bin/nutch generate

Generate is an alias for org.apache.nutch.crawl.Generator

This class generates a subset of a crawl db to fetch. This version allows us to generate fetchlists for several segments in one go. Unlike in the initial version (FetchListTool), the IP resolution is done ONLY on the entries which have been selected for fetching. The URLs are partitioned by IP, domain or host within a segment. We can chose separately how to count the URLS i.e. by domain or host to limit the entries.

Both versions return 0 if one or more segment have been generated, -1 on error and 1 if there aren't any URLs to put in a segment.

Nutch 1.x

```
Usage: bin/nutch generate <crawldb> <segments_dir> [-force] [-topN N] [-numFetchers numFetchers] [-adddays
numDays] [-noFilter] [-noNorm][-maxNumSegments num]
```

<crawldb>: Path to the location of our crawldb directory.

<segments_dir>: Path to the location of our segments directory where the Fetcher Segments are created.

[-force]: This argument will force an update even if there appears to be a lock. blocked URL : CAUTION: advised blocked URL

[-topN N]: Where N is the number of top URLs to be selected. Normally, the "generate" command prepares a fetchlist out of all unfetched pages, or the ones where fetch interval already expired. But if you use -topN, then instead of all unfetched urls you only get N urls with the highest score - potentially the most interesting ones, which should be prioritized in fetching.

[-numFetchers numFetchers]: The number of fetch partitions. Default: Configuration key -> mapred.map.tasks -> 1 (in local mode), possibly multiple in deploy/distributed mode.

[-adddays numDays]: Adds <days> to the current time to facilitate crawling urls already fetched sooner then db.default.fetch.interval. Default: 0

[-noFilter]:Whether to filter URLs or not is read from the crawl.generate.filter property in nutch-site.xml/nutch-default.xml configuration files. If the property is not found, the URLs are filtered. Same for the normalisation

[-noNorm]: The exact same applies for normalisation parameter as does for the filtering option above.

[-maxNumSegments num]: The (maximum) number of segments to be generated. Default: 1 -- Note: if multiple segments are generated, the limit -topN applies to the total number of URLs for all segments taken together, while generate.max.count is applied to every generated segment individually.

Configuration Files

hadoop-default.xml
hadoop-site.xml
nutch-default.xml
nutch-site.xml

Configuration Values

The following properties directly affect how the Generator generates fetch segments:

- generate.max.count: 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.
- generate.count.mode: Determines how the URLs are counted for generator.max.count. Default value is 'host' but can be 'domain'. Note that we do not count per IP in the new version of the Generator.
- partition.url.mode: Determines how URLs are distributed over fetch partitions: "byHost" (default), "byDomain", or "byIP". Cf. the corresponding property "fetcher.queue.mode" in Fetcher used to guarantee delays between successive fetch requests to the same host/domain/IP.

Indirectly, the behavior of Generator is influenced by:

- mapreduce.job.reduces: In a distributed environment (Hadoop) with multiple reducers the max. total number of URLs (-topN) is applied per reduce task as (topN/numReduceTasks). If URLs are not evenly spread over hosts (domains or IPs, see partition.url.mode) or belong to a single host/domain/IP, some partitions may be smaller than expected or even empty. The total number of generated URLs is then lower than topN.

Examples

```
bin/nutch org.apache.nutch.crawl.Generator /my/crawldb /my/segments
```

This example will generate a fetch list that contains all URLs ready to be fetched from the crawldb. The crawldb is located at my/crawldb and the generator will output the fetch list to /my/segments/yyyyMMddHHmmss.

```
bin/nutch org.apache.nutch.crawl.Generator /my/crawldb /my/segments -topN 100 -adddays 20
```

In this example the Generator will add 20 days to the current date/time when determining the top 100 scoring pages to fetch.

Nutch 2.x

```
Usage: GeneratorJob [-topN N] [-crawlId id] [-noFilter] [-noNorm] [-adddays numDays]
    -topN <N>      - number of top URLs to be selected, default is Long.MAX_VALUE
    -crawlId <id>  - the id to prefix the schemas to operate on,
                     (default: storage.crawl.id)");
    -noFilter      - do not activate the filter plugin to filter the url, default is true
    -noNorm        - do not activate the normalizer plugin to normalize the url, default is true
    -adddays       - Adds numDays to the current time to facilitate crawling urls already
                     fetched sooner then db.default.fetch.interval. Default value is 0.
```

CommandLineOptions