Tools in the Harvester Module

Contents

dk.netarkivet.tools.harvester.CreateCDXMetadataFile

Given a specific jobID (e.g. 42) and a harvestnamePrefix this tool can be used to create a metadata-1.warc containing the CDX-entries for all (w)arc-files belonging to that job.

prequisites and arguments

You need to specify the repositoryclient used for accessing your archived-data. If you use the default client JMSArcRepositoryClient you also need to specify the archive replica you will use (defined by setting "settings.common.useReplicaId"), the environmentname, the applicationName, the applicationInstanceId. These can all be defined on the commandline as overrides to the default values, or defined in a local settings.xml file.

Needed jarfiles in the classpath: dk.netarkivet.harvester.jar, dk.netarkivet.archive.jar (if using default repositoryclient)

The tool only has at least two arguments --jobID 42 --harvestnamePrefix 42-3

Optional argument is the -a or -w to choose the metadata format. By default the program outputs metadata warc file, but the -a option makes the program write an metadata arc file.

Sample usage of this tool

```
export INSTALLDIR=/home/test/netarchive
CLASSPATH=$INSTALLDIR/lib/dk.netarkivet.harvester.jar:
export CLASSPATH=$CLASSPATH:$INSTALLDIR/lib/dk.netarkivet.archive.jar
export OPTS=-Ddk.netarkivet.settings.file=localsettings.xml

java $OPTS dk.netarkivet.harvester.tools.CreateCDXMetadataFile [-a|-w]
--jobID 42 --harvestnamePrefix 42-3
```

dk.netarkivet.harvester.tools.CreateLogsMetadataFile (deprecated)

In the beginning, the metadata-1.arc files did not include the Heritrix logs. This tool was made to allow us to make a metadata-2.arc file that contains the heritrix logs associated with a given job.

Consider this tool deprecated. For further information see the javadoc for this method. Note that settings file mentioned below need to contain proper values for the harvesting metadata settings:

```
<metadata>

<heritrixFilePattern>.*(\.xml|\.txt|\.log|\.out)</heritrixFilePattern>
                <reportFilePattern>.*-report.txt</reportFilePattern>
                <logFilePattern>.*(\.log|\.out)</logFilePattern>
            </metadata>
```

Sample usage of this tool

```
export INSTALLDIR=/home/test/netarchive
export CLASSPATH=$INSTALLDIR/lib/dk.netarkivet.harvester.jar
export OPTS=-Ddk.netarkivet.settings.file=localsettings.xml
java $OPTS dk.netarkivet.harvester.tools.CreateLogsMetadataFile \
jobid-harvestid.txt jobsdir
```

dk.netarkivet.harvester.tools.HarvestTemplateApplication

This tools enables you to create (create command), download (download command), update (update command) and show (showall command) the existing templates.

prequisites and arguments

You need to point to a settings file with connection information for your harvest database. In a standard NAS deployment, use the INSTALLDIR/conf/settings_GUIApplication.xml

Sample usage of this tool

```
export INSTALLDIR=/home/test/netarchive
export CLASSPATH=$INSTALLDIR/lib/dk.netarkivet.harvester.jar
export
OPTS=-Ddk.netarkivet.settings.file=$INSTALLDIR/conf/settings_GUIApplicatio
n.xml

java $OPTS dk.netarkivet.harvester.tools.HarvestTemplateApplication
<command> <args>
```

The different <command> <args> possibilities:

1. `create <template-name> <xml-file for this template>`
2. `download [<template-name>]*`
3. `update <template-name> <xml-file to replace this template>`
4. `showall`

Note that with the download command, you can either download all templates in one go (with no args), or select the names of the templates to download (separated by space)

dk.netarkivet.harvester.tools.HarvestdatabaseUpdateApplication

This tools enables you to update the tables in the harvestdatasbase to the versions required this release of NetarchiveSuite. It should be run after installing the software, but before starting the NetarchiveSuite applications.

prequisites and arguments

You need to point to a settings file with connection information for your harvest database. In a standard NAS deployment, use the INSTALLDIR/conf/settings_GUIApplication.xml

And the harvest database needs to be running as well.

Sample usage of this tool

First, the harvestdatabase is started, if it isn't up and running already.
Then the update tool is executed:

```
export INSTALLDIR=/home/test/netarchive
export CLASSPATH=$INSTALLDIR/lib/dk.netarkivet.harvester.jar
java
-Ddk.netarkivet.settings.file=$INSTALLDIR/conf/settings_GUIApplication.xml
\
dk.netarkivet.harvester.tools.HarvestdatabaseUpdateApplication
```

Finally, the harvestdatabase is shutdown, if you're using derby as database.