Fedora Installation Annotated

Monday, April 16, 2007
7:00 AM

Fedora Installation and Configuration Guide

Fedora Release 2.2

Table of Contents

1. Introduction

This is the installation guide for Fedora. It includes instructions for installing the server and client distributions, as well as instructions for installing and compiling the complete source code distribution.

Make sure you have the correct JDK

A. Install Prerequisites

Required:

- Java SE Development Kit (JDK) 5.0.
  Whether installing a binary or source distribution, JDK 5.0 is required. The JDK should be installed on the machine you intend to use as the Fedora server. The JDK is available from http://java.sun.com/.

Optional:

You can use this database

- Database
  The Fedora server is backed in part by a relational database. To simplify installation, the Fedora installer includes and can configure an embedded instance of the McKoi SQL

Database 1.0.3. Fedora also supports MySQL, Oracle 9 and PostgreSQL. If the included McKoi database will not be used, another database server must be installed, configured and running before proceeding with the installation. To configure Fedora to use another database, please see the Database section below for further instructions.

- Servlet Container
  The Fedora installer includes Tomcat 5.0.28. Optionally, Fedora may be installed into any existing servlet container that implements Servlet 2.4/JSP 2.0 or higher. At this time, Fedora has chiefly been tested with Tomcat 5.0.28 but users have reported running Fedora successfully with Tomcat 5.5.x and Jetty.

not applicable

Additional requirements for building Fedora from source:

you'll just use the installer.

- Ant 1.6.5 or later
  Fedora uses Ant for its build environment. Ant is available from http://ant.apache.org/.

B. Download Fedora

The latest version of the software can always be found at http://www.fedora.info/download/.

There are two download options: the Fedora Installer and the source code distribution. Unless you wish to build Fedora from source, you should select the Fedora Installer.

C. Prepare Environment Variables

The following environment variables must be correctly defined:

JAVA_HOME
: 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/java-1.5.0-sun

FEDORA_HOME
: This is the directory where Fedora will be installed, for example, C:\fedora or /usr/local/fedora.

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, Ant 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
: If Fedora is configured to use Tomcat, CATALINA_HOME must be set before starting Fedora. If using the quick install option, CATALINA_HOME should be set to

In Windows DO NOT install in "Program Files" directory. The space character confuses scripts. Install as C:\fedora

Must set circled environment variables!

Try to use defaults to make your life Simple

$FEDORA_HOME/tomcat (or %FEDORA_HOME%\tomcat in Windows).

use quick!

2. Installation

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 fedora-2.2-installer.jar
```

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 the installer created FEDORA_HOME/client log directories.

A. Quick

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 a servlet container and database. Neither SSL support nor XACML policy enforcement is enabled by the quick install.

B. Custom

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

Servlet Container

The installer will automatically configure and deploy to Tomcat 5.0.x and 5.5.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`.

SSL

Configuring SSL support for Fedora's API-M interface is strongly recommended for production environments.

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).

ignore

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
```

C. Client

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.

3. Running 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 embedded McKoi option) and start your servlet container if necessary.

A. 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 Demo Manual for complete descriptions of the demos.

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 demo object ingest files. The demo object conversion is only required if you are ingesting demo objects. If the demo objects are already ingested into the repository (e.g. from a previous installation), there is no need for conversion. The demo 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 demos, at a command prompt, enter:

```
fedora-ingest-demos [hostname] [port] [username] [password] [proto
```

For additional information on the fedora-ingest-demos command, see the documentation for the client command-line utilities. Please note that the demo objects must be ingested before

You should ingest the demo objects But FIRST do 5.A.!!

they can be discovered using the default search interface.

4. Database

Fedora is designed to be RDBMS-independent. Fedora has been tested with McKoi, MySQL, Oracle and PostgreSQL. The embedded version of McKoi included with the installer is provided as a convenience. Large repositories or repositories concerned with scalability and performance should consider using an alternative database. If you choose to use any database other than the embedded McKoi provided by the Fedora Installer, we assume here that the database is already installed.

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

A. 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 `fedora22`, enter:

```
CREATE DATABASE fedora22;
```

Set username, password and permissions for the database. For example, to set the permissions for user `fedoraAdmin` with password `fedoraAdmin` on database `fedora22`, enter:

```
GRANT ALL ON fedora22.* TO fedoraAdmin@localhost IDENTIFIED BY 'fe
```

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 "fedora22", enter:

```
ALTER DATABASE fedora22 DEFAULT CHARACTER SET utf8;
ALTER DATABASE fedora22 DEFAULT COLLATE utf8_bin;
```

B. 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 `fedora22`, log in using SQL*Plus using the following command:

```
sqlplus sys/PASSWORD@fedora22 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;
```

NOTE: 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.

C. 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 `fedora22`, enter the following:

```
CREATE ROLE "fedoraAdmin" LOGIN PASSWORD 'fedoraAdmin';
```

D. Other Databases

To use a database other than McKoi, MySQL, Oracle or PostgreSQL, the database must support common SQL-92 syntax and you must a 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 5.0.x, however, the driver should be installed to TOMCAT_HOME/common/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 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.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 McKoiDDLConverter 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.

5. Configuring the Fedora Server

A. fedora.fcfg

You need to make some changes to this - see the end of the document

The Fedora Server's configuration is chiefly governed by the Fedora Server Configuration File (fedora.fcfg) located at `FEDORA_HOME/server/config/fedora.fcfg`.

The Fedora server configuration file contains: (1) global parameters for the Fedora server, (2) configuration parameters for each server module, and (3) 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 param 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.

B. Logging

Fedora uses Log4J for logging. For detailed information about using Log4J, consult the Log4J Manual: http://logging.apache.org/log4j/docs/manual.html.

The log configuration file is located at `FEDORA_HOME/server/config/log4j.properties`. Changes to the logging configuration will not be reflected until Fedora is restarted.

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 `log4j.logger.fedora` line to the following:

```
log4j.logger.fedora = DEBUG, FEDORA
```

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:

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

To send all DEBUG messages for a package to STDERR, with methods and line numbers add the following lines:

```
log4j.logger.fedora.server.security = DEBUG, SECURITY
log4j.additivity.fedora.server.security = false
log4j.appender.SECURITY = org.apache.log4j.ConsoleAppender
log4j.appender.SECURITY.target = System.err
log4j.appender.SECURITY.layout = org.apache.log4j.PatternLayout
```

To send all DEBUG messages for a package to a dedicated file, with methods and line numbers, add the following lines:

```
log4j.logger.fedora.server.security = DEBUG, SECURITY
log4j.additivity.fedora.server.security = false
log4j.appender.SECURITY = org.apache.log4j.RollingFileAppender
log4j.appender.SECURITY.File =
log4j.appender.SECURITY.MaxFileSize = 20MB
log4j.appender.SECURITY.MaxBackupIndex = 5
log4j.appender.SECURITY.layout = org.apache.log4j.PatternLayout
```

Note: if `log4j.appender.SECURITY.File` is left empty, the file will be automatically

created at FEDORA_HOME/server/logs/security.log

6. Installation from Source

After unpacking the source distribution, change into the newly extracted directory. To build the installer, at a command prompt, enter

```
ant installer
```

Once the installer target has been run, you will be able to run the Fedora installer application. To start the installer, change to the FEDORA_HOME/dist directory and at a command prompt, enter

```
java -jar fedora-2.2-installer.jar
```

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). For more information of the Installer, see the Installer instructions in Section 2 of this document.

To see a list of other build targets in the source distribution, at a command prompt, enter

```
ant -projecthelp
```

 Last Modified 2007/01/19

fedora.fcfg changes

note how I've added a namespace to existing ones. This MUST be the netid of the leader of your project group.

```
 </param>
 <param name="retainPIDs" value="demo test changeme fedora-bdef fedora-bmech tutorial carl">
   <comment>Namespaces of PIDs to retain during the ingest process. When an
            object is ingested, Fedora normally allocates a unique PID within
            pidNamespace for it regardless of what the object says its PID is. This
            option provides a way to override that behavior on a per-pid-namespace
            basis. If specified, this should be a space-delimited list of pid
            namespaces that will be accepted in the object as-is. Default value is
            &amp;quot;demo test&amp;quot;.</comment>
 </param>
```

Screen clipping taken: 4/16/2007, 8:23 AM

```
  <module role="fedora.server.resourceIndex.ResourceIndex" class="fedora.server.resourceIndex.ResourceInde\
xModule">
    <comment>Supports the ResourceIndex.</comment>
    <param name="level" value="1">
      <comment>(required)
            Index level. Currently, only 0, 1 and 2 are supported
            levels. 0 = off: do not load the ResourceIndex 1 = basic: system
            metadata, RELS-EXT, disseminations 2 = basic + method permutations
            WARNING: changing the level (except to 0) requires
            running the Resource Index Rebuilder.</comment>
    </param>
```

note change to 1 from default of 0

Screen clipping taken: 4/16/2007, 8:27 AM

You MUST do this before ingesting any objects or you will need a rebuild.