Using leading-edge building blocks to deploy scale-out data infrastructure

Craig Dunwoody
CTO, GraphStream Incorporated

SNIA Legal Notice

- The material contained in this tutorial is copyrighted by the SNIA unless otherwise noted.
- Member companies and individual members may use this material in presentations and literature under the following conditions:
  - Any slide or slides used must be reproduced in their entirety without modification
  - The SNIA must be acknowledged as the source of any material used in the body of any document containing material from these presentations.
- This presentation is a project of the SNIA Education Committee.
- Neither the author nor the presenter is an attorney and nothing in this presentation is intended to be, or should be construed as legal advice or an opinion of counsel. If you need legal advice or a legal opinion please contact your attorney.
- The information presented herein represents the author's personal opinion and current understanding of the relevant issues involved. The author, the presenter, and the SNIA do not assume any responsibility or liability for damages arising out of any reliance on or use of this information.

NO WARRANTIES, EXPRESS OR IMPLIED. USE AT YOUR OWN RISK.

Abstract

- Using leading-edge building blocks to deploy scale-out data infrastructure
  - Every datacenter includes a set of software and hardware infrastructure building blocks assembled to provide data storage, processing, and networking resources to a set of application workloads. New types of workloads, and new Commercial Off-The-Shelf infrastructure building blocks, are being developed at an increasing rate.
  - These building blocks include a new generation of infrastructure software that can pool and provision hardware resources dynamically, via automation driven by policy and analytics, across a constantly changing and heterogeneous workload mix, at datacenter scale. This enables radical improvements in efficiency and effectiveness of hardware resource usage.
  - Using technical (not marketing) language, and without naming specific products, this presentation covers key storage-related architectural choices and practical considerations for deploying scale-out data infrastructure using the most advanced COTS building blocks.

This presentation

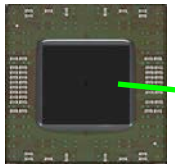
- Slides will be available via Web
  - www.snia.org/education/tutorials
  - Slide sharing site; use favorite search engine
- Big topic
  - I will highlight some key points
- Please feel free to ask questions

Scale-out data infrastructure: example hardware resources, 2015

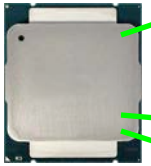
Components

- Small #types, highly replicated
- Physically smaller: faster, less energy

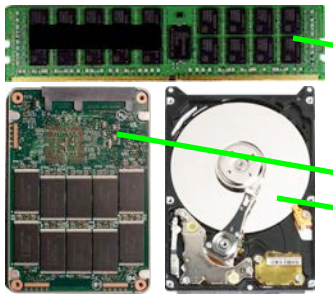
Processor: network-optimized



Processor: general-purpose



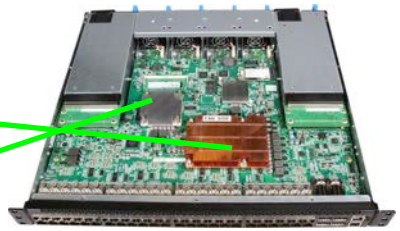
Storage



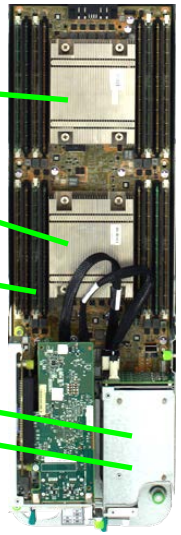
Nodes

- Small #configs cover many use cases
- Differ in processor, network, storage

N Network-optimized server node (“Switch”)



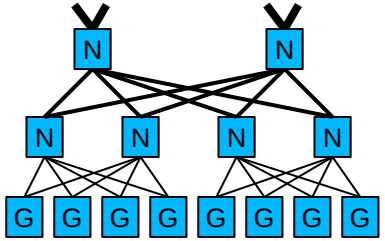
G General-purpose server node



Pods

- #nodes/pod: from <10 to >1000
- Scale-out: multiple pods, sites

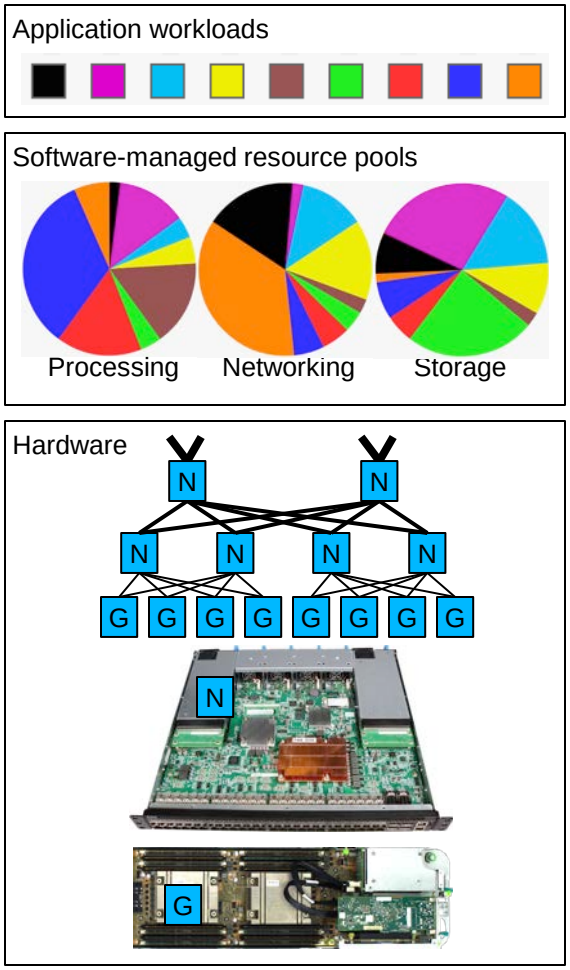
Interconnect topology



Physical layout

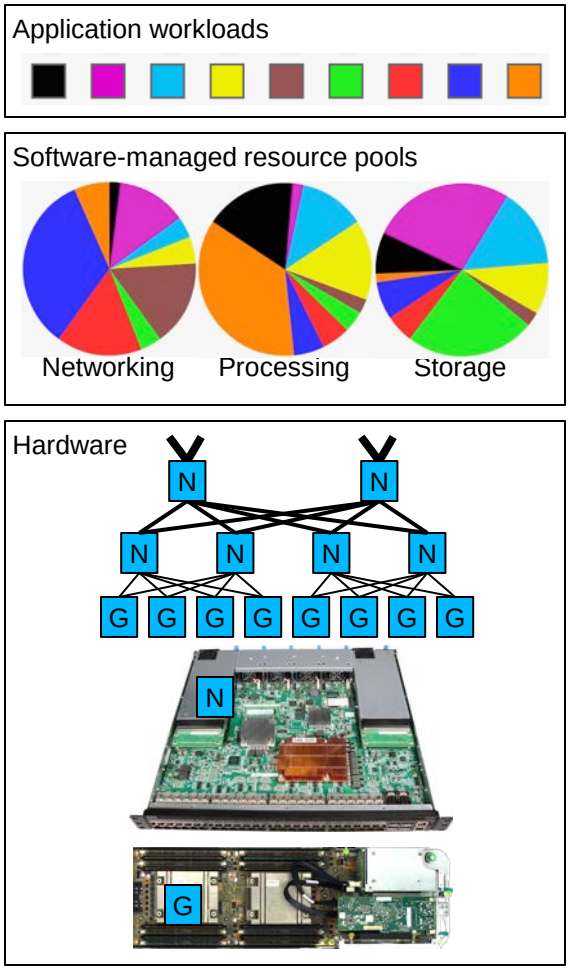
Software to manage resources across multiple nodes, pods, sites

- Operating system software
  - Virtualize & pool infrastructure resources, provision to dynamic set of app workloads
  - 1950s: early single-node OS
  - 2015: very active development on OS for multi-node/pod/site scale-out infrastructure
    - Most advanced: largest infrastructure & application hosting providers; tools used in-house only
    - Many Open Source & commercial efforts, mostly focused on one of processing, networking, storage
    - Open Source examples
      - Processing: Apache Mesos, HTCondor, Kubernetes, OpenStack Nova, SLURM
      - Networking: Floodlight, ONOS, OpenDaylight, OpenStack Neutron
      - Storage: Ceph, GlusterFS, Lustre, OpenAFS, OpenStack Swift, PVFS



“Software-defined” infrastructure, networking (SDN), storage (SDS), etc.

- Objection: infrastructure not “defined” by any single “thing”
- Many very distinct meanings, including:
  - Better multi-node/pod/site scale-out OS software is becoming available
  - Unbundling of integrated hardware+software “appliance” products
    - Enabled by ecosystem of G & N node hardware products with common hardware/software interfaces
    - Processing: unbundling happened years ago; choice of OS & general-purpose apps on G nodes
    - Networking: unbundling happening now; choice of OS & network-services software on G & N nodes; see also Network Function Virtualization (NFV)
    - Storage: unbundling happening now; choice of OS & storage-services software on G & N nodes



This presentation: focused on scale-out storage (SoS) software platforms

- More specifically:
  - Resource-management software platforms, running on scale-out clusters of server nodes, that create & provision pools of virtualized storage resources
- Key characteristics
  - Unified name / address space for files / blocks / objects
  - Cluster nodes based on general-purpose server hardware
  - Nodes interconnected via general-purpose networks
  - Can grow storage capacity incrementally by adding nodes
  - Can grow storage access performance incrementally by adding nodes

“Cambrian explosion” of SoS software platforms

- Well over 50 SoS software platforms currently available
  - Many more in development, various levels of stealth; fast-moving target
  - Many available products very young; small base of field experience
  - Lots of marketing & counter-marketing noise
- Why is this happening now?
  - Growth in demand for storage capacity & access performance
  - Radically better hardware for storage, processing, networking
  - Diaspora of distributed-systems expertise, e.g. from the largest hosting providers
- Key trends
  - Ambition to cover wider range of workloads with a single platform
  - Built-in workload & platform analytics; one key enabler is flash media
  - Automation driven by analytics & policy

Challenge: choosing one of 50+ available SoS software platforms

- You / your clients might benefit greatly from SoS
- Your aggregate storage workload is probably complex
- Each of 50+ available SoS platforms is complex
- How well could each platform support your workload?
  - Depends on many complex interactions between two complex things
  - Only way to find out is to actually test it
  - Such testing is resource-intensive; might be feasible to do POC / bake-off for one or two platforms

How to down-select from 50+ SoS platforms, to a small number?

- Understand your own storage workloads
- Talk with people familiar with specific SoS platforms, ask lots of questions
  - Users
    - Real-world experience
    - Use cases probably don't exactly match yours
  - Vendors, integrators
    - Focused on specific SoS platforms, but incentive to find good product-customer fit
    - Understanding of your storage workloads helps them give you better advice

Agenda

- Understanding your storage workloads
- Some questions you can ask to help with SoS platform down-select

Understanding your storage workloads

- Tools to capture & analyze I/O traces from running workloads
  - blktrace for Linux/GNU platforms
  - Tracing & analysis tools for commercial OS & virtualization platforms
  - Tools to analyze traces & estimate/simulate effect of storage platform parameters (e.g., cache sizes) on app performance
  - Use traces to help understand:
    - How fast each application would run if latency of all I/O operations was zero
    - How application performance degrades as I/O operation latency increases
    - Per-application IOPS and I/O throughput as a function of I/O operation latency

SNIA Tutorial: Solid State Deployments – Recommendations for POC's

Understanding your storage workloads

- At block level, look at:
  - Sequential vs. random read & write requests
  - Read & write distributions across request size
  - Read & write distributions across request time; frequency, interleaving
  - Read & write distributions across address space; working set
- At file/object level, also look at:
  - Distribution of file/object sizes
  - Distribution of file/object lifetimes
  - Create & delete operation distributions across request time
- At file level, also look at other metadata operations:
  - Distribution of #entries per directory
  - Directory operation distributions across request time

Understanding your storage workloads

- At datacenter / service provider level, also look at:
  - How many distinct tenants? How many concurrently active?
  - How many distinct applications? How many concurrently active instances of each application?
  - What storage-specific SLA requirements must be met?
  - How are #tenants, #apps, storage footprint growing over time?

Agenda

- Understanding your storage workloads
- Some questions you can ask to help with SoS platform down-select

SoS platform down-select questions

- These highlight some of many significant capability differences among currently available SoS platforms
  - Far from a comprehensive list; just scratching the surface
  - Easy to come up with many additional questions
- Collectively, currently available SoS platforms have many capabilities
  - Individually, no single SoS platform currently does more than a small subset
  - Lots of “We don’t do that”
  - Lots of “On the roadmap”
- Need to decide which are most important for your use cases

SoS platform down-select: some questions

- Interfaces: API semantics?
- Interfaces: network / wire protocols?
- Networking?
- Capacity scaling & pricing?
- Performance scaling & pricing?
- Cluster node roles?
- Storage media support?
- Data placement & movement?
- Management & monitoring?
- Workload & platform analytics?
- Automation?
- Data durability?
- Data integrity?
- Data efficiency?
- Data services?
- Fault resilience?
- Multi-site replication?
- Continuous availability?
- Online node addition & removal?
- Security?
- Multitenancy?
- Cluster node config flexibility?
- Heterogeneous cluster configs?
- In-cluster app workload support?
- Packaging options?
- Consumption options?

SoS platform down-select questions

- Storage interfaces: API, semantics?
  - In many cases, applications difficult/impossible to change
  - If a platform doesn't support application interfaces you need, it's out
  - Basic API types
    - Block
    - Object
    - File, e.g., POSIX
    - VM-image
  - Consistency semantics
    - E.g., variations on "strong", "eventual"
  - Semantics of individual operations
    - E.g., file locking
  - Cross-API capabilities
    - E.g., individual data object accessible via multiple interfaces, such as Object & File

SoS platform down-select questions

- Storage interfaces: network / wire protocols?
  - Block, e.g. iSCSI
  - File, e.g., NFS v.x, SMB v.x
  - Object, e.g., Swift, S3
  - Custom
    - Specific to individual SoS platform
    - Enable capabilities beyond what is possible with other protocols
    - Needs client-side agent
    - What client platforms are supported?
      - Operating systems?
      - Bare-metal, container virtualization platforms, hypervisor virtualization platforms?
      - Kernel-space client agent available?
      - User-space client agent available?
  - Cross-protocol capabilities
    - E.g., storage volume accessible via multiple protocols, such as NFS & SMB

SoS platform down-select questions

- Networking?
  - If a platform doesn't support network interfaces that you need, it's out
  - Standards
    - E.g., Ethernet 10/25/40/100 Gbps, InfiniBand 32/56/100 Gbps
  - Topologies
    - Separate networks for intra-cluster traffic & client access?
      - Not supported, optional, or mandatory?
    - Redundant links & N nodes, to eliminate single points of failure?
  - Acceleration / SDN
    - N nodes include programmable acceleration hardware (e.g., TCAM) that can make line-rate packet forwarding decisions based on packet-header pattern matching
    - Some SoS platforms use SDN techniques to program this hardware, to help accelerate storage I/O operations

SoS platform down-select questions

- Capacity scaling & pricing?
  - If a platform's capacity scaling & associated economics not workable for your use cases, it's out
  - Scaling limits
    - Pay attention to what has actually been tested/validated, vs. theoretical limits or "unlimited"
    - Min & max supported #nodes per cluster
    - Max volume/container size
    - Max object/file size
  - Usable vs. raw capacity
    - Affected by data durability strategies
    - Affected by data efficiency strategies, interacting with workload characteristics
  - Street pricing related to capacity
    - Per usable TByte in each of years 1..n for purchase, support/subscription

SoS platform down-select questions

- Access-performance scaling & pricing?
  - If a platform's access-performance scaling & associated economics not workable for your use cases, it's out
  - Scaling efficiency
    - Transactions, throughput
    - Challenge claims of "linear scaling"
  - Measuring access performance: difficult during down-select
    - First choice: vendor uses tool to replay your workload traces
    - Second choice: you specify benchmark & parameters, vendor runs it
    - Third choice: vendor-supplied benchmark results
      - At least require basic details, e.g. for IOPS measurement, get read/write mix, block size, working set size, concurrent latency measurement
  - Street pricing related to access performance
    - Per usable [GByte/sec | IOPS] in each of years 1..n for purchase, support/subscription
    - If using vendor performance claims, this is lower bound at best

SNIA Tutorial: Utilizing VDBench to Perform IDC AFA Testing

SNIA Tutorial: Solid-State Deployments – Recommendations for POC's

SoS platform down-select questions

- Cluster node roles?
  - Each node may perform one or more roles, including:
    - Cluster management/monitoring
    - Gateway / proxy / client-access
    - Metadata storage & management
    - Data storage & management
  - Each role must be replicated across multiple nodes for availability
  - To make smaller deployments practical, need to be able to combine multiple roles in each node

SoS platform down-select questions

- Storage media support?
  - Some media options:
    - DRAM, DDR<n> interface
    - Flash, DDR<n> interface
    - Flash, PCIe/NVMe interface
    - Flash, SATA/SAS interface
    - HDD, SATA/SAS interface
    - Storage hosted outside of cluster, Ethernet interface
  - If only in-cluster solid-state media supported:
    - Point solution
    - Typically also need at least one separate platform that includes support for lower-cost media
    - Data silos
    - Manual tiering

SoS platform down-select questions

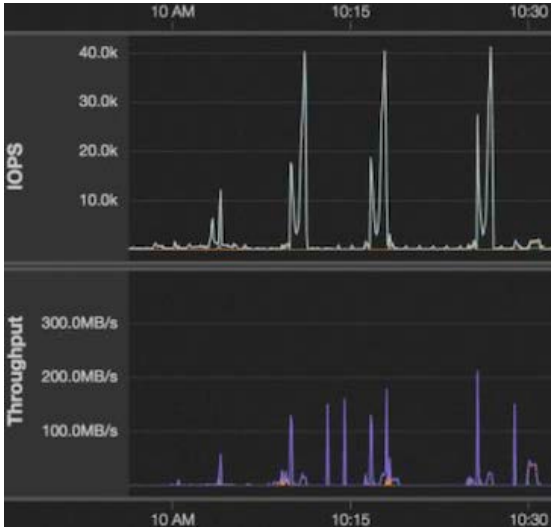
- Data placement & movement?
  - Within & among nodes
  - Tiering vs. caching
  - Do writes go to highest-performance media first?
  - Rebalancing after node addition/removal/failure

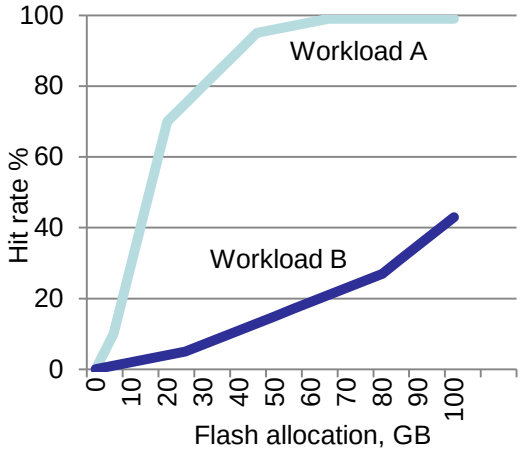
SoS platform down-select questions

- Management & monitoring?
  - Interfaces
    - GUI
    - CLI
    - API
    - Are GUI and CLI implemented entirely on top of API, so all capabilities are available via all interfaces?
  - Integrations with other infrastructure management & monitoring tools
    - E.g., tab in virtualization-platform console
  - Remote services offered by vendor and/or hosting providers
    - Phone-home

SoS platform down-select questions

- Workload & platform analytics?
  - Better analytics facilitated by
    - Better processors & media (flash)
    - Improved data structures & algorithms for metadata & analytics
    - Now feasible to:
      - Capture & retain full workload traces for long periods
      - Support online queries that older systems can't support
  - Example queries
    - Latency, throughput, IOPS correlated over time
    - For file/object: capacity use over time
    - For tiering/caching: hit rate per tier/cache layer, as a function of layer capacity
  - Granularity of queries
    - Per client
    - For file access: per file, per subtree
    - For virtualized clients: per VM





SoS platform down-select questions

- Workload & platform analytics
  - Predictive analytics
    - Pattern recognition, learning algorithms
    - Automatically promote specific data to higher tier in advance of predicted access
    - Recommend addition of hardware resources to avoid predicted shortfalls in storage capacity, access performance
    - Recommend remedial actions in advance of predicted component failures
- Automation?
  - Analytics helps administrators gain insight into workload and platform behavior, make decisions, & take action manually
  - Next step: options to help automate simpler decisions & actions
    - Baked-in policy options for common situations
    - Simple rule systems
    - API to provide access to system state & analytics results, enable arbitrary decision logic to drive actions

SoS platform down-select questions

- Data durability?
  - "The universe hates your data"
    - Challenge to maintain pools of low entropy
  - Estimated time to data loss
    - Uncomfortable topic for vendor, but should be willing & able to discuss with you
  - Commonly used mechanisms
    - Replication
    - Erasure coding: typically slower access, more space-efficient
  - Control & automation
    - E.g., migration of data between replicated & erasure-coded pools
    - Can be based on policies, access statistics

SoS platform down-select questions

- Data integrity?
  - Hardware & software sometimes do bad things to data
  - Storage-media failure mode examples
    - Read
      - Wrong address
      - Data error
    - Write
      - Wrong address (“wild write”)
      - Data error
      - No-op (“lost write”)
  - Bit flips in network
  - End-to-end mechanisms to detect & correct write & read errors along full path from applications to storage media

SoS platform down-select questions

- Data efficiency?
  - Thin provisioning
  - Deduplication
  - Compression
  - Inline vs. post-process
  - Performance tradeoffs
    - Inline compression & dedupe typically add latency to datapath
    - Benefits are workload-specific
  - Control & automation
    - Enable/disable data efficiency mechanisms at per-workload granularity, based on policy, online workload analysis

SoS platform down-select questions

- Data services?
  - Snapshots
  - Clones (writeable snapshots)
  - Backups
  - Object/file versioning
  - Differences among available implementations
    - Performance of each operation
    - Limitations, e.g. max number of snapshots/clones
  - Integration with virtualization platforms

SoS platform down-select questions

- Fault resilience?
  - Rebuild after media or node failure/replacement
    - Impairment of access performance during rebuild
    - Impairment of rebuild performance based on application load during rebuild
  - Sending new writes to flash helps performance of HDD rebuilds
  - Random disconnect/reconnect test
    - Network cables
    - Power cables
    - Storage-media modules
  - Configurable failure domains
    - E.g., rack-awareness – ensuring that replicas of an object are spread across at least two racks, to maintain object availability in the event of power loss affecting a single rack

SoS platform down-select questions

- Multi-site replication / federation?
  - Sychronous / metro
  - Asynchronous
  - Bidirectional replication, for active/active site operation
  - Vendor spec for max network latency
  - Built-in WAN optimization
  - Integration with other tools to orchestrate site failover/failback

SoS platform down-select questions

- Continuous availability?
  - No maintenance windows
  - No data wipe on software/firmware upgrades
  - No-downtime in-service upgrades
    - Software
    - Firmware
      - Server (BIOS, platform controllers, power supplies, etc.)
      - Storage controllers
      - Storage media modules
    - Automated, rolling across cluster
    - Rollback of failed upgrades
- Online, no-downtime, no-admin addition, removal of nodes?
  - Automatic redistribution/rebalancing of existing data
  - Automatic expansion, contraction of capacity pools

SoS platform down-select questions

SNIA Tutorial: Storage Security Best Practices

- Security?
  - Encrypted data at rest
  - Encrypted data in motion
  - Encryption key management
  - Resistance to various types of attacks, incl. DoS
- Multitenancy?
  - Multiple independent, untrusted clients
  - Client isolation
  - Policy-driven Quality of Service
    - Management of SLA constraints
    - Admission control

SoS platform down-select questions

- Cluster node configuration flexibility?
  - Storage media modules
    - DRAM, DDR<n> interface
    - Flash, DDR<n> interface
    - Flash, PCIe/NVMe interface
    - Flash, SATA/SAS interface
    - HDD, SATA/SAS interface
  - Processors
  - Network interfaces
    - Ethernet 10/25/40/100 Gbps
    - InfiniBand 32/56/100 Gbps

SoS platform down-select questions

- Heterogeneous cluster configurations?
  - Within a single technology generation
    - Performance-optimized nodes
      - Solid-state media
      - More processing & networking resources
    - Nodes configured to minimize lifecycle cost per unit of capacity
      - Magnetic media, possibly with spin-down capability
      - Fewer processing & networking resources
  - Across multiple technology generations
    - Ability of platform architecture to take full advantage of upcoming technologies
      - Storage media, processing, networking
      - Radically lower latency at hardware level
    - Collapsing commonly using storage software stacks
      - Need to reduce latency in software stacks, order to benefit from latency reductions at hardware level
  - Node-specific distribution of application workload within SoS cluster
    - Driven by node-specific resource profiles

SoS platform down-select questions

- In-cluster application workload support?
  - "Hyper-convergence": just one of many possible features for SoS
  - Move computation to data, not vice versa
  - Example use cases
    - Read-intensive distributed parallel analytics
    - Storage-latency intolerant workloads
      - E.g., some financial-services apps
    - Virtual Desktop Infrastructure
  - Execution environments for application workloads
    - Bare metal
    - Container-based virtualization platforms
    - Hypervisor-based virtualization platforms

SNIA Tutorial: Separate vs. combined server clusters for app workloads & shared storage

SoS platform down-select questions

- Packaging options?
  - Integrated hardware+software appliances
    - Key advantage
      - Configurations tested/validated by vendor
  - Software, combined with hardware by integrator or end-user
    - Key advantages
      - Can choose commonality with existing hardware infrastructure
      - Can take advantage of price competition among hardware suppliers
      - Can take advantage of new hardware generations sooner
    - Hardware platforms
      - Hardware Compatibility List
    - Hosted-infrastructure platforms
      - E.g., Infrastructure as a Service providers

SoS platform down-select questions

- Consumption options?
  - Capacity-based
  - Access-based
  - Purchase + maintenance
  - Service subscription
  - CAPEX vs. OPEX
    - Vendor marketing mistakenly makes assumptions about end-user preferences
    - Ask the CFO!

Summary

- You / your clients might benefit greatly from SoS
  - Implemented via a software platform running on a cluster of general-purpose server units interconnected by switch server units
- Number of available SoS software platforms is well over 50, & growing
- Only way to really know how well a specific SoS platform will support specific set of workloads, is to test it
  - Resource-intensive; typically feasible to test at most one or two
- This presentation: suggestions to help down-select from 50+
  - Understanding your storage workloads
  - Some questions you can ask about specific scale-out storage platforms

Attribution & Feedback

The SNIA Education Committee thanks the following Individuals for their contributions to this Tutorial.

Authorship History

Craig Dunwoody, 2015/04/07

Additional Contributors

Please send any questions or comments regarding this SNIA Tutorial to ***tracktutorials@snia.org***