Scale-out Storage – Solution and Challenges

Mahadev Gaonkar
iGATE

Table of Content

Scale-out Storage Overview

- Distributed scalable storage
- Distributed scalable IO computing
- Infrastructure versus system
- Interoperable
- Fault tolerant
- Infrastructure manageability

Scale-out NAS Approaches

- **Scale-out NAS with no metadata**
  - Uses consistent hashing (e.g.. MD5 algorithm) for distribution and look-up of files in the cluster
- **Scale-out NAS with some metadata**
  - Uses consistent hashing and maintains some metadata for distribution and look-up of files in the cluster
- **Scale-out NAS with metadata**
  - Uses pre-defined parameters (e.g. file count, capacity of the storage node) for distribution of files
  - Metadata is used for file look-up

Scale-out NAS Approaches

Scale-out NAS with no metadata

- Highlights
  - Fast look-up
  - No single point of failure as no metadata server
  - Good scalability
- Lowlights
  - No load balancing
  - File movement in case of rename, node add
  - Rebalancing mandatory

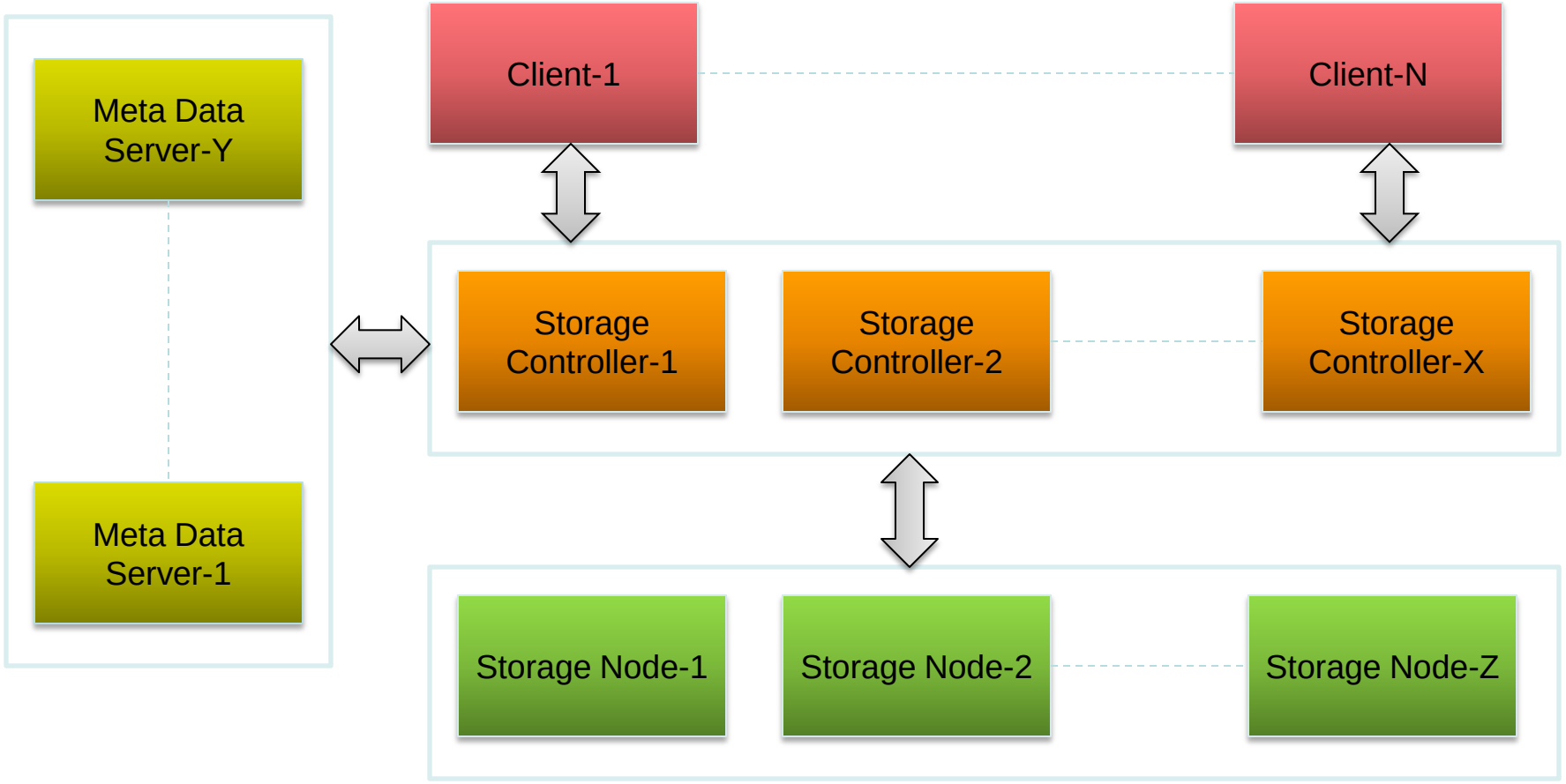
Scale-out NAS with some metadata

- Highlights
  - Good load balancing
  - Only metadata movement and no file movement in case of rename, node add
  - Metadata can be distributed on multiple servers to prevent single point of failure
  - Good scalability
- Lowlights
  - Average look-up time

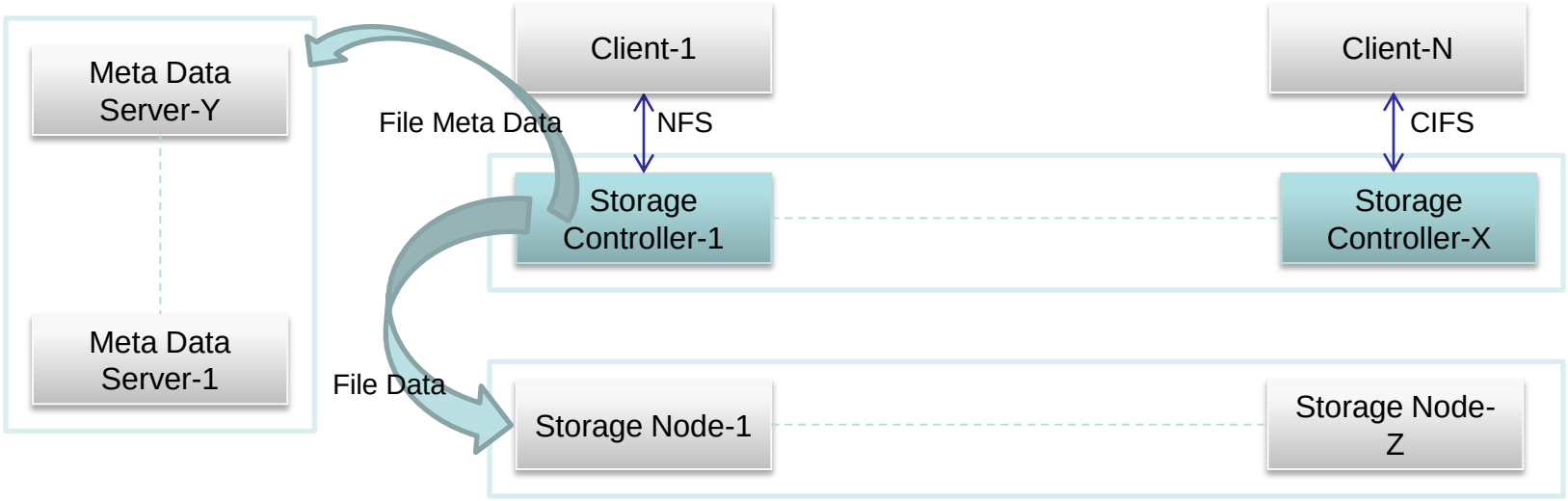
Scale-out NAS with metadata

- Highlights
  - Look-up time could be proportional to number of files
  - Good load balancing
  - No file movement
- Lowlights
  - Large metadata
  - Metadata considerations for scalability

Scale-out NAS Architecture

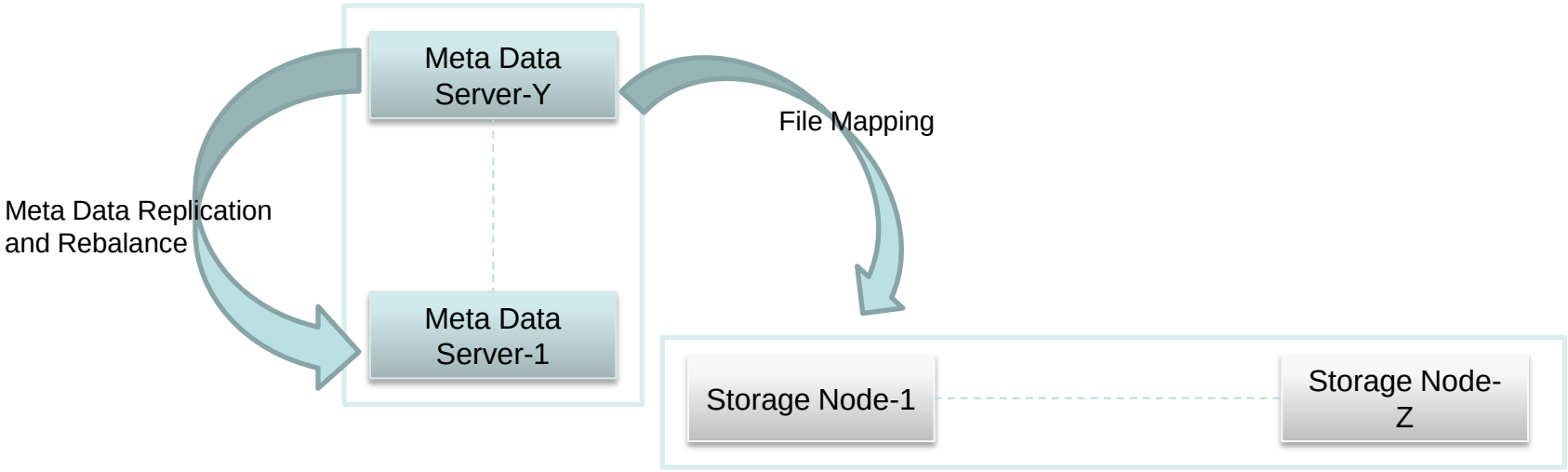


Scale-out NAS Components – Storage Controller



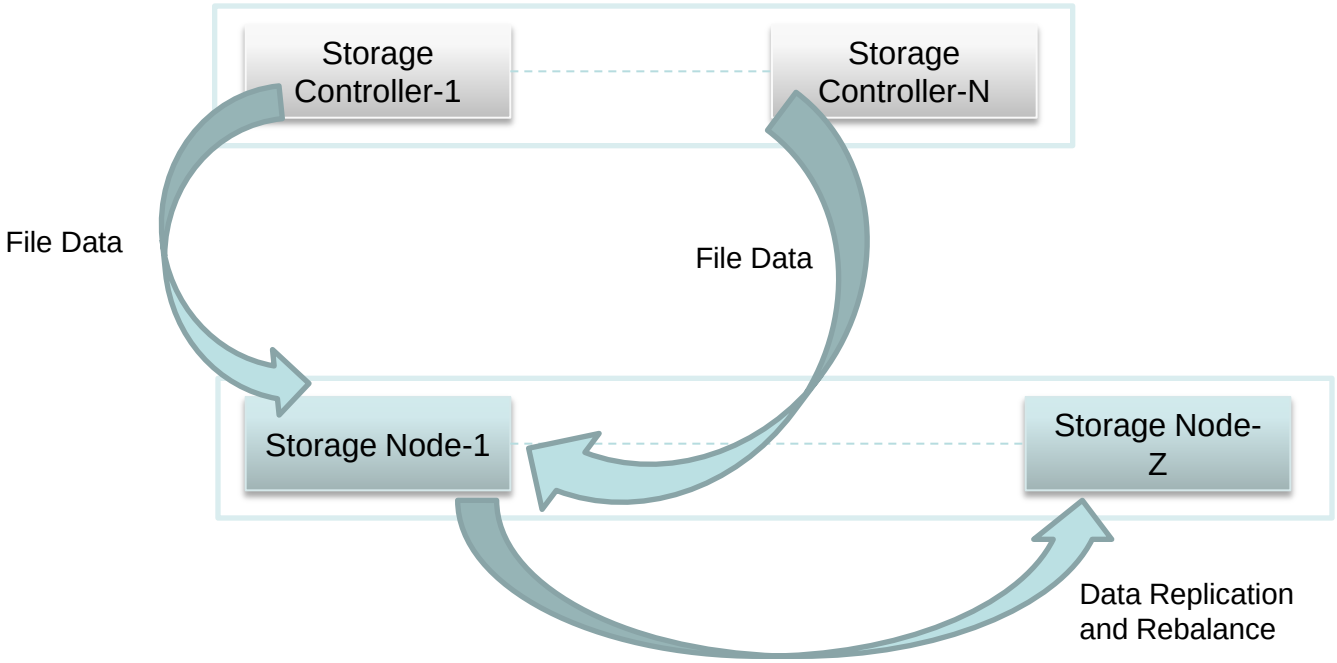
- Storage Controller provides:
  - Global Namespace- Provides a single unified global file system view to clients
  - Metadata Management- Distribution and lookup of file metadata using consistent hashing
  - Data Management - Distribution and lookup of file data using weight-based (on file count, node capacity) algorithm
  - Client Request Management – Accepts and responds to client requests
  - Automatic failover by deploying redundant controllers

Scale-out NAS Components – Metadata Server



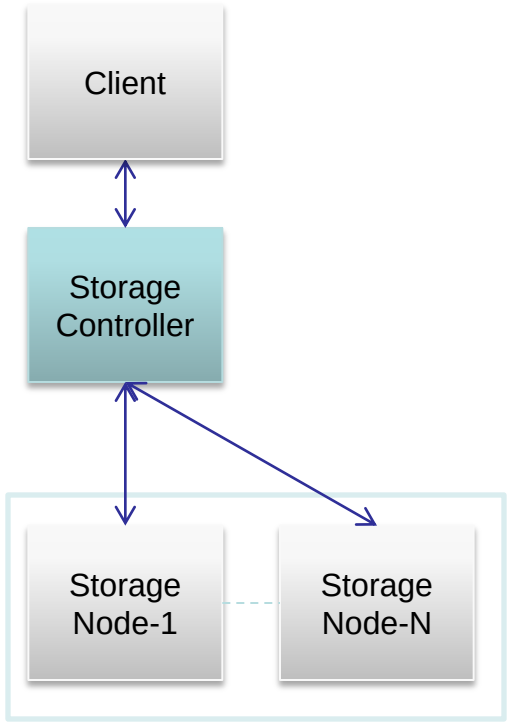
- Metadata server provides:
  - Repository for directory hierarchy
  - File to Storage Node mapping
- Metadata servers are clustered for:
  - Scalability - distribution of file metadata uniformly enables metadata workload distribution and capacity scaling
  - Availability - replication of metadata
  - Elasticity - metadata servers can be added or removed in cluster; metadata is automatically rebalanced

Scale-out NAS Components – Storage Node



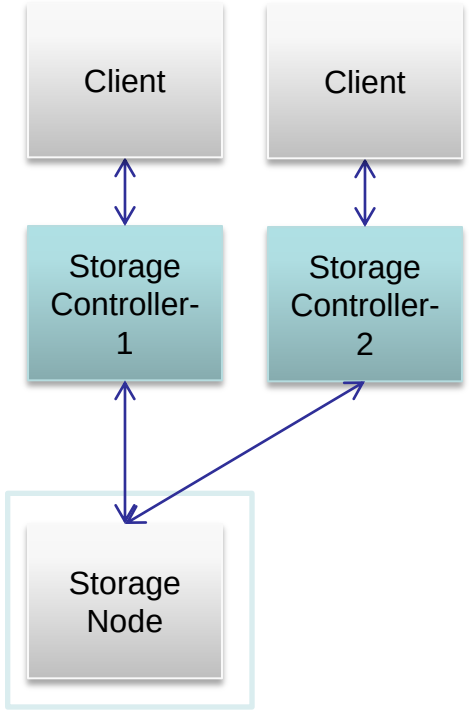
- Storage Node provides:
  - Storage: File based data storage (Native file system)
  - Scalability: Nodes can be added/ removed non-disruptively
  - Load Balancing: Data Rebalancing when nodes are added/ removed
  - Availability: Policy based data replication
  - IO Handling: Services IO requests

Storage IO Workload Distribution



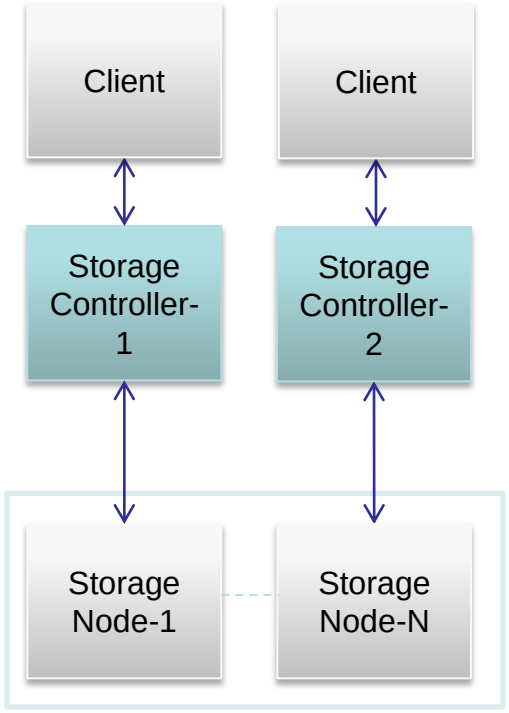
- Single large file IO

  Parallel read/write of file chunks to multiple Storage Nodes



- Multiple files IO

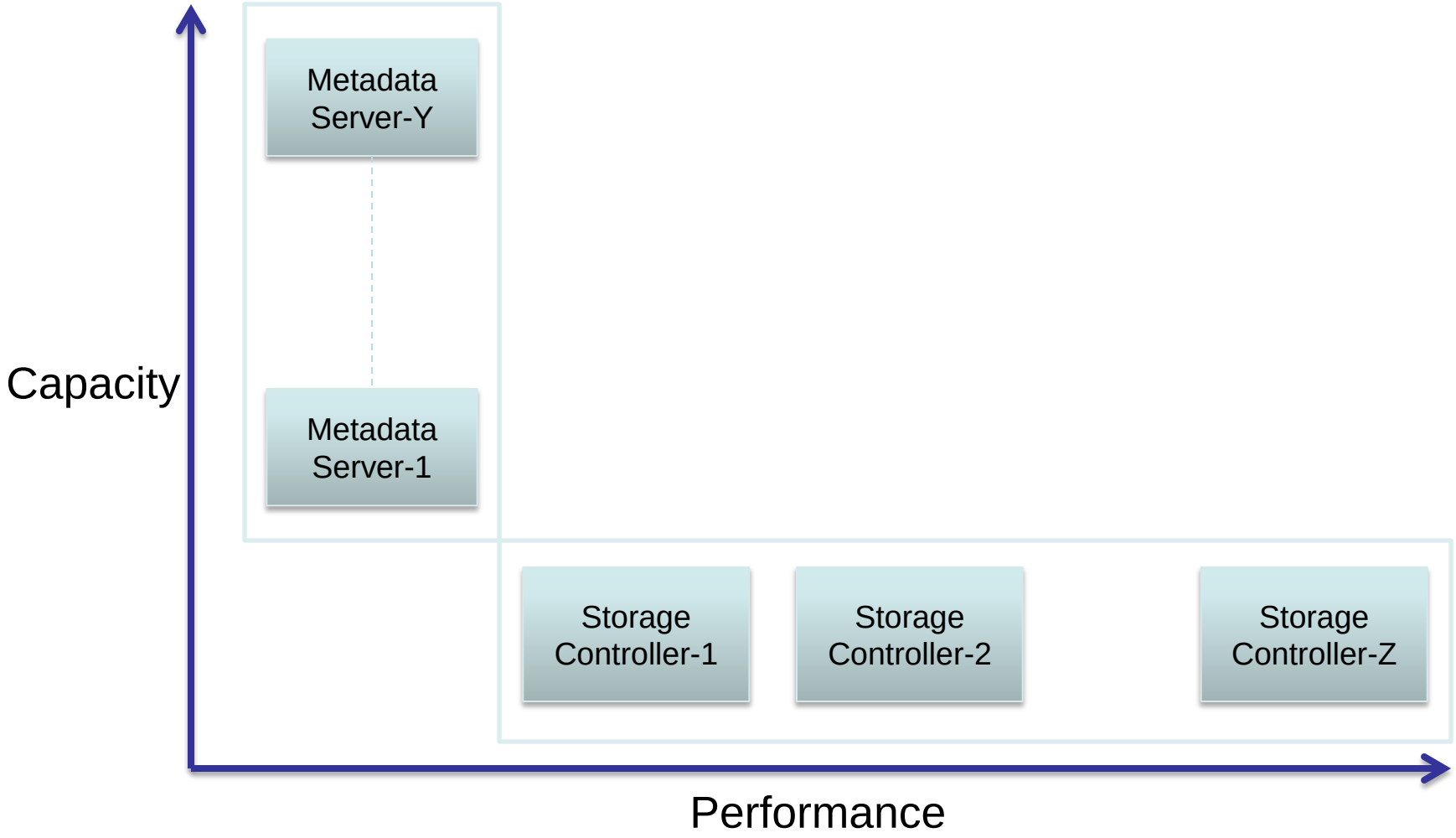
  Parallel read/write to Storage Node with multiple Storage Node Managers



- Multiple clients IO

  Parallel read/write by multiple Storage Controllers

Scale-out NAS Scaling



Technical Challenges

Challenge	SUN-RPC based daemons are single threaded (in Linux)
Issue	Handling simultaneous requests
Solution	•Multiple SUN RPC server processes •Dynamically spawn/kill processes depending upon workload

Challenge	In TCP/IP, sockets once released are not available immediately, to avoid denial of service
Issue	No socket available for Client-Server communication after a threshold
Solution	•Connection pool of sockets •TCP/IP tuning (tcp_fin_timeout, ip_local_port_range, tcp_tw_reuse, tcp_tw_recycle, tcp_max_syn_backlog, netdev_max_backlog, somaxconn) •SUN RPC tuning (tcp_fin_timeout, rpc_timeout)

Technical Challenges cont..

Challenge	Directory tree is replicated on metadata servers
Issue	Synchronizing simultaneous directory operations across metadata servers
Solution	•Use of master metadata server for synchronization. •All directory operations except readdir() are routed through master metadata server

Challenge	Files are distributed across storage nodes leading to directory tree replication on each node
Issue	Synchronization and performance issues associated with create, delete, rename operation of directories on each storage node due to replication of directory tree on multiple storage nodes
Solution	•Maintaining the directory tree at the Metadata Server •File is uniquely identified by prefixing directory inode to file name

Test Strategy - Traditional v/s Scale-out NAS

Traditional NAS Test Considerations

- Designed to handle low volume of data
- Performance bottleneck when multiple users access simultaneously
- Capacity scaling
- Performance degradation as data traffic grows
- Risk of data unavailability in case of NAS head failure

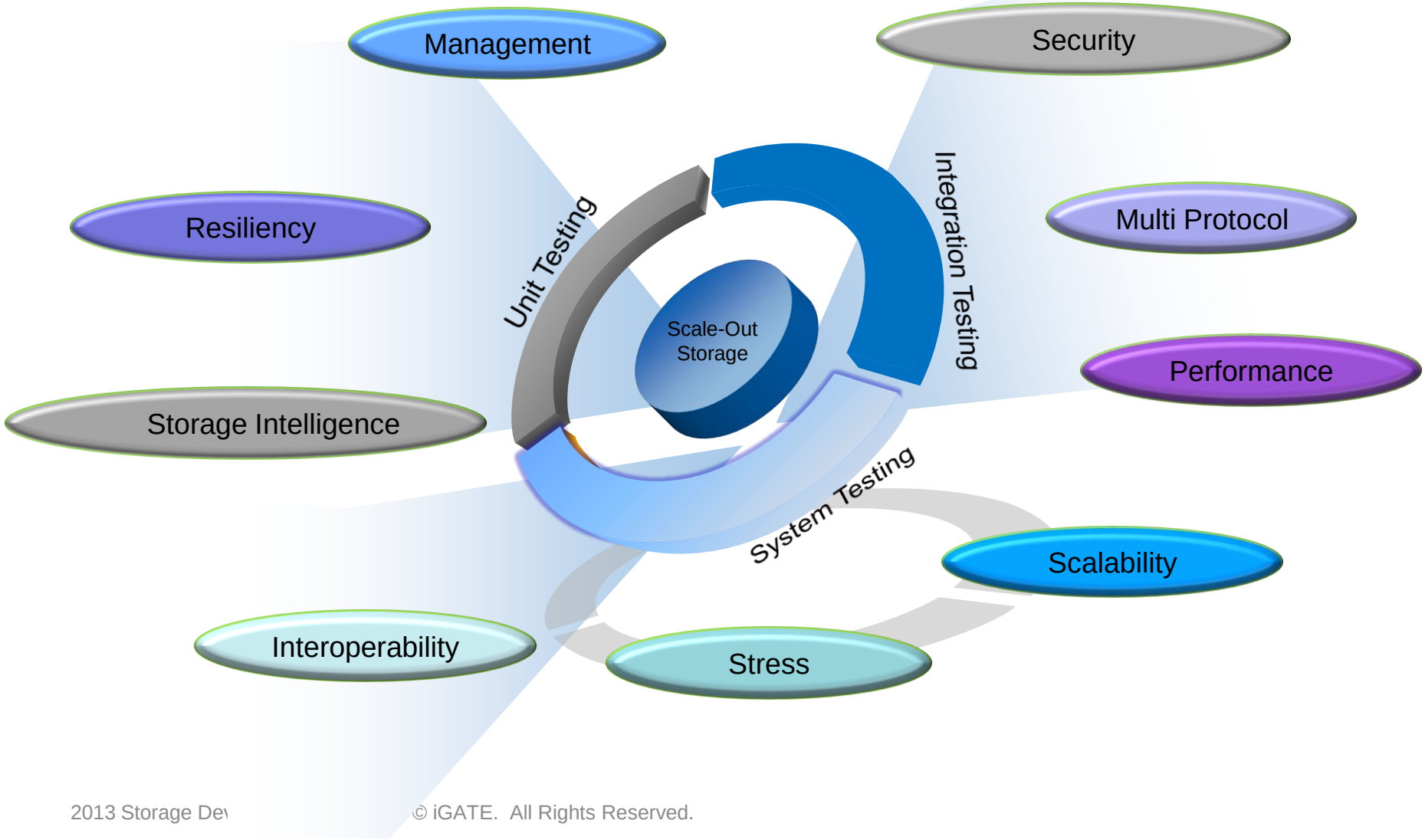
Scale-out NAS Test Considerations

- Ability to scale and store huge data
- Load balancing across nodes – Performance improvement
- Capacity scaling
- Dynamic scaling by on-the-fly addition of nodes

Scale-out NAS Testing challenges

- Huge data creation for scalability testing
- Performance testing to consider node scale-out and scale-down conditions
- Initiation of multiple simultaneous operations from multiple clients

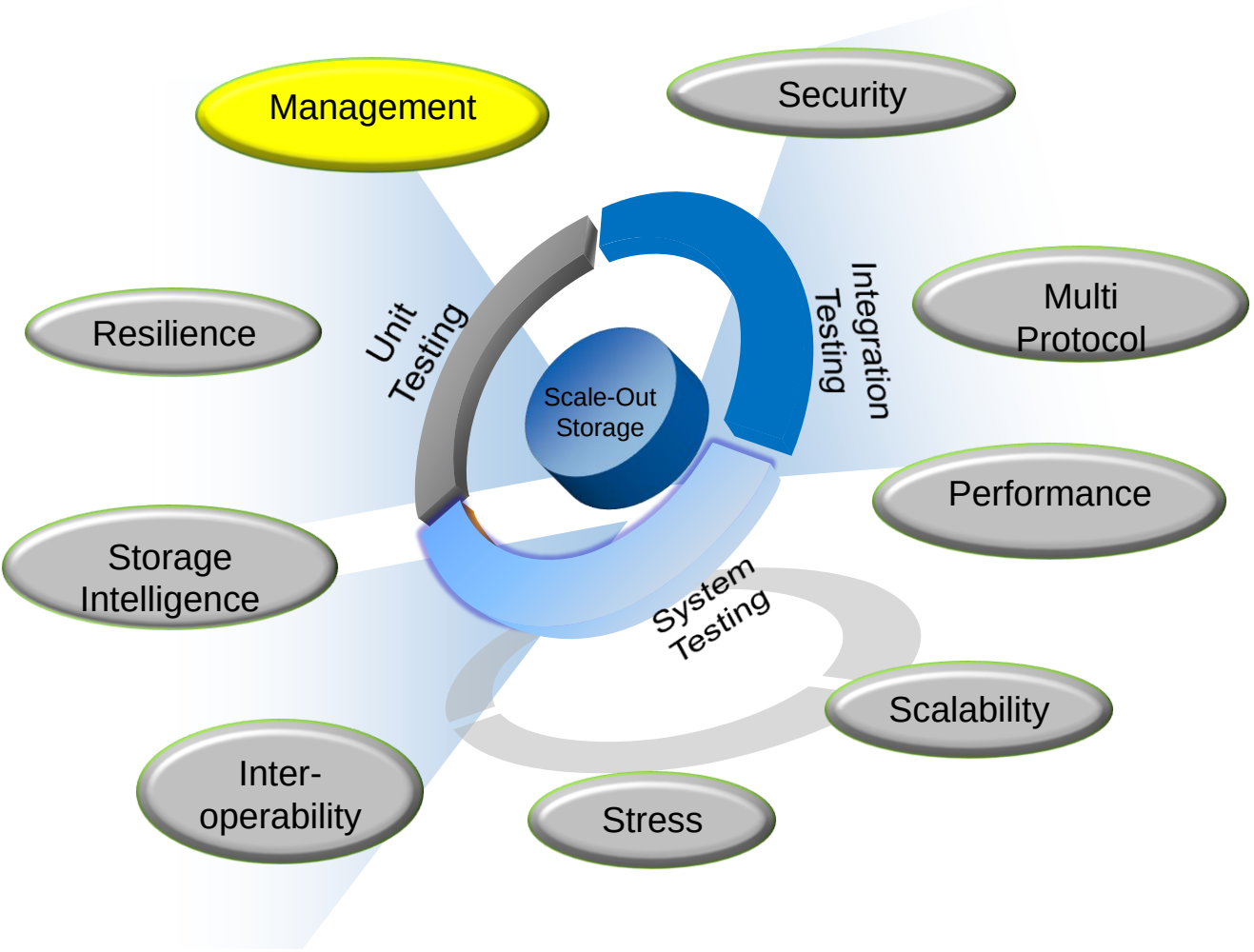
Scale-out NAS Test Strategy



Scale-out NAS Test Tools

Tools	Performance	Stress	Inter-operability	Multi-protocol	Scalability	Security
Bonnie++	Yes	No	No	No	Yes	No
XDD	Yes	No	Yes	Yes	Yes	No
DBench	Yes	Yes	Yes	Yes	Yes	No
FIO	Yes	Yes	Yes	Yes	Yes	No
FFSB	Yes	Yes	No	Yes	Yes	No
Filebench	Yes	Yes	No	No	Yes	No
PGMeter	Yes	Yes	Yes	Yes	No	No
NASPT	Yes	Yes	No	No	No	No
IOZone	Yes	No	Yes	Yes	No	No

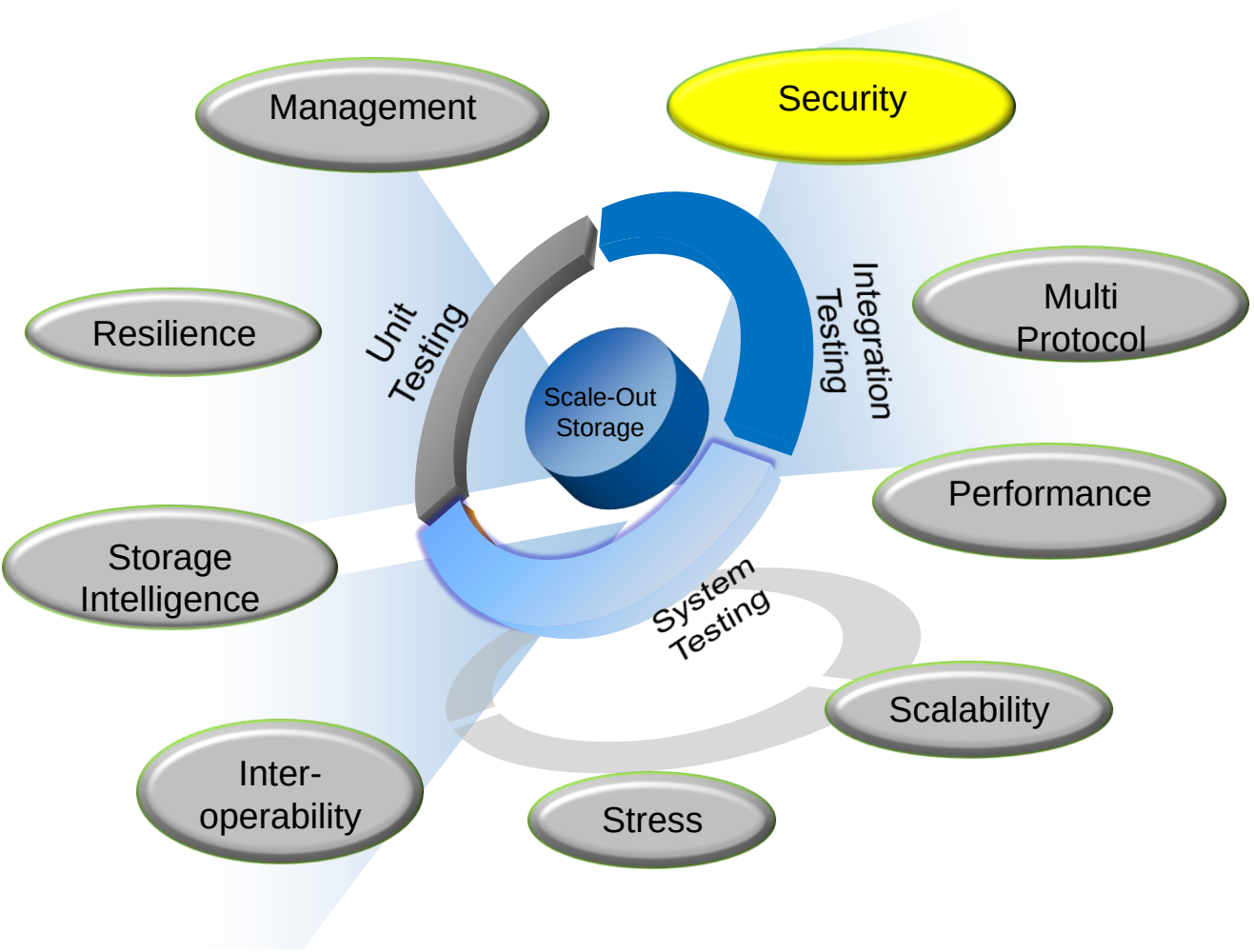
Scale-out NAS Test Strategy



Management

- Orchestration
- Software upgrades and maintenance
- User account management
- GUI complexity
- Ease of deployment

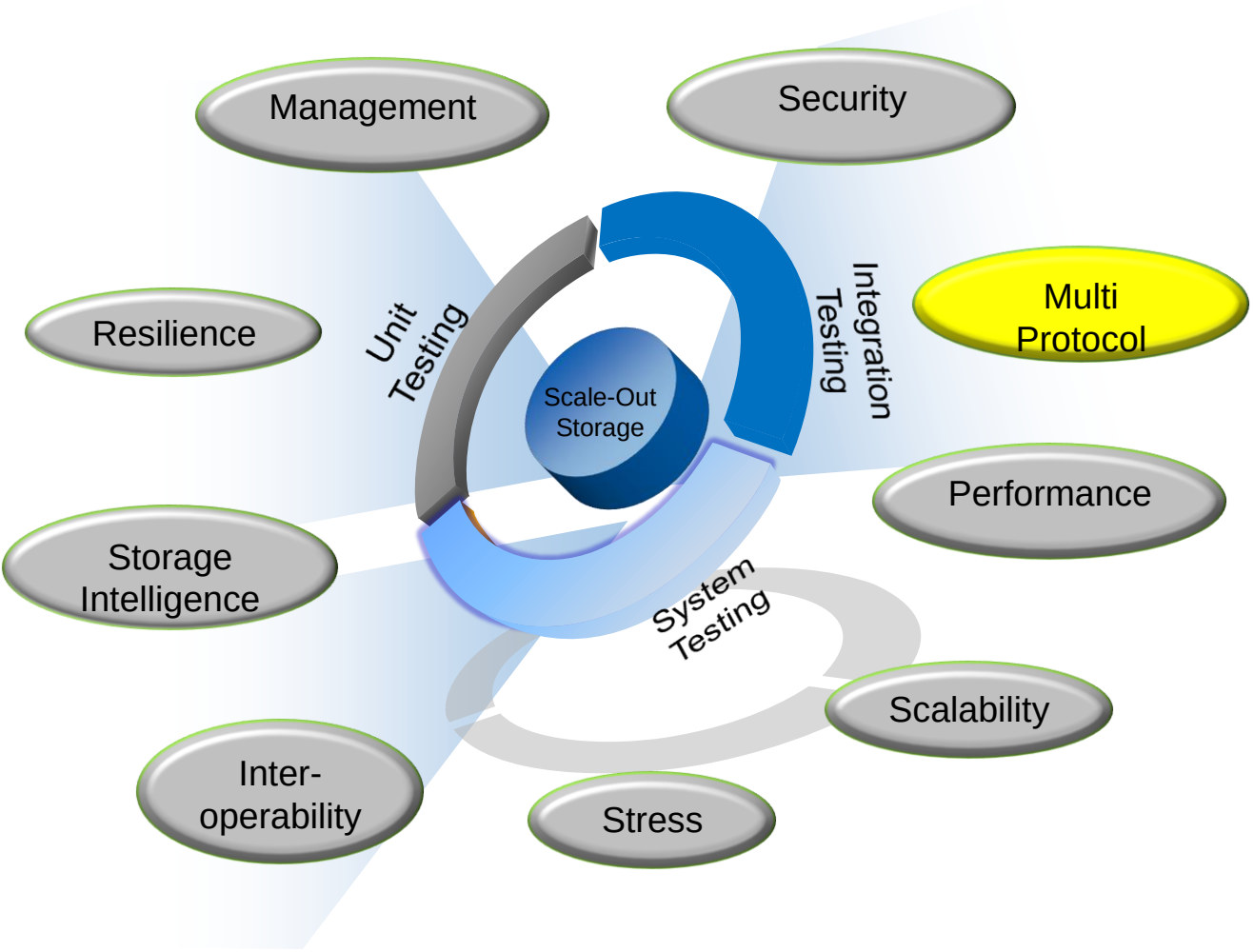
Scale-out NAS Test Strategy



Security

- Authentication (RPC/ AD/ LDAP / NIS)
- Network Security
- Data security
- Antivirus support

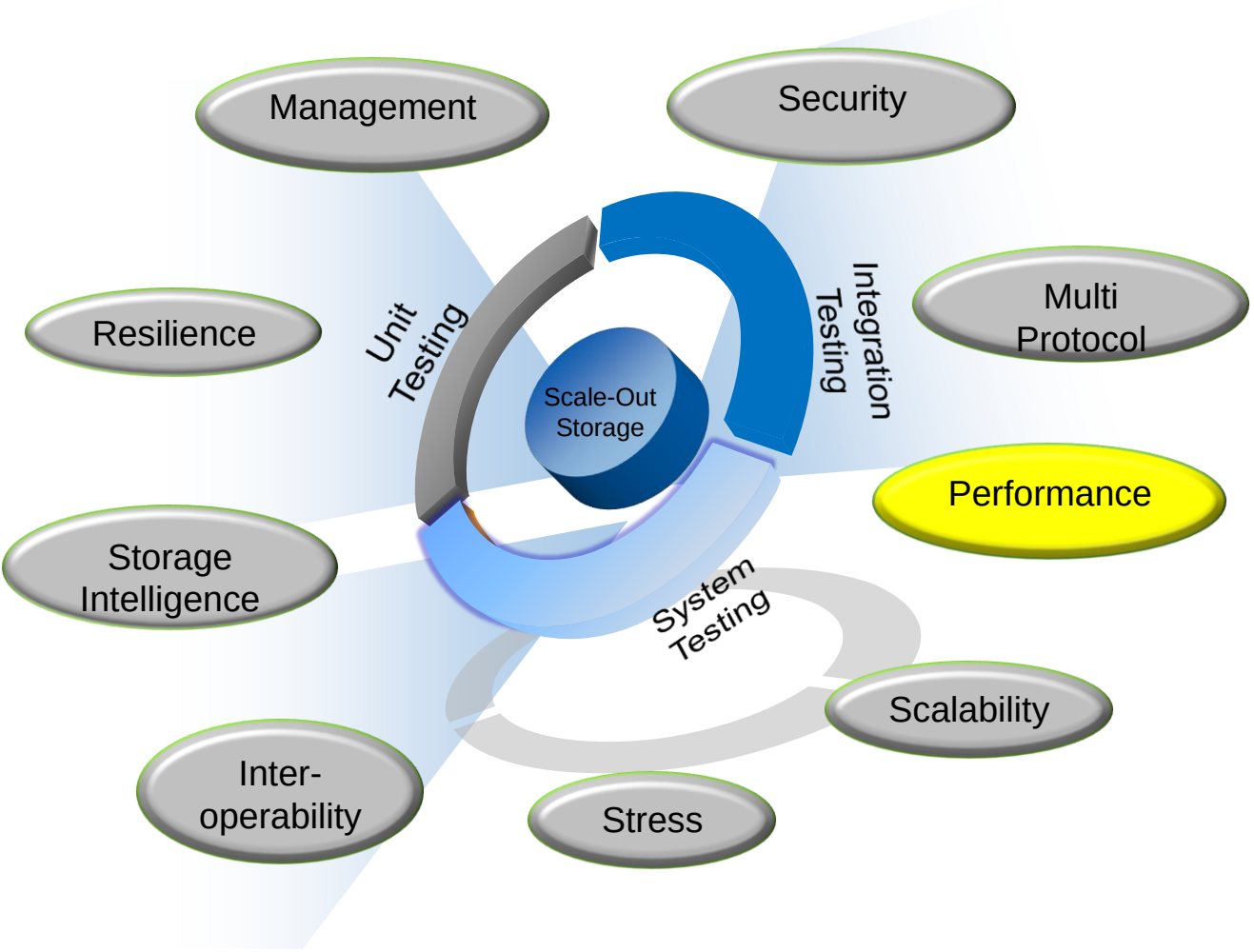
Scale-out NAS Test Strategy



Multi Protocol

- NFS, CIFS, FTP, HTTP protocols
- Backward Compatibility
- Infiniband/FC interoperability
- Support of SSH, SCP for backend operations

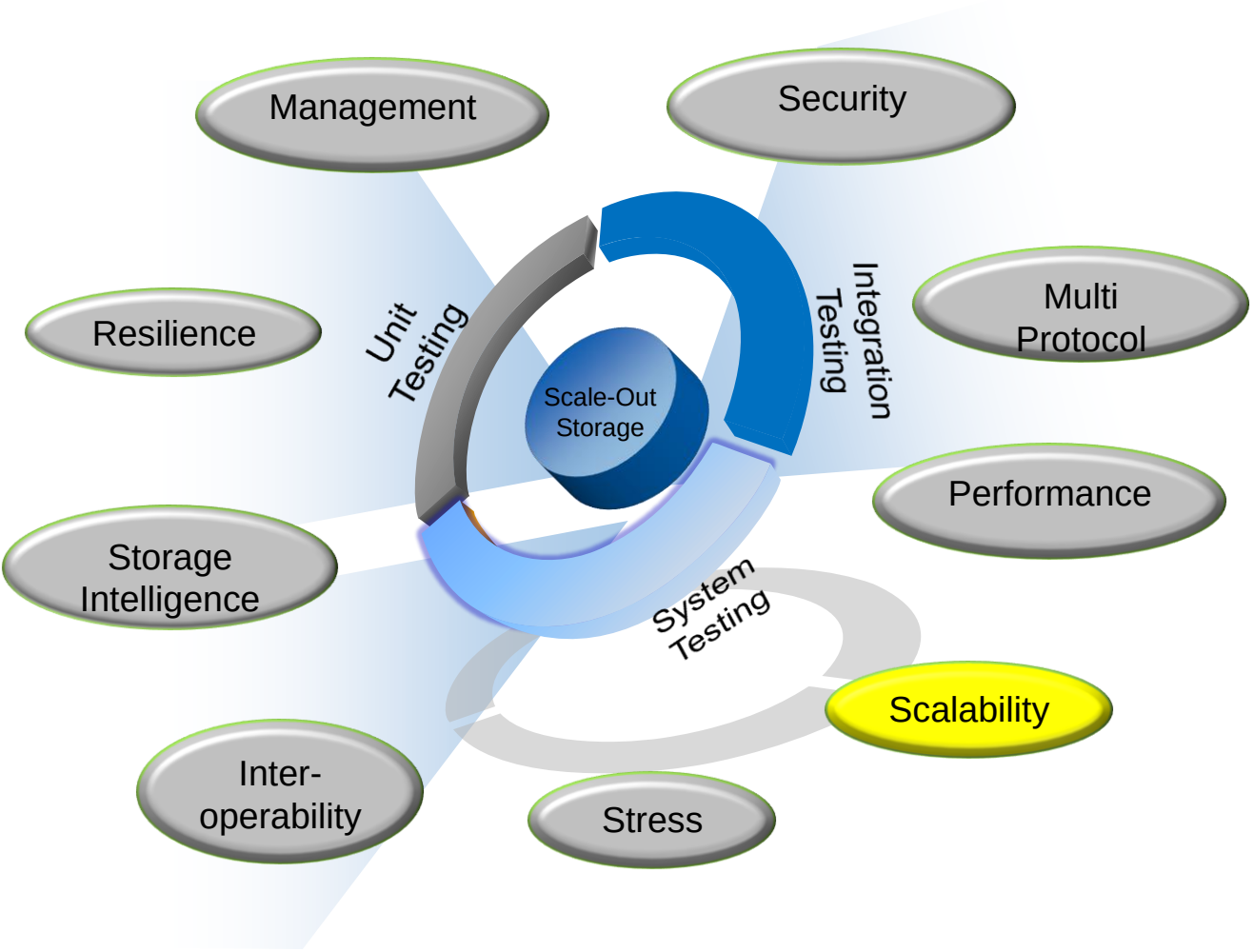
Scale-out NAS Test Strategy



Performance

- IO and User workload
- Distributed File System
- Degraded node
- Benchmarking

Scale-out NAS Test Strategy



Scalability

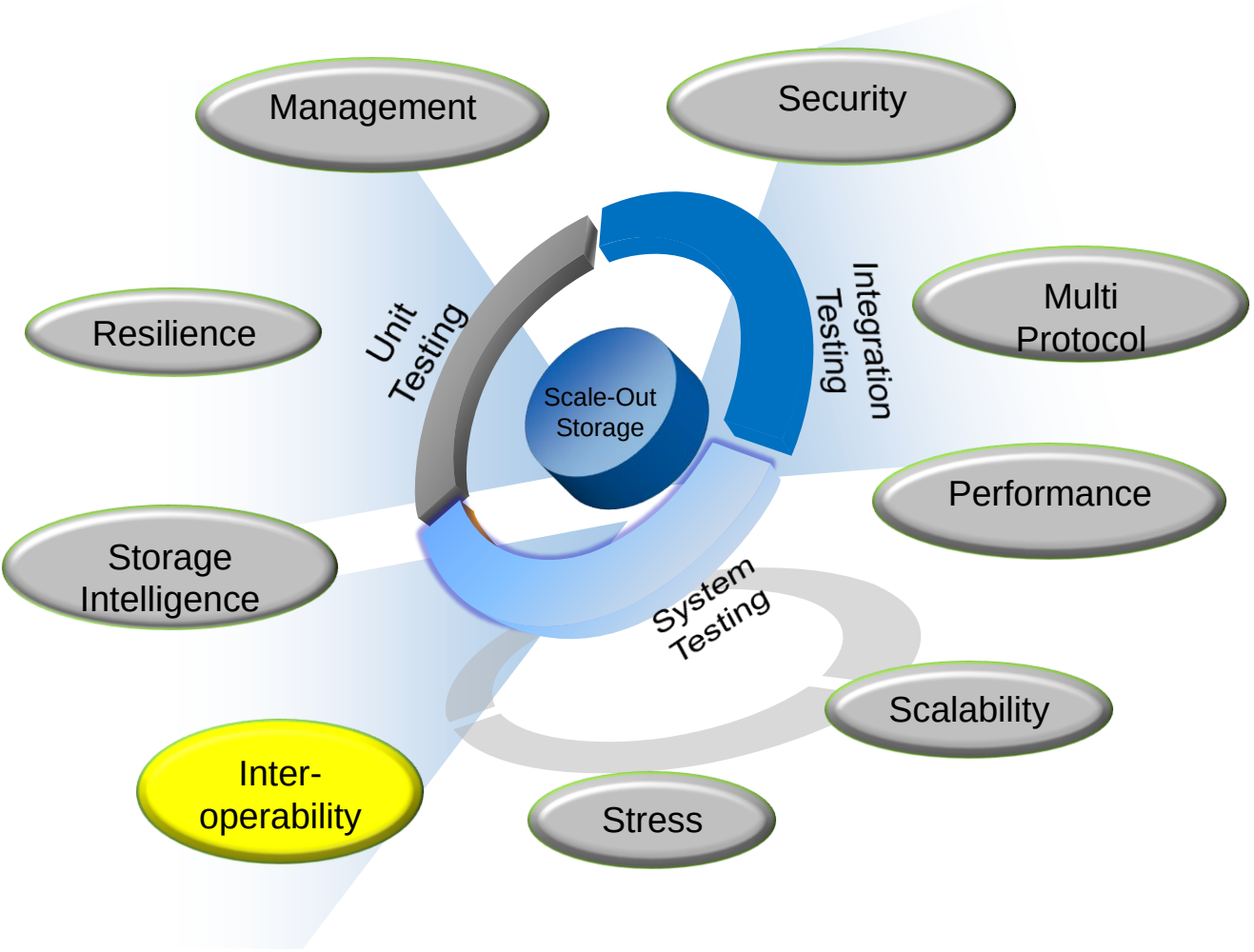
- Determine storage controller threshold for optimum performance
- Determine capacity threshold per storage controller for linear performance
- File system capacity

Scale-out NAS Test Strategy

Stress

- Simultaneous User support
- Continuous heavy IO
- File system load testing
- Degraded mode

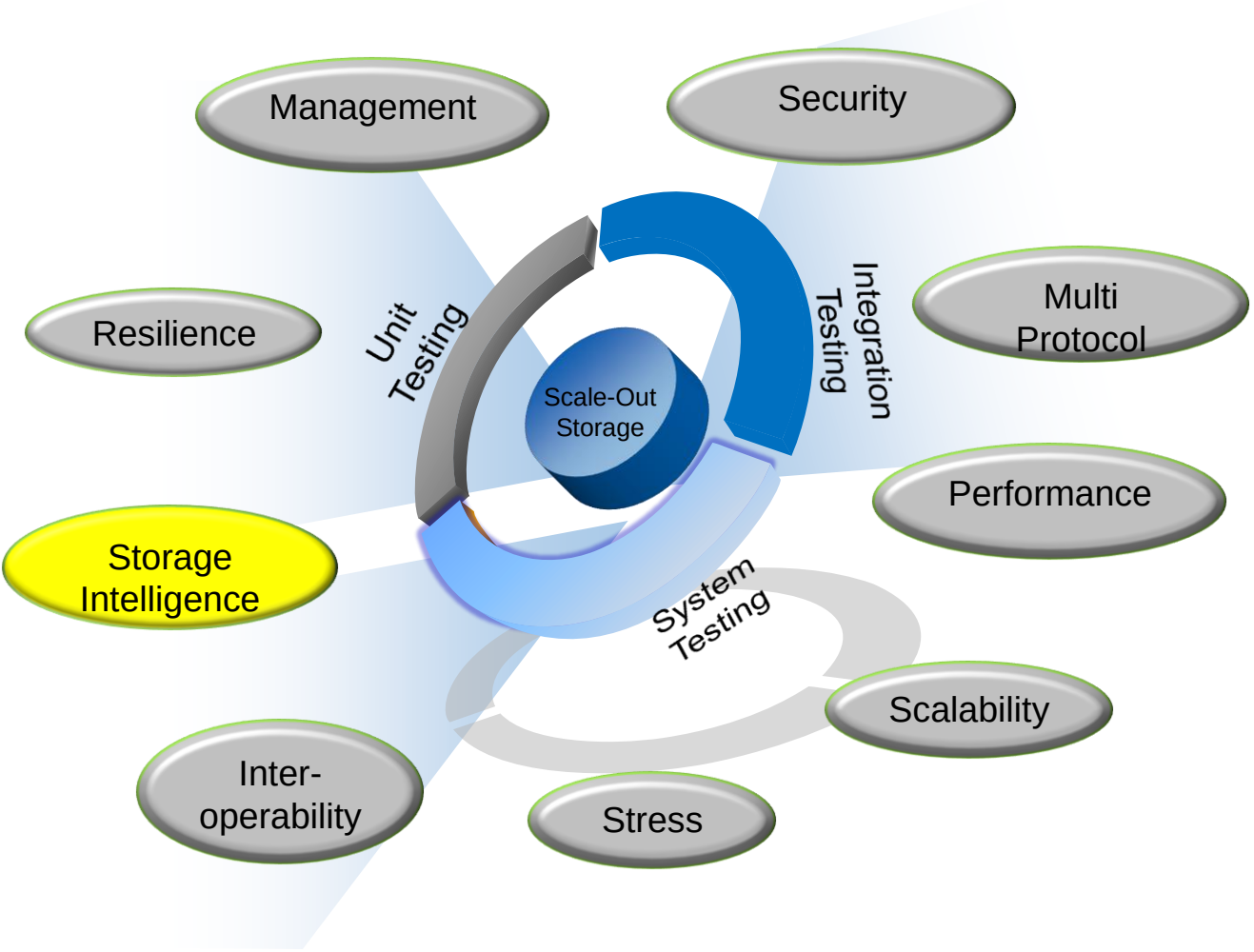
Scale-out NAS Test Strategy



Inter-Operability

- Support to third party applications
  - Backup applications using NDMP support
  - Cloud API support for different vendors

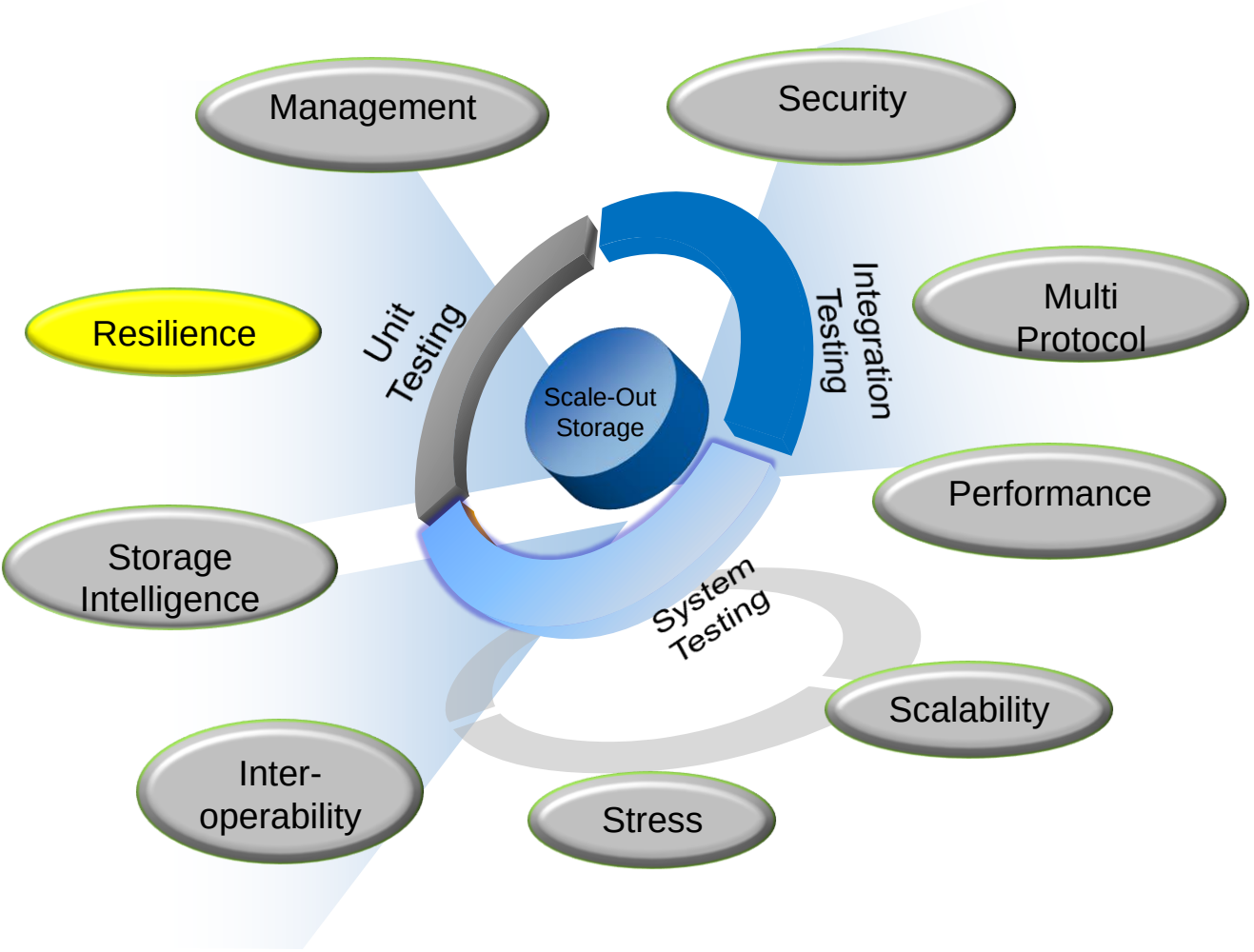
Scale-out NAS Test Strategy



Storage Intelligence

- Load balancing
- Policy based replication, storage tiering, archiving
- Compression, De duplication
- Thin provisioning

Scale-out NAS Test Strategy



Resilience

- Failover/ Failback
- Replication
- Redundant storage access path
- NIC teaming – NIC failover / failback

Scale-out Storage – Next

- Cloud integration (OpenStack etc.)
  - STaaS
  - OpenStack Swift Integration
- Big Data
  - Hadoop
  - Platform for BI Apps
- Solid State Device
  - Scale-out NAS SSD array
  - PCIe SSD cards
- Software Defined Storage (SDS)
  - SLAs & QoS
  - Orchestration & Storage Hypervisor
- Healthcare or other Domain/Industry
  - Online data, history, diagnosis, medication, billing, etc.

Thank You!
Questions