

Can I rebuild without taking Fedora offline?

Question

My repository is very large and I cannot afford to take it offline for a long rebuild. Is there a way to do it while keeping the repository running?

Answer

It is possible to reconstitute the database and/or resource index while the repository is running in read-only mode. This is called a *warm rebuild*. People will still be able to access (but not change) items in your repository while the rebuild is taking place, but a quick restart will be required at the end of the process.

The [fedora-rebuild](#) utility was originally designed to run *cold rebuilds* against offline repositories, but with some extra work, you can also use it to rebuild into a new store (resource index or database) while your repository is still handling requests using the original store. At the end of this process, you will need to tell Fedora to use the new store, which requires a restart.

These instructions have been tested in the following configurations:



- Fedora 3.3 with embedded Derby and local Mulgara (Database and Resource Index rebuild)
- Fedora 2.2.3 with embedded McKoi (Database rebuild only)

Instructions:

- [Question](#)
- [Answer](#)
- [Prepare](#)
 - [Create a temporary Fedora Home directory](#)
 - [Configure the new store](#)
- [Disable writes](#)
- [Run the rebuilder](#)
- [Copy temporary configuration to live repository](#)
- [Restart and verify](#)
- [Re-enable writes](#)
- [Clean up](#)

Prepare

Create a temporary Fedora Home directory

This directory will contain a subset of what is currently in your live repository's \$FEDORA_HOME directory in order to support running the rebuilder with an alternate configuration. This can be anywhere you like. Below, we'll assume it's **/tmp/temp-home**, or **C:\temp-home** if you're running Windows.

Unix-based:

```
export ORIG_HOME=$FEDORA_HOME
export TEMP_HOME=/tmp/temp-home
mkdir $TEMP_HOME
mkdir $TEMP_HOME/server
mkdir $TEMP_HOME/server/bin
mkdir $TEMP_HOME/server/config
cp $ORIG_HOME/server/bin/* $TEMP_HOME/server/bin
cp $ORIG_HOME/server/config/* $TEMP_HOME/server/config
```

Windows:

```
set ORIG_HOME=%FEDORA_HOME%
set TEMP_HOME=C:\temp-home
mkdir %TEMP_HOME%
mkdir %TEMP_HOME%\server
mkdir %TEMP_HOME%\server\bin
mkdir %TEMP_HOME%\server\config
copy %ORIG_HOME%\server\bin\*. * %TEMP_HOME%\server\bin
copy %ORIG_HOME%\server\config\*. * %TEMP_HOME%\server\config
```

Configure the new store

1. Ensure that the temporary Fedora is configured to point at the original low level storage locations.
 - a. Open `$TEMP_HOME/server/config/fedora.fcfig`
 - b. Search for `ILowlevelStorage`
 - c. If the "class" attribute ends with `".DefaultLowlevelStorageModule"`:
 - i. Ensure that the `object_store_base` param specifies an absolute directory (by default, it is a relative directory)
 - ii. Do the same for `datastream_store_base`
 - d. If the "class" attribute ends with `".AkubraLowlevelStorageModule"`, no change is needed; Akubra's configuration already specifies an absolute directory.
 - e. If the "class" attribute is anything else, ensure that all path-oriented params are specifying the path in absolute form.
2. If you will be doing a Resource index rebuild, initialize a new triplestore:
 - a. Open `$TEMP_HOME/server/config/fedora.fcfig`
 - b. Locate the datastore element (near the end of the file) corresponding to the triplestore you're using.
 - c. If you're using Mulgara/Kowari:
 - i. If the `remote` param value is `"false"`, change the path param to point to the new, preferred location of your rebuilt resource index data directory. Specify this as an absolute path. For example, `"/opt/fedora/data/resourceIndexRebuilt"`. Do not create this directory yourself; it will be automatically created at the beginning of the resource index rebuild.
 - ii. If the `remote` param value is `"true"`:
 1. Bring up a new instance of Mulgara/Kowari, on a different host and/or port than your existing instance.
 2. Change/add the `"host"` and `"port"` params accordingly.
 - d. If you're using MPTStore:
 - i. Modify the `jdbcURL` param to use a different database name. For example, if it was `"jdbc:postgresql:riTriples"`, change it to `"jdbc:postgresql:riTriplesRebuilt"`
 - ii. Create this new, empty database using the method recommended by your database vendor and ensure that the database user (username) specified in this section of `fedora.fcfig` has full permission on it.
3. If you will be doing a database rebuild, initialize a new database:
 - a. If you're using a non-bundled database (MySQL, Oracle, PostgreSQL):
 - i. Open `$TEMP_HOME/server/config/fedora.fcfig`
 - ii. Locate the datastore element (near the end of the file) corresponding to the database you're using.
 - iii. Modify the `jdbcURL` param to use a different database name. For example, if it was `"jdbc:postgresql:fedora3"`, change it to `"jdbc:postgresql:fedora3rebuilt"`
 - iv. Create this new, empty database using the method recommended by your database vendor and ensure that the database user (dbUsername) specified in this section of `fedora.fcfig` has full permission on it.
 - b. If you're using the bundled Derby database:
 - i. Open `$TEMP_HOME/server/config/fedora.fcfig`
 - ii. Search for `jdbc:derby`
 - iii. Change the `jdbcURL` param to point to the new, preferred location of your rebuilt database. For example, if the value is currently `"jdbc:derby:/opt/fedora/derby/fedora3:create=true"`, change it to `"jdbc:derby:/opt/fedora/derby/fedora3-rebuilt"`. Do not create this directory yourself; it will be automatically created at the beginning of the database rebuild.
 - c. If you're using the bundled McKoi database:
 - i. Open `$TEMP_HOME/server/config/fedora.fcfig`
 - ii. Search for `jdbc:mckoi`
 - iii. Change the `jdbcURL` param to point to the new, preferred location of your rebuilt database. For example, if the value is currently `"jdbc:mckoi:local:///opt/fedora/mckoi1.0.3/db.conf?create_or_boot=true"`, change it to `"jdbc:mckoi:local:///opt/fedora/mckoi1.0.3-rebuilt/db.conf?create_or_boot=true"`
 - iv. Create this new directory. For example, `"mkdir /opt/fedora/mckoi1.0.3-rebuilt"`
 - v. Copy all files (but not subdirectories) from the original mckoi directory to this new one. For example, `"cp /opt/fedora/mckoi1.0.3/* /opt/fedora/mckoi1.0.3-rebuilt"`.

Disable writes

See [How do I turn off API-M?](#)

Run the rebuilder

First, make sure your `FEDORA_HOME` and `CATALINA_HOME` environment variables are set correctly:

Unix-based:

```
export FEDORA_HOME=$TEMP_HOME
echo $FEDORA_HOME
echo $CATALINA_HOME
```

Windows:

```
set FEDORA_HOME=%TEMP_HOME%
echo %FEDORA_HOME%
echo %CATALINA_HOME%
```

The echoed value of `FEDORA_HOME` should be the path to your temporary Fedora Home directory (e.g. `/tmp/temp-home`) and the echoed value of `CATALINA_HOME` should be the path to your live repository's tomcat directory (e.g. `/opt/fedora/tomcat`).

Next, run the rebuilder [as described here](#), but don't stop the server.

If you need to rebuild *both* the database and resource index, make sure you have configured both, and *rebuild the database first*, followed by the resource index.

Copy temporary configuration to live repository

- Make a backup of `$ORIG_HOME/server/config/fedora.fcfg`
- Copy `$TEMP_HOME/server/config/fedora.fcfg` to `$ORIG_HOME/server/config/fedora.fcfg` (replacing the existing file)

Restart and verify

- Set the `FEDORA_HOME` environment variable back to the original value (`$ORIG_HOME`)
- Run `$CATALINA_HOME/bin/shutdown.sh` (`%CATALINA_HOME%\bin\shutdown.bat` if running Windows)
- Run `$CATALINA_HOME/bin/startup.sh` (`%CATALINA_HOME%\bin\startup.bat` if running Windows)
- Check `$FEDORA_HOME/server/logs/fedora.log` for any sign of errors at startup.
- Try some queries on the `/fedora/search` and `/fedora/risearch` web interfaces and make sure they behave as expected.

Re-enable writes

See [How do I turn off API-M access?](#)

Clean up

You can now safely remove the original database and/or resource index data to recover disk space. You can also remove the `$TEMP_HOME` directory.