NAME

swish++.conf − SWISH++ configuration file format

DESCRIPTION

The configuration file format used by SWISH++ consists of three types of lines: comments, blank lines, and variable definitions.

Comments

Comments start with the # character and continue up to and including the end of the line. While leading whitespace is permitted, **comments are treated as such only if they are on lines by themselves**.

Blank lines

Blank lines, or lines consisting entirely of whitespace, are ignored.

Variable definitions

Variable definition lines are of the form:

variable_name argument(s)

where *variable_name* is a member of one of the types described in the remaining sections, and *argument(s)* are specific to every variable name.

Boolean variables

Variables of this type take one argument that must be one of: `f`, `false`, `n`, `no`, `off`, `on`, `t`, `true`, `y`, or `yes`. Case is irrelevant. Variables of this type are: **RecurseSubdirs** and **StemWords**.

Integer variables

Variables of this type take one numeric argument. A special string of ‘‘infinity’’ is taken to mean ‘‘the largest possible integer value.’’ Case is irrelevant. Variables of this type are: **ResultsMax**, **TitleLines**, **Verbosity**, **WordFilesMax**, and **WordPercentMax**.

String variables

Variables of this type take one argument that is the remainder of the line minus leading and trailing whitespace. Variables of this type are: **IndexFile** and **StopWordFile**.

Set variables

Variables of this type take one or more arguments separated by whitespace. Variables of this type are: **ExcludeClass**, **IncludeExtension**, **ExcludeExtension**, **IncludeMeta**, and **ExcludeMeta**.

Other variables

Variables of this type are: **FilterExtension** (see FILTERS below).

FILTERS

Via the configuration file variable, files having particular extensions can be filtered prior to indexing or extraction. A **FilterExtension** configuration file line is of the form:

`FilterExtension` *extension command*

where *extension* is the filename extension (without the dot) and *command* is the command-line to execute the filter.

Within a command, there are a few % or @ substitutions that are substituted at run-time:

E	Filename with last extension deleted.
e	Extension of filename.
f	Entire filename.

The @ substitution is used to indicate which filename is the target or product of the filter. There must be exactly one @ substitution. This file is subsequently deleted after indexing or extraction. A file can be filtered more than once prior to indexing or extraction, i.e., filters can be ‘‘chained’’ together.

Note, however, that just because a filename has an extension for which a filter has been specified does not mean that a file will be filtered and subsequently indexed or extracted. When **index** or **extract** encounteres a file having an extension for which a filter has been specified, it performs the filename substitution(s) on it first to determine what the target filename would be. If the extension of *that* filename should be indexed or extracted (because it is among the set of extensions specified with either the **−e** option or the **IncludeExtension** variable or is not among the set specified with either the **−E** option or the **ExcludeExtension** variable), then the filter(s) are executed to create it. (See the EXAMPLES.)

EXAMPLES

Filters

To uncompress `gzip`'d and `compress`'d files prior to indexing or extraction, the **FilterExtension** variable lines in a configuration file would be:

```
FilterExtension gz   gunzip -c %f > @E
FilterExtension Z    uncompress -c %f > @E
```

Given that, a filename such as `foo.txt.gz` would become `foo.txt`. If files having `txt` extensions should be indexed, then it will be. Note that the command on the **FilterExtension** line must *not* simply be:

```
gunzip @f
```

because `gunzip` will *replace* the compressed file with the uncompressed one.

Here's an example to convert PDF to plain text for indexing using the **xpdf**(1) package's `pdftotext` command:

```
FilterExtension pdf  pdftotext %f @E.txt
```

Not that if used in conjunction with the uncompression filters above, then compressed PDF files will also be indexed, i.e., filenames ending with wither a `.pdf.gz` or `.pdf.Z` double extension.

SEE ALSO

compress(1), **extract**(1), **gunzip**(1), **gzip**(1), **index**(1), **pdftotext**(1), **search**(1), **uncompress**(1)

AUTHOR

Paul J. Lucas <*pjl@best.com*>