STORAGE DEVELOPER CONFERENCE
SDC 22 | Fremont, CA
September 12-15, 2022
BY Developers FOR Developers

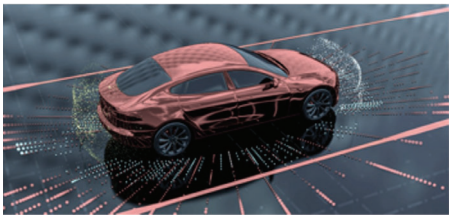
A SNIA Event

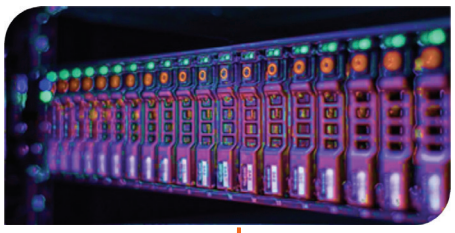
The Path to Autonomous Storage is Broken

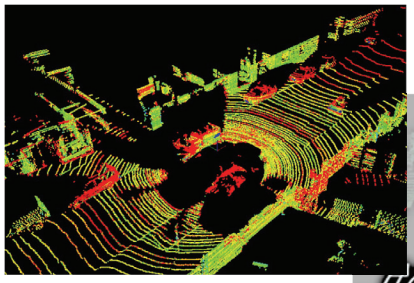
Presented by Eric Wright (@DiscoPosse)
Technology Advocate, Magnition.io

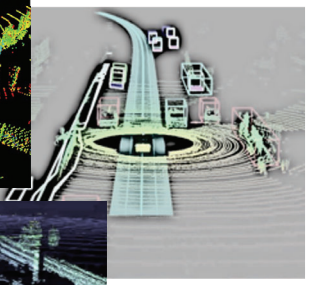
What is preventing the industry from achieving fully **autonomous** storage?

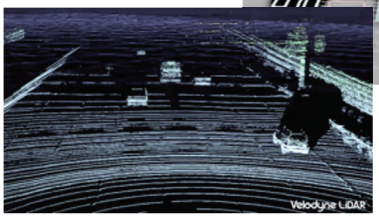
Multi-Dimensional Challenges

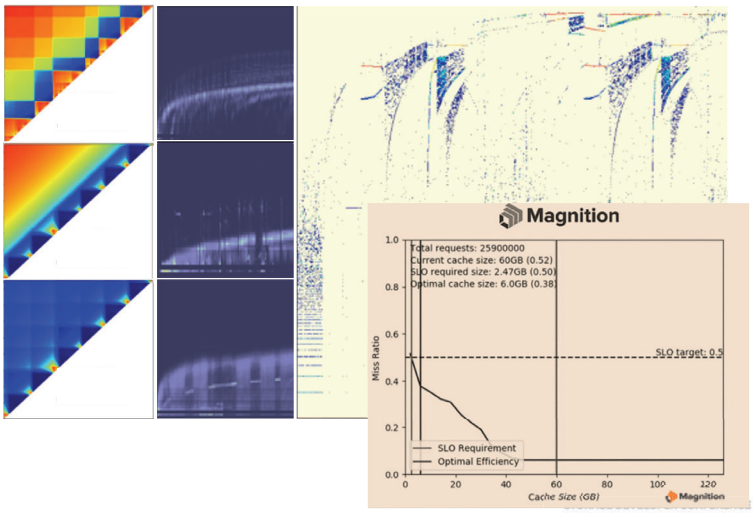












SDC 22

Manual Storage / Memory Management Now Infeasible

Applications and data requirements changing hourly

Increasing hardware complexity

Manually-managed Storage / Memory Infrastructure

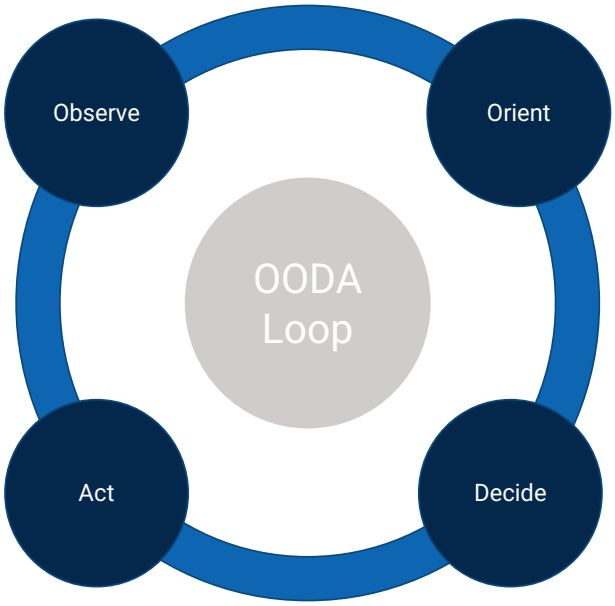
Vulnerable to:

- Thrashing, Scan pollution
- Gross unfairness, Interference
- Unpredictable availability
- Data loss risks

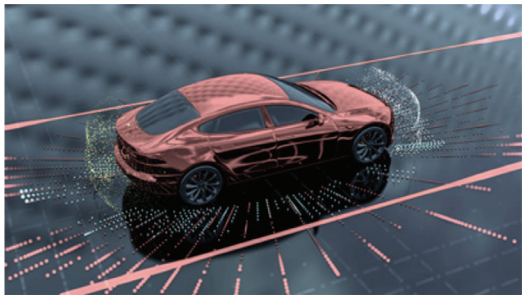
⇒ **Overprovisioning**
⇒ **Lack of Control**
⇒ **Availability & Durability Risk**

How do we cross the chasm to fully autonomous storage and memory hierarchies?

Autonomous Systems Require OODA Loops & Models



Autonomous Storage ML/Models Needed

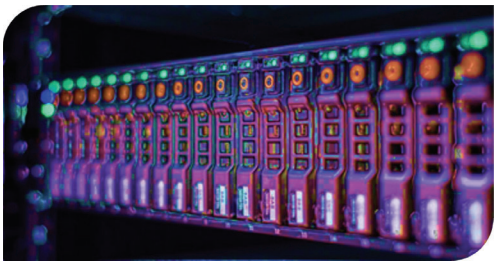


Self-Awareness

Acceleration, braking steering, roll, wear/tear, weight distribution, battery discharge temperature and load models

Environment Awareness

Maps, static obstacles, dynamic obstacles, object capabilities, terrain, distances, relative object velocities, live traffic, GPS, road conditions, weather, law enforcement, etc.

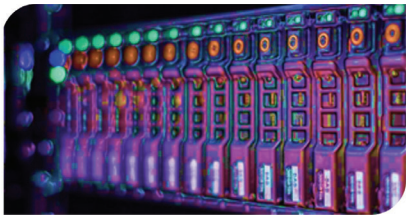


Self-Awareness

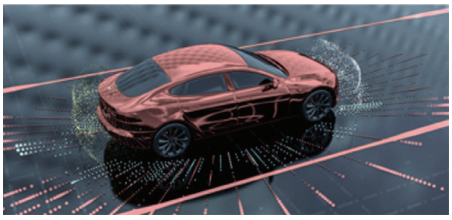
Caches, memories, disks, data paths, latencies, link throughput limitations, media costs, data movement costs, performance capabilities, degraded performance, etc.

Environment Awareness

Dynamic workloads, QoS constraints, competing traffic on links, dynamic IaaS costs, failures, imminent failures, flash wear/tear, power constraints, temperature, dynamic resource costs, etc.



Autonomous Levels

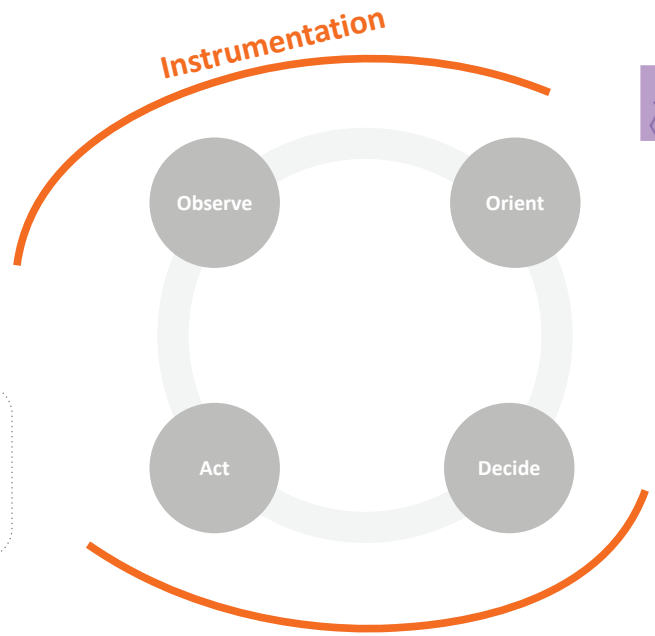


Admin controls the storage device; device can detect and send alerts, etc.	Level 1 **Operator Assistance**	Driver controls the car; car can alert driver to conditions, obstructions, etc.
Automated failure repair, backup, replication, recovery; Admin remains engaged	Level 2 **Partial Automation**	Automated acceleration, steering; Driver remains engaged
Device manages many cost / performance tradeoffs; Admin must be ready to take over	Level 3 **Conditional Automation**	Car manages most safety driving functions; Driver must be ready to take control
Device guarantees QoS constraints at lowest cost, is self-aware, self-troubleshooting; Admin has option to control	Level 4 **High Automation**	Car capable of performing all safety-critical functions; driver has option to control
Device is completely lights-out, hands-off, no control UI, only high-level policy controls; Admin install spares when instructed	Level 5 **Fully Autonomous**	Vehicle completely driverless, no driving equipment (e.g. steering)

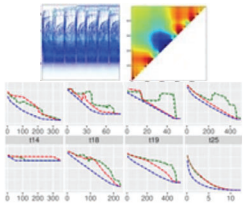
Architecture for Fully Autonomous Storage

Apps & Microservices

Storage, Filesystem, Dedup, Compute, Caches



Model



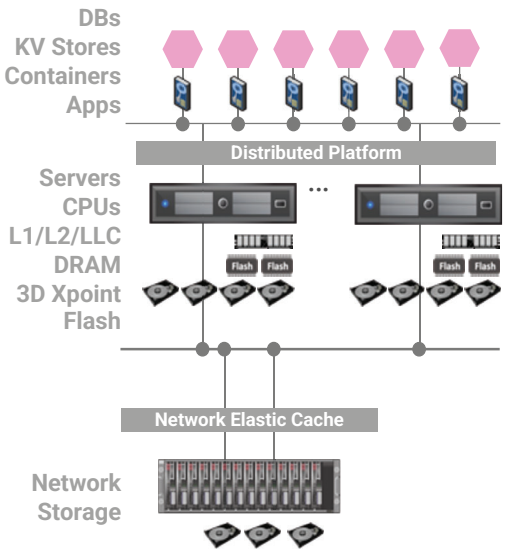
Predict

Enforce SLA
Optimize
Monitor

Control

Auto-tune, Auto Size, Reconfigure, QoS Control

Fully Autonomous Storage / Memories are Self-Aware



Fully Autonomous Storage Needs Must Continuously Answer

- Is this performance good?
- Can performance be improved?
- How much Cache for App A vs B vs …?
- What happens if I add / remove DRAM?
- How much DRAM versus Flash?
- How to achieve 99%ile latency of X µs?
- What if I add / remove workloads?
- Is there cache thrashing / pollution?
- What if I change cache parameters?

Use Case #1: Autonomous QoS SLA

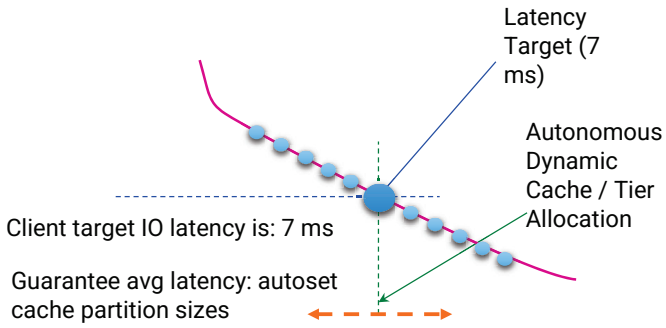
How?

- Users dial-in latency or throughput target and budgets
- Fully Autonomous Storage auto allocates just enough capacity to meet SLAs at all times

Value for Customer

- Automated SLA achievement!
- Set and Forget, ease of mind
- Revenue disruption avoidance
- Improved margins
- Zero OpEx performance scaling
- Dramatically reduced service interruptions

Latency Guarantees



Use Case #2: Autonomous Cost / Performance Optimization

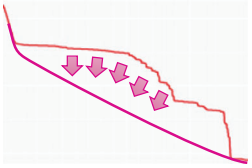
How?

- Real-time workload modeling
- Resource allocation predictions
- Dynamic resource adjustment and isolation
- Auto right-sizing

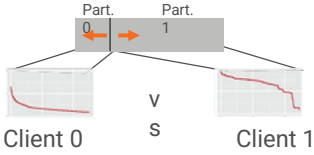
Value for Customer

- Lowest total cost of ownership (TCO)
- Eliminate noisy neighbor problems
- Policy-driven operations
- Lower OpEx for infra teams
- Predictive planning

Cache Size & Latency Reduction
(Thrashing Remediation)

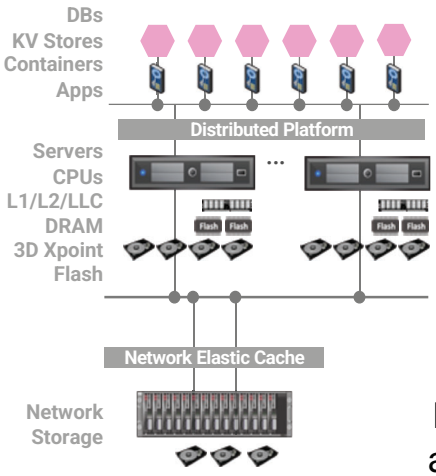


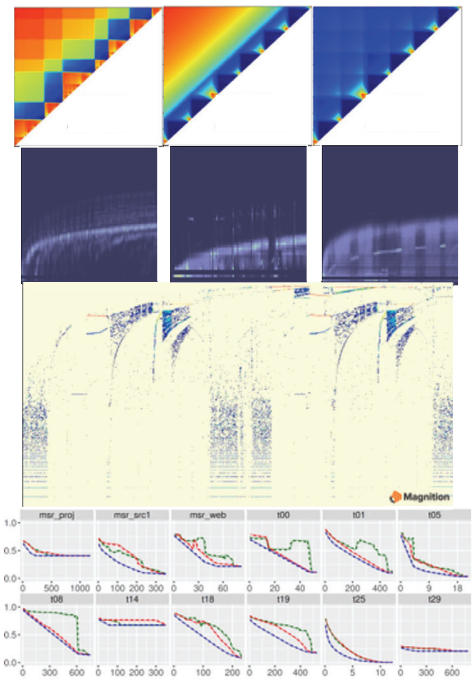
Tenant Isolation




FLEX TAPE
FLEX TAPE
FLEX SEAL

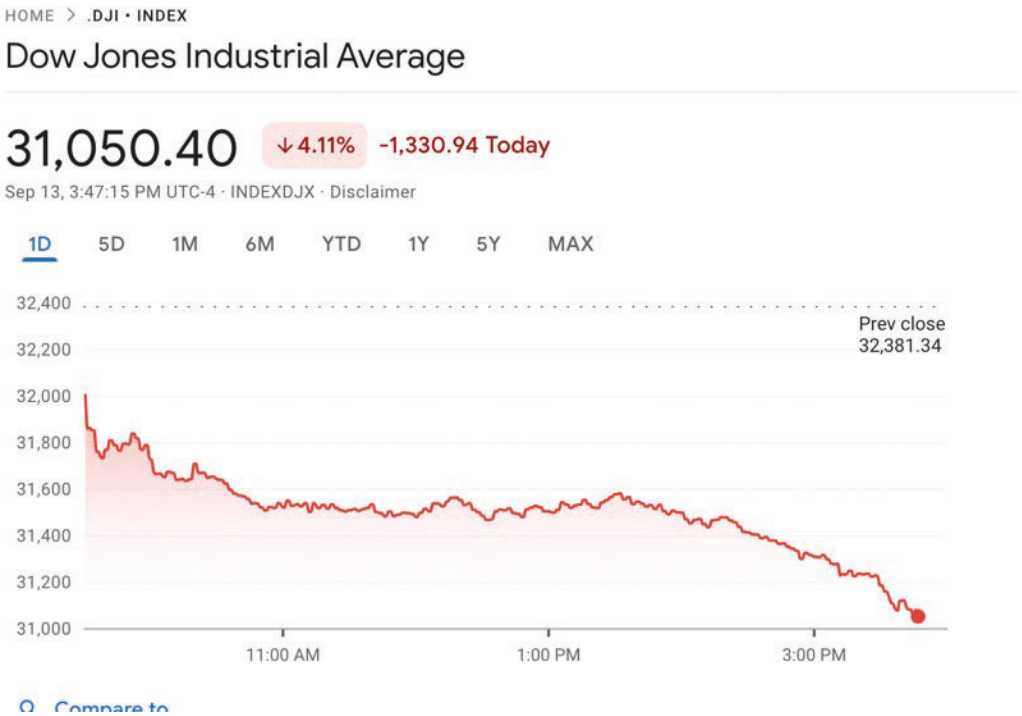
Modeling Storage Performance in Real-Time



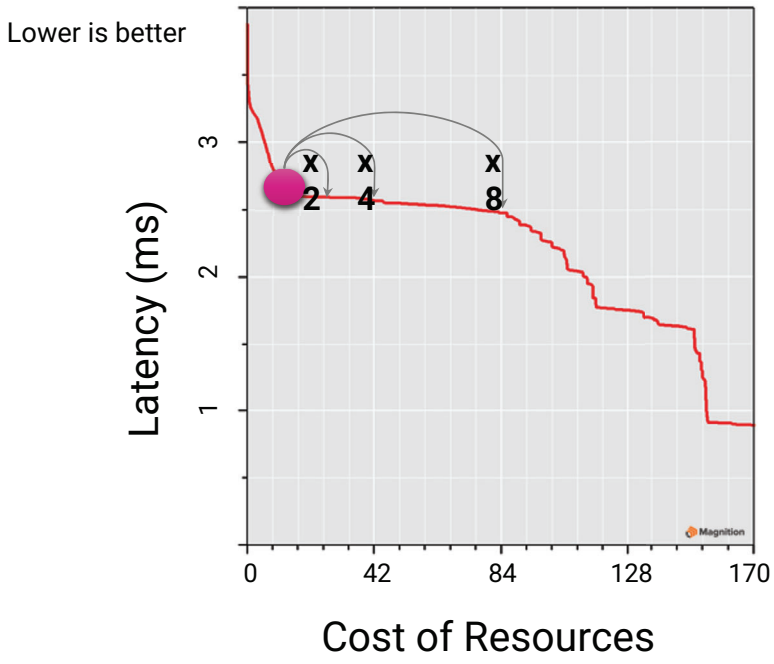


Learn performance model of applications and storage system

Predict the performance of workload as *f(resources, params)*



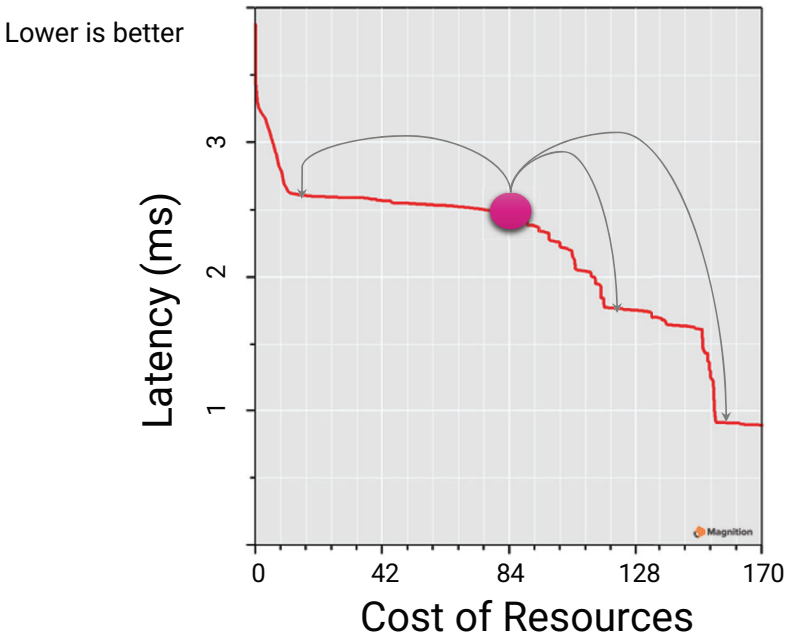
Understanding Autonomous Performance Models



Models help decide useful increments of change.

In this example, no benefit despite an 8x increase in budget.

Understanding Autonomous Performance Models



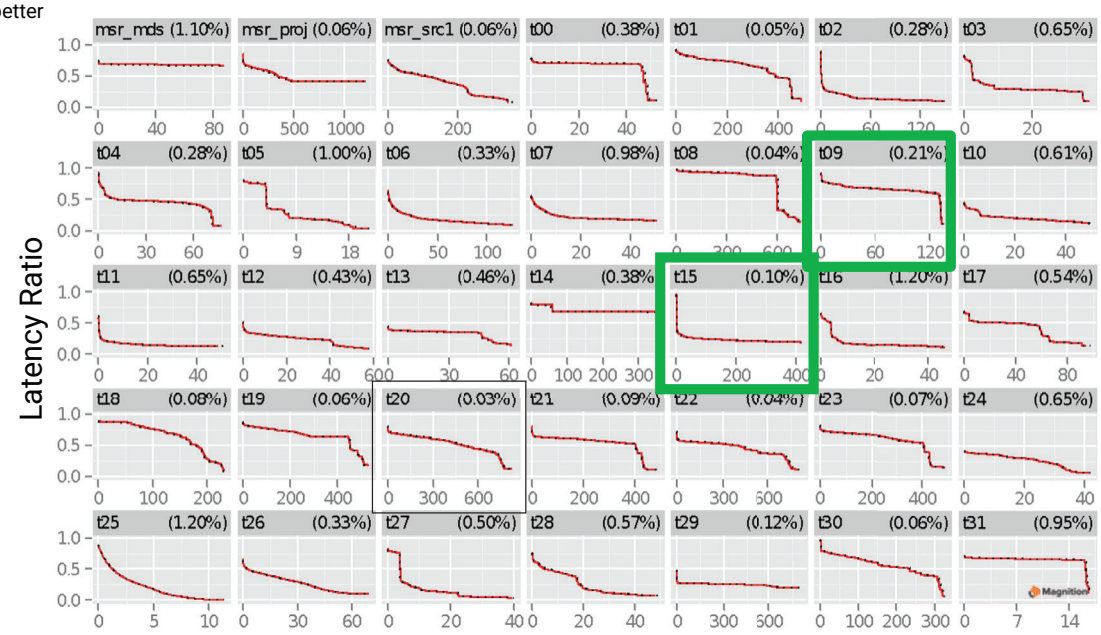
Often, most operating points are highly inefficient.

This system is operating at the lowest ROI point; equivalent performance to 1/8 the budget.

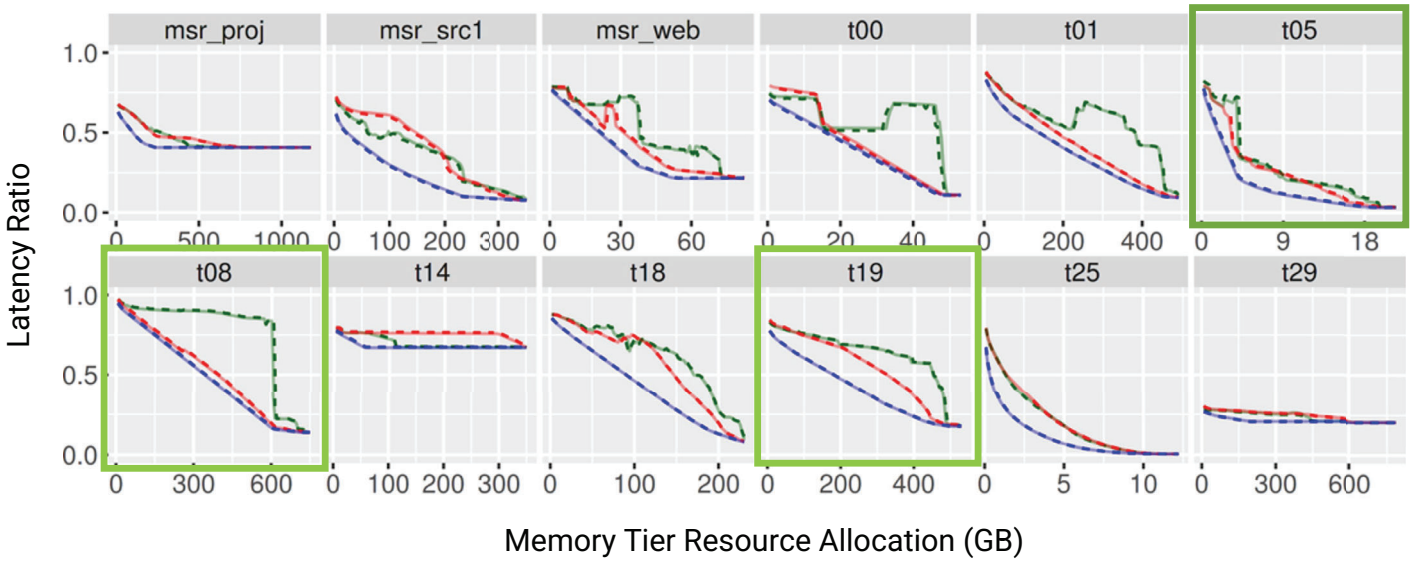
Autonomous memory hierarchies should pick efficient operating points.

Sample Models For Production Applications

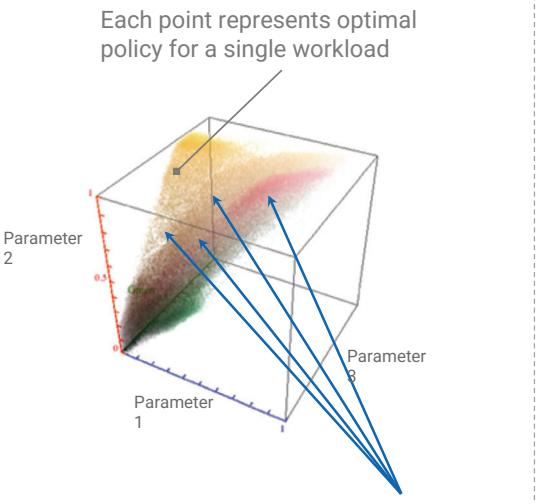
Lower is better



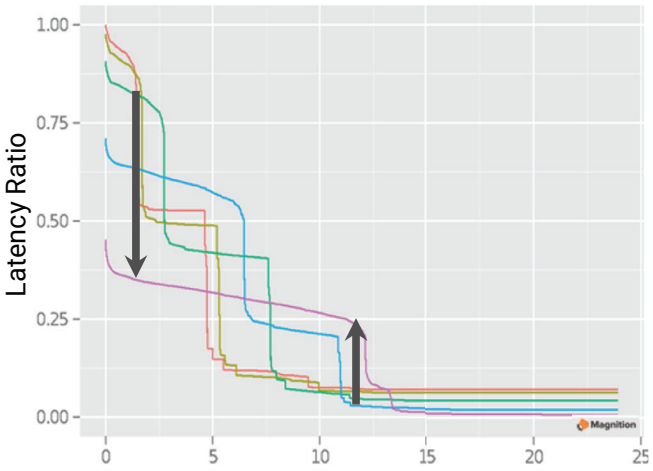
Production Applications with Different Performance Policies



Understanding Fully Autonomous Adaptation



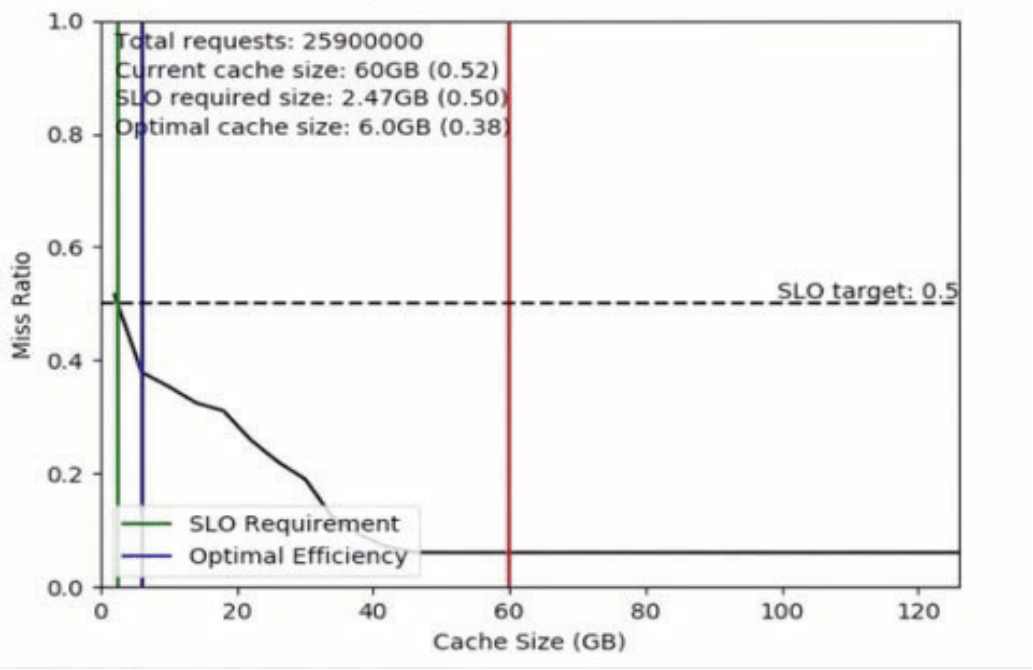
Workloads have dramatically different *optimal* policy and parameters.



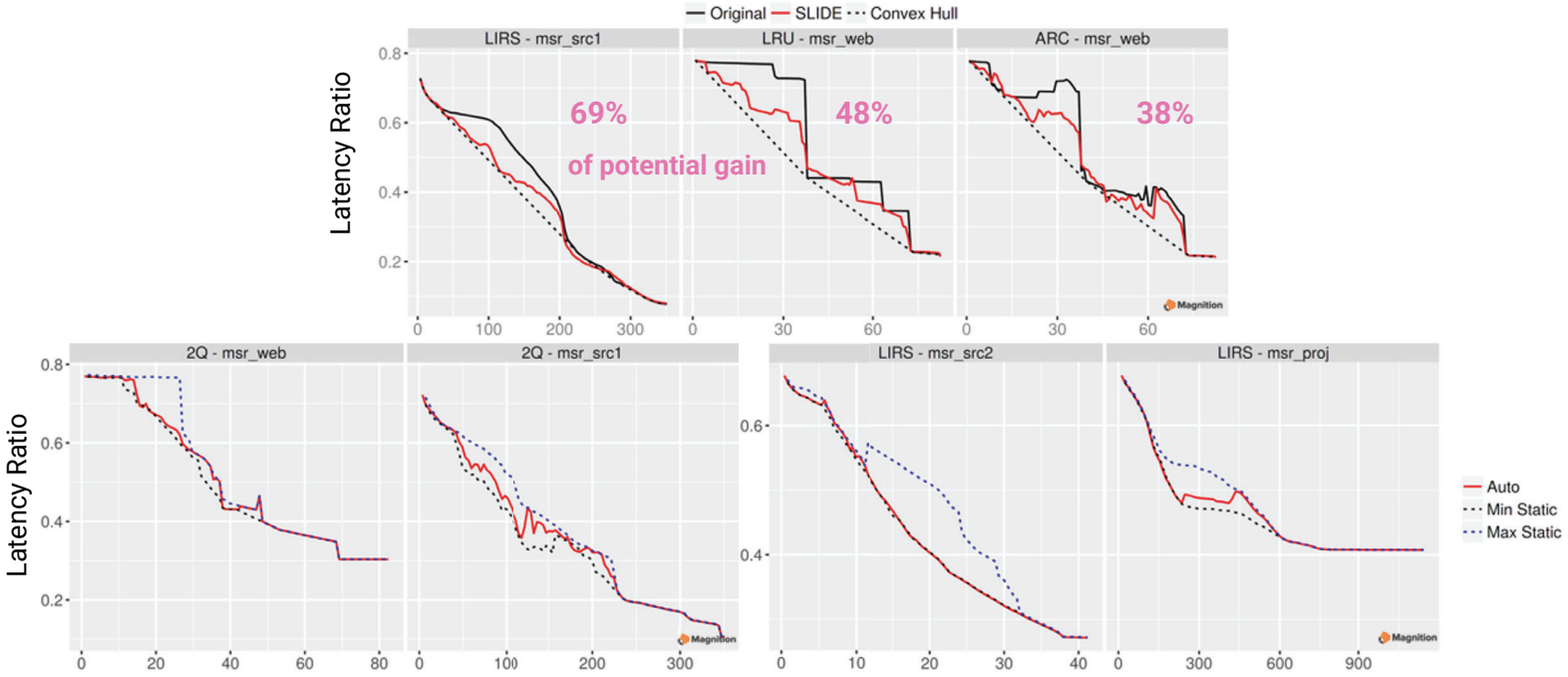
Same Workload. Real-time Performance Prediction under different policies.

Autonomous memory hierarchies would always pick the optimal operating parameters.

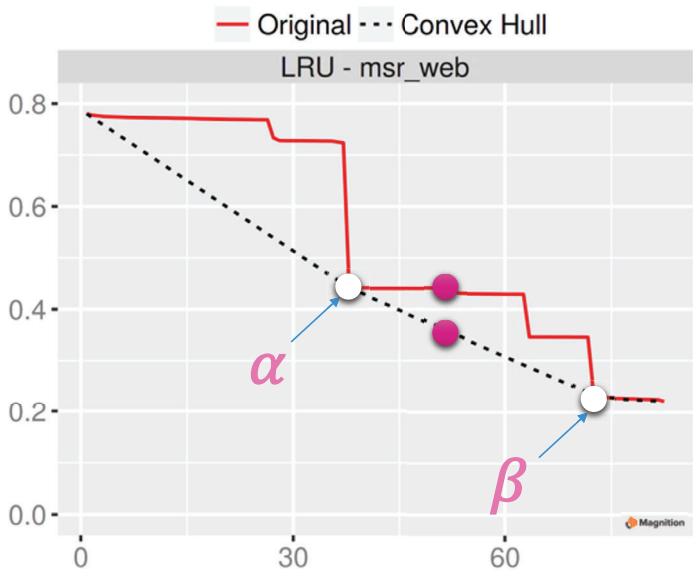
From CacheLab to Autonomous Storage



Fully Autonomous Storage is Self-Adaptive



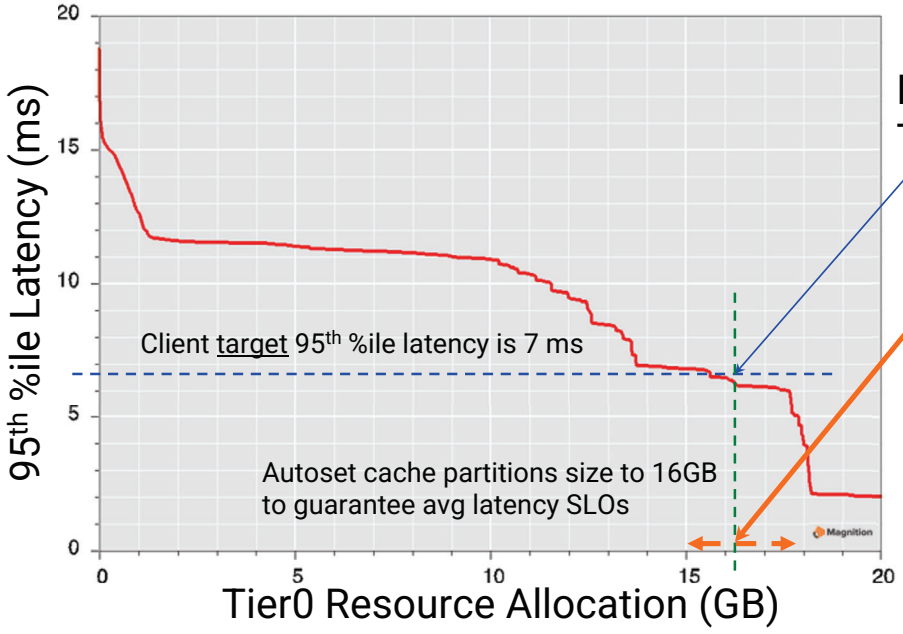
Fully Autonomous Performance Optimizations



- Thrash remediation algorithm
  - Convex hull interpolation
  - Curve steering
- Optimal curve bending cache-unfriendly workloads

hash-based emulation of cache sizes and depend on statistical self-similarity

Fully Autonomous Latency Targets

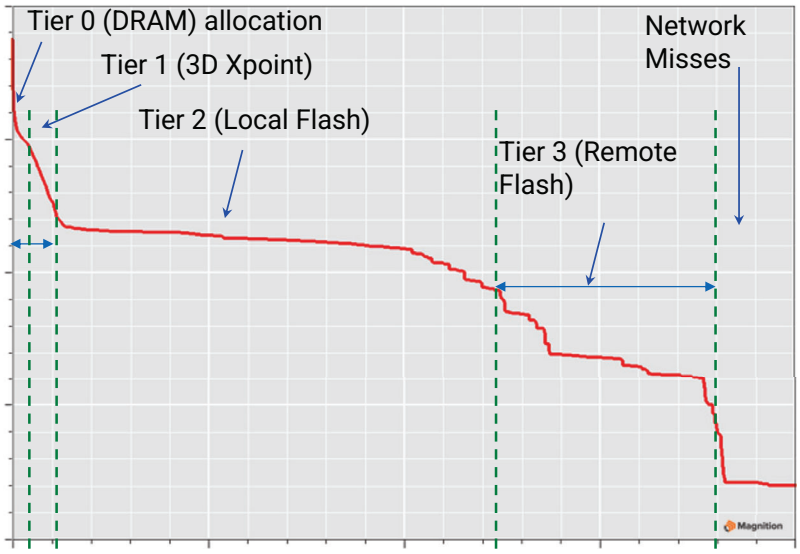


Latency Target (7 ms)

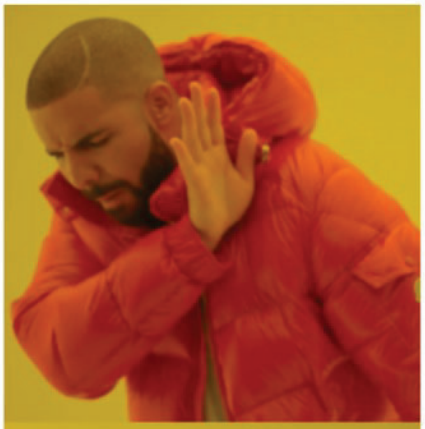
Dynamic Autonomous Tier Allocation

* Throughput targets can be similarly implemented

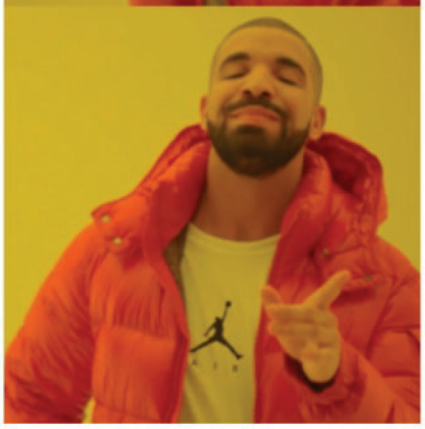
Fully Autonomous Multi-Tier Allocation



* Can model network bandwidth as a function of cache misses from each tier



Implement a custom evacuation algorithm



Use CacheLab to prove your algorithm is better

Fully Autonomous Storage is Within Reach

This is you

This should be your customer ☺️

THANK YOU

web: magnition.io
email: irfan@magnition.io

Please take a moment to rate this session.

Your feedback is important to us.

