NAME

swish++.index − SWISH++ index file format

SYNOPSIS

```
long            num_words;
off_t           word_offset[ num_words ];
long            num_stop_words;
off_t           stop_word_offset[ num_stop_words ];
long            num_directories;
off_t           directory_offset[ num_directories ];
long            num_files;
off_t           file_offset[ num_files ];
long            num_meta_names;
off_t           meta_name_offset[ num_meta_names ];
```

word index
stop-word index
directory index
file index
meta-name index

DESCRIPTION

The index file format used by SWISH++ is as shown above. Every `word_offset` is an offset into the *word index* pointing at the first character of a word entry; similarly, every `stop_word_offset` is an offset into the *stop-word index* pointing at the first character of a stop-word entry; similarly, every `directory_offset` is an offset into the *directory index* pointing at the first character of a directory entry; similarly, every `file_offset` is an offset into the *file index* pointing at the first byte of a file entry; finally, every `meta_name_offset` is an offset into the *mete-name index* pointing at the first character of a meta-name entry.

The index file is written as it is so that it can be mapped into memory via the **mmap**(2) Unix system call enabling ''instantaneous'' access.

BCD Integers

All integers in an index file are stored in BCD (binary coded decimal) format for compactness. A BCD integer that has an odd number of digits is terminated by a low-order nybble with the value `\xA`; an integer that has an even number of digits is terminated by a a byte with the value `\xAA`. (These values were chosen because they are invalid BCD. All other values `\xA0` through `\xFE` are reserved for future use.)

For example, the integers 1−9 are stored as single bytes of `\x1A`−`\x9A`, respectively; the integer 10 is stored as the two bytes of `\x10AA`; the integer 1967 is stored as the three bytes of `\x1967AA`.

Since most integer values in an index file are less than 10000, using BCD results in approximately a 25% storage reduction.

Word Entries

Every word entry in the *word index* is of the form:

word`0`**{***data***}...**`FF`

that is: a null-terminated word followed by one or more *data* entries followed by an `FF` byte where a *data* entry is:

*I***[**`EE`**{***M***}...**`EE`**]***OR*

that is: a file-index (*I*) followed by zero or more meta-IDs (*M*) surrounded by `\xEE` bytes followed by the number of occurrences in the file (*O*) followed by a rank (*R*) followed by an `\xFF` byte. The *file-index* is an index into the `file_offset` table; the *meta-IDs*, if present, are unique integers identifying which meta name(s) a word is associated with in the meta-name index.

Stop-Word Entries

Every stop-word entry in the *stop-word index* is of the form:

stop-word`0`

that is: every word is null-terminated.

Directory Entries

Every directory entry in the *directory index* is of the form:

directory-path`0`

that is: a null-terminated full pathname of a directory (not including the trailing slash). The pathnames are relative to where the indexing was performed (unless absolute paths were used).

File Entries

Every file entry in the *file index* is of the form:

{*D***}***file-name***0{***S***}{***W***}***file-title***0**

that is: the file's directory index (*D*) followed by a null-terminated file name followed by the file's size in bytes (*S*) followed by the number of words in the file (*W*) followed by the file's null-terminated title.

For an HTML or XHTML file, the title is what is between `<TITLE>` ... `</TITLE>` pairs. For a mail or news file, the title is the value of the `Subject` header. For a Unix manual page file, the title is the contents of the first line within the `NAME` section. If a file is not one of those types of files, or is but does not have a title, the title is simply the file (not path) name.

Meta-Name Entries

Every meta-name entry in the *meta-name index* is of the form:

meta-name`0`**{***I***}**

that is: a null-terminated meta-name followed by the ID (*I*).

CAVEATS

Generated index files are machine-dependent (size of data types and byte-order).

SEE ALSO

index(1), **search**(1)

AUTHOR

Paul J. Lucas <*pauljlucas@mac.com*>