

Synchronization

Your project code should always be thread-safe, but for Project 3, you may not use a single global lock around the entire file system. It is fine to have a global lock around the buffer cache, but **you may not perform blocking I/O with the global lock held!** The key point is operations that are independent (e.g. operating on different files, or different parts of the same file) should be able to issue disk I/O operations concurrently, without one waiting for the other to complete. If Thread A and Thread B are performing independent operations, and Thread B is not blocked on I/O, then it's fine for Thread A to block waiting for Thread B.

Note: On a multicore system, it may be desirable to allow Thread A and Thread B to execute concurrently, even if neither is blocked on I/O, to better take advantage of multiple cores. For example, one may prefer to not have a global lock around the buffer cache so that multiple cores can scan the buffer cache at the same time. But, given that Pintos does not support multicore systems, we are not requiring this.

What does it mean for two operations to be "independent?" For this project, **operations are considered independent if they are acting on different disk sectors, and such operations should be allowed to execute concurrently.** If two operations are writing to the same sector or extending the same file, then they are *not* considered independent and you may serialize those operations to maintain data consistency. Concurrent reads are not required.

Here are some examples. Assume that we have have the following file descriptors.

```
int notes = open("/my_files/notes.txt");  
int test = open("/my_files/test.c");
```

`read(notes)` and `write(test)` should be allowed to run concurrently since they operate on two different files that are stored on different sectors. `read(notes)` and `write(notes)` need not be allowed to run concurrently since they operate on the same sector. Note that `open` starts at the beginning of the file, so they operate on sector 0. `read(notes)` and `read(notes)` need not be allowed to run concurrently since they read from the same sector. This requirement applies to all tasks. If you added a global file system lock in Project User Programs, remember to remove it!

