

Configuration Basics - NetarchiveSuite Settings

Contents

- [Setting basics](#)
- [Setting keys with multiple values](#)
- [Default Settings](#)
 - [Common part](#)
 - [Harvester part](#)
 - [Archive part](#)
 - [Monitor part](#)
 - [Wayback part](#)
 - [Plug-in default settings](#)

It is possible to control much of the behaviour of NetarchiveSuite tools and applications using settings. Some settings need to be updated for a distributed system to work, others work best with their default settings.

A complete NetarchiveSuite installation consists of a number (anywhere between a few and several hundred) Java applications communicating with each other via JMS. Each application has its own settings - typically defined in a single xml file, with the possibility to override values from the command line. The NetarchiveSuite Deployment Framework, described in the [Installation Manual](#), provides a mechanism for generating these settings file for each application from a single hierarchically-structured deployment-xml file. In this manual we only consider the structure and contents of the individual per-application settings files.

Below, the basics of settings and default settings are described.

Setting basics

All NetarchiveSuite applications are based on the same type of configuration: Keys can be mapped to values, and the mappings can be set either in a settings file written in XML, or on the command line. If no value is specified for a given configuration key, a default value is used.

The keys are defined in a hierarchy. When naming the keys, we separate the levels in a key with dots, for instance:

```
settings.common.http.port=8076
```

When describing the same keys in XML, we use the XML hierarchy:

```
<settings>
  <common>
    <http>
      <port>8076</port>
    </http>
  </common>
</settings>
```

Setting keys with multiple values

Some settings allow a list of values, rather than just one value. For instance:

```
<settings>
  <archive>
    <bitarchive>
      <baseFileDir>/mnt/storage1</baseFileDir>
      <baseFileDir>/mnt/storage2</baseFileDir>
    </bitarchive>
  </archive>
</settings>
```

It is only possible to specify multiple values using configuration files. This cannot be done on the command line.

If you specify more than one settings file, the first settings file to contain a value for the key specifies **all** values. Values from the settings files will not be merged.

As an example, consider the following two settings files:

settings1:

```
<settings>
  <archive>
    <bitarchive>
      <baseFileDir>/mnt/storage1</baseFileDir>
      <baseFileDir>/mnt/storage2</baseFileDir>
    </bitarchive>
  </archive>
</settings>
```

settings2:

```
<settings>
  <archive>
    <bitarchive>
      <baseFileDir>/mnt/storage3</baseFileDir>
      <baseFileDir>/mnt/storage4</baseFileDir>
    </bitarchive>
  </archive>
</settings>
```

The following command will give the value

```
/mnt/storage5
```

because the command-line overrides the value(s) specified in any settings file:

```
java -Ddk.netarkivet.settings.file=settings1.xml:settings2.xml
-Dsettings.archive.bitarchive.baseFileDir=/mnt/storage5
dk.netarkivet.common.webinterface.GUIApplication
```

The following command will give the values

```
/mnt/storage1
```

and

```
/mnt/storage2
```

:

```
java -Ddk.netarkivet.settings.file=settings1.xml:settings2.xml
dk.netarkivet.common.webinterface.GUIApplication
```

The following command will give the values

```
/mnt/storage3
```

and

```
/mnt/storage4
```

:

```
java -Ddk.netarkivet.settings.file=settings2.xml:settings1.xml
dk.netarkivet.common.webinterface.GUIApplication
```

Default Settings

The NetarchiveSuite package includes default XML setting files with values for the settings that are used to initialize classes if they are not overwritten by separate settings files or on the command line (please refer to [Installation Manual](#)).

The NetarchiveSuite has five main levels under the top `settings` level:

- common
- harvester
- archive
- monitor
- wayback

All settings are defined within these five main levels. In addition there is a separate set of settings used only by the deploy application.

The NetarchiveSuite package includes default values for most defined settings. These are defined in XML setting files that are used to initialize classes, one for each main level and one for each plug-in. (TODO: Name the exceptions). The default settings files can be found in the NetarchiveSuite source tree. For each setting there is a corresponding Java variable or constant, and the settings are documented in Javadoc in the relevant classes. The settings file and the relevant classes are as follows

Settings File
<code>./common/common-core/src/main/resources/dk/netarkivet/common/settings.xml</code>
<code>./harvester/heritrix3/heritrix3-controller/src/main/resources/dk/netarkivet/harvester/heritrix3/settings.xml</code>
<code>./archive/archive-core/src/main/resources/dk/netarkivet/archive/settings.xml</code>
<code>./monitor/monitor-core/src/main/resources/dk/netarkivet/monitor/settings.xml</code>
<code>./wayback/wayback-indexer/src/main/resources/dk/netarkivet/wayback/settings.xml</code>
<code>./harvester/harvester-core/src/main/resources/dk/netarkivet/harvester/settings.xml</code>

The meanings of the different settings are documented in the javadoc of the associated setting classes as listed below.

Common part

In the common part of the settings, we have general purpose settings (e.g. `settings.common.tmpDir`, `settings.common.http.port`), and settings, that allow us to select plug-ins and their associated arguments (e.g. `settings.common.RemoteFile.class`, `settings.common.jms.broker`, `settings.common.arcrepositoryClient`, and `settings.common.indexClient.class`). Furthermore, there are other dedicated common default values for specific plug-in classes defined in the following setting files. All of these are referred to as part of the common part, but are defined with the plug-in itself. Please see section [#Plug-in Default Settings](#).

Harvester part

In the harvester part of the settings, we have settings configuring the harvesting process: scheduling, job splitting etc. Most of these settings are used by the scheduler in `DefinitionsSiteSection` of the `GUIApplication`. `HarvestSettings` is primarily used for generic settings related to harvesting, while `Heritrix3Settings` is used for settings that are specific to `Heritrix3`.

Archive part

In the archive part of the settings, we have settings related to archive-access (e.g. certain timeouts, replicas and their credentials are defined here). Also behaviour of the `BitarchiveApplications` is set here.

Monitor part

In the monitor part of the settings, we have settings for the monitoring shown in the System State in the form of e.g. JMX user name and password and number of shown logged lines.

Wayback part

This defines settings for the workflow for automatic indexing of webpages for use by Wayback. It also includes some settings for the plugins to Wayback which allow it to communicate directly with the `NetarchiveSuite` distributed repository.

Plug-in default settings

At the moment, the following plugins have associated default settings defined in the following classes, where their documentation can be found in the javadoc:

- `EmailNotifications.java` with defaults in `dk.netarkivet.common.utils.EmailNotificationsSettings.xml`.
- `FTPRemoteFileSettings.xml` with defaults in <https://sbforge.org/svn/netarchivesuite/trunk/src/dk/netarkivet/common/distribute/FTPRemoteFileSettings.xml>.
- `HTTPRemoteFile.java` with defaults in `dk.netarkivet.common.distribute.HTTPSRemoteFileSettings.xml`.
- `HTTPSRemoteFile.java` with defaults in <https://sbforge.org/svn/netarchivesuite/trunk/src/dk/netarkivet/common/distribute/HTTPSRemoteFileSettings.xml>.
- `JMSConnectionSunMQ.java` with defaults in `dk.netarkivet.common.distribute.JMSConnectionSunMQSettings.xml`.
- `JMSArcRepositoryClient.java` with defaults in `dk.netarkivet.archive.arcrepository.distribute.JMSArcRepositoryClientSettings.xml`.
- `IndexRequestClient.java` with defaults in `dk.netarkivet.archive.indexserver.distribute.IndexRequestClientSettings.xml`.

