SCSI Referral

Adding Cluster Support to the SCSI Protocol

Ross Zwisler
Dr. Andrew J. Spry
LSI – Engenio Storage Group
Advanced Development
SYSTOR 09 – May 4-6, 2009

Outline

- Block Storage Clustering
- SCSI Referral
- Prototype and Performance
- Conclusion

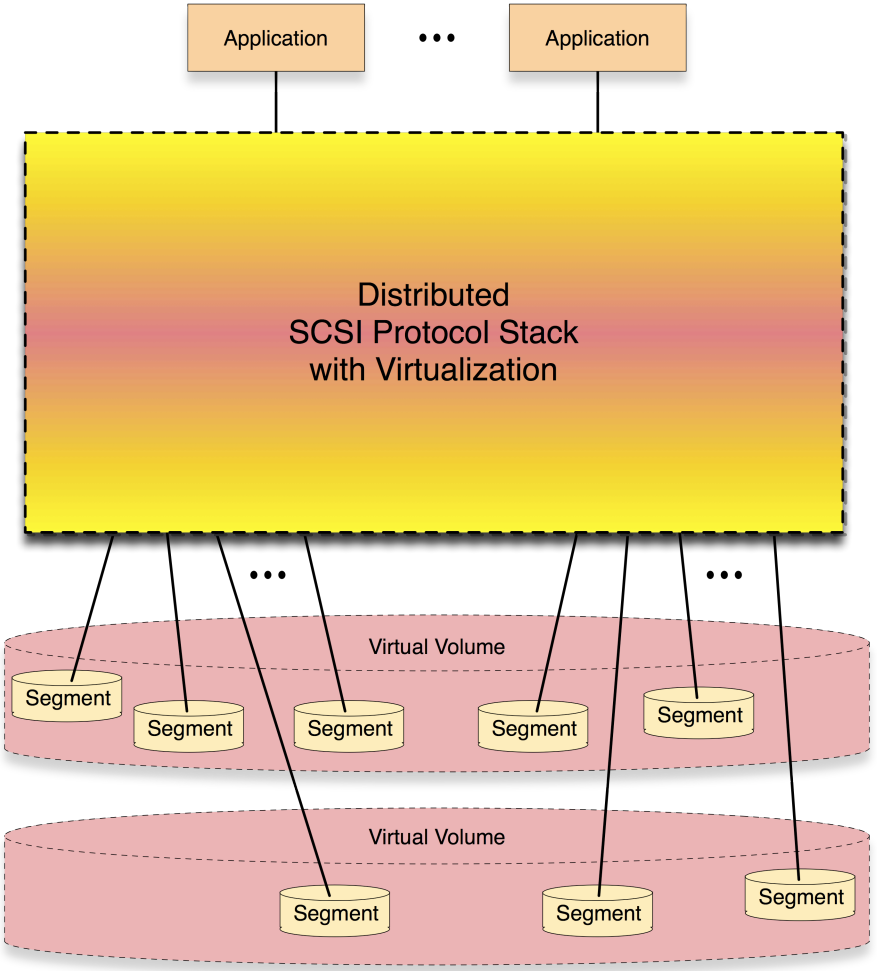
Block Storage Clustering

Block Storage Clustering – Why Bother?

- Capacity
  - Applications need volumes larger than a single storage system
  - Concatenating small volumes into larger capacity volumes
- Performance
  - Applications need data access faster than single storage systems support
  - Stripe across multiple disks and storage devices
- Expandability
  - Storage capacity and performance need to grow with application demands
  - Pay as you grow cluster from 1 node to N nodes
- Affordability
  - Need to build clusters from lower-cost / smaller systems
  - Minimize extra hardware required for cluster support

Block Storage Clustering – 10k Meter Perspective

- Architecture
  - Applications
  - Distributed SCSI Protocol Stack
  - Virtualized Volumes
- Variations
  - Initiator based virtualization
    - Left Hand Networks
    - Linux Logical Volume Manager (LVM)
  - Network based virtualization
    - LSI's StoreAge Data Path Module (DPM)
    - IBM's SAN Volume Controller
  - Target based virtualization
    - Dell's EqualLogic
    - IBRIX Fusion
  - See paper for Pros/Cons

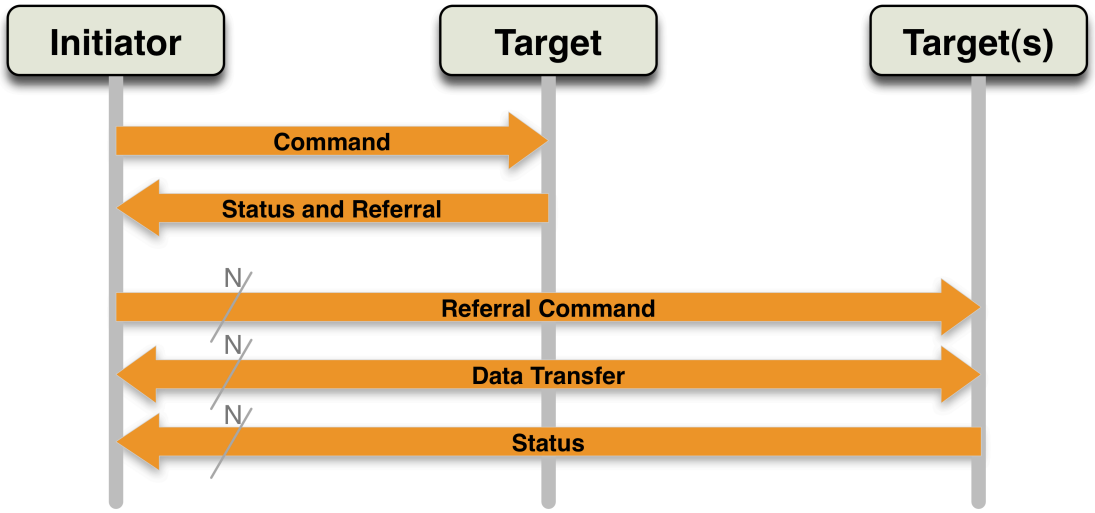


LSI™

SCSI Referral

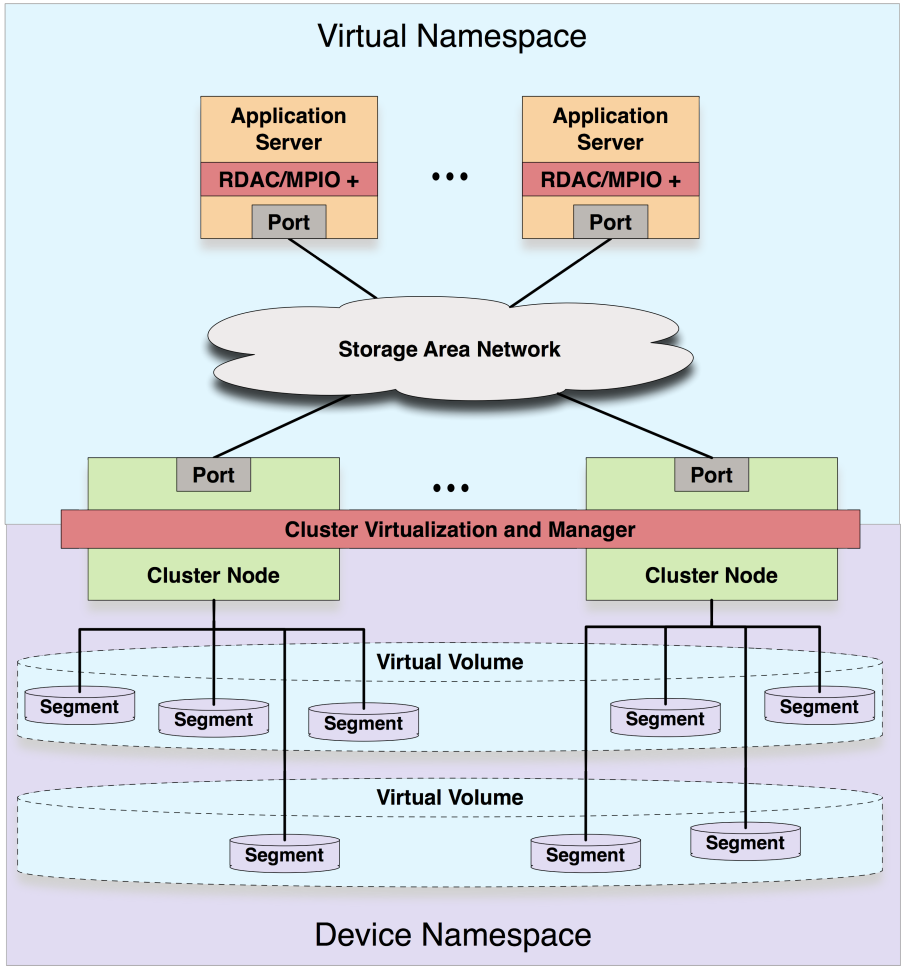
SCSI Referral – How It Works

- Extend T10 Command – Response Protocol
  - When all data available at SCSI port (cluster node):
    - Operate as current T10
  - When all data is not available at SCSI port (cluster node):
    - Target returns immediate response with referral information
      - Each referral includes: **Port Identifier, Offset, Length**
    - Initiator uses descendent **Referral Commands** to access data
    - Initiator gathers descendent command response



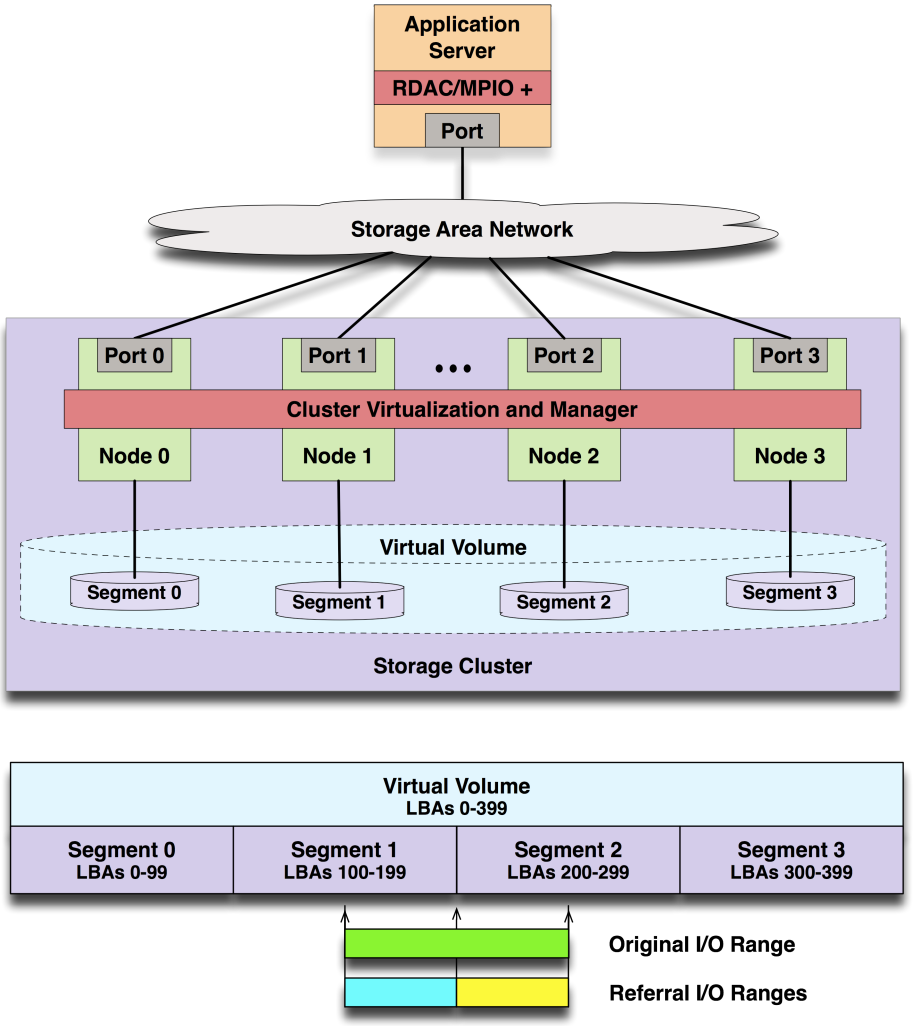
SCSI Referral – Architecture

- Single Device Identifier
  - Each port has a unique identifier
  - Cluster is a single multi-ported device
- Cluster Nodes Are Equal Peers
  - Initiator accesses data on any port
- Unified LUN Namespace
  - Nodes present same LUNs
  - LBA map known by all cluster nodes
- Initiator Handles Referral Response
  - Redirects commands to get data
  - Optionally caches referral information



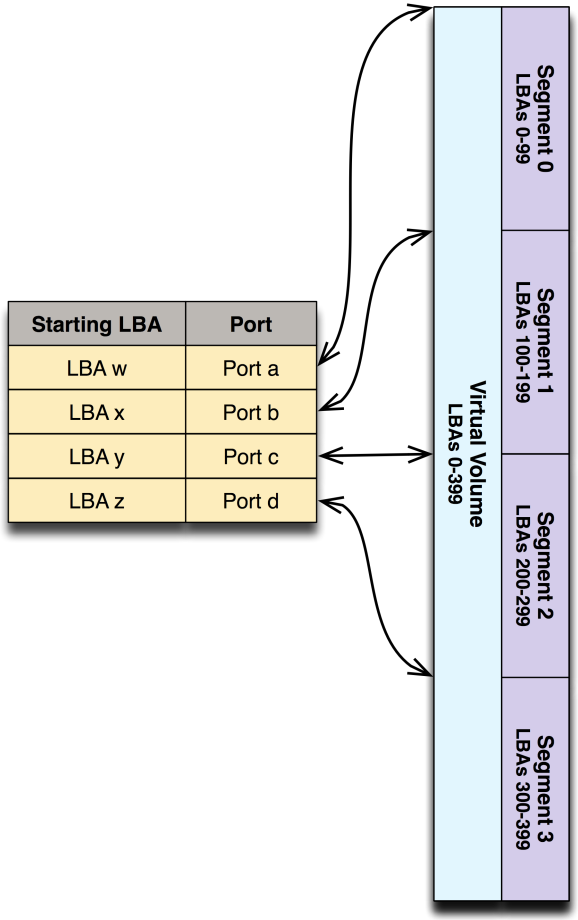
SCSI Referral – Example

- Block Storage Cluster:
  - Four single ported nodes
- Virtual Volume:
  - 100 block segments
  - Concatenated virtual volume
- Server Issues Original I/O:
  - LBA 150, Length 100 to Port 1
- Node 1 Returns Referral List:
  - Port 1, LBA 150, Length 50
  - Port 2, LBA 200, Length 50
- Server Issues Referral I/Os:
  - LBA 150, Length 50 to port 1
  - LBA 200, Length 50 to port 2
- Data Transferred in Parallel
- Status is Merge of Referral I/O Status



SCSI Referral – Caching

- Contain Segment Boundary Information
  - One cache line per boundary
  - Cache entry format:
    - Boundary LBA, Access Port
- Cache Example
  - Cache contents after Example I/O
  - New I/Os can be split using cached boundaries
- Cache Characteristics
  - Not a volume map
  - Updated by referrals
  - Can be incomplete
  - Can be inaccurate
  - Not persistent
  - Target can move data



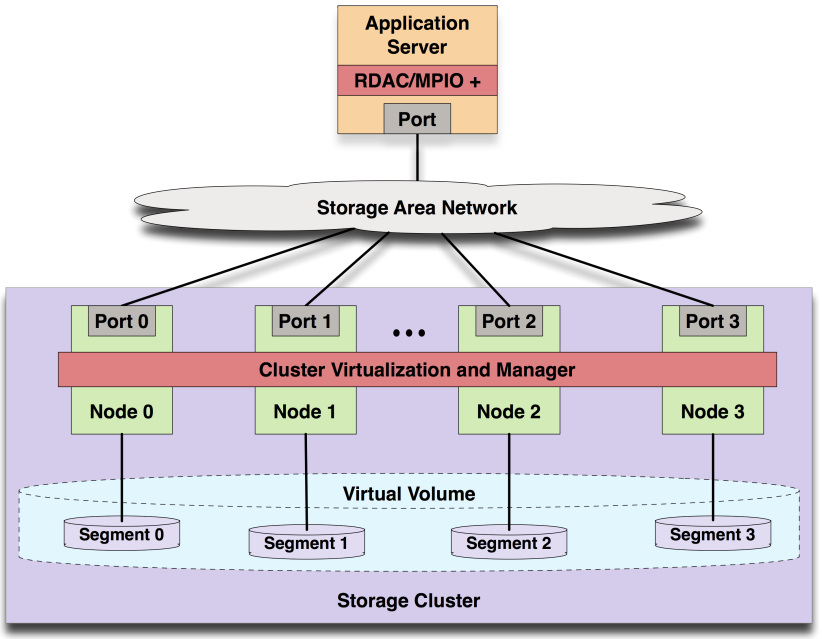
SCSI Referral – Striping, Multiple Paths, …

- Striping
  - Alternate algorithmic format used to describe boundary locations
    - Both in referral and initiator "cache"
    - More efficient lookup for striping, less space in referral, and in initiator cache
  - Cache still used to hold exception boundaries for remapped data
- Multiple Path Support
  - Ports replaced by port lists in Referral and Cache records
  - Supports redundant paths for MPIO and RDAC
- Load Balancing
  - Quality/Priority value associated with each port
  - Allows initiator to pick best port for an I/O from multiple paths
  - Allows target to direct initiator to preferred path
    - Target can dissemble – giving different initiators different views of the data layout
- Arbitrary Scaling
  - Sense data for referral list in command response is limited
  - A referral I/O can result in yet another referral
    - Chains and trees of referrals
- Data Pre-fetching for Referral I/O
  - Node generating referral can pre-fetch data for its referral I/O
  - Storage cluster nodes can exchange pre-fetch notices before referral I/O arrives
  - Data or buffers can be ready for immediate use when referral I/O arrives
- Explicit Segment Boundaries
  - Referral includes starting LBA of each segment referenced by original I/O
  - Initiator can update cache more quickly

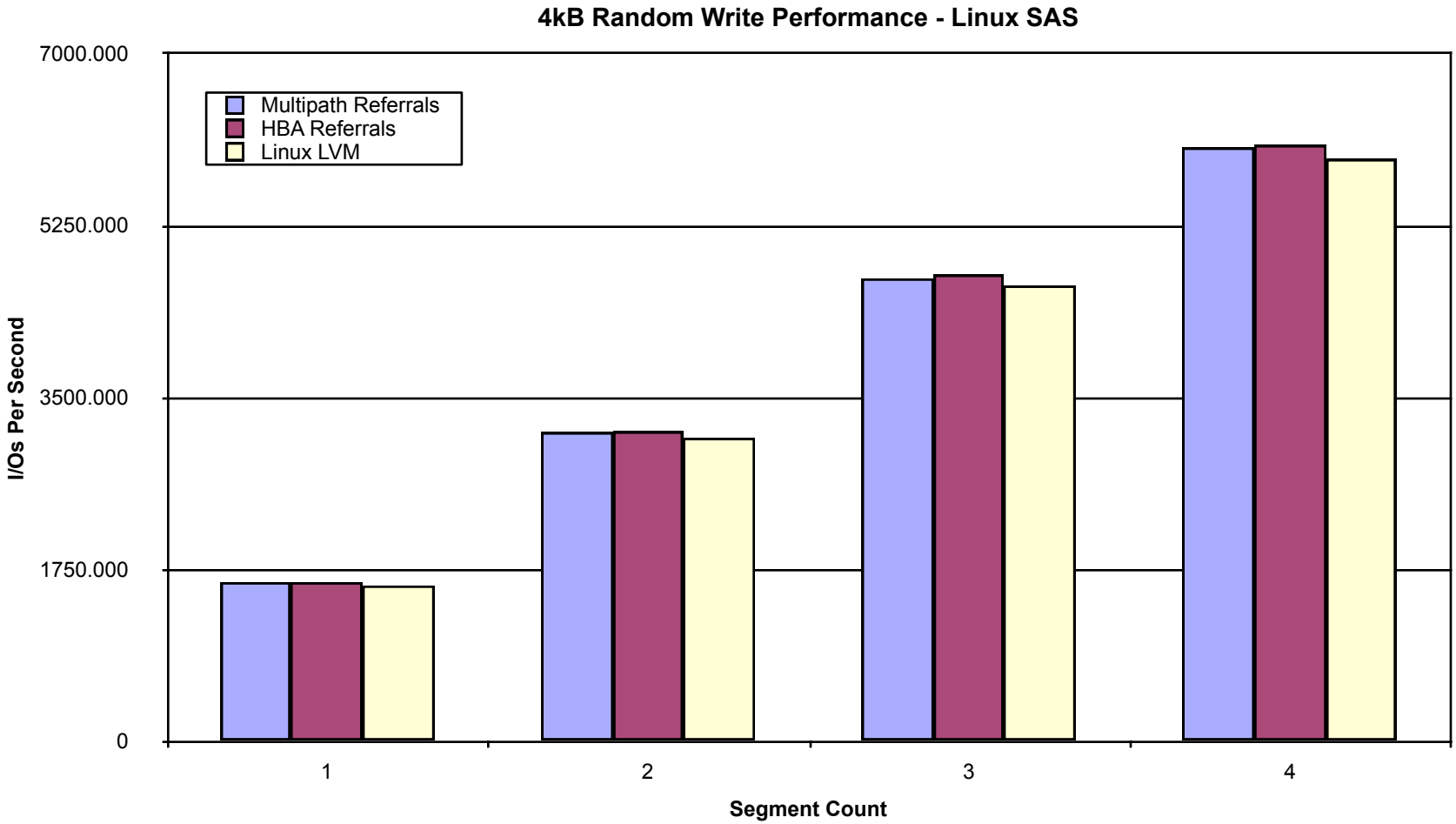
Prototype and Performance

Prototype Implementations

- Storage Node Controller Implementation
  - Controller Firmware, Author: Ross Zwisler
- Three Application Server Implementations
  - Windows RDAC, Author: Scott Masterson
  - Linux RDAC, Author: Ross Zwisler
  - LSI 1068 SAS HBA FW, Author: Mike Fry (LSI - SCG)
- Performance Tests Run in Windows and Linux
  - Concatenated virtual volumes
  - Striped virtual volumes
  - No I/O optimization was done
- Performance Compared to Standard OS Tools
  - Windows Logical Disk Manager (LDM)
  - Linux Logical Volume Manager (LVM)
  - LSI StoreAge SVM Host Agent
- Tested SAN Transports
  - SAS, Fibre Channel, and iSCSI
  - iSCSI prototype took only 2 days
- Important Points:
  - Linear scaling
  - Compares well with established solutions
  - With referrals - **no host side configuration**

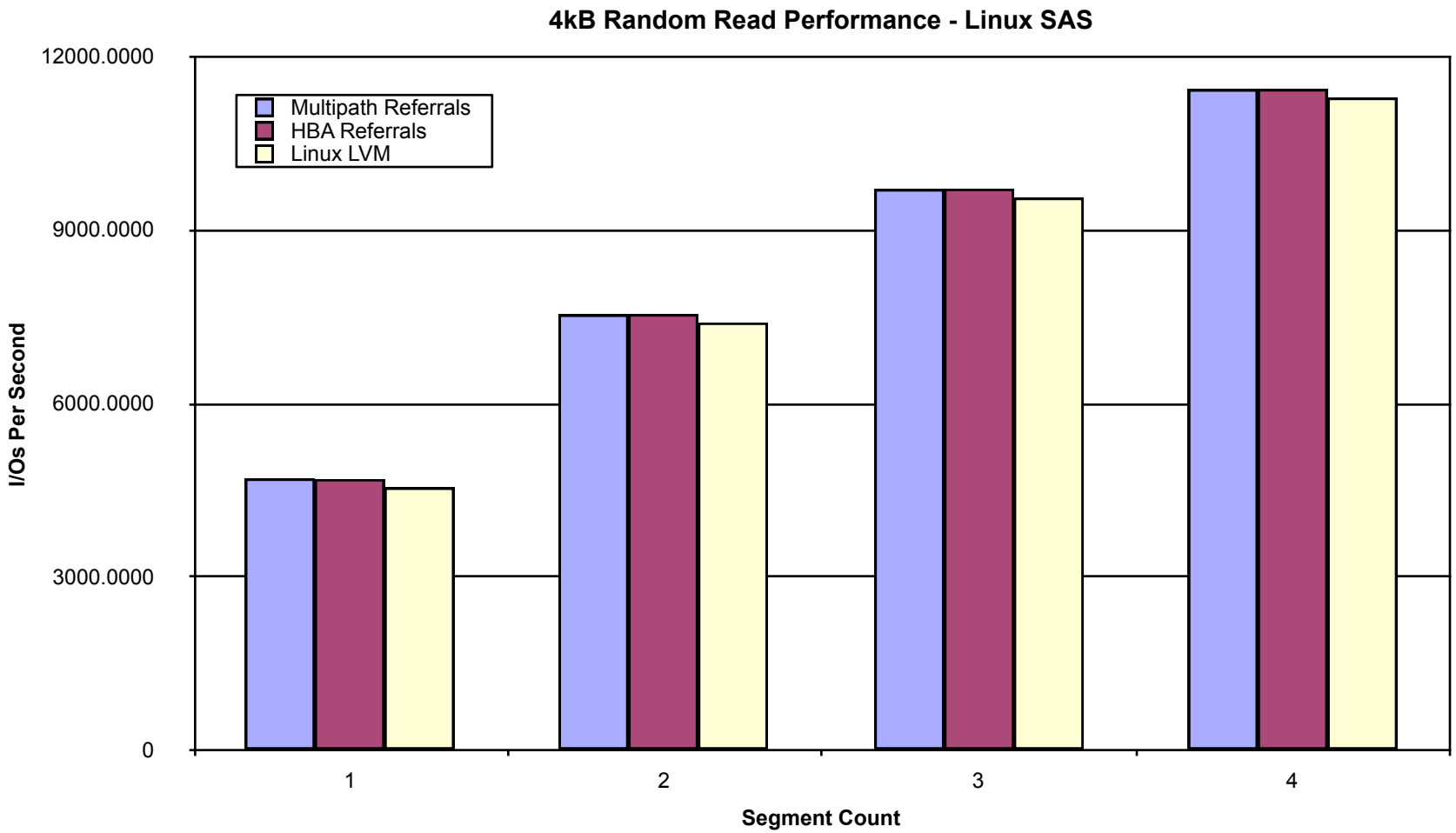


Prototype Performance



SAS, Raw I/O, 200 GB 5+1 RAID 5 volumes, “Iogen” I/O tool, “noop” Linux I/O elevator

Prototype Performance



SAS, Raw I/O, 200 GB 5+1 RAID 5 volumes, “Iogen” I/O tool, “noop” Linux I/O elevator

Conclusion

Benefits of SCSI Referral Approach

- Capacity and Performance Scale Linearly
- No Application Server Configuration Required
- No Additional Hardware Required in SAN or Application Servers.
  - Cluster manager and virtualization done entirely on targets
- Application Servers “learn” Where Data Resides
  - They “relearn” location automatically if data is moved
  - No additional protocols to distribute and maintain volume maps or profiles
  - Target is free to move and reorganize data without informing application servers
- Straightforward RDAC and MPIO Extension for SCSI Referrals
  - New functionality fits naturally with existing responsibilities
  - < 1% of code modified for Linux RDAC prototype
- Any File System Can Use Parallel Storage
- T10 Compatible
  - Planning T10 proposal to add SCSI Referrals


LSI
TM