Full credit is given to the above companies including the OS that this PDF file was generated!

Red Hat Enterprise Linux Release 9.2 Manual Pages on 'cat.1p' command

$ man cat.1p

CAT(1P) POSIX Programmer's Manual CAT(1P)

PROLOG

This manual page is part of the POSIX Programmer's Manual. The Linux implementation of this interface may differ (consult the corresponding Linux manual page for details of Linux behavior), or the interface may not be implemented on Linux.

NAME

cat ? concatenate and print files

SYNOPSIS

cat [-u] [file...]

DESCRIPTION

The cat utility shall read files in sequence and shall write their con? tents to the standard output in the same sequence.

OPTIONS

The cat utility shall conform to the Base Definitions volume of POSIX.1?2017, Section 12.2, Utility Syntax Guidelines.

The following option shall be supported:

-u Write bytes from the input file to the standard output with? out delay as each is read.

OPERANDS

The following operand shall be supported:

file A pathname of an input file. If no file operands are speci? fied, the standard input shall be used. If a file is '-', the

cat utility shall read from the standard input at that point in the sequence. The cat utility shall not close and reopen standard input when it is referenced in this way, but shall accept multiple occurrences of '-' as a file operand.

STDIN

The standard input shall be used only if no file operands are speci? fied, or if a file operand is '-'. See the INPUT FILES section.

INPUT FILES

The input files can be any file type.

ENVIRONMENT VARIABLES

The following environment variables shall affect the execution of cat:

LANG Provide a default value for the internationalization vari? ables that are unset or null. (See the Base Definitions vol? ume of POSIX.1?2017, Section 8.2, Internationalization Vari? ables for the precedence of internationalization variables used to determine the values of locale categories.)

LC_ALL If set to a non-empty string value, override the values of all the other internationalization variables.

LC_CTYPE Determine the locale for the interpretation of sequences of bytes of text data as characters (for example, single-byte as opposed to multi-byte characters in arguments).

LC_MESSAGES

Determine the locale that should be used to affect the format and contents of diagnostic messages written to standard er? ror.

NLSPATH Determine the location of message catalogs for the processing of LC_MESSAGES.

ASYNCHRONOUS EVENTS

Default.

STDOUT

The standard output shall contain the sequence of bytes read from the input files. Nothing else shall be written to the standard output. If the standard output is a regular file, and is the same file as any of

the input file operands, the implementation may treat this as an error.

STDERR

The standard error shall be used only for diagnostic messages.

OUTPUT FILES

None.

EXTENDED DESCRIPTION

None.

EXIT STATUS

The following exit values shall be returned:

0 All input files were output successfully.

>0 An error occurred.

CONSEQUENCES OF ERRORS

Default.

The following sections are informative.

APPLICATION USAGE

The -u option has value in prototyping non-blocking reads from FIFOs.

The intent is to support the following sequence:

```
mkfifo foo
cat -u foo > /dev/tty13 &
cat -u > foo
```

It is unspecified whether standard output is or is not buffered in the default case. This is sometimes of interest when standard output is as? sociated with a terminal, since buffering may delay the output. The presence of the -u option guarantees that unbuffered I/O is available.

It is implementation-defined whether the cat utility buffers output if the -u option is not specified. Traditionally, the -u option is imple? mented using the equivalent of the setvbuf() function defined in the System Interfaces volume of POSIX.1?2017.

EXAMPLES

The following command:

```
cat myfile
```

writes the contents of the file myfile to standard output.

The following command:

cat doc1 doc2 > doc.all

concatenates the files doc1 and doc2 and writes the result to doc.all.

Because of the shell language mechanism used to perform output redi? rection, a command such as this:

cat doc doc.end > doc

causes the original data in doc to be lost before cat even begins exe? cution. This is true whether the cat command fails with an error or silently succeeds (the specification allows both behaviors). In order to append the contents of doc.end without losing the original contents of doc, this command should be used instead:

cat doc.end >> doc

The command:

cat start - middle - end > file

when standard input is a terminal, gets two arbitrary pieces of input from the terminal with a single invocation of cat. Note, however, that if standard input is a regular file, this would be equivalent to the command:

cat start - middle /dev/null end > file

because the entire contents of the file would be consumed by cat the first time '-' was used as a file operand and an end-of-file condition would be detected immediately when '-' was referenced the second time.

RATIONALE

Historical versions of the cat utility include the -e, -t, and -v, op? tions which permit the ends of lines, <tab> characters, and invisible characters, respectively, to be rendered visible in the output. The standard developers omitted these options because they provide too fine a degree of control over what is made visible, and similar output can be obtained using a command such as:

sed -n l pathname

The latter also has the advantage that its output is unambiguous, whereas the output of historical cat -etv is not.

The -s option was omitted because it corresponds to different functions in BSD and System V-based systems. The BSD -s option to squeeze blank

lines can be accomplished by the shell script shown in the following example:

```
sed -n '
# Write non-empty lines.
/./   {
      p
      d
      }
# Write a single empty line, then look for more empty lines.
/^$/  p
# Get next line, discard the held <newline> (empty line),
# and look for more empty lines.
:Empty
/^$/  {
      N
      s/.//
      b Empty
      }
# Write the non-empty line before going back to search
# for the first in a set of empty lines.
      p
'
```

The System V -s option to silence error messages can be accomplished by redirecting the standard error. Note that the BSD documentation for cat uses the term ``blank line'' to mean the same as the POSIX ``empty line'': a line consisting only of a <newline>.

The BSD -n option was omitted because similar functionality can be ob? tained from the -n option of the pr utility.

FUTURE DIRECTIONS

None.

SEE ALSO

more

The Base Definitions volume of POSIX.1?2017, Chapter 8, Environment

Variables, Section 12.2, Utility Syntax Guidelines

The System Interfaces volume of POSIX.1?2017, setvbuf()

COPYRIGHT