Architecting for end to end data integrity
May 5, 2013

Henry Newman
CEO/CTO
Instrumental, Inc.
hsn@instrumental.com

What is data integrity

- The integrity of data includes:
  - Failure of file system or database which corrupts data
  - Failure of storage system which causes data loss
  - Failure of checksums and/or ECC within a channel or memory to correct, or at least detect, faulty data
- Each of these can be mitigated to improve the end to end integrity of data
  - Notice the choice of words “mitigated”

What do 9s means

Instrumental
HIGH PERFORMANCE INNOVATION

		Data Loss in Bytes				
9s	Data Reliability %	1 PB	50 PB	100 PB	500 PB	1 EB
2	99%	11,258,999,068,426	562,949,953,421,312	1,125,899,906,842,620	5,629,499,534,213,120	11,529,215,046,068,500
3	99.9%	1,125,899,906,843	56,294,995,342,131	112,589,990,684,262	562,949,953,421,312	1,152,921,504,606,850
4	99.99%	112,589,990,684	5,629,499,534,213	11,258,999,068,425	56,294,995,342,125	115,292,150,460,672
5	99.999%	11,258,999,068	562,949,953,419	1,125,899,906,838	5,629,499,534,188	11,529,215,046,016
6	99.9999%	1,125,899,907	56,294,995,344	112,589,990,688	562,949,953,438	1,152,921,504,640
7	99.99999%	112,589,991	5,629,499,531	11,258,999,063	56,294,995,313	115,292,150,400
8	99.999999%	11,258,999	562,949,956	1,125,899,913	5,629,499,563	11,529,215,104
9	99.9999999%	1,125,900	56,294,994	112,589,988	562,949,938	1,152,921,472
10	99.99999999%	112,590	5,629,500	11,259,000	56,295,000	115,292,160
11	99.999999999%	11,259	562,950	1,125,900	5,629,500	11,529,216
12	99.9999999999%	1,126	56,294	112,588	562,938	1,152,896

- This is significantly different than people think about availability
  - Where 99.999% is considered an amazing number
- Where does the data loss happen?
  - 1 byte per file or all in 1 file?
  - What if it is metadata or file header data like a jpg header?
  - What if it is the file system?

Agenda

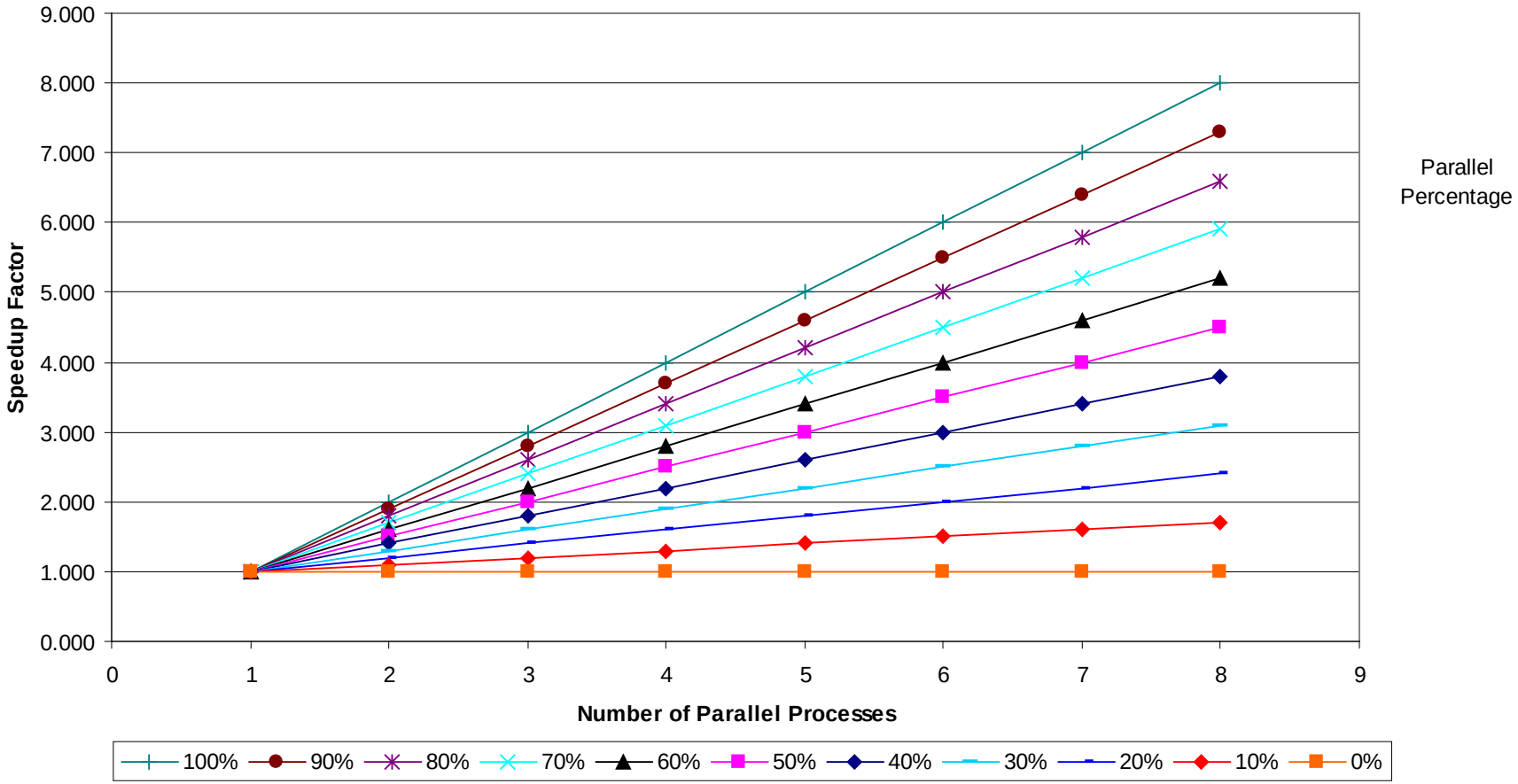
Instrumental
HIGH PERFORMANCE INNOVATION

- Introduction to the issues
- OS to file systems layers
- Hardware, checksum and ECC
- Review of requirements to mitigate data loss

What are the issues

Things have not changed in some ways but changed significantly in other ways

Amdahl's Law:Some Facts



- Where is your hardware and software fit for scalability?
  - Java I/O is pretty bad
  - What about FC, SAS?
  - What about disk seek and latency time?

Disk scalability

Instrumental
HIGH PERFORMANCE INNOVATION

Record Size	% Utilization Disk 7.2K 3.5 inch Constellation Write	Expected Bandwidth 7.2K 3.5 inch Constellation Write	% Utilization Disk 7.2K 3.5 inch Constellation Read	Expected Bandwidth 7.2K 3.5 inch Constellation Read	% Utilization Disk 15K 2.5 inch Read Savio	Expected Bandwdith 15K 2.5 Read Savio in MiB/sec	% Utilization Disk 15K 2.5 inch Read Savio	Expected Bandwdith 15K 2.5 Read Savio in MiB/sec	% Utilization Disk 10K 2.5 inch write Drives	Expected Bandwdith 10K 2.5 Write Savio in MiB/sec	% Utilization Disk 10K 2.5 inch Read Drives	Expected Bandwdith 10K 2.5 Read Savio in MiB/sec	Expected Bandwdith Read/write Pulsar in MiB/sec	% Utilization SSD Seagate Pulsar
256	0.02%	0.02	0.02%	0.02	0.03%	0.05	0.03%	0.05	0.02%	0.03	0.02%	0.04	0.89	0.24%
512	0.03%	0.04	0.04%	0.04	0.06%	0.10	0.06%	0.11	0.04%	0.07	0.05%	0.07	1.78	0.48%
1024	0.07%	0.07	0.07%	0.08	0.11%	0.19	0.13%	0.21	0.09%	0.14	0.09%	0.15	3.54	0.96%
4096	0.27%	0.29	0.29%	0.31	0.45%	0.76	0.50%	0.84	0.36%	0.55	0.38%	0.58	13.78	3.72%
8192	0.53%	0.57	0.58%	0.61	0.90%	1.52	1.00%	1.68	0.71%	1.09	0.75%	1.16	26.56	7.18%
16384	1.06%	1.13	1.14%	1.22	1.79%	3.01	1.98%	3.33	1.41%	2.17	1.49%	2.29	49.57	13.40%
32768	2.10%	2.24	2.26%	2.41	3.52%	5.91	3.89%	6.53	2.78%	4.28	2.94%	4.52	87.42	23.63%
65536	4.11%	4.39	4.42%	4.72	6.80%	11.42	7.48%	12.57	5.41%	8.33	5.70%	8.78	141.43	38.22%
131072	7.90%	8.43	8.47%	9.04	12.73%	21.39	13.92%	23.39	10.26%	15.80	10.79%	16.62	204.64	55.31%
262144	14.64%	15.62	15.62%	16.66	22.59%	37.95	24.44%	41.06	18.61%	28.66	19.48%	30.00	263.53	71.22%
524288	25.55%	27.25	27.02%	28.82	36.85%	61.91	39.28%	66.00	31.38%	48.32	32.61%	50.22	307.82	83.19%
1048576	40.70%	43.41	42.55%	45.38	53.86%	90.48	56.41%	94.77	47.77%	73.56	49.18%	75.74	336.06	90.83%
2097152	57.85%	61.71	59.69%	63.67	70.01%	117.61	72.13%	121.18	64.65%	99.57	65.93%	101.54	352.21	95.19%
4194304	73.30%	78.19	74.76%	79.75	82.36%	138.36	83.81%	140.80	78.53%	120.94	79.47%	122.38	360.89	97.54%
8388608	84.59%	90.23	85.56%	91.26	90.33%	151.75	91.19%	153.20	87.98%	135.48	88.56%	136.38	365.39	98.75%
16777216	91.65%	97.77	92.22%	98.37	94.92%	159.46	95.39%	160.26	93.60%	144.15	93.93%	144.66	367.68	99.37%

Scalability of Data Integrity

- Hardware construction
  - Performance bottlenecks in memory, PCIe, switches, and storage
  - Linear scalability is not occurring
  - For archives or disk migration, times are not scaling
  - Rebuild times are increasing
  - No per file checksum standard
  - Computing software checksums is CPU intensive
  - The promise of T10 PI/DIF has been a promise for a long time
    - But it finally happening
- More hardware means a higher probability of corruption given the parts count increases

Scalability of Data Integrity

Instrumental
HIGH PERFORMANCE INNOVATION

Year	Drive Size in GB	Drive Type	Max Transfer Rate in MB/sec	Estimate Time to Read the disk	Drives to Saturate Channel	Estimate Time to reconstruct RAID-5 8+1 at 10% of full rate	Estimate Time to reconstruct RAID-6 8+2 at 10% of full rate	Total Time in hours RAID-6 reconstruct
1994	4	SCSI	9	556	2.2	36000	40000	11.11
1998	18	USCSI	29	776	1.4	81000	90000	25.00
2002	146	FC	89	2051	2.2	131400	146000	40.56
2005	300	FC	119	3151	3.4	135000	150000	41.67
2009	450	FC	125	4500	6.4	101250	112500	31.25
2009	1500	SATA	105	17857	7.6	337500	375000	104.17
2011	3000	SATA/SAS	112	33482	7.1	675000	750000	208.33
2013 est	4000	SATA/SAS	129	38820	6.2	900000	1000000	277.78

- Increases in rebuild times
  - Higher capacities but transfer rates are not increasing as fast
  - RAID-6 is reaching the end of its useful life
- Declustered methods are going to be needed to ensure data integrity
  - Regular triple failures are on the horizon
    - And here for large systems

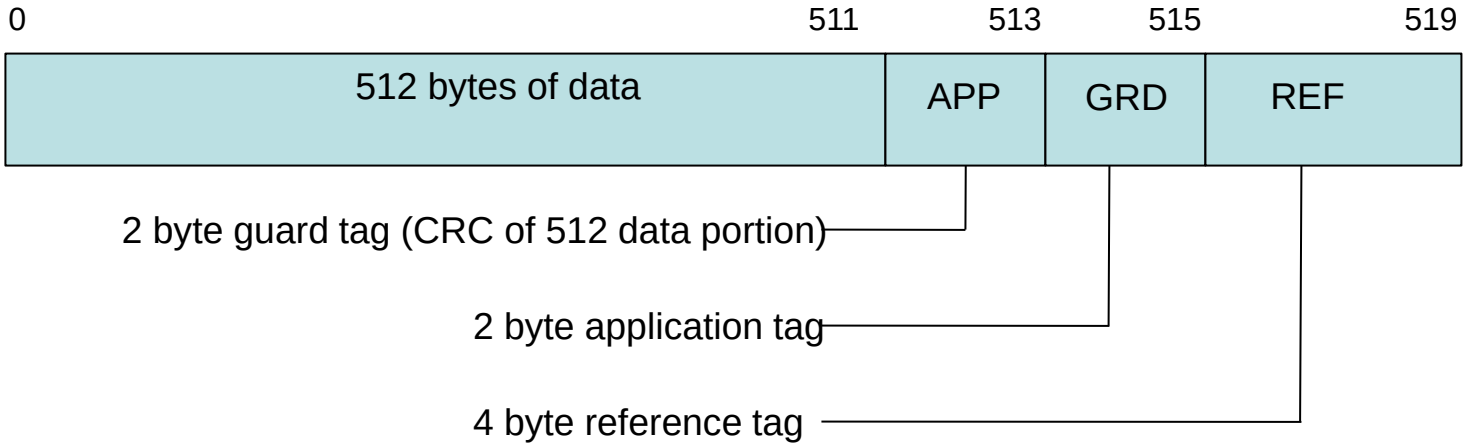
Scalability of Data Integrity

- ECC
  - Error Correcting Code – traditionally used for memory but can also be used for storage
  - 8 bits for 64-bit paths can detect and automatically correct errors of 1 bit and can detect, but only detect errors of 2 bits
- CRC
  - Cyclic Redundancy Check - used in networks and storage devices
  - Usually 16 or 32 bits in length using various algorithms in hardware devices
  - Intel now has a CRC instruction but more on this later
- Different data integrity methods use either ECC or CRC
- There has not been much change since the design of the channels

Scalability of Data Integrity

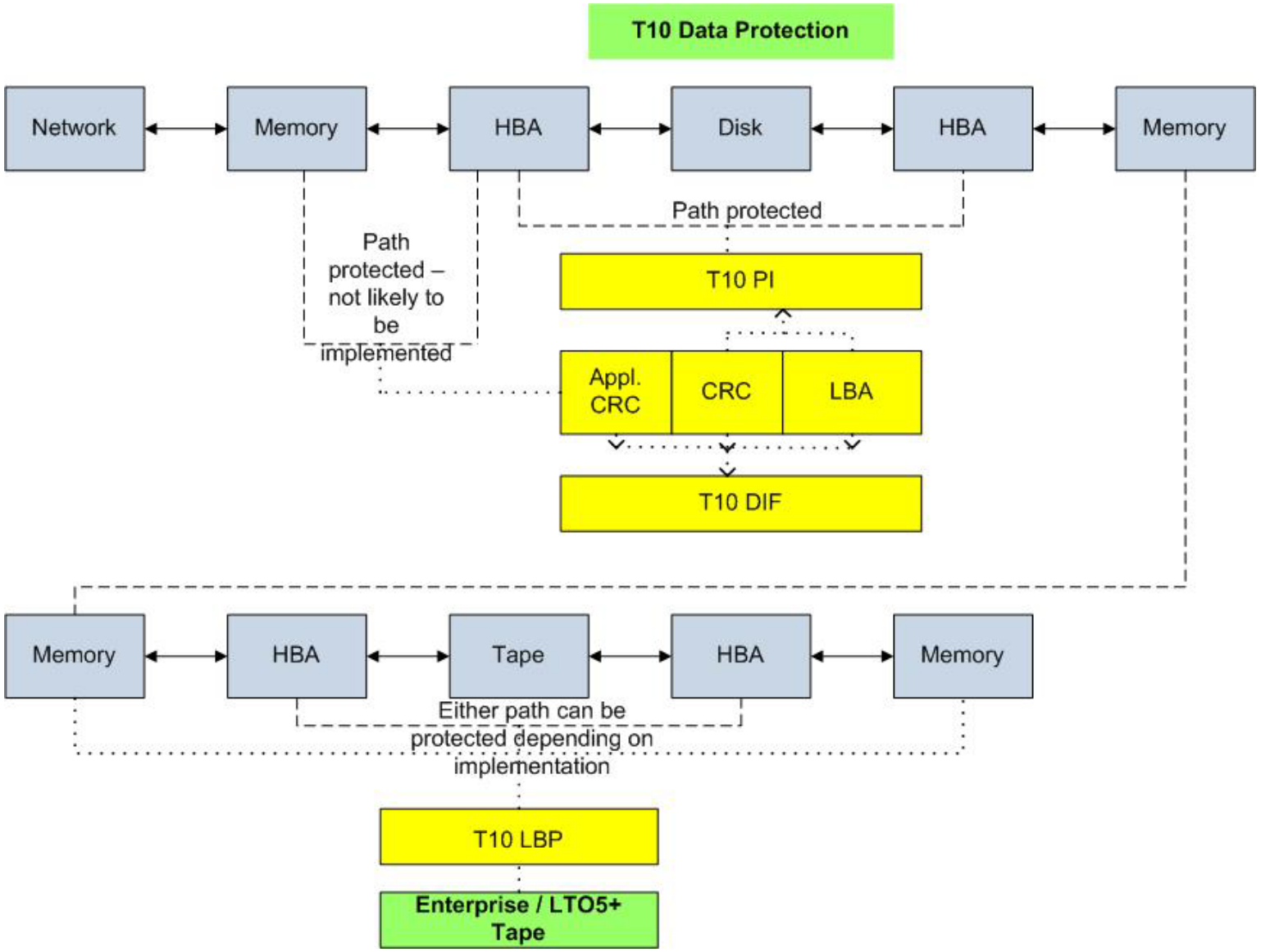
- Attempts to address data integrity via hardware and software
  - Disk
    - T10 DIF (Data Integrity Field disk)
    - T10 PI (Protection of Information disk)
  - Tape
    - T10 (Tape logical block protection)
- No full implementation of standards among vendors and something needs to be done in software
  - Not possible because of POSIX limitations
- No coverage of the entire data path as application CRC not implemented in VFS layer
  - Again POSIX limits this
- Users forced to develop their own data integrity checks and run them periodically
  - This is not efficient and does not scale

The T10 PI Field (520 bytes)



- PI adds an application checksum but:
  - What standard framework exists to allow the application to do this?
    - None is the answer as it is not supported in user space
    - No file system or OS support for application tag
- Application field cannot be passed through VFS layer
  - Requires changes to POSIX
- What about NFS
    - This is not going to happen
- Issues with memory alignment of the APP field
  - 520 is not a power of 2
- For 4K sectors it will change to 4104 bytes

Scalability of Data Integrity



Microsoft
oint 97-2003 Prese

Perfect world numbers

Instrumental
HIGH PERFORMANCE INNOVATION

	Sustain Transfer Rate Per Second for a Year						
	SDC Rate	**0.5 GiB/sec**	**1 GiB/sec**	**10 GiB/sec**	**100 GiB/sec**	**1 TiB/sec**	**10 TiB/sec**	**100 TiB/sec**
SAS T10 PI detection	10E28	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	10E27	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	10E26	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	10E25	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	10E24	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	10E23	0.0	0.0	0.0	0.0	0.0	0.0	0.3
	10E22	0.0	0.0	0.0	0.0	0.0	0.3	2.7
SAS/FC	10E21	0.0	0.0	0.0	0.0	0.3	2.7	27.1
	10E20	0.0	0.0	0.0	0.3	2.7	27.1	270.9
	10E19	0.0	0.0	0.3	2.7	27.1	270.9	2708.9
	10E18	0.1	0.3	2.7	27.1	270.9	2708.9	27089.2
SATA/IB standard	10E17	1.4	2.7	27.1	270.9	2708.9	27089.2	270892.2
	10E16	13.5	27.1	270.9	2708.9	27089.2	270892.2	2708921.8
	10E15	135.4	270.9	2708.9	27089.2	270892.2	2708921.8	27089217.7

- Assume that channels are operating at 1 bit in 10E12 bits which is the standard specification for most channels
  - What happens when the world is not perfect?
  - Significant degradation in SDC rate as the channel error rate increases

Increased reliability in datapath

Instrumental
HIGH PERFORMANCE INNOVATION

- T10 Protection Information PI/DIF allows a checksum to be transmitted from the HBA and application to the disk drive
  - PI uses the application field
- It detects errors and does not correct them
  - If an error is found a SCSI retry is initiated
  - But what about the fact that ACK is already received when the data hits the cache?
    - Good question to ask vendors
- DIF significantly (1 bit in 10E28) improves reliability in the datapath
  - Information estimated by a major disk vendor reliability department
    - No support for SATA
      - Why?

Hard error rates

Instrumental
HIGH PERFORMANCE INNOVATION

Device	Hard error rate in bits	Equivalent in bytes	PB equivalent
SATA consumer	10E14	1.25E+13	0.01
SATA Enterprise	10E15	1.25E+14	0.11
Enterprise SAS/FC	10E16	1.25E+15	1.11
LTO and some Enterprise SAS SSDs	10E17	1.25E+16	11.10
Enterprise Tape	10E19	1.25E+18	1110.22

- Increased rebuild times and increased I/O increase the likelihood that another hard error will occur during the rebuild
  - This impacts the reliability of a file system unless the underlying RAID system is declustered given rebuild time
- The rate has not changed since 2005
  - No plans to change given the cost
- Other protection methods need to be found

TCP/IP checksums and CRC

- Using the standard channel error rate from a statistical point of view, the checksum will fail to detect an error between 1 packet in 16 million and 1 packet in 10 billion
  - Sources old papers but nothing has changed in the CRC and checksum layers
    - http://www.ir.bbn.com/documents/articles/crc-sigcomm00.pdf
    - http://andrei.clubcisco.ro/cursuri/f/f-sym/5master/scpd-soa/lecture-09-tcp-crc-csum.pdf

More on TCP/IP

- Another interesting fact
  - Ethernet uses a 32 bit CRC that loses its effectiveness above about 11455 bytes - after this limit the CRC-32's probability of undetected errors per frame increases. A white paper with the title "Extended Frame Sized for Next Generation Ethernets"
    - http://staff.psc.edu/mathis/MTU/AlteonExtendedFrames_W0601.pdf
- Lots of discussion on the web but a great deal of changes in the protocols will be required
  - Likely not going to happen anytime soon given complexity

Scalability of Data Integrity

- Undetectable bit error rates; a.k.a. silent data corruption:

Type	Channel Error Rate	Undetectable Error Rate
Ethernet	1.0E-12	1.0E-21 estimated
SATA	1.0E-12	1.0E-17
Fibre Channel	1.0E-12	1.0E-21
SAS	1.0E-12	1.0E-21
FC/SAS with T10 PI	1.0E-12	1.0E-28

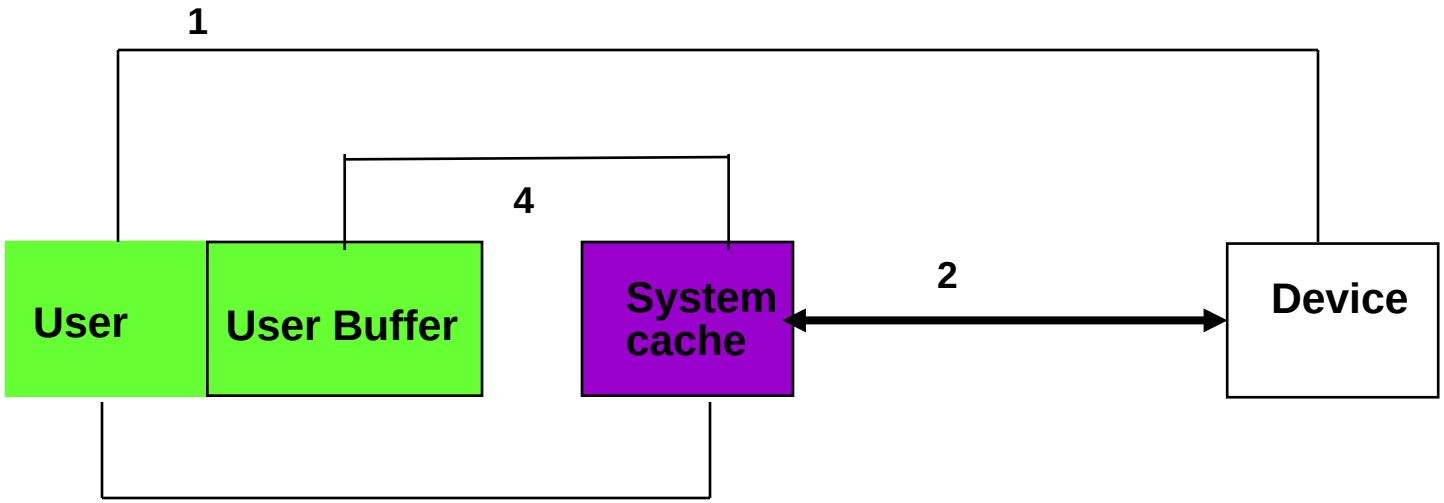
- Some form of data integrity checking is crucial since the transfer of data into and out of large archives will encounter these errors
  - **It is not if these errors will occur but when**

OS to file system layers

How applications interface with OS and POSIX

Linux System I/O Overview

- All data goes through system buffer cache
  - High overhead
- Data must compete with user operations for system cache



1 Raw I/O no file system and/or direct I/O+tape I/O
2 All I/O Under file system read/write calls
3 File system meta-data and data
4 File system metadata and data via libc, stdio etc

POSIX Standards

- Issue for applications where there are many threads writing to a common shared file system
- POSIX atomic behavior is an issue for file systems
  - Databases do not use POSIX file systems for that reason
- POSIX extended attributes - limited or no information on data provenance, backup and archiving, user metadata, file reliability and other attributes
  - No standardization in this area and no plans

POSIX Standards

Instrumental
HIGH PERFORMANCE INNOVATION

- In the past changes to the POSIX I/O standards have been proposed to address some of the outstanding issues
  - None of these have been ratified and ratification is not expected
    - Example of the rejected proposed set of extensions to POSIX for large file systems
      - When file opened by multiple clients, metadata server revokes any read caching and write buffering capabilities to force consistency and this greatly reduces performance
      - Cluster file systems that enforce POSIX consistency require stateful clients with locking subsystems

Just a few POSIX Standards Extensions

Instrumental
HIGH PERFORMANCE INNOVATION

- Examples of POSIX I/O that would help significantly
  - Needed to be more friendly to large file systems, clustering, parallelism, and high concurrency applications
  - Ordering – Replace streams of bytes with more applicable method for distributed memories mapped to many storage devices
  - Coherence – Overhead of cache invalidation for reads is high; block boundaries can present coherence issues for application
    - Method for applications to assume all responsibility for coherency is needed
  - Metadata – Standard support of “lazy” attributes needed along with portable bulk metadata interface for file system metadata
  - **Of Course Agreement on POSIX extended attributes for data integrity, archiving and other areas would have to happen**

POSIX standards for integrity

Instrumental
HIGH PERFORMANCE INNOVATION

- Possible POSIX I/O extensions (cont.)
  - Extensions for archive applications – that use POSIX API
  - T10 DIF support for the application checksum
  - Per file checksum such as a SHA256
  - Provenance to ensure integrity of chain of custody of file
- None of this is going to happen as the vendors that control POSIX do not want the extra work and cost
  - And since I am a cynical person, they do not want standards so they can sell proprietary products

Type of file systems

- POSIX interface
  - Examples include xfs, ext-4, GPFS, Lustre and many others
- REST interface
  - Amazon S3, Caringo, WOS and many others
- There is no common interface for either to pass integrity and no current plans
- Let's say I have a file on my laptop and move to one of the above
  - How do I ensure integrity?
  - How do I ensure a chain of custody?

REST/SOAP is not any better

- No framework for per file checksum
  - Wrappers such as LOCKSS have been developed but not a standard
- Whatever you do it requires work on the part of the site
  - There is a higher probability of change with these standards compared with POSIX
    - It is certainly above 0% chance of change

Hardware, checksums and ECC

Review of channels, disk, tape and some file systems

Background

- Google Study
  - FIT rates from 778-25,000 FIT per Mbit of memory
    - http://www.cs.utoronto.ca/~bianca/papers/sigmetrics09.pdf
  - Highlighted the importance of ECC and Chipkill
- IBM's target 114 FIT for SDC in Power4
  - IBM does proprietary designs for DIMMs
- Memory vendors don't publish MTBF or FIT
- FIT = Measure of failure rate in 10^9 device hours; e. g. 1 FIT = 1 failure in 10^9

- DDR2 FBDIMM (Fully Buffered DIMM)
  - FBDIMM (Intel design) uses ECC, CRC, and data mirroring to lower the SDC FIT rate to 0.10 per channel segment or 1,142,000 years
    - http://www.intel.com/cd/channel/reseller/asmo-na/eng/250634.htm

DDR3 SDC estimate

- DDR3 data channel has no CRC or other means to reduce the error rate.
- FIT rates for DIMMs are documented from 1 to 100, selecting a DIMM Fit rate of 100 from the paper by Smart Modular Technologies
  - Note: All the papers studied point to rapid swap out of failing DIMMs improves overall reliability of a system

DDR3 SDC estimate

Instrumental
HIGH PERFORMANCE INNOVATION

- The range of DIMMs on a server has been selected with a wide range
  - 16 to 64 DIMMs per system, the FIT rates are additive so a system of 16 DIMMS would have a FIT rate of 1600 (16 x 100) while the 64 DIMM system would have a FIT of 6400
- This is the failure rate for a possible SBE (single bit error) while a triple bit error would be much less likely, the values would suggest it is 3 orders of magnitude less likely
  - Thus a SDC (silent data corruption) rate for a DDR3 system would be 1.6 to 6.4 FIT
  - 1X109/1.6 = 6.25 x108 Hours / (24 x 365) = 71,347 Years
  - 1X109/6.4 = 1.5625 x108 Hours / (24 x 365) = 17,837 Years

DDR3 SDC estimate

Instrumental
HIGH PERFORMANCE INNOVATION

- Why 3 orders of magnitude?
  - The Smart paper state the DIMM FIT is one to two orders of magnitude higher than the SDRAM rate which get to the FIT of 100/DIMM
  - Double bit error FITs are another order of magnitude less likely as stated in the paper by B. Schroeder, et al, due to SECDED and other error correction techniques
  - Additional techniques such as Chipkill will further reduce the FIT of an SDC as much as 30 times in paper by V Sridharan, et al
    - I chose to only increase the FIT rate by one more order of magnitude

DDR3 SDC estimate

Instrumental
HIGH PERFORMANCE INNOVATION

- Resources
  - http://www.cs.utoronto.ca/~bianca/papers/sigmetrics09.pdf
  - http://pages.cs.wisc.edu/~shubu/papers/serproblem-hpca2005.pdf
  - http://www.intel.com/cd/channel/reseller/asmo-na/eng/250634.htm
  - http://www.smartm.com/files/salesLiterature/dram/smart_whitepaper_sbe.pdf
  - http://softerrors.info/selse/images/selse_2012/Papers/selse2012_submission_4.pdf

DDR3 SDC estimate

Instrumental
HIGH PERFORMANCE INNOVATION

- To put this in real world numbers
- The whole NCSA BW system has 25,712 nodes
  - Using the range of 17,837 to 71,347 years
  - Yields a possible SDC between 0.7 and 2.8 years in a perfect world

- DDR4 reliability enhancements
  - CRC on data path for data integrity
  - Parity on the command and address bus
  - Error flag and Error status flag
- DDR4 per pin data rate is 3.2 GT/s
  - With this higher data rate the signal integrity will require much more attention during the design cycle

SATA channel and disks

- SATA has less integrity based on the ECC in the channel than SAS
  - This is a command issue not drive
    - Data on disk has more ECC than the data in the channel
- Yes drives are an issue comparing consumer SATA to SATA interface in nearline drives
  - Hard error rate 1 per 10E14 bits
  - AFR much lower (no one listed it)
  - Nearline drives
    - Hard error rate 1 per 10E15 bits
    - AFR 1.4 M hours
- With SAS more error recover is done on the drive than with SATA
- No support for T10 PI/DIF
  - Sector sizes must be power of 2

SAS channel and disks

- Far more reliable than SATA
- Much of the industry is moving to SAS given that SATA will not signal at 12 Gb/sec
  - 8 Gb/sec SATA might be, in my opinion, too little too late
    - Combine this with SATA limitations
- SAS drives and channel support T10 standards for integrity
- More error recovery on the drive than with SATA
  - Reliability of the same drive with SAS interface is better

RAID and parity check on read

- Most RAID devices support parity check on read
  - Works for RAID levels 5 and 6 but not RAID levels 0 and/or 1
  - Confirm that the data read back matches the parity on the disk
    - Does not ensure that the parity written was correct in the 1st place
- Parity check on read only covers a small area of the problem
  - It is not something I would want as my only line of defense for SDCs

Tape

- As a reminder LTO hard error rate is 1 bit in10E17 bits and enterprise tape is generally 1 bit in 10E19th bits
  - Undetectable/mis-corrected error rate listed for LTO is 1 bit in 10E28 bits
  - Undetectable/mis-corrected error listed for vendors at 1 bit in 10E33 bits or greater
- Achilles heal for tape is therefore the fibre channel interface, or for lower end device SAS interface
  - SAS is used on lower end systems
- Tape is significantly more reliable than disk
  - Hard error rate is better
  - Silent data corruption rate of the device is listed
    - We do not know the disk rate as it is not published

Some current file systems

- IBM GPFS Native RAID
  - Software RAID using JBOD disks
    - Uses standard x86 CPUs
  - Uses ECC per block rather than T10 PI
    - Advantage for ECC are obvious
  - You only know about bad data at read time but it can be corrected
    - Given that ECC per block is used this is less of a problem than parity in my opinion

- ZFS has not shown the ability to get a high percentage of the hardware bandwidth
- Same issues at IBM with writes only confirmed on read
  - Many ZFS configuration use SATA drives
- Performance might not be what you want it to be
  - Streaming I/O is not a strength without lots of work and SSDs for logs
- But this is about reliability
  - Reporting

Vendors

- There is IBM with data protection in GPFS
- There is Oracle with ZFS appliance and others using that technology in appliances
- What about everyone else?
- Most, but not all, vendors either have T10 DIF compliant systems or are going to release them this year
  - From LSI RAID cards to enterprise storage and everything in between
  - Here are some questions to ask vendors

? for disk storage vendors

- Are you using 512 byte blocks when the sector size is 512 byte and 4K blocks when the sector size is 4K?
- What about error reporting?
  - What happens if the write fails to disk and the ACK has been sent to the application as the write to cache did not fail?
- What are you doing with the application tag?
  - Some vendors were thinking of using it for internal stuff?
    - Not a good idea to have vendor specific use of a standard

? for disk storage vendors

- Ask about rebuild time?
  - Everyone should be going to declustered RAID to reduce rebuild
    - This is critical to large system integrity
- Does the vendor support parity check on read
  - Even with T10 PI useful to understand where the error is?
- And speaking of errors what about?
  - Error management and reporting?
  - How fast are errors reporting?
  - How can errors be coordinated with other errors?

Hey Henry what about tape?

Instrumental
HIGH PERFORMANCE INNOVATION

- Tape logical block protection is a standard
  - Ask the vendors if they support the standard
  - As of today no HBA vendors support the standard
    - Ask the vendors to tell you their roadmaps and plans
- Both LTO and enterprise tape now support this protection
  - Currently many vendors are discussing what to do
    - Ask the tape vendor and software vendor how this could be supported
  - But questions exist in software

Review of requirements for system architecture for reliability

What is an architect to do

Step 1

Instrumental
HIGH PERFORMANCE INNOVATION

- Develop data collection framework
- You must monitor everything
  - Memory
  - Other things on system board
  - Channels
  - Switches
  - RAID
  - Disks
  - Tapes
  - Facilities
  - You have the picture

Step 2

- It is critical to track down system hardware soft errors
  - You must fix soft errors as soon as you can
    - Soft errors increase the probability that you will have an SDC
      - Multiple soft errors in the path increase the probability of SDC
        - » Stop the hardware, fail it over, fix it
          - » But do not allow a continuous soft error to be in the system
  - Ask your vendors about their monitoring
    - Disks
      - Both soft errors and failures of drives
        - » Remember RAID-6 is about to break
    - Tapes
      - Use either inband or out of band analysis
    - Memory
      - SBE are not your friend fix them as soon as possible
  - Consider global monitoring framework
      - Pennywise and pound foolish not to

Step 3

- Look for trends
  - Use tools for trend analysis on errors
    - Do this to reduce the error counts
      - Do you have a bad lot of memory or bad disk drive lot?
      - I am not saying this is the only answer, but a tool like Splunk to analyze all of the data is going to be required
        - » No human can do it
- What about facilities
  - Heat can cause bad things to happen
    - Especially for and connector (pins, cables etc)
  - Monitor your temperatures!
- Human resources are going to be required
  - If an SDC is found there needs to be serious post mortem
    - Like what was done at Netflix

Example of SDC

- A PC Magazine article (published August 25, 2008) reported a real-life data corruption incident:
  - "Netflix monitors flagged a database corruption problem in its shipping system. Over the course of the day, we began experiencing similar problems in peripheral databases until our shopping system went down." The root cause was determined to be a faulty hardware component, but the problem was that the component "reported no detectable errors."
    - I know for a fact no hard errors were reported but soft errors were reported

Thank you!

Thanks for listening