NAME

"IO::Async::Debug" − debugging control and support for IO::Async

DESCRIPTION

The following methods and behaviours are still experimental and may change or even be removed in future.

Debugging support is enabled by an environment variable called `IO_ASYNC_DEBUG` having a true value.

When debugging is enabled, the `make_event_cb` and `invoke_event` methods on IO::Async::Notifier (and their `maybe_` variants) are altered such that when the event is fired, a debugging line is printed, using the `debug_printf` method. This identifes the name of the event.

By default, the line is only printed if the caller of one of these methods is the same package as the object is blessed into, allowing it to print the events of the most-derived class, without the extra verbosity of the lower-level events of its parent class used to create it. All calls regardless of caller can be printed by setting a number greater than 1 as the value of `IO_ASYNC_DEBUG`.

By default the debugging log goes to `STDERR`, but two other environment variables can redirect it. If `IO_ASYNC_DEBUG_FILE` is set, it names a file which will be opened for writing, and logging written into it. Otherwise, if `IO_ASYNC_DEBUG_FD` is set, it gives a file descriptor number that logging should be written to. If opening the named file or file descriptor fails then the log will be written to `STDERR` as normal.

Extra debugging flags can be set in a comma-separated list in an environment variable called `IO_ASYNC_DEBUG_FLAGS`. The presence of these flags can cause extra information to be written to the log. Full details on these flags will be documented by the implementing classes. Typically these flags take the form of one or more capital letters indicating the class, followed by one or more lowercase letters enabling some particular feature within that class.

AUTHOR

Paul Evans <leonerd@leonerd.org.uk>