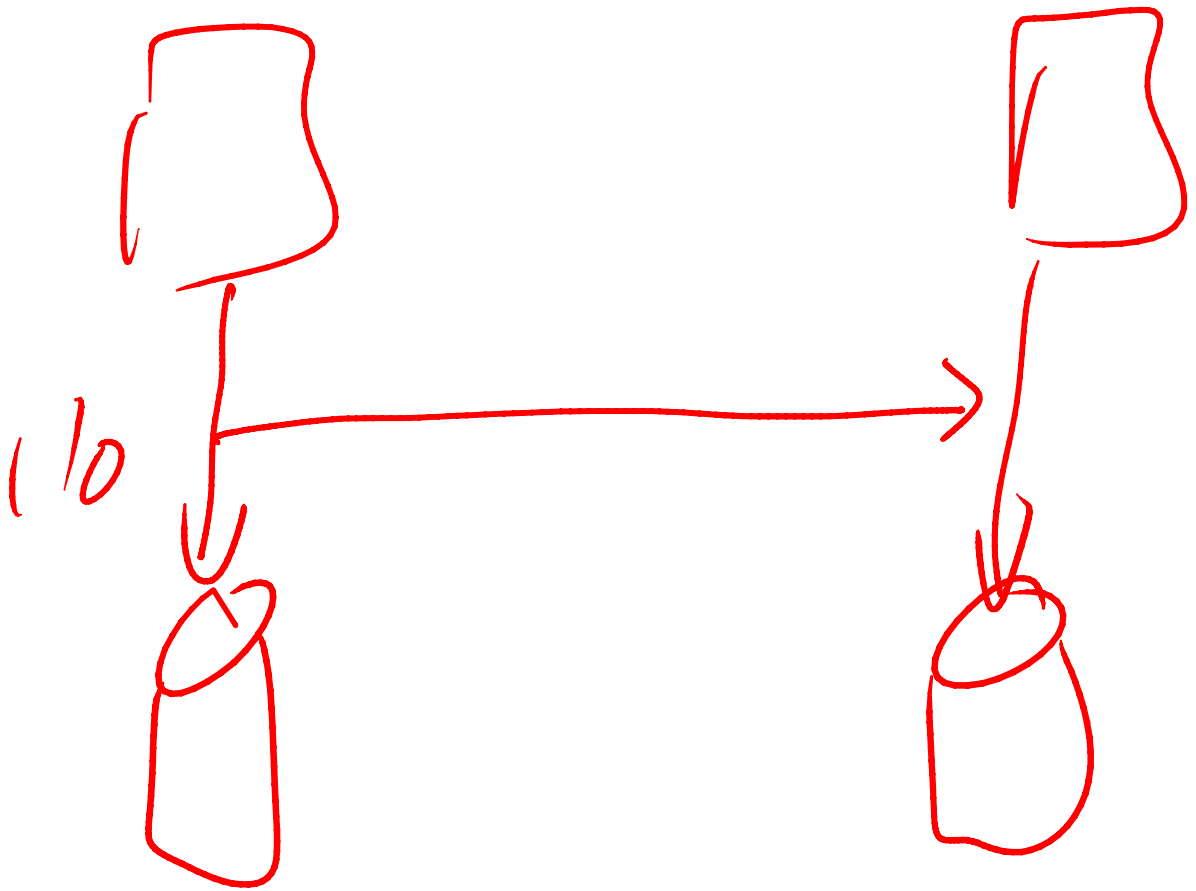
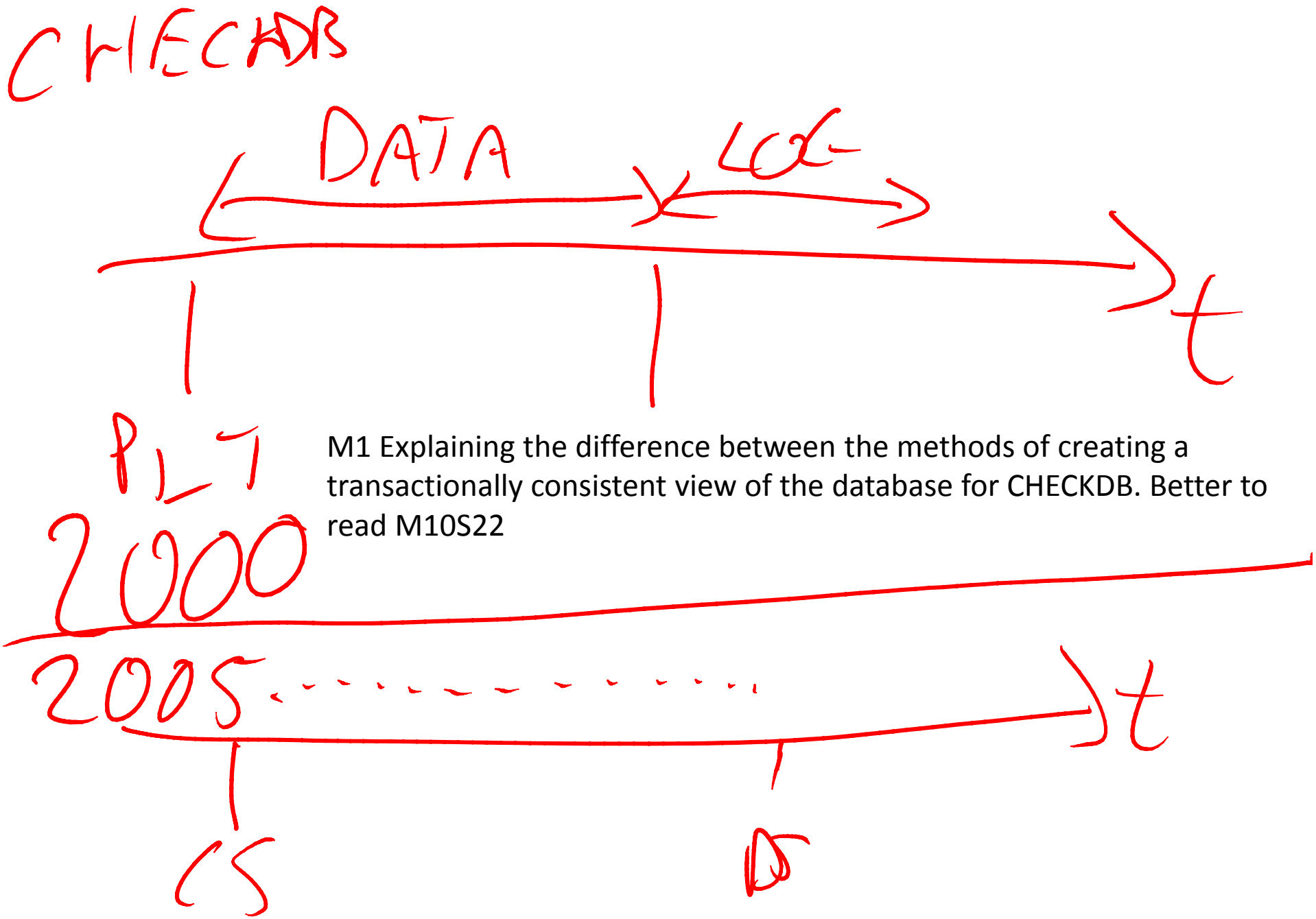
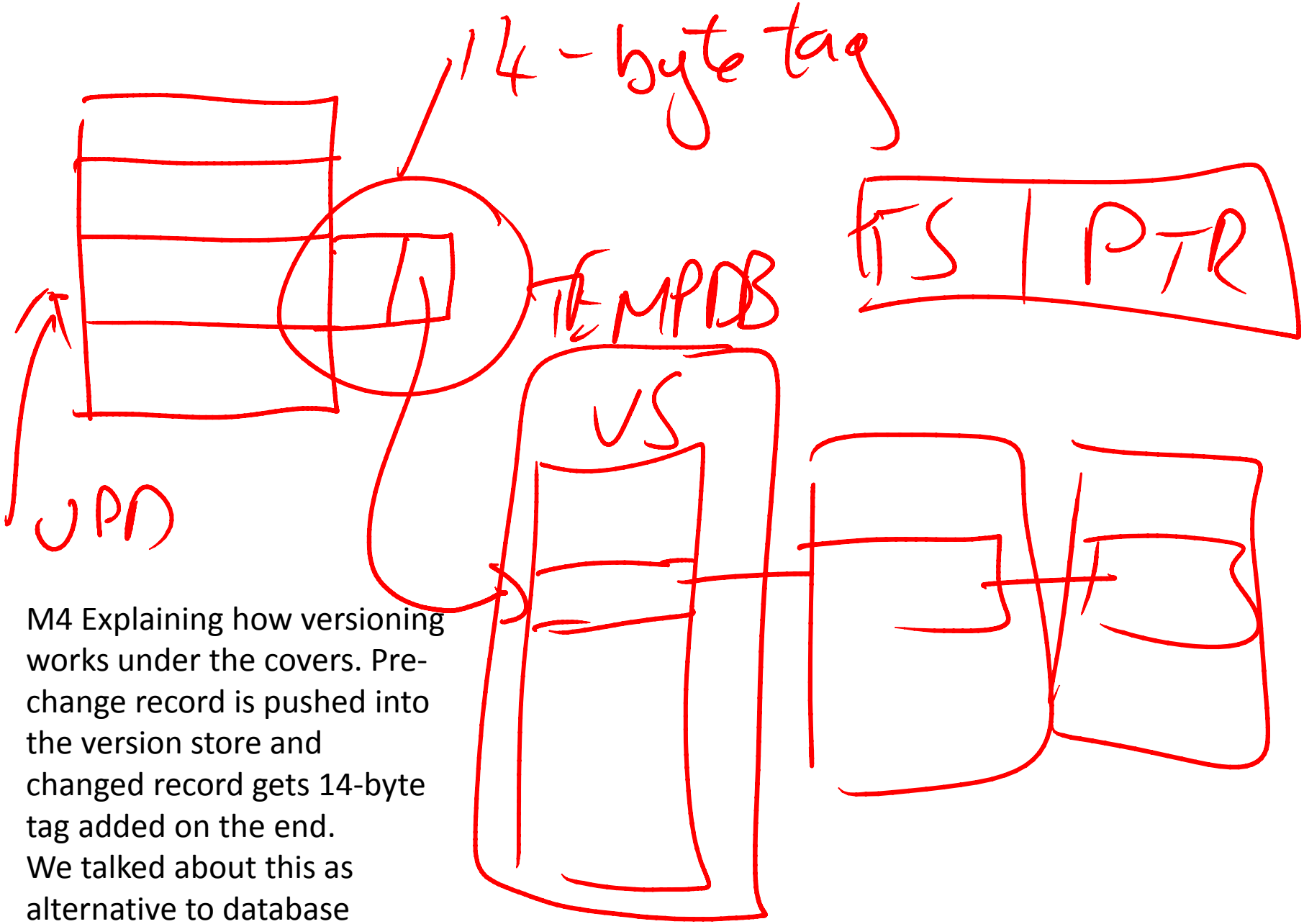


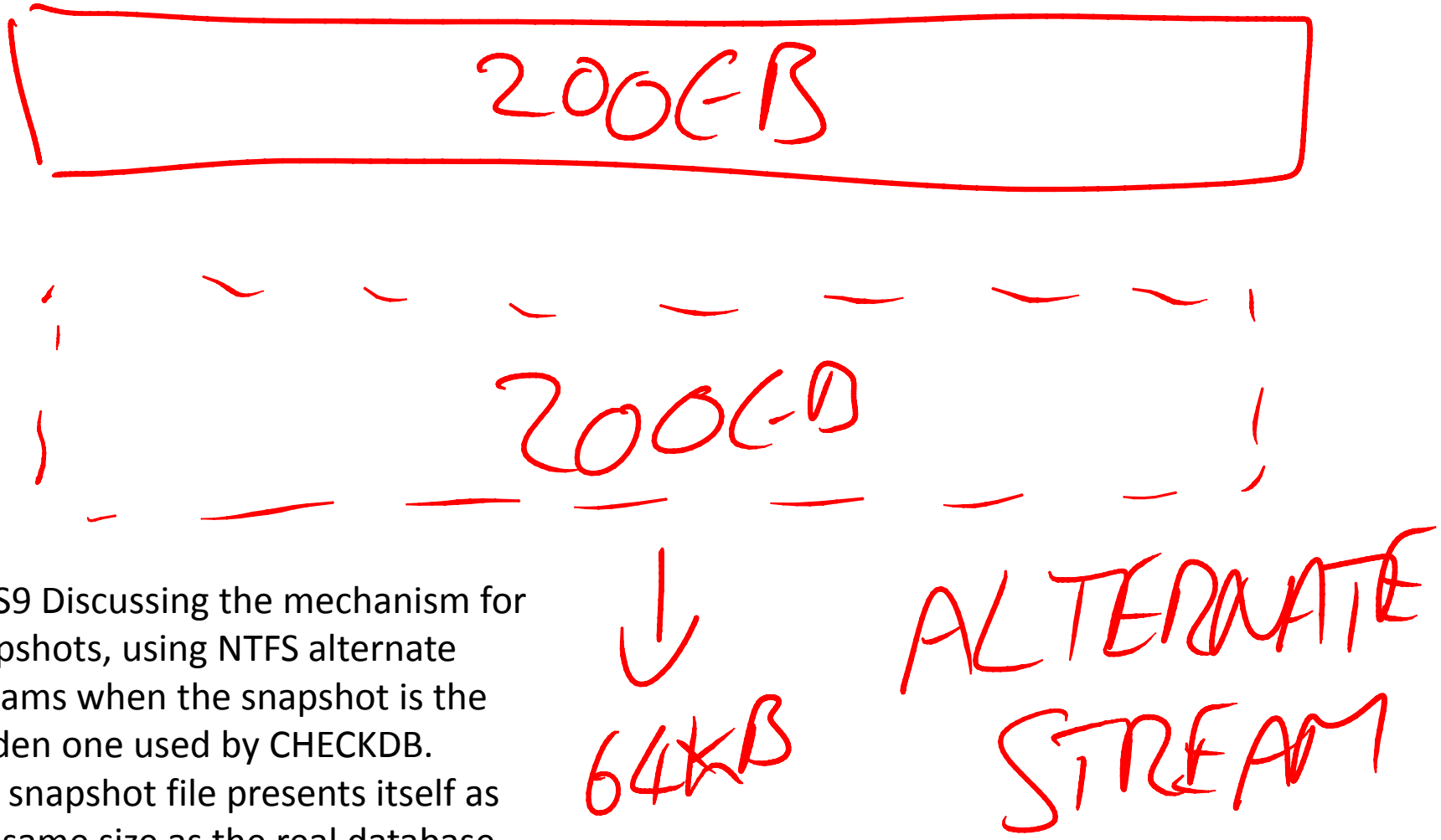
M1 Explaining that I/O corruption doesn’t necessarily get sent to the secondary side of a mirrored SAN if the corruption occurs after the point at which the I/O is sent to the mirror.



M1 Explaining the difference between the methods of creating a transactionally consistent view of the database for CHECKDB. Better to read M10S22



M4 Explaining how versioning works under the covers. Pre-change record is pushed into the version store and changed record gets 14-byte tag added on the end.
We talked about this as alternative to database snapshots

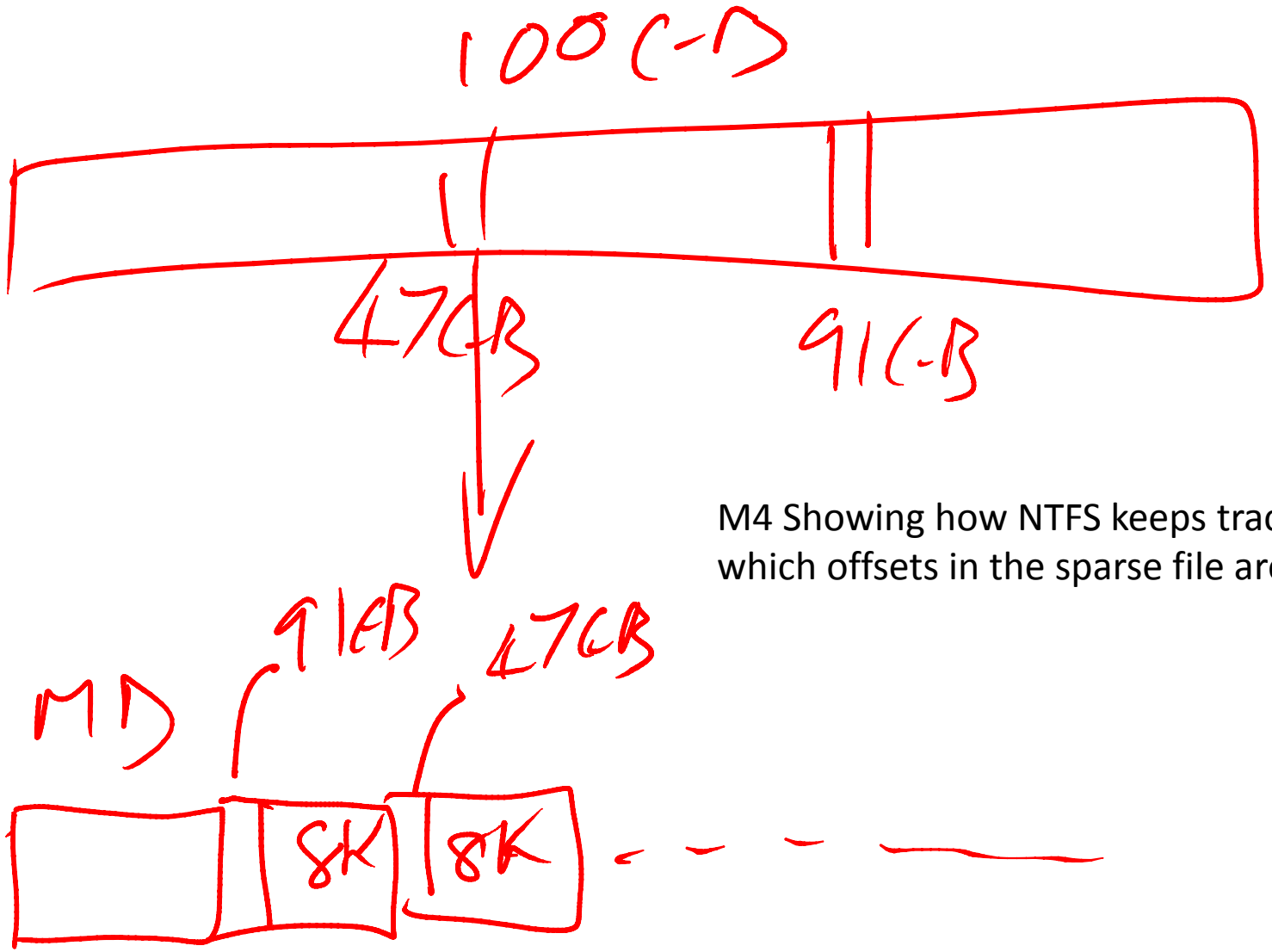


M4S9 Discussing the mechanism for snapshots, using NTFS alternate streams when the snapshot is the hidden one used by CHECKDB. The snapshot file presents itself as the same size as the real database size, but the size on disk is very small.

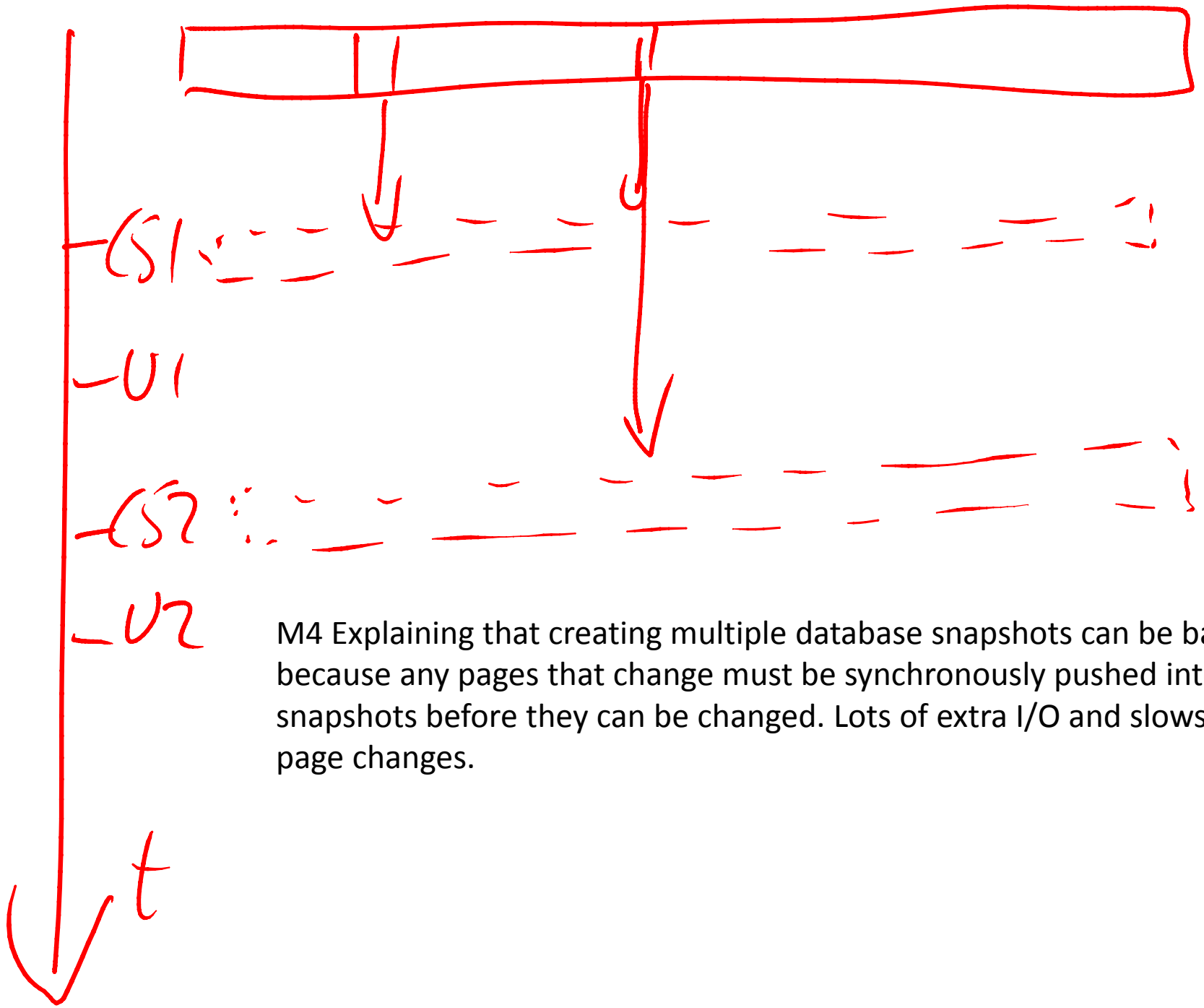
a int[10000000]

a[43113] = 10

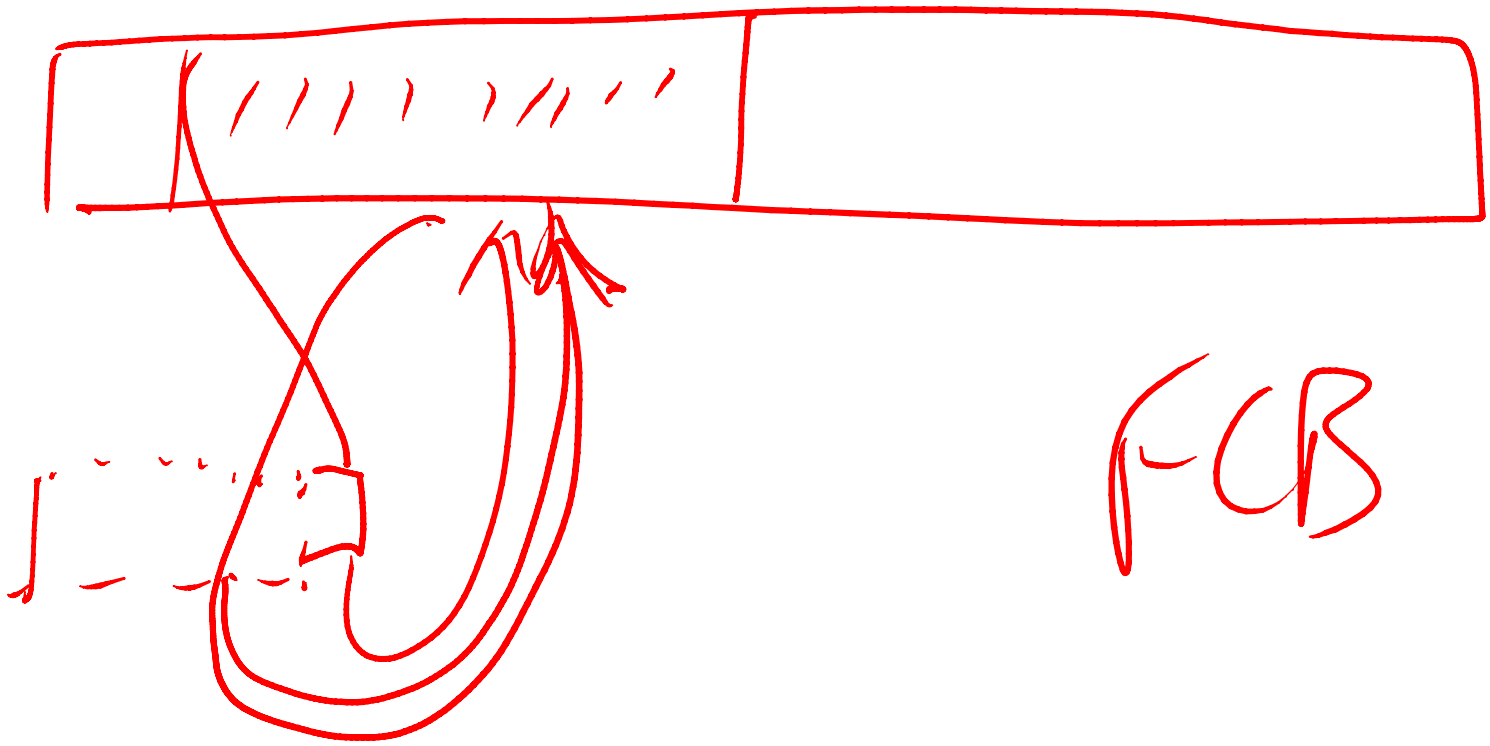
M4 Drawing an analogy between sparse files in NTFS and sparse arrays in various programming languages. You can define an array of arbitrary size but only the populated elements take up memory.



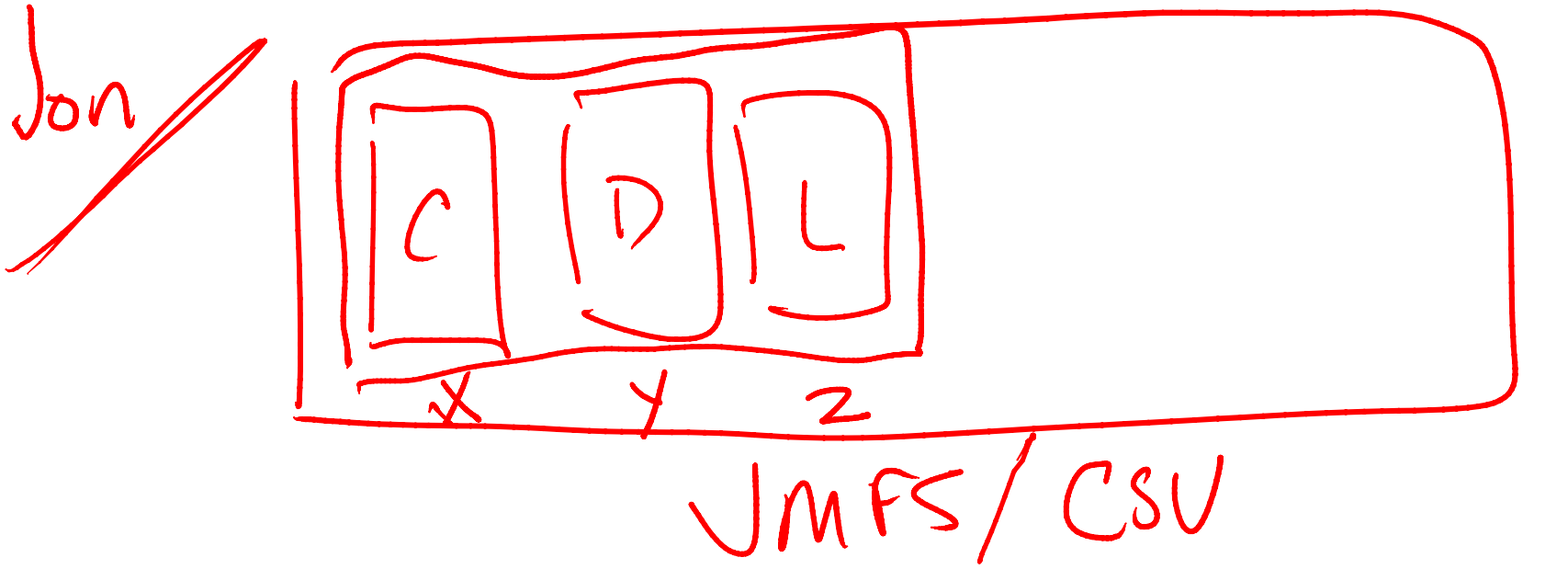
M4 Showing how NTFS keeps track of which offsets in the sparse file are valid.



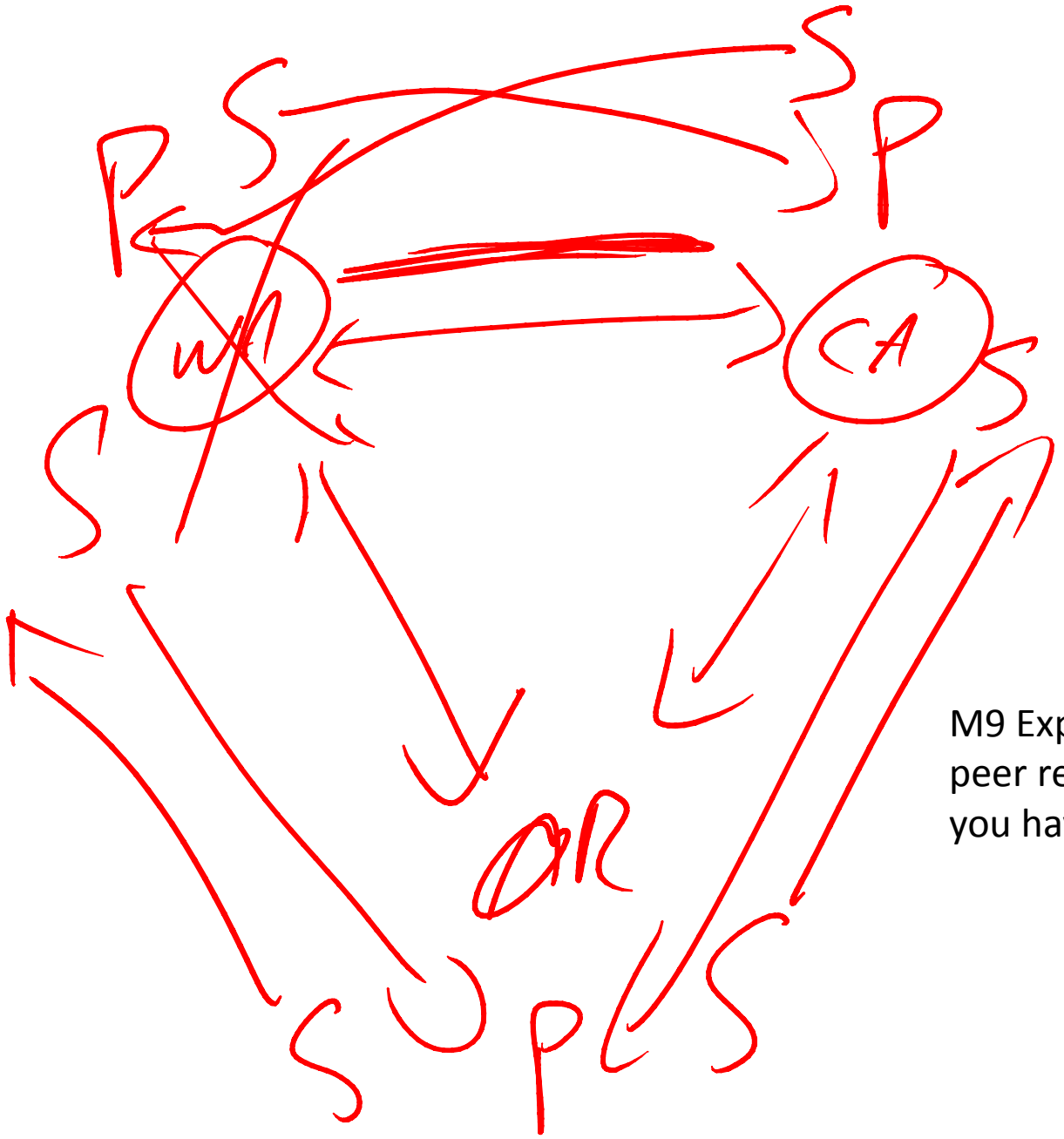
M4 Explaining that creating multiple database snapshots can be bad because any pages that change must be synchronously pushed into all snapshots before they can be changed. Lots of extra I/O and slows down page changes.



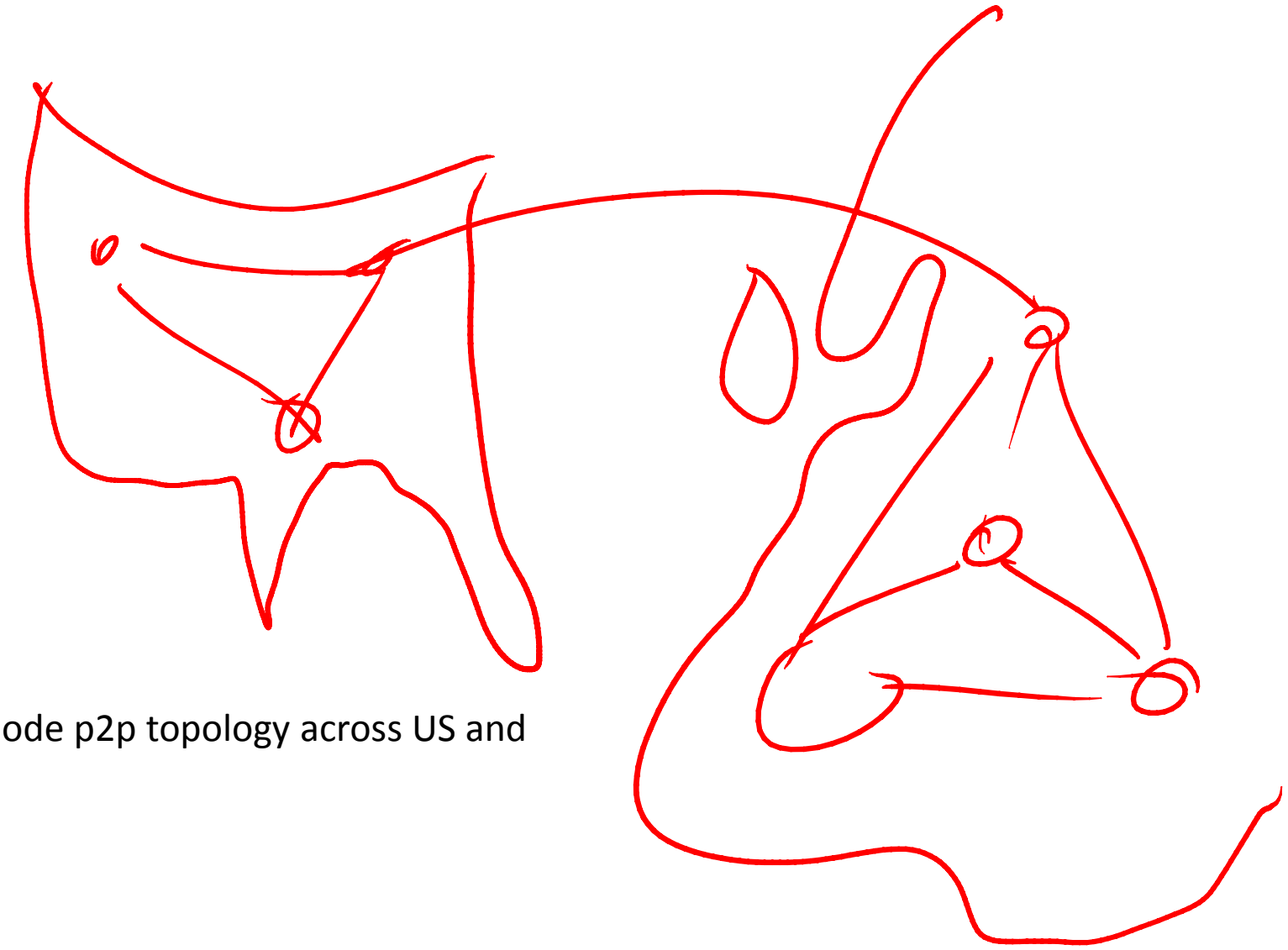
M4 Explaining the demo on recovering data from a snapshot. After dropping the table, the snapshot doesn't grow because the pages from the dropped table haven't been changed yet. As we repopulate the newly-created table though, it's pulling unchanged pages from the source database through the context of the snapshot to populate the new table, thus changing pages and pushing them into the snapshot. A bit of a mind-bender.



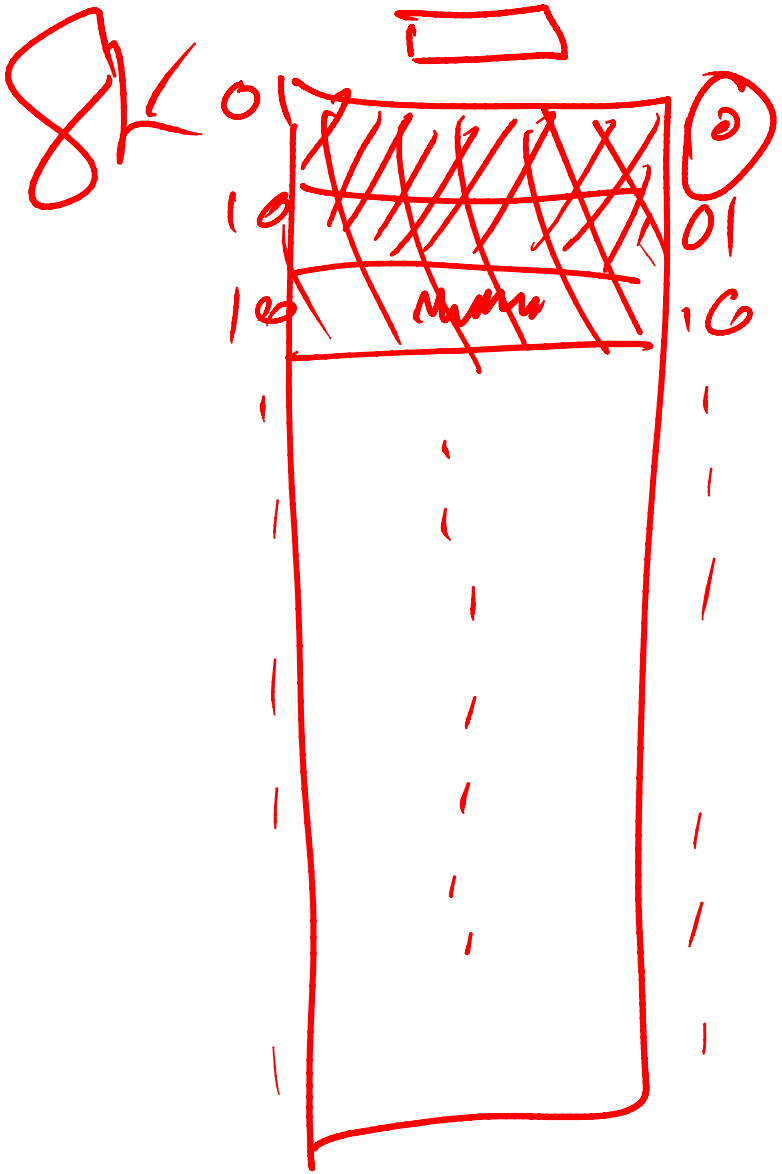
M4 Jon explaining how 3rd-party backup solutions that use VSS snapshots will issue DBCC commands to freeze all I/O on a database before doing the backup – making sure that the backup is consistent.



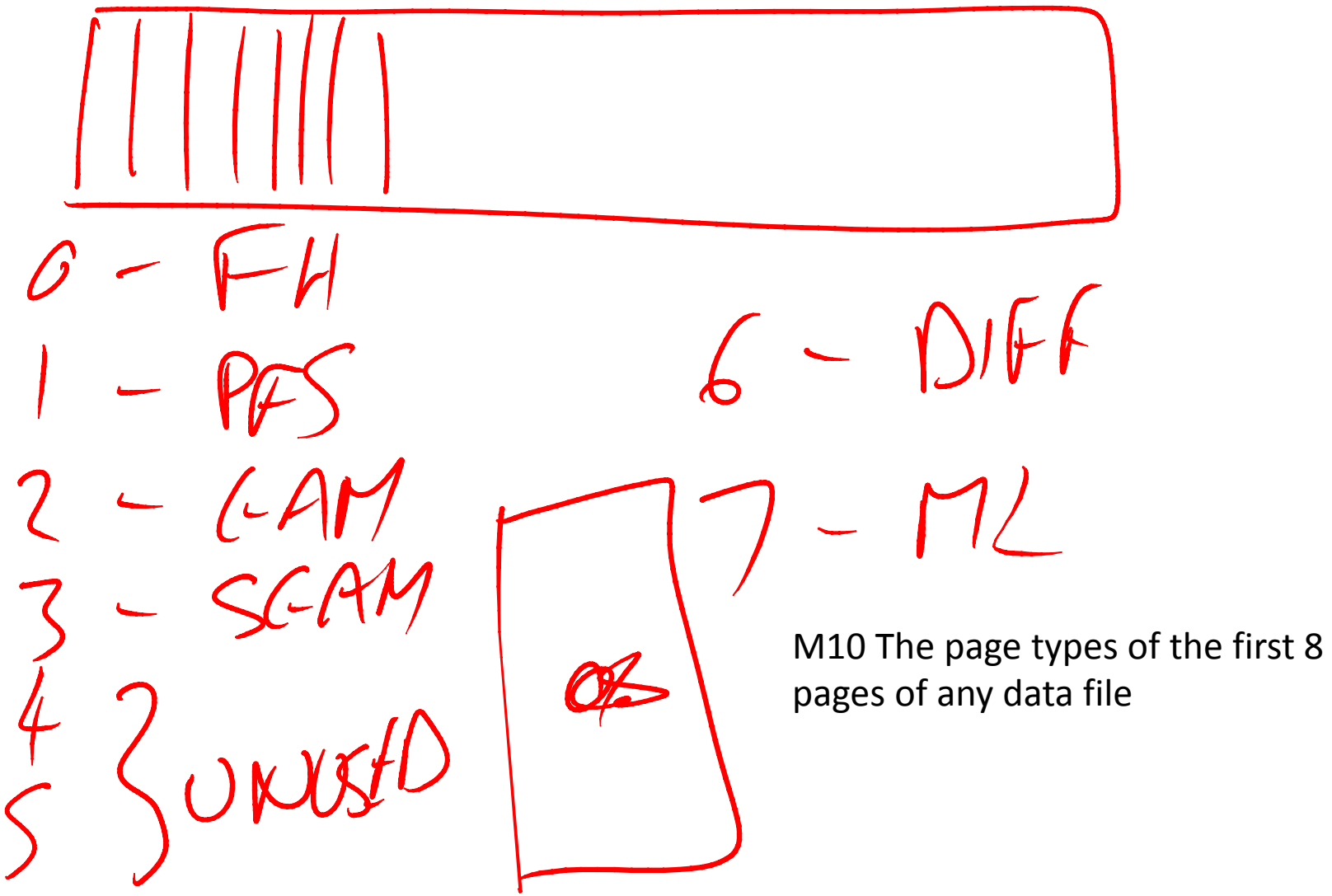
M9 Explaining the peer-to-peer replication demo that you have on your DVD



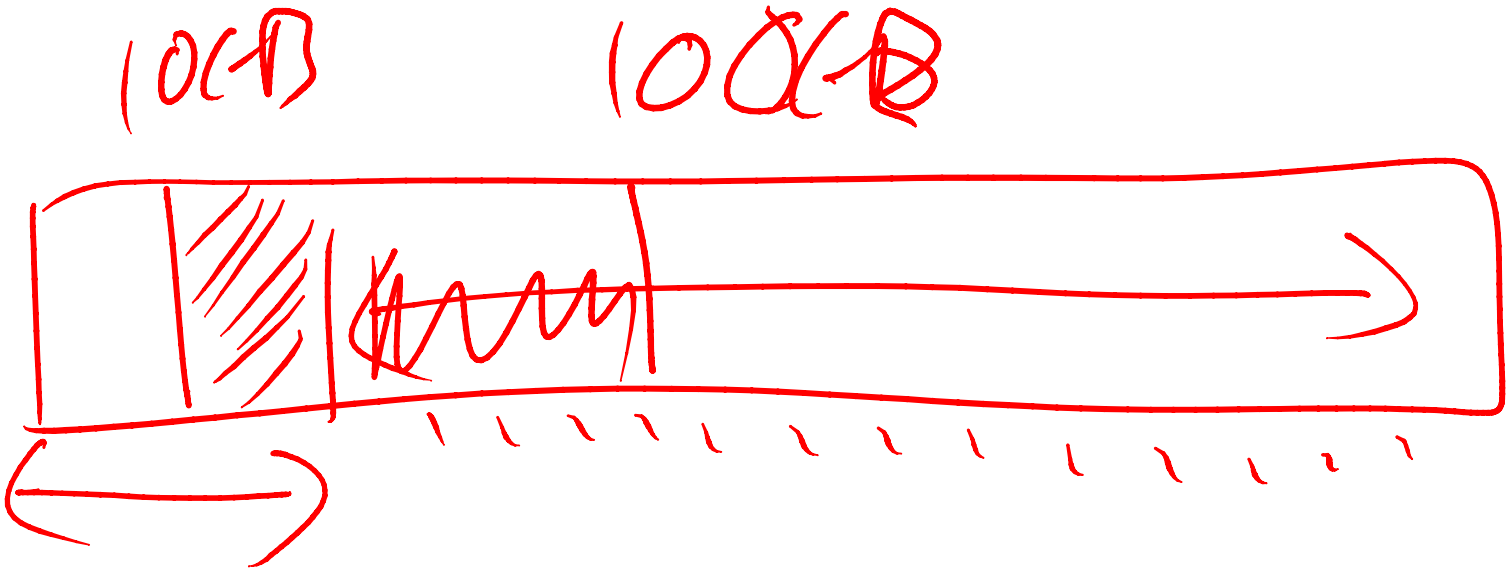
M9 14-node p2p topology across US and Europe.



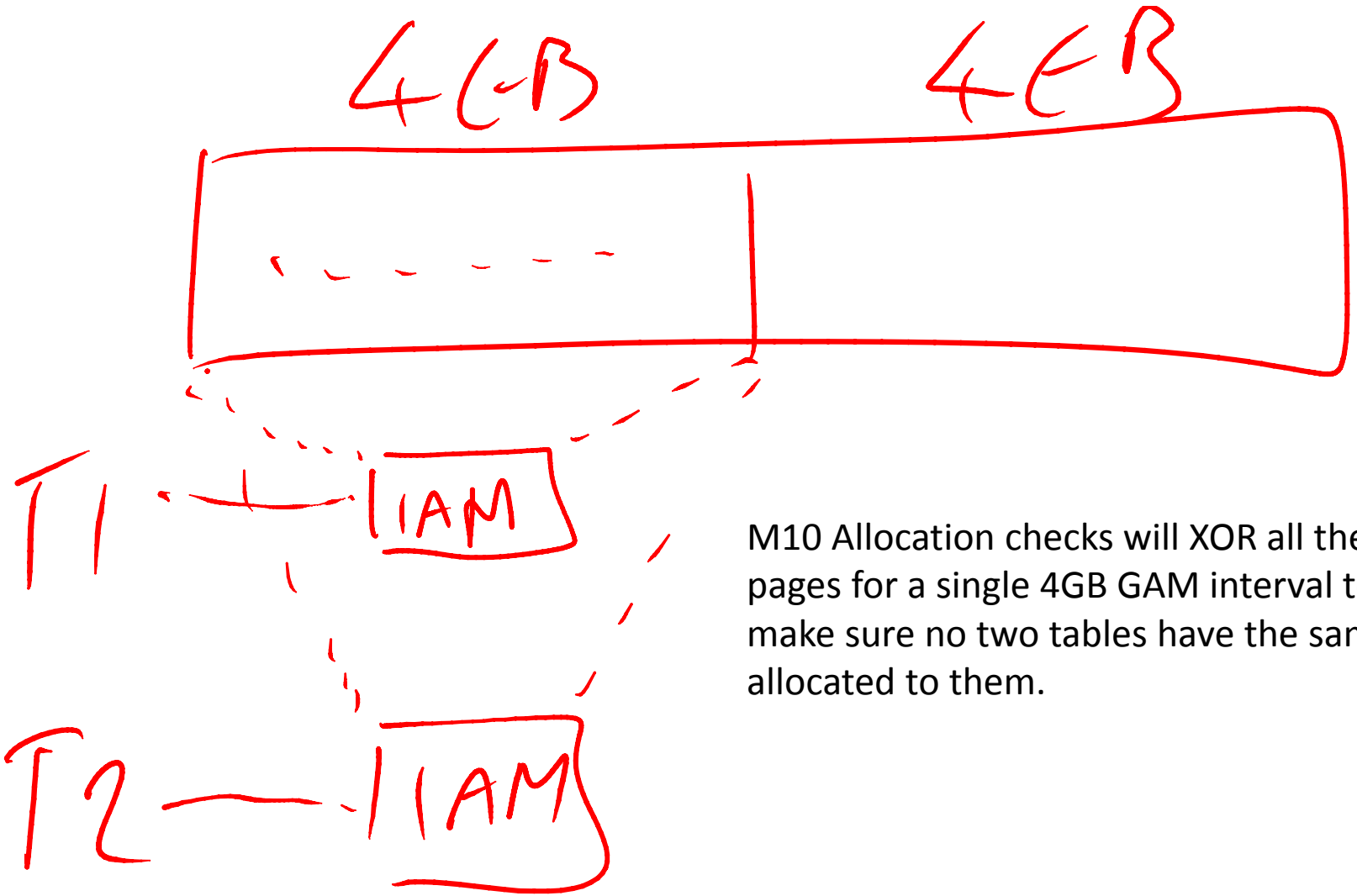
M10S11 Explaining torn-page detection



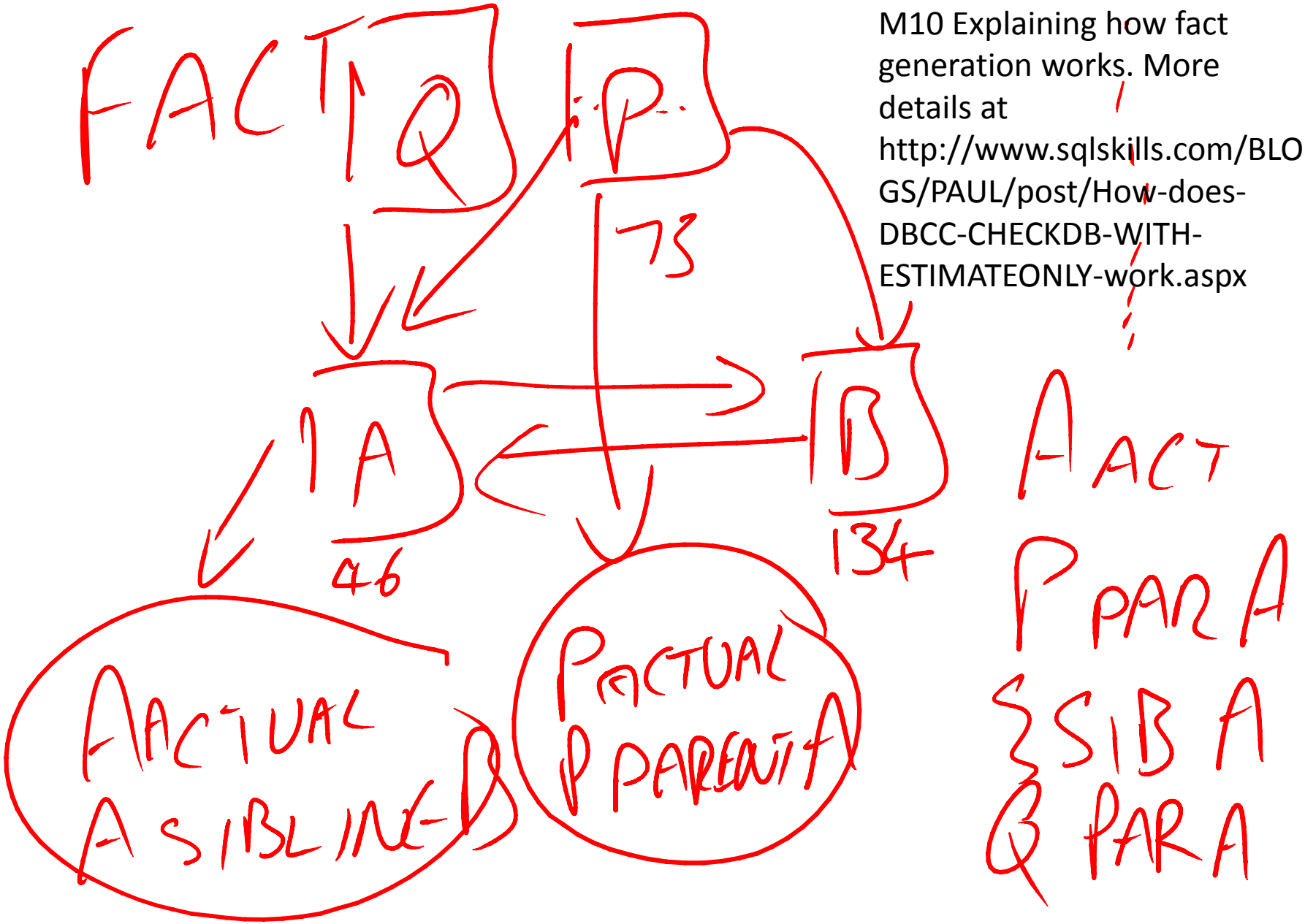
M10 The page types of the first 8 pages of any data file



M10 How a database of 100GB with only 10GB of data in will have allocation bitmaps spread through it every 4GB (GAM, SGAM, DIFF, ML pages) and every 64MB (PFS pages)



M10 Allocation checks will XOR all the IAM pages for a single 4GB GAM interval together to make sure no two tables have the same extents allocated to them.



M10 Explaining how fact generation works. More details at http://www.sqlskills.com/BLOGS/PAUL/post/How-does-DBCC-CHECKDB-WITH-ESTIMATEONLY-work.aspx

M10 Explaining the lazywriter background process in the buffer pool is a clock-hand sweeping through all the buffers in memory and implementing a least-recently-used algorithm