名称

**freqfile** – WAM

書式

**freqfile**

解説

**freqfile** は WAM の外部表現形式であり, 機械および人間に可読な形式になっている. このファイルを mkcw(1) および mkxr(1) を用いてバイナリ表現に変換することにより, libwam(3) を用いて WAM にアクセスすることができるようになる. **freqfile** では各種文字コードが使用可能であり, **freqfile** の段階でユーザアプリケーションが必要とする文字コードにしておくことで後の処理を簡単にすることができる. なお, libwam(3) は多バイト文字の処理は行わないため, NUL 以外で, 特定のバイトが 0 になる文字がある文字コードは使用できない (unicode はその一例?). なお, **freqfile** では, 改行文字として LF (012)を用いる.

**freqfile** は, WAM を行指向のテキスト形式で表現したものであり, WAM の行についての記述を, WAM の行数回, 繰り返して連続して記述したものである. 一般に, WAM の各行についての記述は, **freqfile** では複数の行になる. WAMのある行についての記述はその WAM の行の name を定義した行ではじまり, それに続いてその WAM の行の要素について記述した行が 0行以上現れるという構造になる.

WAMの行の name を定義した行は, 先頭の文字が @ か # である. @ や # の直後の文字から行末まで ( LF は含まない. また, NUL を含めてもならない. )がその WAM の行の name になる.

WAMの各要素について記述した行は, WAMの行の name を定義した行の先頭の文字に応じて2種類の形式がある. いずれの場合もWAMの1要素が1行で記述される.

name を定義した行の先頭の文字が @ の場合, 要素を記述した行は 1個以上の数字(10進)の列, space, 1文字上の任意の文字からなる文字列という形式になる. 最初の space の直前までの数(10進)で表される値がその要素の頻度(TF)になる. そして, それに続く space が丁度1つだけ捨てられ, その直後の文字から行の終わりまでの文字列がその要素が属しているWAMの列の name になる. WAM のある行についての記述中に同じ name の要素が2度以上現れた場合, それらは同一の要素を示すものとして扱われ, 頻度は合計される.

name を定義した行の最初の文字が # の場合, 要素を記述した各行は space, 1文字上の任意の文字からなる文字列という形式になる. そして, 行頭の space が丁度1つだけ捨てられ, その次の文字から改行の直前までの文字列が, その要素が属しているWAMの列の name になる. また, この要素の頻度(TF)は 1 である. @ で始まる場合と同様, 同一の name を持つ要素がWAMの同一行の記述中に現れた場合, その頻度は合計される.

形式 @ と 形式 # とは1つの **freqfile** に混ざって現れて良い. 例:

```
@key1
3 word1
5 word2
1 word3
#key2
 word1
 word3
 word5
 word3
 word4
#key3
 word1
 word9
```

WAM の別名ファイルも **freqfile** と同様の形式のファイルであり, 本頁で説明する. 別名ファイルは, 本名の name を記述した行と, その本名に対する別名の name を記述した行(一般に複数)が交互に現れる, 行指向のファイルである. 本名を記述した行は $ で始まり, その直後の文字から行末までが別名を定義しようとしている本名の name である. その行に続く, 行頭が space である行が別名を記述した行であり, それらの行の先頭の space を取り除いたものが, 先行する(最も近い)本名に対する別名の name となる. 例:

```
$word1
 alias1
 alias2
 ...
$word2
 ...
```

い. 別名の name には任意の name 用いることができる(本名は別名になれる)が, 同一の別名を異なる本名に対応させることはできない(別名から本名へは写像である).

WAM の一部では無いが, 検索システムなどを構築するための補助モジュールとして GETA では getaux(3) を提供している. この AUX ファイルの外部表現も **freqfile** に類似した形式であり, 本頁で説明しておく. AUX ファイルは WAMの各行に対応して NUL を含まない任意の文字列を対応付けるものであり, WAMの各行に対応して % で始まりWAMの行の name を記述した行と, それに続いて先頭が space で始まる, 対応させる文字列を記述した行とからなる. 各行の先頭の space 1文字は捨てられる. 行末を示す LF は文字列に含まれるが, mkaux(1) の引き数により, 捨てるということも可能である. また, % 形式以外に ^ 形式もあるが, 詳細は mkaux(1) を参照されたい. 例:

```
%key
 title title title....
 title title title....
%key2
%key3
 title title tttt
%key4
 ...
```

**SEE ALSO**

mkcw(1), mkxr(1), mkaux(1), maw(1)