

Summary Dialogue on Persistence

Student: *Wow, file systems seem interesting(!), and yet complicated.*

Professor: *That's why me and my spouse do our research in this space.*

Student: *Hold on. Are you one of the professors who wrote this book? I thought we were both just fake constructs, used to summarize some main points, and perhaps add a little levity in the study of operating systems.*

Professor: *Uh... er... maybe. And none of your business! And who did you think was writing these things? (sighs) Anyway, let's get on with it: what did you learn?*

Student: *Well, I think I got one of the main points, which is that it is much harder to manage data for a long time (persistently) than it is to manage data that isn't persistent (like the stuff in memory). After all, if your machine crashes, memory contents disappear! But the stuff in the file system needs to live forever.*

Professor: *Well, as my friend Kevin Hultquist used to say, "Forever is a long time"; while he was talking about plastic golf tees, it's especially true for the garbage that is found in most file systems.*

Student: *Well, you know what I mean! For a long time at least. And even simple things, such as updating a persistent storage device, are complicated, because you have to care what happens if you crash. Recovery, something I had never even thought of when we were virtualizing memory, is now a big deal!*

Professor: *Too true. Updates to persistent storage have always been, and remain, a fun and challenging problem.*

Student: *I also learned about cool things like disk scheduling, and about data protection techniques like RAID and even checksums. That stuff is cool.*

Professor: *I like those topics too. Though, if you really get into it, they can get a little mathematical. Check out some of the latest on erasure codes if you want your brain to hurt.*

Student: *I'll get right on that.*

Professor: *(frowns)* I think you're being sarcastic. Well, what else did you like?

Student: *And I also liked all the thought that has gone into building technology-aware systems, like FFS and LFS. Neat stuff! Being disk aware seems cool. But will it matter anymore, with Flash and all the newest, latest technologies?*

Professor: *Good question! And a reminder to get working on that Flash chapter... (scribbles note down to self) ... But yes, even with Flash, all of this stuff is still relevant, amazingly. For example, Flash Translation Layers (FTLs) use log-structuring internally, to improve performance and reliability of Flash-based SSDs. And thinking about locality is always useful. So while the technology may be changing, many of the ideas we have studied will continue to be useful, for a while at least.*

Student: *That's good. I just spent all this time learning it, and I didn't want it to all be for no reason!*

Professor: *Professors wouldn't do that to you, would they?*