Eusocial Storage

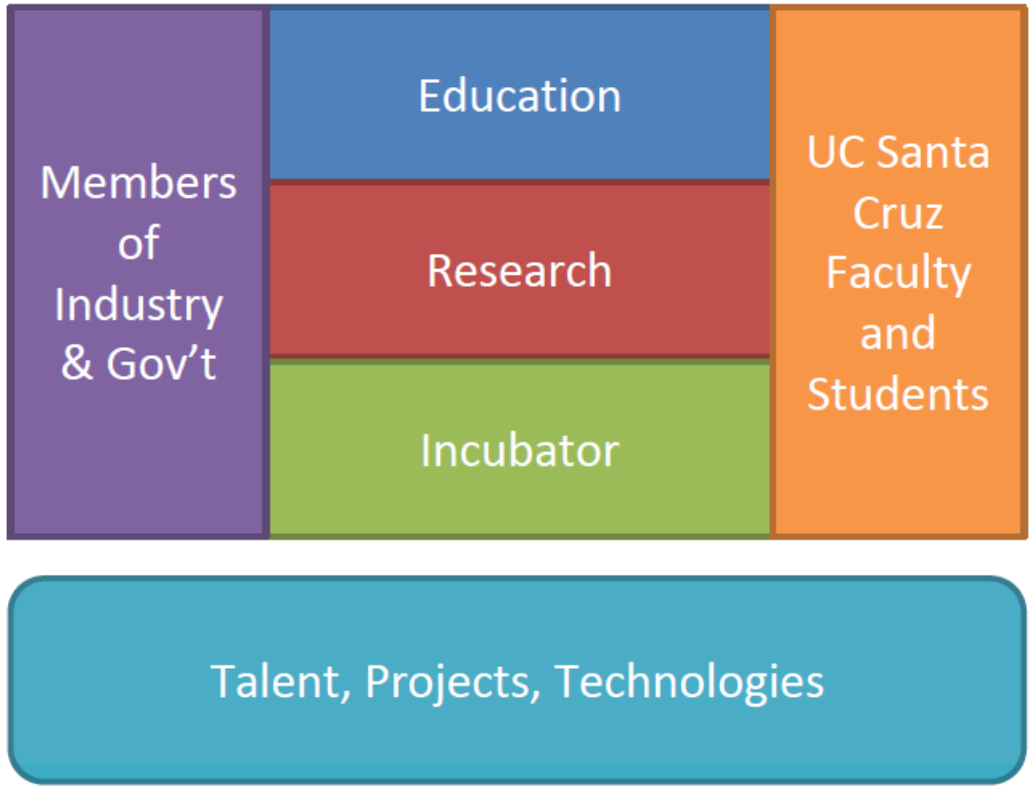
CROSS | CENTER FOR RESEARCH IN OPEN SOURCE SOFTWARE

Baskin Engineering
UC SANTA CRUZ

UCSC CROSS

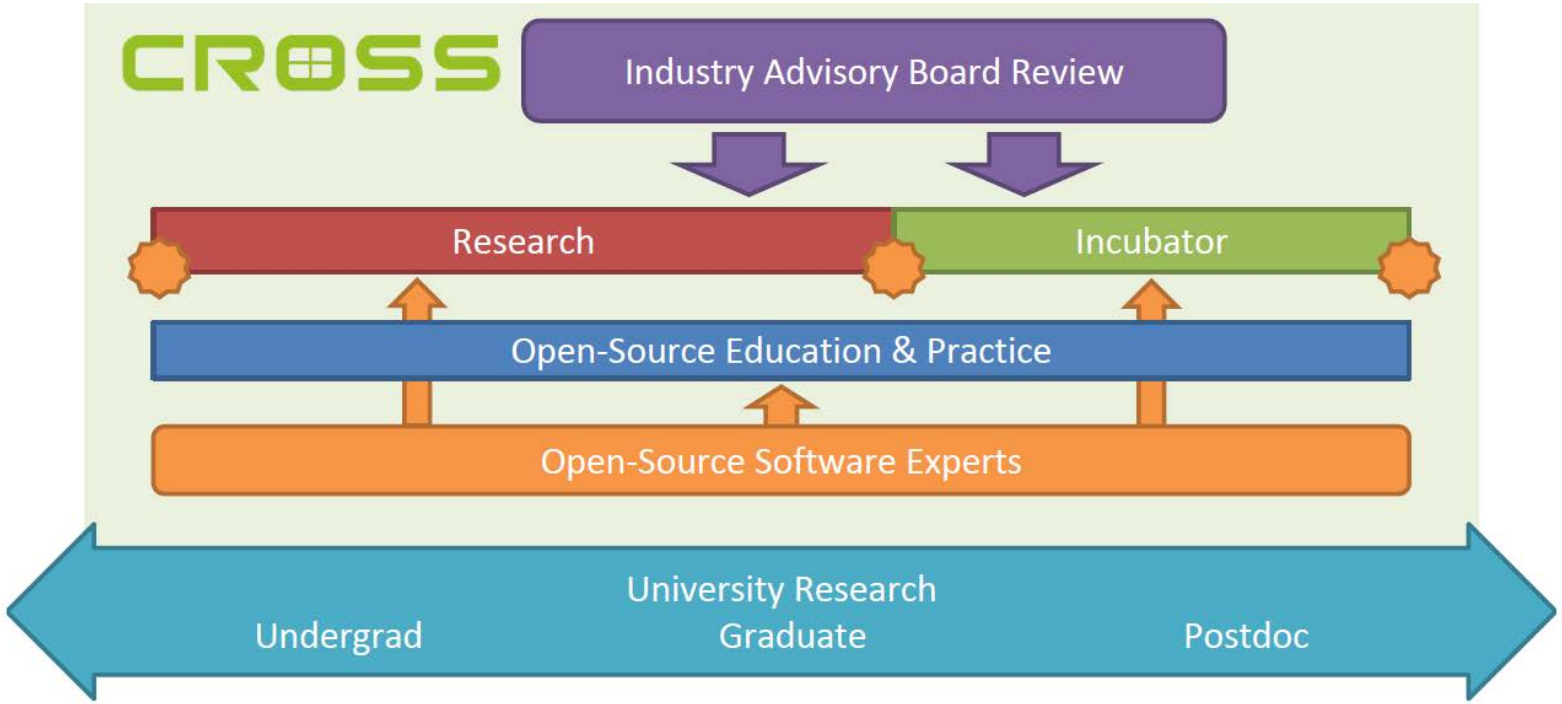
WHAT IS IT

- **Bridges gap between student research & open source projects**
  - Funded by Sage Weil endowment & corporate memberships
- Educate the next generation of OSS leadership
- Leverage OSS culture in university research
- Incubate work beyond graduation to reach critical mass



2018 Storage Developer Conference. © Philip Kufeldt. All Rights Reserved.

- CROSS Operational Model
  - Modelled after NSF's I/UCRCs
  - Adds open-source software focus
  - Sustained through membership fees



2018 Storage Developer Conference. © Philip Kufeldt. All Rights Reserved.

- Eusocial Storage is a new Incubator for Research & Standardization
- First presentation at FAST 18 WiP and Poster
- Published an article in USENIX ;login: 2018 Summer Issue
  - https://www.usenix.org/system/files/login/articles/login_summer18_05_kufeldt.pdf

- More info: UCSC CROSS
  - https://cross.ucsc.edu/

- 2018 CROSS Symposium & Oktoberfest
  - https://cross.ucsc.edu/2018-symposium

2018 Storage Developer Conference. © Philip Kufeldt. All Rights Reserved.

Eusocial Storage

WHY AND WHAT

eu·so·cial

/yo͞oˈsōSHəl/

adjective Zoology

: (of an animal species, especially an insect) showing an advanced level of social organization, in which a single female or caste produces the offspring and nonreproductive individuals cooperate in caring for the young. Eusocial species often exhibit extreme task specialization, which makes colonies potentially very efficient in gathering resources.

eu·so·cial stor·age

/yo͞oˈsōSHəl ˈstôrij/

noun Information Technology

: storage media showing an advanced level of autonomy and organization, in which a single cloud of storage provides definable classes of service by organizing storage media into highly efficient castes of storage and managing data flow through the castes.

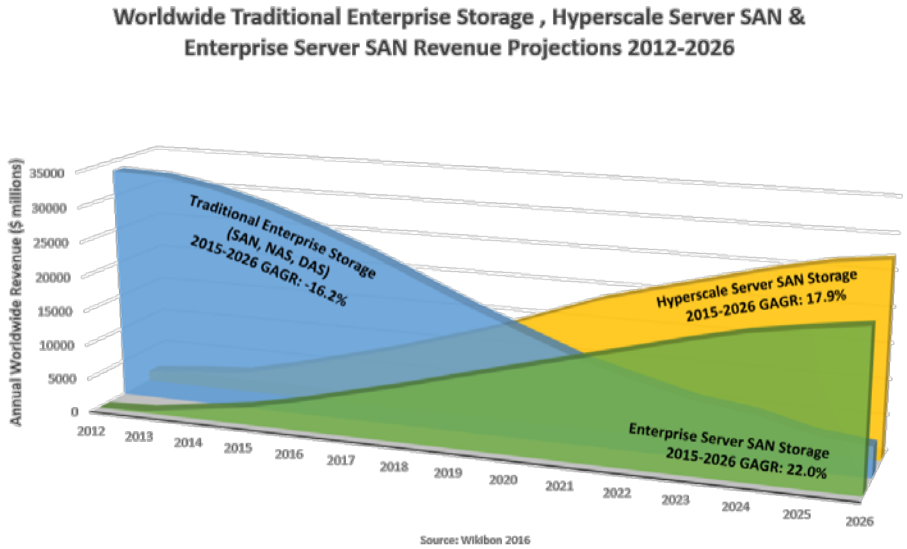
Why Eusocial Storage

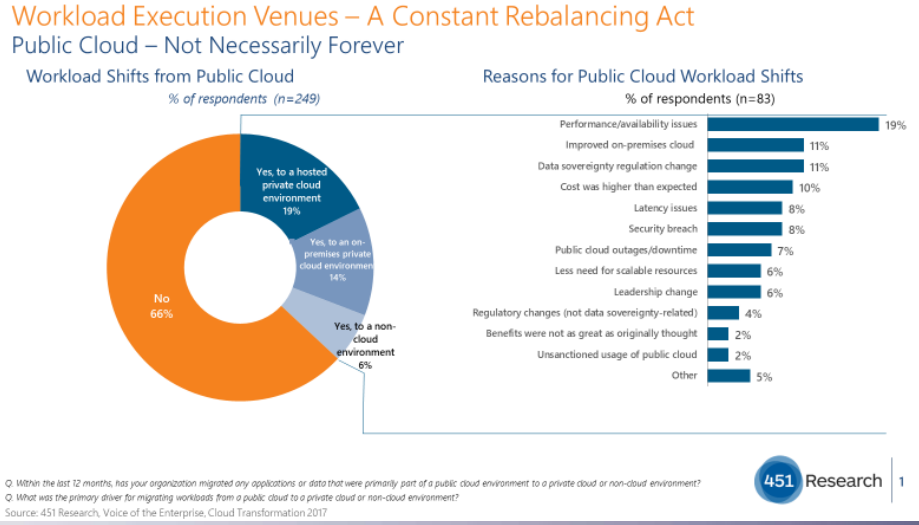
- Trends point toward needing eusocial storage
  - Public/Private Cloud
  - Server Offload
  - Disaggregation
- Need to adapt to changing environment
  - Methods of the past may not fit in

2018 Storage Developer Conference. © Philip Kufeldt. All Rights Reserved.

Public Cloud

- Private storage cloud predicted to grow faster than Public storage cloud.
- Big Datacenter's taking notice
  - Want to blunt private cloud; direct customers to their offering
  - Azure selling its stack
  - Google partnering with Nutanix
- Worst outcome for entire industry is to have 5 customers for all compute and storage
- Why doesn't this seem to be happening now?
  - Public Server SAN features need to be in privately available Server SAN solutions
- Highly functional but **simple to deploy** and manage storage cloud environment is required



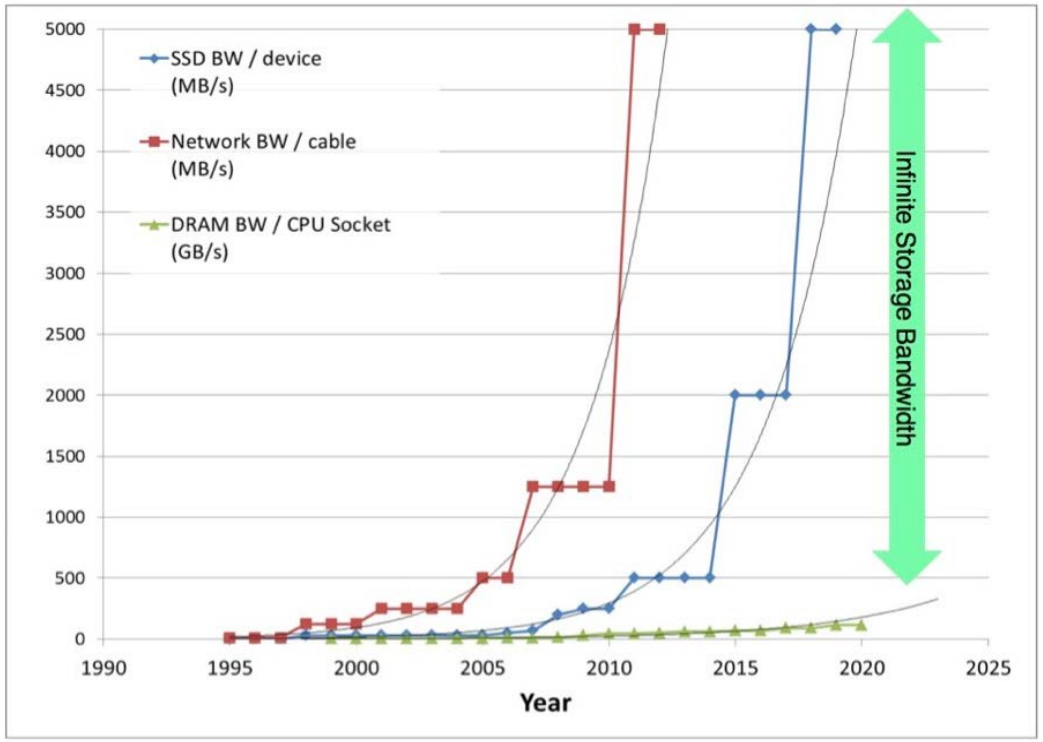


2018 Storage Developer Conference. © Philip Kufeldt. All Rights Reserved.

Server Offload

- Thanks to discussions started by Fritz Kruger and Allen Samuels
- Storage and Network throughput growth outstripping CPU
  - By 2020 only a couple SSDs per socket will be needed to outstrip ability to shuttle data in and out
  - SCM is going to make this worse
- Disaggregation elongates data bus
  - Data movement can be more costly
- Storage heads will be the bottleneck
  - DMAs steal memory bandwidth
- Restrict DMAs to only move the necessary data to the app
  - relegating data management to the device

- **Network, Storage and DRAM Trends**
  - DRAM throughput is a proxy to CPU capability
  - Storage Bandwidth is not literally infinite
  - But the ratio of Network and Storage to CPU throughput is widening very quickly

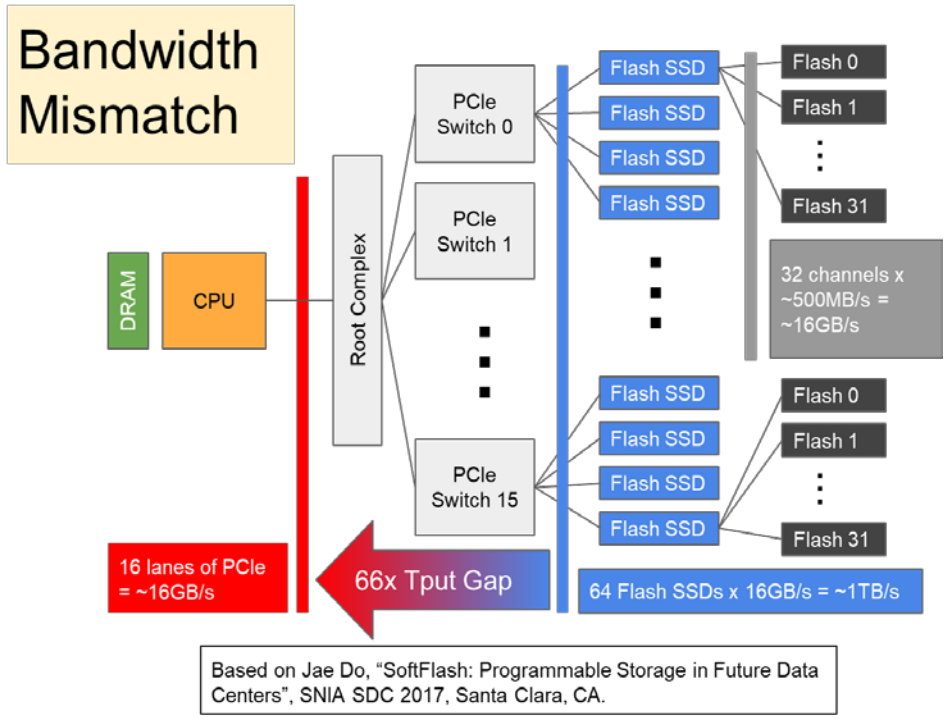


CPU Bandwidth – The Worrisome 2020 Trend, Fritz Kruger, Mar 2016

2018 Storage Developer Conference. © Philip Kufeldt. All Rights Reserved.

Problem

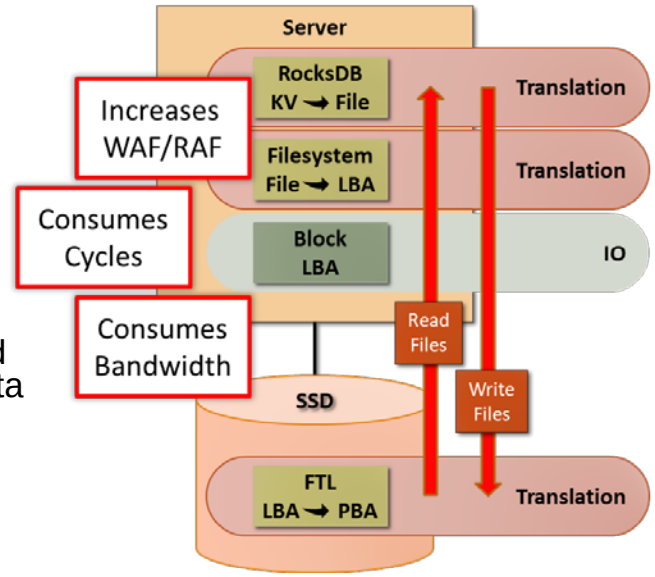
- Server Bandwidth Mismatch
  - Flash and Network BW catching up and will surpass DDR BW
- Storage IO
  - Application Requirements
    - North and South (N-S)
      - Real work: Primary data path
  - Data Management Requirements
    - North and South
      - Translation, Compaction, Deduplication, Scrubbing
    - East and West (E-W)
      - Redundancy, Recovery, Rebalancing, Tiering and Caching



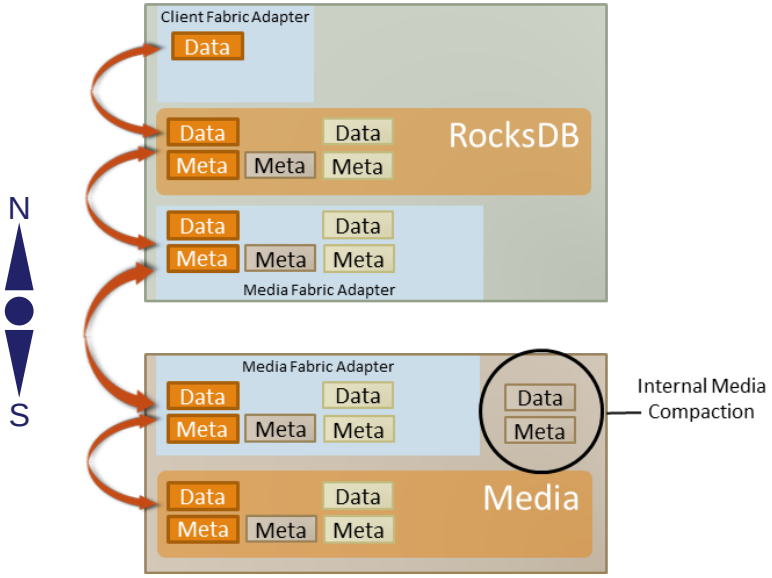
2018 Storage Developer Conference. © Philip Kufeldt. All Rights Reserved.

Case study: Server Data Management

- Redundant translations and behaviors
  - Translate higher abstraction in media format
  - Compaction occurring in multiple locations
- DMA use for data management
  - Scrubbing, Compaction
- Example: KV Stores
  - LSM implementations
    - Multiple translation layers
    - Compaction, scrubbing all read and write the data, bringing data in and out of the server
      - Consumes cycles and bandwidth
    - Increases WAF and RAF



Translations for data management



DMA use/copies for data management

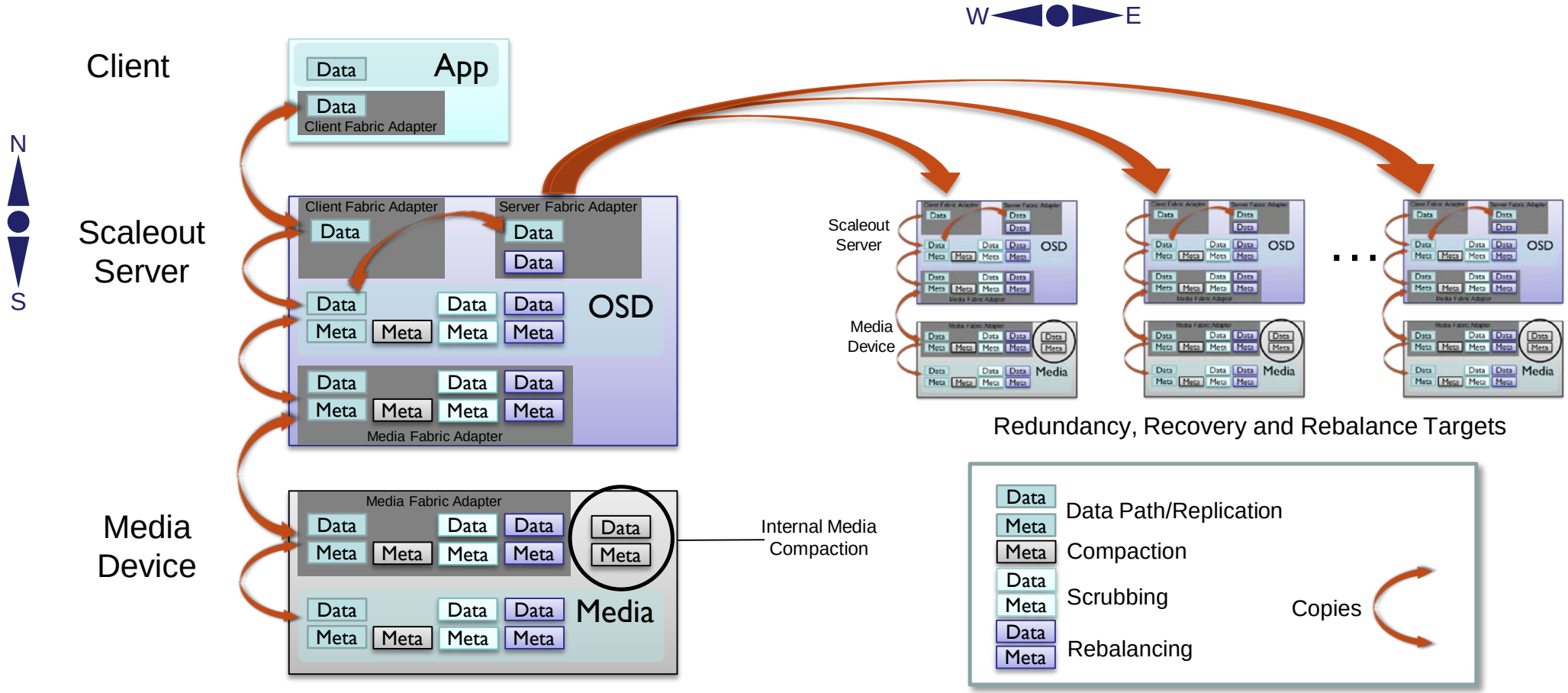
2018 Storage Developer Conference. © Philip Kufeldt. All Rights Reserved.

Case Studies: Scale out

- Scaleout servers use more DMA
- North and South
  - Move data into scale out server for data scrubbing
  - Use KV Stores, like RocksDB, for meta data
    - Compaction and additional scrubbing
- Add East and West
  - Move data from clients to media and to other scaleout servers for redundancy
  - Move data from scaleout server to scaleout server for recovery
  - Move data from scaleout server to scaleout server for rebalancing
- Add external data movement
  - Data migrations from scaleout servers to other servers for the purposes of QoS
    - Tiering and caching
- Strong candidate for offloading

2018 Storage Developer Conference. © Philip Kufeldt. All Rights Reserved.

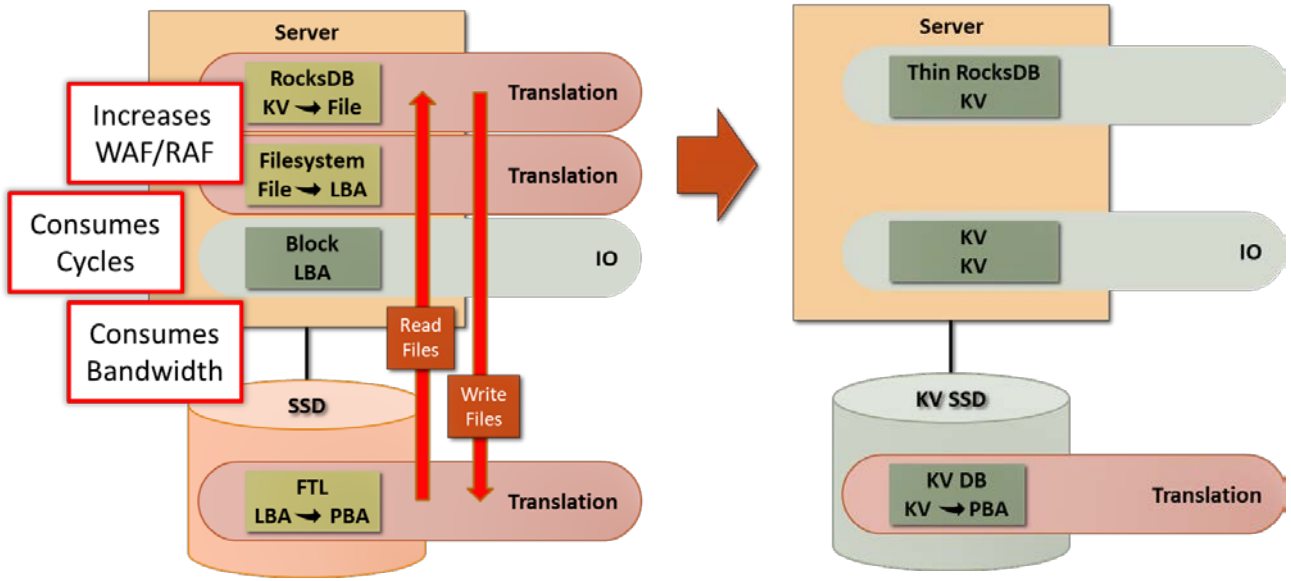
Scale Out Adds to the problem



2018 Storage Developer Conference. © Philip Kufeldt. All Rights Reserved.

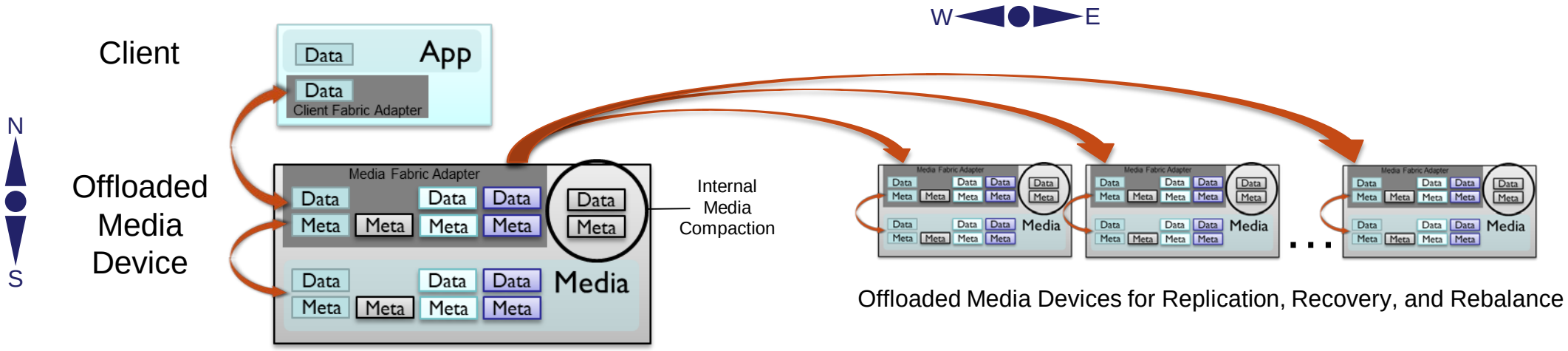
If Offloaded

- North South Offloading
  - Scrubbing, Compaction, Translation
  - Example: KV Stores
    - Moving the LSM off the server
      - Reduces the stack
      - alleviates these issues

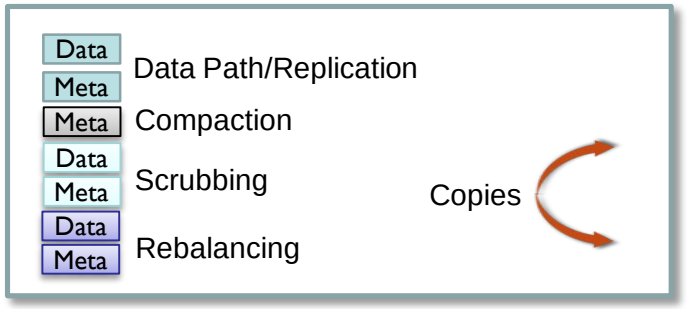


2018 Storage Developer Conference. © Philip Kufeldt. All Rights Reserved.

If Offloaded



- North South, East West Offloading
  - Requires smarter devices
  - Removes servers
  - Dramatically reduces copies

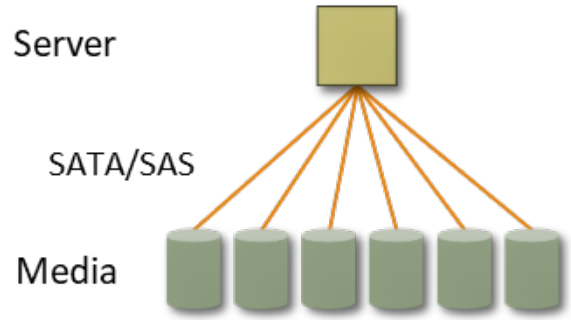


2018 Storage Developer Conference. © Philip Kufeldt. All Rights Reserved.

Importance of Disaggregation

Aggregation

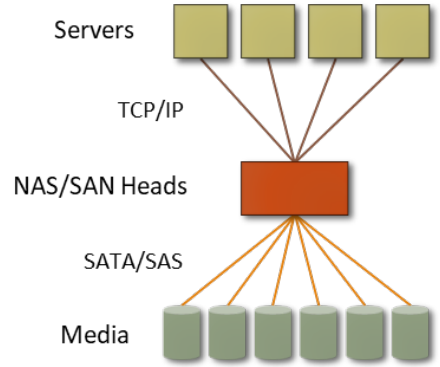
- Traditionally media needed for an application was physically connected to the server
  - Physical limitations
    - Number or drives dependent on chassis, connection paths, CPU, DDR
  - Bandwidth limitations
    - Apps limited number of devices and pathways
  - Capacity limitations
    - Physical limitations
  - Software configuration limitations
    - Apps locked on server as they use local storage
    - How many apps fit along side media and what are their capacity/bandwidth requirements
    - Planning difficult and static, cannot adapt
  - Availability configurations
    - Local RAID for drive failures
    - Standby systems with replicated or dual ported devices for server failure
- This does not scale
  - Each server, its workload and availability scheme must be planned in advance
  - Cannot adapt to workload changes
- Disaggregating storage from the app server gives flexibility

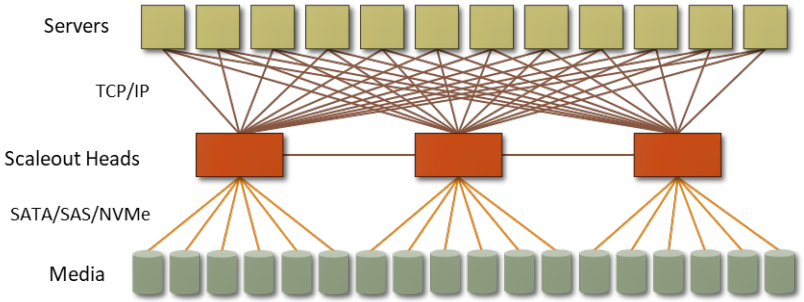


2018 Storage Developer Conference. © Philip Kufeldt. All Rights Reserved.

Disaggregation

- NAS/SAN was disaggregation at array/LUN level
  - Separates apps from storage allowing more sharing flexibility
  - Storage failover without necessarily restarting apps
  - NAS/SAN heads are scale up – aggregated
    - Still has all of the other limitations
  - No cooperation between NAS/SAN Heads other than replication
- Scale out systems are disaggregated
  - First disaggregation to scale bandwidth and capacity
  - Availability is across heads
  - Scale out heads are scale up – aggregated
    - Still has physical limitations, memory bottleneck and planning issues
  - Cooperation between scale out heads to provide redundancy, performance and capacity balance





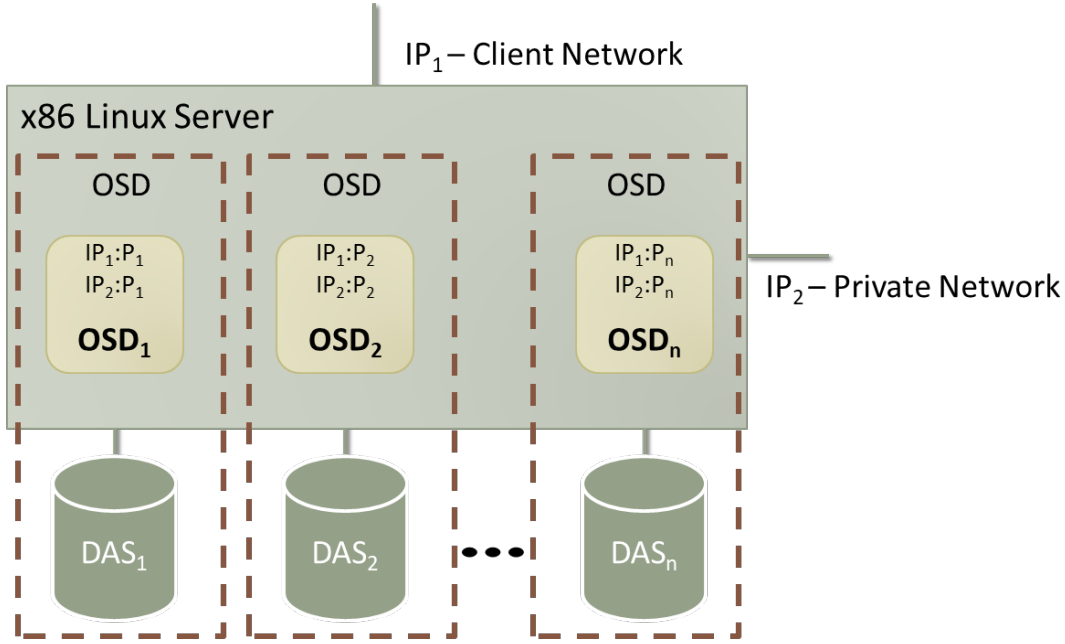
2018 Storage Developer Conference. © Philip Kufeldt. All Rights Reserved.

Scaleout Is Disaggregated

Ceph Example

- Each drive seen as network entity
  - An example of eusocial storage, but actors tied to a server
- Problems
  - Media is still aggregated, i.e. local
  - Fixed hardware
    - Ratio of Drives to Server fixed at purchase time
  - Failure domain
    - Server domain
  - Dedicated servers
    - Although disaggregated from clients OSD must run on the server attached to drive
    - Could use app servers and hyper-converge, Rook does this, but consume app resources
  - No easy way to define classes of services, i.e. inter castes cooperation

- e.g. Ceph Scaleout Head



2018 Storage Developer Conference. © Philip Kufeldt. All Rights Reserved.

Kinetic, Disaggregated Media

- An attempt to disaggregate HDDs
- Used TCP/IP interconnect and a Key Value storage interface (no block)
  - Basic Get/Put/Del interface
  - Had both Batch and Iterator interfaces
  - Had Peer 2 Peer (P2P) data movement capabilities
    - Under utilized feature
- Problems
  - Applications needed to be rewritten
  - Most software efforts were to retro-fit storage applications to Kinetic
    - Nobody really tried to rearchitect for the capabilities
    - P2P not in existing SW so retro fits never leveraged these features
  - No multi vendor
  - No software to create unified colony

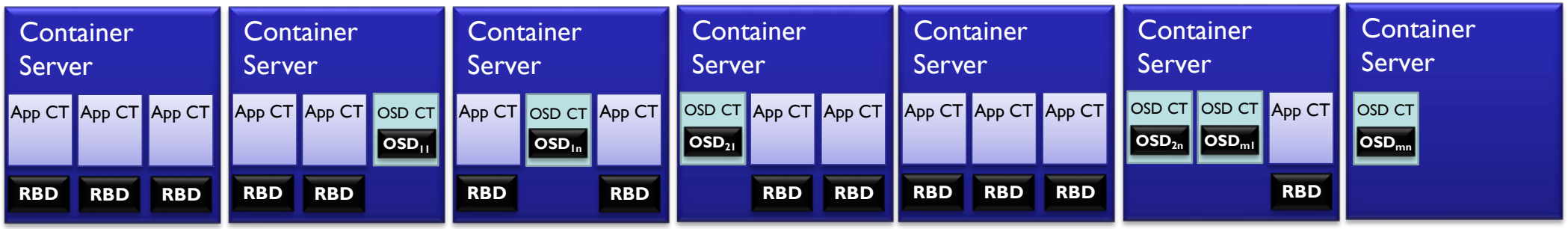
2018 Storage Developer Conference. © Philip Kufeldt. All Rights Reserved.

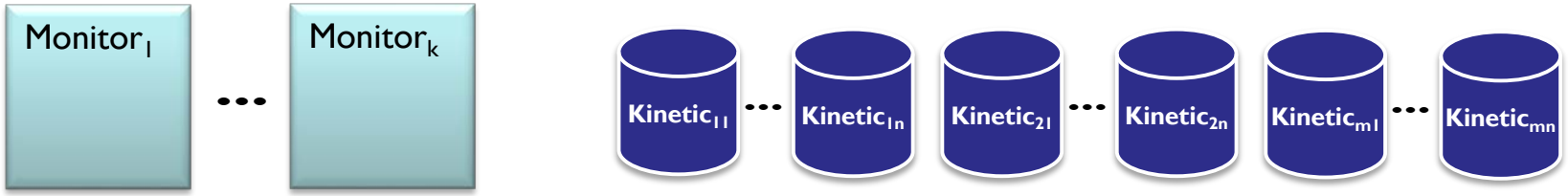
NVMe Disaggregation

- NVMe over Fabrics is the current media disaggregation mechanism
  - IB, FC, RDMA, and iWarp current standards
    - TCP being defined with favorable results
  - Could be at either array/LUN or media level
  - Still block access mechanism
    - Impossible to scale out with out server
  - Allows scale out solutions with scale out heads
- NVMeoF TCP could be a game changer
  - Provides commodity disaggregation solutions
  - Media could be armed with ethernet only
  - Moves in the direction of Kinetic

2018 Storage Developer Conference. © Philip Kufeldt. All Rights Reserved.

Possible Microservices with Kinetic/NVMe-oF





Still remains Fan In Fan Out

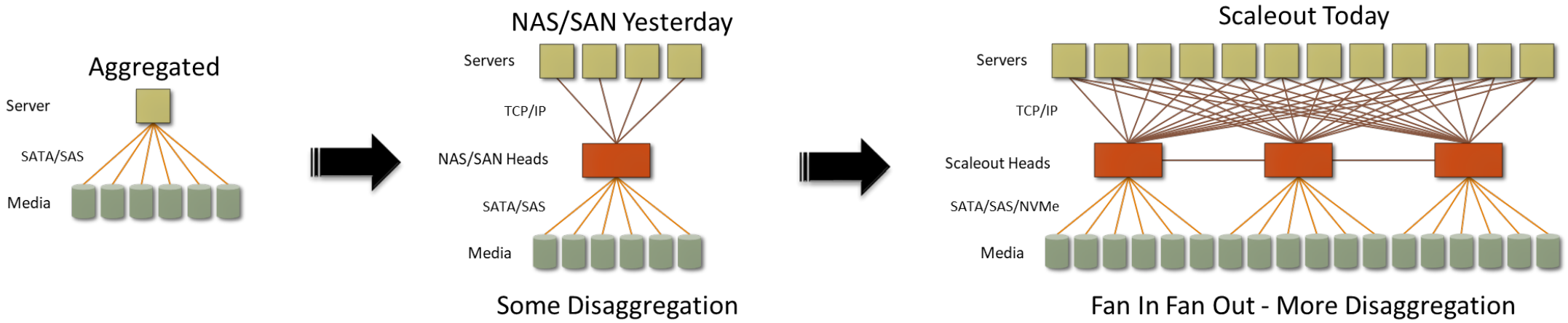
- No offloading
  - All data management in server
  - Steals DMA bandwidth
  - Steals CPU cycles
- No class of service
- No in store compute

Can now performance load-balance Ceph, by migrating OSD CTs

Rook is almost there but right now but need disaggregation (e.g. NVMe-oF)

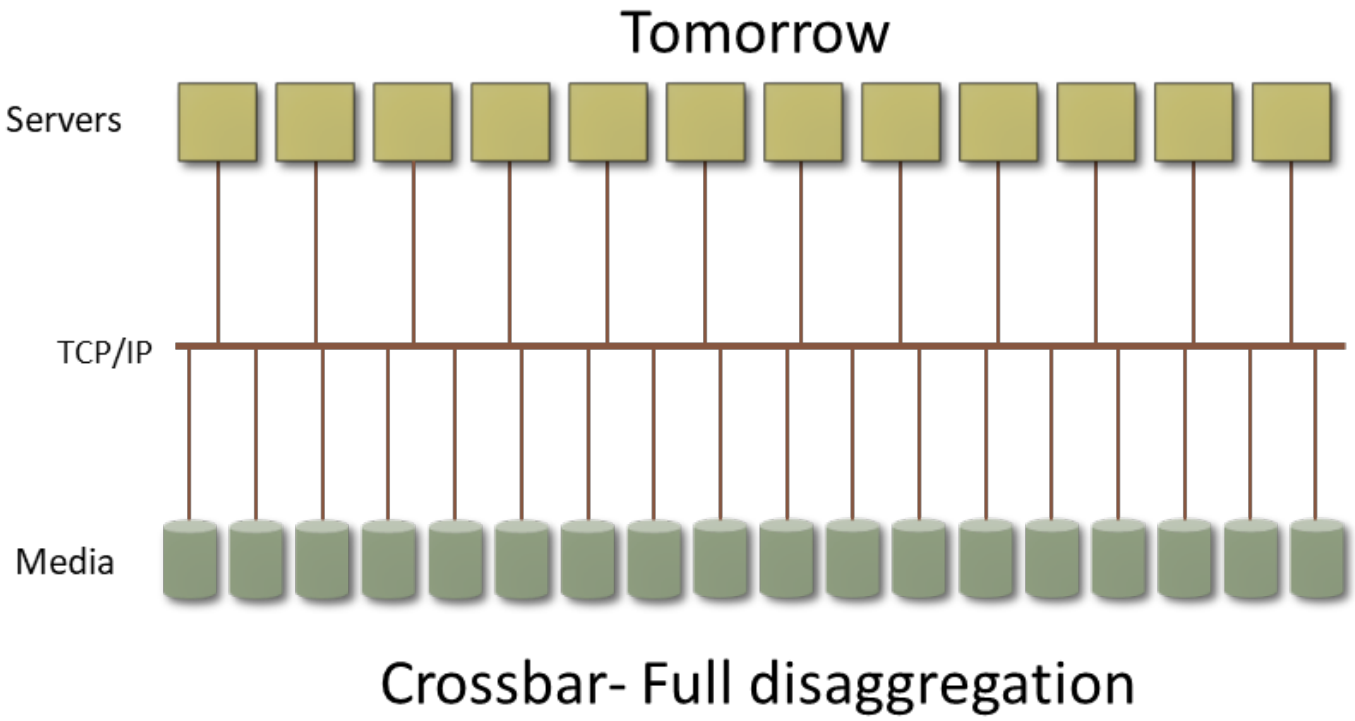
2018 Storage Developer Conference. © Philip Kufeldt. All Rights Reserved.

What is this suggesting

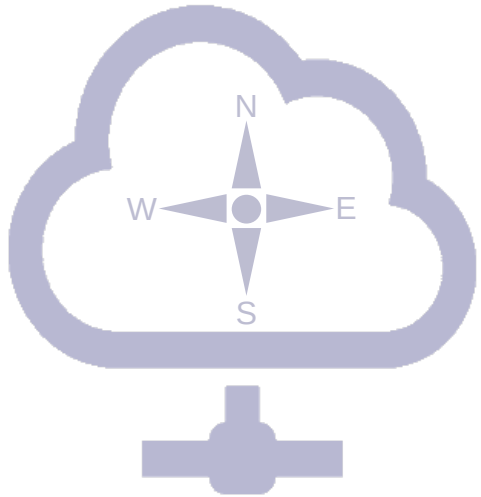


2018 Storage Developer Conference. © Philip Kufeldt. All Rights Reserved.

What is this suggesting



All elements can freely talk



North and South as well as East and West

2018 Storage Developer Conference. © Philip Kufeldt. All Rights Reserved.

What is Eusocial Storage

Eusocial Storage

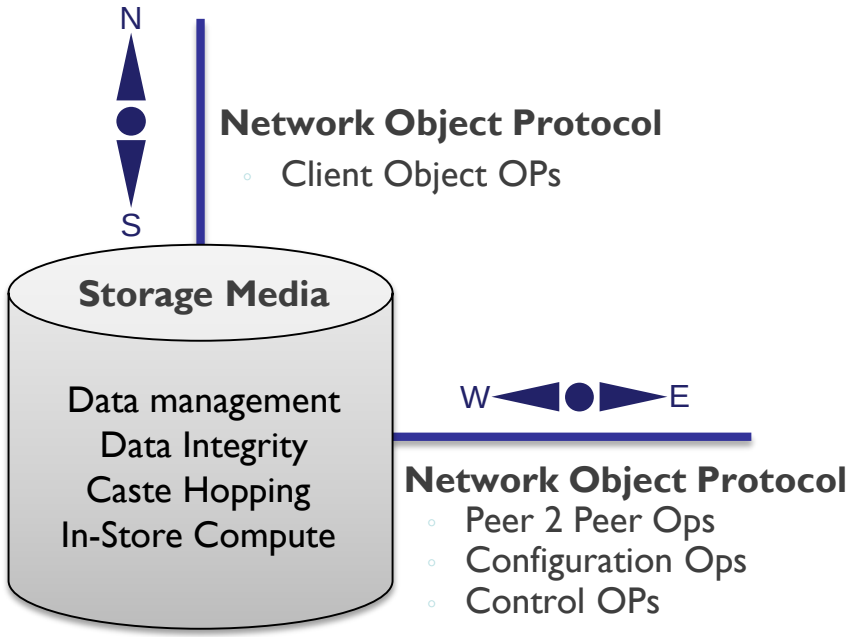
- Crossbar concept makes storage elements independent actors
  - To provide an overall storage system they must cooperate
    - For data availability
    - For scaling capacity
    - For scaling access
    - To create consistent lines of service
    - To create classes of service that weave together lines of service
- Eusocial storage
  - Must have north south (access) APIs
    - Devices can be individuals or grouped
  - Must have east west (organizational) APIs
    - Scale out and provide for redundancy
    - Organize scaleout groups around QoS/Lines of Service
    - Link lines of service together to create ILM
  - Since computational resources are needed, computational storage is a goal

2018 Storage Developer Conference. © Philip Kufeldt. All Rights Reserved.

What is Eusocial Storage

A software abstraction

- Standardized Object Protocol
  - Network/Fabric based
    - Disaggregates
  - Mechanism based
    - Policy is configured
  - Cluster Operations
  - Client Object Operations
  - Peer 2 Peer Operations
  - Control Operations
- Configuration aware
- Data Integrity Mechanisms
- Tiering Mechanisms
- Improved Failure Domains
- Improved Placement/Rebalancing
- Will support In-Store Compute



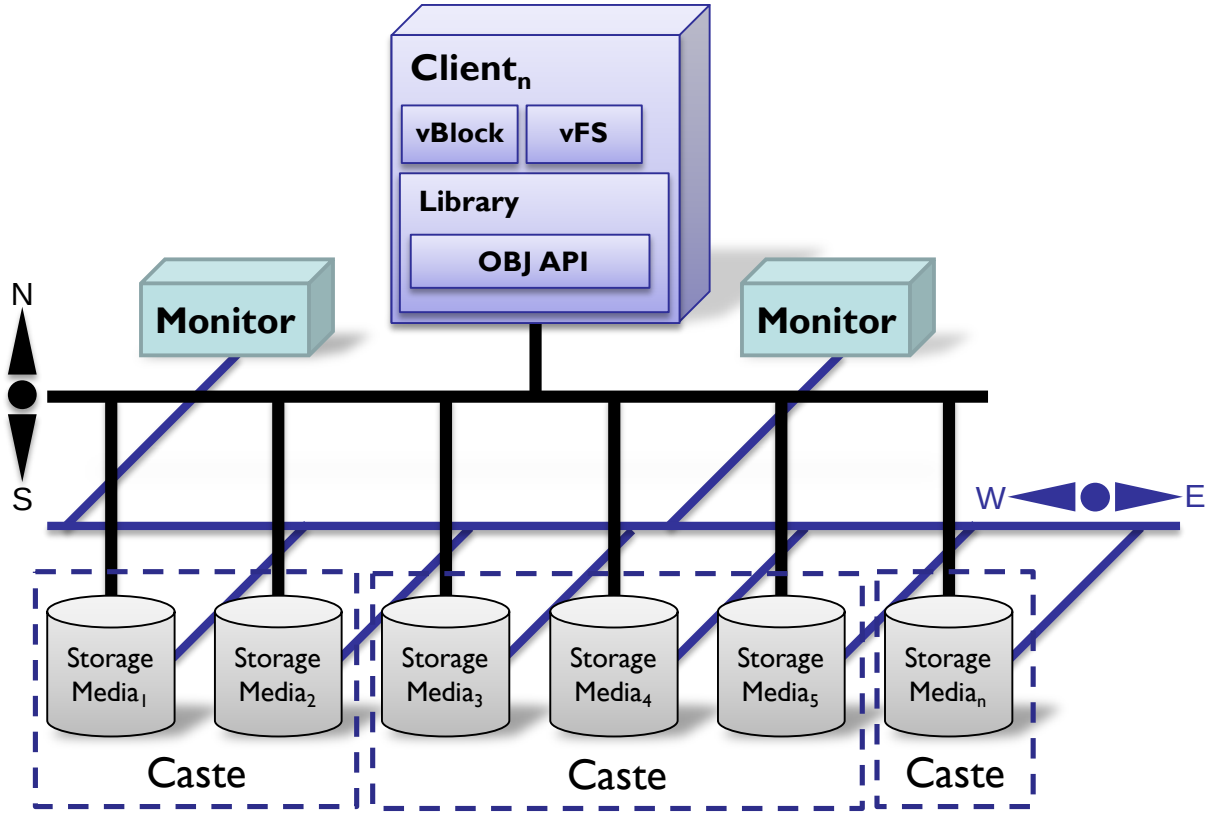
Does not define hardware

- Highly optimized for media type
  - Can be any combination of HW
  - Could be a
    - An ethernet connected SSD
    - Small server and HDDs
    - Gateway to S3
- Must be fabric attached media
  - Must support bi-directional communications
  - Public/private paths recommended
- No restrictions on
  - Media type
  - Form factor
  - Capacity
  - Components
  - Fabric type

2018 Storage Developer Conference. © Philip Kufeldt. All Rights Reserved.

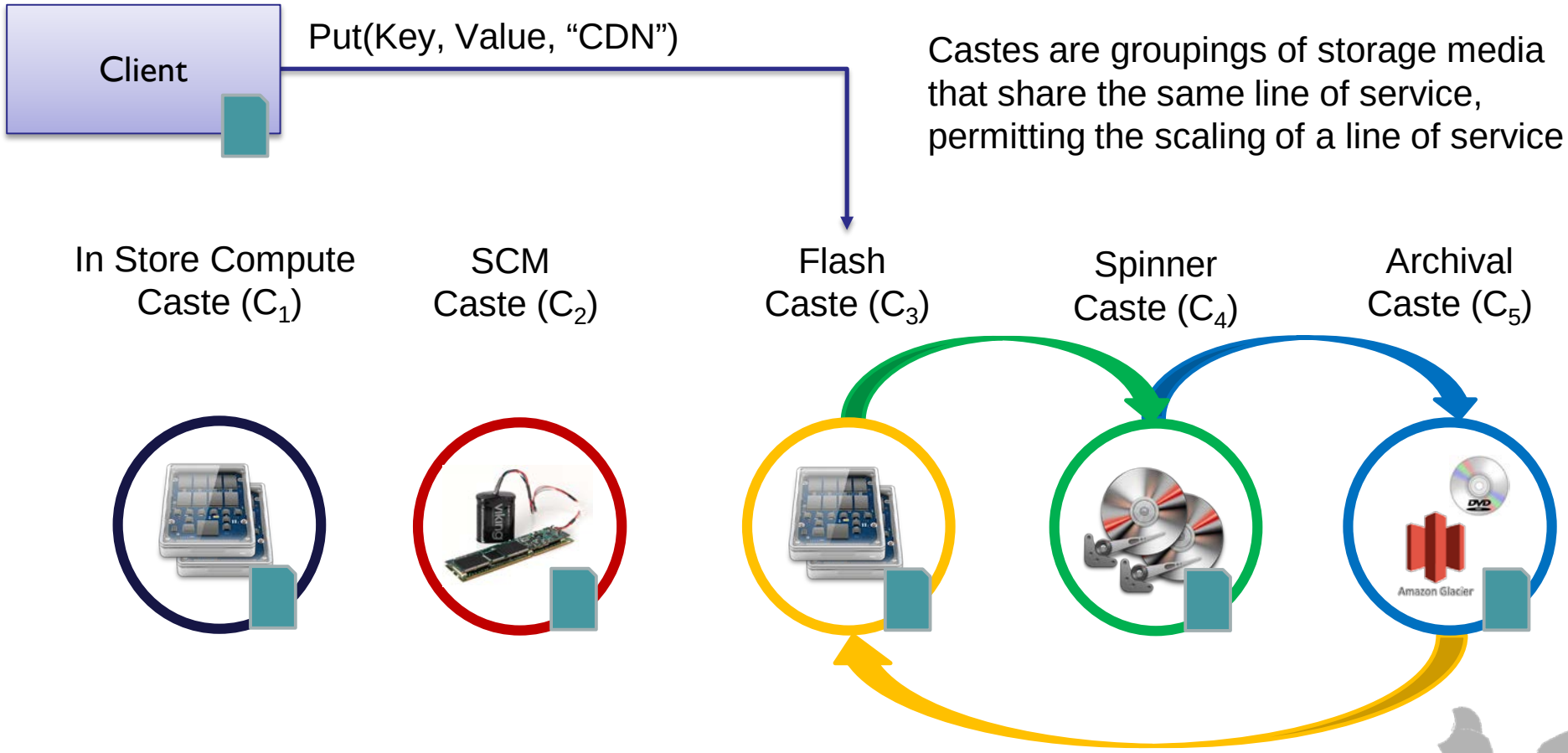
How is Eusocial Storage Organized

- Storage Media
  - Highly optimized, autonomous units of storage
  - Define lines of service
    - Throughput, latency, media type, compute availability
- Castes
  - Groups of Storage Media that provide similar lines of service
    - Similar media, throughput, latency
    - Similar functionality, each member is a replacement for the others
  - Permit the scaling of a line of service
  - Defines availability
    - Eraser coded caste, replicated caste
- Cluster
  - Defines all Storage Media and Castes
  - Manages events and cluster configuration (cluster map)
  - Can be hierarchically managed on global and caste levels



2018 Storage Developer Conference. © Philip Kufeldt. All Rights Reserved.

Eusocial Storage Castes

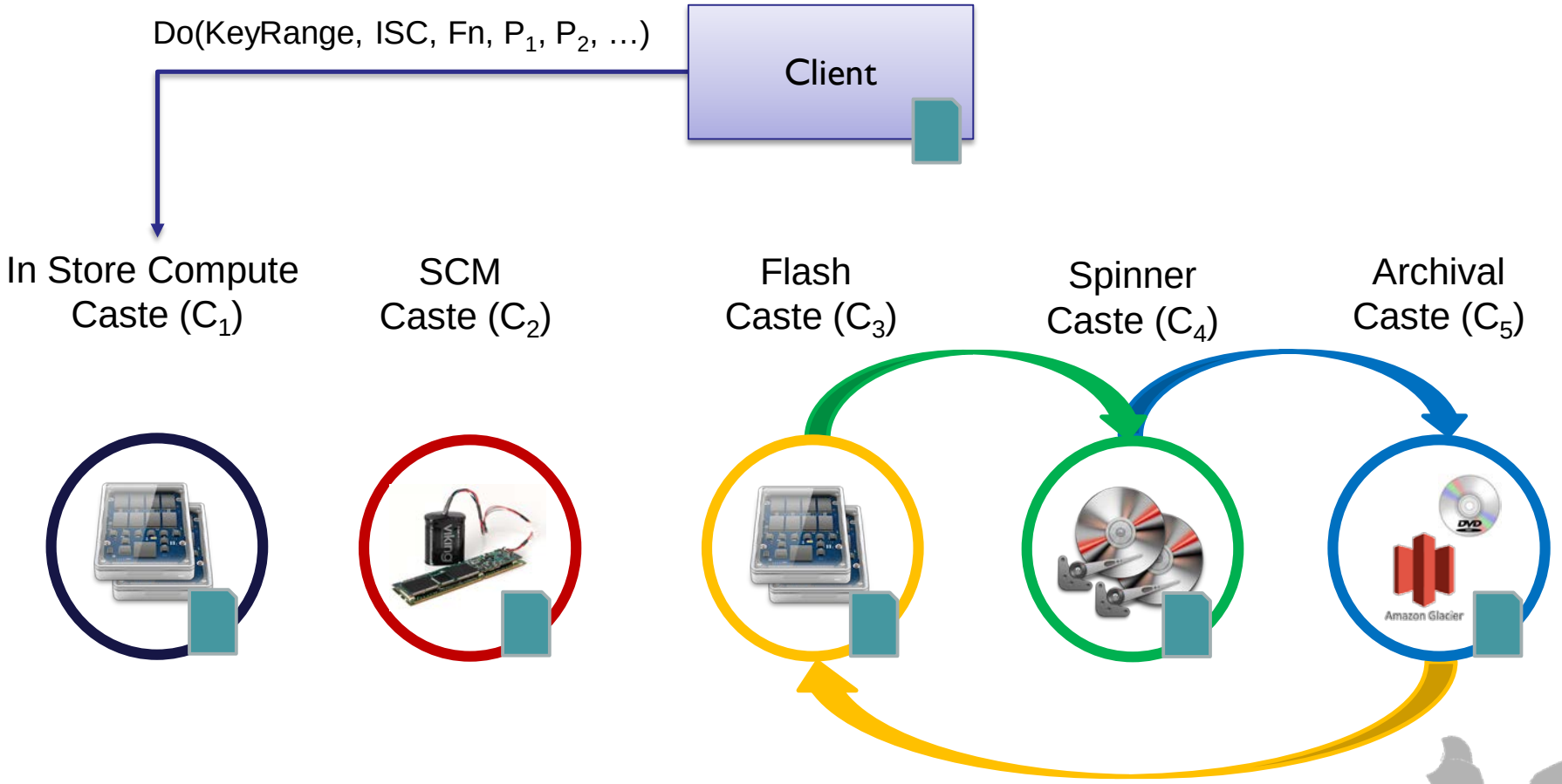


2018 Storage Developer Conference. © Philip Kufeldt. All Rights Reserved.

Eusocial Storage Castes

Cluster Map
SM_1 … SM_n
C_1: SM_1 … SM_i
C_2: SM_{i+1} … SM_j
C_3: SM_{j+1} … SM_k
C_4: SM_{k+1} … SM_n
C_5: SM_{l+1} … SM_n

Class Of Service
Tag: ISC
Graph: C_1
Tag: CDN
Graph: C_3->C_4->C_5->C_3
Transitions: C_3C_4: 12h
C_4C_5: 1w
C_5C_3: Read



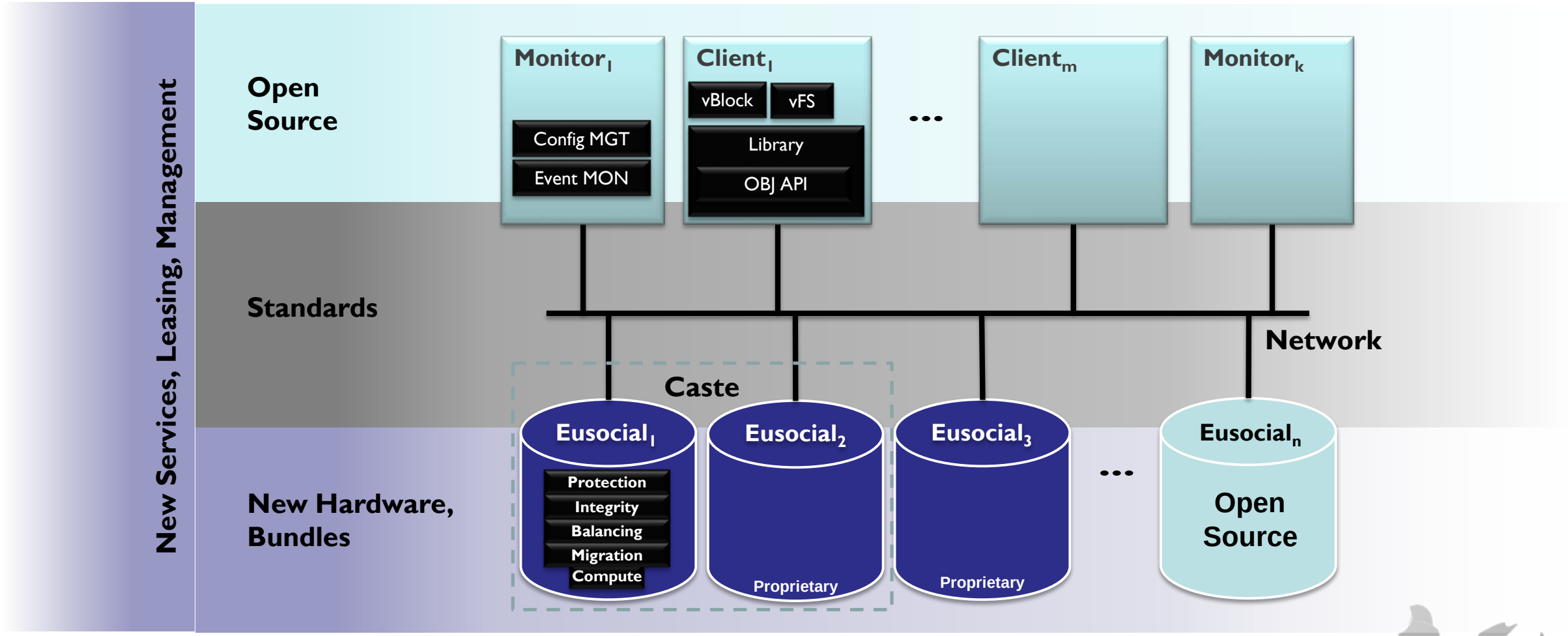
2018 Storage Developer Conference. © Philip Kufeldt. All Rights Reserved.

Eusocial Storage: Why now?

- Private Cloud
  - Basis for more functional and flexible scale out
  - Full disaggregation is not suited to block interface
    - Data location is better done with key value
- Offloads server for both Scaleout and KV store use cases
  - Data movement for data management can be reduced and in some cases removed completely
- Supports existing scaleout, like Ceph
- Can support multiple castes of storage
  - Each caste provides for a different class of service
  - Objects can have life cycles that migrate them through castes
- Permits in storage compute

2018 Storage Developer Conference. © Philip Kufeldt. All Rights Reserved.

Eusocial Storage Eco System



2018 Storage Developer Conference. © Philip Kufeldt. All Rights Reserved.

UCSC CROSS

EUSOCIAL RESEARCH PROJECTS

Eusocial Research Projects

- Offload Evaluation
  - Empirically validate offload strategy
- Full API definition
- Efficiency gains
- In store compute

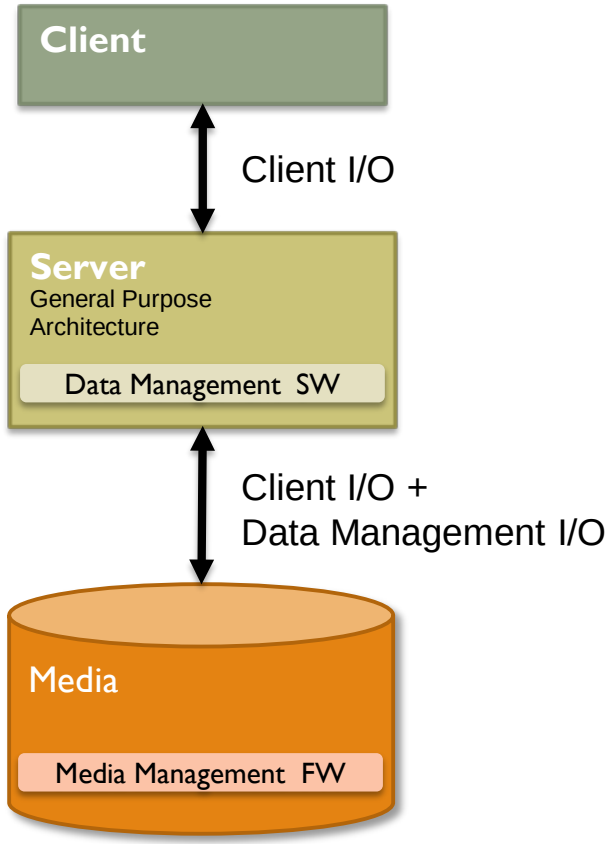
2018 Storage Developer Conference. © Philip Kufeldt. All Rights Reserved.

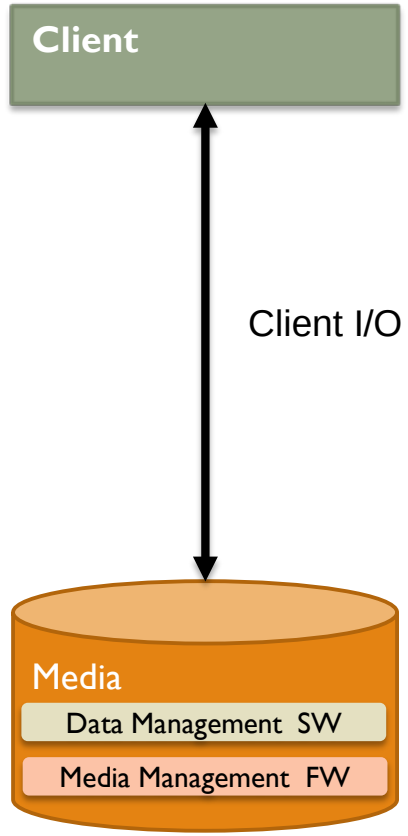
Offload Evaluation

EUSOCIAL BENEFITS

Question

Is there a quantifiable benefit to offloading and consolidating data management on the media?





2018 Storage Developer Conference. © Philip Kufeldt. All Rights Reserved.

What we need

- Need a defined unit of work that can be equally done in any environment
  - Should be independent of the environment it is performed on
  - Some environments may be capable of doing multiple units
  - While other environments may only be capable doing a fraction
- Once have we have this work unit (WU) defined look at cost of doing the work
  - $/WU (CapEx)
  - kW·Hr/WU (OpEx)
  - ManageHr/WU(OpEx)
- We can then measure the WU for many environments and compare them
  - Too simple huh?
- The problem is what is a Work Unit?

2018 Storage Developer Conference. © Philip Kufeldt. All Rights Reserved.

First simplify

- Consider only North and South features
  - Avoid getting lost in network details of East and West to begin
- Use local fabrics
  - Local fabrics are dedicated resources
  - Keep the fabric cost out of the question
  - Keep fabric sharing out of the question
- Avoid getting lost in platform details
  - Cycles, configuration, etc.
  - Leave the Intel vs AMD vs ARM vs RISC-V to the cost and power evaluation
- Use hardware that can be consistently, reproducibly and equally saturated as the basis

2018 Storage Developer Conference. © Philip Kufeldt. All Rights Reserved.

Introducing the Media Based Work Unit

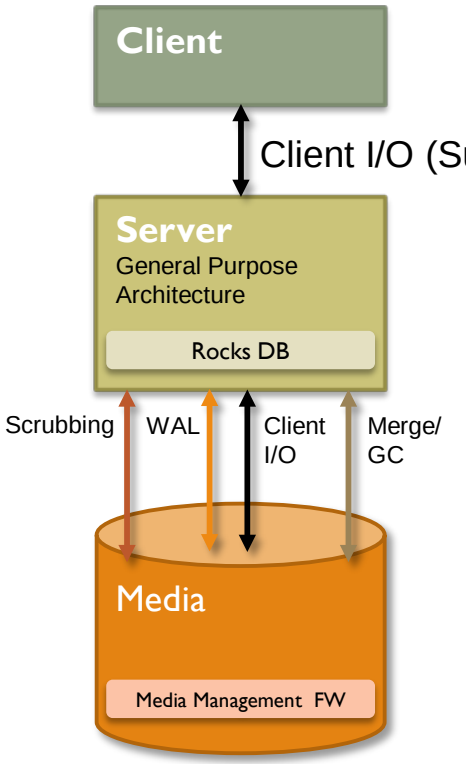
- MBWU (/mi ˈbē wo͞o/)
  - Some load over time
  - Defined in terms of media not access platforms
    - Media remains consistent between access platforms
    - Effect: To compare access platforms, same media must be used
    - As with everything in storage, it will be workload dependent
  - MBWU = maximum sustained transactions per second, where the underlying media is saturated
    - when defining MBWU no other restrictions other than media must be experienced
      - Only bottleneck is the media
  - Sustained transaction rate must include
    - Client IO + Data management IO

2018 Storage Developer Conference. © Philip Kufeldt. All Rights Reserved.

Example

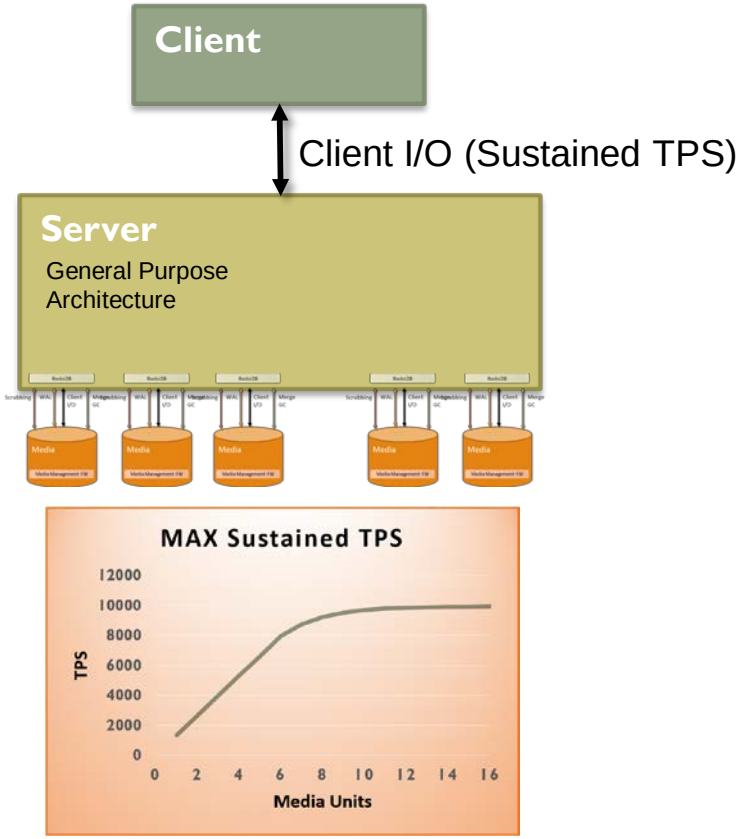
Sustained TPS / Sustained TPS_{mbwu} = %MBWU

Measure 1 MBWU

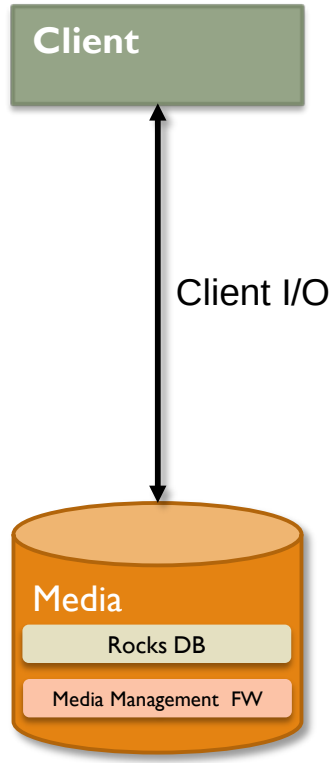


Max out a single SSD

Multiple MBWUs



Fractional MBWUs



Sustained TPS = Sustained TPS_{mbwu}

2018 Storage Developer Conference. © Philip Kufeldt. All Rights Reserved.

Fini