Installation and Configuration

This guide will show you how to install a new Fedora Repository using the installer, or from source code. If you are upgrading from a previous release, please see Upgrading from 2.x or Upgrading from 3.x. If you have never installed Fedora before, please start with some background considerations.

On this page:

Prerequisites

Download Fedora 3.8.1

- Fedora 3.8.1 Installer (161.5M)
- Fedora 3.8.1 Source Code (7.3M) (if installing from source, see Installation From Source for instructions on building the installer)
- See also the Downloads page.

Java SE Development Kit (JDK) 8.
Whether installing a binary or source distribution, JDK 8 is required. The JDK should be installed on the machine you intend to use as the Fedora server. It is available from http://www.java.com/. While historically we have recommended Sun/Oracle's JDK, there have been several reports of success in using OpenJDK as well. Look here for more information on installing Java.

Database
Fedora uses a relational database to support some of its functions. To simplify installation, the Fedora installer includes and can configure an embedded instance of the Derby SQL Database 10.9.1.0. Fedora supports four external databases: MySQL, Oracle, PostgreSQL and Microsoft SQL Server. The embedded Derby database should only be used for evaluation and development purposes; Derby should not be used for any production repository. It is recommended that you use one of the supported external databases which must be installed, configured and running before proceeding with the installation.

Application Server
The Fedora Repository installer includes Tomcat 7.0.55. Optionally, Fedora may be installed into any existing application server that implements Servlet 2.5 /JSP 2.1 or higher. At this time, Fedora has chiefly been tested with Tomcat 6.0.x and 7.0.x, but users have reported running Fedora successfully with Jetty and JBoss.

Maven 2
Fedora uses Maven for its build environment. Maven2 is available from http://maven.apache.org/.

Installation Steps

Prepare Environment Variables

The following environment variables may need to be set while installing or running Fedora, or executing Fedora command line utilities.

JAVA_HOME
Optional.

If set, tomcat and fedora command line utilities will use the specified java vm. Otherwise, all utilities will attempt to use the java version available by default within the user's PATH.   This should point to the base directory of your Java installation. On Windows systems, this might be `C:\java`. For UNIX derivatives, this might be something like `/usr/local/jdk1.8.0_31`.

FEDORA_HOME
Required for running command line utilities. Optional for running installer. Ignored when running the Fedora server unless fedora home directory is not otherwise specified.

This is the directory where Fedora will be installed, for example, `C:\fedora` (Windows) or `/usr/local/fedora` (UNIX derivatives). Note: This is only required when running the Fedora client command line utilities. The server also requires this information at run time, but can accept it from the following sources (listed in order of preference):

- The `fedora.home` context-param configured provided by the webapp container (preferred). Most webapp containers provide some means for providing context params to individual web applications at startup. This configuration is separate from the deployed webapp. The installer will automatically provide this configuration when installing to a Tomcat container in `$CATALINA_HOME/conf/Catalina/localhost /<webapp_name>.xml` (typically `fedora.xml`).
- The fedora.home system property, configured as appropriate for your web application server of choice.
- The FEDORA_HOME environment variable, as available when the web application server starts.

PATH
This must include the Java and Fedora bin directories. For UNIX derivatives, this will be `$FEDORA_HOME/server/bin`, `$FEDORA_HOME/client/bin` and usually `$JAVA_HOME/bin`. For Windows, this will be `%FEDORA_HOME%\server\bin`, `%FEDORA_HOME%\client\bin` and usually `%JAVA_HOME% \bin`.

If you will be building from source, Maven should also also be on your path.

JAVA_OPTS
If Fedora is configured to use SSL, `JAVA_OPTS` must include the `javax.net.ssl.trustStore` and `javax.net.ssl.trustStorePassword` properties. See the SSL section below for more information.

CATALINA_HOME
Required for command line applications.

Command line applications currently load libraries from the exploded web application in `$CATALINA_HOME/webapps/$webapp_name/WEB-INF`

DISPLAY (Unix-only)
When running a Fedora server in a Unix-like operating system (Linux, Solaris, OS X, etc), you should ensure that this environment variable is NOT set by the user who will be running the application server in which Fedora is installed (e.g. Tomcat). *Background:* Fedora and the included web applications are designed to run without access to a graphics output device. Although rare, having this environment variable set has been reported to cause stability problems in certain installations of Fedora. Since a graphic output device should never be needed by the Fedora server, it is safest to ensure this environment variable is not set.

Prepare Database

Fedora is designed to be RDBMS-independent. Fedora has been tested with Derby, MySQL, Oracle, PostgreSQL and Microsoft SQL Server. The embedded version of Derby included with the installer is provided as a convenience; Derby is not recommended for use in production repositories. If you choose to use any database other than the embedded Derby provided by the Fedora Installer, you must install that database first.

Follow the instructions below for the RDBMS of your choice in order to create the user and tables required by Fedora.

MySQL

Please note that the MySQL JDBC driver provided by the installer requires MySQL v3.23.x or higher.

The MySQL commands listed below can be run within the mysql program, which may be invoked as follows:

```
mysql -u root -p
```

Create the database. For example, to create a database named "fedora3", enter:

```
CREATE DATABASE fedora3;
```

Set username, password and permissions for the database. For example, to set the permissions for user fedoraAdmin with password fedoraAdmin on database "fedora3", enter:

```
GRANT ALL ON fedora3.* TO fedoraAdmin@localhost IDENTIFIED BY 'fedoraAdmin';
GRANT ALL ON fedora3.* TO fedoraAdmin@'%' IDENTIFIED BY 'fedoraAdmin';
```

MySQL 4.1.x users must also specify the default character set for the Fedora database as "utf8" and the default collation as "utf8_bin". For example, to set the default character set and collation on a database named "fedora3", enter:

```
ALTER DATABASE fedora3 DEFAULT CHARACTER SET utf8;
ALTER DATABASE fedora3 DEFAULT COLLATE utf8_bin;
```

Oracle

To prepare Oracle for use with Fedora, the following steps should be taken by an administrative user. First, using the Database Configuration Assistant, ensure that the database you'll be using is created with the UTF8 charset. Next, you'll need to create a Fedora tablespace and user in the database. Assuming the administrative user is sys and the SID is fedora3, log in using SQL*Plus using the following command:

```
sqlplus sys/PASSWORD@fedora3 as sysdba
```

To create a tablespace named "fedora_tblspace" with data in `/var/lib/oracle`, enter the following:

```
CREATE TABLESPACE fedora_tblspace
DATAFILE '/var/lib/oracle/fedora_tblspace.dat' SIZE 1024M REUSE
AUTOEXTEND ON NEXT 256M MAXSIZE UNLIMITED
SEGMENT SPACE MANAGEMENT AUTO;
```

To create a user "fedoraAdmin" with password "fedoraAdmin", using the "fedora_tblspace", enter the following:

```
CREATE USER fedoraAdmin IDENTIFIED BY fedoraAdmin
DEFAULT TABLESPACE fedora_tblspace;
```

Using the GRANT command, make sure the user has permission to connect, create, alter, and drop tables sequences, triggers, and indexes in this tablespaces. For example:

```
GRANT ALL PRIVILEGES TO fedoraAdmin;
```

Due to distribution license restrictions, the Fedora Installer does not include the Oracle JDBC driver. Oracle JDBC drivers are available from http://technet. oracle.com/software/tech/java/sqlj_jdbc/content.html. The installer will prompt you for the location of the driver on your filesystem. Also, if you run Fedora in Java 7 as is required by Fedora 3.8, you will need an Oracle Java 7 jdbc jar such as `ojdbc7.jar`.

PostgreSQL

Please consult the documentation at http://www.postgresql.org/docs/ for more detailed information about configuring PostgreSQL.

Launch the PostgreSQL interactive terminal, `psql`, (optionally appending the -U argument to connect as a different user).

```
psql -d postgres
```

To create a user "fedoraAdmin"" with password "fedoraAdmin" and database named "fedora3", enter the following:

```
CREATE ROLE "fedoraAdmin" LOGIN PASSWORD 'fedoraAdmin';
CREATE DATABASE "fedora3" WITH ENCODING='UTF8' OWNER="fedoraAdmin";
```

Microsoft SQL Server

David Handy has contributed a guide for interfacing Fedora with MS SQL Server

Other Databases

To use a database other than Derby, McKoi, MySQL, Oracle, PostgreSQL and Microsoft SQL Server, the database must support common SQL-92 syntax and you must have a JDBC version 3 driver available.

The JDBC driver will need to be installed manually. For most containers, the driver may be placed in the Fedora webapp's `WEB-INF/lib` directory. For Tomcat 6.0.x and 7.0.x, however, the driver should be installed to `TOMCAT_HOME/lib`. The JDBC URL will need to be configured appropriately in the Fedora Server Configuration File.

Upon startup, Fedora checks the database for all required tables. If the tables do not exist, Fedora will create them. Because table creation is much less standardized task across RDBMSs than SQL queries you must do one of the following:

1. Create the tables and indexes and auto-increments yourself in your own database (see the file: `src/dbspec/server/fedora/server /storage/resources/DefaultDOManager.dbspec` in the source distribution for the RDBMS-neutral table specifications).
2. Write a subclass of `fedora.server.utilities.DDLConverter` for your database software, include it in the Fedora `WEB-INF/classes` directory or in a jar file in the Fedora `WEB-INF/lib` directory, and associate it with the JDBC driver inside the `FEDORA_HOME/server/config /fedora.fcfg` file (see how it's done by looking at the MySQLDDLConverter and DerbyDDLConverter associations with their respective drivers

in the `fedora.fcfg` file, and the classes' implementations in the source distribution). If you choose option #2, please tell us about it, as it will be useful for other users of Fedora. Option 2 is harder, but it will make future installations of new versions of Fedora (where the db schema will likely change) much easier for you if you plan on using that database later.

Run Installer

The Fedora Installer provides three installation options: *quick*, *custom*, and *client*.

To start the installer, change to the directory where you downloaded the installer and at a command prompt, enter:

```
java -jar fcrepo-installer-3.8.1.jar
```

Fedora can also be installed in non-interactive mode by specifying an install.properties file as an argument to the installer. After installing interactively, you will find an install.properties file in your $FEDORA_HOME/install/ directory. You can use this file as a template for future, non-interactive installations. *Take care if you are using an install.properties file from an earlier release as the file may not contain some properties defined in the later release, default settings will be used in this case, which may or may not be appropriate for your installation.
Please ensure that the user account that is running the installer has sufficient permissions to write to the directories where Fedora will be installed (if deploying to an existing Tomcat installation, this includes permissions to the Tomcat directory). Installer created files will usually be owned by the user running the installer. Consequently, for example, after installation users of the Fedora Admin client will need write permissions to the log files defined by `config/logback.xml`.

Quick Install

The quick option is designed to get Fedora up and running as quickly as possible, with a minimum of advanced options. The quick install will automatically install Tomcat pre-loaded with the Fedora Repository and the Derby database. Neither SSL support nor XACML policy enforcement is enabled by the quick install.

Custom Install

The custom option provides the most flexibility in configuring an installation. Options include the choice of servlet container, database, the host, ports and application server context Fedora will be running on, enabling optional services, as well as security options including SSL, XACML policy enforcement, and FeSL.

Servlet Container

The installer will automatically configure and deploy to Tomcat 6.0.x and 7.0.x servlet containers. However, if an existing Tomcat installation (as opposed to the Tomcat bundled with the installer) was selected, the installer will not overwrite your existing `server.xml`, but rather, place a modified copy at `FEDORA_HOME/install` so that you may review it before before installing it yourself.

Other servlet containers will require manual deployment of the *war* files located at `FEDORA_HOME/install`.

Application Server Context

The installer provides the option to enter an application server context name under which Fedora will be deployed. The context name defaults to Fedora (resulting in http[s]://host:port/fedora), however any other valid context name can be supplied. The installer will name the resulting *war* file according to the supplied context name (defaults to `fedora.war`). Please ensure that the servlet container configuration reflects the name of the Fedora context name in case it needs to be configured explicitly. For further details see Alternative Webapp Context Configuration.

SSL

Configuring SSL support for Fedora's API-M interface is an optional feature. It strongly recommended for production environments if Fedora is exposed to unsecured application and users. However, if your installation is within a managed data center with firewall services, you may choose to provide SSL using a software or hardware front-end instead. For example, a reverse proxy implemented using the Apache HTTP Server and hiding Fedora generally provides better SSL performance.

If the Tomcat servlet container is selected, the installer will configure `server.xml` for you. However, as noted above, if an existing Tomcat installation was selected, the installer will not overwrite your existing `server.xml`.

Please consult your servlet container's documentation for certificate generation and installation. (In particular, the example certificate provided by the installer for Tomcat should not be used in a production environment).

If Fedora is configured to use SSL, the `JAVA_OPTS` environment variable must include the `javax.net.ssl.trustStore` and `javax.net.ssl.trustStorePassword` properties. The value of `javax.net.ssl.trustStore` should be the location of the truststore file and the value of `javax.net.ssl.trustStorePassword` is the password for the keystore. The following values may be used with the sample keystore included with the installer:

```
-Djavax.net.ssl.trustStore=$FEDORA_HOME/server/truststore -Djavax.net.ssl.trustStorePassword=tomcat
```

FeSL

The Fedora Security Layer is an experimental feature introduced from Fedora 3.3. FeSL consists of two separate components, which can be selected independently during the installation: FeSL Authentication and FeSL Authorisation.

FeSL Authentication is now the default authentication mechanism, however Fesl Authorization is still considered experimental. FeSL Authorization is a replacement for the legacy XACML policy enforcement, so you should not enable XACML policy enforcement if you are going to use FeSL Authorization, as this will provide an alternative XACML policy enforcement engine. See FeSL Installation for more information about FeSL requirements that must be satisfied prior to installation.

Resource Index

If the Resource Index is enabled, Fedora will use Mulgara as its underlying triplestore, with full-text indexing disabled.

Messaging
If Messaging is enabled, Fedora will create and send a message via JMS whenever an API-M method is called.

Client Install

Both the quick and custom options will install the Fedora client software in addition to the Fedora server. The client option, however, will install only the Fedora client software.

Start Fedora

If you selected the quick install option, you will find Tomcat installed in `FEDORA_HOME/tomcat`. To run Fedora, start Tomcat by entering:

```
$FEDORA_HOME/tomcat/bin/startup.sh
```

(or for Windows)

```
%FEDORA_HOME%\tomcat\bin\startup.bat
```

If you selected the custom install option, ensure that your database server is running (unless you selected the included Derby option which will be automatically started when the first database connection is made).

Installing the Demo Objects

If you just started Fedora for the first time, it's a good idea to check out the demonstration objects to get an idea of how Fedora works. See the Demonstration documentation for complete descriptions.

NOTE: If, during a custom install, you entered values other than the defaults for *fedoraServerHost* (localhost) or *fedoraServerPort* (8080), you must run the **demo object converter utility** script to change the host and/or port in the demonstration object ingest files. The demonstration object conversion is only required if you are ingesting demonstration objects. If the demonstration objects are already ingested into the repository (e.g. from a previous installation), there is no need for conversion. The demonstration objects are shipped with references to "localhost:8080" and these references must reflect the new values of *fedoraServerHost fedoraServerPort*. Refer to the Command-line Utilities documentation for additional details on running the demo object converter.

To ingest the demonstration objects, at a command prompt, enter:

```
fedora-ingest-demos.sh [hostname] [port] [username] [password] [protocol]
```

(or for Windows)

```
fedora-ingest-demos.bat [hostname] [port] [username] [password] [protocol]
```

For additional information on the fedora-ingest-demos command, see the documentation for the Client Command-line Utilities. Please note that the demonstration objects must be ingested before they can be discovered using the default search interface.

Additional Configuration

fedora.fcfg

The Fedora Server's configuration has historically been governed by the Fedora Server Configuration File, `fedora.fcfg`, located at `FEDORA_HOME/server/config/fedora.fcfg`. This means of configuration is slowly being replaced with spring bean configuration

The Fedora server configuration file contains:

- Global parameters for the Fedora server

- Configuration parameters for each server module
- Configuration parameters for each persistent data store

The configuration file has a simple schema. It starts with a server element, under which a series of parameter elements occur, followed by a series of module elements, followed by a series of `datastore` elements. The parameter elements directly following the root server element are used to control what are considered generic server functionality; for example: the port on which the server is exposed.

The module elements are used to configure specific parts of Fedora. For instance, the module with the role attribute `fedora.server.search. FieldSearch` is used to configure the field-searching component of the server. Inside the module element, several `param` elements are included. These are specific to that module's implementation. Descriptions of each parameter can currently be found in the configuration file itself.

The `datastore` elements are used to configure various databases that might be used by the system. Although the sample configuration file holds several, you will typically only need one. The `datastore` elements are associated with the modules by means of a parameter inside the associated module. In the sample configuration file, for example, the `poolNames` parameter of the `fedora.server.storage.ConnectionPoolManager` module refers to one of the `datastore` elements in its value.

There are many other parameters you can configure with Fedora. Refer to the Fedora Server Configuration File itself (`fedora.fcfg`) for internal documentation on all the parameters.

Spring configuration

For more details, see the full [Spring Configuration](#) page.

Fedora configuration via spring is eventually planned to replace the existing `fedora.fcfg` configuration, as well as eliminate the need for custom configuration if `web.xml` within the fedora server war file.

Ar present, spring bean and property definitions can be found in:

1. `server/config/spring`
    - used to define configuration for fedora server modules
2. `server/config/spring/web`
    - contains configuration for servlet filters on fedora's web interfaces, most notably related to [Spring Security](#)

Logging

Fedora uses the Simple Logging Facade for Java (SLF4J) framework for logging with Logback as the actual logging implementation. For detailed information about using SLF4J, consult the SLF4J Manual: http://www.slf4j.org/manual.html, and for information about using Logback consult the Logback manual: http://logback.qos.ch/manual/index.html.

The log configuration file is located at `FEDORA_HOME/server/config/logback.xml`. One of the benefits of using SLF4J and Logback is that configuration changes take effect without needint to restart the server.

Normally, coarse-grained logs for Fedora are written to `FEDORA_HOME/server/logs/fedora.log`. The following examples show the kinds of configuration changes you can make to aid in debugging.

To change the level to DEBUG for all Fedora classes, change the `logger name="org.fcrepo"` line to the following:

```
<logger name="org.fcrepo" additivity="false" level="DEBUG">
```

To change the level to DEBUG for just one class, add the following lines:

```
log4j.logger.fedora.server.utilities.SQLUtility = DEBUG, FEDORA
log4j.additivity.fedora.server.utilities.SQLUtility = false
```

To change the level to DEBUG for a whole package, add the following lines:

```
<logger name="org.fcrepo.server.resourceIndex" additivity="false" level="DEBUG">
<appender-ref ref="FEDORA"/>
```