

# Considerations for Benchmarking Network Performance in Containerized Infrastructure

**draft-dcn-bmwg-containerized-infra-11**

Minh-Ngoc Tran (Soongsil University), Sridhar Rao (The Linux Foundation),  
Jangwon Lee, Younghan Kim (Soongsil University)

# A Tribute to Al Morton

- We would like to express our heartfelt tribute to our beloved BMWG chair – Al Morton.
- We all remember his great leadership and contribution to BMWG and IETF. As for our draft, to get to the current state, he kindly welcomed us from the beginning and provided us many valuable reviews.
- We all thank and miss him dearly.

# Introduction

- This draft aims to provide additional considerations as specifications to guide containerized infrastructure benchmarking, compared with previous benchmarking methodology of common NFV infrastructure

- The consideration include:

- Investigation of **different container networking models** based on the usage of different packet acceleration techniques

4.1. Networking Models . . . . .	5
4.1.1. Kernel-space non-Acceleration Model . . . . .	6
4.1.2. User-space Acceleration Model . . . . .	7
4.1.3. eBPF Acceleration Model . . . . .	8
4.1.4. Smart-NIC Acceleration Model . . . . .	13
4.1.5. Model Combination . . . . .	14

- Investigation of **different resources configuration settings** (NUMA, hugepages, etc.) that might make performance impacts on network performance

4.2. Resources Configuration . . . . .	15
4.2.1. CPU Isolation / NUMA Affinity . . . . .	15
4.2.2. Pod Hugepages . . . . .	16
4.2.3. Pod CPU Cores and Memory Allocation . . . . .	16
4.2.4. Service Function Chaining . . . . .	17

# Draft Development

- **v00: March, 2019 - IETF 104**
  - Initial Proposal
- **v01: July, 2019**
  - First version after comments from IETF 104
- **v02 - v09: 2019 - 2023**
  - Self Update based on benchmarking tests from several IETF Hackathons
- **v10: March, 2023 – IETF 116**
  - Update based on reviews from ViNePERF Anuket Project (Sridhar – Linux Foundation and Al Morton)
  - Sridhar joined as co-author
  - Agreed on 4 Containerized Network Models types and 4 Resources Configuration considerations
  - First WG adoption call
- **v11: July, 2023 – IETF 117**
  - Update based on reviews from IETF BMWG 116 (Gábor and Vratko)
  - About Container Network Overview description, Resources Configuration consideration and Benchmarking Appendixes

# Updates Summary (from v10 to v11)

- Thanks to reviews from Gabor and Vratko, we updated
  - General containerized infrastructure description to be consistent with the draft contents
  - Additional Information in Resources Configuration consideration
  - Benchmarking proof-of-concept appendixes align with the proposed consideration in the draft

<b>version 10</b>	
1. Introduction . . . . .	3
2. Terminology . . . . .	4
3. Containerized Infrastructure Overview . . . . .	4
4. Benchmarking Considerations . . . . .	5
4.1. Networking Models . . . . .	5
4.2. Resources Configuration . . . . .	15
5. Security Considerations . . . . .	17
6. References . . . . .	18
6.1. Informative References . . . . .	18
Appendix A. Benchmarking Experience(Contiv-VPP) . . . . .	20
A.1. Benchmarking Environment . . . . .	20
A.2. Trouble shooting and Result . . . . .	24
Appendix B. Benchmarking Experience(SR-IOV with DPDK) . . . . .	25
B.1. Benchmarking Environment . . . . .	26
B.2. Trouble shooting and Results . . . . .	29
Appendix C. Benchmarking Experience(Multi-pod Test) . . . . .	29
C.1. Benchmarking Overview . . . . .	29
C.2. Hardware Configurations . . . . .	30
C.3. NUMA Allocation Scenario . . . . .	32
C.4. Traffic Generator Configurations . . . . .	32
C.5. Benchmark Results and Trouble-shootings . . . . .	32



<b>CURRENT - version 11</b>	
1. Introduction . . . . .	3
2. Terminology . . . . .	4
3. Containerized Infrastructure Overview . . . . .	4
4. Benchmarking Considerations . . . . .	5
4.1. Networking Models . . . . .	5
4.2. Resources Configuration . . . . .	15
5. Security Considerations . . . . .	18
6. References . . . . .	18
6.1. Informative References . . . . .	18
Appendix A. Benchmarking Experience (Networking Models) . . . . .	21
A.1. Benchmarking Environment . . . . .	21
A.2. Benchmarking Results . . . . .	24
Appendix B. Benchmarking Experience (Resources Configuration in Single Pod Scenario) . . . . .	25
B.1. Benchmarking Environment . . . . .	25
B.2. Benchmarking Results . . . . .	27
Appendix C. Benchmarking Experience (Networking Model Combination and Resources Configuration in Multi-Pod Scenario) . . . . .	28
C.1. Benchmarking Environment . . . . .	28
C.2. Benchmarking Results . . . . .	30

# Detailed Updates (1)

## Introduction and Overview inconsistency with remain draft contents

### Vratko's review: veth is not a general concept

- Use “container network plugin” as the general container networking mechanism
  - In terms of networking, to route traffic between containers which are isolated in different network namespaces, [virtual ethernet \(vETH\) interface pairs are used](#) to create ...
  - In terms of networking, to route traffic between containers which are isolated in different network namespaces, [a container network plugin is required](#). This network plugin creates ...

### Vratko's review: CNI is specific to Kubernetes, there are other container orchestration services.

- Kubernetes is the main and most popular orchestration platform nowadays, so we can use Kubernetes' Container Network Interface (CNI) for the draft. All of networking models in the draft require Kubernetes CNIs:
  - Kernel-Space non-Acceleration: normal Kubernetes CNI (i.e. flannel)
  - User-Space Acceleration: Userspace K8s CNI
  - eBPF Acceleration: Cilium/AFXDP/Userspace K8s CNI
  - Smart-NIC Acceleration: SR-IOV K8s CNI

### Vratko's review: List of networking model might inevitably be incomplete

- The Networking Model consideration list in the draft does not list out container networking techniques. It is a list of all possible categories
- Any additional technique can fall into one of the considered categories in the draft

# Detailed Updates (2)

## Resources Configuration Additions

**Vratko's review: Some resources can also be applied to VM-VNF**

- Update in Draft: Mentions about NUMA and CPU Isolation can also be applied for VM, others are specific to pod

**Vratko's review: Most consideration also applies for other SUT components, not only NFV DUTs**

- Added in Draft

**Vratko's review: Nosiy neighbor is not the only use-case of Resource Isolation practices,**

- Added in Draft

**Vratko's review: Recommend varying non-DUT resources in Resource Isolation benchmarking**

- Under test and will be consider to add

**Gábor's review: Hugepage size value of 2MB and 1GB**

- Updated to current standard.

# Detailed Updates (3)

## Benchmarking Appendixes

**Vratko suggestion about input/output/result reporting, and putting Benchmarking result as a main draft section**

- We provided Benchmarking results in our draft as a kind of “proof-of-concept” for verifying the benchmarking considerations proposed in our draft.
- Containerized Networking Benchmarking Reporting standard is out of scope of our draft

**Gábor’s review: Packet Frame size values are not provided**

- Updated the benchmarking results with frame size values (as in example figures here)

NUMA Alignment Scenarios						
	s1	s2	s3	s4	s5	s6
Throughput	39.31	23.67	29.23	37.25	23.58	29.36

Figure 17: Different resource configurations 1518-byte packet size's zero packet loss throughput test result in single pod scenario (Gbps)

Frame Size (bytes)	Model	
	Userspace (VPP)	Combined (SRIOV-VPP)
64	7.23	9.62
128	13.38	15.71
256	19.23	23.91
512	25.58	31.76
1024	30.07	39.15
1280	31.16	39.33
1518	31.25	39.32

Figure 16: Networking Model Combination Zero Packet Loss Throughput Test Results (Gbps)

# Detailed Updates (3)

## Benchmarking Appendixes

Update our latest Benchmarking Results and re-organize to align with proposed consideration in draft

4. Benchmarking Considerations . . . . .	5
4.1. Networking Models . . . . .	5
4.1.1. Kernel-space non-Acceleration Model . . . . .	6
4.1.2. User-space Acceleration Model . . . . .	7
4.1.3. eBPF Acceleration Model . . . . .	8
4.1.4. Smart-NIC Acceleration Model . . . . .	13
4.1.5. Model Combination . . . . .	14
4.2. Resources Configuration . . . . .	15
4.2.1. CPU Isolation / NUMA Affinity . . . . .	15
4.2.2. Pod Hugepages . . . . .	16
4.2.3. Pod CPU Cores and Memory Allocation . . . . .	16
4.2.4. Service Function Chaining . . . . .	17



Appendix A. Benchmarking Experience (Networking Models) . . . . .	21
A.1. Benchmarking Environment . . . . .	21
A.2. Benchmarking Results . . . . .	24
Appendix B. Benchmarking Experience (Resources Configuration in Single Pod Scenario) . . . . .	25
B.1. Benchmarking Environment . . . . .	25
B.2. Benchmarking Results . . . . .	27
Appendix C. Benchmarking Experience (Networking Model Combination and Resources Configuration in Multi-Pod Scenario) . . . . .	28
C.1. Benchmarking Environment . . . . .	28
C.2. Benchmarking Results . . . . .	30

# Conclusion

- With current update after several reviews, we think our draft is quite stable
- We would like ask adoption of this draft as a working group draft
- Feedbacks and comments are always welcome

Backup Slides

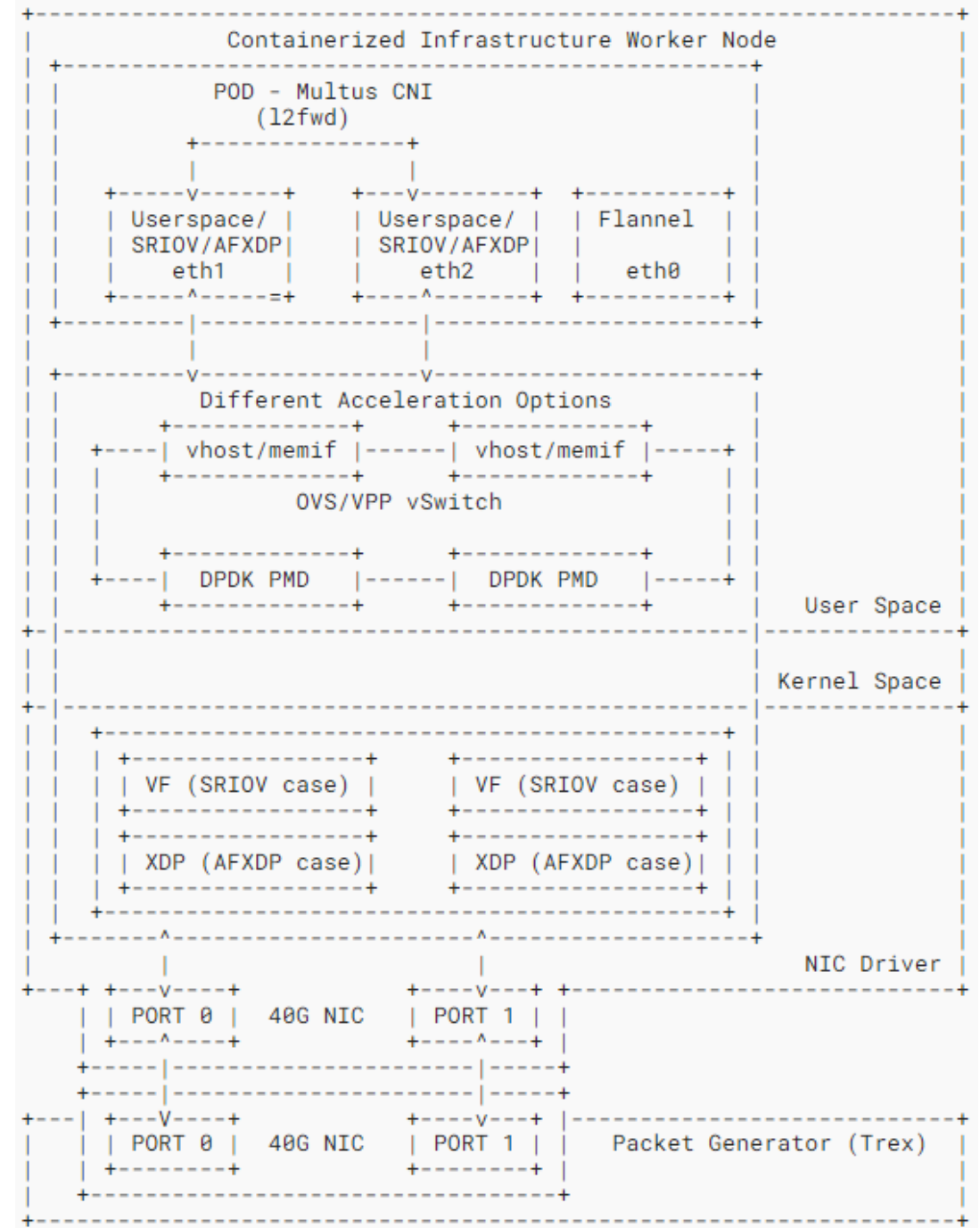
# Benchmarking Environment

Node Name	Specification	Description
Master Node	<ul style="list-style-type: none"><li>- Intel(R) Xeon(R) Gold 5220R @ 2.4Ghz (10 Cores)</li><li>- MEM 128GB</li><li>- DISK 500GB</li><li>- Control plane : 1G</li></ul>	<ul style="list-style-type: none"><li>Container Deployment and Network Allocation</li><li>- Centos 7.7</li><li>- Kubernetes Master</li><li>- MULTUS CNI</li><li>Userspace CNI</li><li>Kubernetes SRIOV plugin</li><li>Kubernetes AFXDP plugin</li></ul>
Worker Node	<ul style="list-style-type: none"><li>- Intel(R) Xeon(R) Gold 5220R @ 2.4Ghz (80 Cores)</li><li>- MEM 256G</li><li>- DISK 2T</li><li>- Control plane : 1G</li><li>- Data plane : XL710-qda2 (1NIC 2PORT- 40Gb)</li></ul>	<ul style="list-style-type: none"><li>Container Service</li><li>- Ubuntu 22.04 (18.04 fpr VPP test)</li><li>- Kubernetes Worker</li><li>- Layer 2 Forwarding DPDK application</li><li>- MULTUS CNI</li><li>Userspace CNI</li><li>Kubernetes SRIOV plugin</li><li>Kubernetes AFXDP plugin</li></ul>
Packet Generation Node	<ul style="list-style-type: none"><li>- Intel(R) Xeon(R) Gold 6148 @ 2.4Ghz (2Socket X 20Core)</li><li>- MEM 128G</li><li>- DISK 2T</li><li>- Control plane : 1G</li><li>- Data plane : XL710-qda2 (1NIC 2PORT- 40Gb)</li></ul>	<ul style="list-style-type: none"><li>Packet Generator</li><li>- CentOS 7.7</li><li>- installed Trex 2.4</li><li>Benchmarking Application</li><li>- T-Rex Non Drop Rate</li></ul>

# BM Networking Models

Frame Size	Model			
	Userspace (VPP)	eBPF (OVS-AFXDP)	eBPF (AFXDP CNI)	Smart-NIC (SR-IOV)
64	7.25	1.64	4.32	10.48
128	13.32	2.69	8.32	25.37
256	19.26	3.54	14.47	30.38
512	25.62	7.32	27.13	37.11
1024	30.12	13.42	37.16	39.10
1280	31.23	17.83	39.23	39.23
1518	31.26	21.37	39.25	39.28

Figure 10: Different Networking Models Zero Packet Loss Throughput Test Results (Gbps)



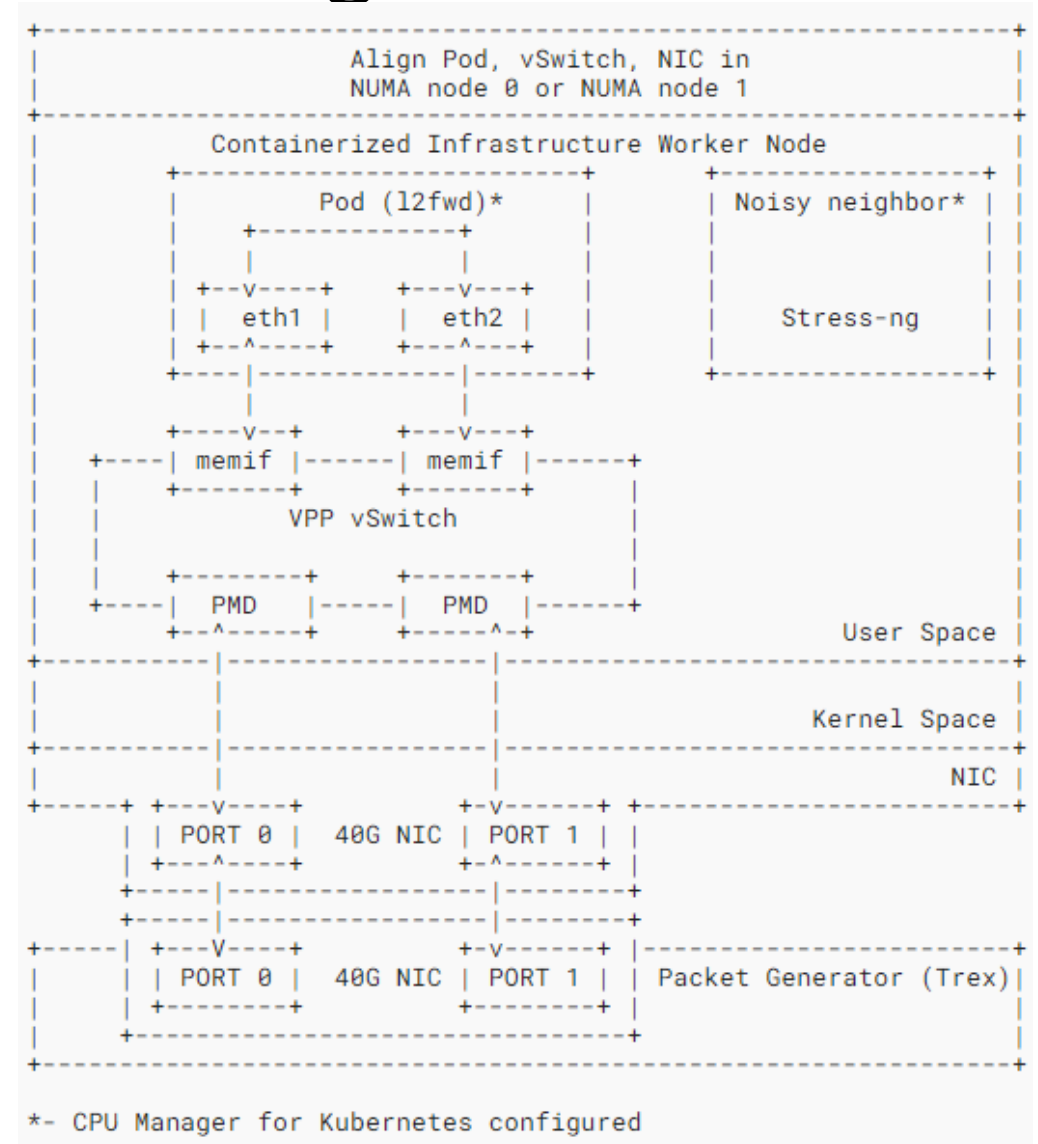
# BM Single-Pod – Resources Configuration

## o NUMA Alignment Scenarios

Scenario	NIC	vSwitch	pod
s1	NUMA0	NUMA0	NUMA0
s2	NUMA0	NUMA0	NUMA1
s3	NUMA0	NUMA1	NUMA1
s4	NUMA0	NUMA1	NUMA0

CPU Pinning	NUMA Alignment Scenarios			
Scenarios	s1	s2	s3	s4
Without CMK	4.78	2.34	4.39	2.41
CMK-Exclusive Mode	15.63	7.67	14.33	7.84
CMK-shared Mode	11.16	5.47	10.23	5.52

Figure 13: Different resource configurations 1518-byte packet size's zero packet loss throughput test result in single pod scenario (Gbps)



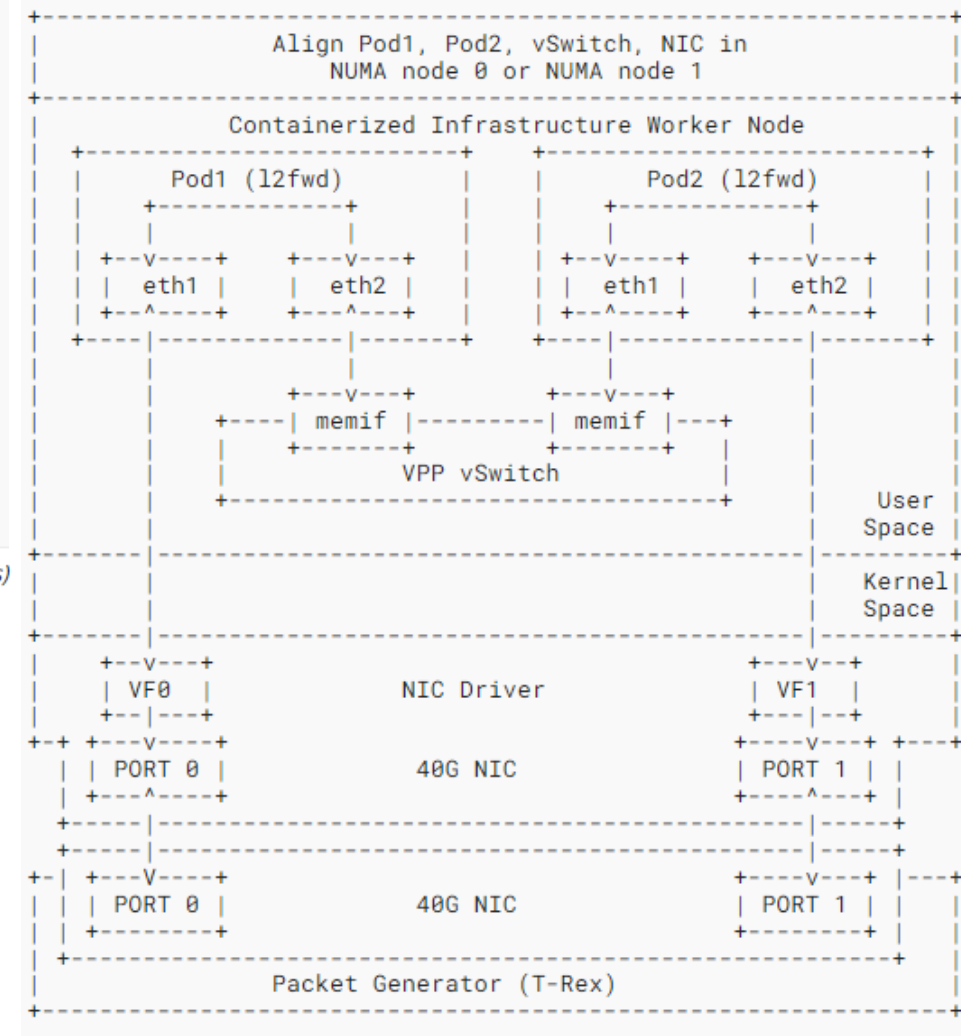
# BM Multi-Pod – Resources Configuration

Scenario	NIC	vSwitch	pod1	pod2
s1	NUMA0	NUMA0	NUMA0	NUMA0
s2	NUMA0	NUMA0	NUMA0	NUMA1
s3	NUMA0	NUMA0	NUMA1	NUMA0
s4	NUMA0	NUMA1	NUMA1	NUMA1
s4	NUMA0	NUMA1	NUMA1	NUMA0
s4	NUMA0	NUMA1	NUMA0	NUMA1

Figure 15: NUMA Alignment Scenarios in Multi-Pods scenario

Frame Size (bytes)	Model	
	Userspace (VPP)	Combined (SRIOV-VPP)
64	7.23	9.62
128	13.38	15.71
256	19.23	23.91
512	25.58	31.76
1024	30.07	39.15
1280	31.16	39.33
1518	31.25	39.32

Figure 16: Networking Model Combination Zero Packet Loss Throughput Test Results (Gbps)



NUMA Alignment Scenarios						
	s1	s2	s3	s4	s5	s6
Throughput	39.31	23.67	29.23	37.25	23.58	29.36

Figure 17: Different resource configurations 1518-byte packet size's zero packet loss throughput test result in single pod scenario (Gbps)