NAME

inotify_init, inotify_init1 − initialize an inotify instance

LIBRARY

Standard C library (*libc*, *−lc*)

SYNOPSIS

#include <sys/inotify.h>

int inotify_init(void);
int inotify_init1(int *fla gs***);**

DESCRIPTION

For an overview of the inotify API, see **inotify**(7).

inotify_init() initializes a new inotify instance and returns a file descriptor associated with a new inotify event queue.

If *flags* is 0, then **inotify_init1**() is the same as **inotify_init**(). The following values can be bitwise ORed in *flags* to obtain different behavior:

IN_NONBLOCK
Set the **O_NONBLOCK** file status flag on the open file description (see **open**(2)) referred to by the new file descriptor. Using this flag saves extra calls to **fcntl**(2) to achieve the same result.

IN_CLOEXEC
Set the close-on-exec (**FD_CLOEXEC**) flag on the new file descriptor. See the description of the **O_CLOEXEC** flag in **open**(2) for reasons why this may be useful.

RETURN VALUE

On success, these system calls return a new file descriptor. On error, −1 is returned, and *errno* is set to indicate the error.

ERRORS

EINVAL
(**inotify_init1**()) An invalid value was specified in *fla gs*.

EMFILE
The user limit on the total number of inotify instances has been reached.

EMFILE
The per-process limit on the number of open file descriptors has been reached.

ENFILE
The system-wide limit on the total number of open files has been reached.

ENOMEM
Insufficient kernel memory is available.

VERSIONS

inotify_init() first appeared in Linux 2.6.13; library support was added in glibc 2.4. **inotify_init1**() w as added in Linux 2.6.27; library support was added in glibc 2.9.

STANDARDS

These system calls are Linux-specific.

SEE ALSO

inotify_add_watch(2), **inotify_rm_watch**(2), **inotify**(7)