STORAGE DEVELOPER CONFERENCE
SDC 21
BY Developers FOR Developers
Virtual Conference
September 28-29, 2021

Uncovering production issues – with Real World Workload emulation

Byju Ravindran, Swati Chawdhary
Samsung Semiconductor Research India

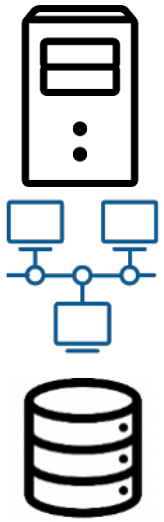
Agenda

- Storage architectures and trends
- Benefits of SDS and Composable architecture
- Data center environment
- Enterprise storage workloads
- Our test methodology

Infrastructure evolution and impact on Storage

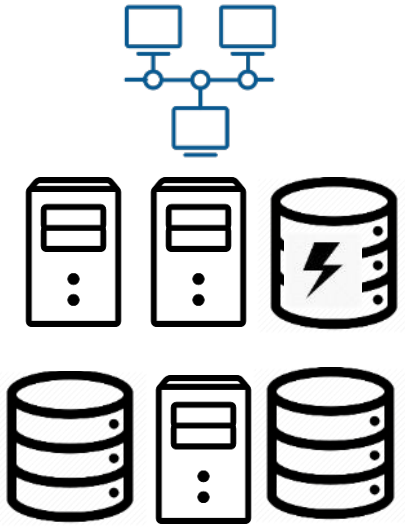
Storage Architecture trends

Traditional



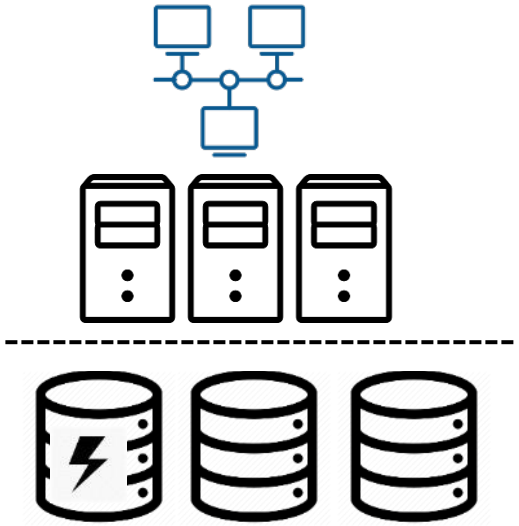
- Established technology
- Proven reliability
- Lack of central mgmt.
- Inefficient
- Inflexible

HCI



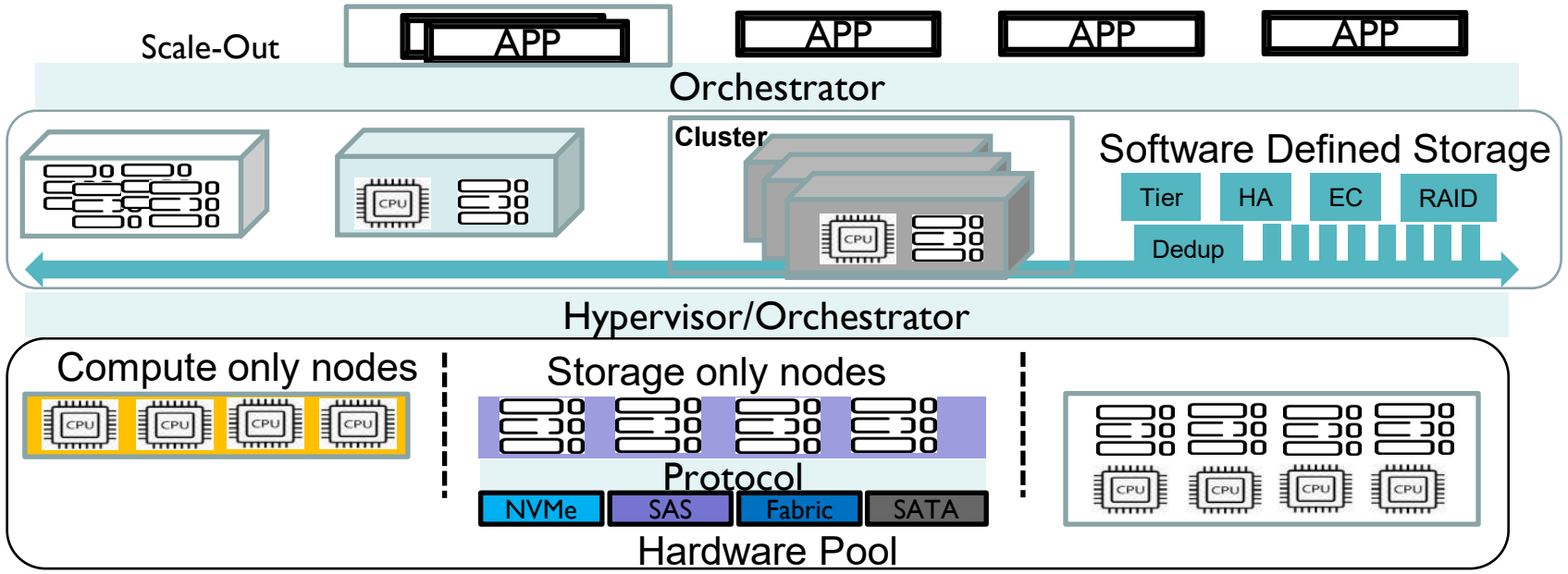
- Economic
- Easy to manage- Rapid Deployment
- Commodity box (vendor specific)
- Fixed Building blocks

Composable



- App centric
- Pools (Disaggregated)
- Lower Capex/Opex
- Scale-out architecture

Cloud Storage architecture

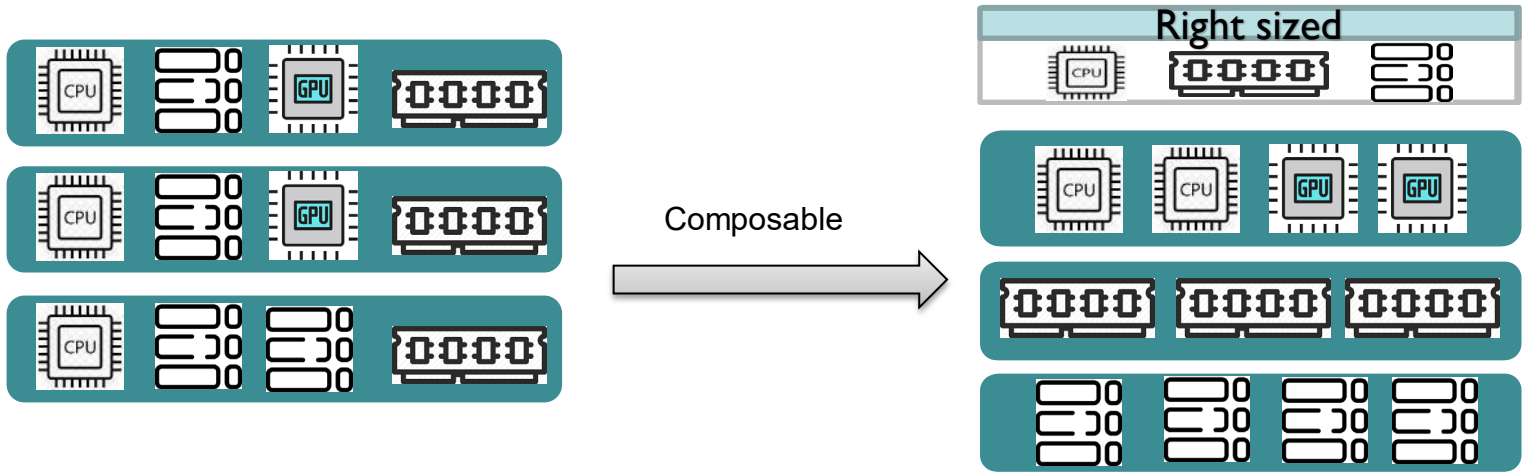


- ✓ Anything as a service
- ✓ Shared resources
- ✓ Self-contained and stateless
- ✓ Virtual everything
- ✓ Compose based on usecase
- ✓ Scale on demand
- ✓ Centralized control

Benefits of Software Defined Storage

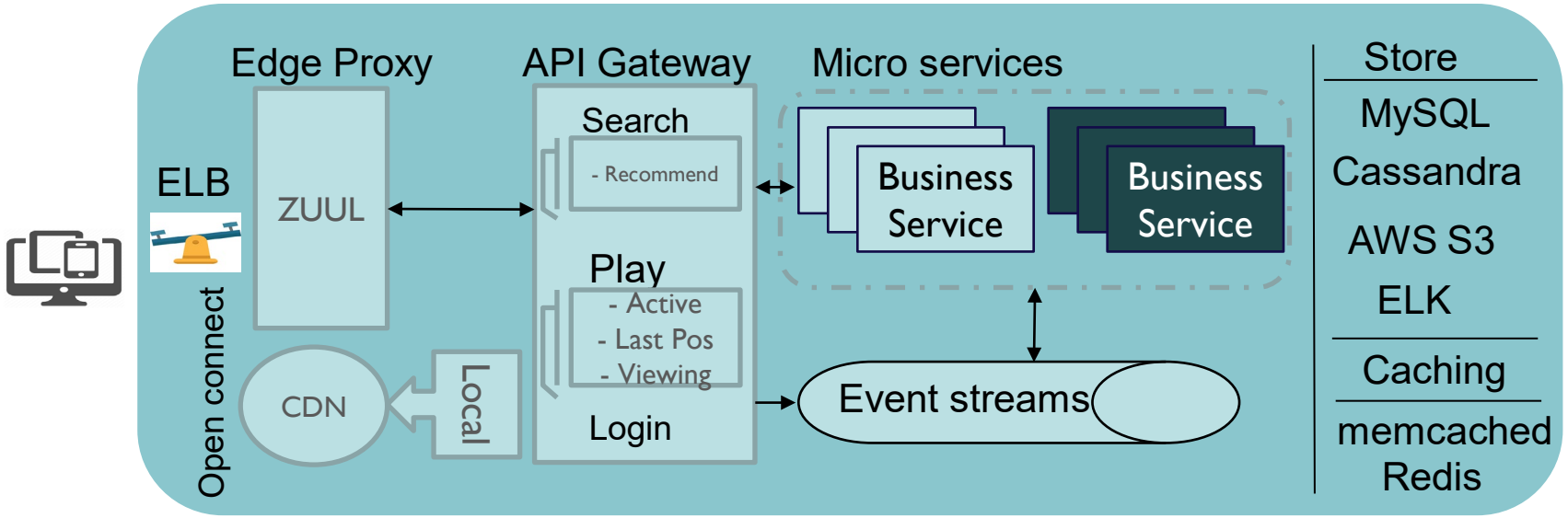
- Cloud native:
  - Cloud native deployment - Availability, Scalability, Performance(Distributed), Consistency, Durability
  - Plethora of storage services – Thin provisioning, Dedup, Replication, Zone(HA), Self healing
  - Unified platform – No vendor lock, Multiple drivers supported, Platform agnostic , Resource aggregation
  - Centralized API driven storage consumption - On-demand storage resources
    - Abstract the underlying storage complexities
    - Whole storage lifecycle management (Creation, access, destruction)
  - Industrywide collaboration and development
- App centric:
  - Storage characterization using policy based IO – QoS, tiering, Provisioning, data protection, life cycle management
  - Profile based workflow orchestration – Provision/Decompose, consume, move/grow/shrink and manage
  - Real – Focus on real world requirements
- Simple
  - Ease of deployment (containerized / Virtual environment)
  - Focus on Application development and not worry of deployment

Why is composability needed?



- 4Vs – Value, Velocity, Variety, Volume
- Diversity of application workloads
- Avoid Underutilized resources
- Application cannot generate so much IO that a device can consume
- Server limitations (slots, config etc)
- Speed of deployment and scale

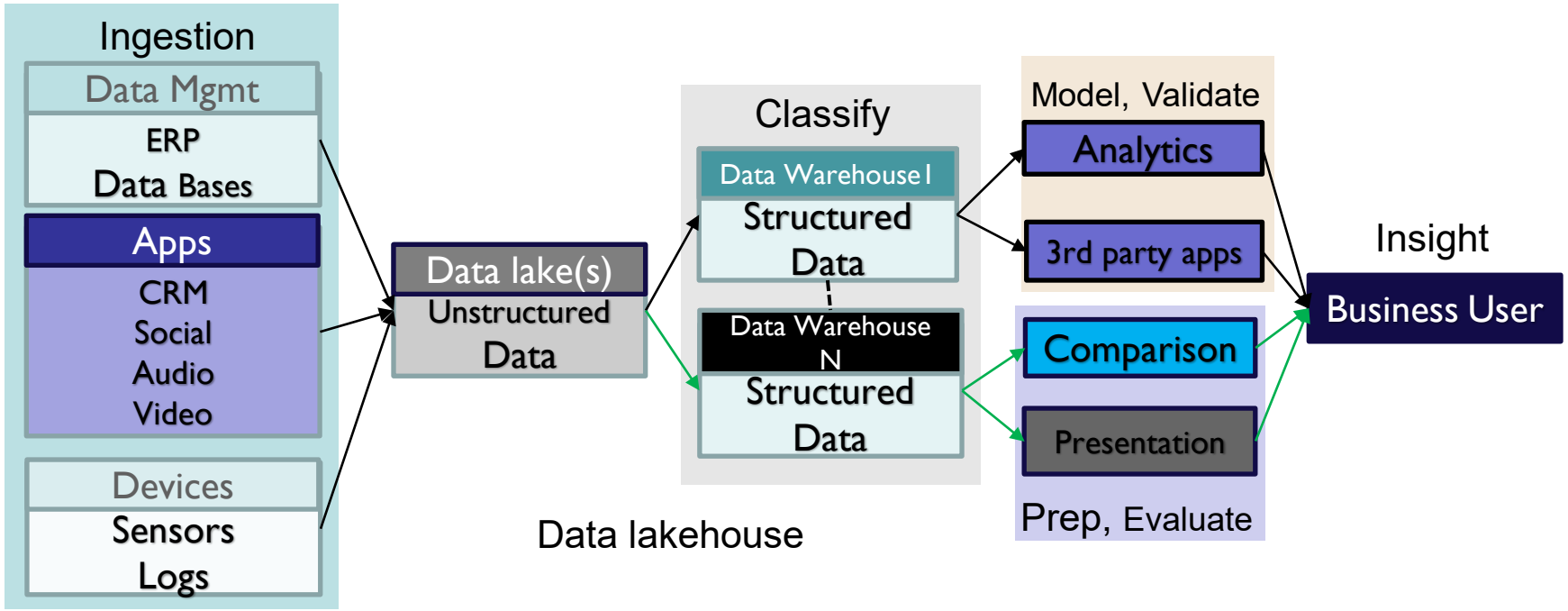
Data Center Environment



- Multi tenancy
- Multiple streams
- On demand micro-services
- Multiple Storage endpoints
- Multiple Storage services(SDS)
- Real time analytics

Source: Netflix

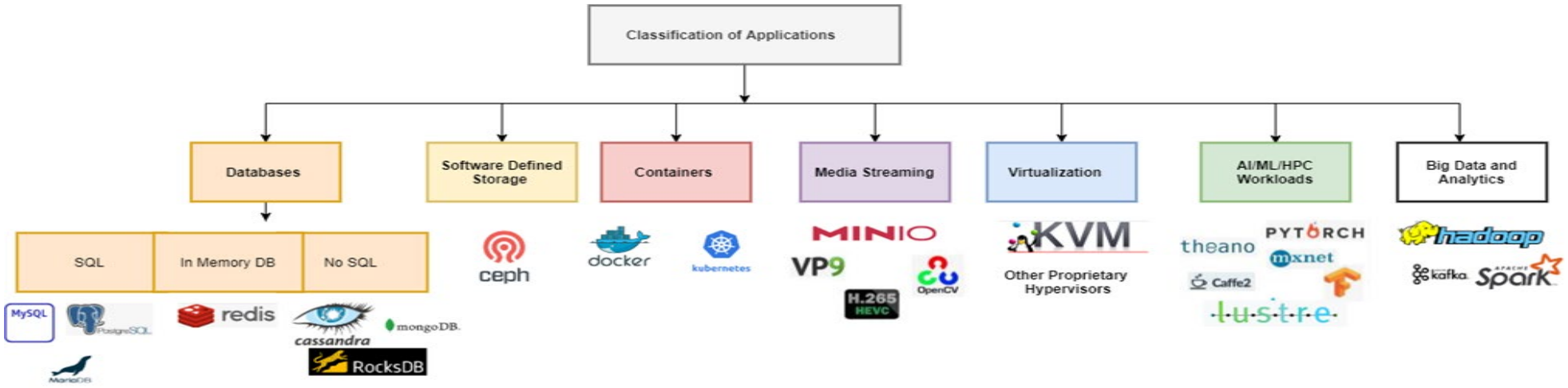
Data pipeline commonalities across data centers



- Multiple sources (ever growing)
- Multiple consumers
- Multiple workloads (App based)
- Multiple streams from the same data source
- Multiple storage points
- Data undergoes a lot of change

Enterprise Storage Workloads and Our Test Methodology

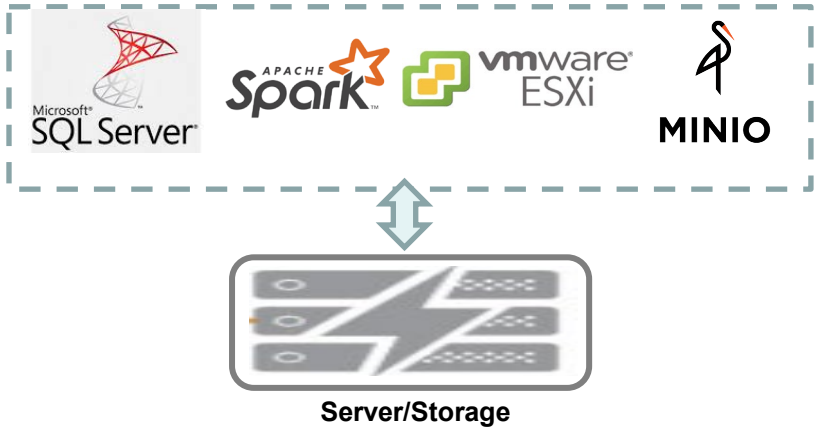
Enterprise storage workloads



Primary	Online Transaction Processing (Databases)
	Server Virtualization
	VDI
	Email/Messaging
Secondary	Backup and recovery
Emerging	Big Data/Analytics
	High-Performance Computing
	Media and entertainment
	Cloud-based services
	AI/ML

Online Transaction Processing(OLTP)	~8k block size 25% read, 75% write ~80 % random
Server Virtualization	Mostly Random (due to I/O blender effect in a virtualized environment) Read/Write% and Block Size depends upon the application running on the VM
Online analytical Processing (OLAP)	~64k block size 75% read, 25% write ~80% sequential

How Real World Workloads look like?



- Real World Enterprise Storage workloads consist of **multiple mixed** applications like Databases, Analytics Platforms, Server Virtualization, VDI etc.
- Typically these applications are deployed on virtualized hypervisor based platforms
- Each application has a **unique I/O request profile**
- The **IO access pattern changes** at each layer of software abstraction such as - operating system activities, fragmentation, merging, compression, encryption, deduplication, virtualization etc.
- I/O generated to storage is a **constantly changing combination of IO access patterns** with varying Queue Depths, Idle times, IO bursts and IO block size

Enterprise Storage Test recommendations

✓ **Emulate data center environment and applications as close to production, to uncover complex issues**

- Create tests using generic data sets, data streams, and workflows designed to model real world workloads
  - Acquire historical production data for workload profiling
  - Simulate the I/O profiles of a production environment
- Test with locality and hot spots
- Test with enterprise features like deduplication, compression, snapshots, clones and so on
- Test performance impact of virtualization, scale out etc.
- Generate load to adequately stress the performance capabilities of the storage

× **Corner tests and benchmarks are not enough**

Limitations of testing with standard tools

Iometer VDBENCH FIO

- Traditional benchmarking tools used for testing block storage
- Designed for measuring read/write performance of HDD disk drives
- Single or few I/O access patterns
- Cannot realize production environment test patterns
- No micro burst control

Block Testing
Functional Tests

YCSB HammerDB SysBench

- Benchmarking tools for testing applications like databases etc.
- Synthetic tests
- Predefined tests with limited control
- Limited load generation
- Tedious task of preparing scalable load test environment

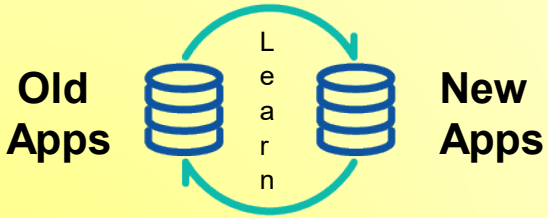
Application Testing
System / Integration Tests

Enterprise test challenges and solution

- Continuously evolving application workloads
- Learning IO patterns from newer/modern applications is highly time consuming
- Testing a broad range of storage products at once is a big challenge
- Setting up data center software stack takes long time
- Big challenge to generate, debug and reproduce workloads

A Test Solution that enables:

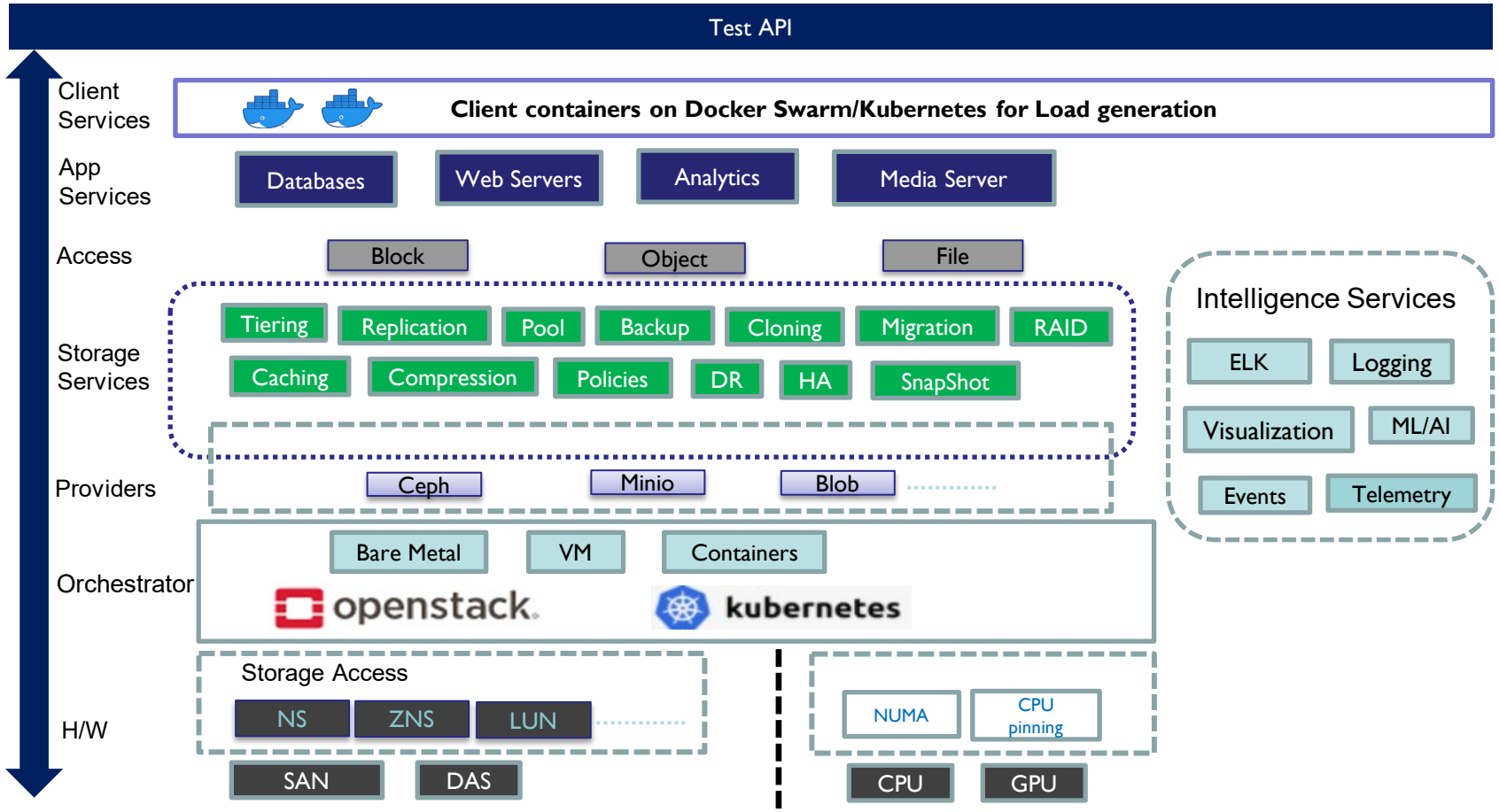
- ✓ Real world emulation/simulation
- ✓ New Applications/Workloads targeted for enterprise flash storage
- ✓ IO Patterns that standard tools can't create fast enough



Mix-Application Testing

Real world Tests

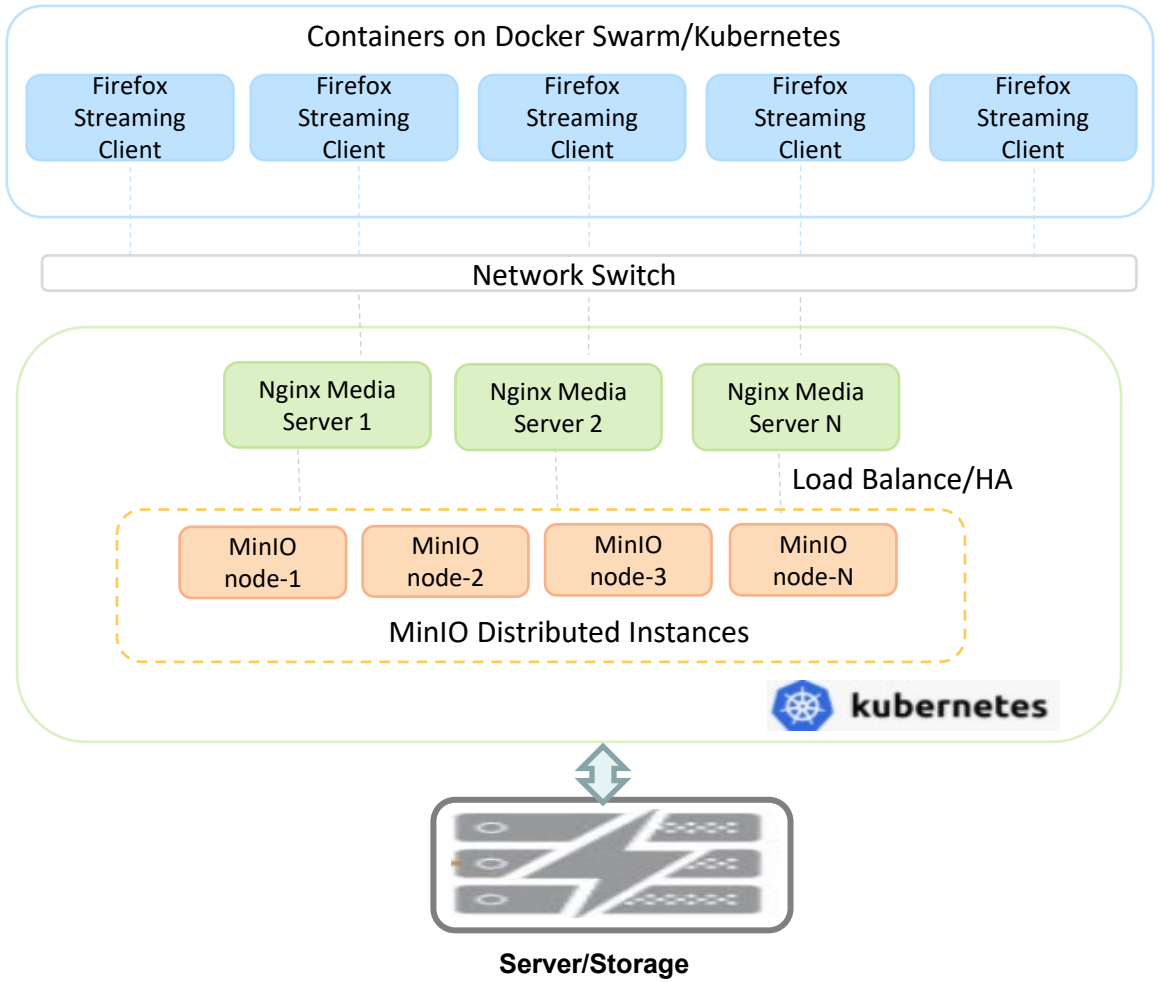
Test Framework



Framework features

- ✓ Composability
  - H/W composability with Openstack
  - Container Orchestration with Kubernetes, Docker
- ✓ SDS Providers
  - Support for multiple providers(Ceph, Minio etc.)
- ✓ Support for diverse workloads
  - NoSQL/SQL Databases, Analytics applications, Streaming applications
- ✓ Load generation
  - Client services on Docker Swarm cluster
- ✓ Log collection and Visualization
  - Workload analysis with blk_trace
  - Log collection and disk statistics collection with logstash and metricbeat
  - Visualization with ELK Kibana
- ✓ Workloads supported
  - Online Transaction Processing
  - Big Data/Analytics
  - Media Streaming
  - Video Surveillance
  - Machine Learning

Media Streaming Workload



Observations -

- ✓ Emulation datasets are small, in comparison to huge Real world datasets
- ✓ Storage utilization is low, when the entire emulation data set fits into system memory. The dataset is read from storage only once and subsequent reads, it is read from system memory
- ✓ Limiting resources(memory etc.) of the VMs/Containers, helps in accurately modeling Real world environment
- ✓ Resulting in drastic increase in the storage utilization

Real World I/O Access pattern

Workload	Access Pattern	Random/ Sequential	Block Size	Read/ Write	IO Count
Media Streaming on MinIO	RND 4K R	SEQ	4096	R	3598
	RND 16K R	SEQ	16384	R	256
	RND 32K R	SEQ	32728	R	101
	RND 64K R	SEQ	65536	R	129
	RND 128K R	SEQ	131072	R	99163
	RND 256K R	SEQ	262144	R	130

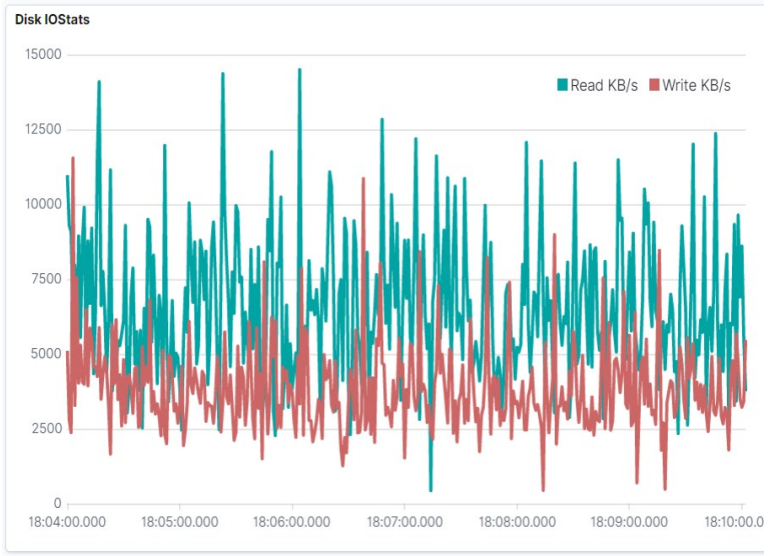
Media streaming on Minio Object Store

Application	Access Pattern	Random/ Sequential	Block Size	Read/ Write	IO Count
MySQL	RND 16K R	RND	16384	R	1213
	RND 4K W	RND	4096	W	1515
	RND 8K W	RND	8192	W	368
	RND 12K W	RND	12288	W	215
	RND 16K W	RND	16384	W	1279
	RND 20K W	RND	20480	W	49
	RND 24K W	RND	24576	W	31
	RND 28K W	RND	28672	W	15
	RND 32K W	RND	32768	W	20
	RND 36K W	RND	36864	W	16
	RND 40K W	RND	40960	W	15
	RND 256K W	RND	262144	W	12
	RND 1024K W	RND	1048576	W	4

Web Store on MySQL Database

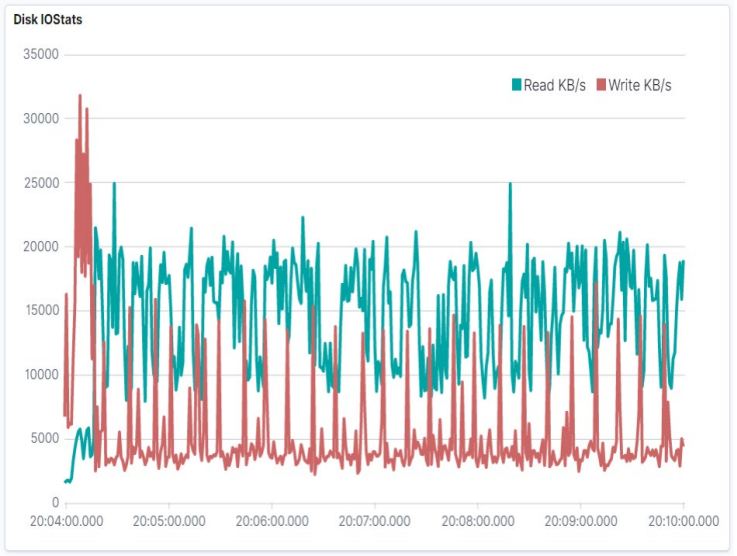
Diverse I/O access patterns created by Real World Applications

Heavy Load generation



2,321.955 Total Read MB
1,365.081 Total Write MB

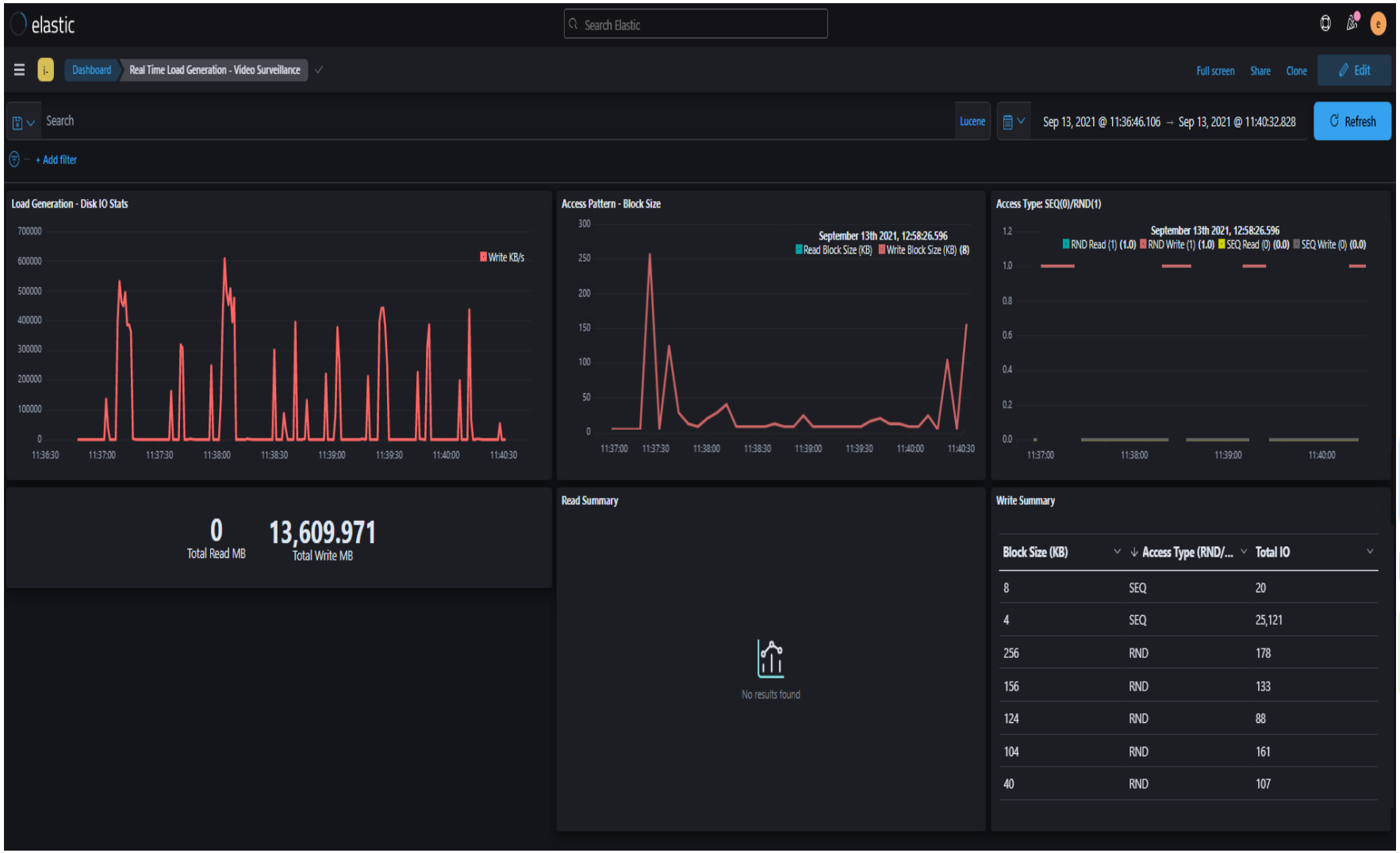
Load generation by HammerDB
On a HammerDB warehouse on MySQL

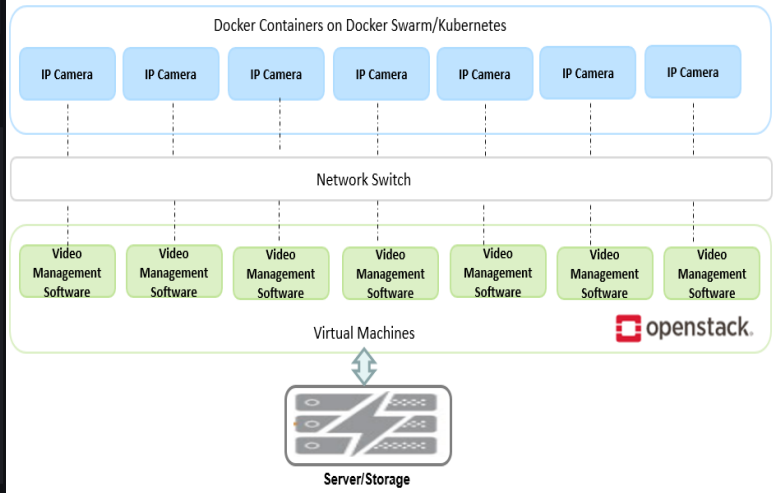


5,090.703 Total Read MB
1,892.047 Total Write MB

Load generation by Jmeter docker services
On a Webstore on MySQL

Video Surveillance Workload





Machine Learning Workload

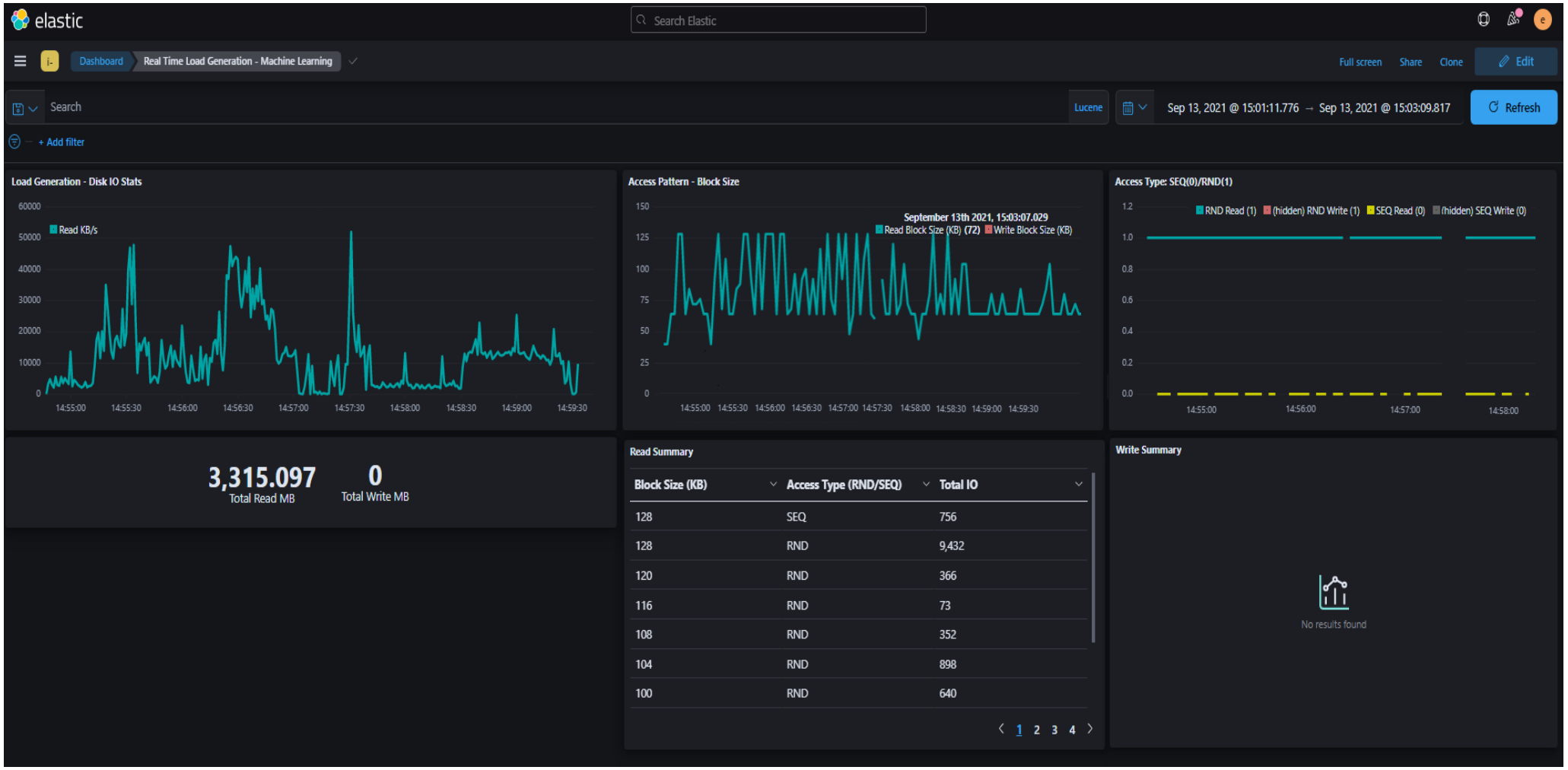


Image Classification Using the ImageNet dataset containerized on top of our test framework

Conclusion

- Continuous growth in storage technologies and applications call for new methodologies for testing enterprise storage
- Standard benchmarking tools cannot generate real world application I/O access patterns and are insufficient for testing enterprise storage
- With Containerization and SDS, it is possible to emulate real world behavior of enterprise applications
- Our test environment helps to mimic real world workloads and identify real world defects in early development stages

Future Work

- Enable real world I/O Profiles – For e.g. Workload Modeling with load generator parameterization
- Identify other client load generation tools and enable scale out testing using containerization
- Enable additional workloads like Server Virtualization/VDI etc.
- Enable and Support SDS features like deduplication, compression, snapshots, clones, replication and so on

References

https://www.marketsandmarkets.com/PressReleases/software-defined-storage-sds.asp

https://medium.com/@narengowda/netflix-system-design-dbec30fede8d

https://www.gartner.com/doc/reprints?id=1-1OHW336I&ct=190913&st=sb&submissionGuid=821e7a40-6728-4124-a4cd-7a66a865fe90

https://www.snia.org/sites/default/files/SSSI/SNIA%20Tutorial%20Understanding%20Real%20World%20Storage%20Workloads%20v0.2.pdf

https://www.marketresearch.com/IDC-v2477/AFA-Performance-Testing-Framework-11363242/

Acknowledgements

- Madan Udaykumar
- Sagar Kumar Sahu
- Amit Devgan
- Sandeep Yadav
- Sandeep Agarwal

Thank You!

Please take a moment to rate this session.

Your feedback is important to us.

