**NAME**

wnintro − introduction to WordNet library functions

**DESCRIPTION**

This section of the *WordNet Reference Manual* contains manual pages that describe the WordNet library functions and API.

Functions are organized into the following categories:

| Category | Manual Page | Object File |
|---|---|---|
| Database Search | wnsearch (3WN) | search.o |
| Morphology | morph (3WN) | morph.o |
| Misc. Utility | wnutil (3WN) | wnutil.o |
| Binary Search | binsrch (3WN) | binsrch.o |

The WordNet library is used by all of the searching interfaces provided with the various WordNet packages. Additional programs in the system, such as **grind**(1WN), also use functions in this library.

The WordNet library is provided in both source and binary forms (on some platforms) to allow users to build applications and tools to their own specifications that utilize the WordNet database. We do not provide programming support or assistance.

The code conforms to ANSI C standards. Functions are defined with function prototypes. If you do not have a compiler that accepts prototypes, you must edit the source code and remove the prototypes before compiling. **gcc**(1), version 2.95.2 was used to compile all of the Unix libraries. The same code was compiled on all platforms, using **cpp**(1) **#ifdef** statements to conditionally compile platform specific code. The platform specific definitions are:

| | |
|---|---|
| **UNIX** | Defined for all Unix operating systems |
| **PC** | Defined for all Windows operating system |
| **WINDOWS** | Defined for Windows operating system |

Note that when **WINDOWS** is defined, **PC** is also defined. The redundancy goes back to when we supported both DOS and Windows.

**LIST OF WORDNET LIBRARY FUNCTIONS**

Not all library functions are listed below. Missing are mainly functions that are called by documented ones, or ones that were written for specific applications or tools used during WordNet development. Data structures are defined in **wntypes.h**.

**Database Searching Functions (search.o)**

**findtheinfo** Primary search function for WordNet database. Returns formatted search results in text buffer. Used by WordNet interfaces to perform requested search.

**findtheinfo_ds** Primary search function for WordNet database. Returns search results in linked list data structure.

**is_defined** Set bit for each search type that is valid for the search word passed and return bit mask.

**in_wn** Set bit for each syntactic category that search word is in.

**index_lookup** Find word in index file and return parsed entry in data structure. Input word must be exact match of string in database. Called by **getindex()**.

**getindex** Find word in index file, trying different techniques − replace hyphens

| | |
|---|---|
| | with underscores, replace underscores with hyphens, strip hyphens and underscores, strip periods. |
| **read_synset** | Read synset from data file at byte offset passed and return parsed entry in data structure. Calls **parse_synset()**. |
| **parse_synset** | Read synset at current byte offset in file and return parsed entry in data structure. |
| **free_syns** | Free a synset linked list allocated by **findtheinfo_ds()**. |
| **free_synset** | Free a synset structure. |
| **free_index** | Free an index structure. |
| **traceptrs_ds** | Recursive search algorithm to trace a pointer tree and return results in linked list. |
| **do_trace** | Do requested search on synset passed returning formatted output in buffer. |

**Morphology Functions (morph.o)**

| | |
|---|---|
| **morphinit** | Open exception list files. |
| **re_morphinit** | Close exception list files and reopen. |
| **morphstr** | Try to find base form (lemma) of word or collocation in syntactic category passed. Calls **morphword()** for each word in string passed. |
| **morphword** | Try to find base form (lemma) of individual word in syntactic category passed. |

**Utility Functions (wnutil.o)**

| | |
|---|---|
| **wninit** | Top level function to open database files and morphology exception lists. |
| **re_wninit** | Top level function to close and reopen database files and morphology exception lists. |
| **cntwords** | Count the number of underscore or space separated words in a string. |
| **strtolower** | Convert string to lower case and remove trailing adjective marker if found. |
| **ToLowerCase** | Convert string passed to lower case. |
| **strsubst** | Replace all occurrences of *from* with *to* in *str*. |
| **getptrtype** | Return code for pointer type character passed. |
| **getpos** | Return syntactic category code for string passed. |
| **getsstype** | Return synset type code for string passed. |
| **FmtSynset** | Reconstruct synset string from synset pointer. |
| **StrToPos** | Passed string for syntactic category, returns corresponding integer value. |
| **GetSynsetForSense** | Return synset for sense key passed. |
| **GetDataOffset** | Find synset offset for sense. |
| **GetPolyCount** | Find polysemy count for sense passed. |
| **GetWORD** | Return word part of sense key. |
| **GetPOS** | Return syntactic category code for sense key passed. |
| **WNSnsToStr** | Generate sense key for index entry passed. |
| **GetValidIndexPointer** | Search for string and/or base form of word in database and return index |

structure for word if found.

**GetWNSense** Return sense number in database for sense key.

**GetSenseIndex** Return parsed sense index entry for sense key passed.

**default_display_message** Default function to use as value of **display_message**. Simply returns **-1**.

**Binary Search Functions (binsrch.o)**

**bin_search** General purpose binary search function to search for key as first item on line in sorted file.

**copyfile** Copy contents from one file to another.

**replace_line** Replace a line in a sorted file.

**insert_line** Insert a line into a sorted file.

## LIST OF HEADER FILES

**license.h** Text of WordNet license in various C data structure formats

**setutil.h** Functions for creating and working with sets. Used to perform **RELATIVES** search.

**wn.h** Top level WordNet include file that includes most others. This should be sufficient for most applications. Also lists function prototypes for library API.

**wnconsts.h** Constants used by library functions and applications. Conditionally compiled code determines default pathnames and buffer sizes for different platforms,.

**wnglobal.h** External declarations for global variables initialized in **wnglobal.c**. These variables are static: they do not change while library code is run, and they do not influence how the library operates.

**wnhelp.h** External declaration for helptext initialized in **wnhelp.c**.

**wnrtl.h** External declarations for global variables and flags used and set by the WordNet library functions at run-time.

**wntypes.h** C typedefs for data structures used in library.

## NOTES

All library functions that access the database files expect the files to be open. The function **wninit**(3WN) must be called before other database access functions such as **findtheinfo**(3WN) or **read_synset**(3WN).

Inclusion of the header file **wn.h** is usually sufficient, as it includes several other WordNet header files.

The command line interface is a good example of a simple application that uses several WordNet library functions.

Many of the library functions are passed or return syntactic category or synset type information. The following table lists the possible categories as integer codes, synset type constant names, syntactic category constant names, single characters and character strings.

| Integer | Synset Type | Syntactic Category | Char | String |
|---|---|---|---|---|
| 1 | NOUN | NOUN | n | noun |
| 2 | VERB | VERB | v | verb |
| 3 | ADJ | ADJ | a | adj |
| 4 | ADV | ADV | r | adv |
| 5 | SATELLITE | ADJ | s | *n/a* |

**ENVIRONMENT VARIABLES**

**WNHOME** Base directory for WordNet. Unix default is **/usr/local/WordNet-2.0**, Windows default is **C:\Program Files\WordNet\2.0**.

**WNSEARCHDIR** Directory in which the WordNet database has been installed. Unix default is **WNHOME/dict**, Windows default is **WNHOME\dict**.

**FILES**

| | |
|---|---|
| **WNHOME/lib/libwn.a** | WordNet library (Unix) |
| **WNHOME\lib\wn.lib** | WordNet library (Windows) |
| **WNHOME/include/***| header files for use with WordNet library (Unix) |
| **WNHOME\include***| header files for use with WordNet library (Windows) |

**SEE ALSO**

**wnintro**(1WN), **binsrch**(3WN), **morph**(3WN), **wnsearch**(3WN), **wnutil**(3WN), **wnintro**(5WN), **wnintro**(7WN).

Fellbaum, C. (1998), ed. "*WordNet: An Electronic Lexical Database*". MIT Press, Cambridge, MA.

**BUGS**

Please report bugs to **wordnet@princeton.edu**.