

Choose an Installation Scenario

Contents

- 1 [Choose a platform](#)
- 2 [Choose Repository](#)
- 3 [Choose the type of database](#)
 - 3.1 [Derby Database](#)
 - 3.2 [MySQL Database](#)
 - 3.3 [PostgreSQL Database](#)
- 4 [Choose a JMS broker](#)
- 5 [Java](#)
- 6 [Choose the set of machines taking part in the installation/deployment](#)
- 7 [Choose other plug-ins](#)

Choose a platform

NetarchiveSuite can be installed in a number of different ways, with varying numbers of machines on different sites. There is a number of separate applications in play, most of which can be put on separate machines as needed. To keep clear what is necessary for which setups, we will consider the following types of setup:

- **A. Single-machine setup.** This corresponds to the setup used in the [Quick Start Manual](#), where all applications run on the same machine, and file transfer are done by simply copying files locally. It is the simplest setup, but does not scale very well.
- **B. Single-site setup.** In this scenario, multiple machines are involved, necessitating file transfer between machines and multiple installations of the code. However, the machines are expected to be within the same firewall, so port setup should be no problem.
- **C. Single-site setup with duplicate archive.** This expands on the single-site set-up in that more than one copy of the archived files are used, using the concept of separate "Replica" to indicate the duplicates.
- **D. Multi-site setup.** When more than one site (physical location) is involved, separated by firewalls, extra issues of opening ports and specifying the correct site come into play. This is the most complex scenario, but also more secure against systematic errors, hacking, and other disasters.

Choose Repository

Scenario A and B from section *Choose a platform* involves having a local archive repository without means of bitarchive replicas. This is configured by a plug-in (please refer to [Configure Plugins](#) in the Configuration Manual).

Scenarios C and D from section *Choose a platform* involves having a distributed bitarchive replicas. In these scenarios we have at least two bitarchive replicas. The Replica information must be configured before deployment either in the local settings file or included in the deploy configuration file for your system (please refer to [Configure Repository](#) in the Configuration Manual).

Choose the type of database

The !NetarchiveSuite can use three types of database:

- Derby database (default)
- PostgreSQL
- MySQL database

By default, the NetarchiveSuite uses an external Derby. Note that from release 3.14.* the choice of an embedded Derby database has been removed to allow several applications to access the database simultaneously. The choice of the database is further described in the section on [Plugins](#).

Besides the configuration of the plug-in (where Derby database is the default), there are additional installations and configurations that must be done as described below.

Note that `<deployInstallDir>`, `<deployDatabaseDir>` and `<deployMachine>` will be used as reference to items corresponding deploy settings. The meaning of them are described in the [Deploy Settings](#).

Derby Database

If you want to use a Derby database, you have to make it run as a separate process.

1. Start Derby separately
2. `cd "directory with the extracted database" (e.g. <deployInstallDir>/<deployDatabaseDir>)`
3. `export CLASSPATH=<deployInstallDir>/lib/db/derbynet-10.4.2.0.jar:<deployInstallDir>/lib/db/derby-10.4.2.0.jar`
4. `java org.apache.derby.drda.NetworkServerControl start -p port`

The default port is 1527.

For the NetarchiveSuite to use this kind of external database, you need to

- Set the setting `settings.common.database.class` to `dk.netarkivet.harvester.datamodel.DerbyServerSpecifics`.
- Set the setting `settings.common.database.url` to `jdbc:derby://<deployMachine>:1527/fullhddb` (substitute the server host for `<deployMachine>` and 1527 for correct port)
- Need to add a permission to the policy file used by your installation, if you use security (see below). The following will allow NetarchiveSuite to access a Derby database on port 1527.

```
grant {
    permission java.net.SocketPermission "127.0.0.1:1527",
        "connect, resolve";
};
```

Firewall note: You will need to allow the GUIApplication and the !HarvestTemplateApplication to be able to access port 1527 on the server where you run the database.

More details on using Derby as a server are available on <http://db.apache.org/derby/docs/dev/adminguide/cadminov825266.html> the derby pages.

MySQL Database

If you want to use a MySQL database, you have to:

- Set the setting `settings.common.database.class` to `dk.netarkivet.harvester.datamodel.MySQLSpecifics`
- Set the setting `settings.common.database.url` correctly: `jdbc:mysql://localhost/fullhddb?user=root&password=secret` (substitute the server host for localhost, and username/password for root/secret)
- Install the MySQL database (v. 5.0.X) on a machine of your choice
- Download a `mysql-connector-java-5.0.X-bin.jar` from <http://dev.mysql.com/downloads/connector/j/5.0.html>
- Add a permission to the policy file used by your installation, if you use security. The following will allow !NetarchiveSuite to access MySQL on localhost on the default port 3306.

```
grant {
    permission java.net.SocketPermission "127.0.0.1:3306",
        "connect, resolve";
};
```

Firewall note: You will need to allow the GUIApplication and the !HarvestTemplateApplication to be able to access port 3306 on the server where you run the database.

This jar must then be added to the classpath for the applications, that accesses the database: GUIApplication and HarvestTemplateApplication

You can do this manually, when starting these applications. Alternatively, you can add the `mysql-connector-java-5.0.X-bin.jar` to the `lib/db` directory, and modify `build.xml` accordingly:

- Add a line `db/mysql-connector-java-5.0.X-bin.jar` to the property `jarclasspath` just below the line `db/derby-10.1.1.0.jar`.
- Add a line `<include name="db/mysql-connector-java-5.0.X-bin.jar"/>` below `include name="db/derby-10.1.1.0.jar/>`

You can then generate a new !NetarchiveSuite zipball with

```
ant releasezipball
```

This assumes, that you have downloaded the source distribution of the NetarchiveSuite.

PostgreSQL Database

To be written.....

Choose a JMS broker

NetarchiveSuite requires a JMS broker to run. The only type of JMS broker supported at this time is the SunMQ broker and its open source counterpart Open Message Queue.

The installation and start-up of a JMS broker is described in [Appendix A](#).

For description of how to configure the JMS broker, please refer to the [Configure JMS Broker](#).

Firewall note: The machine that runs the JMS broker must be accessible from all machines in the installation on not only port 7676, but also port 33700 (from RMI).

Java

All machines must run Java version 1.6.0 or higher.

Choose the set of machines taking part in the installation/deployment

When you have chosen a scenario, you must decide on the number of machines, you want to use in the deployment of the !NetarchiveSuite. For scenario A, the answer is of course one. For the scenarios B, C, and D, the answer is more complicated.

An extra complication is added by installing the system at two different physical location (here referred as EAST and WEST). The distinction between different physical location are relevant if the system is installed at two different institutions with firewalls between them.

At the Danish installation, we operate with 4 kinds of machines:

- Admin machine (one server): Here we deploy one or more !BitarchiveMonitorApplications (one for each bitarchive Replica), one !ArcrepositoryApplication, one GUIApplication, and a JobManagerApplication, which takes care of job scheduling.
- Harvester machines (one or more): Here we deploy the !HarvesterControllerApplications.
- Bitarchive machines (one or more): These machines only run one !BitarchiveApplication each (there must be at least one for each bitarchive Replica).
- Access servers (one or more): On these machines, we have the !ViewerproxyApplication enabling us to browse in already stored webpages, and the !IndexServerApplication. The latter must only be installed on one of the access-servers, as there can only be one in the system.

Apart from the !HarvestControllerApplications, there is no requirement that the applications are placed like this, but we will use it as an example throughout the rest of the manual. In the standard set-up used in our test-environment, we have 9 machines:

- 1 bitarchive server (on physical location WEST)
- 2 bitarchive servers (on physical location EAST)
- 1 admin machine (placed on physical location EAST)
- 1 harvester-machines (placed on physical location WEST)
- 2 harvester-machines (placed on physical location EAST)
- 1 access server (placed on physical location WEST)
- 1 access server (placed on physical location EAST)

Choose other plug-ins

Except from the plug-ins described in this section, the installation of plug-ins only consist of the configuration of them.

