Aalto University

Storage in Astronomy

Beyond the limits of bandwidth, capacity, and location

Ari Mujunen
Aalto University Metsähovi Radio Observatory
ASTRON-JIVE Colloquium 17-Feb-2011 in Dwingeloo

NEXPReS Research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007- 2013) under grant agreement n° RI-261525. This presentation reflects only the author's views. The European Union is not liable for any use that may be made of the information contained therein.

Data Explosion of New Instruments

- Direct digitization of wide bands
  - Multi-Gbps (e-)VLBI, LOFAR, SKA,...
  - Leads to high-bandwidth streaming of high-volume data
- (Almost) continuous operation
- Geographically distributed
- Desire to post-process raw data multiple times

View of a typical future observatory control room? -->

Development of Storage Subsystems

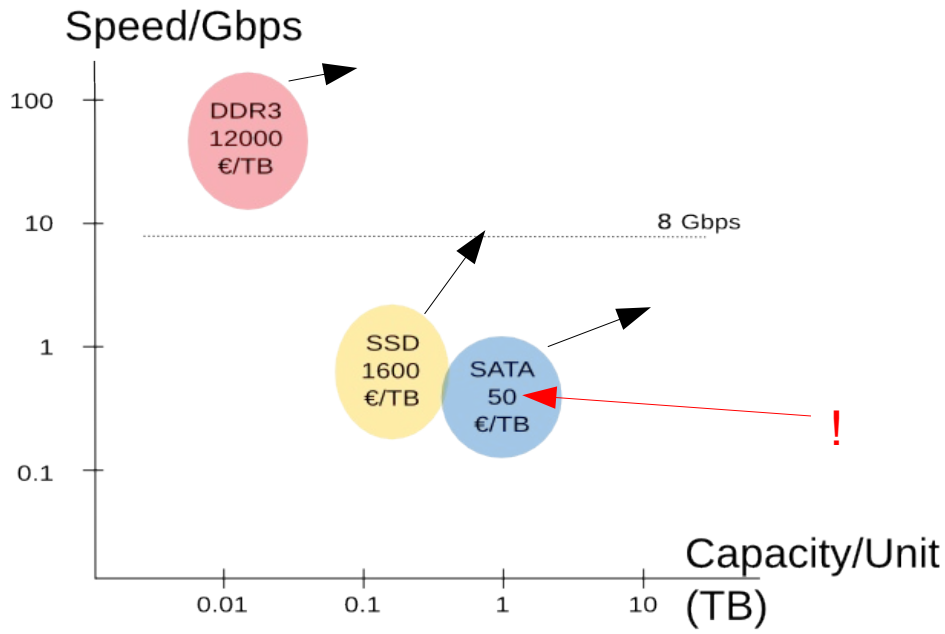
- Slow but steady increase in storage density
  - Not only more data in less physical space but...
  - ...also, implicitly, faster streaming capability
  - 2001: 130–270Mbps/disk, 2011: 400–1200Mbps/disk
- 3.5” drives, 2.5” drives, Solid-State Disks (SSDs),...
- Faster controllers, faster SATA I/II/III interfaces
- More and faster PCI Express connectivity in PCs
- Faster memory subsystems
  - Up to 30–60Gbps of real-world memory transfer bandwidth
- More and faster CPU cores to do the data transfers

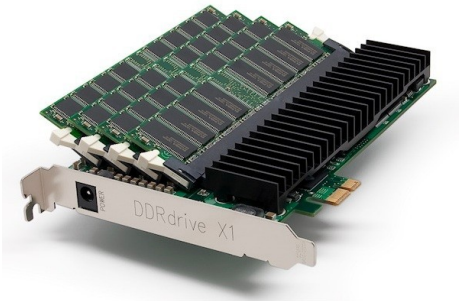
Why Disks? Why 3.5” Disks?

- For the foreseeable (near-term) future:
  - SSDs are at least 15x (15–50x) more expensive than 3.5” disks for the same capacity
    - But prices are dropping rapidly
    - Only 2–4 times faster than 3.5” disks
    - Can do simultaneous read and write (at ~half rate!)
  - 2.5” disks are roughly 2–2.2x more expensive than 3.5” disks for the same capacity
    - Only slightly slower in streaming speed
    - Much more compact when trying to reach top speeds with large number of disks operating in parallel

Hard Disk/SSD/RAM/Exotic

- Speed/Capacity/Price Comparison





The Dark Side of Hard Disks...

- Reliability specifications give perplexing and contradicting predictions
  - Unrecoverable error in 1E14–1E15 bits read
    - Read a 2TB disk only 6–62 times through and get an error?!?
  - MTBF 300000–1000000h, Annual Failure Rate 0.88–2.93%
    - Works for 34–114 years before failing, on the average?!?
- Statistical studies (CMU 2006, Google 2009) suggest that the real AFR lies between 2–6(–25!)%
  - Run 100 hard disks and each year 2–6 of them will blow up
- Any attempt to use a large number of hard disks must be prepared for their constant-rate random failures

Hard Disk Peculiarities

- Disks are usually perceived as deterministic devices... whereas a computer with complex multitasking firmware would be a more appropriate classification today
  - reallocated blocks, retries, recals → unpredictable performance
- Environmental sensitivity
  - power supplies, temperature, vibration, air pressure
- Outer tracks accommodate more bits than inner tracks
  - with constant rpm, start of is faster than end of disk
- Failures detected only after retries & long timeouts

Which Manufacturers Make Crappy Disks?

- All of them do—occasionally!
  - Seagate, Western Digital, Samsung, Hitachi/IBM, all have had their share
- Once you find out, it will be too late
  - The best approach is to use disks from all manufacturers
    - The probability to encounter a bad batch gets divided by the number of manufacturers / models you have

Balancing Subsystems (Disks ↔ Net)

- For the highest capacity and bandwidth at the lowest cost, 3.5” hard disks are still (for a while) the best match
- For quick bandwidth determination:
  - One 3.5” disk can sustain 400Mbps
  - Will want 3 disks to sustain guaranteed 1024Mbps
  - With 1GE, one 1GE network port per 2 disks would match
    - So 6 disks and 3 1GE ports will do guaranteed 2048Mbps
  - To fill one 10GE port you might need up to 30 disks
    - But 20 disks should sustain 8000Mbps
    - So the appropriate number of disks for 10GE ports is something between 20—30 disks

Balancing the Rest of the Subsystems

- Memory bandwidth, (PCIe) bus bandwidth, CPU cores
- During early (~2002) we realized that for a single 1GE with ~4 disks
  - A single ~1—2GHz core
  - ~512MB of <1GB/s memory is enough
- Later (~2008) tests with 4Gbps (via one 10GE) needed
  - Four ~2GHz cores
  - ~2GB of ~2GB/s memory
- Expecting 8Gbps via 10GE to need
  - 4–6 ~3GHz cores, ~16GB of 4–8GB/s memory

Commercial Storage Solutions

- Disk frames for standard (rack) enclosures
  - Low density at 3—5 disks per each rack 1U
- Rack enclosures with backplanes and front-mounted disks
  - Unique tray designs with med density (5dr/1U)
- SAN/NAS systems
  - 10Gbps new, streaming speed avg, €€€
- All are relatively expensive compared to the price of disks they host

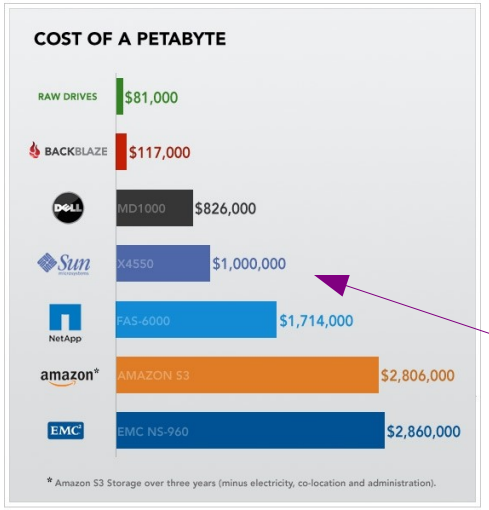
Lian Li EX-34H

Chenbro RM31616M-G

Dell EqualLogic PS6010E

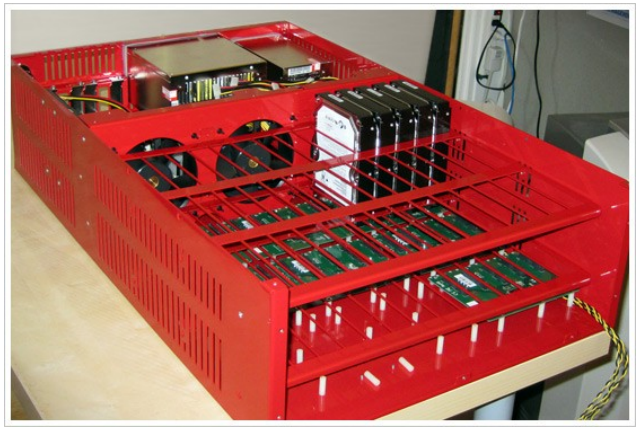
High Density x86 Storage Servers

- E.g. 4U high Sun Fire X4500, X4540, X4550 servers
  - Best rack disk density in industry at 12 disks per each 1U
  - Slightly unbalanced: too many disks to match the x86 power
  - Discontinued now after Oracle acquired Sun...



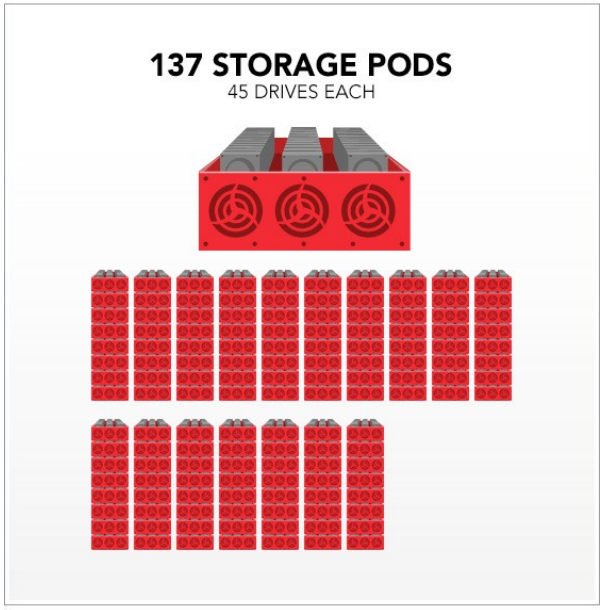
Servers Almost at Cost of Disks

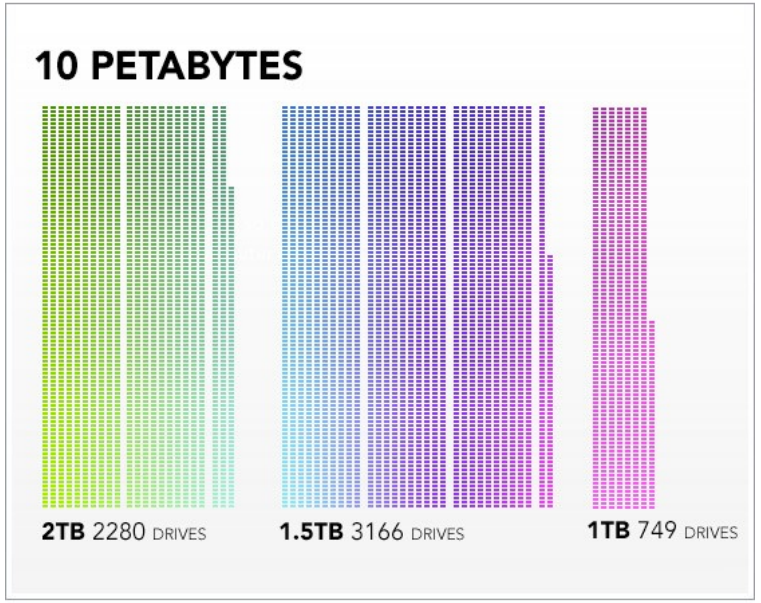
- E.g. Backblaze “Storage Pod”
  - 45 disks, 2 regular PSUs, 6 fans, and a crappy PC in a semicustom “ProtoCase-made” 4U rack box
- The hw foundation of a commercial backup service provider company, Backblaze Inc.
- Even more unbalanced than Sun Fire X4540 for high-speed streaming---aims at max capacity



Nevertheless...

- BackBlaze Inc. claims that they have been able to sustain a profitable business based on these steel boxes
  - And in late 2010 they have 10PB of storage online (at only ~1M$ cost)





Balanced High Density and High Speed

- Based on the previous balance calculations (and physical constraints) we are devising a “2011–2013 vintage” reference COTS implementation of 8Gbps capable 28 disk storage unit in a standard 4U rack enclosure
  - 56–84TB or 15–23 hours of 8Gbps
  - Put in 28 disks “Backblaze-style”, vibration-dampened and easily replaceable in front of enclosure
  - Add a heavy-duty 6-core MB with 10GE and disk controllers
  - Cf. the recent Mark6 proposal

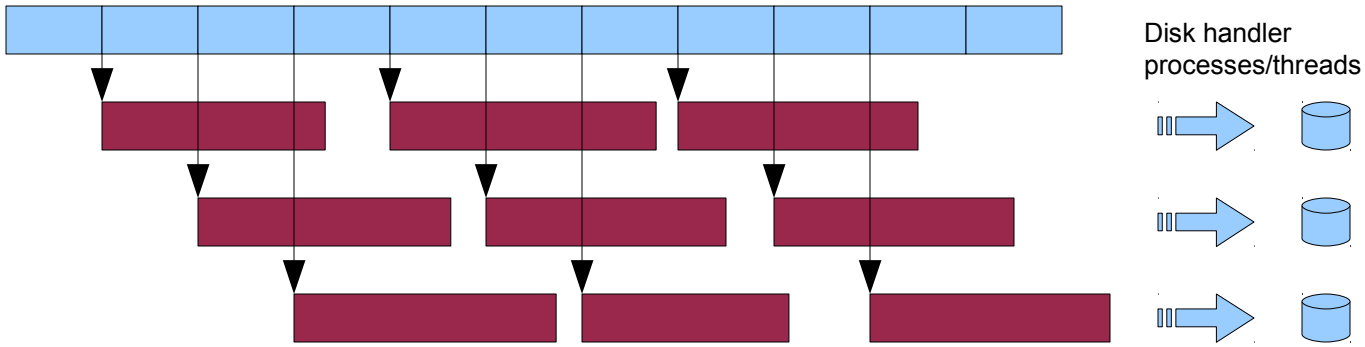
Chieftec UNC-410F-B

RAID Arrays?

- For high bandwidth and high capacity at the same time, one cannot really use other RAIDs but raid0 (striping)
- Raid5/Raid6 slow down in streaming writing, especially in “degraded” mode (with failed disks)
  - False illusion of guard against disk failures
- Cannot afford losing half the disk space with raid10
- But raid0 is vulnerable to single-disk failures
- Why use any RAIDs at all for astronomy data?
  - Why use any form of redundancy for volatile, non-archival data?

Writing to a set of disks

- Divide incoming data stream into consecutive largish (50—500MB) chunks of memory buffer
- Write chunks into files on a set of “stand-alone” disks
  - Local—or even remote disks

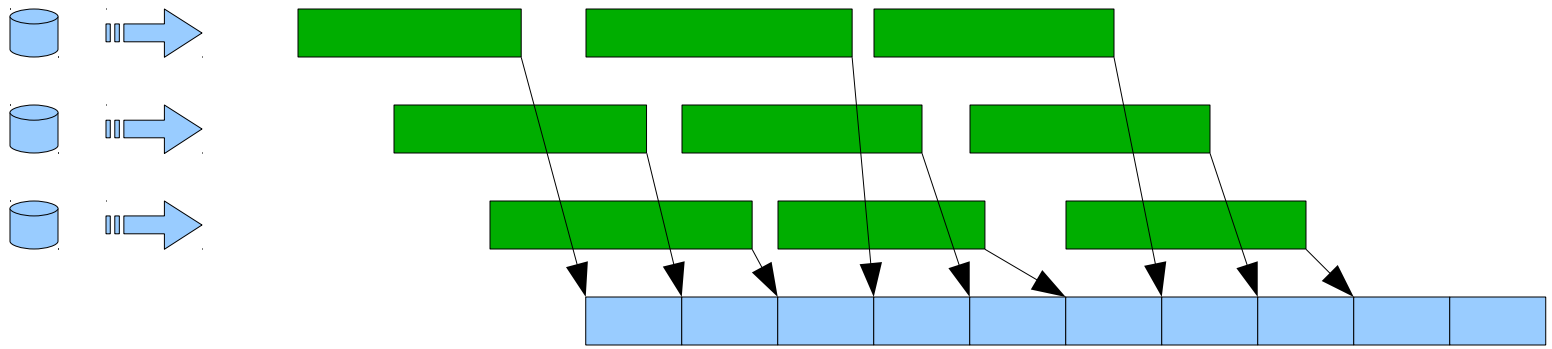


Minimizing CPU Usage & Memory Bandwidth

- At 10Gbps rates, avoiding unnecessary memory copies becomes essential to avoid exhausting CPU & memory
  - Current memory bandwidth is only 30–60Gbps in total
- The standard Linux disk r/w model makes one extra memcpy() between user mode and kernel buffer cache
  - Must use O_DIRECT with large enough transfer sizes to allow disk drives to queue large enough DMA request / NCQ queues
- By default the network stack makes another memcpy() when adding/removing IP packet headers
  - Can be circumvented with splice() or raw packet ring buffer

Reading from a set of disks

- Initiate preloading of memory buffers from a set of disks
  - Once all disks in a set have had a chance to preload, consuming the streaming data can commence at full rate

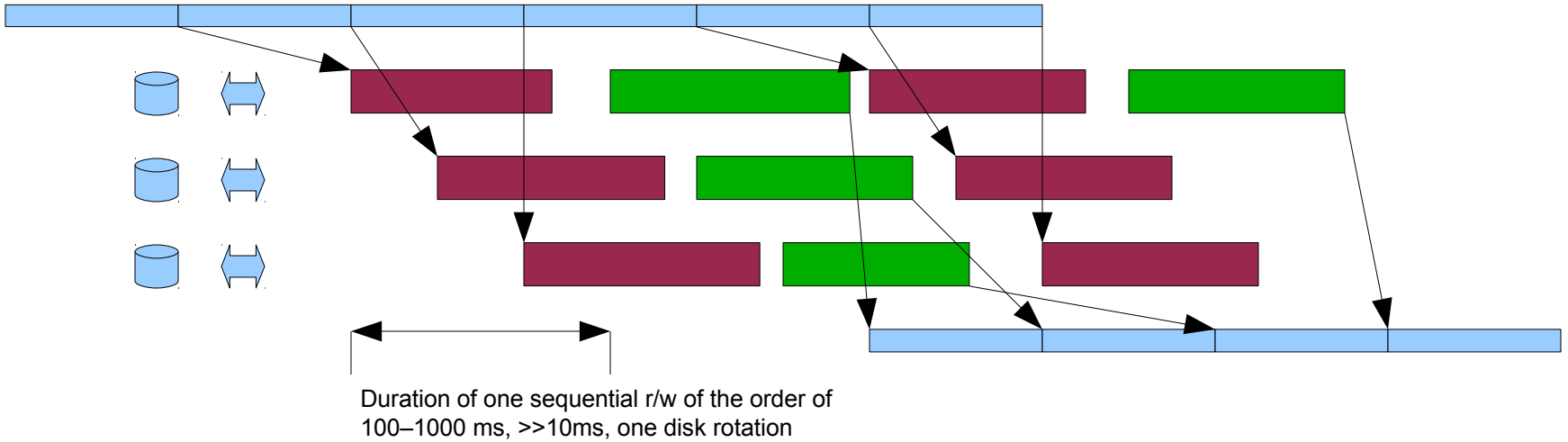


Remote Disks

- The easiest concept to add remote disks in the scenario just described is to have the memory buffer chunks “remoted” with a UDP-based protocol (derivative of Tsunami)
  - Even TCP would be fine for low-latency networks, but the easiest way to achieve high throughput regardless of network latency is to use a long delay between initial bulk packet transmission (via UDP) and the retransmission request(s) of missing packets
- The recording computers can just write to a set of disks located in the local and/or remote network

Simultaneous Read and Write

- Two problems, can be solved with interleaving:
  - HDD seek time, slows down using more than one “spot” of disk
  - Even without, double data streaming bandwidth required throughout the internal data paths



Simultaneous Read and Write

- Avoiding hard disk 8–12ms seek times
  - Must stream in either read or write direction for much longer than seek time (or full disk rotation time, approx. 8.3+1.1ms)
    - E.g. read streaming for about 120ms gives the data from 10 rotations, then wastes the time of only one rotation, 90% efficiency
  - So nothing really fancy is required **but:**
- Usually simultaneous read and write will halve the total bandwidth
  - If the bottleneck at one subsystem can be removed (e.g. hard disks → SSDs), another bottleneck will surface in another subsystem
  - Memory and/or disk controller (PCIe) bandwidth limits are the most probable bottleneck candidates

Can SSDs Do Anything Useful?

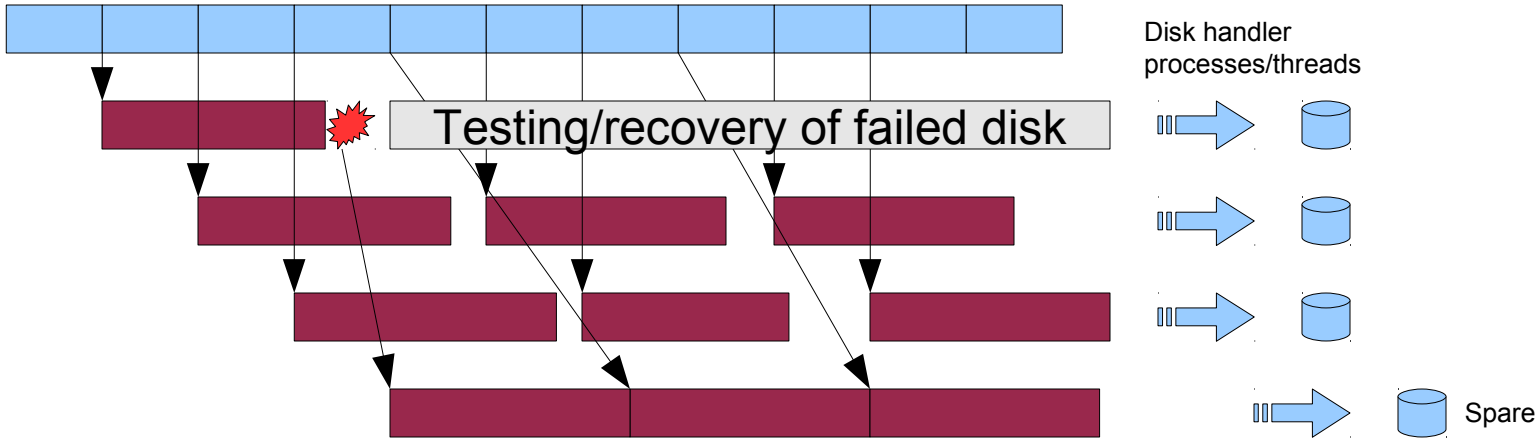
- On paper, SSDs sound like the “dream FIFO buffer” in front of a set of hard disks
- In reality, one cannot (yet) afford large enough capacity to do anything interesting with the SSD “front buffer”
- It would be great to replace all hard disks with SSDs if not 15–30–50x more expensive for the same capacity
  - Some day in the future they will be low-cost enough to replace many(or most of) magnetic hard disks
  - Though hard disk mechanical unreliability will be replaced with unreliability characteristics of electronics
    - Long-term archival reliability remains to be seen...

A Desperate Scenario for SSDs...

- An imaginable scenario with SSDs for two scans:
  - Record a scan & transmit in real-time
  - If scan realtime processing is ok before start of next scan, skip copying to a local disk buffer
  - If not ok, copy to a local disk buffer while simultaneously recording a second scan
    - But what can you do with the “saved” disk buffer space—you need to be prepared for all scans to go to disk buffer anyway..?
- SSDs benefit most in high-seek scenarios
  - Which astronomy streaming applications rarely have...
- Any simultaneous r/w scenario where you cannot write directly to the destination where SSD-read data is going?

Error Recovery

- If one disk handler does not complete in allotted time, it can hand over the memory block to another hander
  - A spare disk
  - Or even to another disk in the current set, provided there is enough “slack time” in the set, so it can eventually catch up



Read Error Recovery

- Failure to read a chunk file from one disk causes the loss of a well-defined (0.1–1sec) time interval in the data stream
  - In worst case (a total disk failure) the loss repeats at n*(0.1–1s) intervals
  - Can adjust read retry strategy according to whether the reader/consumer can wait or it need quasi-realtime data

Turning Recordings into Archives

- If some time after a recording is made it is found that
  - It needs to be retained/archived for a longer period of time
  - And there is spare disk capacity available
  - And there is some idle time for the data disk set available
- Then redundancy (aka raid5/raid6) can be calculated afterwards (file based) and the ECC information stored onto the spare disks
- If later a disk of the disk set fails, it can be replaced and its content files restored by recovering from the ECC files

“The Inconvenient Truths” :-)

- About networked data streaming:
  - “A given station cannot really sustain recording bandwidth larger than their network connectivity—unless given an unlimited disk buffer or long enough breaks between recordings.”
  - For the VLBI case: “A single slow (or high-latency, like shipping) connection in a given e-VLBI network will force others (or some buffering party) to buffer most of the VLBI data of the whole network, if not all.”
- About buffers and archives:
  - “Huge disk buffers with thousands of disks (whether distributed or centralized) will cost a fortune, age rapidly, and be fragile (even with the highest-end equipment) and in constant need of (hw) maintenance.”