Test runs with different WriteModes

# What is measured

The program initializes the logger, and then writes a huge number of log entries from multiple parallel threads. Each log entry is written with a timestamp that is in all cases created by the application thread. The diagrams show how often a certain time interval between consecutive log entries from the same thread occurred.  
No measures were taken to “silence” the rest of the machine.  
The x-axis is in nanoseconds in all diagrams.

# WriteMode::Direct, without cleanup thread: Peak at about 120ns, outliers at 100ms

* Direct writing causes file I/O with every log message
* Outliers are caused by file rotation and cleanup.

# WriteMode::Buffered, without cleanup thread: Peak below 80ns, outliers at 100ms

* Buffered writing reduces file I/O
* Outliers are caused by file rotation and cleanup

# WriteMode::Buffered, with cleanup thread: Peak below 80ns, outliers at 4-7ms

* Impact of file rotation and cleanup is minimized by doing it in a separate thread

# WriteMode::Async: Peaks at below 10 and below 50ns, very flat tail, no outliers

* A separate thread is doing all (buffered) file I/O, and the cleanup
* Impact of logging on application threads is minimized, pushback hardly recognizable
* Additional dependency to crossbeam