

A background image featuring a complex network of glowing blue nodes connected by thin white lines, set against a dark blue gradient. The nodes are scattered across the left and center of the frame, with some appearing brighter than others. The overall aesthetic is technical and futuristic.

# NVMe Over Fabric and Direct Attached Host with NVMe SSD SR-IOV

Brian Pan  
Aug 2018

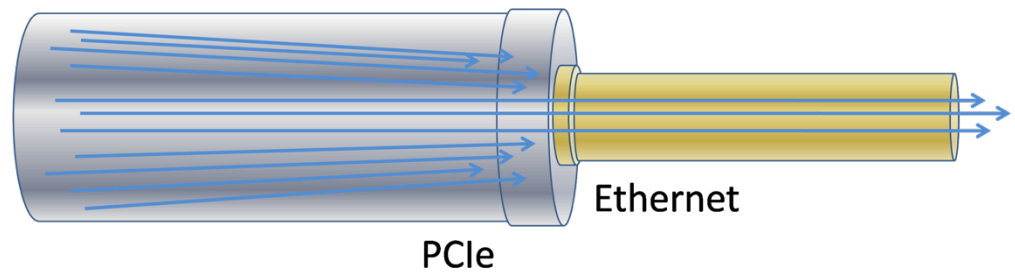


**CONFIDENTIAL**

# Complexity and Bottleneck

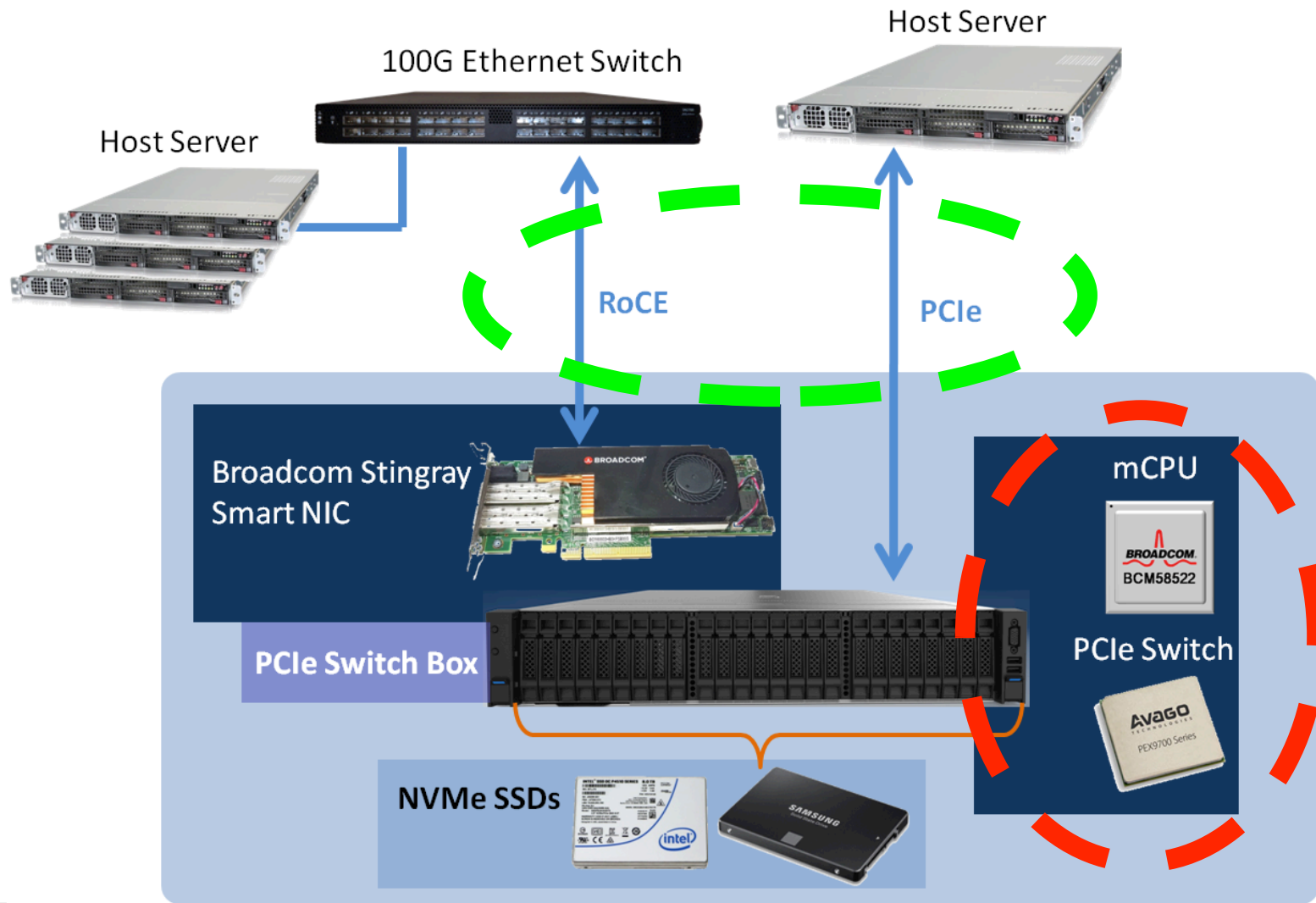


**Power sequence, bus number, memory address, hotplug, OS and driver**



**Performance bottleneck**

# System Architecture-- Direct PCIe and NVMe-oF



**CONFIDENTIAL**

# System Specification

- Host connection
  - 2x 100G Smart NIC for NVMe over fabric
  - 2x PCIe Gen3 x16 for host connection
- NVMe SSD
  - 16x U.2 NVMe SSD (PCIe Gen3 x4)
- System specification
  - 2U form factor
  - 1+ 1 redundant 1400W
  - 4+1 redundant fan



# NVMe SSD Pooling by PCIe Switch

Spec	Details
<b>Dimension</b>	<b>2U</b>
<b>Front Port</b>	<b>16 U.2 NVMe SSD</b>
<b>Back Port</b>	<b>Stingray, management, power</b>
<b>Power</b>	<b>1400W, 1+1 Redundant</b>



## SMART NIC– Stingray

- ARM SoC with 100G smart NIC for ROCEv2

## NVMe SSD device allocation

- Dynamically assign VF of NVMe SSD to smart NIC

## Hot-plug NVMe SSD

- Remove/ add/ re-allocate VF of NVMe SSD from one NIC to another NIC without system shutdown

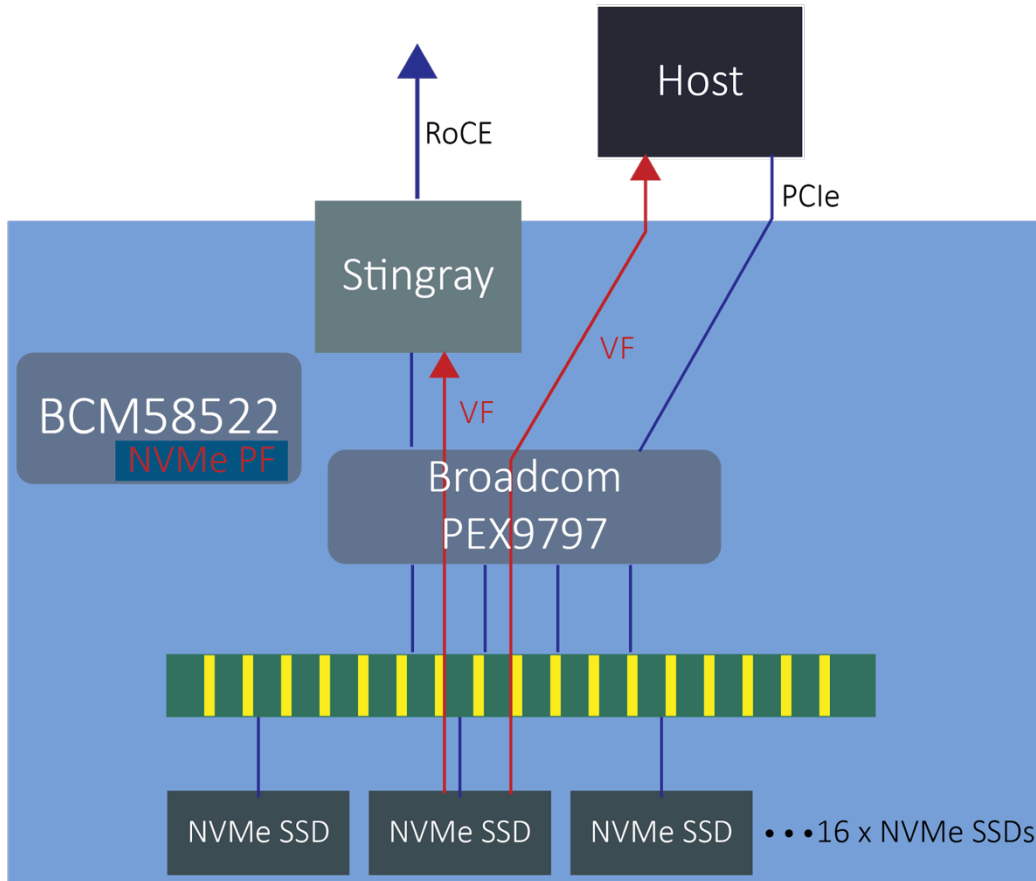
## Host-device port configuration

- Assign the PCIe slot as PCIe host connection or device ports dynamically (restart required)

## Management API

- Follow the redfish standard and integrate with Intel RSD management software

# NVMe-oF System Architecture



## System

- 1x 25G Smart NIC
- 16x NVMe SSD with SR-IOV capability
- 1x 96 Lane PCIe switch
- PF of all NVMe SSD are installed in PCIe mCPU
- Each PF is with 5 VF. 5 VF are assigned to Smart NIC and host.

# Demo of NVMe SR-IOV

## 1. Create Namespace in NVMe

Create Namespace

Port#: 0:2:0:1 Model Name: MODEL SAMSUNG MZWLL6T4HMLS-00003  
 Serial Number: S3H9NX0J700013  
 Total Capacity: 5961.63 GB Available Capacity: 4944 GB

Create Namespace:  GB  Multipath & sharing

Namespace	Capacity (GB)	Multipath
1	1000	--

Buttons: Create, Delete

## 2. Map NS with VF

Attach Namespace with VF

Port#: 0:2:0:1 Model Name: MODEL SAMSUNG MZWLL6T4HMLS-00003  
 Serial Number: S3H9NX0J700013  
 NS ID: 2 Capacity: 2000 GB

Virtual Function ID: 1 2 3 4 5 6 7 8 9 10 11 12 13 14  
 Select VF to attach:

NS	Capacity (GB)	Multipath	VF1	VF2	VF3	VF4	VF5	VF6	VF7	VF8	VF9	VF10	VF11	VF12	VF13	VF14
1	1000	--	v	--	--	--	--	--	--	--	--	--	--	--	--	--
2	2000	--	--	v	--	--	--	--	--	--	--	--	--	--	--	--

Buttons: Apply

## 3. Assign VF to Smart NIC/ host

Allocate Virtual Function

Port#: 0:2:0:1 Model Name: MODEL SAMSUNG MZWLL6T4HMLS-00003  
 Serial Number: S3H9NX0J700013  
 Host Port:  NS ID List: 2

VF ID	Total Number of Attached NS	Allocated Host Port
1	1	0:13:0
2	1	0:14:0
3	0	
4	0	
5	0	
6	0	
7	0	
8	0	
9	0	
10	0	
11	0	
12	0	
13	0	
14	0	

Buttons: Allocate another VF to port14.  
 Port14 is NVMe over Fabric by Smart NIC.

## 4. Run FIO test

```

root@ubuntu1111:/usr/home/mhs
# fio --name=fio --directory=/mnt/ --ioengine=libaio --iodepth=16
o-2.16
Starting 16 processes
bs: 16 [r=0]: [f(16)] [100.0% done] [2567MB/0KB/0KB /s] [657K/0/0 iops] [eta 00m:00s]
nvfme: (groupid=0, job=16): err=0: pid=3573: Fri Aug 3 14:20:43 2018
Description: [fio_read_read 4K 1OPS]
read: io=77669MB, bw=2588.9MB/s, iops=662744, runt= 30001msec
slat (usec): min=25, max=33887, avg=44.81, stdev=31.82
clat (usec): min=159, max=32233, avg=338.66, stdev=73.72
lat (usec): min=153, max=34089, avg=380.49, stdev=79.65
clat percentiles (usec):
| 1.00th=[ 197], 5.00th=[ 229], 10.00th=[ 249], 20.00th=[ 274],
| 30.00th=[ 298], 40.00th=[ 310], 50.00th=[ 330], 60.00th=[ 346],
| 70.00th=[ 370], 80.00th=[ 394], 90.00th=[ 434], 95.00th=[ 466],
| 99.00th=[ 532], 99.50th=[ 564], 99.90th=[ 636], 99.95th=[ 666],
| 99.99th=[ 740]
lat (usec): 250=10.40%, 500=97.27%, 750=2.33%, 1000=0.01%
lat (msec): 20=0.01%
cpu: us=3.57%, sy=20.45%, cx=13154889, majf=0, minf=76
IO depths : 1=0.0%, 2=0.0%, 4=0.0%, 8=0.0%, 16=116.7%, 32=0.0%, >64=0.0%
submit : 0=0.0%, 4=0.0%, 8=0.0%, 16=100.0%, 32=0.0%, 64=0.0%, >64=0.0%
complete : 0=0.0%, 4=0.0%, 8=0.0%, 16=100.0%, 32=0.0%, 64=0.0%, >64=0.0%
issued : total=1533008/4=0/0=0, shortr=0/4=0/0=0, drop=0/4=0/0=0
latency : target=0, window=0, percentile=100.00%, depth=16
Me over Fabric with SR-IOV 0803
nvfme: (groupid=0, job=16): err=0: pid=3573: Fri Aug 3 14:20:43 2018
slat (usec): min=25, max=33887, avg=44.81, stdev=31.82
clat (usec): min=159, max=32233, avg=338.66, stdev=73.72
lat (usec): min=153, max=34089, avg=380.49, stdev=79.65
clat percentiles (usec):
| 1.00th=[ 197], 5.00th=[ 229], 10.00th=[ 249], 20.00th=[ 274],
| 30.00th=[ 298], 40.00th=[ 310], 50.00th=[ 330], 60.00th=[ 346],
| 70.00th=[ 370], 80.00th=[ 394], 90.00th=[ 434], 95.00th=[ 466],
| 99.00th=[ 532], 99.50th=[ 564], 99.90th=[ 636], 99.95th=[ 666],
| 99.99th=[ 740]
lat (usec): 250=10.40%, 500=97.27%, 750=2.33%, 1000=0.01%
lat (msec): 20=0.01%
cpu: us=3.57%, sy=20.45%, cx=13154889, majf=0, minf=76
IO depths : 1=0.0%, 2=0.0%, 4=0.0%, 8=0.0%, 16=116.7%, 32=0.0%, >64=0.0%
submit : 0=0.0%, 4=0.0%, 8=0.0%, 16=100.0%, 32=0.0%, 64=0.0%, >64=0.0%
complete : 0=0.0%, 4=0.0%, 8=0.0%, 16=100.0%, 32=0.0%, 64=0.0%, >64=0.0%
issued : total=1533008/4=0/0=0, shortr=0/4=0/0=0, drop=0/4=0/0=0
latency : target=0, window=0, percentile=100.00%, depth=16

```

Bandwidth is 2588.9 MB/S.  
 Latency is 380.49 usec.



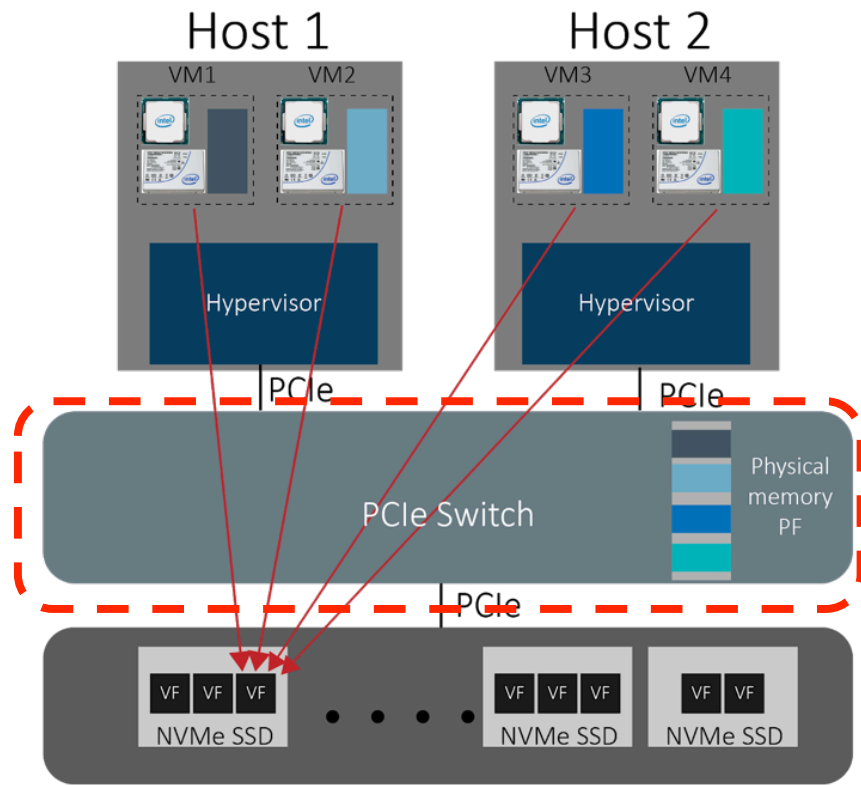
**CONFIDENTIAL**

# FIO Test Result of Direct Attach vs Smart NIC

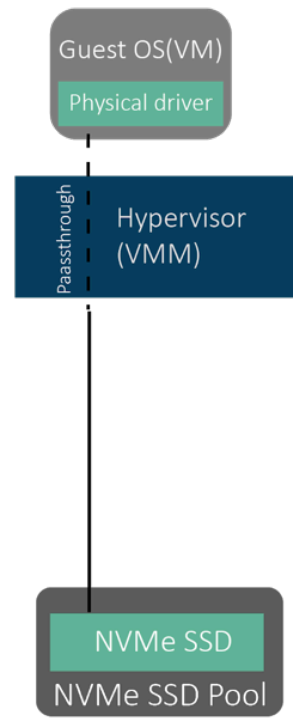
Performance	Throughput	Latency
Directly attach to host	2853.5 MB/s	345.03 usec
NVMe over Fabric	2588.9 MB/s	380.49 usec



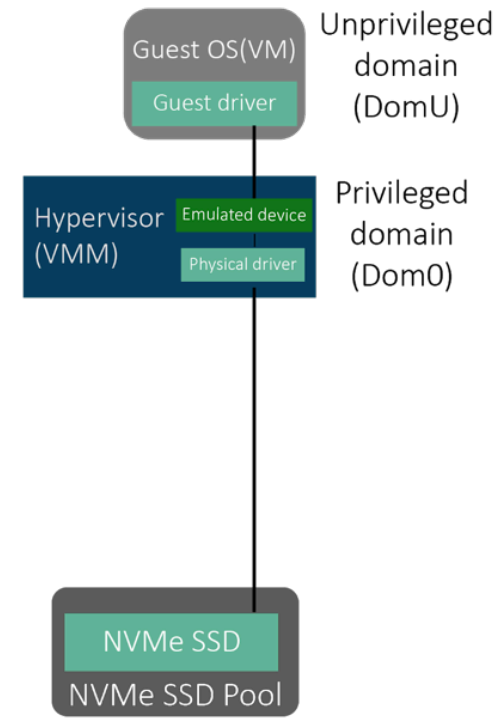
# NVMe with or without SR-IOV Capability



**NVMe with SR-IOV. VM talk to VF directly. PF is sit on PCIe switch.**



**NVMe without SR-IOV. Passthrough model**



**NVMe without SR-IOV. Hypervisor manage VMs to NVMe SSD**



**CONFIDENTIAL**

# Performance Results of NVMe with SR-IOV

NVMe Virtual Functions on Linux KVM (Passthrough)

## Read Performance

Tested Functions	4K Read (Random)		
	MB/s	IOPS	Latency (us)
VM_1 access to VF_1	280	68.5k	202
VM_2 access to VF_2	280	68.4k	202
VM_3 access to VF_3	277	67.5k	206
VM_4 access to VF_4	283	69.0k	200
VM_5 access to VF_5	281	68.5k	202
VM_1 access to VF_1	310	75.6k	180

The performance measured using `Fio` in CentOS 7.5, with queue depth 16 by 1 worker.

VM_1 access to PF	90	22k	647
-------------------	----	-----	-----



**CONFIDENTIAL**

# Performance Results of NVMe without SR-IOV

NVMe Physical Function on Linux KVM (Shareable)

## Read Performance

Tested Functions	4K Read (Random)		
	MB/s	IOPS	Latency (us)
VM_1 access to PF	90	22k	647
VM_2 access to PF	97	23.8k	596
VM_3 access to PF	92	22.5k	643
VM_4 access to PF	95	23k	616
VM_5 access to PF	90	22k	649
VM_1 access to PF	119	29k	488

The performance measured using `Fio` in CentOS 7.5, with queue depth 16 by 1 worker.



**CONFIDENTIAL**

# Key Benefits of NVMe SR-IOV

- Performance
  - The VF latency is only  $\frac{1}{3}$  of PF latency in multi-VMs environment
  - The performance is 3 times of PF performance
  - PCIe Gen3 x4 performance can be shared by multi-VMs
- Cost saving
  - Tens of VFs associated with a single PF, extending the capacity of a device and lowering hardware cost
  - With better latency and performance, the utilization rate will be higher to further reduce the hardware cost
  - Reduce NVMe SSD amount by sharing NVMe via PCIe

# Key Benefits of NVMe SR-IOV

- Multi-path IO via PCIe
  - The name space on NVMe can be accessed by different hosts through PCIe connection
- Flexibility configuration
  - Dynamic control by the PF through registers designed to turn on the SR-IOV capability, eliminating the need via direct access to hardware from the virtual machine environment.
- Inter-operatability
  - A standard way of sharing the capacity of any given I/O device thus allowing for the most efficient use of that resource in a virtual system



**CONFIDENTIAL**

---

# Brian Pan | H3

GM

 huaiyangpan

 www.h3platform.com

 brian.pan@h3platform.com

 +886 2 2698 3800#110



**CONFIDENTIAL**