NAME

srchtxt – display contents of, or search for a text string in, message data bases

SYNOPSIS

srchtxt [–**s**] [–**l** *locale*] [–**m** *msgfile*, . . .] [*text*]

AVAILABILITY

SUNWloc

DESCRIPTION

The **srchtxt** utility is used to display all the text strings in message data bases, or to search for a text string in message data bases (see **mkmsgs**(1)). These data bases are files in the directory **/usr/lib/locale/***locale***/LC_MESSAGES** (see **setlocale**(3C)), unless a file name given with the –**m** option contains a **/**. The directory *locale* can be viewed as the name of the language in which the text strings are written. If the –**l** option is not specified, the files accessed will be determined by the value of the environment variable **LC_MESSAGES**. If **LC_MESSAGES** is not set, the files accessed will be determined by the value of the environment variable **LANG**. If **LANG** is not set, the files accessed will be in the directory **/usr/lib/locale//C/LC_MESSAGES**, which contains default strings.

If no *text* argument is present, then all the text strings in the files accessed will be displayed.

If the –**s** option is not specified, the displayed text is prefixed by message sequence numbers. The message sequence numbers are enclosed in angle brackets: **<***msgfile***:***msgnum***>.**

msgfile name of the file where the displayed text occurred

msgnum sequence number in *msgfile* where the displayed text occurred

This display is in the format used by **gettxt**(1) and **gettxt**(3C).

OPTIONS

–**s** Suppress printing of the message sequence numbers of the messages being displayed.

–**l** *locale* Access files in the directory **/usr/lib/locale/***locale***/LC_MESSAGES**. If –**m** *msgfile* is also supplied, *locale* is ignored for *msgfile*s containing a **/**.

–**m** *msgfile* Access files specified by one or more *msgfile*s. If *msgfile* contains a **/** character, then *msgfile is* interpreted as a pathname; otherwise, it will be assumed to be in the directory determined as described above. To specify more than one *msgfile*, separate the file names using commas.

text Search for the text string specified by *text* and display each one that matches. *text* can take the form of a regular expression; see **regexp**(5).

EXAMPLES

The following examples show uses of **srchtxt**.

Example 1:

If message files have been installed in a locale named **french** by using **mkmsgs**(1), then you could display the entire set of text strings in the **french** locale (**/usr/lib/locale/french/LC_MESSAGES/***) by typing:

example% srchtxt –l french

Example 2:

If a set of error messages associated with the operating system have been installed in the file **UX** in the **french** locale (**/usr/lib/locale/french/LC_MESSAGES/UX**), then, using the value of the **LANG** environment variable to determine the locale to be searched, you could search that file in that locale for all error messages dealing with files by typing:

example% setenv LANG=french; export LANG
example% srchtxt –m UX "[Ff]ichier"

If **/usr/lib/locale/french/LC_MESSAGES/UX** contained the following strings:

```
Erreur E/S\n
Liste d'arguments trop longue\n
Fichier inexistant\n
Argument invalide\n
Trop de fichiers ouverts\n
Fichier trop long\n
Trop de liens\n
Argument hors du domaine\n
Identificateur supprim\n
Etreinte fatale\n
  .
  .
  .
```

then the following strings would be displayed:

```
<UX:3>Fichier inexistant\n
<UX:5>Trop de fichiers ouverts\n
<UX:6>Fichier trop long\n
```

Example 3:

If a set of error messages associated with the operating system have been installed in the file **UX** and a set of error messages associated with the INGRESS data base product have been installed in the file **ingress**, both in the **german** locale, then you could search for the pattern **[Dd]atei** in both the files **UX** and **ingress** in the **german** locale by typing:

example% srchtxt –l german –m UX,ingress "[Dd]atei"

ENVIRONMENT

If any of the **LC_*** variables (**LC_CTYPE, LC_MESSAGES, LC_TIME, LC_COLLATE, LC_NUMERIC,** and **LC_MONETARY**) (see **environ**(5)) are not set in the environment, the operational behavior of **srchtxt** for each corresponding locale category is determined by the value of the **LANG** environment variable. If **LC_ALL** is set, its contents are used to override both the **LANG** and the other **LC_*** variables. If none of the above variables is set in the environment, the "C" (U.S. style) locale determines how **srchtxt** behaves.

LC_CTYPE Determines how **srchtxt** handles characters. When **LC_CTYPE** is set to a valid value, **srchtxt** can display and handle text and filenames containing valid characters for that locale. **srchtxt** can display and handle Extended Unix Code (EUC) characters where any individual character can be 1, 2, or 3 bytes wide. **srchtxt** can also handle EUC characters of 1, 2, or more column widths. In the "C" locale, only characters from ISO 8859-1 are valid.

FILES

/usr/lib/locale/C/LC_MESSAGES/*	default files created by **mkmsgs**(1)
/usr/lib/locale/*locale***/LC_MESSAGES/***	message files created by **mkmsgs**(1)

SEE ALSO

exstr(1), **gettxt**(1), **mkmsgs**(1), **gettxt**(3C), **setlocale**(3C), **environ**(5), **regexp**(5)

DIAGNOSTICS

The error messages produced by **srchtxt** are intended to be self-explanatory. They indicate an error in the command line or errors encountered while searching for a particular locale and/or message file.