## NAME

inotify_add_watch − add a watch to an initialized inotify instance

## SYNOPSIS

**#include <sys/inotify.h>**

**int inotify_add_watch(int** *fd*, **const char *** *pathname*, **uint32_t** *mask*);

## DESCRIPTION

**inotify_add_watch**() adds a new watch, or modifies an existing watch, for the file whose location is specified in *pathname*; the caller must have read permission for this file. The *fd* argument is a file descriptor referring to the inotify instance whose watch list is to be modified. The events to be monitored for *pathname* are specified in the *mask* bit-mask argument. See **inotify**(7) for a description of the bits that can be set in *mask*.

A successful call to **inotify_add_watch**() returns a unique watch descriptor for this inotify instance, for the filesystem object (inode) that corresponds to *pathname*. If the filesystem object was not previously being watched by this inotify instance, then the watch descriptor is newly allocated. If the filesystem object was already being watched (perhaps via a different link to the same object), then the descriptor for the existing watch is returned.

The watch descriptor is returned by later **read**(2)s from the inotify file descriptor. These reads fetch *inotify_event* structures (see **inotify**(7)) indicating filesystem events; the watch descriptor inside this structure identifies the object for which the event occurred.

## RETURN VALUE

On success, **inotify_add_watch**() returns a nonnegative watch descriptor. On error, −1 is returned and *errno* is set appropriately.

## ERRORS

**EACCES**
Read access to the given file is not permitted.

**EBADF**
The given file descriptor is not valid.

**EEXIST**
*mask* contains **IN_MASK_CREATE** and *pathname* refers to a file already being watched by the same *fd*.

**EFAULT**
*pathname* points outside of the process's accessible address space.

**EINVAL**
The given event mask contains no valid events; or *mask* contains both **IN_MASK_ADD** and **IN_MASK_CREATE**; or *fd* is not an inotify file descriptor.

**ENAMETOOLONG**
*pathname* is too long.

**ENOENT**
A directory component in *pathname* does not exist or is a dangling symbolic link.

**ENOMEM**
Insufficient kernel memory was available.

**ENOSPC**
The user limit on the total number of inotify watches was reached or the kernel failed to allocate a needed resource.

**ENOTDIR**
*mask* contains **IN_ONLYDIR** and *pathname* is not a directory.

## VERSIONS

Inotify was merged into the 2.6.13 Linux kernel.

## CONFORMING TO

This system call is Linux-specific.

## SEE ALSO

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

## COLOPHON

This page is part of release 5.05 of the Linux *man-pages* project. A description of the project, information about reporting bugs, and the latest version of this page, can be found at https://www.kernel.org/doc/man−pages/.