SDC 20

BY Developers FOR Developers

Storage Developer Conference
September 22-23, 2020

Enabling Ethernet Drives

Mark Carlson
Kioxia

The Evolution of Storage Networks

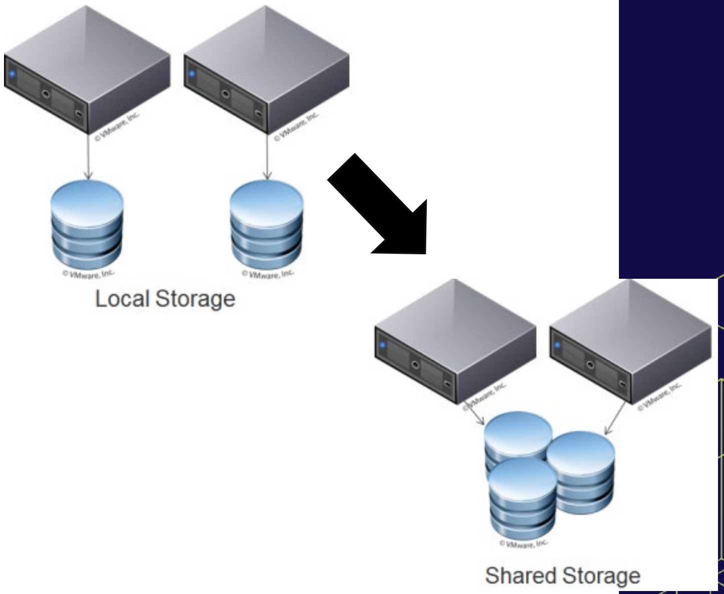
- Direct attached storage: Single host owns storage
- Storage Area Networks: Multiple hosts share storage
  - Avoid “silos” of storage and enables storage efficiencies
  - Examples include Fibre Channel & iSCSI storage networks
    - But require “Storage Controllers” to front storage
- Hyperscale: DAS storage on commodity systems
  - Special software manages many hyperscale nodes in a solution
- Industry moving to NVMe / NVMe-oF™ technology
  - Now, systems AND devices on native Ethernet as a Storage Network

The Ethernet as a Storage Network

- Initially, just a transport
  - End points performed all the storage services (iSCSI)
- Use of Ethernet matured: Specialized protocols
  - Key/value protocol to access data in mainframe context
  - Object protocol to access massive amounts of unstructured data
- Now, NVMe over Ethernet: Storage in a queuing paradigm
  - High performance / low latency / few or no processing blockages
  - No longer gated by transaction paradigm (wait for ACK)
- Next step, NVMe over Ethernet to the drive
  - Removes “Storage Controller” processing blockage

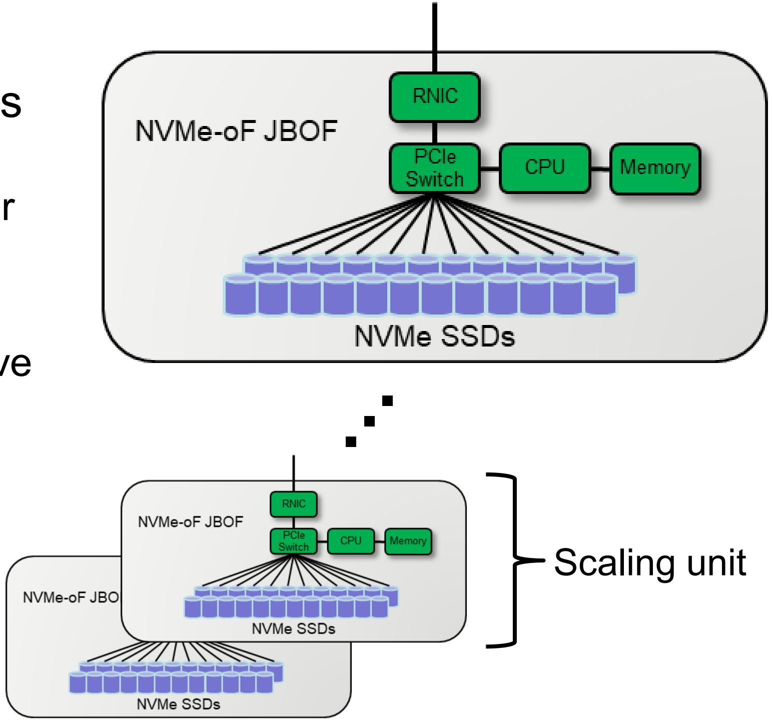
NVMe over Fabrics (NVMe-oF)

- Sharing NVMe based storage across a Network
  - Better utilization: capacity, rack space, power
  - Better scalability: management, fault isolation
- NVMe-oF standard at NVMe.org
  - 50+ contributors
  - Version 1.0 released in 2016
  - Fabrics: Ethernet, InfiniBand, Fibre Channel
- Products now in the market from most major storage system vendors



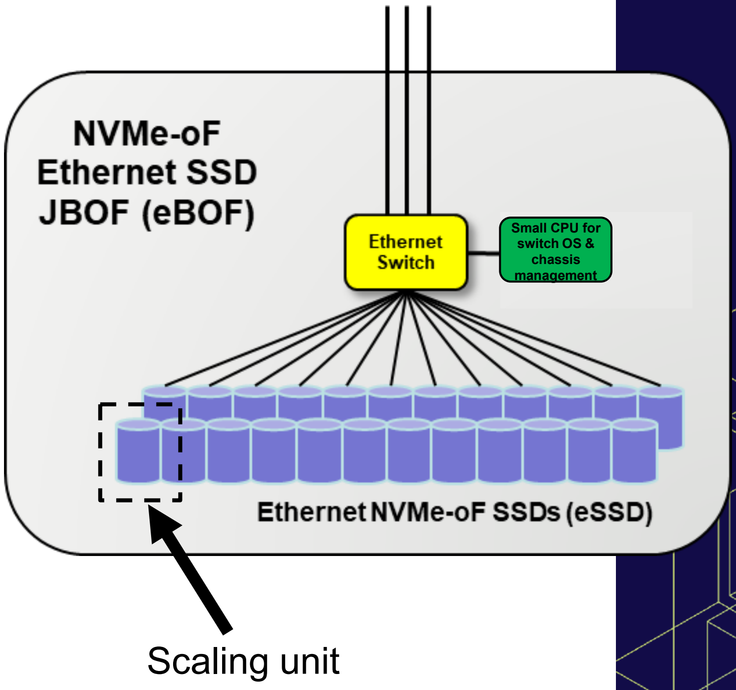
NVMe-oF Storage Targets Today

- Systems terminate the NVMe-oF connection and use PCIe based SSDs internally
  - SSDs behind an array/JBOF controller
- Performance Limits
  - SSD performance increasing faster than CPU NVMe-over-Ethernet-to-drive use cases
  - NIC performance
  - Latency - Store and Forward architecture
- Cost – CPU, SOC/rNICs, Switches, Memory don't scale well to match increasing SSD performance



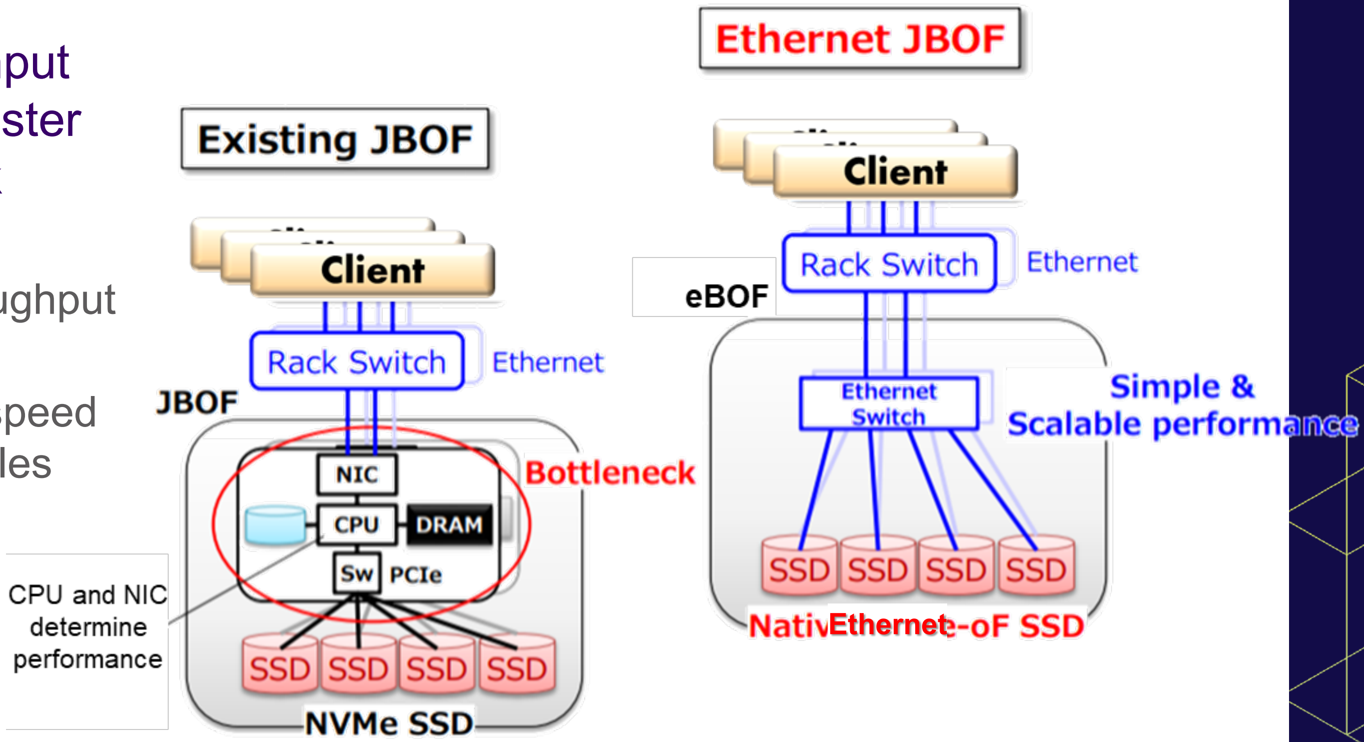
NVMe-oF Ethernet SSDs

- With NVMe-oF termination on the drive itself, controller functionality is now distributed
  - Scaling point becomes a single drive in an inexpensive enclosure
  - Enables eBOFs (Ethernet-attached Bunch Of Flash)
    - Power, cooling, SSDs, and an Ethernet Switch
- Does this make each drive more expensive?
  - Maybe initially, but now customer buys their “controller” incrementally, as needed for new capacity
  - Efficiencies of scale now are applied to controller functionality
  - Lower cost/bandwidth and cost/IOPS



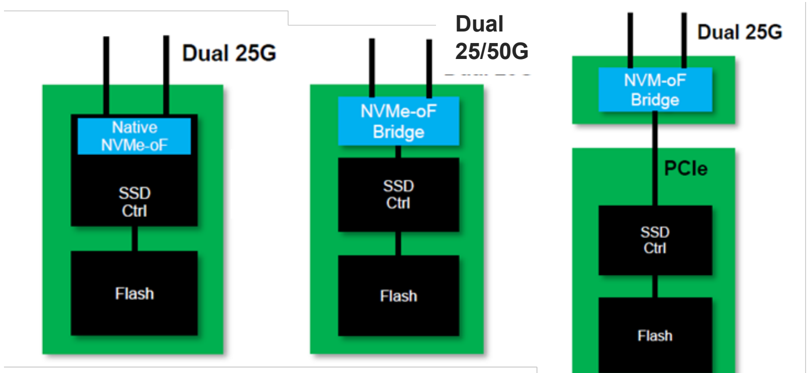
JBOF CPU/NIC Complex can be a Bottleneck

- SSD throughput increasing faster than network bandwidth
  - SSD throughput will triple
  - Network speed only doubles



eSSDs

- Different eSSD designs today
- Some will support multiple interfaces and protocols
  - Ethernet, PCIe, SAS, SATA
  - RoCE, TCP

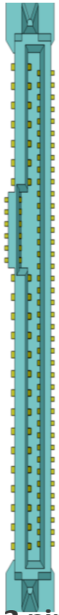


Integrated with SSD controller

On-board chip

External bridge/dongle

Name	Pin
GND	S1
S0T+ (A+)	S2
S0T- (A-)	S3
GND	S4
S0R- (B-)	S5
S0R+ (B+)	S6
GND	S7
RefClk1+	E1
RefClk1-	E2
3.3Vaux	E3
ePERst1#	E4
ePERst0#	E5
RSVD	E6
RSVD(Wake#) /SASAct2	P1
sPCIeRst/SAS	P2
RSVD(DevSLP#	P3
IfDet#	P4
Ground	P5
	P6
5 V	P7
	P8
	P9
PRSNT#	P10
Activity	P11
Ground	P12
12 V	P13
	P14
	P15



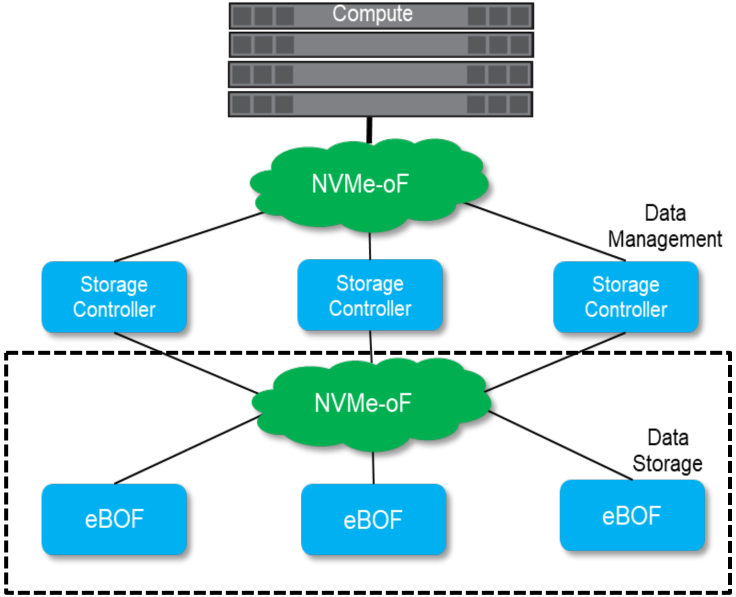
Pin	Name	SAS & Ethernet Signals proposal1	PCIe & Ethernet Signals proposal2
E7	RefClk0+		
E8	RefClk0-		
E9	GND		
E10	PETp0	TX1+	
E11	PETn0	TX1-	
E12	GND		
E13	PERn0		RX0-
E14	PERp0		RX0+
E15	GND		
E16	RSVD		
S8	GND		
S9	S1T+		
S10	S1T-		
S11	GND		
S12	S1R-	RX1-	
S13	S1R+	RX1+	
S14	GND		
S15	RSVD		
S16	GND		
S17	PETp1/S2T+		TX0+
S18	PETn1/S2T-		TX0-
S19	GND		
S20	PERn1/S2R-	RX0-	
S21	PERp1/S2R+	RX0+	
S22	GND		
S23	PETp2/S3T+		TX1+
S24	PETn2/S3T-		TX1-
S25	GND		
S26	PERn2/S3R-		
S27	PERp2/S3R+		
S28	GND		
E17	PETp3	TX0+	
E18	PETn3	TX0-	
E19	GND		
E20	PERn3		RX1-
E21	PERp3		RX1+
E22	GND		
E23	SMClk		
E24	SMDat		
E25	DualPortEn		

Fig1. U.2 pin assignment

SFF-8639 connector

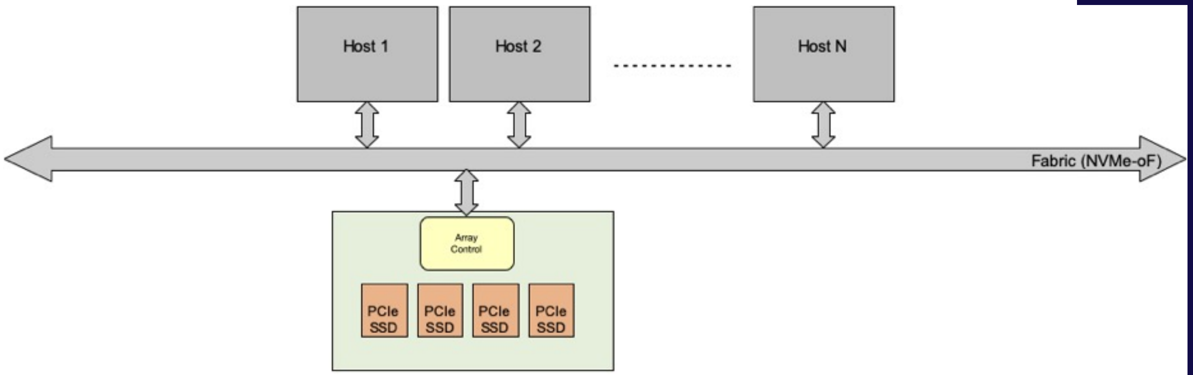
Use Case: Behind the Controller

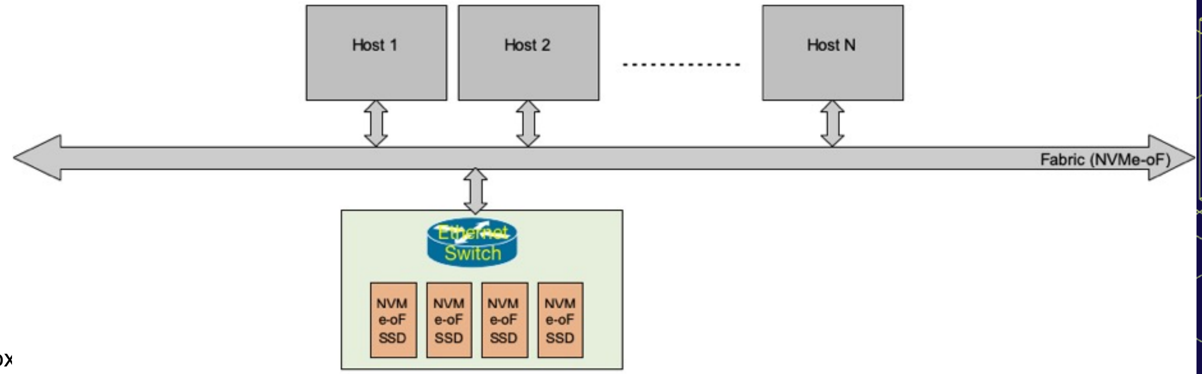
- Scale storage capacity with large pools of disks
  - Many NVMe SSDs in many enclosures
  - PCIe only scales so far and at JBOF increments
- Using eSSDs allows much higher scaling
  - Still hiding individual SSD management from users
- Data services in the storage controllers ➔ value add
  - Orchestration between hosts and large pools of disks
    - Whole disks or slices of disks that provide massive pools effectively
  - Robust data protection schemes / distributed solution controllers



Use Case: Disaggregated SSD Storage

- Today: Array controller handles conversion from NVMe-oF to PCIe based drives
- With eSSD: Ethernet drives only require an Ethernet Switch and fit into an eBOF for power and cooling





SNIA Native NVMe-oF Drive Specification

- Discover and Configure: the drives, their interfaces, the speeds, the management capabilities
- Connectors
  - Some connectors may need to configure the PHY signals based on the type of drive interface
  - Survivability and mutual detection is important
- Pin-outs
  - For common connectors and form factors
- NVMe-oF integration
  - Discovery controllers / Admin controllers
- Management
  - Through Ethernet/TCP for Datacenter-wide management

Management

- Scale out orchestration of 10’s of thousands of drives possible by using a RESTful API such as DTMF Redfish™
- Redfish/SNIA Swordfish™ follow a principal that each element report it’s own management information
  - Follow links in higher level management directly to the drive’s management endpoint
  - HTTP/TCP/Ethernet based
- NVMe-oF Drive Interoperability Profile
  - Mock up to start
  - Push new models through Swordfish contributions
  - Publish Interoperability Profile at DMTF
- Map the profile to NVMe & NVMe-MI properties and actions

The Latest Joint Work: Mapping NVMe to RF and SF

- A three-way effort, hosted by the SNIA SSM TWG (develops Swordfish)
- Chartered work from RF/SF/NVM Discussions in 2019:
  - How will NVMe (and NVMe-oF) be managed in large scale environments? This is where RF (and SF) are targeting
  - Also: Come up with a common way, that all orgs agree with, to represent NVMe and NVMe-oF in RF/SF (RF had some NVMe drive properties added, but not a comprehensive view)
- Other goals:
  - Provide a clear "map" for NVMe folks that don't know RF/SF to understand
  - Provide commonality where possible between existing RF and SF models and NVMe solutions

Fitting the Standards Together

- RF/SF use the available low-level transports to get device / transport specific information into the common models
  - RF/SF uses the commands that are provided in the NVMe/NVMe-oF/NVMe-MI specs
  - NVMe-MI can be used as the low-level to get the information into the high-level management environment as OOB access mechanism when appropriate
- Scope:
  - NVMe Subsystem, NVMe-oF and NVMe Domain Models

A Partial View: The Overall NVMe Subsystem Model

- Model reflects a unified view of all NVMe device types.
- Devices will instantiate an appropriate subset of the model

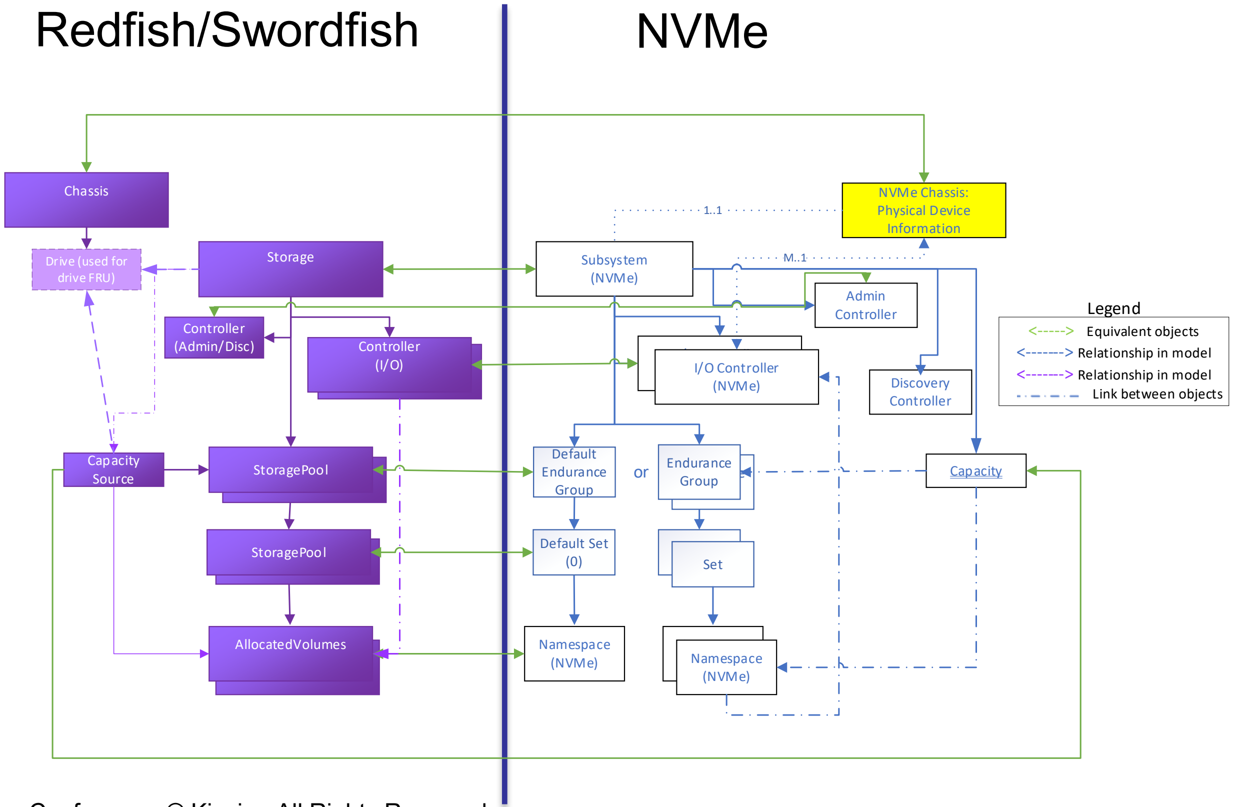
- The model diagrams do not reflect all available schema elements.

- Model leverages and coarsely maps to existing (Redfish and) Swordfish storage model

Major NVM Objects Mapped to RF/SF

- NVM Subsystem
  - An NVM subsystem includes one or more controllers, zero or more namespaces, and one or more ports. Examples of NVM subsystems include Enterprise and Client systems that utilize PCI Express based solid state drives and/or fabric connectivity.
- NVM Controller (IO, Admin and Discovery)
  - The interface between a host and an NVM subsystem
  - Admin controller: controller that exposes capabilities that allow a host to manage an NVM subsystem
  - Discovery: controller that exposes capabilities that allow a host to retrieve a Discovery Log Page
  - I/O: controller that implements I/O queues and is intended to be used to access a non-volatile memory storage medium
- Namespace
  - A quantity of non-volatile memory that may be formatted into logical blocks. When formatted, a namespace of size n is a collection of logical blocks with logical block addresses from 0 to (n-1)
- Endurance Group
  - A portion of NVM in the NVM subsystem whose endurance is managed as a group
- NVM Set
  - An NVM Set is a collection of NVM that is separate (logically and potentially physically) from NVM in other NVM Sets.
- NVM Domain
  - A domain is the smallest indivisible unit that shares state (e.g., power state, capacity information).
  - Domain members can be NVM controllers, endurance groups, sets or namespaces

NVMe Subsystem Model



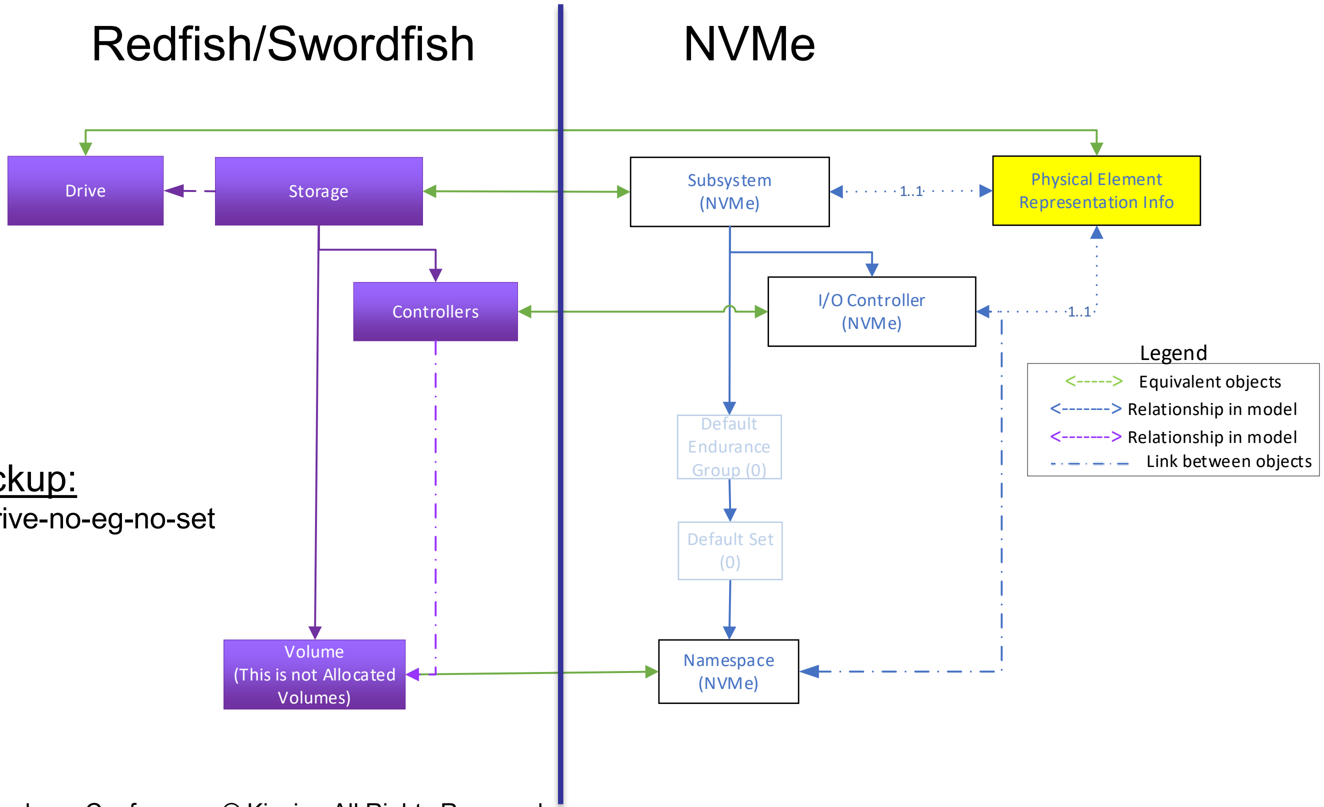
Mapped Models and Documented Permutations

- Device Model – NVMe
  - Simple SSD
    - Default Endurance Group / Default Set
    - Single Endurance Group / Single Set
  - JBOF – PCIe front-end attach to set of drives
  - EBOF – Ethernet front-end attach to set of drives
  - Fabric Attach Array
    - RBOF – Simple RAID front-end attach to set of drives
  - Opaque Array
    - Front end is NVMe, back end is vendor choice (may incorporate existing technologies with NVMe)
- Subsystem (Fabric) Model – NVMe-oF
  - Fabric-attached subsystem presenting logical subsystem, controller, namespace, port and allowed host
  - Simple SSD with NVMe-oF Attach
    - Default Endurance Group / Default Set

  - NVMe Domains

Yellow – TBD Mockups

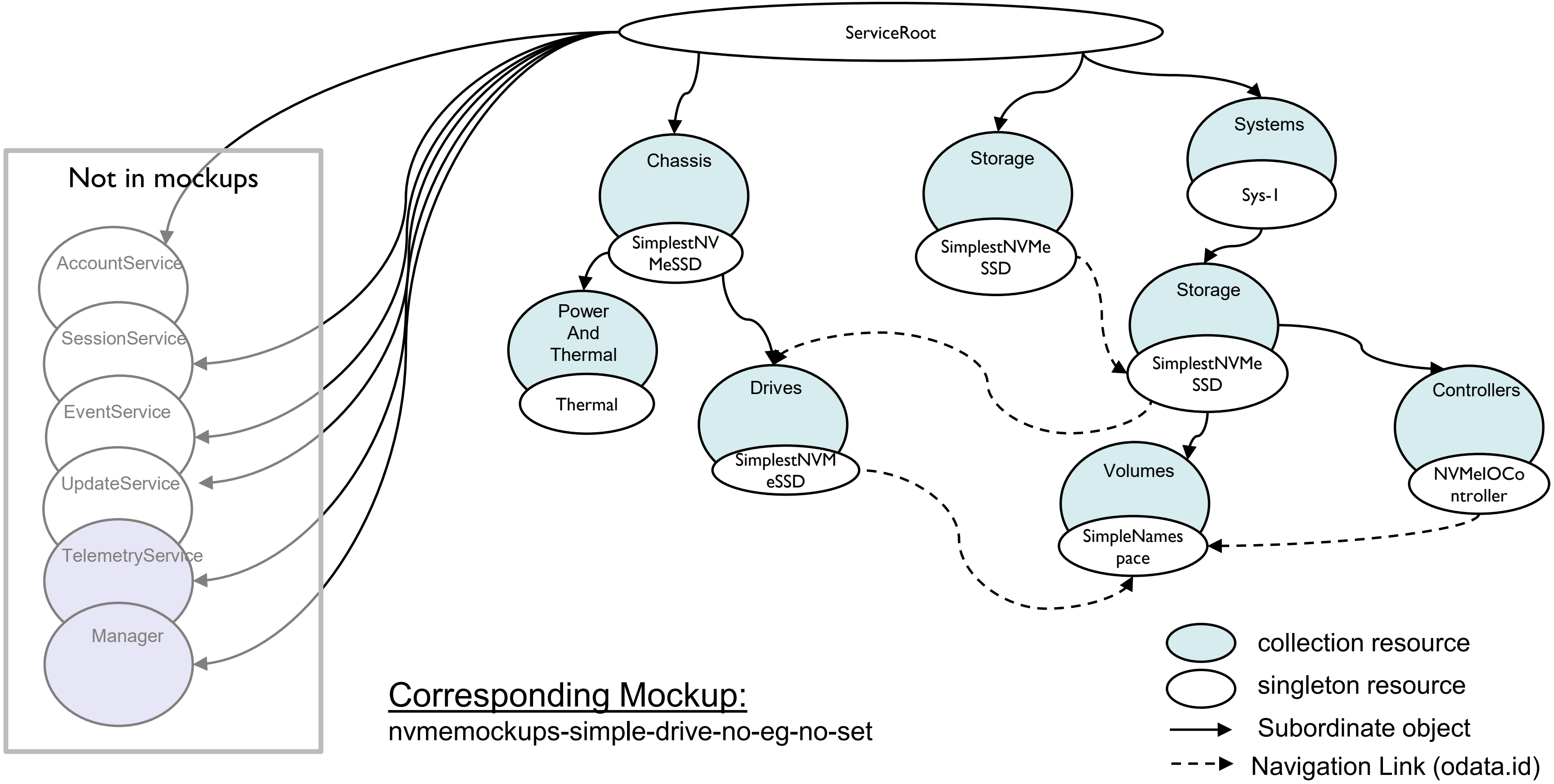
Simple SSD Implementation: Default Endurance Group and Set (Not Implemented in RF / SF)



Corresponding Mockup:

nvmemockups-simple-drive-no-eg-no-set

Simple NVMe Drive: No Namespace Mgmt, No EG, No Set



<u>Corresponding Mockup:</u>
nvmemockups-simple-drive-no-eg-no-set

Who is Developing Redfish and Swordfish*?

Redfish (DMTF) only:

American Megatrends, Inc.
Artesyn Embedded Technologies
Atos
Eaton
Ericsson AB
Google
Insyde Software Corp.
Mellanox Technologies
New H3C Technologies Co., Ltd
NVIDIA
Quanta Computer
Supermicro
Vertiv
Xilinx

Both:

Arm Limited
Broadcom Inc.
Cisco
Dell Inc.
Fujitsu
Hewlett Packard Enterprise
Huawei**
IBM
Intel Corporation
Kioxia
Lenovo
Microchip Technology Inc.
NetApp
Seagate
Western Digital
VMware Inc

Swordfish (SNIA) only:

Hitachi
Inspur
Kalray
Microsoft
NEC Corporation
NGD Systems, Inc.
Pure Storage
Samsung
Silicon Motion, Inc.
SK Hynix
StarWind

*as of February 2020

** Membership suspended