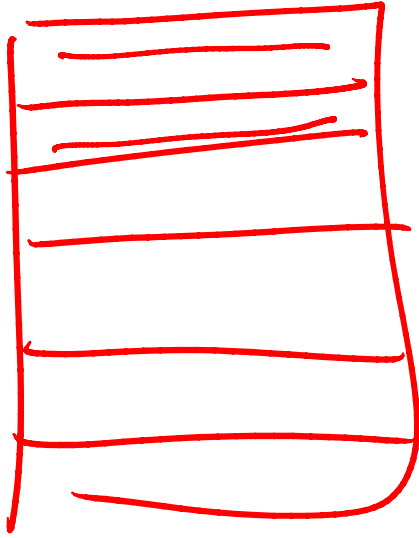
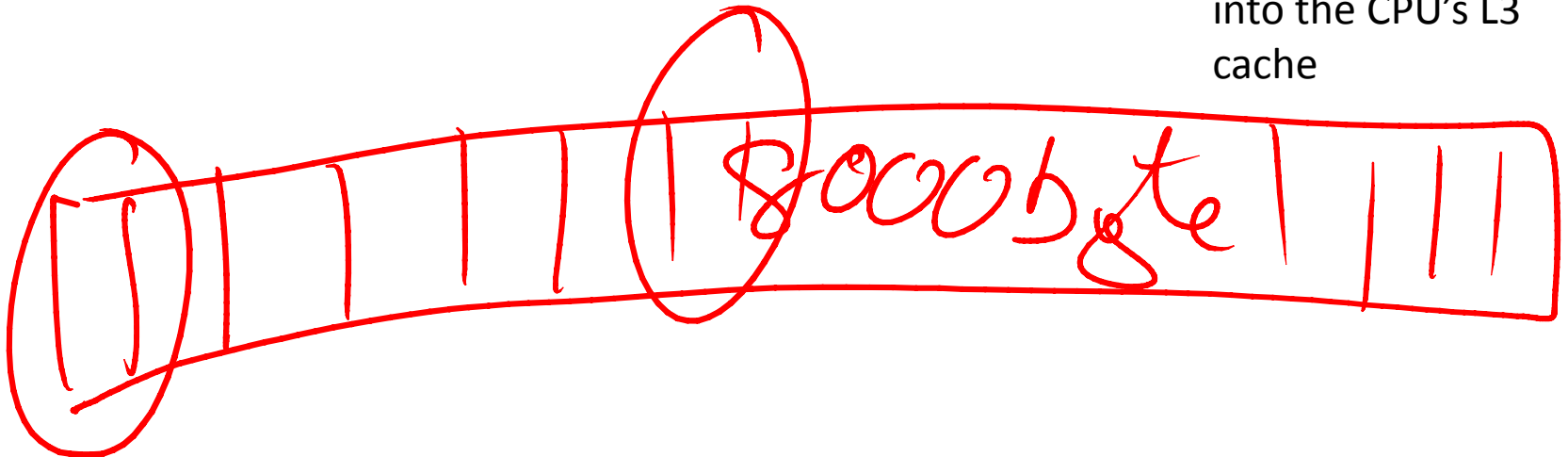


CACHE LINE INVALIDATION

CACHE LINES
64b / 128b



M1S06 Discussing
why the NULL
bitmap is a perf
optimization by
avoiding having to
read record parts
into the CPU's L3
cache



c1, c2... c10



HEAP
CL

SEL c1, c6, 8

W. c2 = x

PTR

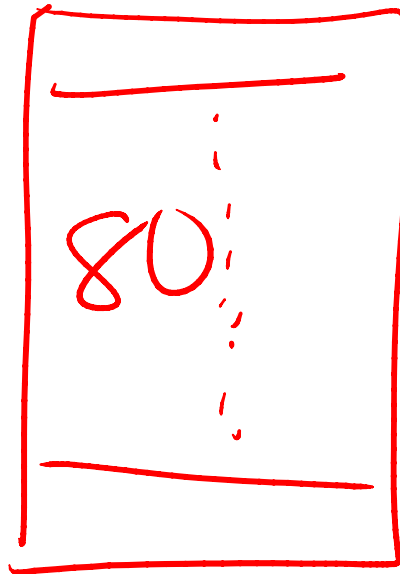
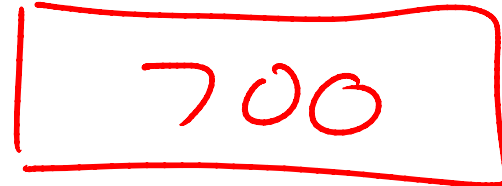
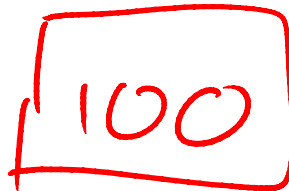
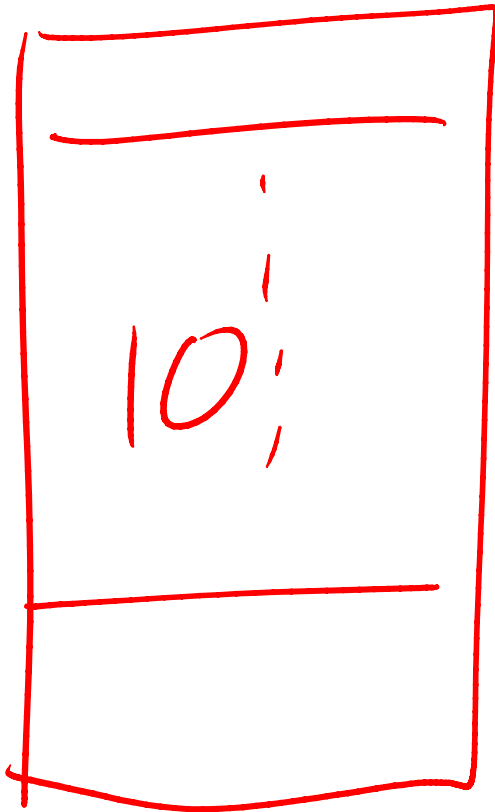
PHYSICAL RID

LOGICAL



M1S09 Explaining NC index back-links to the data records as part of describing why forwarding/forwarded records exist. Without them, moving a heap row would entail changing all NC index records with a back-link to that row.

(FILE : PAGE : SLOT)
(CL KEY(S))



M1S12 Having a large LOB value in row that's not used much makes for large rows and low data density. Choose how to store LOB data wisely.

100	700
-----	-----

100	10
-----	----

700	10
-----	----

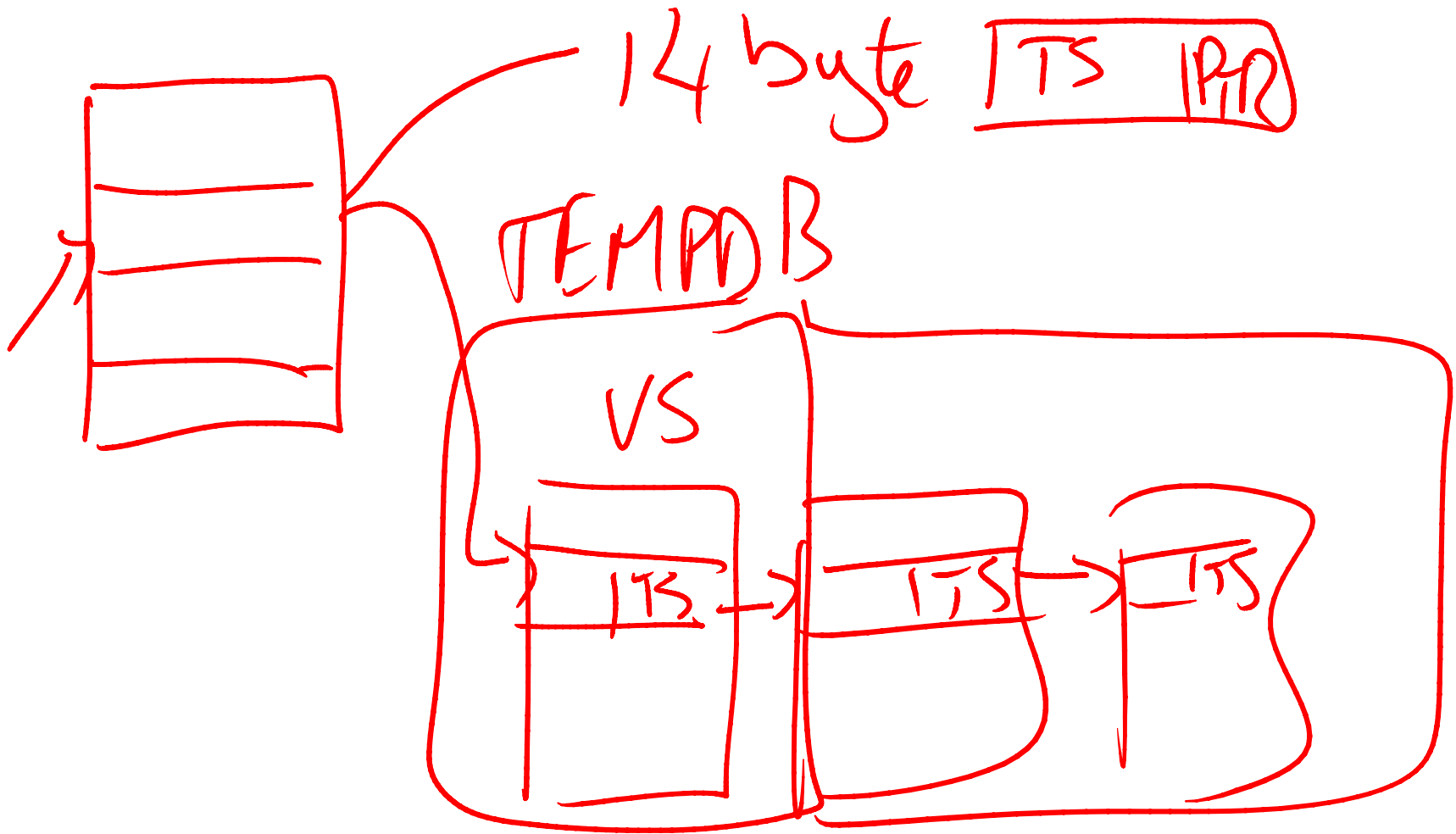
M1S12 The 700 byte LOB could be stored in a separate table and then JOINed in when needed...

100	24
-----	----

PTIR
LEANS
TS

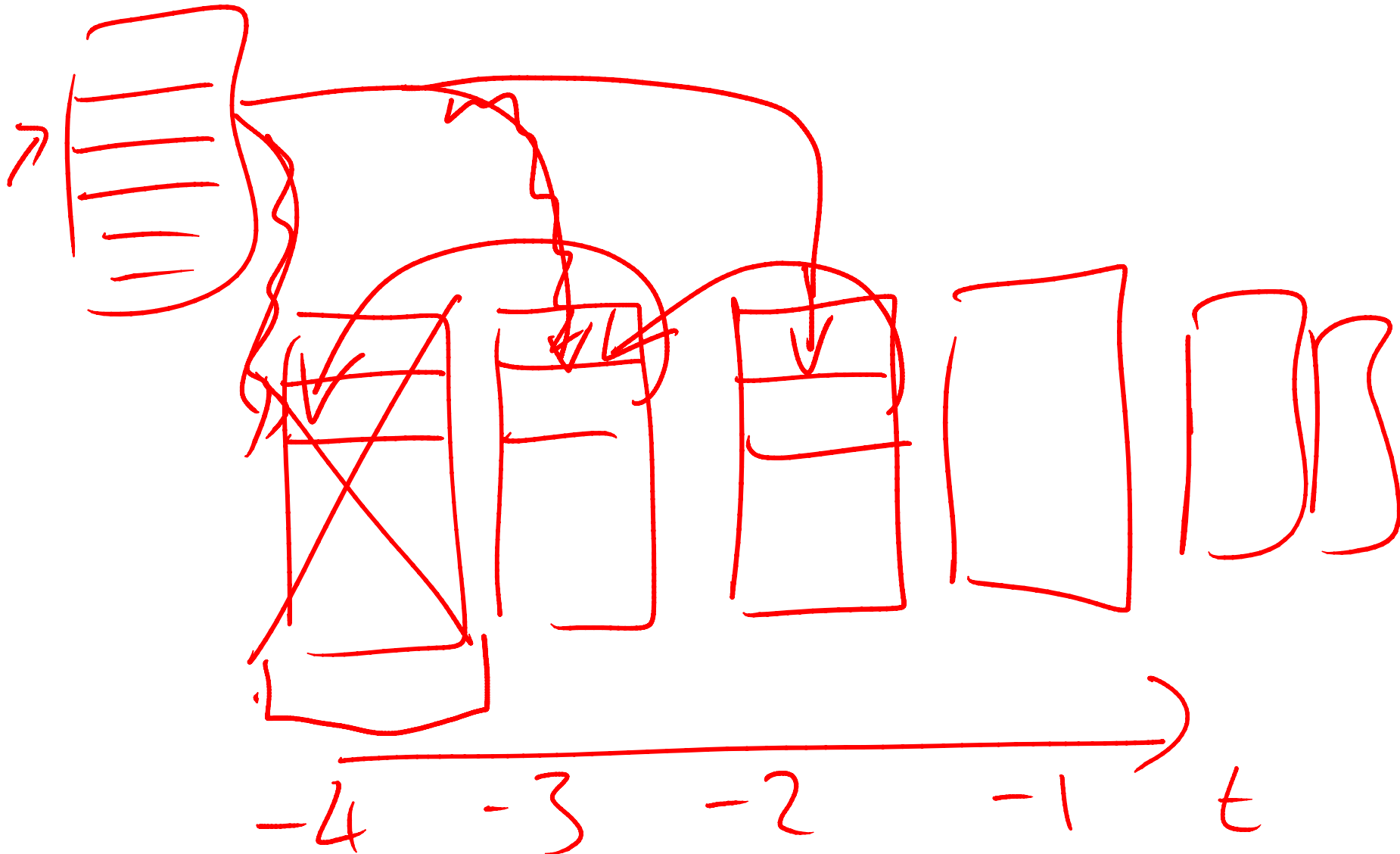
TEXT
700

M1S12 ... or it could be stored off-row in the existing table



M1S14 When an update occurs to a record and snapshot isolation is enabled, the pre-change record is copied into the version store in tempdb. The changed record has a 14-byte tag on the end with the timestamp and pointer to previous record version. A chain of previous versions is possible.

M1S14 & M2S35 Explaining how there's a new portion of the version store created every minute.



~~DATABASE~~ > PERFMON
TRANSACTIONS

V.S. SIZE

V.S. GEN. RATE

V.S. CLEANUP RATE

sys.dm_db_file_space_usage

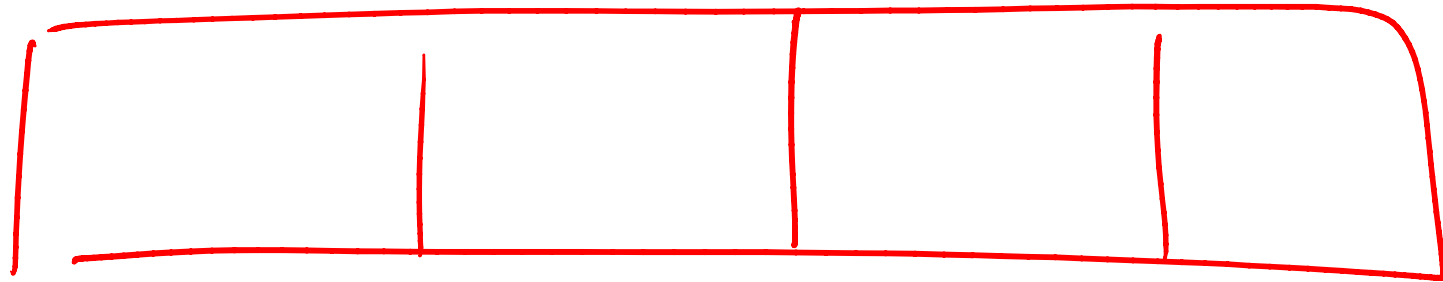
M1S14 Monitoring the version store using the Transactions perfmon object: Version Store Size, Generation Rate, Cleanup Rate. And using the sys.dm_db_file_space_usage DMV.

TF 662

DBCC TRACEON(662, -1)

3605

M1S15 When enabling TF 662 to watch the ghost cleanup task, you need to use , -1 in the DBCC TRACESTATUS to make the flag enabled in all connections. You may need to also enable TF 3605 as well.



0 — 7 8 — 15 16 — 23

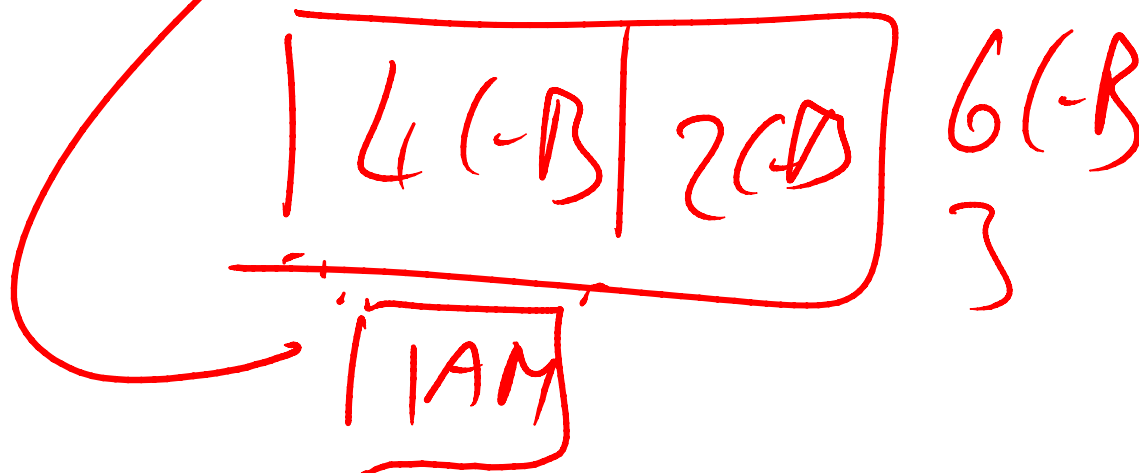


M1S24 Explaining the difference between mixed and dedicated extents. Mixed extent is one in which the 8 pages could be allocated to 8 different objects. Dedicated extent is one where all 8 pages are reserved for exclusive use of a single object.

Later in the deck I refine 'object' to really be 'allocation unit'.



M1S36 Example of IAM chains.
 Allocations from each 4GB
 chunk of a data file require an
 IAM page to track the
 allocation. Multiple IAM pages
 make an IAM chain.



System tables.

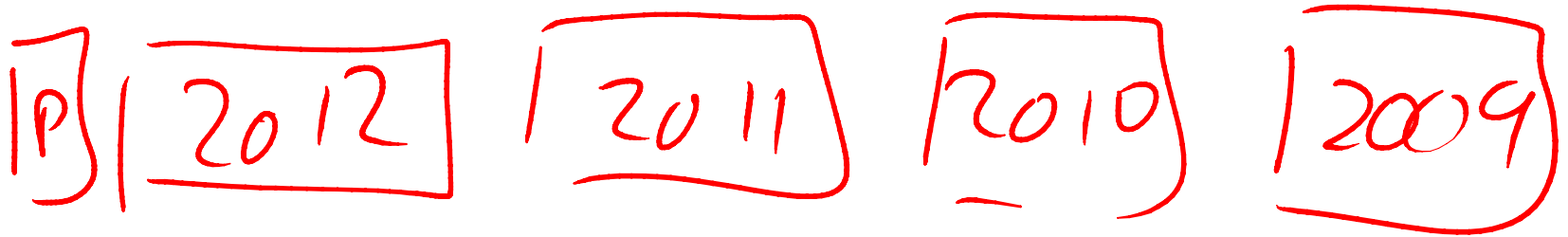
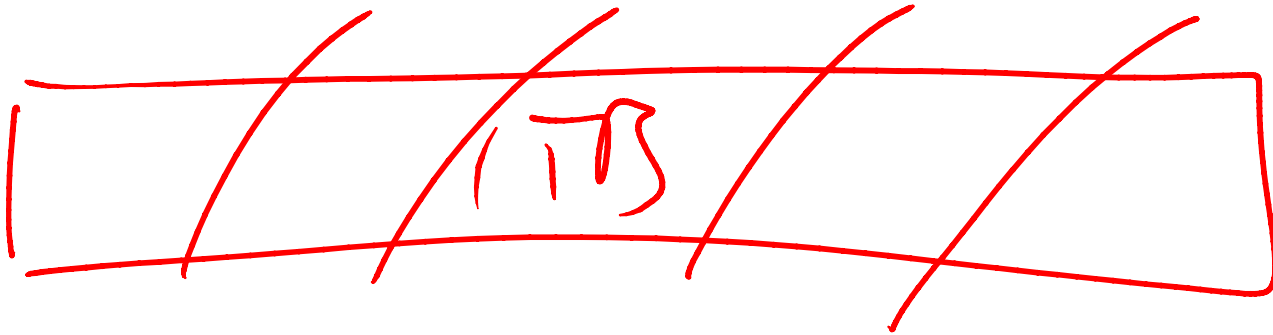
M1 This was a reminder for me to put in info about the system tables.

Sysindexes -> sys.sysidxstats

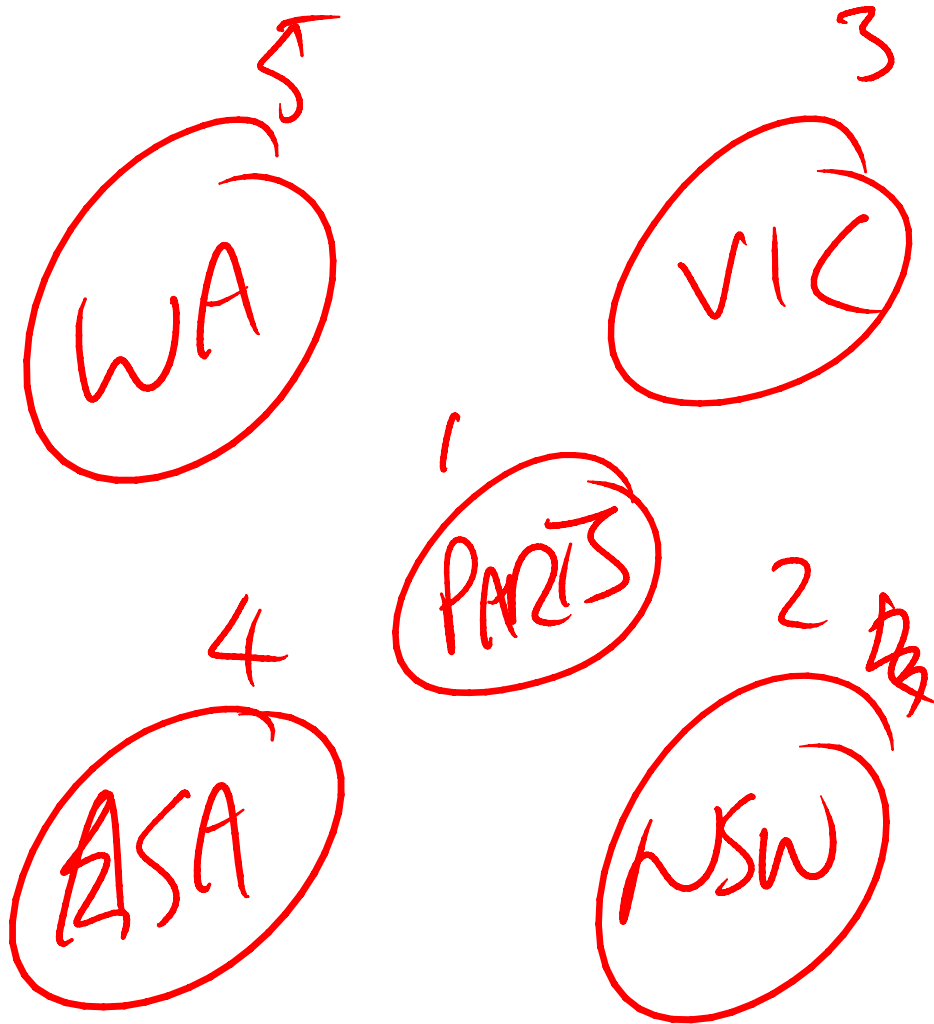
Syscolumns -> sys.syscolpars

Sysobjects -> sys.sysschobjs

Storage Engine metadata tables: sys.sysallocunits, sys.sysrowsets, sys.sysrscols (and they were different in 2005)

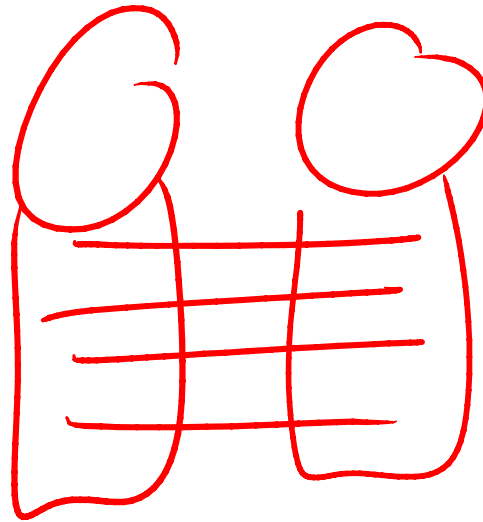


M2S7 Splitting a large database up into multiple filegroups can enable you to do small, targeted restores or even a partial restore from scratch – in the event that the database is damaged or destroyed.



M2S7 Explaining some ways to split up a database to allow targeted restore. On the left, auto parts sales system – makes sense to bring parts and NSW online first.

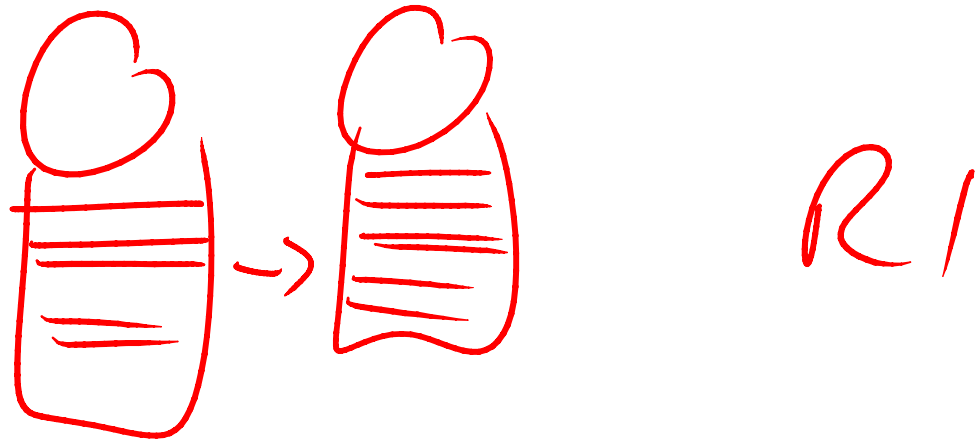
STRIPING



RAID 0

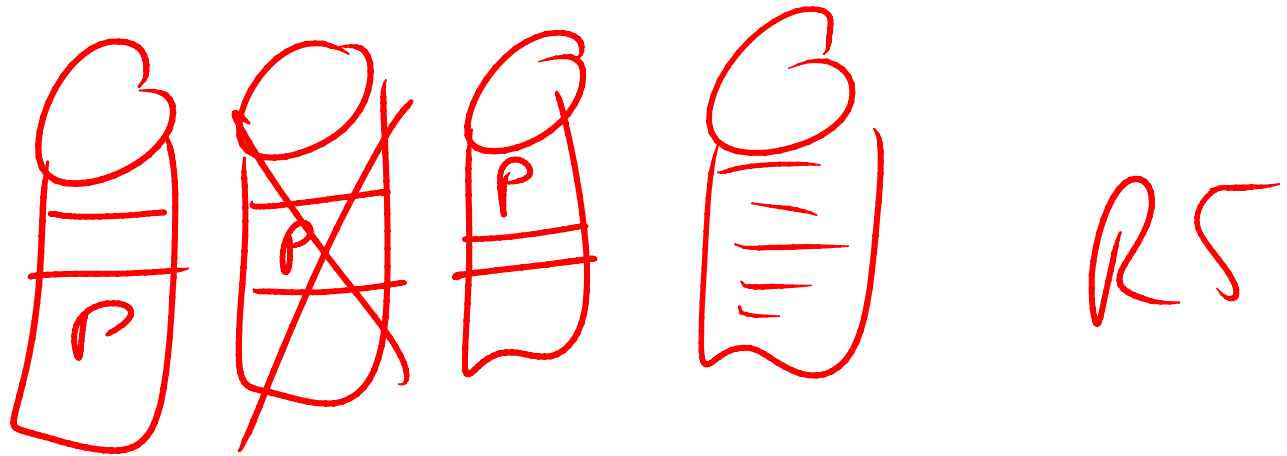
http://en.wikipedia.org/wiki/Standard_RAID_levels

MIRRORING



http://en.wikipedia.org/wiki/Standard_RAID_levels

NOTATING PARITY

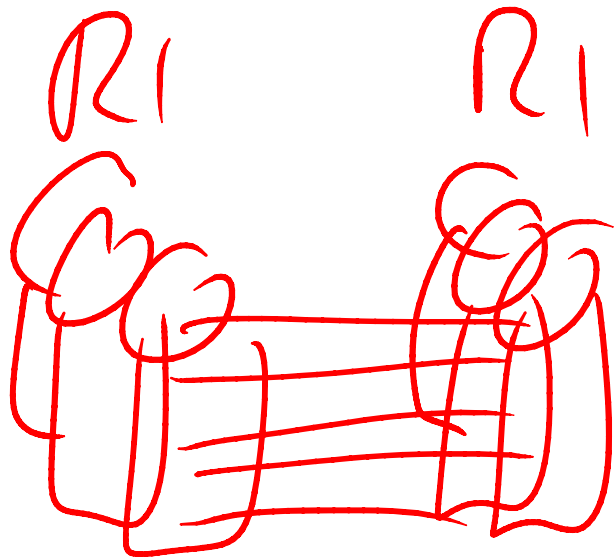


http://en.wikipedia.org/wiki/Standard_RAID_levels

$$R10 = R1 + G$$

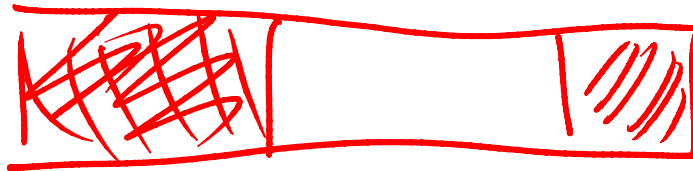
= MIRRORING

+ STRIPING



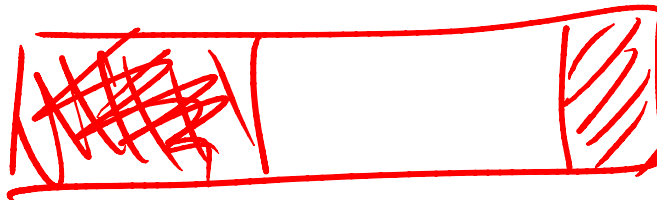
http://en.wikipedia.org/wiki/Standard_RAID_levels

RR

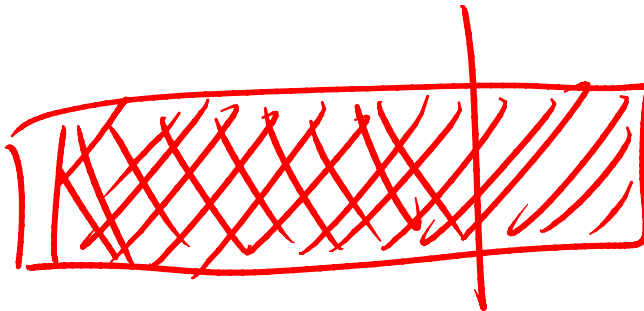


PROP FILL
WEIGHTING

~~8100~~ ←
SO (8099)



~~8100~~ ←
SO (8099)



1 ← ✓
8100

M2S19 Explaining round-robin
allocation and proportional fill

ALLOC. HOTSPOT

sys.master_files

sys.dm_io_virtual_file_stats

Physical Disk / Avg. Disk write Q
Length

M1 Looking for an allocation hotspot

SQL TRACE

FILE GROWTH EVENTS

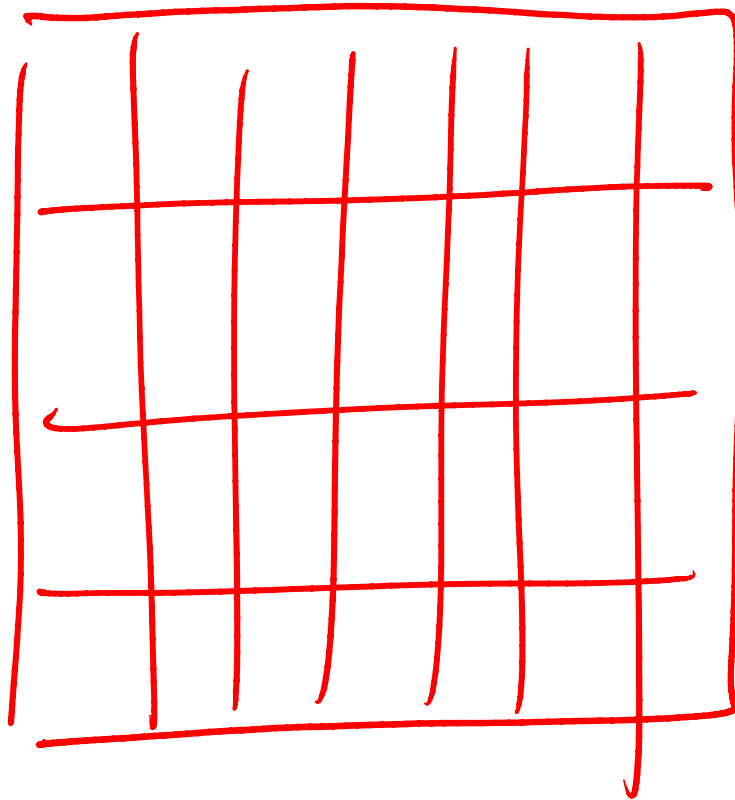
XEVENT

INSIDER

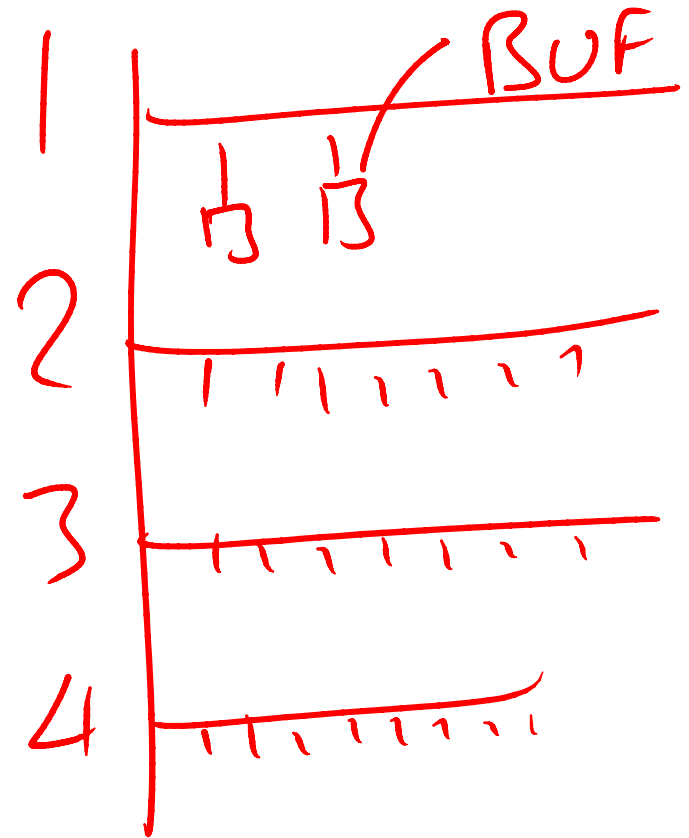
VIDEO

Database - file growth

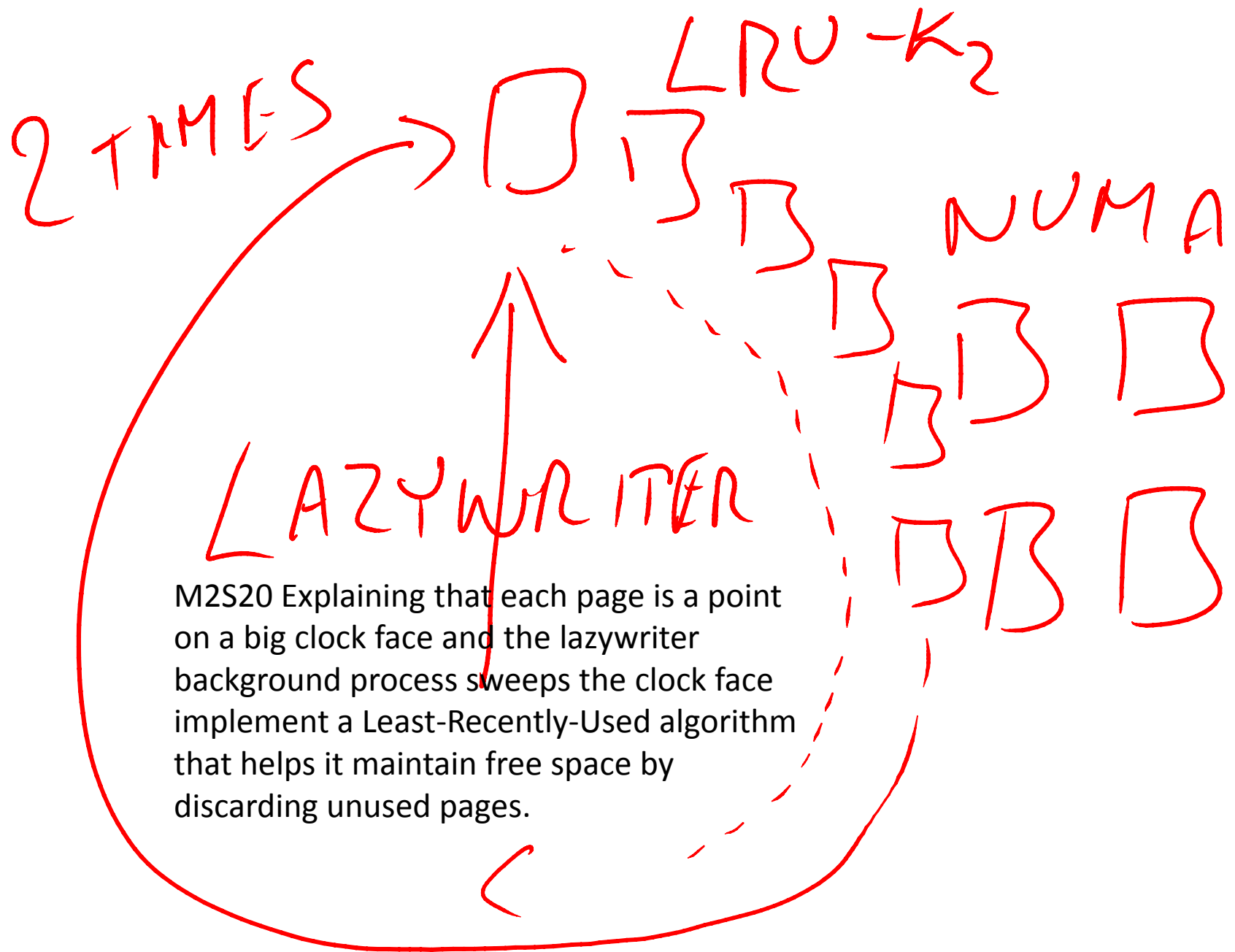
M1 Looking for file growths occurring. See demo video from
<http://www.sqlskills.com/insidercontent/201403/20140303newsletter.pdf>



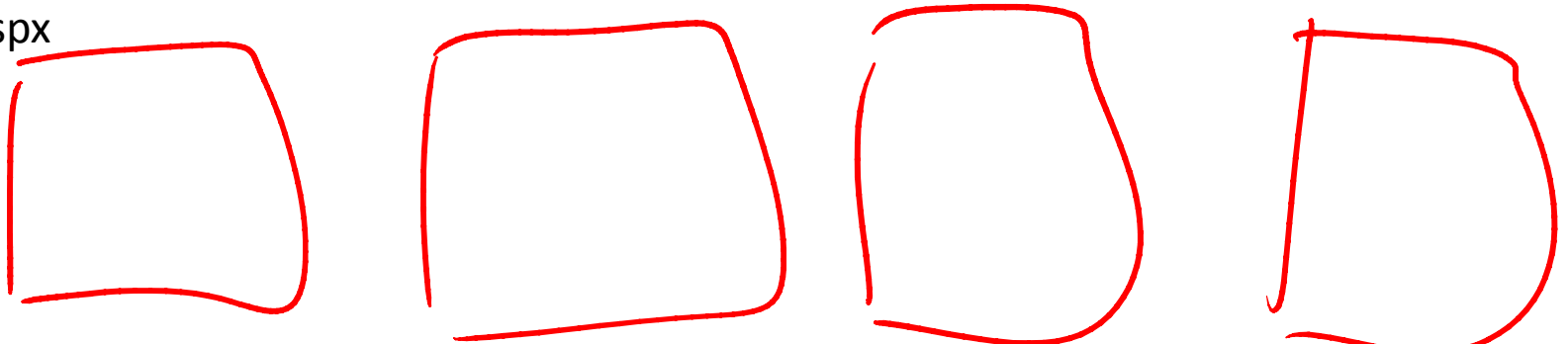
HASH LISTS



M2S20 Explaining the hash lists per database that allow the buffer pool to easily determine whether a page is already in memory or not



See <http://www.sqlskills.com/BLOGS/PAUL/post/Page-Life-Expectancy-isnt-what-you-think.aspx>



PLE
4000

PLE
4000

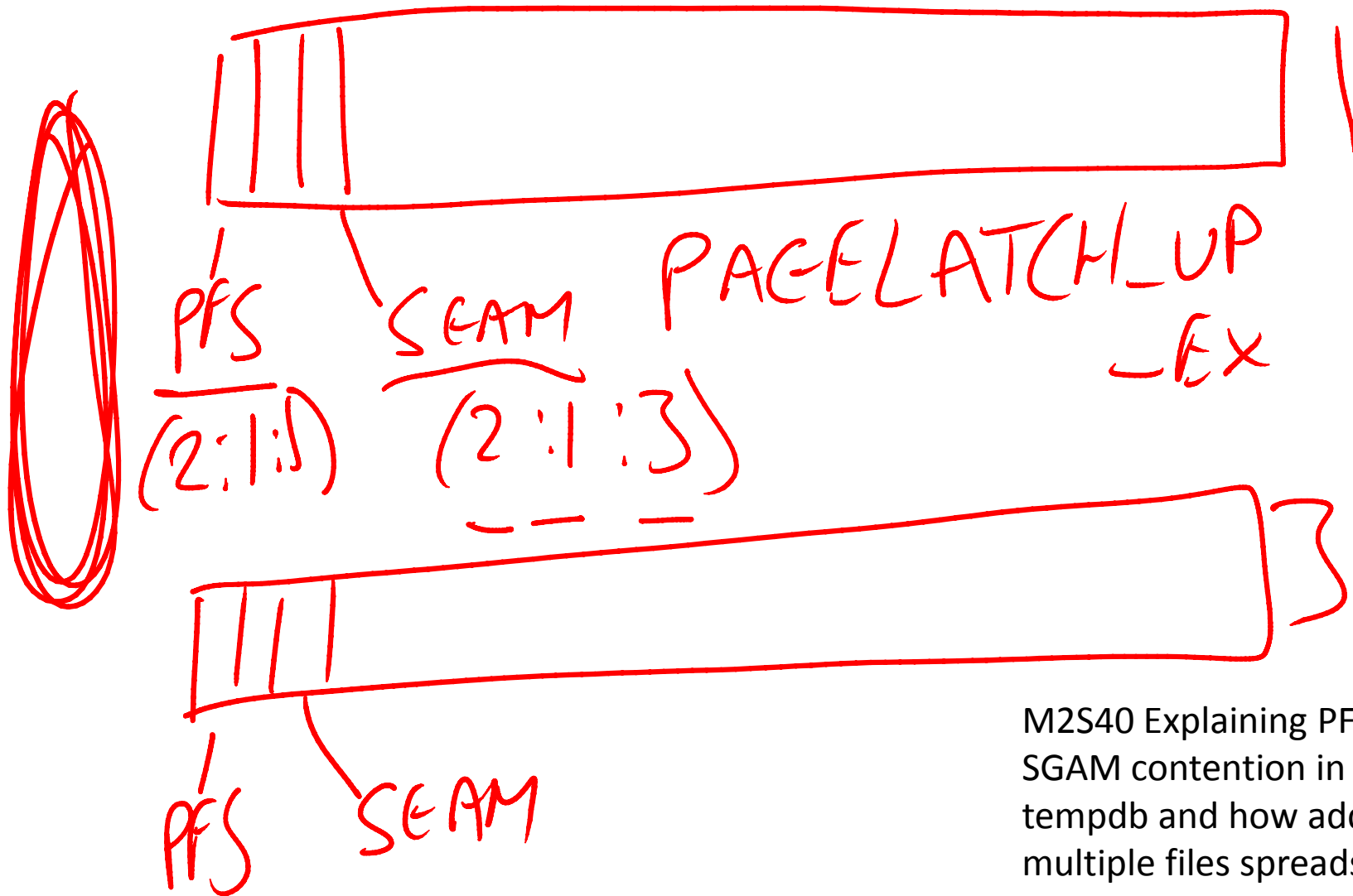
PLE
4000

PLE
4000

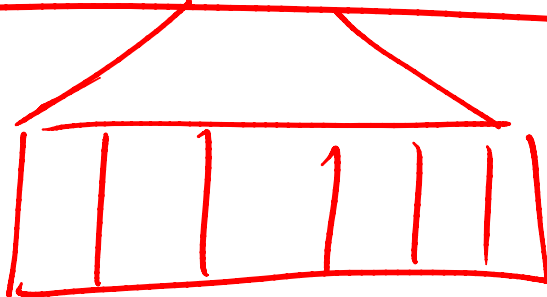
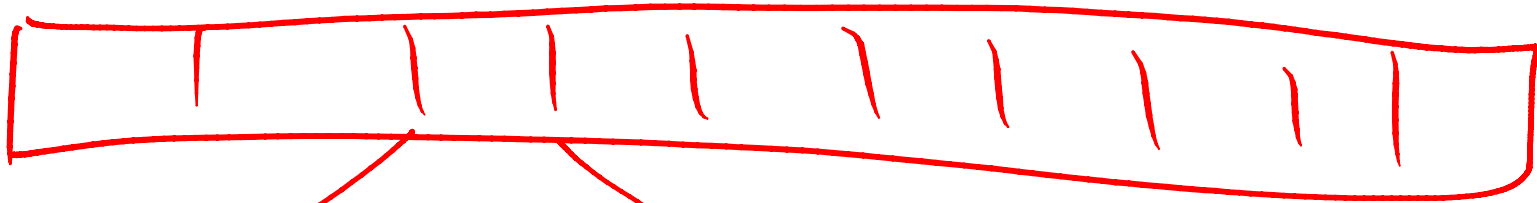
$$BM/PLE = (1 + 2 + 3 + 4) / 4 \downarrow$$
$$= 4000 \quad 1000$$

This should be Buffer
Node instead of Buffer
Partition - my mistake.

3625
BUFFER MANAGER $\Rightarrow$ BUFFER PARTITION



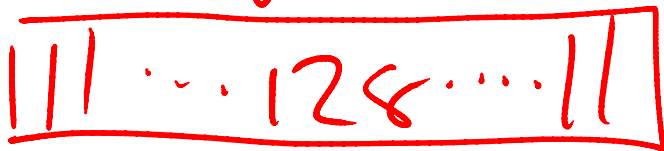
M2S40 Explaining PFS and SGAM contention in tempdb and how adding multiple files spreads the contention over multiple files, thus reducing it.



M5 Explaining log flushing. Hard limit of 32 outstanding flushes up to 2008R2. 112 from 2012 onward.

WRITELOG

LOGBUFFER



LOG CACHE

LOG FLUSH CACHE

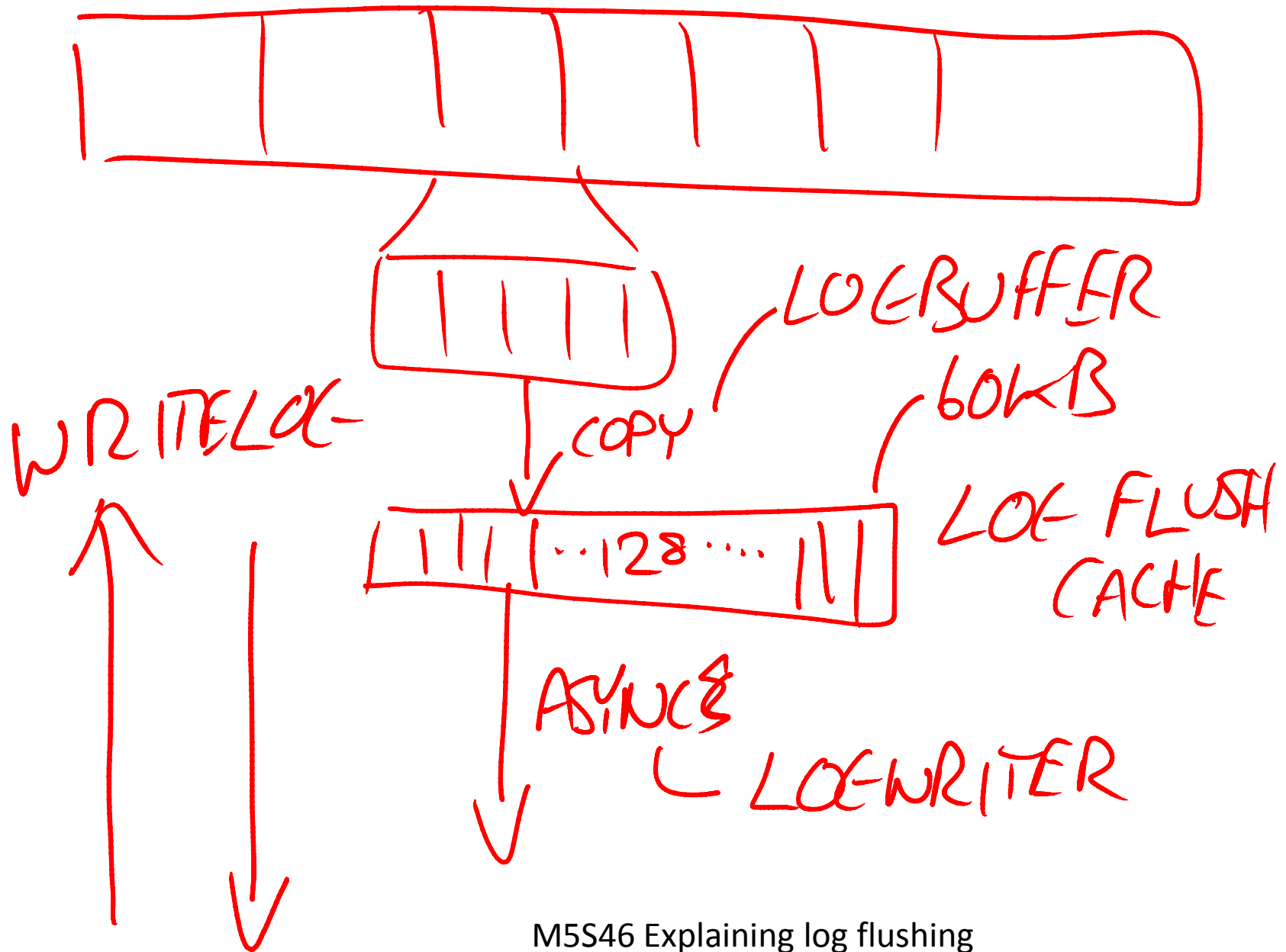
ASYNC
WRITE

LOGWRITER

48 32

112 128

16 3860K 2008R2 2012

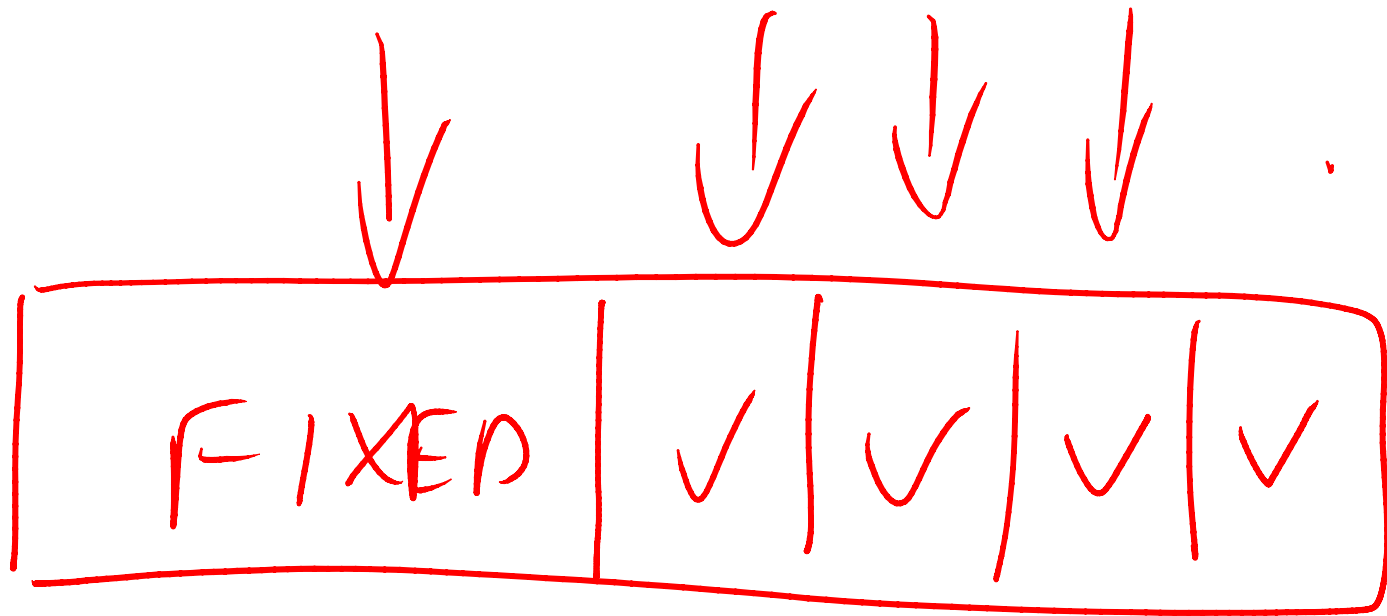


M5S46 Explaining log flushing

sys.dm_io_pending_io-
requests

Phys & Disk Avg Disk W & Len

M5 Correlate WRITELOG time with average write latency for the log. If WRITELOG higher, there is contention in the log flush mechanism.



LOP_MODIFY_ROW

LOP_MODIFY_COLUMN

M5S19 Explaining how the Access Methods determines how to log UPDATES efficiently by splitting the logging by portion of the record. Each variable length column is a portion, as is the fixed-width portion (up to 16 bytes in each log record)

T2

IX

T1

IX

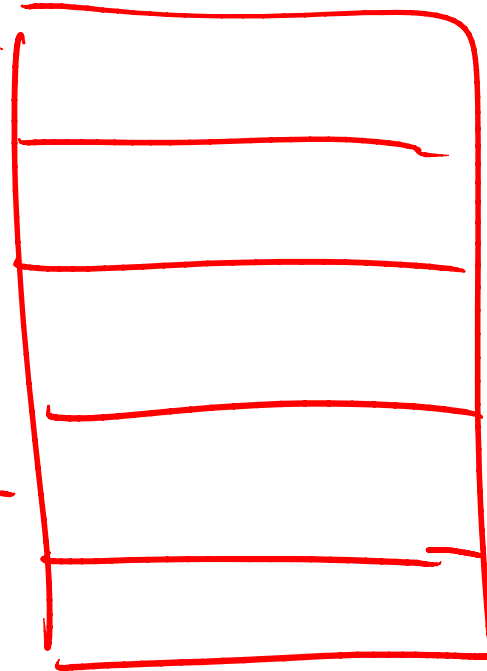
LATCH

IX
SH

X

LATCH_

X

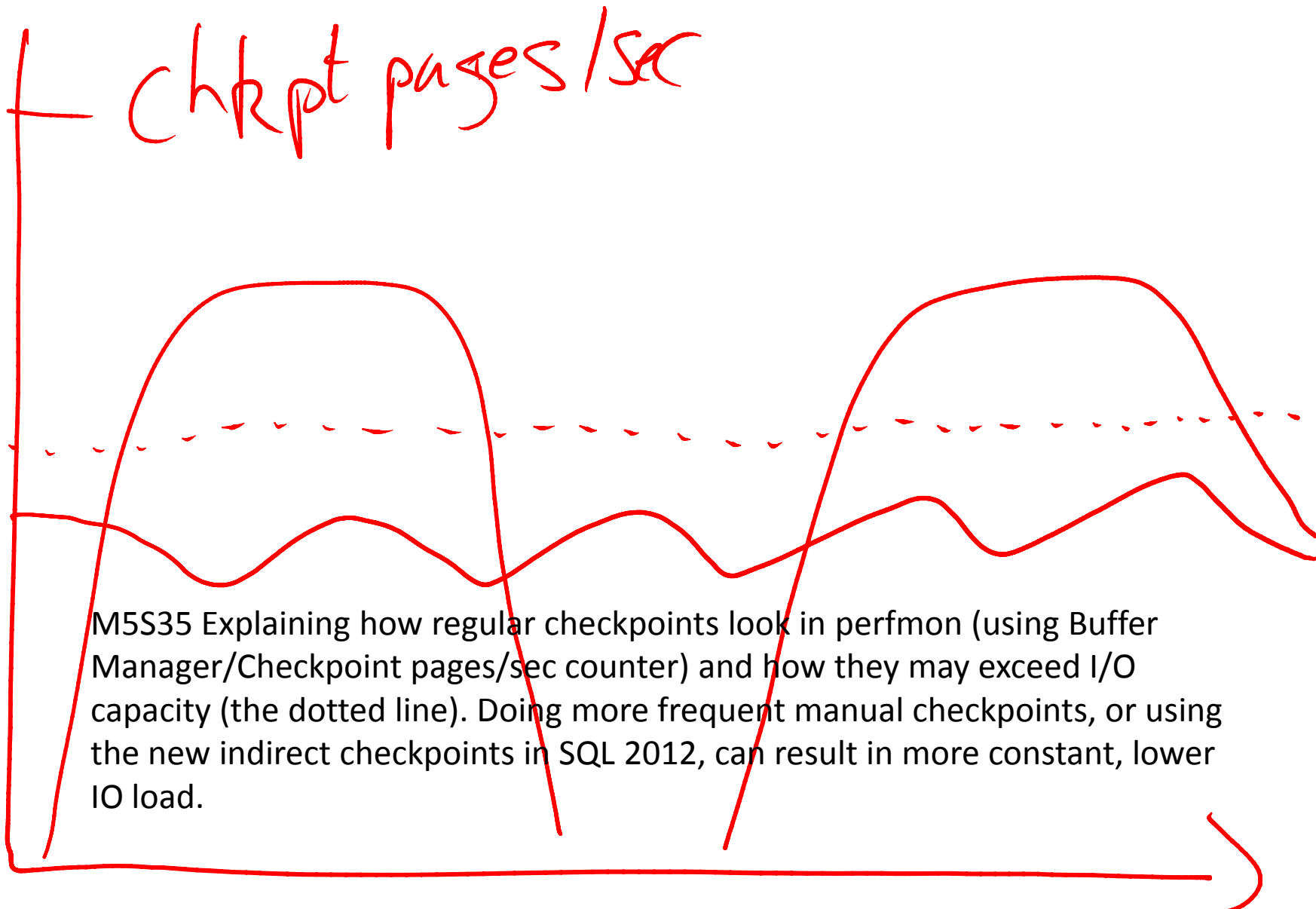


M5 Discussing locks vs. latches. Locks protect a change in transactions. Latches give a thread access to a page in memory.

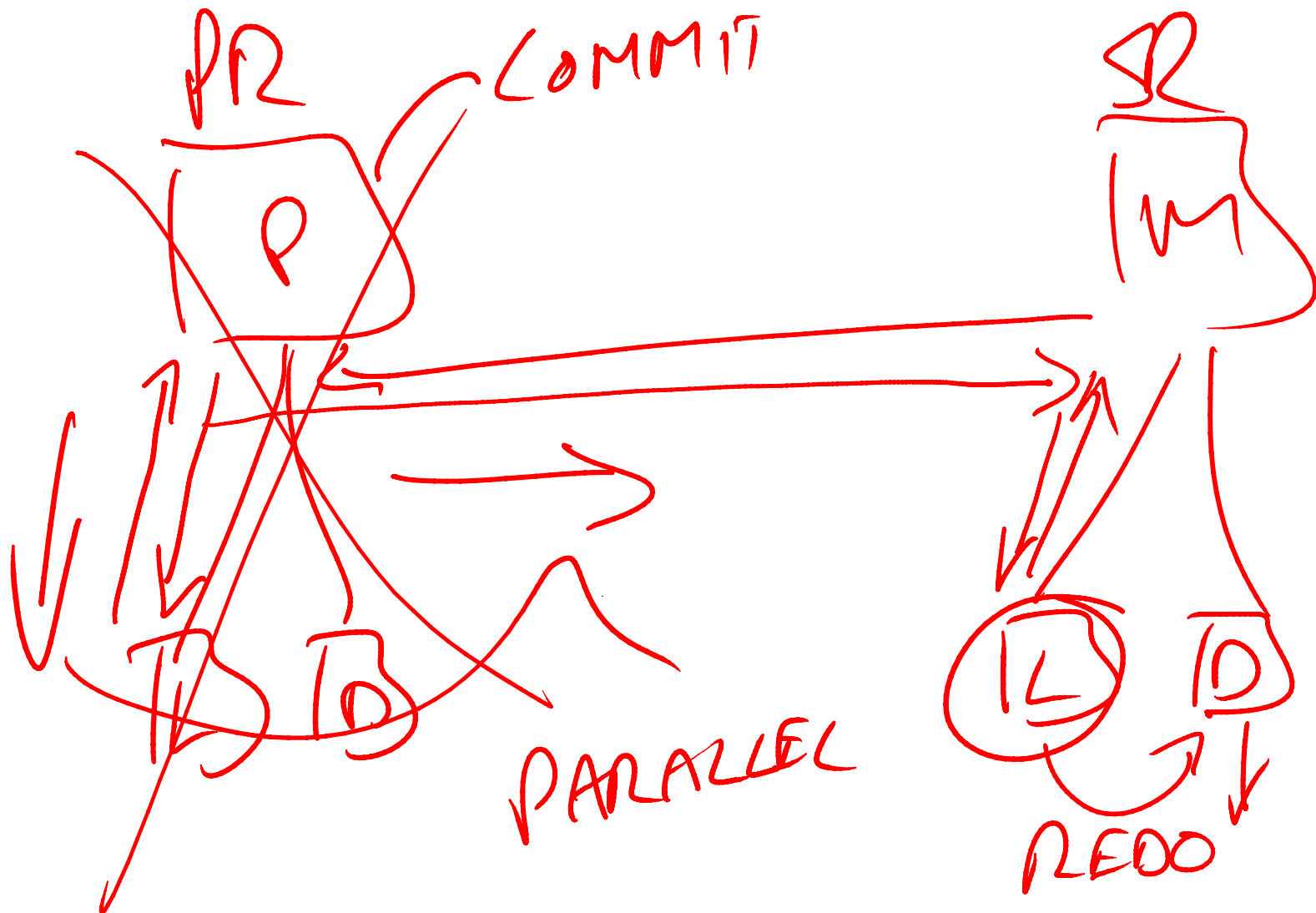
sys.dm.os_latch_stats

PAGELATCH_
PAGEIO_LATCH_

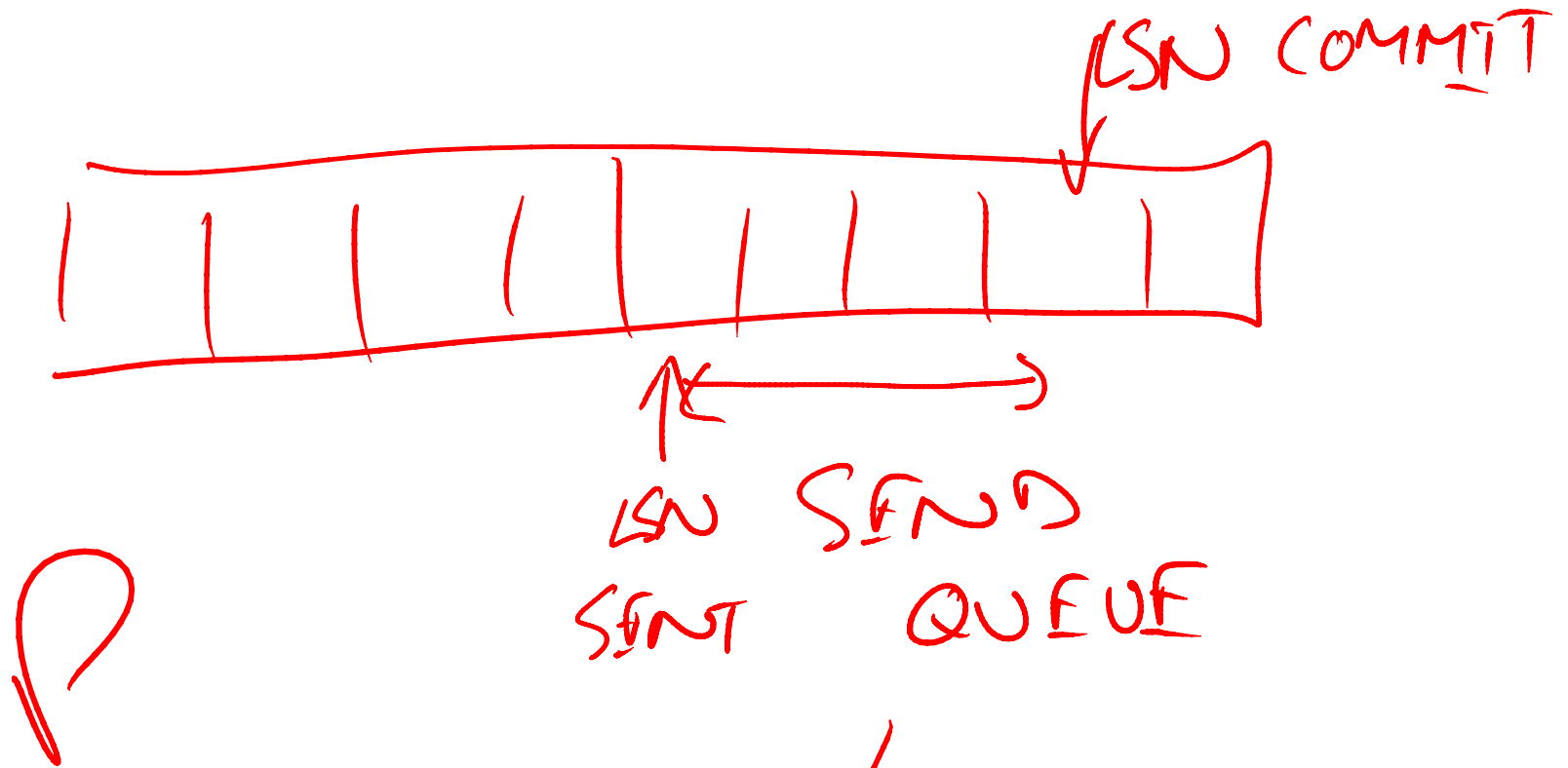
chkpt pages/sec



M5S35 Explaining how regular checkpoints look in perfmon (using Buffer Manager/Checkpoint pages/sec counter) and how they may exceed I/O capacity (the dotted line). Doing more frequent manual checkpoints, or using the new indirect checkpoints in SQL 2012, can result in more constant, lower IO load.

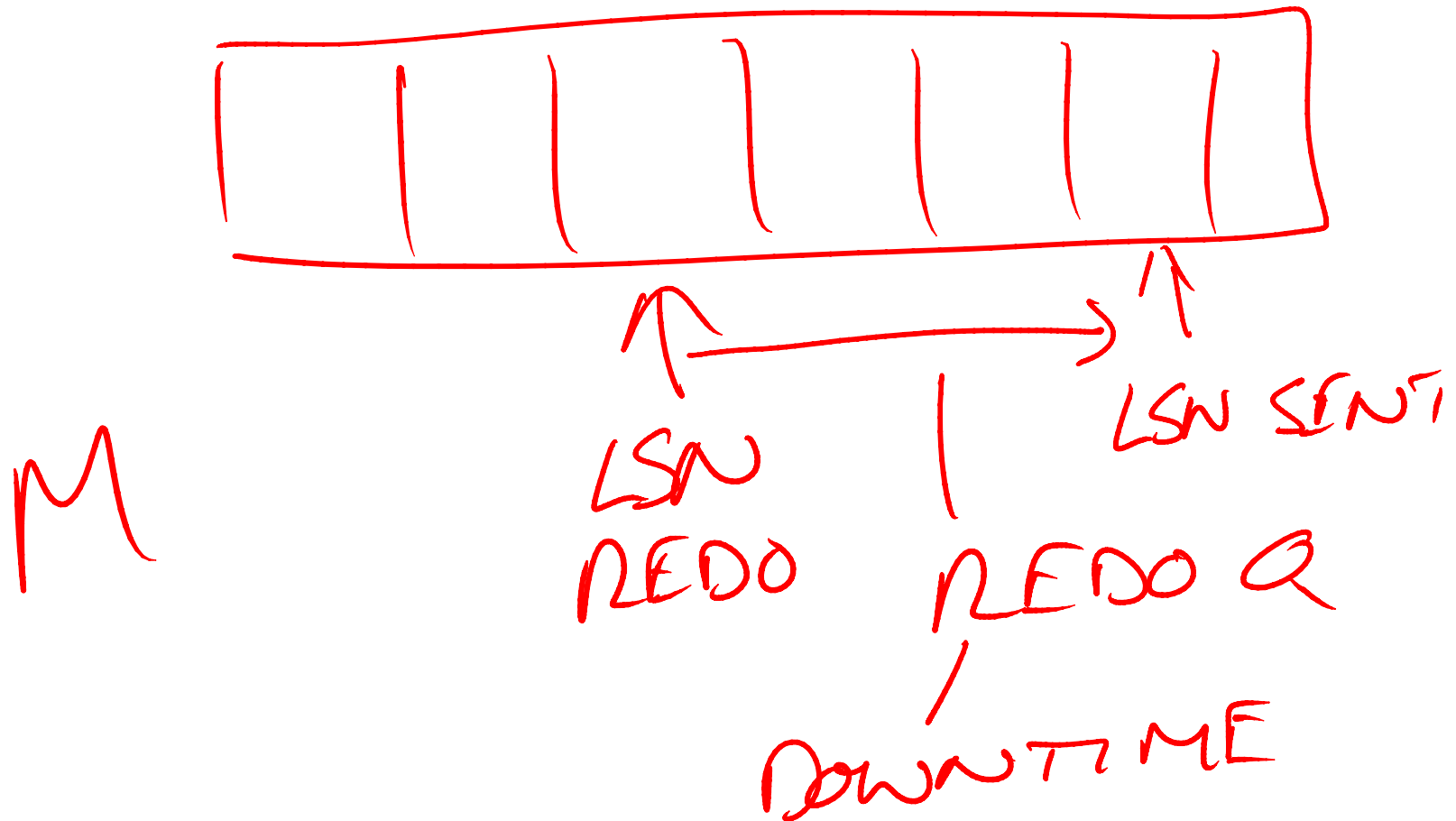


M5 Showing that with sync mirroring or AG, log writes have to happen locally and on the mirror/replica before the local transaction can commit. The I/Os are issued in parallel, not local, complete local, remote.

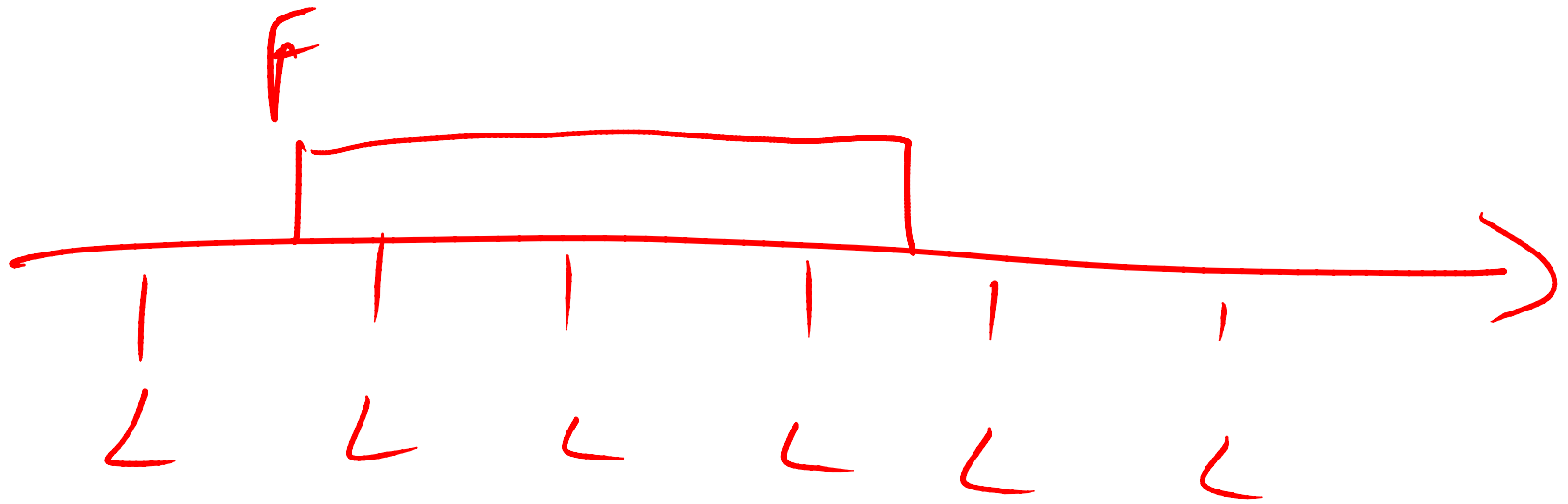


log-reserved - ~~data~~ DATA/WORK LOSS
DATABASE - ~~mirror~~ ONLINE

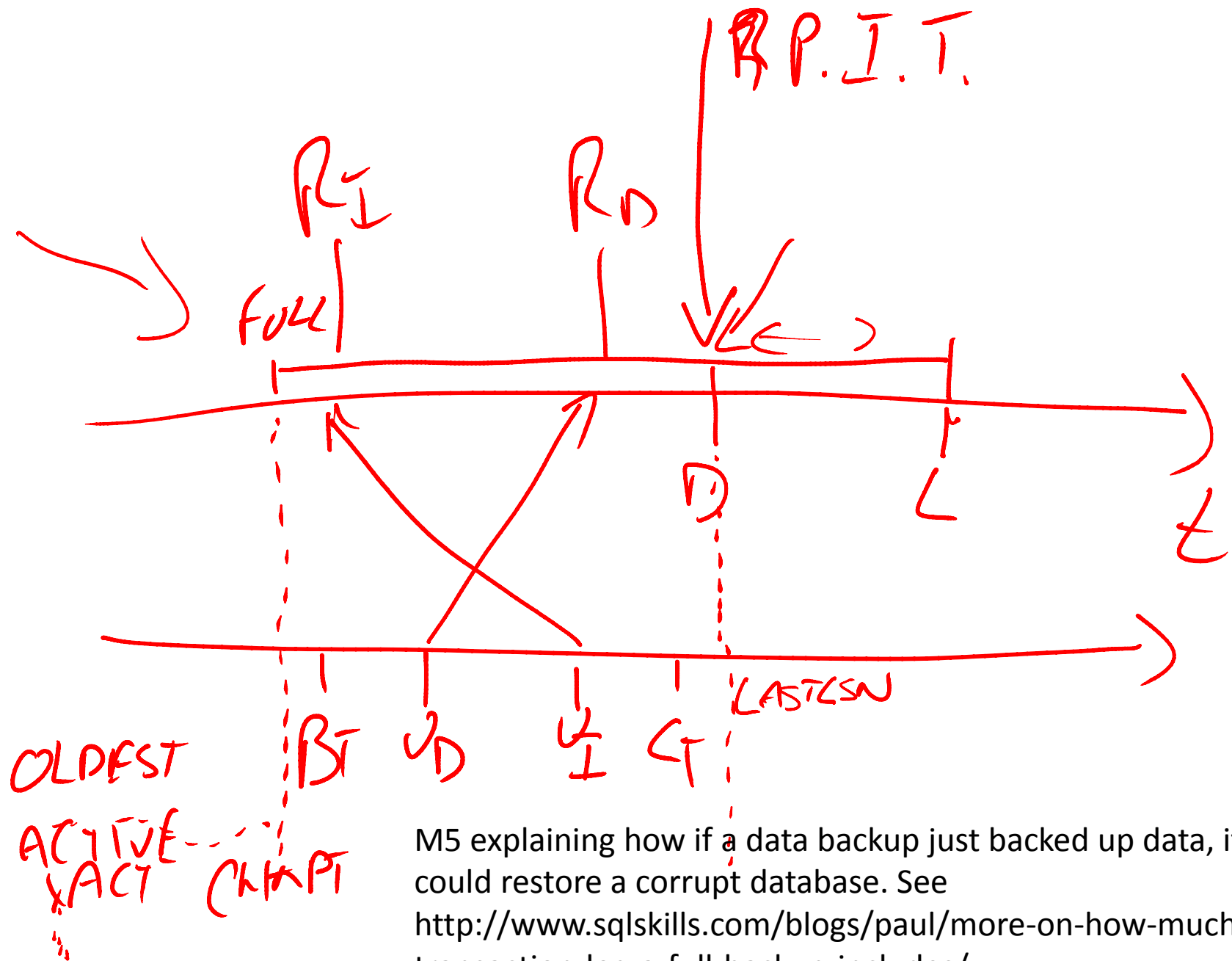
M5 Discussing the log that can build up on the principal before it's sent to the mirror during async mirroring – called the SEND queue. Represents data loss on principal crash.



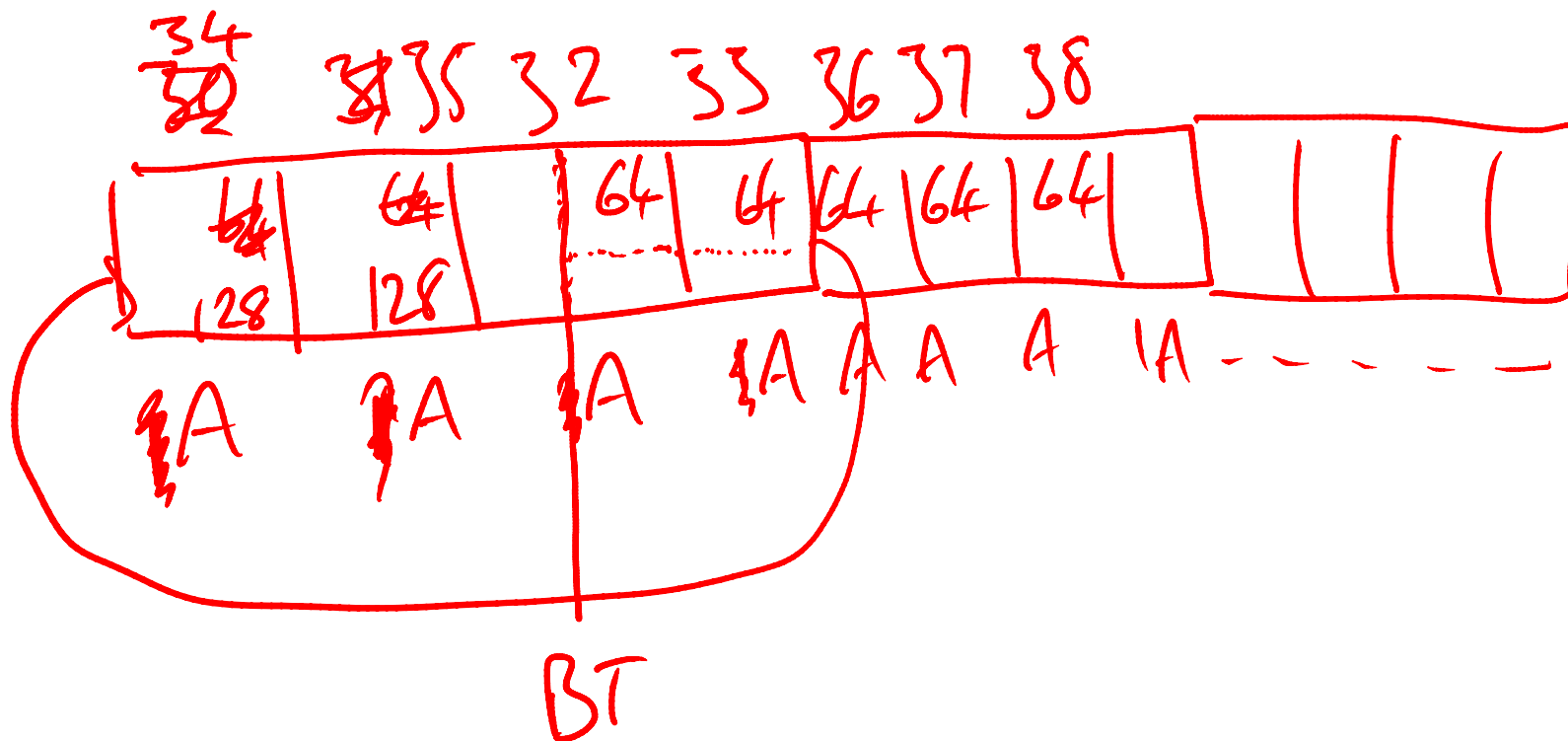
M5 Discussing the log that can build up on the mirror before it's been replayed, during all modes of mirroring – called the REDO queue. Represents downtime on principal crash.



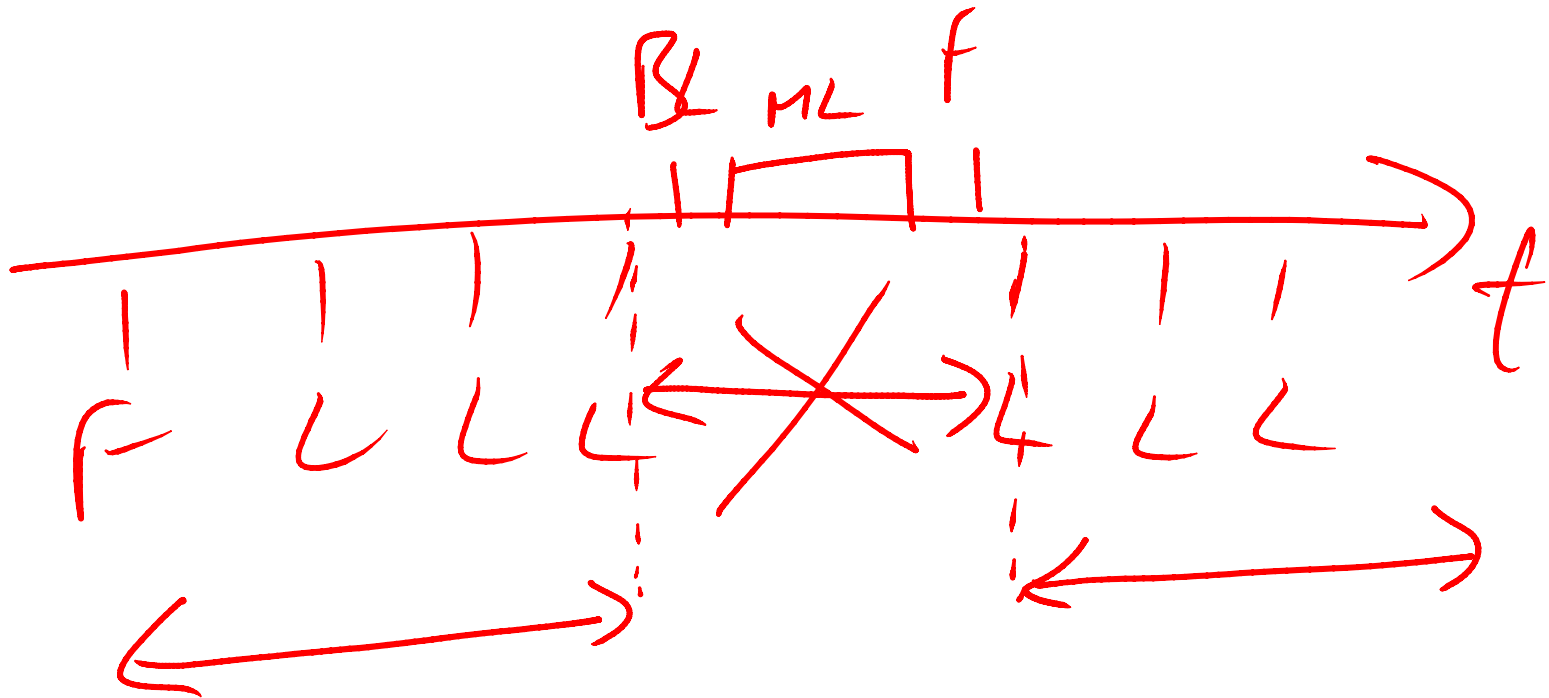
M5S42 It's possible for concurrent log backups to occur with a full backup from 2005 onward. Log clearing cannot happen though. It's deferred until the full backup completes, giving the impression that a full backup clears the log.



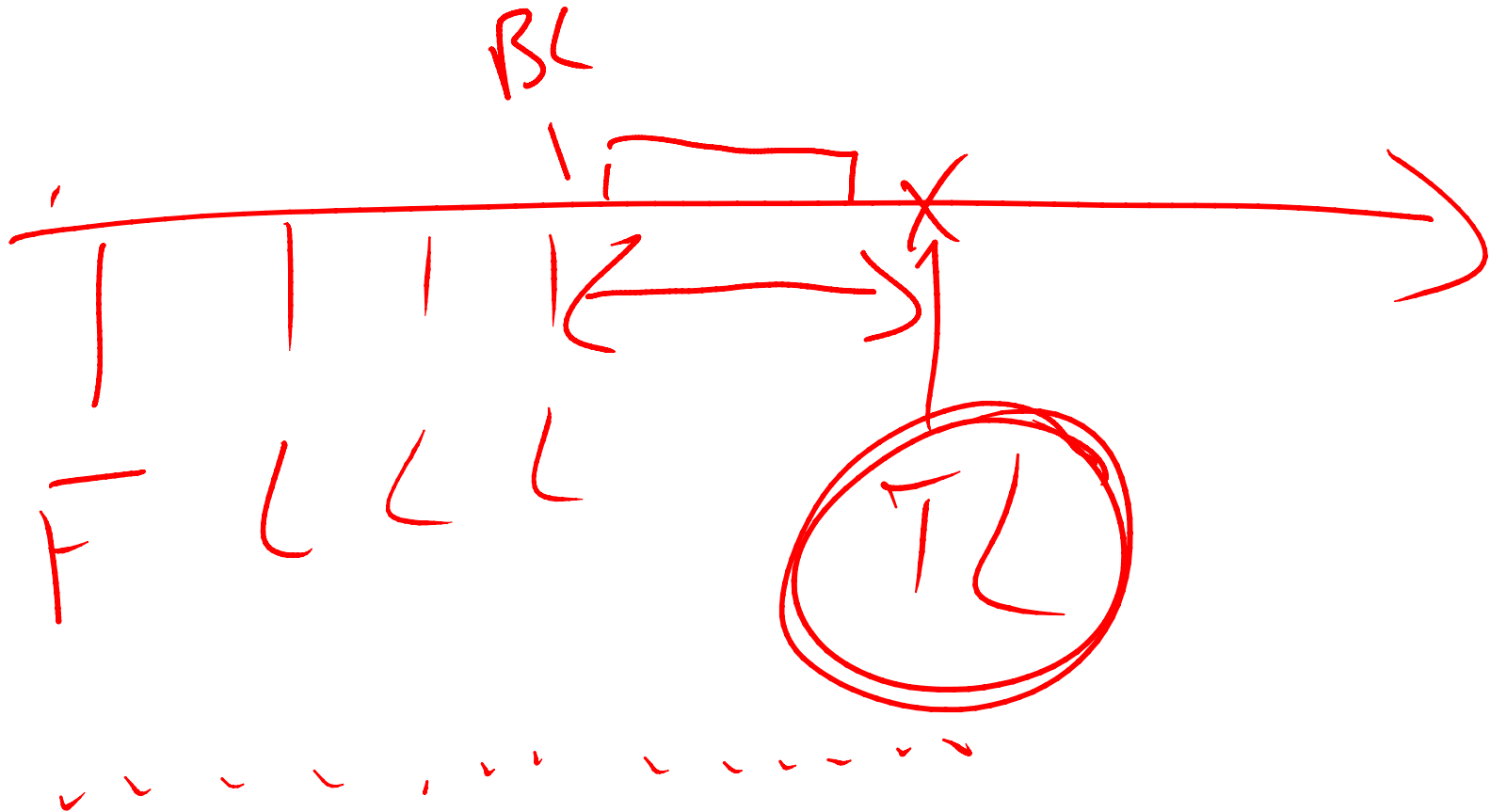
M5 explaining how if a data backup just backed up data, it could restore a corrupt database. See <http://www.sqlskills.com/blogs/paul/more-on-how-much-transaction-log-a-full-backup-includes/>



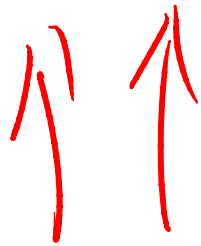
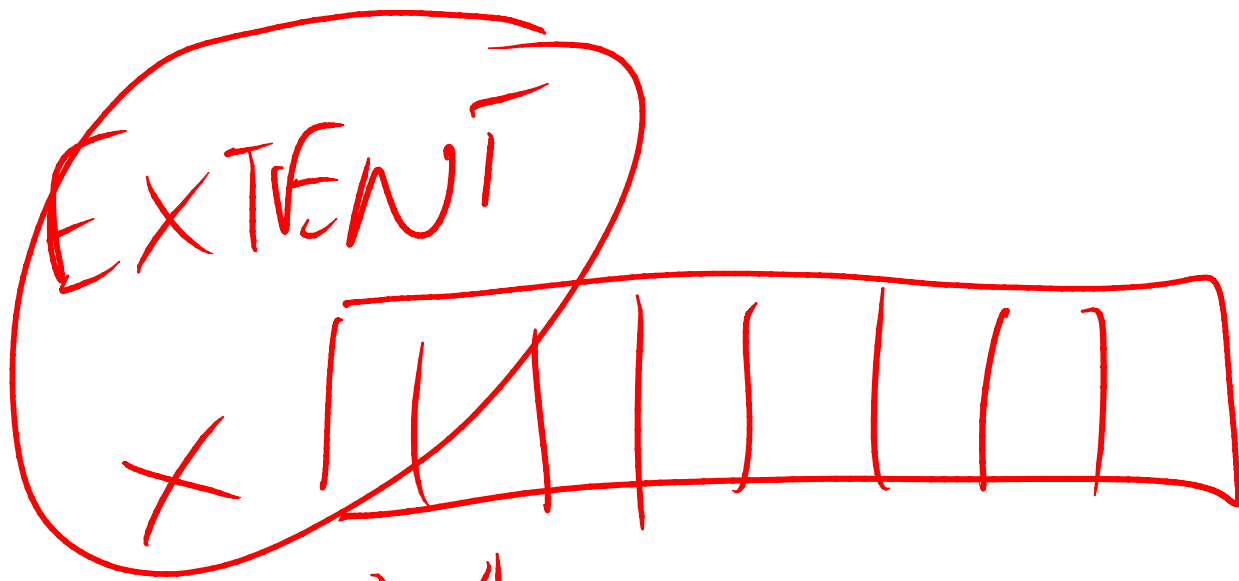
M5S48 Explaining the circular log demo. We discussed log space reservation that happens to ensure the active transactions can roll back without having to grow the log – this accounts for extra log growths. Space reserved in the log does not make a VLF become active – as there are no log records written out, just empty space reserved.



M5S60 If a log backup contains a minimally-logged operation, it cannot be used to do a point-in-time restore using STOPAT for any point in time between the beginning and end of the backup. It can be used as part of a restore sequence as normal though.



M5S60 Explaining that if a minimally-logged operation is present in a log backup, you cannot do a point-in-time restore to any time covered by that backup (except the start or end of that time period). You also cannot do a tail-of-the-log backup after a crash that destroys the data files if there has been a minimally-logged operation since the last log backup. Well in 2008 R2/2012, you can, but the log backup is invalid.



PROBE

M5 Explaining part of the deferred-drop functionality. Before an extent can be deallocated, an X lock is acquired on it and all 8 page locks are probed (instant acquire X lock and release) to make sure nothing else has them locked.

100,000

10

M5S60 Explaining that full recovery doesn't mean there's always one log record per operation. An index rebuild in full recovery will log full LOP_FORMAT_PAGE log records instead of multiple LOP_INSERT_ROWS log records. It's more efficient, but is still fully logged.

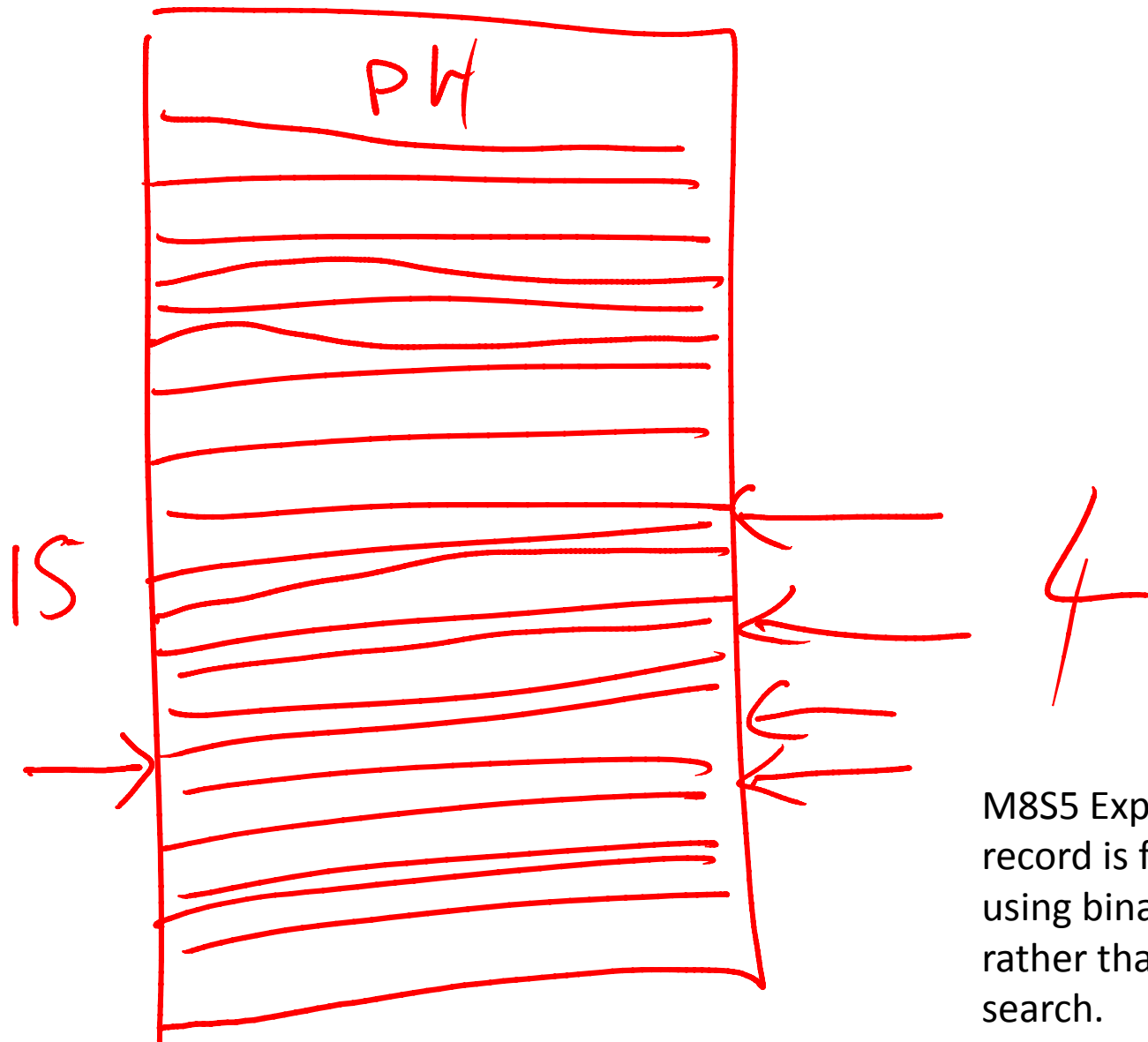
~~INS
INS
INS~~

F - PAGE
F - PAGE

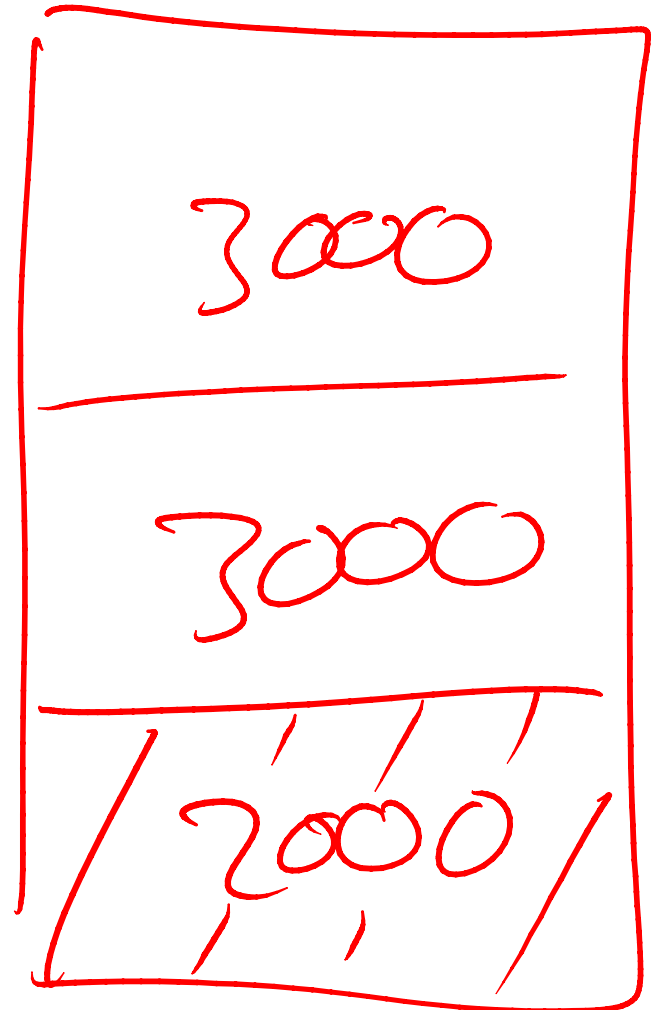
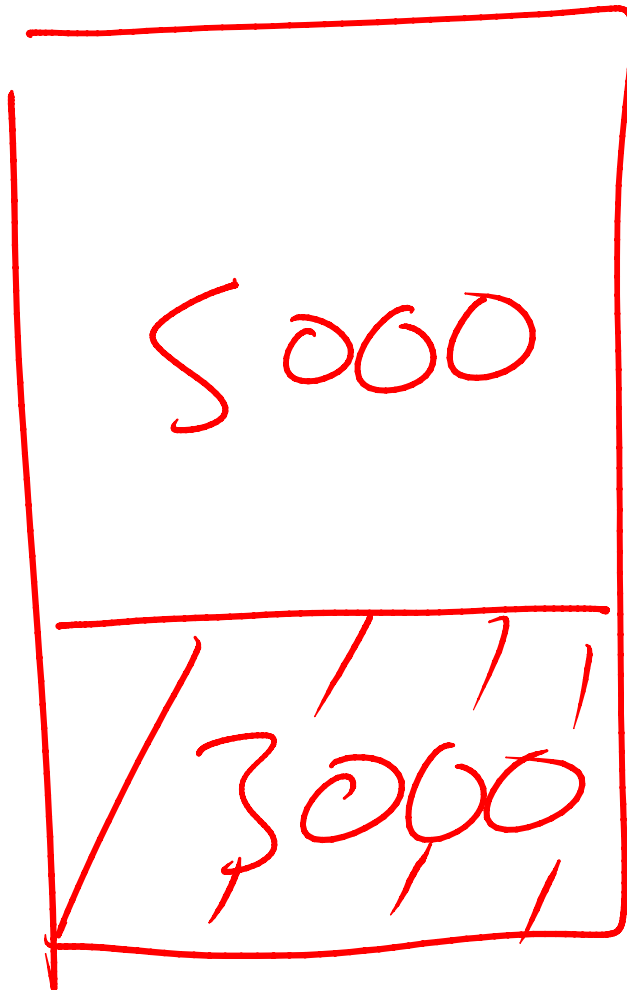
10



M5S25 VLF fragmentation is bad because of having to search through the VLFs for a particular VLF sequence number. See the KB article link for more info.

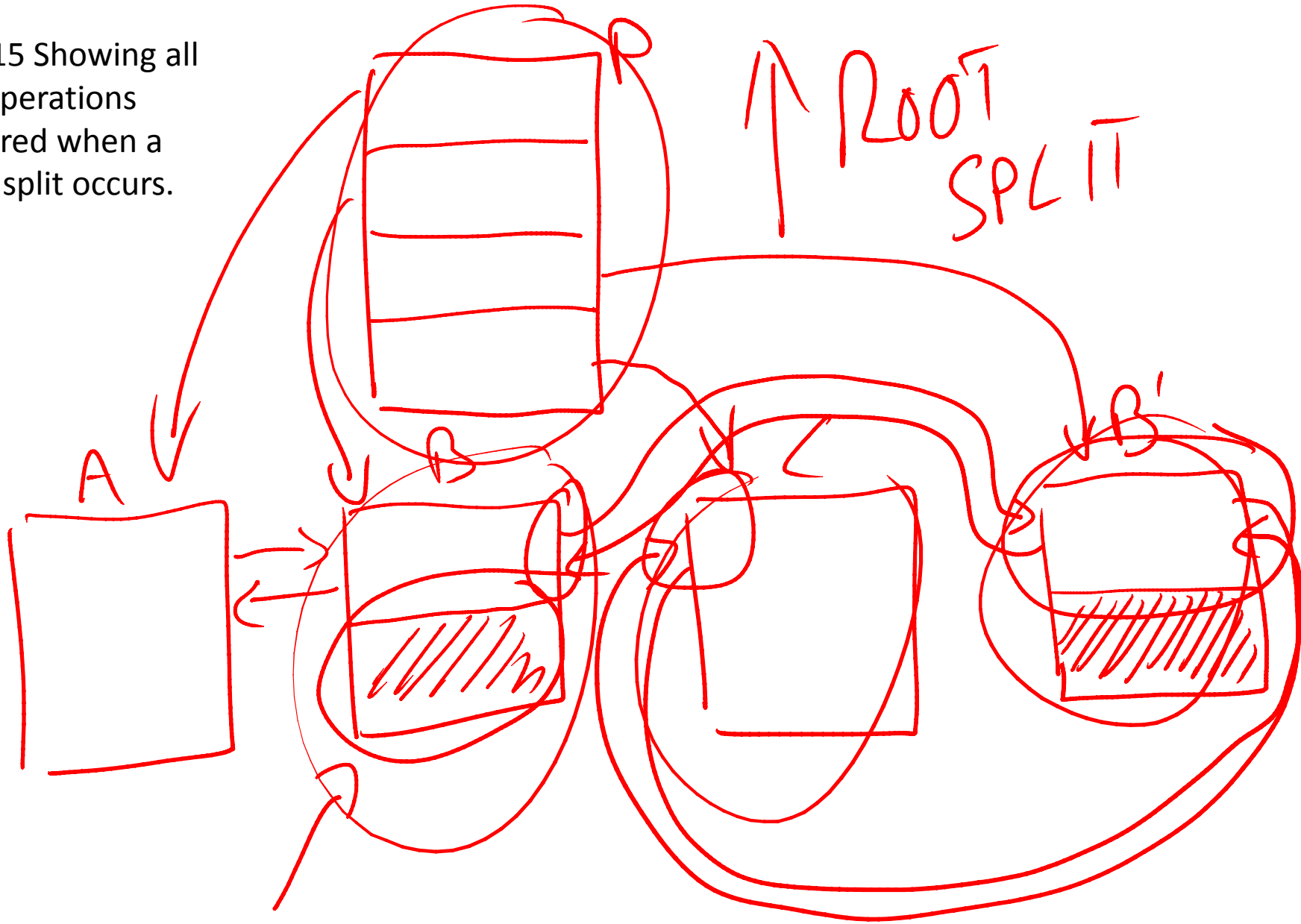


M8S5 Explaining how a record is found on a page using binary search rather than brute force search.



M8S13 Explaining how fixed-size large rows leads to low page density and wasted space.

M8S15 Showing all
the operations
required when a
page split occurs.



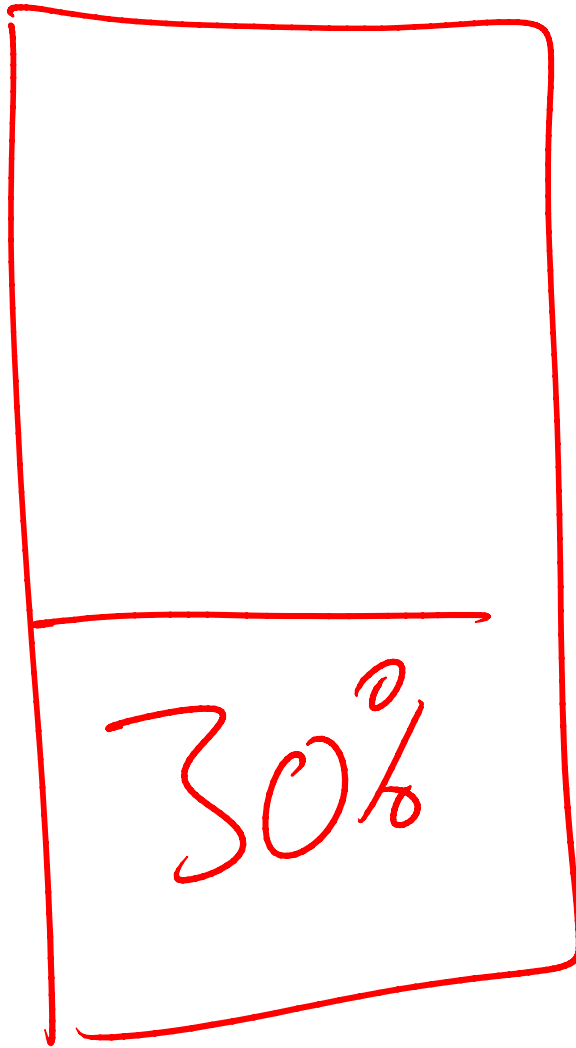
	NO SPLIT	SPLIT	X
1000	1256	7572	x5.5
100	356	6764	x18
10	268	11164	x47

M8S22 The numbers from the page split logging cost demo.

COMMENTS

PAUL	CREG	DAMIENS
MEMBERID	KIMBERLY	

M8S23 The MySpace story. The 'K' is the tiny portion of comments on Kimberly's page because she wasn't very popular back then... ☺



$$FF = 70$$

M8S25 To leave 30% free space, set the fillfactor to 70.

ITB



M8S37 Explaining staggered index maintenance with database mirroring. See <http://bit.ly/IS04W5> for more explanation