bin/nutch mergesegs

Mergesegs is an alias for org.apache.nutch.segment.SegmentMerger

This tool takes several segments and merges their data together. Only the latest versions of data is retained. Optionally, you can apply current URLFilters to remove prohibited URL-s. Also, it's possible to slice the resulting segment into chunks of fixed size.

Important Notes

Which parts are merged?

It doesn't make sense to merge data from segments, which are at different stages of processing (e.g. one unfetched segment, one fetched but not parsed, and one fetched and parsed). Therefore, prior to merging, the tool will determine the lowest common set of input data, and only this data will be merged. This may have some unintended consequences: e.g. if majority of input segments are fetched and parsed, but one of them is unfetched, the tool will fall back to just merging fetchlists, and it will skip all other data from all segments.

Merging fetchlists

Merging segments, which contain just fetchlists (i.e. prior to fetching) is not recommended, because this tool (unlike the {@link org.apache.nutch.crawl.Generator} doesn't ensure that fetchlist parts for each map task are disjoint.

Duplicate content

Merging segments removes older content whenever possible (see below). However, this is NOT the same as de-duplication, which in addition removes identical content found at different URL-s. In other words, running a command to delete duplicates is still necessary.

For some types of data (especially ParseText) it's not possible to determine which version is really older. Therefore the tool always uses segment names as timestamps, for all types of input data. Segment names are compared in forward lexicographic order (0-9a-zA-Z), and data from segments with "higher" names will prevail. It follows then that it is extremely important that segments be named in an increasing lexicographic order as their creation time increases.

Merging and indexes

Merged segment gets a different name. Since Indexer embeds segment names in indexes, any indexes originally created for the input segments will NOT work with the merged segment. Newly created merged segment(s) need to be indexed afresh. This tool doesn't use existing indexes in any way, so if you plan to merge segments you don't have to index them prior to merging.

There are no prerequisites for correct operation of merging segments except for a set of already fetched segments (they don't have to contain parsed content, only fetcher output is required).

Usage:

```
bin/nutch mergesegs <output_dir> (-dir segments | seg1 seg2 ...) [-filter] [-slice NNNN]
```

<output_dir>: This is the path name of the parent directory for output segment slice(s)

-dir segments: The path to the parent directory containing several segments.

seg1 seg2 ...: This parameter should be a comprehensive list of segment directories to be merged.

[-filter]: This enables us to filter out URLs based upon current URLFilters we wish to implement. This can be used to improve the quality of the resulting segments after a merge is executed.

[-slice NNNN]: This arguement should be passed if we wish to create many output segments, each containing NNNN URLs. e.g. If we wanted to merge 10 segments each containing 20 URLS into 5 segments each containg 40 URLs.

CommandLineOptions