Enterprise SSDs in Scale-Out NAS Design

Rob Peglar

CTO, Americas

EMC Isilon

A Blast from the Past – Happy 31st! August 1981

IBM Model 5150 Specifications

Processor	Intel 8088
Speed	4.77 MHz
RAM	16KB
Storage	Cassette Tape, optionally 5.25" 160KB floppy drives
Expansion	5 expansion slots
Bus	Industry Standard Architecture (ISA)
Video	Initially CGA (320x200x16 color, 640x200x2 color) or monochrome (80x25 text only))
I/O	Parallel, Serial
OS	Microsoft Basic 1 (ROM)
Killer App	VisiCalc

Fast Forward – to 2012

- Today, we have CPUs which are ~1,000x
  - 1’s of GHz clocks instead of 1’s of MHz
- Today, we have RAM which is 10,000,000x
  - 100’s of GB instead of 10’s of KB
  - In some cases, 100,000,000x (1’s of TB)
- Today, we have storage which is ~20,000,000x
  - 3 TB per drive instead of 160 MB
- So what’ s the problem?

The Problem WAS – AND IS – I/O

- In a perfect world, I/O would not be necessary
  - 1st level store would hold everything, forever
- Access Density – IOPS/GB
  - Getting WORSE over time for rotating magnetic
  - Will it get worse over time for non-rotating SSD?
- Example:
  - IBM Model 5150 – 625 KB/s, 8.33 ms, 3,600 RPM
    - – IOPS/GB = 20 / .001 = **20,000**
  - Today – 170,000 KB/s, 2.9 ms, 15,000 RPM
    - – IOPS/GB = 200 / 300 = **0.667**

The Advent of Solid State Disk

- Technology Choices – Boiled Down to Two
- NAND Flash
  - Slow (writes), cheap, dense, non-volatile
  - JFFSx
  - ONFI
  - Next up – Phase Change Memory (PCM), ReRAM, Spintronics, (?)
  - Is it cache, or is it disk?
- DRAM
  - Very fast, dense, not cheap, volatile
  - No internal file system
  - Is it cache, or is it disk?

NAS Thoughts

- Filesystems don’t want disks
  - They want space (more is better)
- Filesystems don’t want IOPS
  - They want time (less is better)
- Filesystems do block I/O because they *have* to
  - But they don’t really *want* to
- Where to use SSD in NAS?
  - Backing store for file data?
  - Read cache for files?
  - Write cache for files?
  - Backing store for metadata?
- Answer – think about the way filesystems work

Real-World NAS Workloads

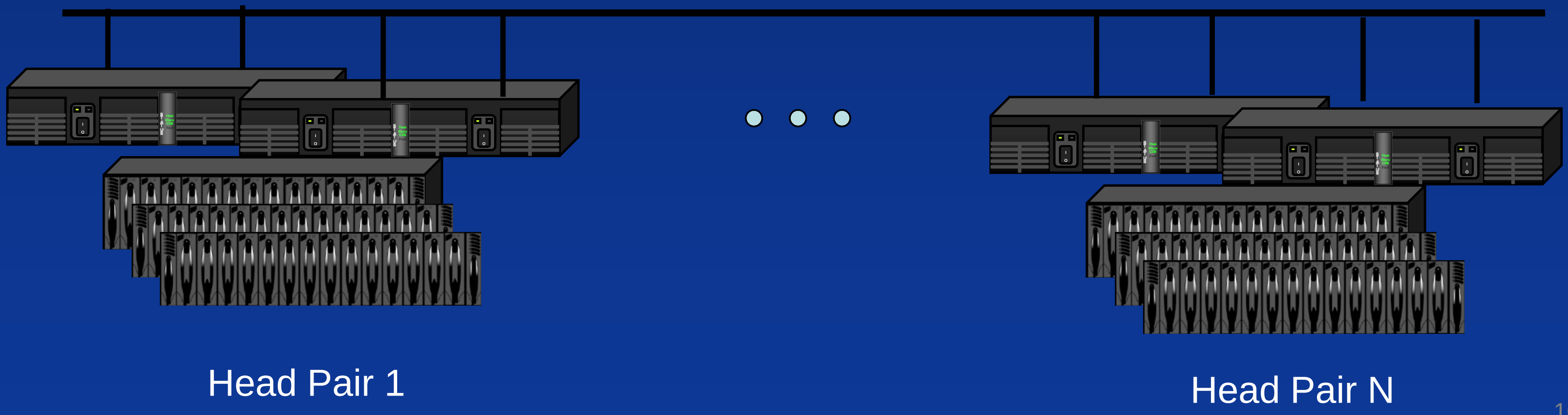
- Unstructured data
  - Huge collections of files (10s of billions)
  - Streaming ingest from disparate sources
  - Parallel access – both read and write
  - Fast enumeration of metadata is critical (save time)
  - Automated tiering is critical (save $)
- Structured data
  - So 2011
- Filesystems have key elements
  - It’s all about the metadata!
  - Scale-out metadata over scale-out data

How should I Design my NAS for SSD?

- SSD introduces a new complexity into NAS
  - Or does it?
- SSD <u>should be used to store metadata</u>
- SSD as a file data cache?
  - Read cache – nope
    - Random read workloads – too many cache misses at scale, wasted resource, pathologic use case
    - Sequential read workloads – might as well do aggressive read-ahead on HDD into DRAM, streaming
  - Write cache – nope
    - Flash is too slow, at least the type NAS users will pay for
    - Poor match with wear characteristics
- That leaves metadata – which is a perfect fit

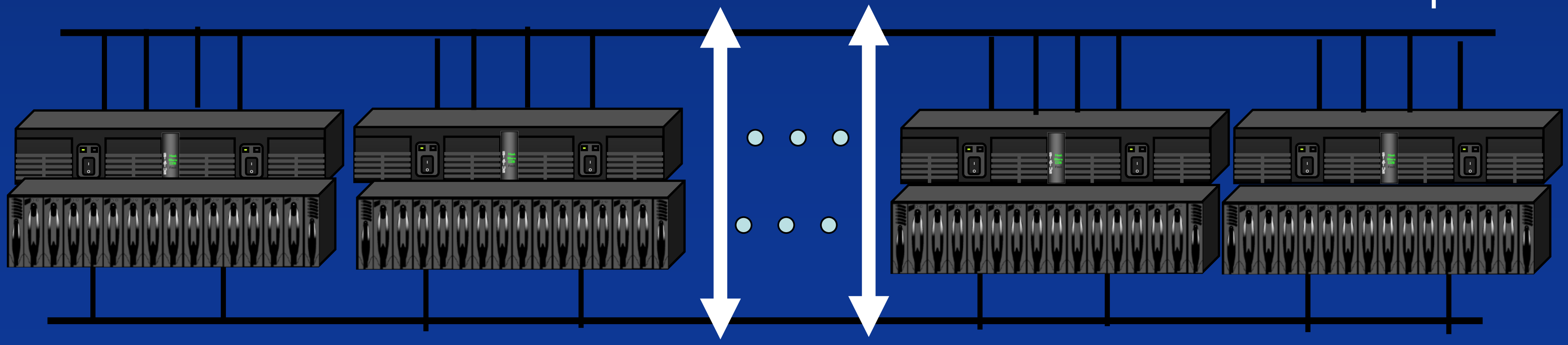
Two Types of NAS Storage

- Type 1 - Dual-access captive storage
  - Pairs of controller heads integrated with disk shelves
- Clustering is non-optimal
  - Flaw - Inter-head latency over enet to reach captive disks
  - Flaw – RAID groups & sparing across pairs (can't do it)
  - Flaw – multiple tiny filesystems, captive to head pairs



Two Types of NAS Storage

- Type 2 - Multi-access non-captive storage
  - N nodes networked with low-latency interconnect
    - – Client traffic on Ethernet, disk traffic on IB
- Clustering is optimal – scale CPUs & disks
  - Any-to-any communication to CPUs, RAM & disks
  - Single filesystem, single namespace, file-level ECC
  - Metadata in SSD – fast enumeration, attribute retrieval
    - – You want to enumerate 10 billion files from SATA HDD? Nope!



THANK YOU

Q&A