Considerations for Exascale Filesystems

Paul Nowoczynski
Pittsburgh Supercomputing Center
pauln@psc.edu

Increasing Efficiency from Storage Components for Parallel I/O

Being on the wrong side of Moore's Law makes our work challenging

- Several sites already have TB/s of disk bandwidth on their floors today
- ***Do the math*** – At large scale today's storage systems achieve a only fraction of their maximum sustainable rate ~(20% - 30%)
  - Inconsistent media rates (outer vs. inner cylinders is a factor of 2X)
  - Backing filesystems and RAID layers add to the problem
- Moore's Law is a tough thing to fight.. @10's TB/s such inefficiency is unaffordable!
- For checkpoint workloads this penalty can be avoided
  - CP workloads are inherently bursty – maximum performance is only needed for brief period
  - I/O systems should be capable of achieving peak rates for short amounts of times followed by a restoration period (***i.e. a cheetah***)
- Some pieces of the puzzle already exist
  - log structuring, copy-on-write

Spindles?? What about Flash?

- @10's TB/s, systems like *Fusion-IO* will likely be a necessity
  - Interspersing SSDs into the machine fabric leverages the supercomputer's interconnect.
- The (potential) hurdles
  - Price: @~2k/GB/s the price is still largely prohibitive
  - Interspersing amongst compute nodes increases access and fault tolerance complexity for the filesystem
    - Making copies is not as simple as it seems
    - Asynchronous draining may introduce jitter and/or interfere with fine grain collective operations
    - What are the expected filesystem semantics regarding unmigrated data?
  - Still require a large spindle based store – how much bandwidth is required here?
    - Remember a 60TB/s requires (optimistically) ~50k spindles
  - Total storage system cost == ~200million!

In the end price dynamics / economies of scale will be the determining factor

I/O Node Performance Variability must be a Design Consideration

- Today's parallel I/O load balancing techniques are not suitable for wide, horizontal scaling
  - Assuming homogeneous performance characteristics across a large set of I/O nodes has sufficed but will soon become a losing hand
  - We can no longer get away with being as fast as the slowest node..
- Exascale I/O systems cannot allow global performance to be severely impacted by small sets of sick or oversubscribed components
- Today's methods for inferential metadata layouts are a large part of the problem
  - Succinctly solves the issue of metadata compactness in parallel filesystems .. *BUT* ..
  - Result in overly deterministic placement models which are difficult to load-balance in the face of slow or oversubscribed components
- Convenient layout models may need to be abandoned to enable techniques which can further increase I/O efficiency