the essential abstractions defined by the Fedora architecture that are implemented in DuraSpace's Fedora software. *(coming soon)*
- An introduction to Fedora digital objects - A detailed description of the four types of Fedora objects and what they can do.
- An introduction to FOXML - The details about the XML encoding of Fedora objects.
- Relationships using RDF - A description of how Fedora can use the Resource Description Framework (RDF) to represent object relationships.
- Content modeling - A general introduction to using Fedora to manage different kinds of information. *(coming soon)*

Using the Fedora Repository Software

We provide a test repository instance that starts up your own instance of Fedora in the cloud. You can use this to play with the web-based administrator client to get a feel for making objects and managing a repository. Note that the instance of Fedora that is started up for you will stay active for one hour, at which time it will be terminated, removing all objects that you have created.

When you are first getting started with setting up a Fedora repository on your own machine, a quick start option is provided that makes it easy to get going. This type of installation does not use any of the security features that Fedora provides, eliminating much of the complexity that often trips up new users. It is recommended that you start with that, then turn on the security features as you get comfortable with the basics.

There are two ways to create objects in your Fedora repository. You can use the web-based client to create them interactively one at a time, or you can construct your own workflows that create FOXML files which can be ingested into Fedora, either singly or as batch from a single directory.

When you are ready to start using service objects, we provide an application called EZService that makes it easy to create basic service objects from some XML template files. When you are ready to get more into the details, take a look at the Content Model Architecture (CMA) Construction Guide to get more of the fine details of what is possible.

When you are getting started with Fedora it is usually best to keep the security and policy enforcement functionality turned off. When you are interested in using that functionality, Fedora provides complete policy expression and enforcement systems that allow you to write policies that can be applied repository wide, to any object or any component of any object.

- Use our test repository- We provide instance of a Fedora repository running in the cloud that you can use without having to download and configure software.
- Installation and Configuration on your machine - This is the starting point for setting up a Fedora repository on your server, which includes a quick start option.
- The Fedora Web-based Administrator - You can use this to interactively create and update objects, as well as to administer your repository.
- Creating and Updating Data Objects - The complete guide to building objects.
- EZService - A quick start for creating service objects.
- The CMA Construction Guide - A detailed guide to creating content model and service objects
- Security

Search and Discovery

The nature of an object-based repository, like Fedora, is to manage all information in the most modular manner, in a way that is as independent of any particular software as possible.There is no database that holds metadata fields. There are services, such as GSearch and PrOAI, that harvest content and metadata from objects in various ways for various purposes. The best practice for building access systems for a Fedora repository is to use such services to build one or more indexes that are tailored to your needs.

There is a built-in search, Basic Search, that was included so that repository managers would have something to use to help them manage their repository. It indexes the required DC datastream, which is either an in-line XML datastream or a managed content datastream. If you provide this information as in-line XML it is best to keep this datastream as small as possible, as the data is actually stored in the FOXML file. If your FOXML files average larger than about 20 k in size, performance can be affected in situations where you have many simultaneous users. It is best to put basic info in the DC datastream that is useful in repository management, but not elaborate descriptive info. If you want a rich Dublin core record it is best to put it in a managed content datastream and index it using GSearch.

Both GSearch and PrOAI are designed to let you be selective in which objects you want included and to let you specify which datastreams you want to be included, either actual datastreams or virtual ones that result from a service call. GSearch lets you use one or more search engines that are already included to define different kinds of indexes. It also provides a way for you to create a plugin for your favorite search engine, if it is not already included. PrOAI lets you selectively expose your repository under the Open Archives Initiative (OAI) scheme.

- Basic Search - The built-in search that is intended for repository managers use, not intended to be exposed externally.
- GSearch - The Generic Search Service that makes Fedora's features useful to different search engines.
- PrOAI - The OAI provider service that is designed to take advantage of Fedora's features.
- The RDF-based Resource Index - This is Fedora's built-in semantic store.

Tutorials

- Tutorial 1 - Introduction to Fedora
- Tutorial 2 - Creating Fedora Objects

Fedora-based Applications

Below is a list of applications that run on the current 3.x versions of Fedora (or will soon be available). For a more complete community software registry that includes applications that run on earlier generations of Fedora, or are other useful tools and utilities, see our Community Software Registry .

- ActiveFedora- Built on RubyFedora, this ruby gem provides an active record oriented way of working with objects in Fedora
- ESciDoc - An eResearch environment developed specifically for use by scientific and scholarly communities.
- The Fascinator - A front end to Fedora commons repository that uses Solr to handle all browsing, searching, and security.
- Hydra - Will provide a "Lego Set" of web services and templates that can be used for a wide range of content management workflows. (Not released yet but coming soon.)
- Islandora - A Drupal module that allows users to view and manage objects stored in Fedora.
- RODA - An OAIS-compliant, service-oriented digital repository system designed to preserve government authentic digital objects.