NAME

"IO::Async::File" – watch a file for changes

SYNOPSIS

```
use IO::Async::File;

use IO::Async::Loop;
my $loop = IO::Async::Loop->new;

my $file = IO::Async::File->new(
   filename => "config.ini",
   on_mtime_changed => sub {
      my ( $self ) = @_;
      print STDERR "Config file has changed\n";
      reload_config( $self->handle );
   }
);

$loop->add( $file );

$loop->run;
```

DESCRIPTION

This subclass of IO::Async::Notifier watches an open filehandle or named filesystem entity for changes in its `stat()` fields. It invokes various events when the values of these fields change. It is most often used to watch a file for size changes; for this task see also IO::Async::FileStream.

While called "File", it is not required that the watched filehandle be a regular file. It is possible to watch anything that `stat(2)` may be called on, such as directories or other filesystem entities.

EVENTS

The following events are invoked, either using subclass methods or CODE references in parameters.

on_dev_changed `$new_dev`**,** `$old_dev`
on_ino_changed `$new_ino`**,** `$old_ino`
...
on_ctime_changed `$new_ctime`**,** `$old_ctime`

Invoked when each of the individual `stat()` fields have changed. All the `stat()` fields are supported apart from `blocks` and `blksize`. Each is passed the new and old values of the field.

on_devino_changed `$new_stat`**,** `$old_stat`

Invoked when either of the `dev` or `ino` fields have changed. It is passed two File::stat instances containing the complete old and new `stat()` fields. This can be used to observe when a named file is renamed; it will not be observed to happen on opened filehandles.

on_stat_changed `$new_stat`**,** `$old_stat`

Invoked when any of the `stat()` fields have changed. It is passed two File::stat instances containing the old and new `stat()` fields.

PARAMETERS

The following named parameters may be passed to `new` or `configure`.

handle => IO

The opened filehandle to watch for `stat()` changes if `filename` is not supplied.

filename => STRING

Optional. If supplied, watches the named file rather than the filehandle given in `handle`. The file will be opened for reading and then watched for renames. If the file is renamed, the new filename is opened and tracked similarly after closing the previous file.

interval => NUM

Optional. The interval in seconds to poll the filehandle using `stat(2)` looking for size changes. A default of 2 seconds will be applied if not defined.

METHODS

handle

```
$handle = $file->handle
```

Returns the filehandle currently associated with the instance; either the one passed to the `handle` parameter, or opened from the `filename` parameter.

AUTHOR

Paul Evans <leonerd@leonerd.org.uk>