NAME

speak – word to voice translator

SYNOPSIS

speak [**–efpsv**] [vocabulary [output]]

DESCRIPTION

Speak turns a stream of words into utterances and outputs them to a voice synthesizer, or to the specified *output.* It has facilities for maintaining a vocabulary. It receives, from the standard input

- working lines: text of words separated by blanks
- phonetic lines: strings of phonemes for one word preceded and separated by commas. The phonemes may be followed by comma-percent then a ‘replacement part’ – an ASCII string with no spaces. The phonetic code is given in *vs* (V).
- empty lines
- command lines: beginning with **!.** The following command lines are recognized:

!r file	replace coded vocabulary from file
!w file	write coded vocabulary on file
!p	print phonetics for working word
!l	list vocabulary on standard output with phonetics
!c word	copy phonetics from working word to specified word
!d	print decomposition of working word into substrings
!f *n*	turn off (or on) English preprocessing rule number *n* (see listing for meaning of *n)*

Each working line replaces its predecessor. Its first word is the ‘working word’. Each phonetic line replaces the phonetics stored for the working word. In particular, a phonetic line of comma only deletes the entry for the working word. Each working line, phonetic line or empty line causes the working line to be uttered. The process terminates at the end of input.

Unknown words are pronounced by rules, and failing that, are spelled. For the builtin part of the rules, see the reference. Spelling is done by taking each character of the word, prefixing it with ‘*’, and looking it up. Unspellable words burp.

Words not found verbatim in the vocabulary are pronounced piecewise. First the word is bracketed by sharps: ‘#...#’. The vocabulary is then searched for the longest fragment that matches the beginning of the word. The phonetic part of the phonetic string is uttered, and the matched fragment is replaced by the replacement part of the phonetic string, if any. The process is repeated until the word is exhausted. A fragment is entered into the vocabulary as a working word prefixed by ‘%’.

Speak is initialized with a coded vocabulary stored in file */usr/lib/speak.m.* The vocabulary option substitutes a different file for */usr/lib/speak.m.* Other vocabularies, to be used with option **–e**, exist in /usr/vs/latin.m and /usr/vs/polish.m.

A set of single letter options may appear in any order preceded by **–.** Their meanings are:

e	suppress English preprocessing
f	equivalent to ‘f1, f2,...’
p	suppress pronunciation by rule
s	suppress spelling
v	suppress voice output

The following input will reconstitute a coded vocabulary, ‘speak.m’, from an ascii listing, ‘speak.v’, that was created using **!l**.

```
(cat speak.v; echo !w speak.m) | speak –v /dev/null
```

FILES

/usr/lib/speak.m

SEE ALSO

M. D. McIlroy, ‘‘Synthetic English Speech by Rule,’’ Computing Science Technical Report #14, Bell Laboratories, 1973
vs (V), vs (IV)

BUGS

Excessively long words cause dumps.
Space is not reclaimed from changed entries; use **!w** and **!r** to effect reclamation.
!p doesn’t always work as advertised.