Tutorial A-21: Enterprise Caching

Turbocharging Caches for Enterprise Applications

Som Sikdar
CTO, GridIron Systems

www.gridironsystems.com

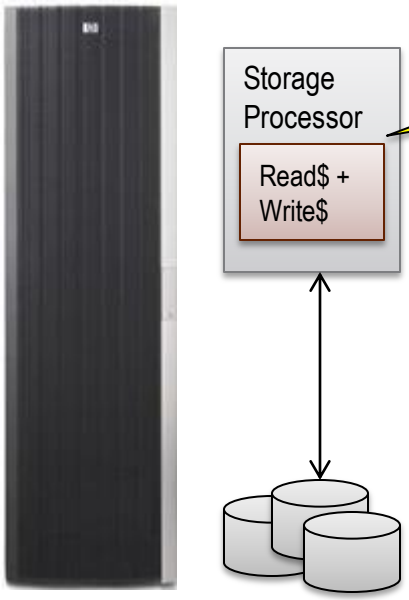
Why Think About Enterprise Caching?

- Flash Storage Technology is exciting! (More Speed. Less Power.)
- Flash is also very challenging. (Retention, Wear, asymmetric R/W access issues)
- While this Flash revolution is underway, other transitions are also taking place in the Enterprise data centers:
  1. Data access and usage demands are exploding (Volume, Velocity, Variety)
  2. IO rates are going higher (driven by multi-core CPUs and faster connectivity)
  3. Physical servers are consolidating into virtualized environments

The Caching layer is the "brain" of Storage Arrays.

All the Tier-1 Storage Arrays today were designed with a reference architecture when the Data Center environment was very different.

Why Now? We've Been Caching For 40 Years…



Legacy Caching:
Cache populate algorithm : **Cache all**
Cache eviction : **LRU/ARC/pattern**

Example:
Tier-1 storage array introduced by a leading vendor
State-of-the-art storage services
240TB usable capacity/cabinet

RAM-$ capacity = 360GB (15-way hard-partition)
SSD-$ capacity = 6TB (15-way hard-partition)

RAM Cache to back-end ratio	**= 1: 666 to 1: 10,000**
SSD Cache to back-end ratio	**= 1: 40 to 1: 600**

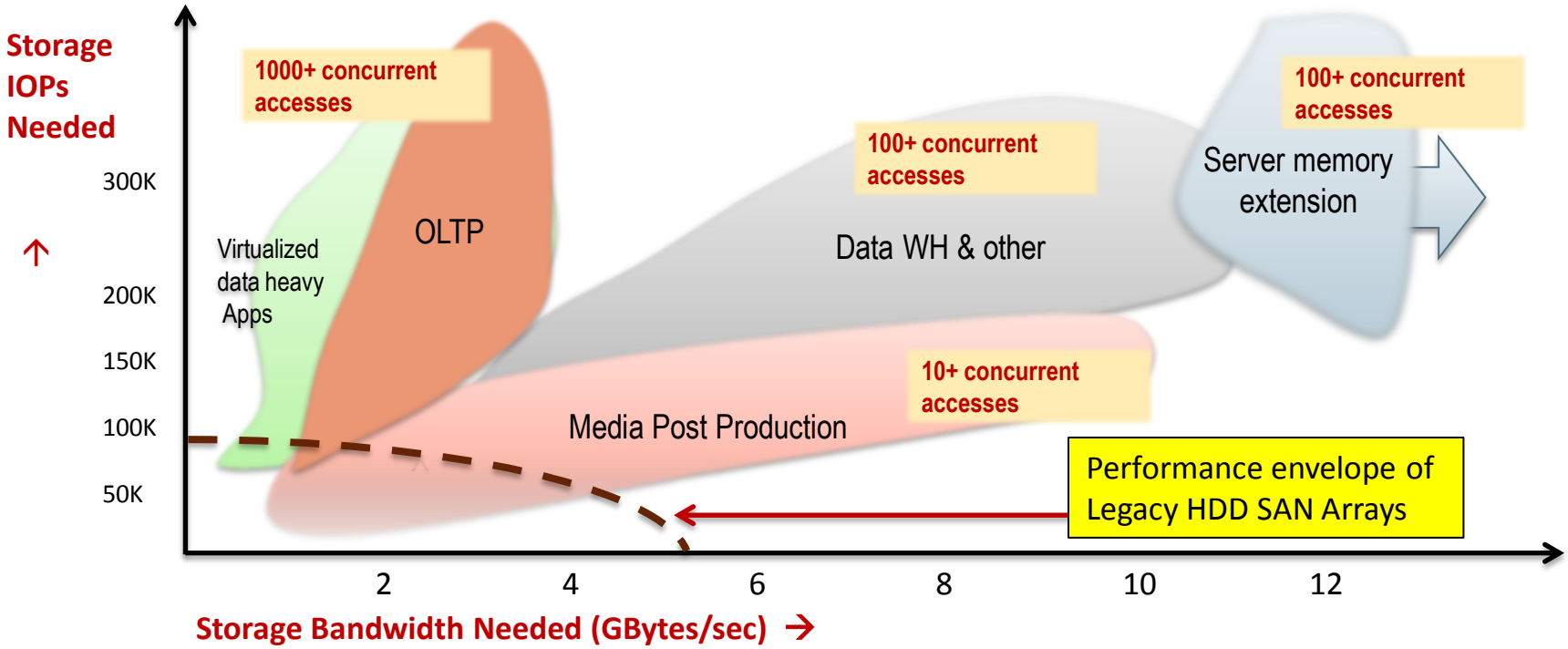
1. Cache to back-end ratio on legacy arrays are **TOO SMALL.**
2. LEGACY caches use **HIT Counters** instead of **Active Working Set RATE or SCALE parameters**
3. LEGACY caches **DO NOT PROFILE** input to determine **Concurrency, IOPs, Lat, Throughput** demand

"Active Working WHAT?"

We Count Hits. Why Is That Not Enough?

- Current caching schemes were derived from CPU caches
  - Designed to take the **least amount** of processing
  - Eviction-based ("shoot the least popular guy") rather than **profiling** or **history**
  - Counts hits on cache-lines NOT the overall statistics of the **entire storage map**
- Incomplete focus on IOPs and Latency ONLY
  - HDD systems had poor random IOPs and Latency – OLTP suffered.
  - First generation flash-vendors went gaga with IOPs and latency numbers – ignoring
    - Concurrent access (number of independent "threads" accessing the storage)
    - Bandwidth (more servers, more cores, faster SANs, shared storage → more BW needed)
- Multi-zoned Active Working Set (AWS) – the elephant horde in the room!
  - Active Working Set is **MOST PROFITABLE** cacheable data – not the most HIT
  - The AWS hits speed up apps since it identifies **access dependencies**
  - AWS parameters can be time-of-day or other **causal access behavior dependent**

Storage Access Demand: One Size Does NOT Fit All!



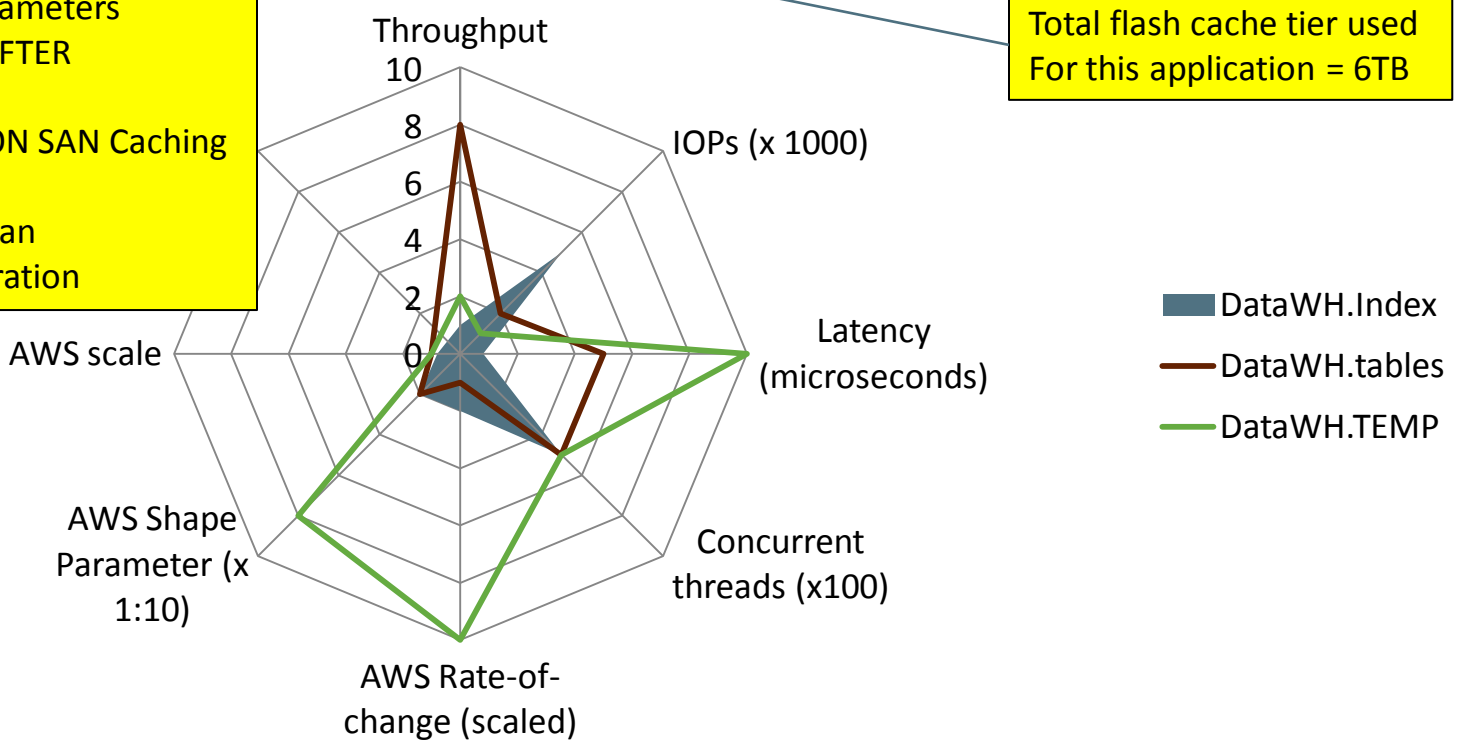
- Different applications demand VERY different performance from Storage
- **Important Access parameters:** Concurrency, IOPs, Latency and Throughput
- **Important Active Working Set parameters:** Rate of Change (ROC) and Shape

And An App May Have Different Behavioral Zones

Real life example: Workload optimized **20TB** DSS Data Warehouse (3-node Oracle RAC)

This database parameters were measured AFTER the Set-Ranking process (GRIDIRON SAN Caching algorithms) has converged to an optimum configuration

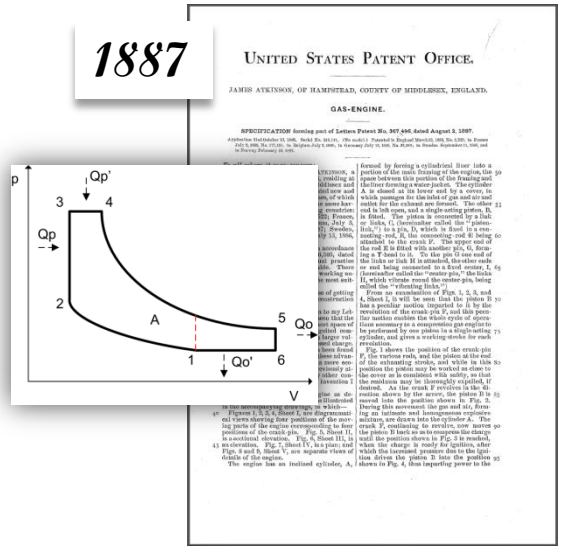
Total flash cache tier used For this application = 6TB



1. Data Tables, Index tables and TEMP areas have COMPLETELY different IO behavior
2. LEGACY caching approaches will be completely inadequate (was verified)
3. This application **needed more than 1: 8 cache capacity ratios** in each zone for max performance

The Science Is Old. The Engineering Is Brand New.

Atkinson Engine Patent



Honda Atkinson Engine, SAE demo

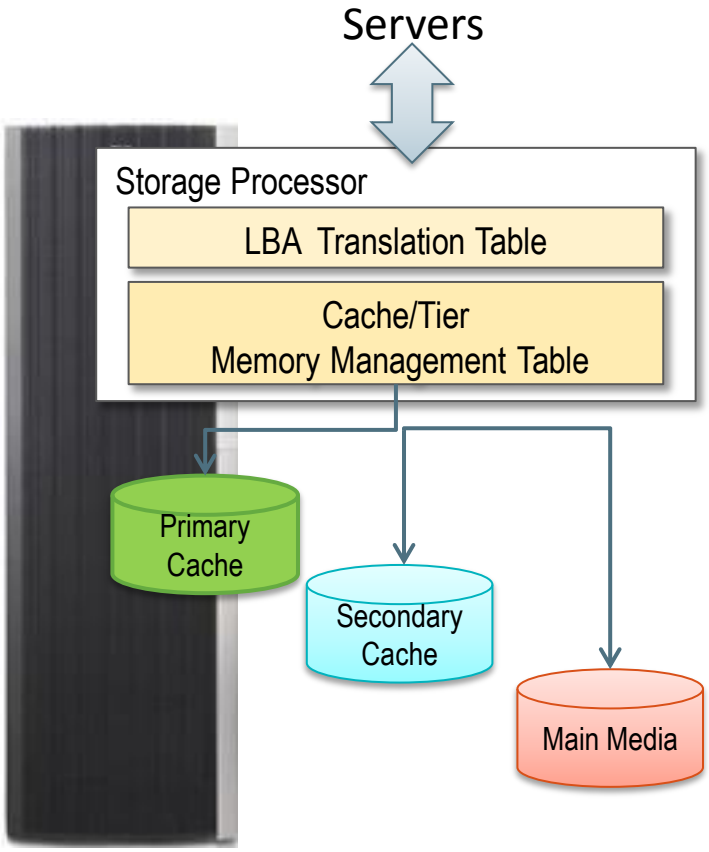
2009

1997

First-generation Prius

- Working set analysis can be done with standard statistical methods
- Multi-zone behavioral profiling can be done many existing ways
- **WARNING** – Pure software approaches are unlikely to work!
  - This is **VERY** compute intensive without silicon assist
  - The underlying caching MMU needs to operate **at microsecond level** – **unlikely** with pure SW approach

"Flash-Generation" Caching for Enterprise :
Anatomy of A Storage Data-path

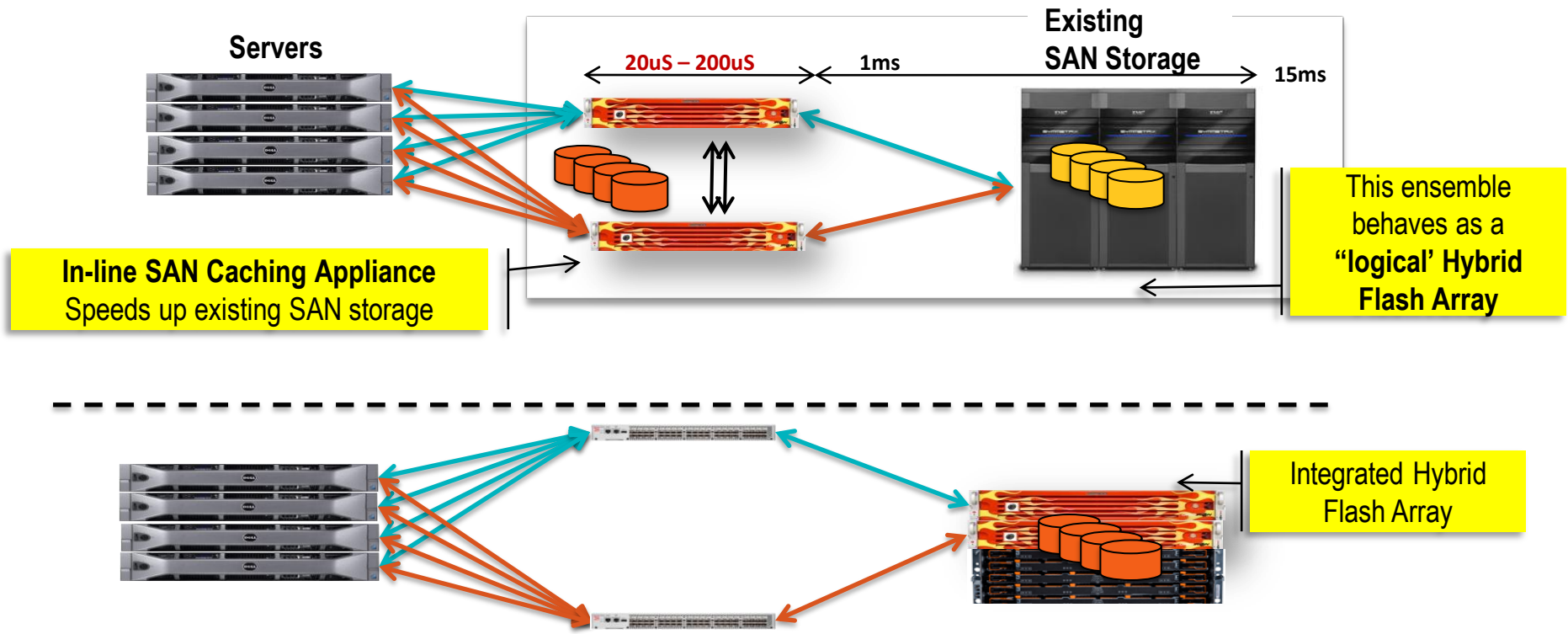


	Before 2010	Now (Hybrid)	Future All-flash
LBA processor	CPU based	CPU	CPU / Silicon
Cache MMU	CPU	Silicon	Silicon
Cache Scheme	LRU/ARC. Scripts. No history.	History, Working set and input behavior optimized	History, Working set and input behavior optimized
Primary cache medium	RAM	RAM	RAM
Intermediate $ medium	FC disks	SLC and MLC flash	MLC flash
Cache to capacity ratio	1: 100 to 1: 1000	1:4 to 1: 20	1: 4 to 1:10
Main capacity Medium	Hard disks	Hard-disks	TLC flash or cheap MLC flash

1. **The Cache layer** is **the BRAIN** of a storage array and is crucial to making flash tech work
2. **Workload** demand for storage is going through **rapid evolution**
3. **Profiling, historical analysis and high-performance table lookup is** crucial to success

“Flash-Generation” Caching :

Can be Deployed In Both Existing or New Installations



- **Transition** of Enterprise data centers to a flash-based environment **has already started**
- **Exploding demand** for data is driving the need for **better ways of matching workload** to storage
- **Turbocharging** the caching brains of enterprise data centers **can be achieved today** for both **existing** and **green-field** environments

A Real-life Case Study of Enterprise SAN Caching:

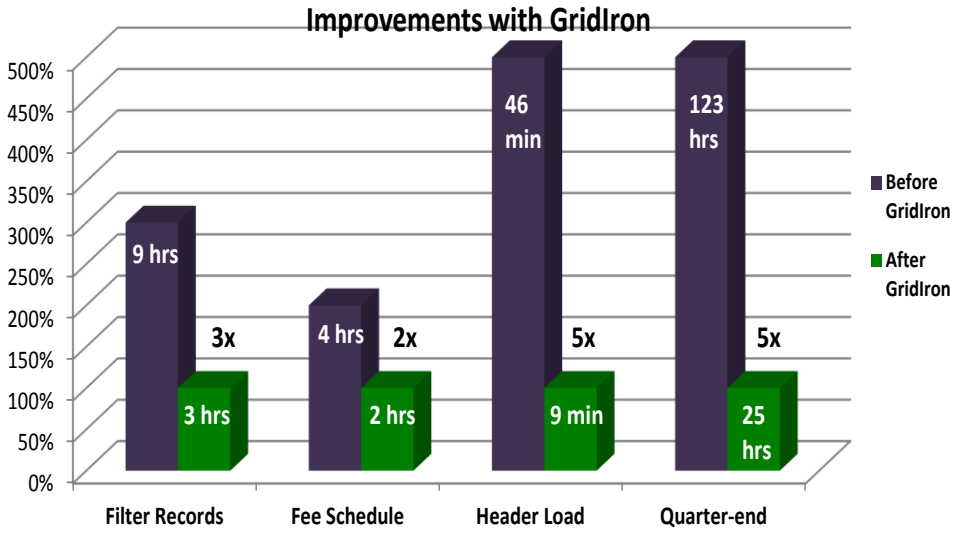
Erik de la Iglesia in Tutorial G-21: Enterprise Apps Part 1 - Handling Big Data

MULTIPLAN : Oldest and Largest PPO network in NA

- 900,000 Healthcare Providers, 57M Consumers, 110M Claims per year
- Secure and Regulated environment (HIPPA)

Benefits

- Reduced business risk
  - Brought reporting times down to match business needs
  - Consistent high performance thru peak usage cycles
  - Scaled up to handle growth
  - Met backup windows
- Maximized usage of data assets by allowing multiple concurrent access to same databases
- Accelerated delivery of new business functions through increased productivity of developers
- **Saved $400,000 in CapEx versus alternatives**



Deploying SAN caching in an existing environment meant NO TOUCH to existing applications, servers or storage!

For more information:

Website:	www.gridironsystems.com
Blog:	www.gridcto.com
Twitter:	twitter.com/gridcto