NetarchiveSuite Quick Start Manual

Printer friendly version

Contents

Introduction

edit

This manual provides instructions for quickly getting a basic NetarchiveSuite system up and running. It uses a pre-built script that starts all components on the same machine. This allows you to start experimenting with the functionality without having to do any more setup than absolutely necessary.

It should not require much technical skill to evaluate the system. What it requires is a computer running a Linux operating system and with sun java 1.5 or above installed. You do not need root/administrator access.

Going through this quick start should take about an hour.

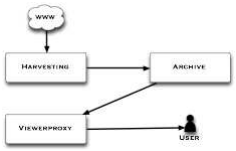
System overview

edit

The primary function of the NetarchiveSuite is to plan, schedule and archive web harvests of parts of the internet. We use Heritrix as our webcrawler. The NetarchiveSuite can organize three different kinds of harvests:

- Event harvesting (organize harvests of a set of domains related to a specific event, e.g. 9/11, Royal Weddings, Elections and so on).
- Selective harvesting (recurrent harvests of a set of domains).
- Snapshot harvesting (organizing a complete snapshot of all known domains)

The NetarchiveSuite is split into three main modules corresponding to harvesting, archiving and accessing via viewerproxy.



Please refer to the overview description for more details.

Download and installation

edit

For a quick start, we have prepared a bash script that starts all the necessary components on one machine. We will use this script throughout this quickstart manual to allow you to get a feel for what the system can do and how it works without having to deal with issues of distributing to other servers.

Base system required

For the quick startup, NetarchiveSuite requires

- a Linux system.
  Note that for the quickstart, you must be able to run a browser on the machine that you run the system on - this is an artifact of the quickstart system and is not the case in the full system. Root access is not required.
- Sun Java SE (Standard Edition) JDK version 1.5.0_06 running on the Linux system.
  Newer versions of Java may work, but have not been tested. Older versions of Java will **not** work correctly. The current download version of Sun Java 5 SE is "JDK 5.0 Update 12"

To check that you have the right version of Java do the following

- start a terminal login to the linux system as a ordinary user
- check java version is version 1.5.0_06 (or higher) by writing:
  `$ java -version`
  you should then see

```
linux>java -version
java version "1.5.0_06"
Java(TM) 2 Runtime Environment, Standard Edition (build
1.5.0_06-b05)
Java HotSpot(TM) Client VM (build 1.5.0_06-b05, mixed mode, sharing)
```

Downloading

Download and unzip of the newest release is described here

- start a terminal login to the linux system as a ordinary user in a bash shell
- make a directory for the download e.g. directory `~/netarchive`
  `$ mkdir ~/netarchive`

- start a web browser, e.g. Firefox
- follow the registration and download instructions on Get NetarchiveSuite and save the download file to the directory you created earlier
- go to the directory
  ```
  $ cd ~/netarchive
  ```
- unzip the binary package
  ```
  $ unzip NetarchiveSuite*.zip
  ```

JMS

NetarchiveSuite uses JMS for inter-process communication. JMS is the Java Messaging Service, which provides asynchronous communication between processes. You do not need any knowledge of JMS to use NetarchiveSuite.

Currently only the open-source version of Sun's JMS implementation is supported, since some functionality of other implementations does not match our assumptions well.

To download and install it, do the following:

- open this link in a browser window • https://mq.dev.java.net/downloads.html
- click the Linux Link under version 4.1 to download a file `mq4_1-binary-Linux_X86-20070518.jar` (or later)
- save the download file to the created directory you created earlier e.g. `~/netarchive`
- go to the directory
  ```
  $ cd ~/netarchive
  ```
- unpack the jar file (this creates a directory mq and three files with licensing information)
  ```
  $ jar xvf mq4_1-binary-Linux_X86-20070518.jar
  ```
- run imqbroker in order to create settings file
  ```
  $ chmod +x ./mq/bin/imqbrokerd
  $./mq/bin/imqbrokerd
  ```
- check that `imqbrokerd` starts and that the last message is `"Broker <localhost>:7676 ready"`
- stop imqbroker by pressing `control-C`
- edit settings to allow for enough listeners to a queue by doing
  - edit `~/netarchive/mq/var/instances/imqbroker/props/config.propertie`
  - uncomment and specify count=20 for listeners by changing line

    `"#  imq.autocreate.queue.maxNumActiveConsumers"`

    ->

    `"  imq.autocreate.queue.maxNumActiveConsumers=20"`

Configuration

Assuming the NetarchiveSuite files were unzipped to the directory `~/netarchive` as described above, you must do the following to configure the NetarchiveSuite for your system:

- make the three shell-scripts executable:
  ```
  $ chmod +x ~/netarchive/scripts/simple_harvest/*.sh
  ```

For the simple harvest setup, the startup script needs to know a few paths. You can either do this by exporting the environment variables in your shell or by changing the harvest script.

The following describes how to set the environment variables:

- set `JAVA` to point to your java installation directory, e.g. `/usr/java/jdk1.5.0_06`.
  ```
  $ export JAVA=/usr/java/jdk1.5.0_06
  ```
- set `IMQ` to the full path of the executable of the Sun JMS broker, e.g. `~/netarchive/mq/bin/imqbrokerd`
  ```
  $ export IMQ=~/netarchive/mq/bin/imqbrokerd
  ```

Note that these environment variables need to be set every time you log in, if you wish to use the quickstart scripts. You can also edit the file `harvest.sh` and set the variables there, if you wish the settings to be persistent.

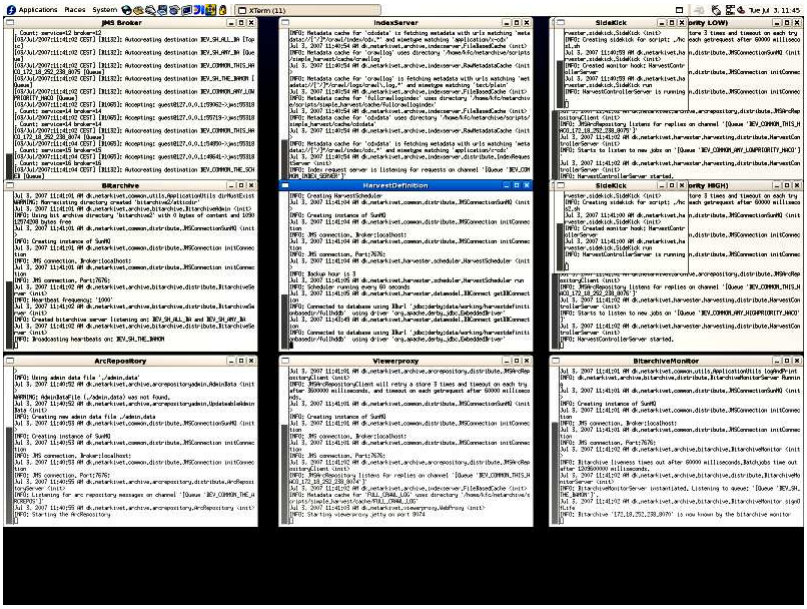
Starting the system

Note: Starting the script clears all data from previous runs. This is a feature of it being a "playground" setup.

To start the program do the following:

- start a terminal login to the linux system as a ordinary user in a `bash` shell
- set the environment variables `JAVA` and `IMQ` as described above, if they are not set already.
- go to the simple harvest directory and run program
  ```
  $ cd ~/netarchive/scripts/simple_harvest
  $ ./harvest.sh
  ```
  note that it will start several xterm windows, one for each application running (including JMS). All the xterms will display the logs of their applications (see figure below) -- you can follow how they start up and when they are ready to use.
- wait until the xterm with the title `HarvestDefinition` has stopped writing messages. It should include towards the end one saying
  ```
  "Scheduler running every 60 seconds"
  ```

A typical set of windows looks like this:



- start a web browser by e.g. `$ mozilla` Note that it is important that the browser is started on the same machine as the simple harvest script is run on.
- write url in the started browser http://localhost:8073/HarvestDefinition
  You can now see the webinterface in the browser

Stopping the system

When you are done experimenting, you can use the scripts described below to stop the programs. After that, it should be possible to restart from scratch by following the instructions described above under section "Starting simple_harvest version".

Do the following:

```
$ cd ~/netarchive/scripts/simple_harvest
$ ./killhard.sh
```

Running a simple harvest

edit

The system is now up and running, and you can try out the harvesting and archiving capabilities.

This section will guide you through the steps needed to

- harvest and store a domain
- browse the harvested material in a browser

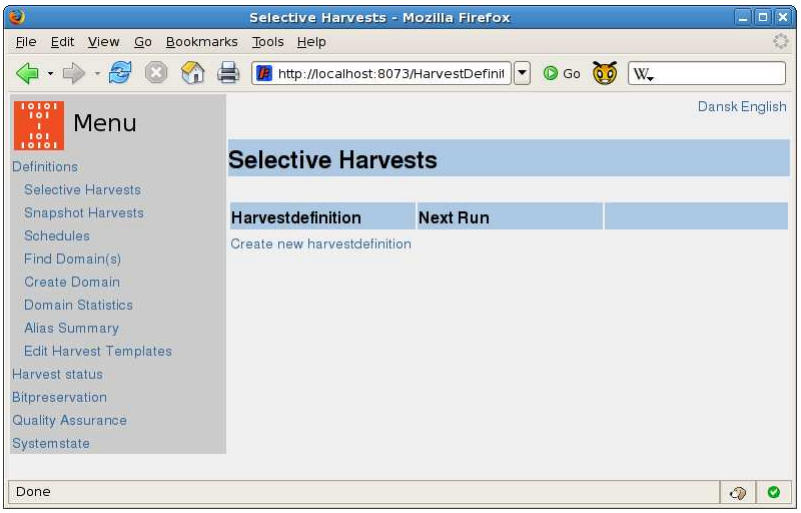
Setting up the harvest

Start the program as described in section "Starting simple_harvest version".

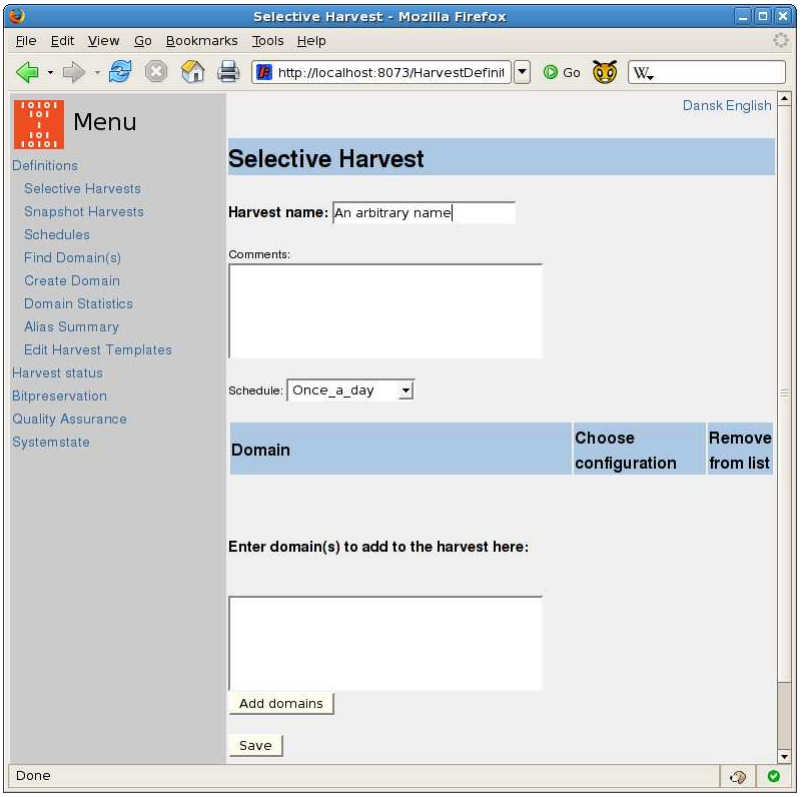
Open http://localhost:8073/HarvestDefinition in a browser on the local machine.

You can now define a new harvest.

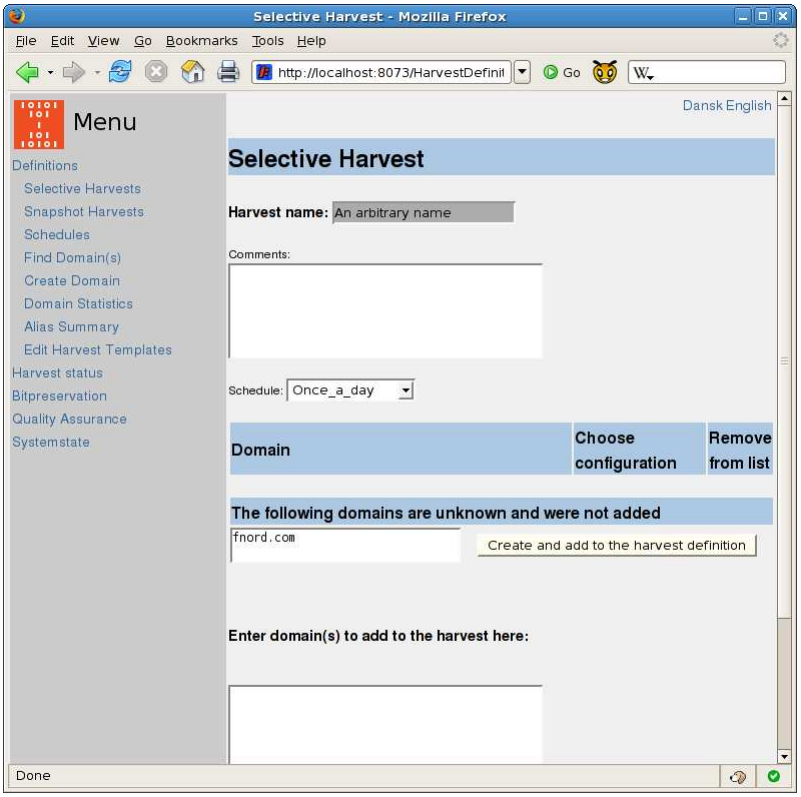
Click 'Selective Harvests' under menu 'Definitions'



Click 'Create new harvest definition' under the (empty) table of existing harvests.



Enter an arbitrary name for the harvest in the top. Enter some second-level domain name (e.g., fnord.com) in the box and press 'Add domains'. Preferably the domain should be one that you know you have permission to harvest. You can add more domains if you want by repeating the procedure, but in this example we will only add one domain.



Since the domain didn't exist in the database, the system suggests you add it. Click 'Create and add to harvest definition'. You can now click 'Save' on the 'Selective Harvest' page

Now you have defined a harvest definition for this domain. It will however not start a harvest before it is changed to active state.

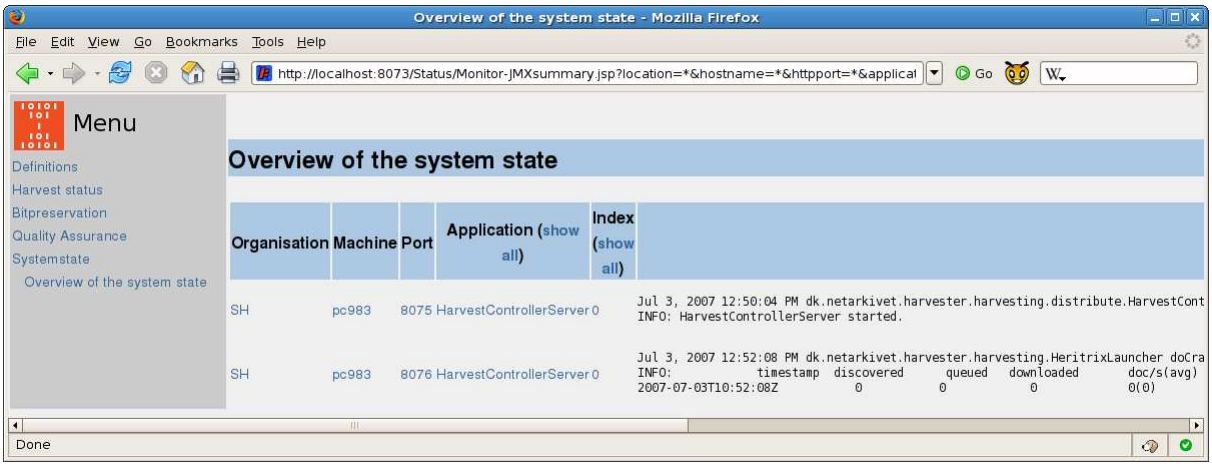
Click 'Activate' for the newly defined harvest.

The harvest definition will generate harvest jobs.

Go to the Job Status page by clicking 'Harvest status'. Refresh it periodically until a job appears and changes to state "Started". This should take no more than two minutes. At this point, a harvester has started harvesting, using the Heritrix web harvester.

Now you can monitor the system state for what is going on in the various components. That way you can see how the harvester is doing with the job:

Go to the System Status page by clicking 'Systemstate'. Click on the application HarvestControllerServer. The most recent log record will give status information from Heritrix. You can find more application information by clicking on 'Show all' in the Index column.



Use the System Status and Job Status pages to monitor your job.

Go to the Job status page by clicking 'Harvest status'. It will take a little while for the job to finish and to upload the harvested files to the NetarchiveSuite archive. Refresh the page until the job changes state to "Done".

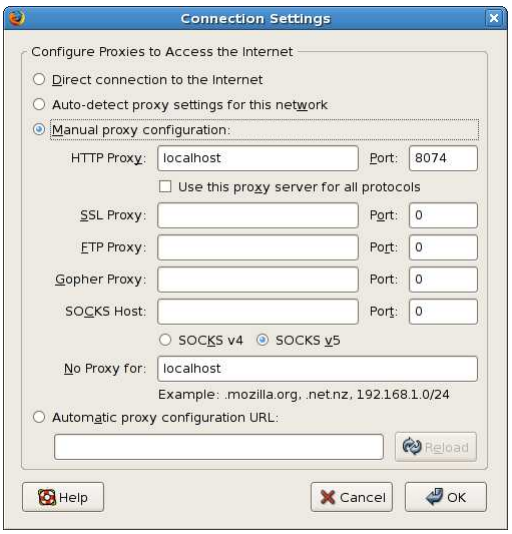
Viewing the results

Harvested jobs can be viewed in an ordinary browser. Part of the NetarchiveSuite is a "viewerproxy", that integrates with your browser to show you harvested material for

Quality Assurance.

Once that some web pages have been harvested, you can use the viewerproxy part to view them.

Set your browser to use localhost:8074 as a proxy for everything except localhost. An example of a browser setting window is given in the illustration.

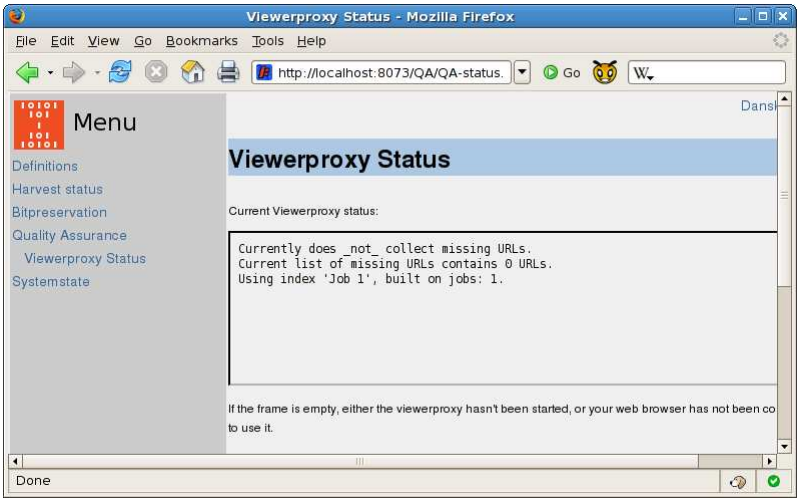


The viewerproxy has now been set up. Before it is ready, it needs to know which material you wish to browse.

Go to the 'System Status' page, and click on the link with the Job Id.

Click on 'Select this job for QA with viewerproxy'.

This will make the viewerproxy browse in this job. It will take it a while to generate an index. It will then go to the viewerproxy status page.

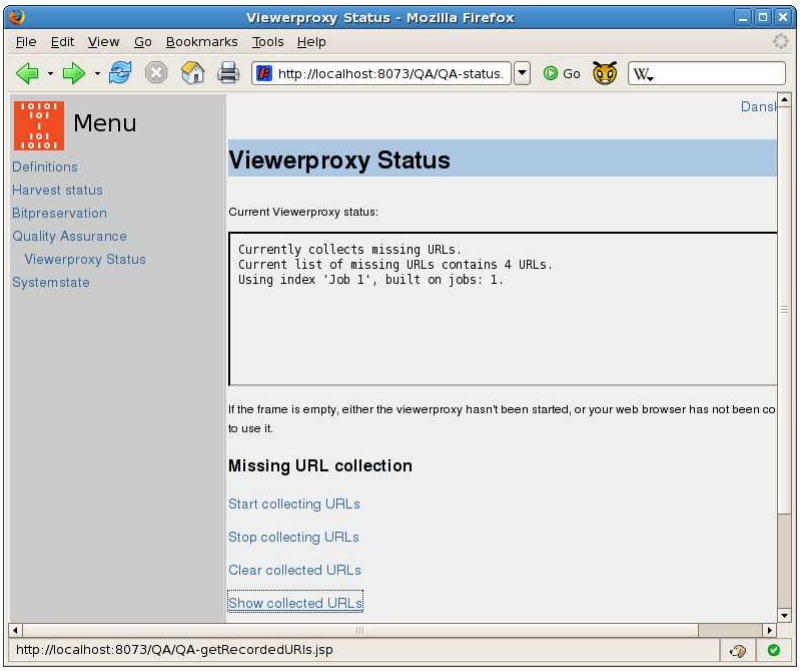


Now simply enter the URL that you started harvesting from (with www), e.g. `www.fnord.com`. It shows you the harvested material. If you go to a URL in another domain, you will get an error. Depending on the layout of the domain you harvested, there may also be missing pages or images from that domain.

The NetarchiveSuite allows automatic collection of unharvested URLs during browsing,

i.e. the NetarchiveSuite allows you to browse in the collected material while it automatically collects URLs for missing pages or images that you request. This makes it easy to identify missing harvested material, when you are doing Quality Assurance on the harvested material.

To try this, go back to the viewerproxy status page and click 'Start collecting URLs'. Now browse in the collected material until you find a page or image that did not get harvested. Go back to the viewerproxy status page and click 'Show collected URLs'.



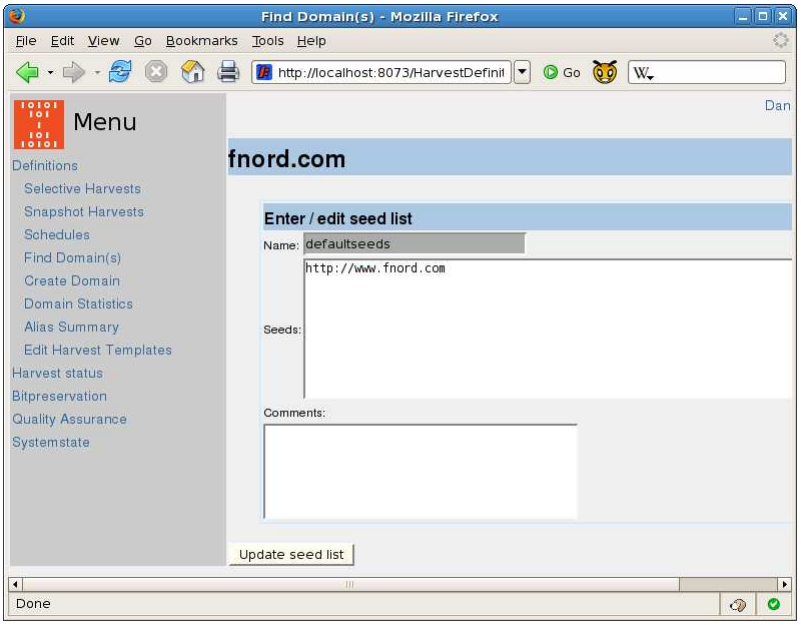
The list will contain several URLs, including the ones you just requested and found missing during collection of URIs.

Let us make sure these URLs are harvested next time we harvest the domain.

Copy the URLs for your harvested domain that were found missing into the clipboard. Go to the domain definition page by clicking 'Find Domain(s)' under 'Definitions' and search for your domain.

You will now get a page with information used when harvesting that domain. In this case, we wish to add the collected URLs to the list of seeds we start our web harvests from.

On the domain definition page, click 'Edit' next to the seed list.



Add the URLs from the clipboard to the seed list and press 'Update'.

These URLs will be used as seeds the next time the domain is harvested, i.e. the harvest will include these URLs in the harvest. To see this in effect, create another harvest of this domain following all the steps above. Wait for the domain to finish harvesting, then go to the 'job status' page for the new job. Limit the viewerproxy to the new job only and browse the material again. The URLs that were missing last time should now be found.

edit

Running a snapshot harvest

edit

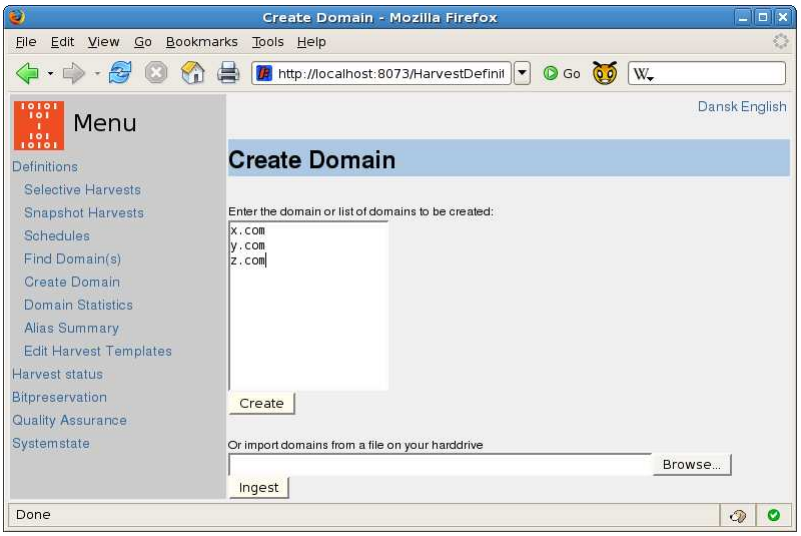
A snapshot harvest harvests all known domains up to a given byte limit, i.e. a limit of bytes that you harvest from each domain. This is used for national-wide harvests of **all** domains.

Each domain has one "default configuration" automatically generated when the domain is created. The default configuration is used to determine how to harvest the domain in a snapshot harvest. Typically, the default configuration is good enough for most purposes, but if you want to have a domain excluded from the snapshot harvest (e.g. if the domain is outside the group you're interested in) you may want to set the harvest limit on the default configuration for that domain to 0. The default configuration is also the one used in a selective harvest unless another configuration is chosen in the drop-down menu on the selective harvest page. The other way to control how a snapshot harvest is executed is by choosing a different harvest template. Descriptions of how harvest templates work are in the user manual.

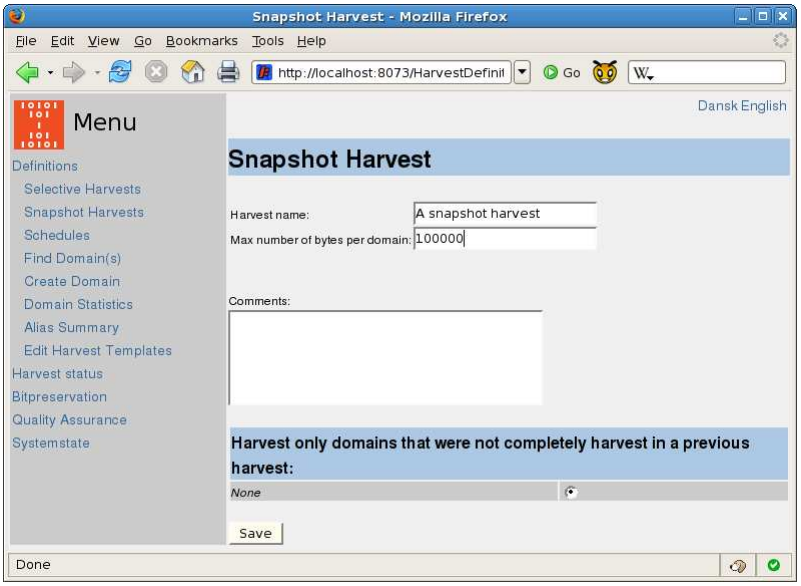
NetarchiveSuite has support for mass creation of domains, for instance by ingesting (loading) a list of domains given by a national TLD (top-level-domain) administrator.

To ingest, go to the "Create Domain" page under "Definitions" and specify the file

containing the list of domains. You can also type domains in the text window, but this is only usable for a smaller number of domains. The list should be a newline-separated list of domain names including the top level domain, but not including subdomains, protocol specifications or URL paths. Thus `netarkivet.dk` or `archive.org` are useable, while `http://foo.com`, `bar.dk/hest` or `news.bbc.co.uk` are not. Note that we assume only one level under the TLD at the moment. When the file is specified, press "Ingest" and wait while the domains are ingested. For a first test, you probably want to keep it to a fairly small number of sites, to make sure the test harvest doesn't take too long.



After ingest, you can click on 'Domain statistics' under 'Definitions' to see an overview of how many domains are registered under the TLDs. To create a snapshot definition, go to 'Snapshot harvests' and press 'Create new snapshot harvest'. The harvest definition presented will require you to enter a harvest name, and also allows you to add comments or changing the limit of how many bytes to collect per domain. Keep this to a fairly low number for a first test, to make sure the harvest doesn't run too long.



When you have entered the information, press 'Save' and then press 'Activate'.

You can monitor the harvest and browse the harvested material exactly as you did in the previous harvests.

Carrying on...

edit

This concludes the quickstart manual. While you can of course continue to play around with this simple setup, there are numerous options and possibilities that are not mentioned herein that are useful for scalability and for adapting the harvesting to your needs. Further information about installation and configuration can be found in the Installation Manual, and more details on how to use the web interface can be found in the User Manual. The Developer Manual has information on how to program new or altered functionality for NetarchiveSuite in Java.

Enjoy!

QuickStart Manual (last edited 2007-07-03 09:28:51 by KaareChristiansen)