

2. Before You Begin

Before you begin, you can perform some tasks to ensure the efficiency and success of your upgrade process.

- [Optimizing your NiFi Directory Structure for Upgrade](#)
- [Preserving your custom processors](#)
- [Preserving customized NAR files](#)

Optimizing your NiFi Directory Structure for Upgrade

When you install NiFi and perform the minimum configuration requirements by editing the `nifi.properties` file, your installation directory contains several files and directories. Assuming you have installed into an `opt/` directory, the NiFi installation directory structure is similar to:

```
/
--> opt/
    |--> nifi-current_version
        |--> LICENSE
        |--> NOTICE
        |--> README
        |--> bin/
        |--> conf/
        |--> docs/
        |--> lib/
        |--> work/
        |--> content_repository/
        |--> flowfile_repository/
        |--> logs/
        |--> provenance_repository/
```

To optimize the NiFi directory structure for upgrade:

1. Move the following directories out of the NiFi installation directory:

- Database repository
- All content repositories
- Flowfile repository
- NiFi log repository
- All provenance repositories

See the [System Administrator's Guide](#) for information on options for creating these repositories to optimize performance and stability.

2. Move static items that do not change from version to version out of the `conf` directory. These items include:

- `flow.xml.gz`
- `authorized-users.xml`
- The templates directory

Note: NiFi 1.0.0 no longer contains `authorized-users.xml`. Instead, it contains `authorizations.xml` and `users.xml`. During upgrade, preserve `authorized-users.xml` so that NiFi 1.0.0 can use it to populate the values of the `authorizations.xml` and `users.xml` files.

A good directory structure that supports easy upgrading looks similar to:

```
/
|--> opt/
    |--> nifi-current_version
        |--> LICENSE
        |--> NOTICE
        |--> README
        |--> bin/
        |--> conf/
        |--> docs/
        |--> lib/
        |--> work/
    |--> Configuration_resources/
        |--> authorized-users.xml
        |--> flow.xml.gz
        |--> templates/
        |--> custom_lib/
    |--> database_repository/
    |--> content_repository_1/
    |--> content_repository_N/
    |--> flowfile_repository/
    |--> NiFi-logs/
    |--> provenance_repository_1/
    |--> provenance_repository_N/
```

Important: Upgrading from NiFi 0.x to 1.x converts your database, content, flowfile, log, and provenance repositories automatically. Once upgraded, you can not revert the repositories back to 0.x format.

Preserving your custom processors

If you have written any custom NARs, first test and validate them in a separate new NiFi release sandbox prior to upgrading. The verified NARs can be preserved during the upgrade by storing them in a centralized location as follows:

1. Create a second library directory, called `custom_lib`.
2. Move your custom NARs to this new lib directory.
3. Add a new line to the `nifi.properties` file to specify this new lib directory:

```
nifi.nar.library.directory=./lib
```

```
nifi.nar.library.directory.custom=/opt/configuration_resources/custom_lib
```

Preserving customized NAR files

If you have modified any of the default NAR files, an upgrade will overwrite these changes. To preserve the customization:

1. Identify and save the changes you have made to the default NAR files in your existing instance.
2. Perform your upgrade to NiFi 1.0.0.
3. Redo the changes you have made to the NAR files in your NiFi 1.0.0 instance.