WARP ® MECHANICS

Using Lustre/ZFS as an Erasure Code System

Technical Discussion for 2015 IWLE

Q1-2015 – Annapolis MD

Josh Judd, CTO

Overview

- Briefly, what is erasure code protection?
- Why would we want erasure code in a parallel FS?
- What's wrong with the non-Lustre approaches?
- Can a Lustre/ZFS-centric approach give the benefits without the pitfalls?

WARP MECHANICS®

What is erasure code?

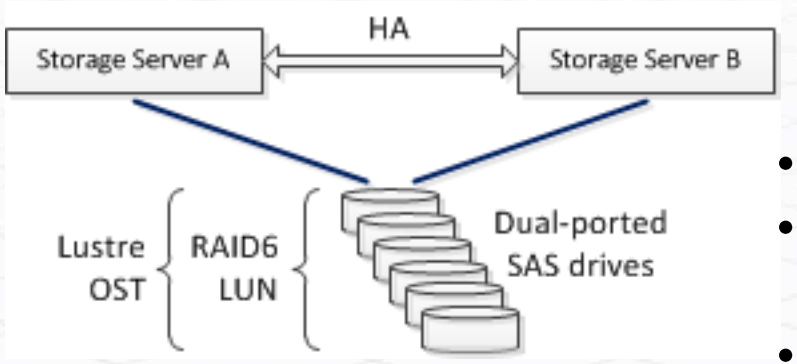
- Non-scientific answer: It's **not** the "death of RAID"; it *is* RAID... –ish.
- Classic RAID is a *form* of EC
- Instead of distributing redundancy across disks within a storage server like RAID, the EC method we now *call* "EC" distributes protection across servers

- Does Lustre do this? As Eric B. said yesterday... No. Maybe in the future.
- Replicas have been proposed, which is not space efficient – 3x cost in some cases
- "Longer term thing" to get to use "commodity" storage – "years not months"
- "Seven years to develop a RAID stack"

- This means we Lustre folk have a substantial cost delta vs. EC systems

RAID vs. EC

Classic Lustre: RAID with HA

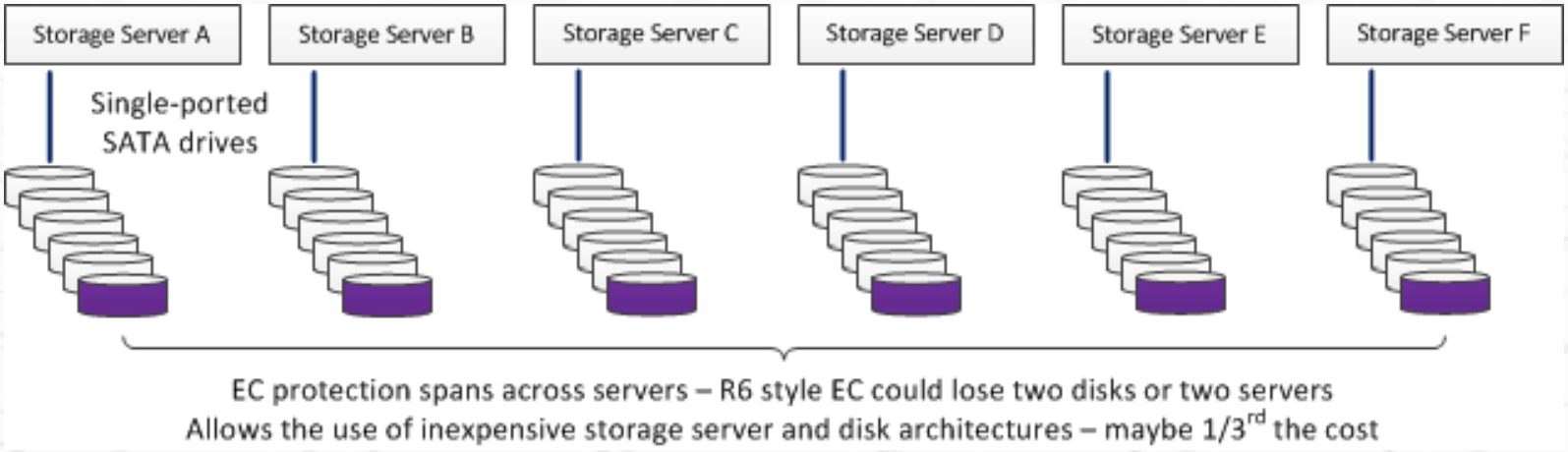


- OSSs are arranged in HA pairs
- Each node in a pair attaches to shared disks via SAS
- Disk failure handled by RAID
- Node failure handled by HA

WARP MECHANICS®

RAID vs. EC (cont.)

Erasure code: spread redundancy across nodes



- Theoretically could support same schemes as RAID
- In this example, you can lose any two disks *or* servers
- Advantage: cost and HA

WARP MECHANICS®

Current Situation

- Lustre is the incumbent parallel FS in HPC; we want it to stay that way
- ZFS under Lustre dramatically enhanced its capabilities and reliability
- Today, it requires SAS drives and HA controllers... >$ compared to EC

- Early-stage Object Storage systems exist (eg Swift, Ceph, Scality)
- These support *limited* Erasure Code (EC) protection methods (eg 2+1)
- That at least gets them... *close* to cost-parity with WARP's Lustre/ZFS
  - They *claim* savings, but get there with unrealistic configurations or misstatements
- Reliability is unproven, unlikely to scale, and known to be slow
- But if you want a *small slow* system, it's there today, and will improve

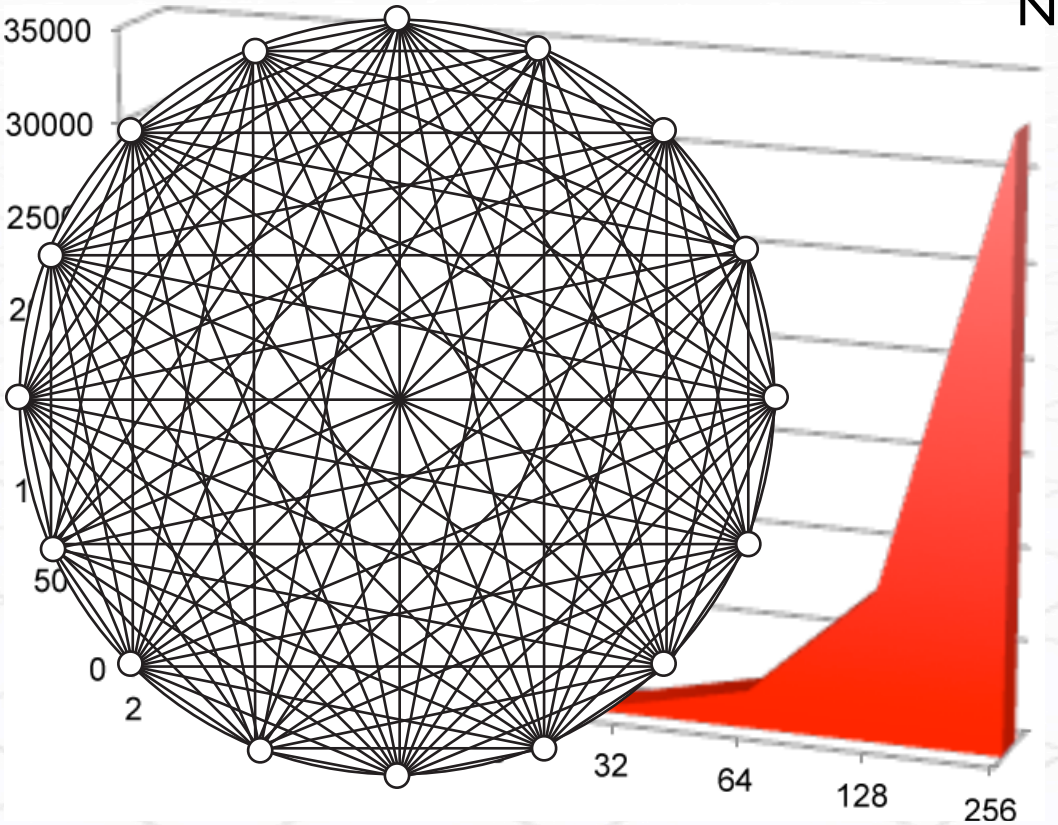
Object EC Systems' Design Flaw

- In "tightly coupled" clusters, each node needs to "know a lot" about the configuration and state of every other node
- This creates an exponential problem as the cluster grows
- If you add a node, you add one conversation *per other node* in the cluster
- This is known not to scale – Ethernet, Fibre Channel, etc.

- Systems like Isilon, Ceph, Swift, Scality etc didn't learn from past mistakes
- Lots of *marketing* success... But... not so much with the "actually working"
- Some of us need things that actually work ☺

WARP MECHANICS®

Exponential Cluster Scalability Issue

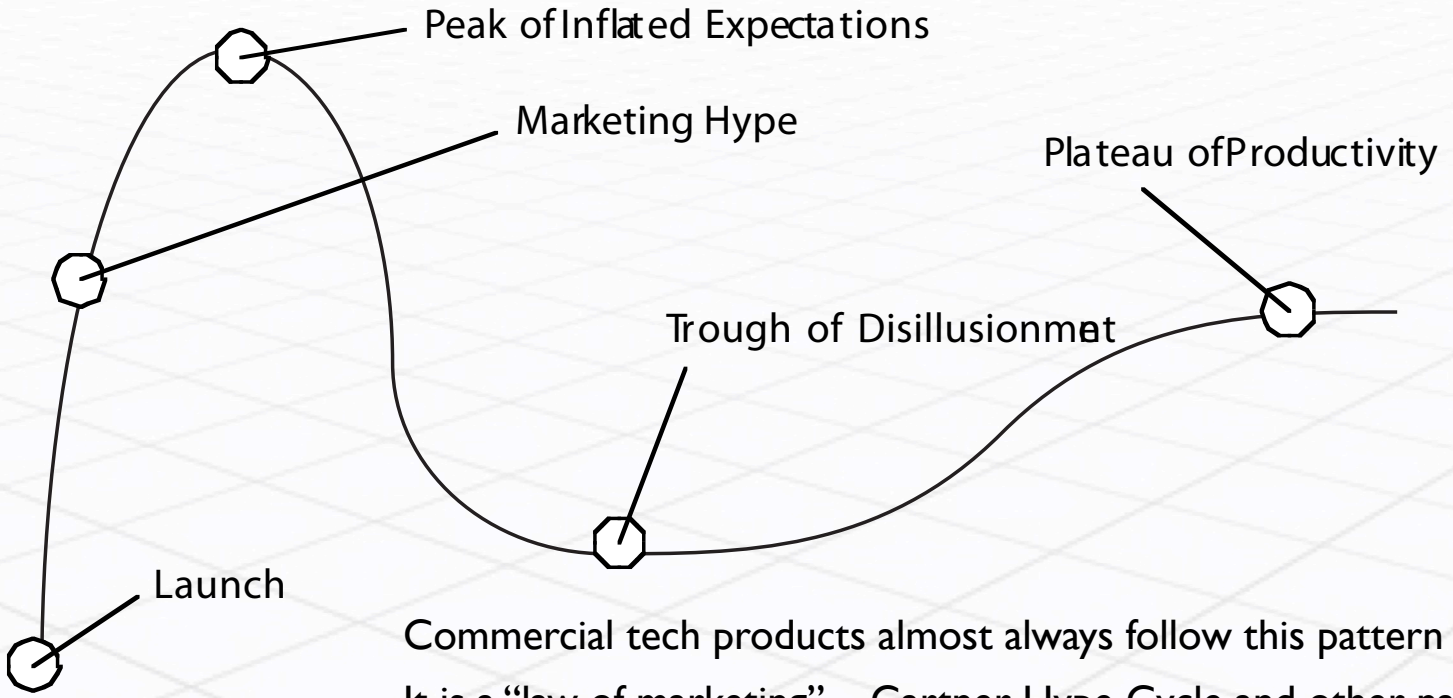


Nodes = Connections:

2n	=	1c
4n	=	6c
8n	=	28c
16n	=	120c
32n	=	496c
64n	=	2016c
128n	=	8128c
256n	=	32640c

WARP MECHANICS®

So why all the buzz? Hype cycle:



Commercial tech products almost always follow this pattern

It is a "law of marketing" – Gartner Hype Cycle and other names

WARP MECHANICS®

Solution = Same basic idea as Lustre itself

- What if you could get *real*, full-featured EC, with the "working" option?
- You'd get the cost of EC, better HA, and equal scale/performance

- Strength of Lustre is that it has layers to the FS horizontally and vertically
- Protection is in groups – not everybody to everybody
- Meta data is in layers – OSSs handle block layer internally

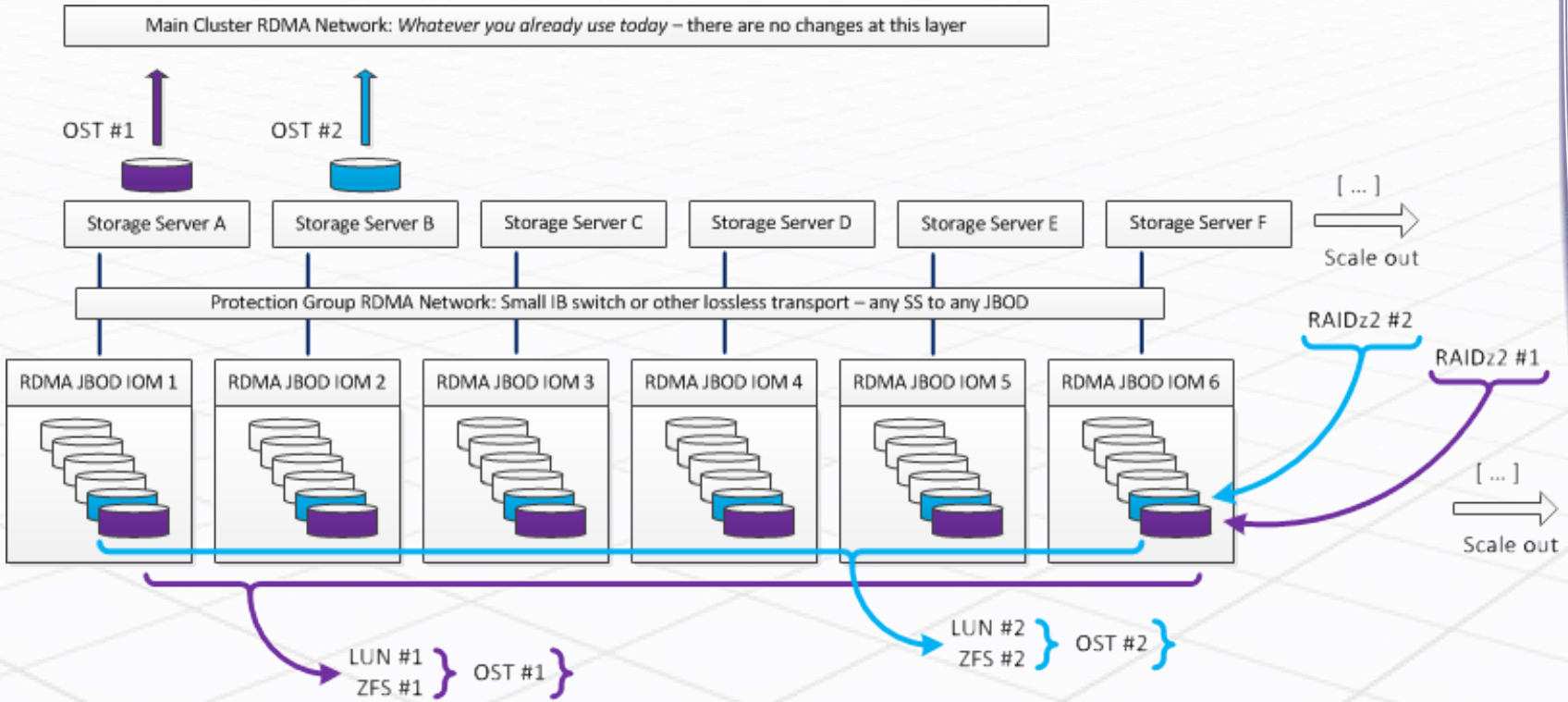
- So just don't have everybody talk to everybody else about everything
  - Avoids non-linear scalability issue
  - Allows leveraging *all* existing Lustre code

E.g.: WARP-Z Technical Walk-Through

- Lustre OSSs are classically arranged in 2-node HA "protection groups"
- This is *analogous* to a RAID-1 mirror of *controllers*
- WARP-Z can change that arbitrarily, using any ZFS-style scheme
- For example, it could use a 10-node group, with R6 style protection
- There could be any number of these "*n*"-node groups in the Lustre FS just as there can be any number of 2-node HA pairs in a current Lustre FS
- "Tight coupling" only occurs within the *manageably small* protection group
- Outside PGs, everything works and scales *exactly* as normal

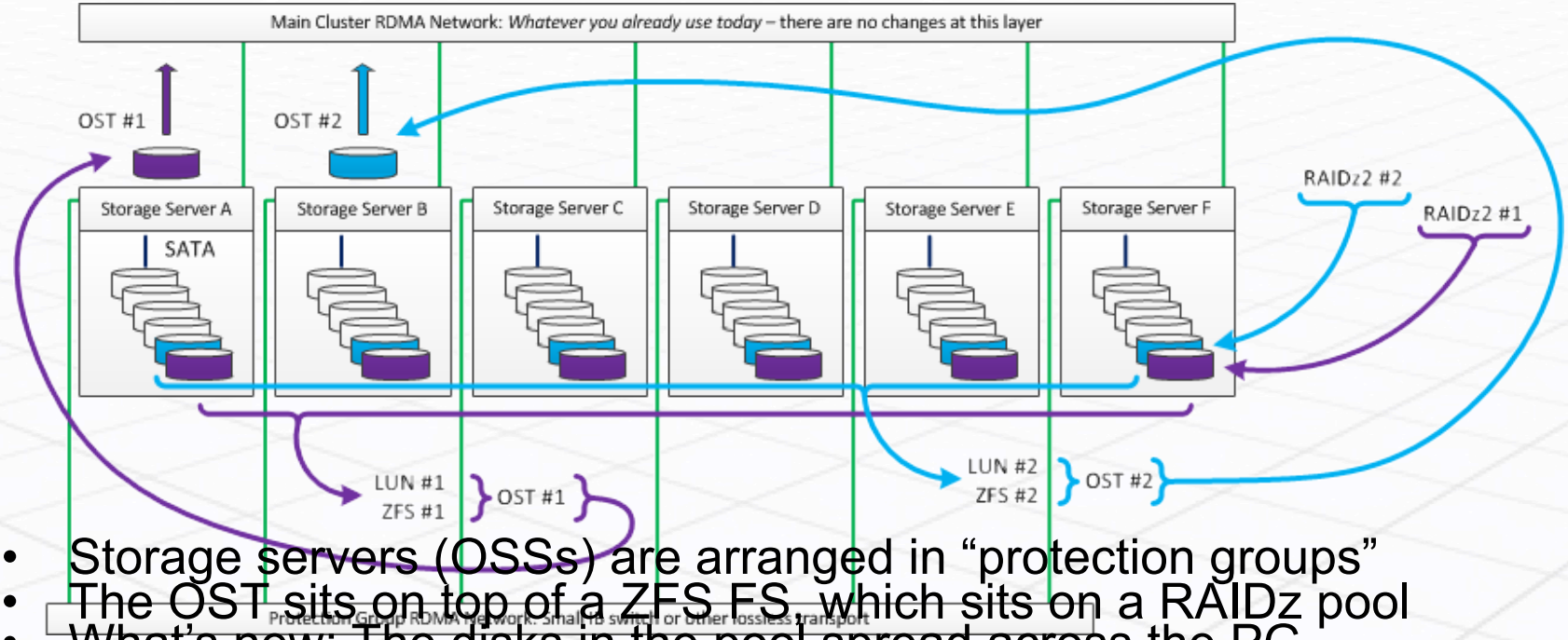
WARP MECHANICS®

WARP Mechanics' approach



WARP MECHANICS®

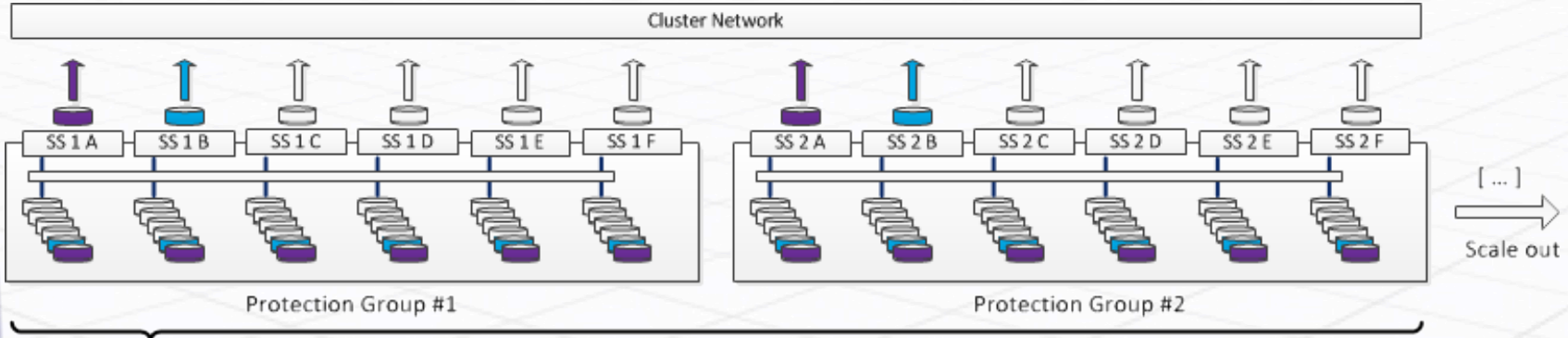
WARP Mechanics' approach (cont.)



- Storage servers (OSSs) are arranged in "protection groups"
- The OST sits on top of a ZFS FS, which sits on a RAIDz pool
- What's new: The disks in the pool spread across the PG
- These are just like the 2-node HA pairs, but can be any size
- Any given OST is "owned" by one OSS, but can shift to any other
- This works because the nodes share disks via RDMA

WARP MECHANICS®

WARP Mechanics' approach (cont.)



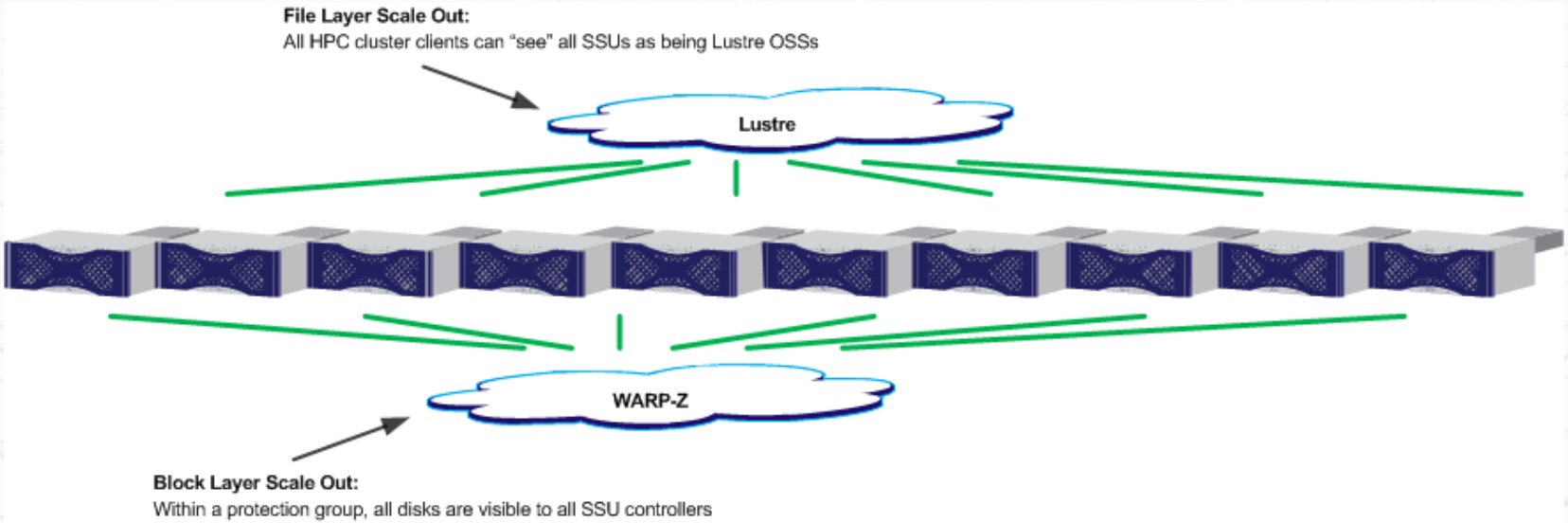
Scale out re: number of PGs is limited by *Lustre's* scalability. The PGs share nothing between them, so we don't get the exponential problem. Each tightly coupled group is a manageable size. Granted, it's bigger than the 2-node HA groups used today, but not so big that it hits the "hockey stick" part of the curve.

WARP MECHANICS®

WARP-Z HPC Storage System (cont.)

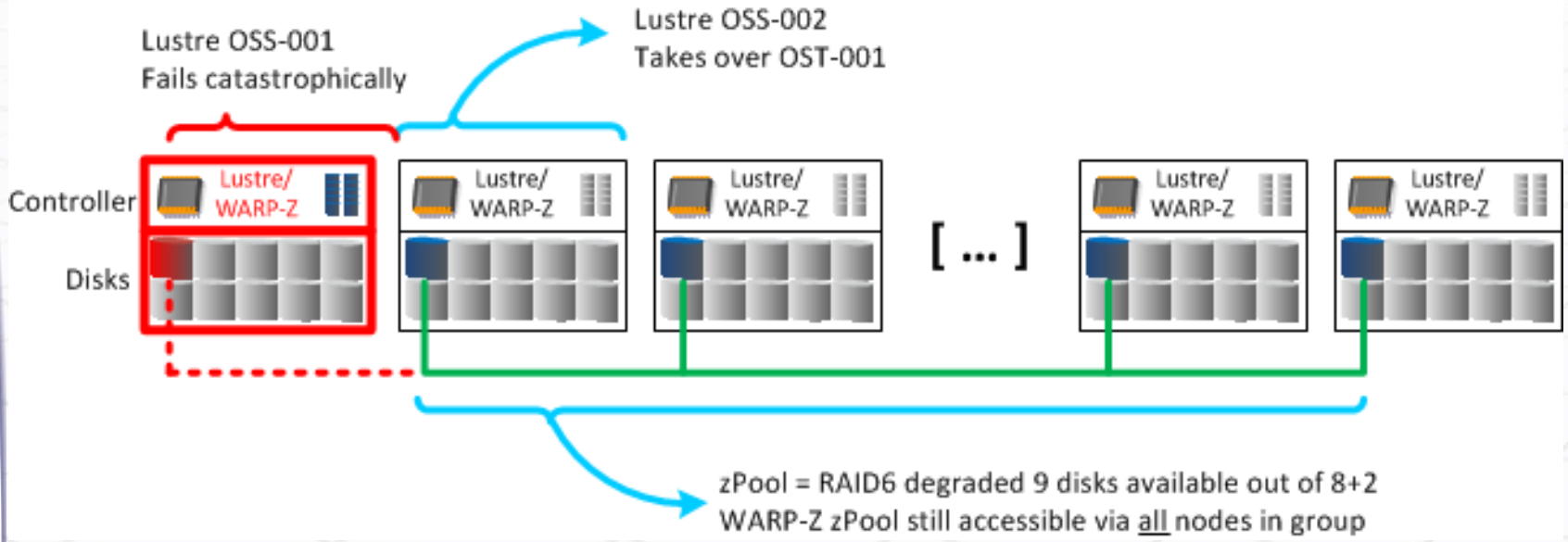
- WARP integrates ZFS and Lustre onto the same controller
- With WARP-Z, you don't need 2x redundancy for *OSSs*
- You can have a *parity* style of protection for the *storage cluster*
- If an OSS fails, you can fail its OSTs over to any OSS in the group
- If you integrate JBODs into OSSs, then at most you will lose 1x OSS and 1x disk from each RAID set – with RAIDz2 or z3 this is no problem
- Any of the remaining OSSs can import the pool and serve the OST!
- Either way, no more redundant controllers; no more SAS drives
- HA actually is *better* because you can lose *any* two, not just *specific* two

Failure Mode Example



WARP MECHANICS®

Failure Mode Example (cont.)



WARP MECHANICS®

State of the art on WARP-Z

- Developed last year as extension of WARP's multi-year Lustre/ZFS effort
- Demonstrated via "NDA" sessions at SC 14
- Oddly, all core functions worked "right off the bat"
- Gathered feedback from key industry players to assess viability
- Moving forward with productizing (management tools etc.)

- In the mean time...
- Let me know if you want collaborative early access
- Also: It's all open source, so feel free to DIY something similar

WARP ® MECHANICS

WARP-Z Lustre Architecture

Technical Discussion for 2015 IWLE

Q1-2015 – Annapolis MD

Josh Judd, CTO

Q&A

WARP: Company Milestones

- R&D began in 2008
- Incorporated in 2010 by Josh Judd – storage industry veteran from Brocade
- Engineering focus; not a marketing company
- Began production sales in 2011 – mostly Hollywood studios (e.g. Fox, Technicolor)
- Fortune 500 customers/partners – e.g., Hyve/Synnex and Sanmina-SCI
- Products now support many PBs of mission critical applications
- Began development efforts related to ZFS/Lustre products in 2011
- Lustre over ZFS HPC storage layer sales began in 2013 using Solaris ZFS; now use ZOL
- Direct relationship with "leadership" HPC accounts e.g. LLNL, ORNL, NCSA, & IU
- On track for $100M+ revenue over next CY
- Expected growth of 100+ FTE over next CY
- Rich ecosystem of upstream OEM-tier relationships – e.g. Intel, Western Digital, HGST
- VERY rapid growth curve at present – WARP is trending!