

NutchConfigurationFiles

The primary (core) Nutch configuration files are

- [conf/nutch-default.xml](#): This file contains generic default settings for Nutch specific configuration properties.
- [conf/nutch-site.xml](#): This file contains site specific settings for Nutch specific configuration properties.

Hadoop configuration

- [core/hadoop-default.xml](#): This file contains generic default settings for Hadoop daemons and Map/Reduce jobs.
- [core-site.xml](#): This file contains site specific settings for all Hadoop daemons and Map/Reduce jobs.
- [mapred-site.xml](#): This file contains site specific settings for the Hadoop Map/Reduce daemons and jobs.

For more information on the Hadoop configuration files please see [GettingStartedWithHadoop#Configurationfiles](#)

Dennis Kubes explains:

Configuration has two levels, default and final. It is supplied by the [org.apache.hadoop.conf.Configuration class](#) and extended in Nutch by the [org.apache.nutch.util.NutchConfiguration class](#).

Although it is configurable, by default `hadoop-default.xml` and `nutch-default.xml` are default resources and `hadoop-site.xml` and `nutch-site.xml` are final resources. Resources (i.e. resource files) can be added by filename to either the default or final resource set and in fact this is how Nutch extends the `Configuration` class, by adding `nutch-default.xml` and `nutch-site.xml`.

Final resource values overwrite default resource values and final resource values added later will overwrite final resource values added earlier. When I say values I am talking about the individual properties not the resource files. Resource files are found by name in the classpath with the `HADOOP_CONF_DIR` or `NUTCH_CONF_DIR` being configured in the `nutch` and `hadoop` scripts as the first setting in the classpath. You can change the `conf dir` to pull configuration files from different directories and many tools in `nutch` and `hadoop` now provide a `-conf` options on the command line to set the `conf` directory.

So for example if you define the property in `hadoop-default.xml` or `nutch-default.xml` and it is not defined in either `hadoop-site.xml` or `nutch-site.xml` then the property will stand. If you define the property in either `nutch-site.xml` or `hadoop-site.xml` then it will override `nutch-default.xml` and `hadoop-default.xml` settings. And if you define it in both `hadoop-site.xml` and `nutch-site.xml` then the `nutch-site.xml` will override the `hadoop-site.xml` settings because `nutch-site.xml` is added after `hadoop-site.xml`. And remember only individual properties are overridden not the entire file.

Practically you should define properties having to do with Hadoop (i.e. the [HDFS](#), [MapReduce](#), etc) in the `hadoop-site.xml` and properties having to do with Nutch (i.e. `fetcher`, `url-normalizers`, etc) in the `nutch-site.xml`.

Andrzej Bialecki further details:

There are other two important config files:

- `mapred-default.xml` - (Hadoop specific) this is loaded as default resource when a new map-reduce [JobConf](#) is created - which means that it is loaded as the last default resource when you prepare the job configuration. Usually you should keep its content to a bare minimum. This is the best place to specify the default number of map and reduce tasks per job. If you feel adventurous you could also put some other stuff there, e.g. set the default compression with `mapred.compress.map.output` and so on.
- `job.xml` - (Hadoop specific **Deprecated**) this file is created dynamically, and represents a serialized [JobConf](#). When map-reduce tasks are started they read this file as their last default resource (note - this is NOT a final resource!). So, if you accidentally distributed `mapred-default.xml` to all cluster nodes, but in your job you specified a different number of map or reduce tasks, your settings will take precedence. The same with other settings, such as e.g. the compression setting.

HOWEVER ... a common error is to put too many properties such as default number of map and reduce tasks in `hadoop-site.xml`. As Dennis explained, this is a final resource - which means that the values you specify there will ALWAYS override your job settings. This is bad, so don't do it 😞 - put them in `mapred-default.xml`.

In other words: use `hadoop-site.xml` only for things that are always the same for the whole cluster, such as the FS name, jobtracker name:port, temp directories, etc. because values you put there will always override your job settings. And do not put there things that are job-dependent.