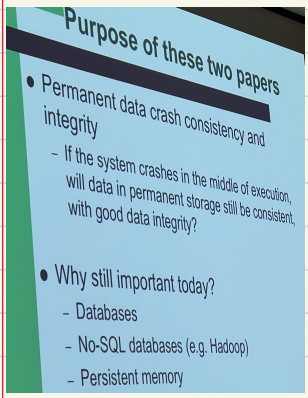
11/9 (Quiz 3, 10/28-11/9, will include the following two papers)

- Gregory R. Ganger, Marshall Kirk McKusick, Craig A.N. Soules, and Yale N. Patt. "Soft Updates: A Solution to the Metadata Update Problem in File Systems," ACM Transactions on Computer Systems, Vol. 18, No. 2, May 2000, Pages 127-153.
- P. M. Chen, W. T. Ng, S. Chandra, C. Aycock, G. Rajamni, and D. Lowell, "The Rio File Cache: Surviving Operating System Crashes," In *Proceedings of the Seventh International Conference on Architectural Support for Programming Languages and Operating Systems*, SIGPLAN Notices 31(9):74-83, September 1996.

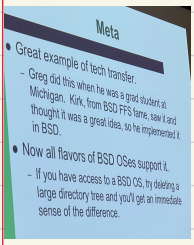
Optional further readings on Rio:
David E. Lowell and Peter M. Chen, Free Transactions with Rio Vista, In *Proceedings of the Sixteenth ACM Symposium on Operating Systems Principles* October 1997, St. Malo, France, pp. 92.101.
Q: Do you believe the underlying argument in Rio? Why or why not? Would you use a system that recovered data using Rio?



checkpoint : does it go thru?

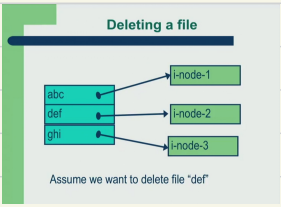
Note: Software company – Amazon
traditional company – Walmart

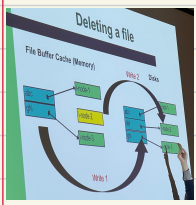
what's the difference.



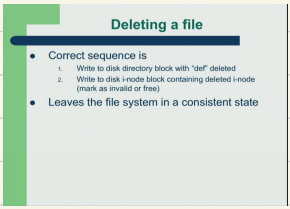
inode : only data write back
inode are sync

Write Example



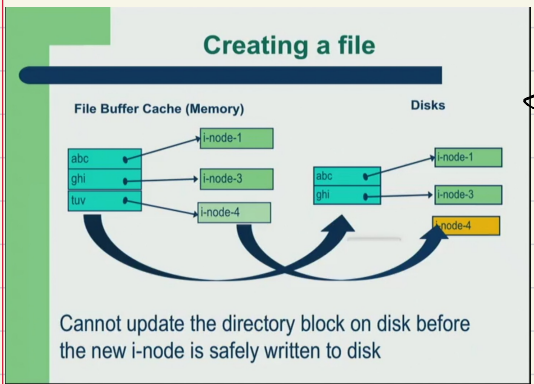


Seq : 1 -> 2



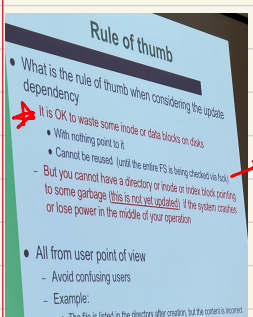
cannot remove i-node on disk (or mark as free) before update the directory block to remove def entry

Creating a File

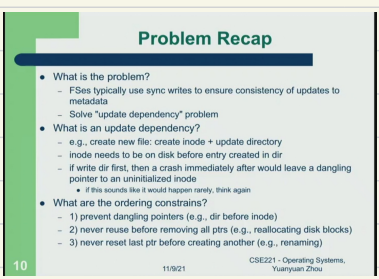


never have invalid pointer.

Rule of thumb

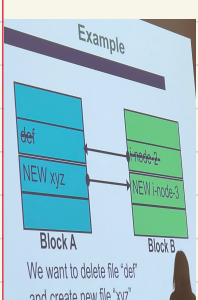


→ these block will be invalid until recheck.



Ex.

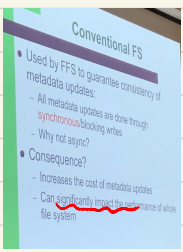
circular deps



we will never in such situation.

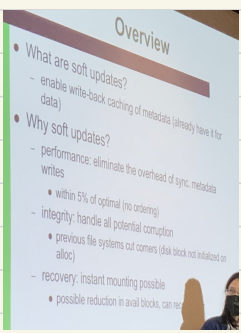
we do this sync, if deleted

do it first. do one at a time

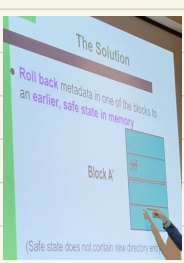


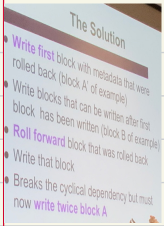
performance penlty.

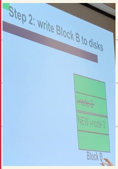
main point

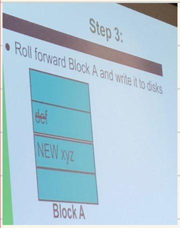


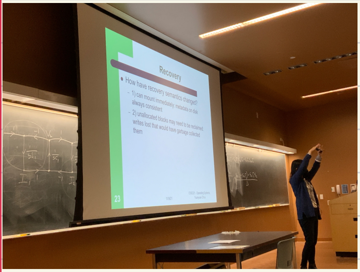
Algs to calculate state.

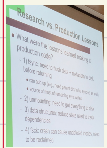


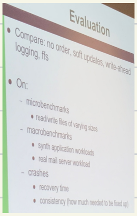

The Solution
Write first block with metadata that were rolled back (block A' of example)
Write blocks that can be written after first block has been written (block B of example)
Roll forward block that was rolled back
Write that block
Breaks the cyclical dependency but must now write twice block A


Step 2: write Block B to disks
~~i-node-2~~
NEW i-node-3
Block B

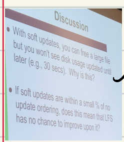

Step 3:
Roll forward Block A and write it to disks
~~def~~
NEW xyz
Block A


Recovery
How have recovery semantics changed?
1) can mount immediately: metadata on disk always consistent
2) unallocated blocks may need to be reclaimed: writes lost that would have garbage collected them
23
11/9/21
CSE221 - Operating Systems, Yuanyuan Zhou





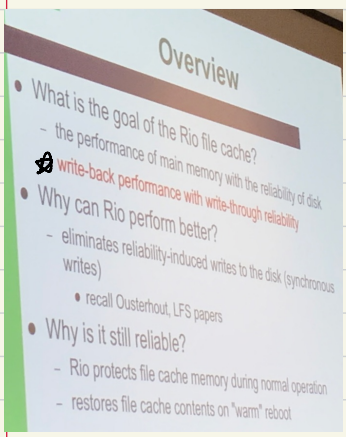
thy do → fault injection → use bad harddrive?

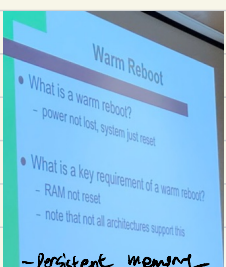


↗ updated later.

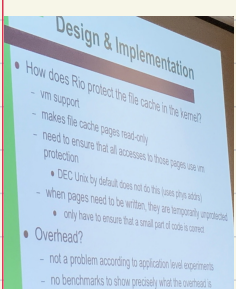
Pure Academic Paper no application

RIO

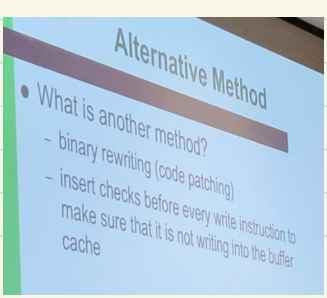


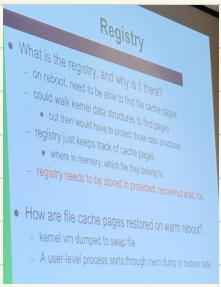


Prevent DATA corruption

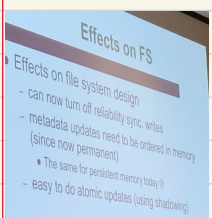


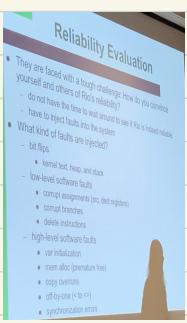
turn on the protection after write.

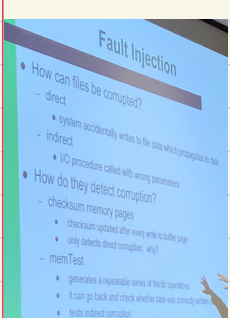




file data buffer

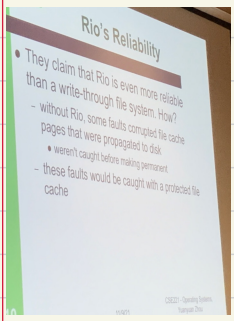


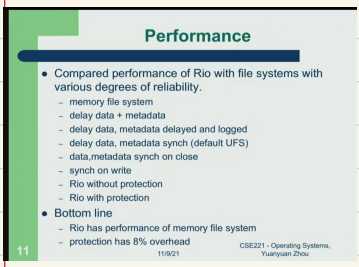


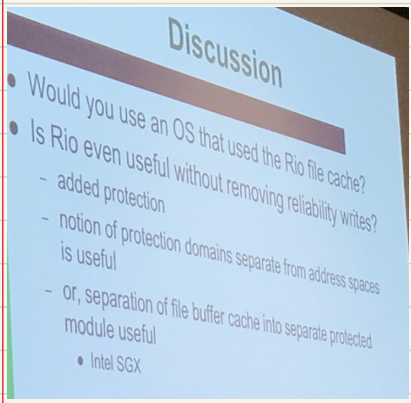


write and get blocked

→ write persist → check failed







can be used for security protection.