# Questions

* What does it actually mean to mount something?
  + What data do we need to actually store?
* Can two clients open the same file concurrently?
  + If so, can they write to the same file at the same time?
    - If so, what does the result become?
  + If you delete or change using another client process, what does another client process see?
* Can you only delete a file if no one has it open?
  + Do you need to have a file open to delete it?
* What type of consistency do we need?
* Global lock vs. file-level lock
* Are we queueing requests if the file is not currently available?
  + If so, does “delete” have higher priority over other operations?
* Can the same process open a file, and do both read and write operation on the file?
  + If so, where does the read/write get indexed? For instance, where does the write go to?
* How big are the files we are working with?

# Ideas

* Hash table at server to keep track of locks on files. Use hash of file path.
* On fsOpen with read, server sends the client a full copy of the file to read locally. A client does not see the effects of writes to a file until it does another fsOpen to the file. If a lock is acquired on fsOpen, this achieves entry consistency