

NAME

morphinit, re_morphinit, morphstr, morphword

SYNOPSIS

```
#include "wn.h"

int morphinit(void);
int re_morphinit(void);
char *morphstr(char *origstr, int pos);
char *morphword(char *word, int pos);
```

DESCRIPTION

The WordNet morphological processor, Morphy, is accessed through these functions:

morphinit() is used to open the exception list files. It returns **0** if successful, **-1** otherwise. The exception list files must be opened before **morphstr()** or **morphword()** are called.

re_morphinit() is used to close the exception list files and reopen them, and is used exclusively for WordNet development. Return codes are as described above.

morphstr() is the basic user interface to Morphy. It tries to find the base form (lemma) of the word or collocation *origstr* in the specified *pos*. The first call (with *origstr* specified) returns a pointer to the first base form found. Subsequent calls requesting base forms of the same string must be made with the first argument of **NULL**. When no more base forms for *origstr* can be found, **NULL** is returned. Note that **morphstr()** returns a pointer to a static character buffer. A subsequent call to **morphstr()** with a new string (instead of **NULL**) will overwrite the string pointed to by a previous call. Users should copy the returned string into a local buffer, or use the C library function **strdup** to duplicate the returned string into a *malloc'd* buffer.

morphword() tries to find the base form of *word* in the specified *pos*. This function is called by **morphstr()** for each individual word in a collocation. Note that **morphword()** returns a pointer to a static character buffer. A subsequent call to **morphword()** will overwrite the string pointed to by a previous call. Users should copy the returned string into a local buffer, or use the C library function **strdup** to duplicate the returned string into a *malloc'd* buffer.

NOTES

morphinit() is called by **wninit()** and is not intended to be called directly by an application. Applications wishing to use WordNet and/or the morphological functions must call **wninit()** at the start of the program. See **wnutil(3WN)** for more information.

origstr may be either a word or a collocation formed by joining individual words with underscore characters (_).

Usually only **morphstr()** is called from applications, as it works on both words and collocations.

pos must be one of the following:

- | | |
|----------|---------------------|
| 1 | NOUN |
| 2 | VERB |
| 3 | ADJECTIVE |
| 4 | ADVERB |
| 5 | ADJECTIVE_SATELLITE |

If **ADJECTIVE_SATELLITE** is passed, it is treated by **morphstr()** as **ADJECTIVE**.

SEE ALSO

wnintro(3WN), **wnsearch(3WN)**, **wndb(5WN)**, **morphy(7WN)**.

WARNINGS

Passing an invalid part of speech will result in a core dump.

The WordNet database files must be open to use **morphstr()** or **morphword()**.

BUGS

Morphy will allow non-words to be converted to words, if they follow one of the rules described above. For example, it will happily convert **plantes** to **plants**.