NAME

catopen, catclose – open/close a message catalog

SYNOPSIS

#include <nl_types.h>

nl_catd catopen(char **name*, **int** *oflag***);**

int catclose(nl_catd *catd***);**

MT-LEVEL

MT-Safe

DESCRIPTION

catopen( ) opens a message catalog and returns a message catalog descriptor. *name* specifies the name of the message catalog to be opened. If *name* contains a "/", then *name* specifies a complete pathname for the message catalog; otherwise, the environment variable **NLSPATH** is used and **/usr/lib/locale/***locale***/LC_MESSAGES** must exist. If **NLSPATH** does not exist in the environment, or if a message catalog cannot be opened in any of the paths specified by **NLSPATH**, then the default path **/usr/lib/locale/***locale***/LC_MESSAGES** is used. In C locale, **catopen( )** will always succeed without even looking at the implementation defined default search path.

The names of message catalogs, and their location in the filesystem, can vary from one system to another. Individual applications can choose to name or locate message catalogs according to their own special needs. A mechanism is therefore required to specify where the catalog resides.

The **NLSPATH** variable provides both the location of message catalogs, in the form of a search path, and the naming conventions associated with message catalog files. For example:

NLSPATH=/nlslib/%L/%N.cat:/nlslib/%N/%L

The metacharacter **%** introduces a substitution field, where **%L** substitutes the current setting of either the **LANG** environment variable, if the value of *oflag* is **0**, or the **LC_MESSAGES** category, if the value of *oflag* is **NL_CAT_LOCALE**, and **%N** substitutes the value of the *name* parameter passed to **catopen( )**. Thus, in the above example, **catopen( )** will search in **/nlslib/$LANG/***name***.cat**, if *oflag* is **0**, or in **/nlslib/{LC_MESSAGES}/***name***.cat**, if *oflag* is **NL_CAT_LOCALE**.

NLSPATH will normally be set up on a system wide basis (in **/etc/profile**) and thus makes the location and naming conventions associated with message catalogs transparent to both programs and users.

The full set of metacharacters is:

- **%N** The value of the name parameter passed to **catopen( )**.
- **%L** The value of **LANG** or **LC_MESSAGES**.
- **%l** The value of the *language* element of **LANG** or **LC_MESSAGES**.
- **%t** The value of the *territory* element of **LANG** or **LC_MESSAGES**.
- **%c** The value of the *codeset* element of **LANG** or **LC_MESSAGES**.
- **%%** A single %.

The **LANG** environment variable provides the ability to specify the user's requirements for native languages, local customs and character set, as an ASCII string in the form

LANG=language[_territory[.codeset]]

A user who speaks German as it is spoken in Austria and has a terminal which operates in ISO 8859/1 codeset, would want the setting of the **LANG** variable to be

LANG=De_A.88591

With this setting it should be possible for that user to find any relevant catalogs should they exist.

Should the **LANG** variable not be set then the value of **LC_MESSAGES** as returned by **setlocale( )** is used. If this is **NULL** then the default path as defined in **nl_types( )** is used.

If the value of *oflag* argument is **0**, the **LANG** environment variable is used to locate the catalogue without regard to the **LC_MESSAGES** category. If the *oflag* argument is **NL_CAT_LOCALE**, the **LC_MESSAGES** category is used to locate the message catalogue.

catclose( ) closes the message catalog identified by *catd*.

RETURN VALUES

If successful, **catopen( )** returns a message catalog descriptor for use on subsequent calls to **catgets( )** and **catclose( )**; otherwise **catopen( )** returns **(nl_catd)** –**1**.

catclose( ) returns **0** if successful, otherwise –**1**.

SEE ALSO

gencat(1), **gettext**(3I), **catgets**(3C), **setlocale**(3C), **environ**(5), **nl_types**(5)

NOTES

catopen and **catclose** can be used safely in a multi-thread application, as long as **setlocale**(3C) is not being called to change the locale.