SCALING AN ALL FLASH ARRAY

THE DEVIL IS IN THE DRAM

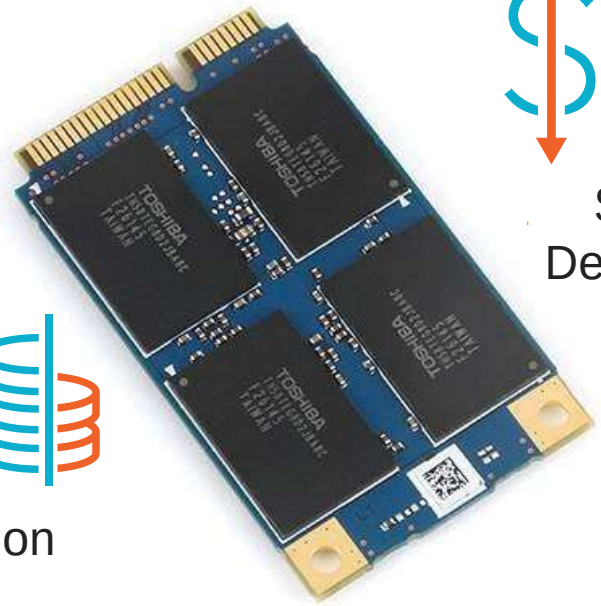
NEIL VACHHARAJANI
LEAD SOFTWARE ARCHITECT

PURE STORAGE

The Economics of an All Flash Storage Array

The Media

Low $/GB



$/GB
Steep
Descent

Data
Reduction

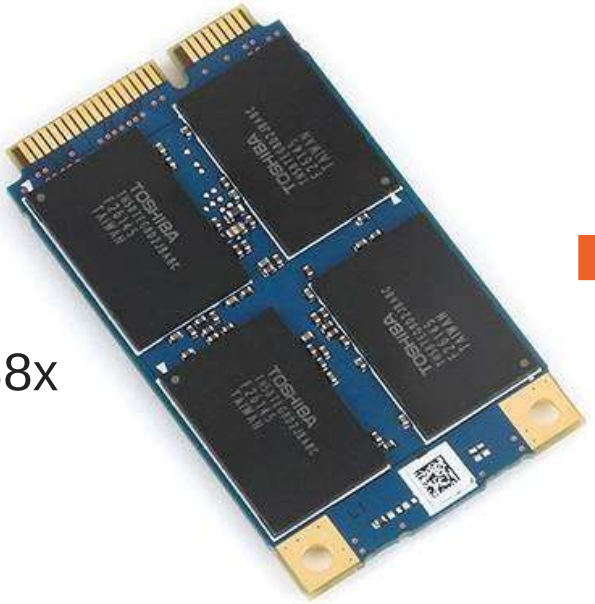
~~High $/GB~~

Low $/GB usable

The Economics of an All Flash Storage Array

The Full System

88x



4x Intel Xeon CPUs
512 GB – 1TB DRAM
4x FC HBAs
6x SAS HBAs
2x drive enclosures
1x server chassis
~~battery backup~~
~~Infiniband HCA~~
~~Infiniband switch~~

SIGNIFICANT COST OF A SYSTEM LIES OUTSIDE OF FLASH

Scale-Up vs Scale-Out

A False Choice

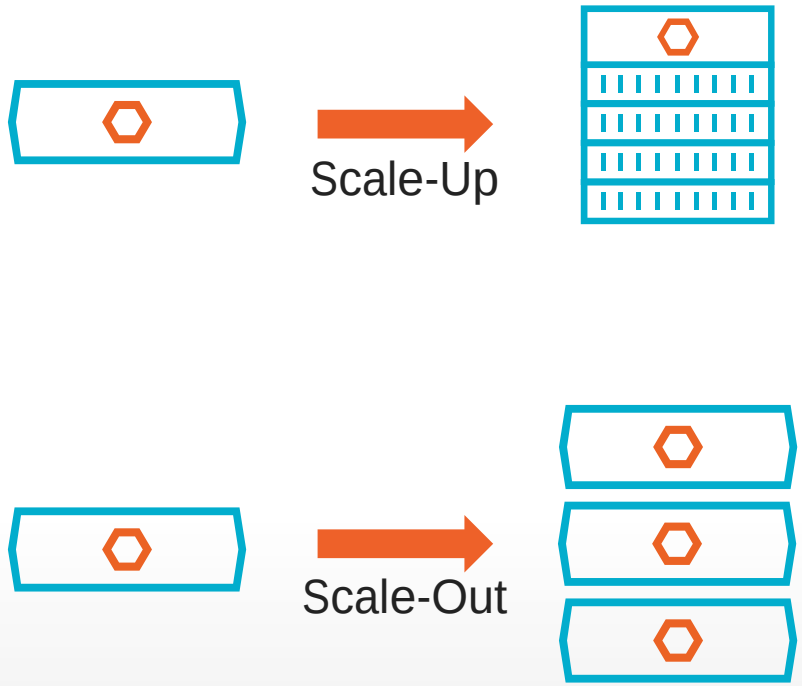
Scale-Up

- Amortizes the overhead over more flash
- Capacity and performance scale separately
- Limitations on how far you can scale

Scale-Out

- No limitations on how far you can scale
- Capacity *and* performance scale together
- Fixed overhead per TB of storage
  - CPU and DRAM
  - HBAs, NICs, switches, switch ports
  - Power supplies, rack space, cooling

YOU DON'T HAVE TO PICK!

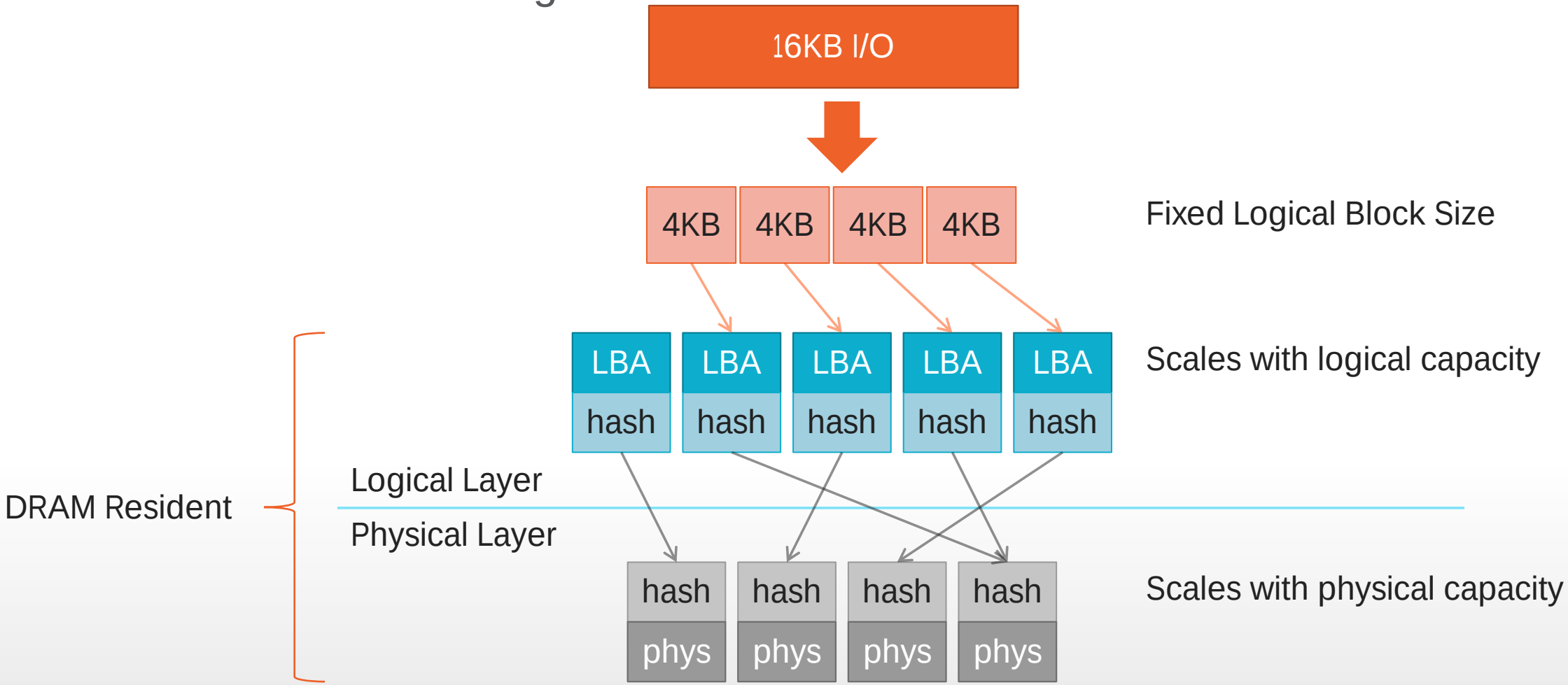


THE STRAWMAN ARCHITECTURE

CONTENT-ADDRESSED STORAGE

How It Scales

Content Addressed Storage



 |

Doing the Math

Content Addressed Storage

Quantity	Type
50TiB	Physical capacity
4KiB	Logical block size
12.5 Gi	Blocks
20B	Cryptographic hash size
250 GiB	Hash metadata the *physical layer*

Quantity	Type
250 TiB	Logical capacity
4KiB	Logical block size
62.5 Gi	Blocks
20B	Cryptographic hash size
1250 GiB	Hash metadata for *logical layer*

DRAM HUNGRY ARCHITECTURE

1500 GiB *JUST* FOR HASHES!

SCALES LINEARLY

2X CAPACITY → 2X DRAM

AT ODDS WITH DATA REDUCTION

MORE LOGICAL CAPACITY → MORE DRAM

TRADE DRAM FOR DEDUPE

2X BLOCK SIZE → ½ DRAM

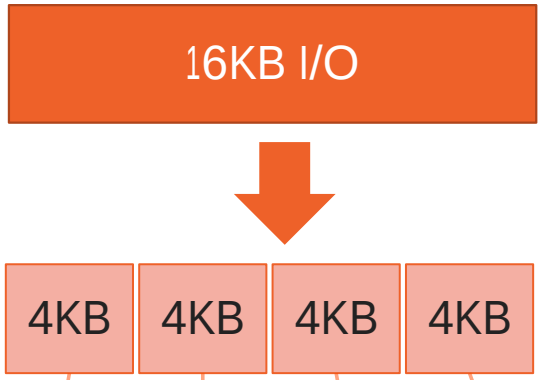
2X BLOCK SIZE → LESS DEDUPE

A MORE SCALABLE ARCHITECTURE

DECOUPLING DEDUPE FROM THE DATA PATH

Making Metadata Not DRAM Resident

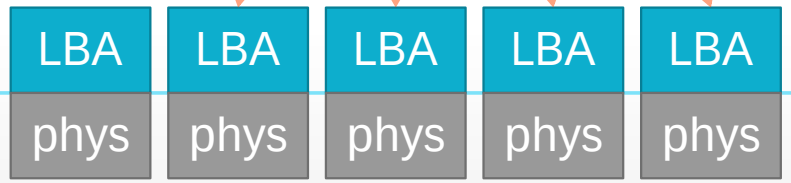
A More Scalable Architecture



Fixed Logical Block Size

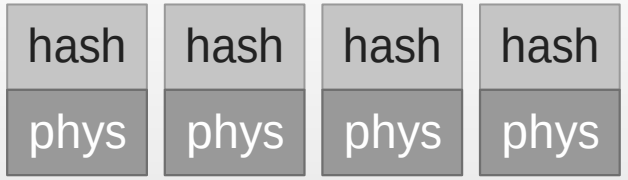
Logical Layer

Physical Layer



Scales with logical capacity

Scales with physical capacity



Efficiently Encoding the Block Map

A More Scalable Architecture

Efficiently Encoding the Dedupe Map

A More Scalable Architecture

Quantity	Type
4KiB	Dedupe block size
20B	**Cryptographic** hash size
0.5%	Hash size/block size

Quantity	Type
4KiB	Dedupe block size
8B	**Non-Cryptographic** hash size
0.2%	Hash size/block size

2.5X SAVINGS FROM SIMPLER HASH

SMALLER HASH → MORE HASHES IN DRAM

VERIFY DATA IS REALLY DUPE

8B HASH → HIGHER CHANCE COLLISION
JUST CHECK THE DATA

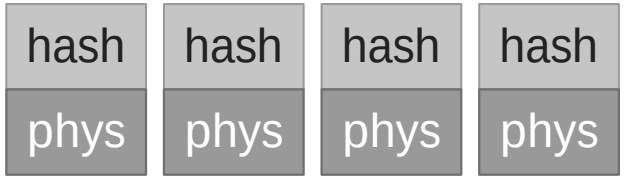
DATA SET VS DATA STREAM

SET: MANY COPIES OF IDENTICAL DATA
STREAM: MOSTLY UNIQUE DATA

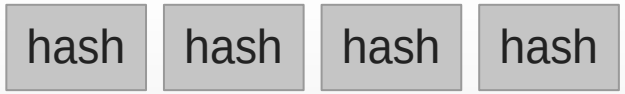
Squeezing Dedupe Map into the Dedupe Set

A More Scalable Architecture

Dedupe Map



Dedupe Set



MOST BLOCKS ARE UNIQUE

SPEND LESS ON BLOCKS THAT WON'T DEDUPE

LOOKUP IN MAP ONLY IF IN SET

SET CAN BE REPRESENTED WITH LESS DRAM

DEDUPE SET OPTIMIZATIONS

NO FALSE NEGATIVES
ALLOW FALSE POSITIVES
MUCH MORE COMPACT REPRESENTATION!

CONCLUSIONS

THE DEVIL IS TRULY IN THE DETAILS

Conclusions

The Devil is Truly in the Details

WHAT WE BUILT

- FlashArray //m
- //m70 SCALES TO 136 TB
- MORE SCALE, SAME HW, COMING SOON…

CONCEPTS ARE SIMPLE…
IMPLEMENTATIONS MATTER TOO!

MANAGING FLASH IS TRICKY

ECONOMICS IS ONLY PART OF THE STORY

- PERFORMANCE MATTERS
- SIMPLICITY MATTERS
- RELIABILITY MATTERS
- OPERATIONS MATTER

THANK YOU.

QUESTIONS?