Tiered vs. All-SSD Approaches in the Cloud

Dave Wright - Founder & CEO

Cloud Storage Landscape

Local Storage

- Data processing
- Temp files
- Swap
- Cache

Bulk Object Storage

- Media files
- Content distribution
- Backup
- Archival

High Performance Block Storage

- Primary storage
- VM images
- Databases
- Application files

The Cloud Needs Better Storage

Performance

- Unable to manage performance independent of capacity
- Can not guarantee storage performance

Efficiency

- Low and inefficient utilization rates
- Lack of high-performance in-line data reduction

Management

- Complex manual management that lacks automation

Scale

- Limited scalability of both capacity and performance
- Manage multiple islands of storage

Flash Benefits for Block Storage

- Performance
  - Restore balance between IOPS and Capacity
  - Better performance with varied workloads
- Efficiency
  - Remove performance as a limiter on utilization
  - Reduce storage space, power, and cooling
- Management
  - Better reliability
  - Easier to isolate workloads on shared media

What's the downside?

- **Cost**
  - Partially a perception issue
  - Can't be ignored – cloud is all about value

- **Existing Storage Architectures**
  - Ability to get full performance from flash
  - Write amplification & endurance issues

How Flash is being used today

- Direct-attached
- Host-based caching
- Array-based caching
- Array-based tiering
- All-solid-state arrays

Direct-Attached Flash

- Examples:
  - PCIe Cards
  - 2.5" SSD

- Pros:
  - Best performance

- Cons:
  - Cost
  - Limited capacity / flexibility
  - No sharing
  - Limited availability / redundancy

Host Based Caching

- Examples: maxCache ioTURBINE
  - f FlashCache, Adaptec, Marvell

- Pros:
  - Larger flash footprint
  - Lower latency than array based caching

- Cons:
  - Read-only, or affects data integrity/availability
  - Huge delta in performance between cached/un-cached reads

Array Based Caching

- **Examples:**
  - FlashCache, **EMC**^{2} FAST Cache
- **Pros:**
  - Invisible / seamless improvement
  - Provides good boost for hottest data
- **Cons:**
  - Requires expensive SLC flash due to churn
  - Can cause irregular performance in multi-tenant environment

- Examples:
  - 3PAR EMC² compellent
- Pros:
  - Potentially large capacity available
  - Ability to automate or manually place data in tiers
- Cons:
  - Requires expensive SLC flash due to churn
  - Performance overhead moving data around
  - Irregular performance for multi-tenant
  - Complex to scale / size right

All Solid State Arrays

- Examples: **PURE**STORAGE **SOLID**FIRE _tms_

  NIMBUS DATA **Violin** MEMORY

- Pros:
  - Consistent, high performance for all data
  - Best $/IOP
  - less space/power/cooling

- Cons:
  - Cost/GB?
  - Cloud scale?

SOLIDFIRE

All-SSD storage solution designed specifically for cloud service providers

- Clustered Scale-Out Architecture
- Industry-Standard: Hardware and 10 GigE iSCSI
- Massive scale: 100 nodes, 2PB, 5M IOPS
- Cost Effective: Usable $/GB similar to traditional SANs

The SolidFire Approach

- Completely separate data storage from higher level representations
- Allocate physical space in byte-size chunks
- Optimize storage of small blocks
- Utilize entire drive pool for every volume
- Enables:
  - Performance virtualization
  - De-dupe and compression without tradeoffs
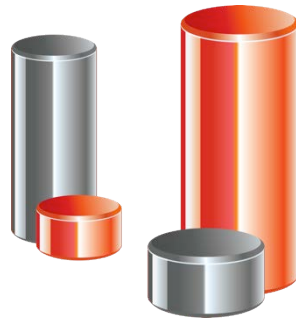 - HA without performance impact

LUN 1

LUN 2

LUN 3

Volumes

Volume Metadata

Block Data
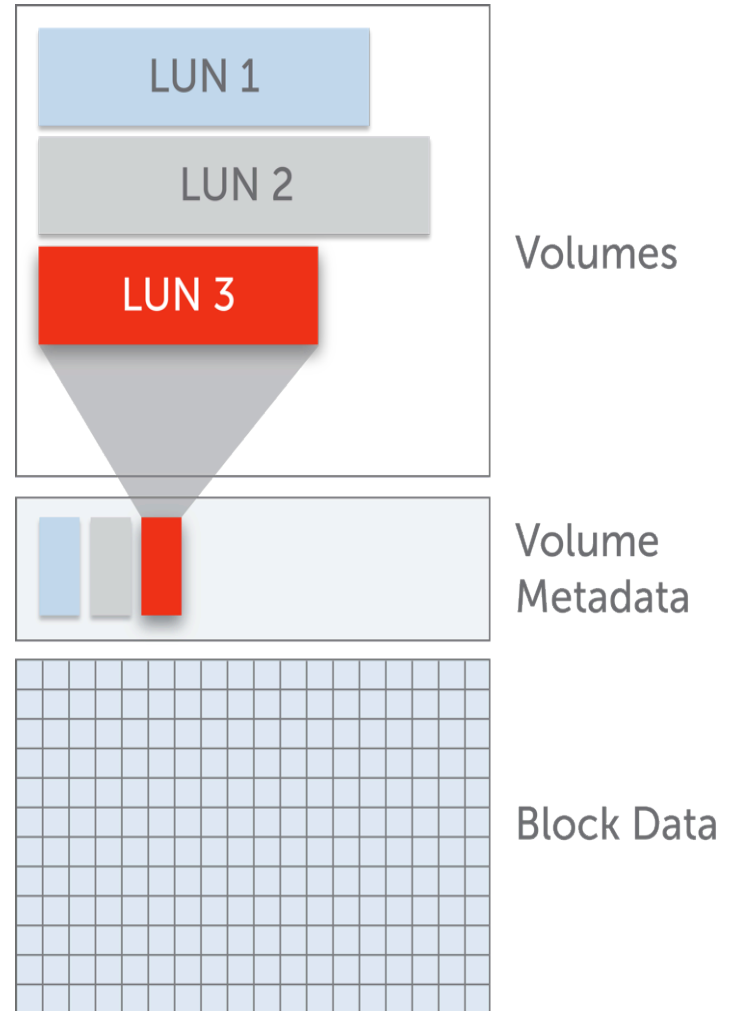
Key Benefits of all-SSD with SolidFire
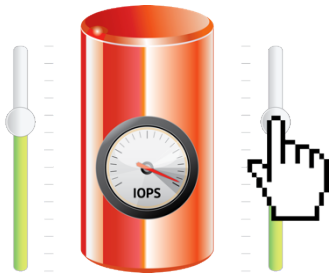
Performance virtualization

Guaranteed fine-grain performance per volume

Profitable efficiency

In-line data reduction and 85% utilization requires less purchased capacity

Complete automation

REST-based API for complete control

Cloud Scaleability

Simultaneous capacity and performance scaling

Summary

- Flash overcomes key primary storage issues in the cloud
- Many possible approaches to using it
- New techniques are reducing $/GB and supporting cloud scale
- The future cloud will be all SSD for primary storage