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 'fmtmsg.h.0p' command

$ man fmtmsg.h.0p

fmtmsg.h(0P) POSIX Programmer's Manual fmtmsg.h(0P)

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

fmtmsg.h ? message display structures

SYNOPSIS

#include <fmtmsg.h>

DESCRIPTION

The <fmtmsg.h> header shall define the following symbolic constants:

MM_HARD Source of the condition is hardware.

MM_SOFT Source of the condition is software.

MM_FIRM Source of the condition is firmware.

MM_APPL Condition detected by application.

MM_UTIL Condition detected by utility.

MM_OPSYS Condition detected by operating system.

MM_RECOVER Recoverable error.

MM_NRECOV Non-recoverable error.

MM_HALT Error causing application to halt.

MM_ERROR Application has encountered a non-fatal fault.

MM_WARNING Application has detected unusual non-error condition.

MM_INFO Informative message.

MM_NOSEV No severity level provided for the message.

MM_PRINT Display message on standard error.

MM_CONSOLE Display message on system console.

The table below indicates the null values and identifiers for fmtmsg() arguments. The <fmtmsg.h> header shall define the symbolic constants in the Identifier column, which shall have the type indicated in the Type column:

Argument	Type	Null-Value	Identifier
label	char *	(char*)0	MM_NULLLBL
severity	int	0	MM_NULLSEV
class	long	0L	MM_NULLMC
text	char *	(char*)0	MM_NULLTXT
action	char *	(char*)0	MM_NULLACT
tag	char *	(char*)0	MM_NULLTAG

The <fmtmsg.h> header shall also define the following symbolic con? stants for use as return values for fmtmsg():

MM_OK The function succeeded.

MM_NOTOK The function failed completely.

MM_NOMSG The function was unable to generate a message on standard error, but otherwise succeeded.

MM_NOCON The function was unable to generate a console message, but otherwise succeeded.

The following shall be declared as a function and may also be defined as a macro. A function prototype shall be provided.

```
int fmtmsg(long, const char *, int,
    const char *, const char *, const char *);
```

The following sections are informative.

APPLICATION USAGE

None.

RATIONALE

None.

FUTURE DIRECTIONS

None.

SEE ALSO

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

COPYRIGHT