FAQ

This is the official Nutch FAQ.

General

House Keeping

Questions that are not answered in the FAQ or in the documentation should be posted to the appropriate mailing list.

Please stick to technical issues on the discussion forum and mailing lists. Keep in mind that these are public, so do not include any confidential information in your questions!

You should also read the **Mailing Lists Developer Resource** (http://www.apache.org/dev/#mail) before participating in the discussion forum and mailing lists.

NOTE:
Please do NOT submit bugs, patches, or feature requests to the mailing lists. Refer instead to Commiter's_Rules and HowToContribute areas of the Nutch wiki.

Are there any mailing lists available?

There's a user, developer, commits and agents lists, all available at http://nutch.apache.org/mailing_lists.html.

How can I stop Nutch from crawling my site?

Please visit our "webmaster info page"

Will Nutch be a distributed, P2P-based search engine?

We don't think it is presently possible to build a peer-to-peer search engine that is competitive with existing search engines. It would just be too slow. Returning results in less than a second is important: it lets people rapidly reformulate their queries so that they can more often find what they're looking for. In short, a fast search engine is a better search engine. We don't think many people would want to use a search engine that takes ten or more seconds to return results.

That said, if someone wishes to start a sub-project of Nutch exploring distributed searching, we'd love to host it. We don't think these techniques are likely to solve the hard problems Nutch needs to solve, but we'd be happy to be proven wrong.

Will Nutch use a distributed crawler, like Grub?

Distributed crawling can save download bandwidth, but, in the long run, the savings is not significant. A successful search engine requires more bandwidth to upload query result pages than its crawler needs to download pages, so making the crawler use less bandwidth does not reduce overall bandwidth requirements. The dominant expense of operating a large search engine is not crawling, but searching.

Won't open source just make it easier for sites to manipulate rankings?

Search engines work hard to construct ranking algorithms that are immune to manipulation. Search engine optimizers still manage to reverse-engineer the ranking algorithms used by search engines, and improve the ranking of their pages. For example, many sites use link farms to manipulate search engines' link-based ranking algorithms, and search engines retaliate by improving their link-based algorithms to neutralize the effect of link farms.

With an open-source search engine, this will still happen, just out in the open. This is analagous to encryption and virus protection software. In the long term, making such algorithms open source makes them stronger, as more people can examine the source code to find flaws and suggest improvements. Thus we believe that an open source search engine has the potential to better resist manipulation of its rankings.

What Java version is required to run Nutch?

Nutch 1.0 requires Java 6 and up.

I have two XML files, nutch-default.xml and nutch-site.xml, why?

nutch-default.xml is the out of the box configuration for Nutch, and most configurations can (and should unless you know what your doing) stay as per. nutch-site.xml is where you make the changes that override the default settings.

Compiling Nutch

How do I compile Nutch?

Install ANT and call 'ant' on the command line from the directory containing the Nutch source code. Note : this won't work for the binary release for obvious reasons.

How do I compile Nutch in Eclipse?

Nutch uses ANT+IVY to compile the code and manage the dependencies (see above). There are instructions on how to get Nutch working with Eclipse on [http://wiki.apache.org/nutch/RunNutchInEclipse] but the easiest way of doing is to use ANT for compiling and rely on Eclipse just for visualising the code. You can also debug with Eclipse using the remote debugging and setting e.g. "export NUTCH_OPTS=-Xdebug -agentlib:jdwp=transport=dt_socket, server=y,address=8000" prior to calling the nutch script in /runtime/local/bin.

Injecting

What happens if I inject urls several times?

Urls which are already in the database, won't be injected.

Fetching

Can I parse during the fetching process?

In short yes, however this is disabled by default (justification follows shortly). To enable this simply configure the following in nutch-site.xml before initiating the fecth process.

```
<property>
  <name>fetcher.parse</name>
  <value>true</value>
  <description>If true, fetcher will parse content. Default is false, which means
  that a separate parsing step is required after fetching is finished.</description>
</property>
```

N.B. In a parsing fetcher, outlinks are processed in the reduce phase (at least when outlinks are followed). If a fetcher's reducer stalls you may run out of memory or disk space, usually after a very long reduce job. Behaviour typical to this is usually observed in this situation.

In summary, if it is possible, users are advised **not** to use a parsing fetcher as it is heavy on IO and often leads to the above outcome.

Is it possible to fetch only pages from some specific domains?

Please have a look on PrefixURLFilter. Adding some regular expressions to the regex-urlfilter.txt file might work, but adding a list with thousands of regular expressions would slow down your system excessively.

Alternatively, you can set db.ignore.external.links to "true", and inject seeds from the domains you wish to crawl (these seeds must link to all pages you wish to crawl, directly or indirectly). Doing this will let the crawl go through only these domains without leaving to start crawling external links. Unfortunately there is no way to record external links encountered for future processing, although a very small patch to the generator code can allow you to log these links to hadoop.log.

How can I recover an aborted fetch process?

Well, you can not. However, you have two choices to proceed:

- 1) Recover the pages already fetched and than restart the fetcher.
    - You'll need to create a file fetcher.done in the segment directory and then: updatedb, generate and fetch . Assuming your crawl data is at /crawl

```
% touch /index/segments/2005somesegment/fetcher.done

% bin/nutch updatedb /crawl/db/ /crawl/segments/2005somesegment/

% bin/nutch generate /crawl/db/ /crawl/segments/2005somesegment/

% bin/nutch fetch /crawl/segments/2005somesegment
```

- All the pages that were not crawled will be re-generated for fetch. If you fetched lots of pages, and don't want to have to re-fetch them again, this is the best way.
- 2) Discard the aborted output.
    - Delete all folders from the segment folder except the fetchlist folder and restart the fetcher.

Who changes the next fetch date?

- After injecting a new url the next fetch date is set to the current time.
- Generating a fetchlist enhances the date by 7 days.
- Updating the db sets the date to the current time + db.default.fetch.interval - 7 days.

I have a big fetchlist in my segments folder. How can I fetch only some sites at a time?

- You have to decide how many pages you want to crawl before generating segments and use the options of bin/nutch generate.
- Use -topN to limit the amount of pages all together.
- Use -numFetchers to generate multiple small segments.
- Now you could either generate new segments. Maybe you whould use -adddays to allow bin/nutch generate to put all the urls in the new fetchlist again. Add more then 7 days if you did not make a updatedb.
- Or send the process a unix STOP signal. You should be able to index the part of the segment for crawling which is allready fetched. Then later send a CONT signal to the process. Do not turn off your computer between! 🙂

How many concurrent threads should I use?

This is dependent on your particular set-up; unless one understands system/network environment variables it is impossible to accurately measure thread performance. The Nutch de-facto is an excellent start point.

How can I force fetcher to use custom nutch-config?

- Create a new sub-directory under $NUTCH_HOME/conf, like conf/myconfig
- Copy these files from $NUTCH_HOME/conf to the new directory: common-terms.utf8, mime-types.*, nutch-conf.xsl, nutch-default.xml, regex-normalize.xml, regex-urlfilter.txt
- Modify the nutch-default.xml to suite your needs
- Set NUTCH_CONF_DIR environment variable in $NUTCH_HOME/bin/nutch to point into the directory you created
- run $NUTCH_HOME/bin/nutch so that it gets the NUTCH_CONF_DIR environment variable. You should check the command outputs for lines where the configs are loaded, that they are really loaded from your custom dir.
- Happy using.

bin/nutch generate generates empty fetchlist, what can I do?

The reason for that is that when a page is fetched, it is timestamped in the webdb. So basiclly if its time is not up it will not be included in a fetchlist. So for example if you generated a fetchlist and you deleted the segment dir created. calling generate again will generate an empty fetchlist. So, two choices:

1) Change your system date to be 30 days from today (if you haven't changed the default settings) and re-run bin/nutch generate
2) Call bin/nutch generate with the -adddays 30 (if you haven't changed the default settings) to make generate think the time has come... After generate you can call bin/nutch fetch.

How can I fetch pages that require Authentication?

See the HttpAuthenticationSchemes wiki page.

Speed of Fetching seems to decrease between crawl iterations... what's wrong?

A possible reason is that by default the 'partition.url.mode' is set to 'byHost', which is a reasonable setting, because in the url-subsets for the fetcher threads in different map steps, you want to have disjoint subsets to avoid that urls are loaded twice from different machines.

Secondly the default setting for 'generate.max.count' could also be set to -1. This means the more urls you collect, especially from the same host, the more urls of the same host will be in the same fetcher map job!

Because there is also a policy setting (please do this at home!!) to wait for a delay of 30 secs. between calls to the same server, all maps which contains urls to the same server are slowing down. Therefore the resulting reduce step will only be done when all fetcher maps are done, which is a bottleneck in the overall processing step.

The following settings may solve your problem:

Map tasks should be splitted according to the host:

```
<property>
  <name>partition.url.mode</name>
  <value>byHost</value>
  <description>Determines how to partition URLs. Default value is
'byHost',  also takes 'byDomain' or 'byIP'.
  </description>
</property>
```

Don't insert in a single fetch list more than 10000 entries!

```
<property>
  <name>generate.max.count</name>
  <value>10000</value>
  <description>The maximum number of urls in a single
  fetchlist.  -1 if unlimited. The urls are counted according
  to the value of the parameter generator.count.mode.
  </description>
</property>
```

Wait time between two fetches to the same server.

```
<property>
 <name>fetcher.max.crawl.delay</name>
 <value>10</value>
 <description>
 If the Crawl-Delay in robots.txt is set to greater than this value (in
 seconds) then the fetcher will skip this page, generating an error report.
 If set to -1 the fetcher will never skip such pages and will wait the
 amount of time retrieved from robots.txt Crawl-Delay, however long that
 might be.
 </description>
</property>
```

What do the numbers in the fetcher log indicate ?

While fetching is in progress, the fetcher job will log such statement to indicate the progress of the job:

```
0/20 spinwaiting/active, 53852 pages, 7612 errors, 4.1 12 pages/s, 2632 7346 kb/s, 989 URLs in 5 queue
```

Here is the explanation of each of all the fields:

- Fetcher threads try to get a fetch item (url) from a queue of all the fetch items (this queue is actually a queue of queues. For details see [0]). If a thread doesn't get a fetch-item, it spinwaits for 500ms before polling the queue again. The 'spinWaiting' count tells us how many threads are in their "spinwaiting" state at a given instance.

- The 'active' count tells us how many threads are currently performing the activities related to the fetch of a fetch-item. This involves sending requests to the server, getting the bytes from the server, parsing, storing etc.
- 'pages' is a count for total pages fetched till a given point.
- 'errors' is a count for total errors seen.
- Next comes pages/s. First number comes from this:

```
((((float)pages)*10)/elapsed)/10.0
```

  second one comes from this:

```
(actualPages*10)/10.0
```

  . "actualPages" holds the count of pages processed in the last 5 secs (when the calculation is done). First number can be seen as the overall speed for that execution. The second number can be regarded as the instanteous speed as it just uses the #pages in last 5 secs when this calculation is done.
- Next comes the kb/s values which are computed from: `(((float)bytes)*8)/1024)/elapsed` and

```
((float)actualBytes)*8)/1024
```

  . This is similar to that of pages/sec.
- 'URLs' indicates how many urls are pending and 'queues' indicate the number of queues present. Queues are formed on the basis on hostname or ip depending on the configuration set.

Updating

Isn't there redudant/wasteful duplication between nutch crawldb and solr index?

Nutch maintains a crawldb (and linkdb, for that matter) of the urls it crawled, the fetch status, and the date. This data is maintained beyond fetch so that pages may be re-crawled, after the a re-crawling period. At the same time Solr maintains an inverted index of all the fetched pages. It'd seem more efficient if Nutch relied on the index instead of maintaining its own crawldb, to !store the same url twice? The problem we face here is what Nutch would do if we wished to change the Solr core which to index to?

Whats described above could be done with Nutch 2.0 by adding a SOLR backend to GORA. SOLR would be used to store the webtable and provided that you setup the schema accordingly you could index the appropriate fields for searching. Further to this, because Nutch is a crawler intending to write to more than one search engine. Besides, the crawldb is gone, as a flat file, in trunk (2.0). Also, Solr is really slow when it comes to updating millions of records, the crawldb isn't when split over multiple machines.

For more information see here

Indexing

Is it possible to change the list of common words without crawling everything again?

Yes. The list of common words is used only when indexing and searching, and not during other steps. So, if you change the list of common words, there is no need to re-fetch the content, you just need to re-create segment indexes to reflect the changes.

How do I index my local file system?

The tricky thing about Nutch is that out of the box it has most plugins disabled and is tuned for a crawl of a "remote" web server - you **have** to change config files to get it to crawl your local disk.

- 1) regex-urlfilter.txt needs a change to allow file: URLs while not following http: ones, otherwise it either won't index anything, or it'll jump off your disk onto web sites.
  - Change this line: -(file|ftp|mailto|https): to this: -(http|ftp|mailto|https):
    2) regex-urlfilter.txt may have rules at the bottom to reject some URLs. If it has this fragment it's probably ok:
  - # accept anything else +.*
    3) By default the protocol-file plugin is disabled. nutch-site.xml needs to be modified to allow this plugin. Add an entry like this:

```
<property>
  <name>plugin.includes</name>
  <value>protocol-file|...copy original values from nutch-default here...</value>
</property>
```

where you should copy and paste all values from nutch-default.xml in the plugin.includes setting provided there. This will ensure that all plug-ins normally enabled will be enabled, plus the protocol-file plugin.

Now you can invoke the crawler and index all or part of your disk.

Nutch crawling parent directories for file protocol

By default, Nutch will step into parent directories. You can avoid this by setting the following property to false:

```
<property>
  <name>file.crawl.parent</name>
  <value>false</value>
  <description>The crawler is not restricted to the directories that you specified in the
    Urls file but it is jumping into the parent directories as well. For your own crawlings you can
    change this behavior (set to false) the way that only directories beneath the directories that you specify
get
    crawled.</description>
</property>
```

Alternatively, you could add a regex URL filter rule, e.g.

```
+^file:/c:/top/directory/
-.
```

- and don't forget to make sure that the plugin urlfilter-regex is enabled in plugin.includes.

A note on slashes after file:

When converting `file:` URLs from the Java URL class back only one slash remains:

```
String url = "file:///path/index.html";
java.net.URL u = new java.net.URL(url);
url = u.toString();  // url is now file:/path/index.html
```

Because such conversions are quite frequent, you better writer URLs (and also URL filter rules, etc.) with a single slash (`file:/path/index.html`). Nutch's URL normalizers in the default configuration also normalize file: URLs to have only one slash.

How do I index remote file shares?

At the current time, Nutch does not have built in support for accessing files over SMB (Windows) shares. This means the only available method is to mount the shares yourself, then index the contents as though they were local directories (see above).

Note that the share mounting method suffers from the following drawbacks:

1) The links generated by Nutch will not work except for queries from local host (end users typically won't have the exact same shares mounted in the exact same way)
2) You are limited to the number of mounted shares your operating system supports. In *nix environments, this is effectively unlimited, but in Windows you may mount 26 (one share or drive per letter in the English alphabet)
3) Documents with links to shares are unlikely to work since they won't link to the share on your machine, but rather to the SMB version.

Segment Handling

Do I have to delete old segments after some time?

If you're fetching regularly, segments older than the db.default.fetch.interval can be deleted, as their pages should have been refetched. This is 30 days by default.

MapReduce

What is MapReduce?

Please see the MapReduce page of the Nutch wiki.

How to start working with MapReduce?

- edit $HADOOP_HOME/conf/mapred-site.xml <property>

    - <name>fs.default.name</name> <value>localhost:9000</value> <description>The name of the default file system. Either the literal string "local" or a host:port for NDFS.</description>
      </property>
      <property>
    - <name>mapred.job.tracker</name> <value>localhost:9001</value> <description>The host and port that the MapReduce job tracker runs at. If "local", then jobs are run in-process as a single map and reduce task.</description>
      </property> edit conf/mapred-default.xml
      <property>
    - <name>mapred.map.tasks</name> <value>4</value> <description>define mapred.map.tasks to be multiple of number of slave hosts < /description>
      </property>
      <property>
    - <name>mapred.reduce.tasks</name> <value>2</value> <description>define mapred.reduce tasks to be number of slave hosts< /description>
      </property> create a file with slave host names

```
% echo localhost >> ~/.slaves
% echo somemachin >> ~/.slaves
```

- start all ndfs & mapred daemons

```
% bin/start-all.sh
```

- create a directory with seed list file

```
% mkdir seeds
% echo http://www.cnn.com/ > seeds/urls
```

- copy the seed directory to ndfs

```
% bin/nutch ndfs -put seeds seeds
```

- crawl a bit

```
% bin/nutch crawl seeds -depth 3
```

- monitor things from adminstrative interface open browser and enter your masterHost : 7845

NDFS

What is it?

NutchDistributedFileSystem

How to send commands to NDFS?

- list files in the root of NDFS

```
[root@xxxxxx mapred]# bin/nutch ndfs -ls /
050927 160948 parsing file:/mapred/conf/nutch-default.xml
050927 160948 parsing file:/mapred/conf/nutch-site.xml
050927 160948 No FS indicated, using default:localhost:8009
050927 160948 Client connection to 127.0.0.1:8009: starting
Found 3 items
/user/root/crawl-20050927142856 <dir>
/user/root/crawl-20050927144626 <dir>
/user/root/seeds        <dir>
```

- remove a directory from NDFS

```
[root@xxxxxx mapred]# bin/nutch ndfs -rm /user/root/crawl-20050927144626
050927 161025 parsing file:/mapred/conf/nutch-default.xml
050927 161025 parsing file:/mapred/conf/nutch-site.xml
050927 161025 No FS indicated, using default:localhost:8009
```

```
050927 161025 Client connection to 127.0.0.1:8009: starting
Deleted /user/root/crawl-20050927144626
```

Scoring

How can I influence Nutch scoring?

Scoring is implemented as a filter plugin, i.e. an implementation of the ScoringFilter class. By default, the OPIC Scoring Filter is used. There are also numerous scoring filter properties which can be specified within nutch-site.xml.

Searching

How can I find out/display the size and mime type of the hits that a search returns?

In order to be able to find this information you have to modify the standard `plugin.includes` property of the nutch configuration file and add the `index-more` filter.

```
<property>
  <name>plugin.includes</name>
  <value>...|index-more|...|...</value>
  ...
</property>
```

After that, don't forget to crawl again and you should be able to retrieve the mime-type and content-length through the class HitDetails (via the fields "primaryType", "subType" and "contentLength") as you normally do for the title and URL of the hits.

Crawling

Nutch doesn't crawl relative URLs? Some pages are not indexed but my regex file and everything else is okay - what is going on?

The crawl tool has a default limitation of 100 outlinks of one page that are being fetched. To overcome this limitation change the **db.max.outlinks.per.page** property to a higher value or simply -1 (unlimited).

file: conf/nutch-default.xml

```
<property>
  <name>db.max.outlinks.per.page</name>
  <value>-1</value>
  <description>The maximum number of outlinks that we'll process for a page.
  If this value is nonnegative (>=0), at most db.max.outlinks.per.page outlinks
  will be processed for a page; otherwise, all outlinks will be processed.
  </description>
</property>
```

see also: http://www.mail-archive.com/nutch-user@lucene.apache.org/msg08665.html

Discussion

Grub has some interesting ideas about building a search engine using distributed computing. *And how is that relevant to nutch?*

CategoryHomepage