




# OAI Provider Service 1.2.2

Current released version

 OAI Provider Service **1.2.2** is the currently released version. [See release notes](#)  
Compatibility

 Compatible with Fedora Version 3.0-3.5.  
FCREPO-832

 Due to an outstanding issue, the oaiprovider requires a default character encoding of UTF-8.

Supply the parameter `-Dfile.encoding=utf-8` to the JVM, for instance by including this in your `CATALINA_OPTS`. This is only necessary if your default platform encoding is not UTF-8.

## Table of contents:

- [About This Service](#)
- [Major Features](#)
- [Installation](#)
- [Demonstration and Test Setup](#)
- [Configuration](#)

## About This Service

The Fedora OAI Provider Service is part of the Fedora Service Framework and offers a vast improvement over the previous, simple OAI provider that the Fedora Repository used in previous versions. The new provider is based on [PROAI](#), an open source caching, polling OAI provider.

## Major Features

- Supports any metadata format available through your Fedora Repository via a Datastream or dissemination service
- Supports sets that are expressed as RDF relationships in your digital objects' `RELS-EXT` Datastreams as exposed via the Resource Index
- Runs as a Web application in any servlet container, acting as a Web service client to the Fedora Repository
- Caches the content of Fedora Repository disseminations and Datastreams intended to be exposed as OAI records allowing for fast response times and ensuring that the OAI provider can continue to run *even when the Fedora Repository is temporarily stopped*.
- Automatic schema validation - Incoming record XML can now be automatically validated and rejected if it doesn't conform to the XML schema it declares.
- Support for Mulgara (formerly Kowari) and MPTStore - This service requires that Fedora's Resource Index is enabled, and can be used whether it is backed by Mulgara, Kowari (Fedora 2 and earlier), or MPTStore.

See the [Configuration](#) section for more detail on using these capabilities.

## Installation

To install the service:

1. Make sure you have a suitable database installed (MySQL, PostgreSQL, Oracle, or McKoi) and a database user account that can create tables in the database.

Make sure your Fedora Repository is running with the [Resource Index turned ON](#). This is necessary because the OAI provider periodically queries the resource index to discover which records of interest have changed.

1. Deploy the `oaiprovider.war` file into your servlet container.
2. Configure the OAI Provider as described in the [Configuration](#) section below.
3. Re-start the Web application (this is often done by restarting the servlet container itself).

## Demonstration and Test Setup

The source distribution of the OAI Provider service includes several test Fedora digital objects. You can use these objects and the default `proai.properties` configuration file to quickly understand how the service works.

1. Complete installation steps 1-4 above. Start with the default values in the `proai.properties` configuration file and ensure the following properties are set according to your own Fedora Repository installation:
  - `driver.fedora.baseUrl`
  - `driver.fedora.user`
  - `driver.fedora.pass`
2. Make sure your Fedora Repository installation is configured to retain (rather than re-generate) PIDs of objects in the "demo" PID namespace on ingest. You can check this in your `fedora.cfg` file: If one of the values of "retainPIDs" is "demo" or "" (asterisk), your repository is configured correctly. Otherwise, you should add this value and re-start it.

3. Use the `fedora-admin` GUI or `fedora-ingest` command-line utility to ingest all demonstration objects in the `src/test/foxml` directory of the Fedora OAI Provider service source distribution.
4. Start the Web application.

Upon starting, the Fedora OAI Provider service will poll the repository for objects that provide OAI record content and have changed since its last update. It will find the demonstration objects you just ingested, request appropriate disseminations of each, and save them in its cache. Once it has successfully completed a cache update cycle, you should be able call the `oaiprovider` client service interface with OAI-PMH verbs. Here are some examples (note that you may have to substitute the name of your host for `localhost`):

```
http://localhost/oaiprovider/?verb=Identify
```

```
http://localhost/oaiprovider/?verb=ListRecords&metadataPrefix=oai_dc
```

Now try using the `fedora-admin` command line application to edit a Datastream in one of the demonstration objects you just ingested. The next time the OAI Provider service polls the repository for modified objects, it should pick up this change and make it available via the `oaiprovider` client service interface.

Note that the poll frequency is ten minutes by default - this can be configured to be more or less frequent by changing the `proai.driverPollSeconds` property in the `proai.properties` file.

## Configuration

Please consult the [Configuration Reference](#) for a complete list of `proai.properties` configuration options and what they do.

You should also take a look at the `log4j.xml` that resides alongside the `proai.properties` file in the Web application's `WEB-INF/classes` directory. This file controls where log messages will be sent and how much detail they will contain. When first setting up the Fedora OAI Provider service, it is recommended that you keep the log level high (DEBUG for most categories).