Empowering container-based NFVi with VPP on Arm architecture

Trevor Tao Trevor.Tao@arm.com

Song Zhu Song.Zhu@arm.com

25/06/2018

Agenda

- Background
- FD.io/VPP Enablement on Arm Platform
- FD.io/VPP integrated with Container Networking Solution
  - Vhost-user CNI
  - Contiv/VPP netplugin
- Future Plan and Use Cases

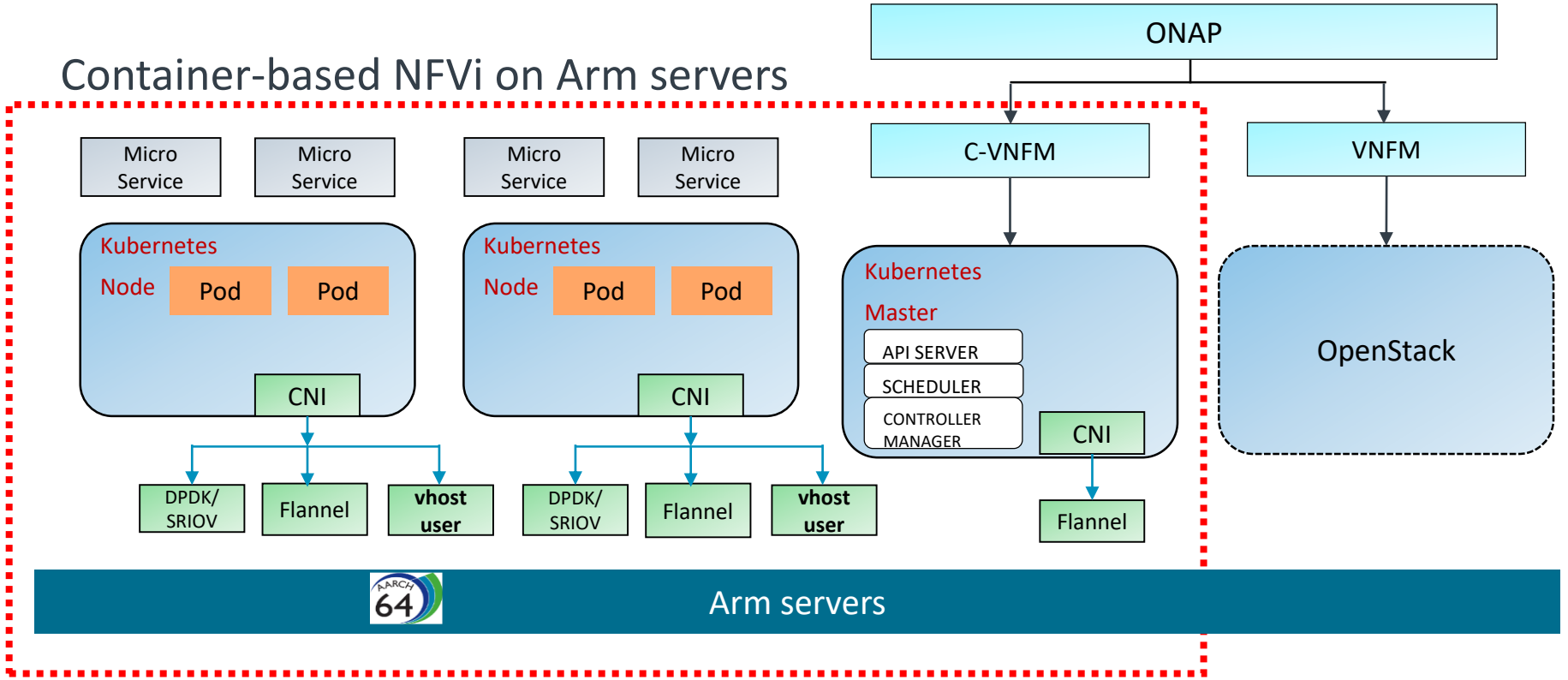
Background

arm

Background

- Trends: Container-based platforms for OPNFV
- Containerized OpenStack or Kubernetes as VIM
- OPNFV Euphrates release delivered Kubernetes integration
- OPNFV projects: **Container4NFV**, Auto, Clover…
- Containerized VNFs with Data Plane Acceleration (SRIOV)
- **Acceleration for inter-container communication with VPP**

Container-based NFV Architecture



- Kubernetes as VIM
- Flannel/SRIOV/vhost user CNI plugins integrated
- SRIOV CNI: enable VF passthrough
- Vhost-user CNI: enable VPP-based container networking

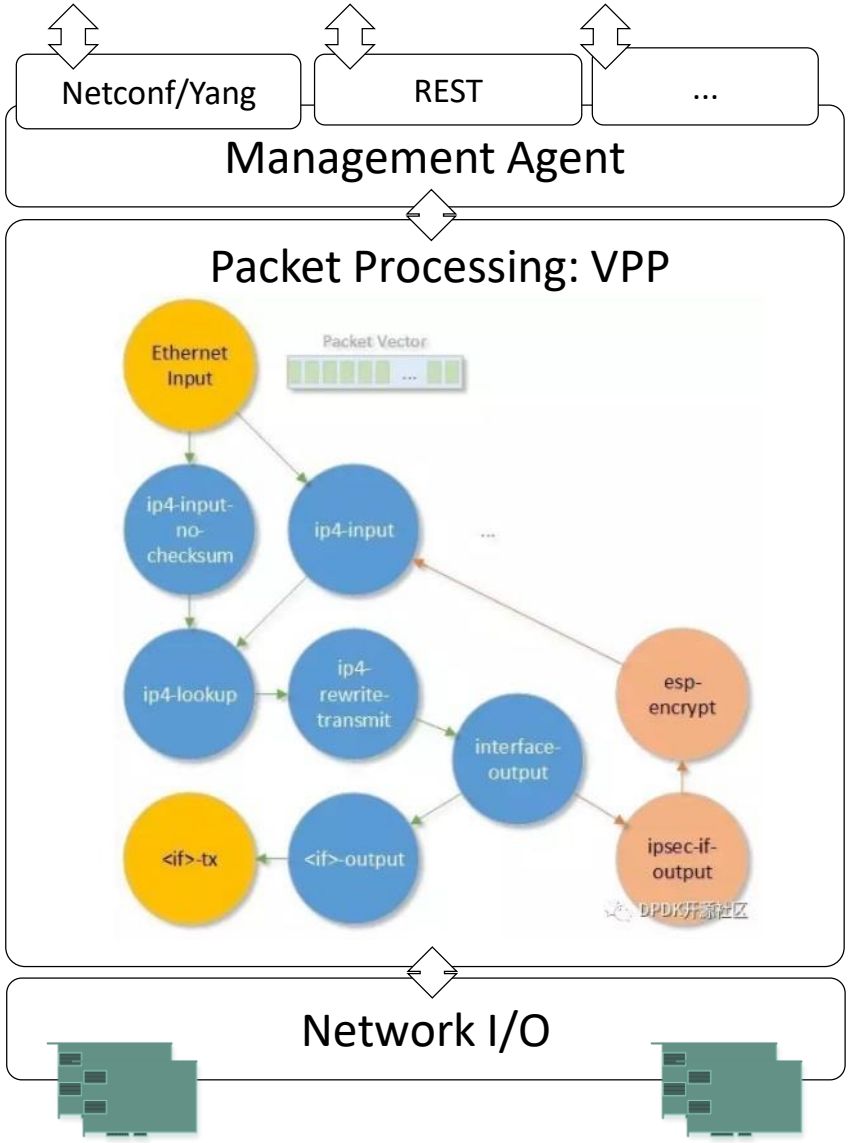
Ref: Container4NFV Architecture

FD.io/VPP Enablement on Arm Platform

arm

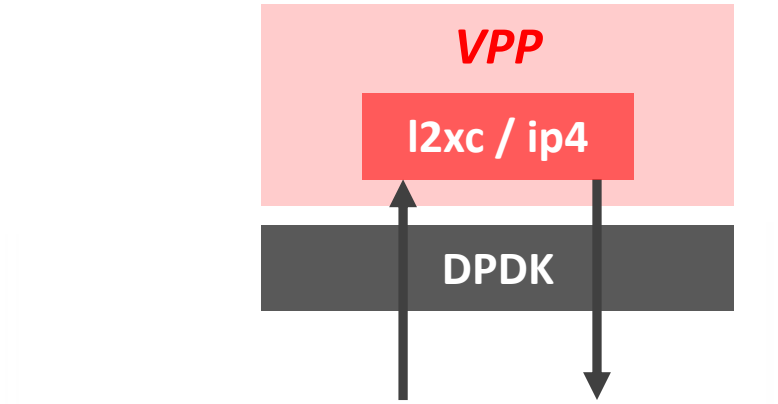
FD.io/VPP (Vector Packet Processing)

- User Space software platform providing switch/router functionalities
- Aiming to run on commodity CPUs
- Cisco developed it from 2002 and open sourced it in FD.io (Linux Foundation) on Feb 2016
- Leverage DPDK, XDP, netmap… as fast I/O
- Batch packet processing - more efficient iCache utilization
- Packet processing graph: modular, flexible, and extensible
- **Fast**, **scalable** and **deterministic**
  - 14+ Mpps per core, tested to 1TB
  - Scalable FIB: supporting millions of entries
  - 0 packet drops, ~15µs latency



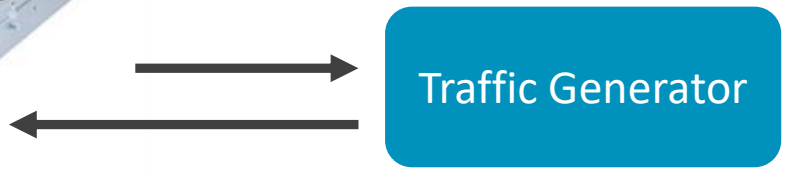
arm

A Simple Use Case for Performance Tuning



- **L2xc** – forward all packets received on Port0 to Port1 and vice versa
- **IPv4** – route packets across IPv4 subnets

Arm serves with Cortex A72 Processors



arm

Performance Benchmarking and Tuning

64B packet – single flow – single core

Observations

- Most hotspots are memory accesses
- Software-defined data placement consumes processing cycles
- Unintentionally ordering memory accesses can slow the system down
- Compiler may fuse loops which alters memory access pattern from original program order

Further Directions

- RFC2544 testing
- Multicore scaling
- PMU data
- Cache stashing
- Compiler and C library versions
- Other platforms

The path to .io on Arm

Workload Scale	Performance Analysis	Software
CSIT	Hotspot & Bottleneck Identification	Upstream
		Libraries
		OS
		Toolchain
FD.io Lab	Tuning & Optimization	**Hardware**
		Processors
		I/O
		Accelerators

CAVIUM HUAWEI MARVELL

arm

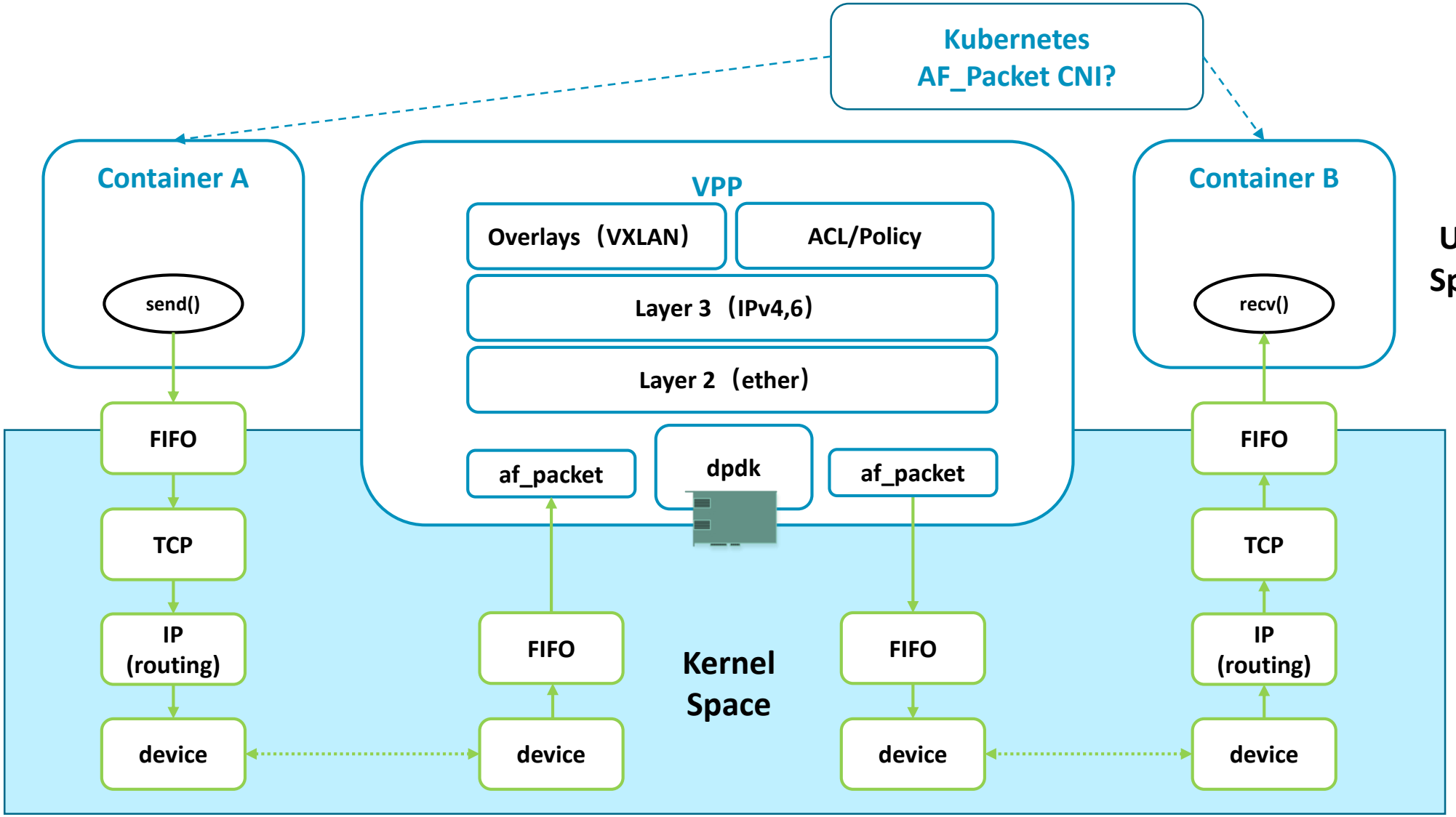
FD.io/VPP as Container Networking Solution

arm

Why Use VPP for Container Networking

- Container networking requirements for NFV
  - High performance on packet processing
  - High scalability
  - High flexibility
- What VPP provides
  - High performance
  - Abundant interfaces: ssvm, virtio/vhost, af_packet, tap, memif…
  - Abundant features for control and management

VPP for Container Networking with AF_Packet interface



Pros:

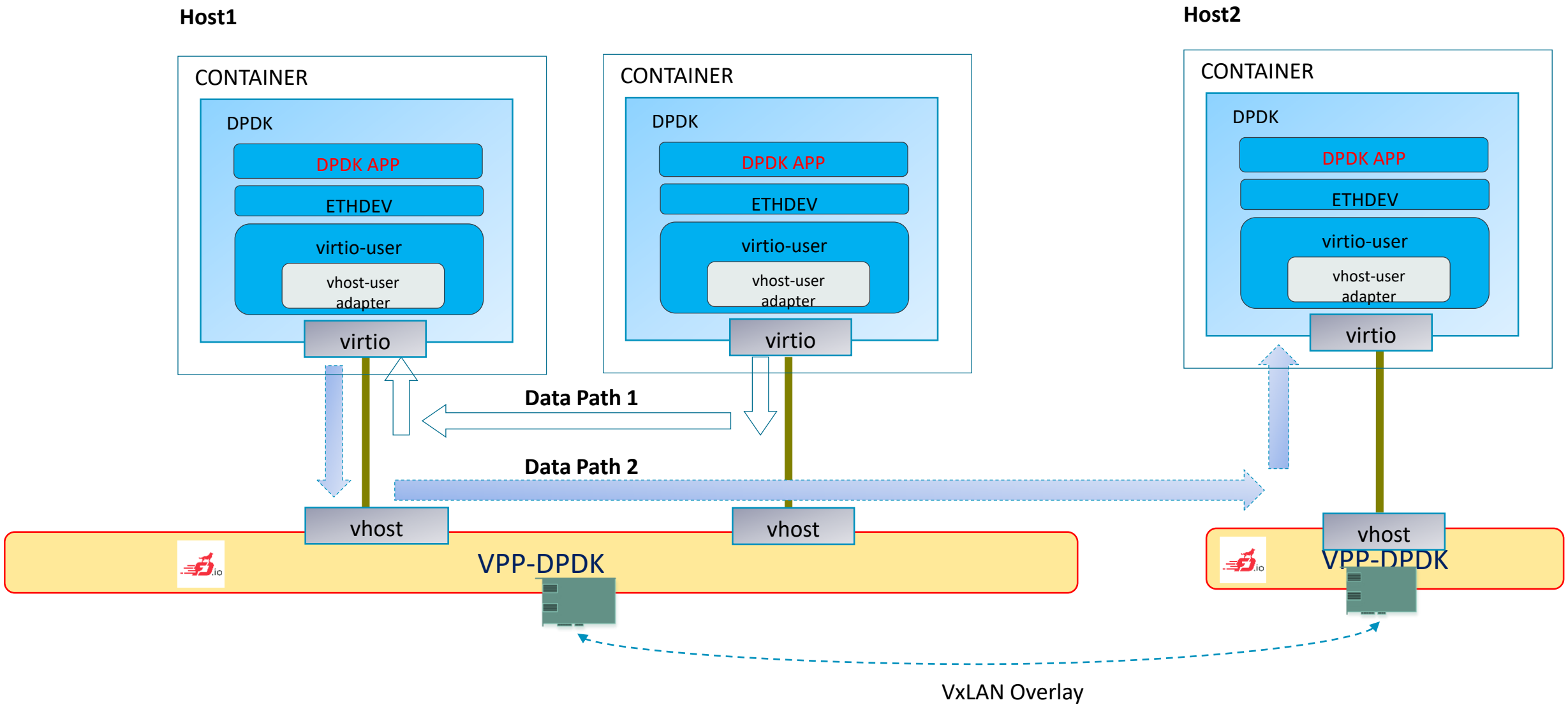
Support Linux kernel stack which is required by most applications with performance higher than Flannel

Cons:

Performance is lower than vhost-user/memif interface

arm

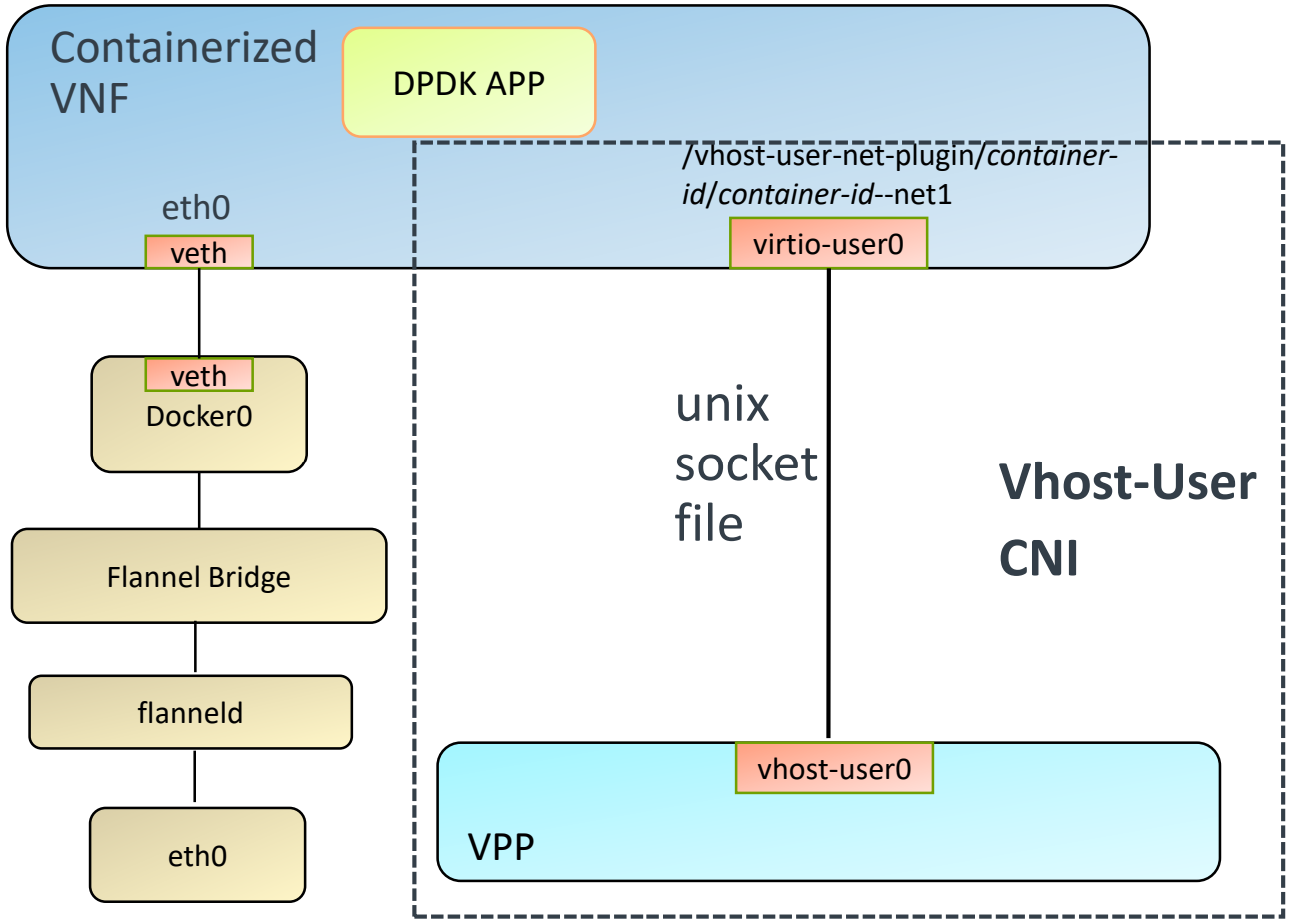
VPP for Container Networking with Virtio-Vhost Interface



arm

Vhost-user CNI for Kubernetes

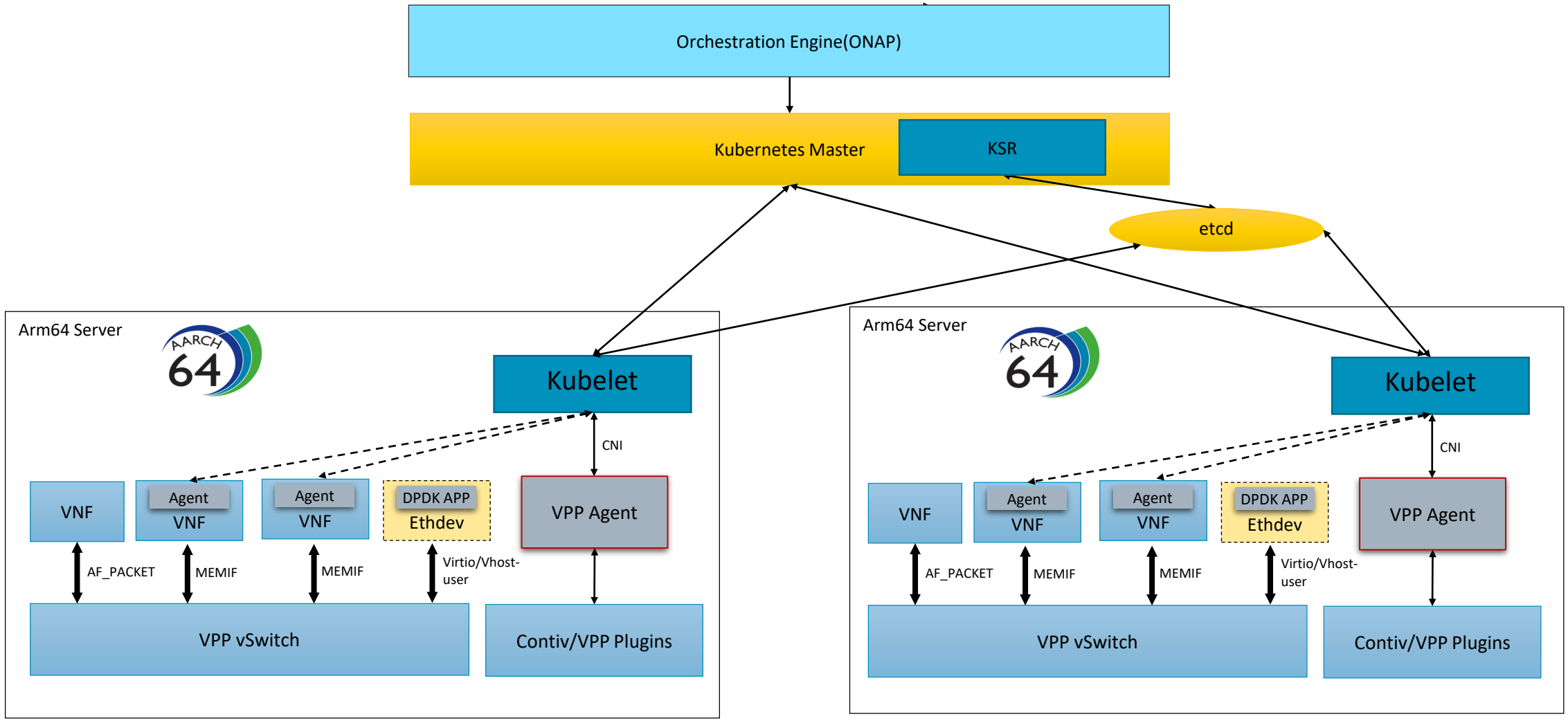
K8S POD



- Vhost-user server socket(interface) is created in VPP
- After adding the vhost user CNI path, the virtio-user interface is used as a virtual device of DPDK

arm

Contiv/VPP Integration on Arm Platform



(Ref: VPP, VNF Agent and Contiv integration)

arm

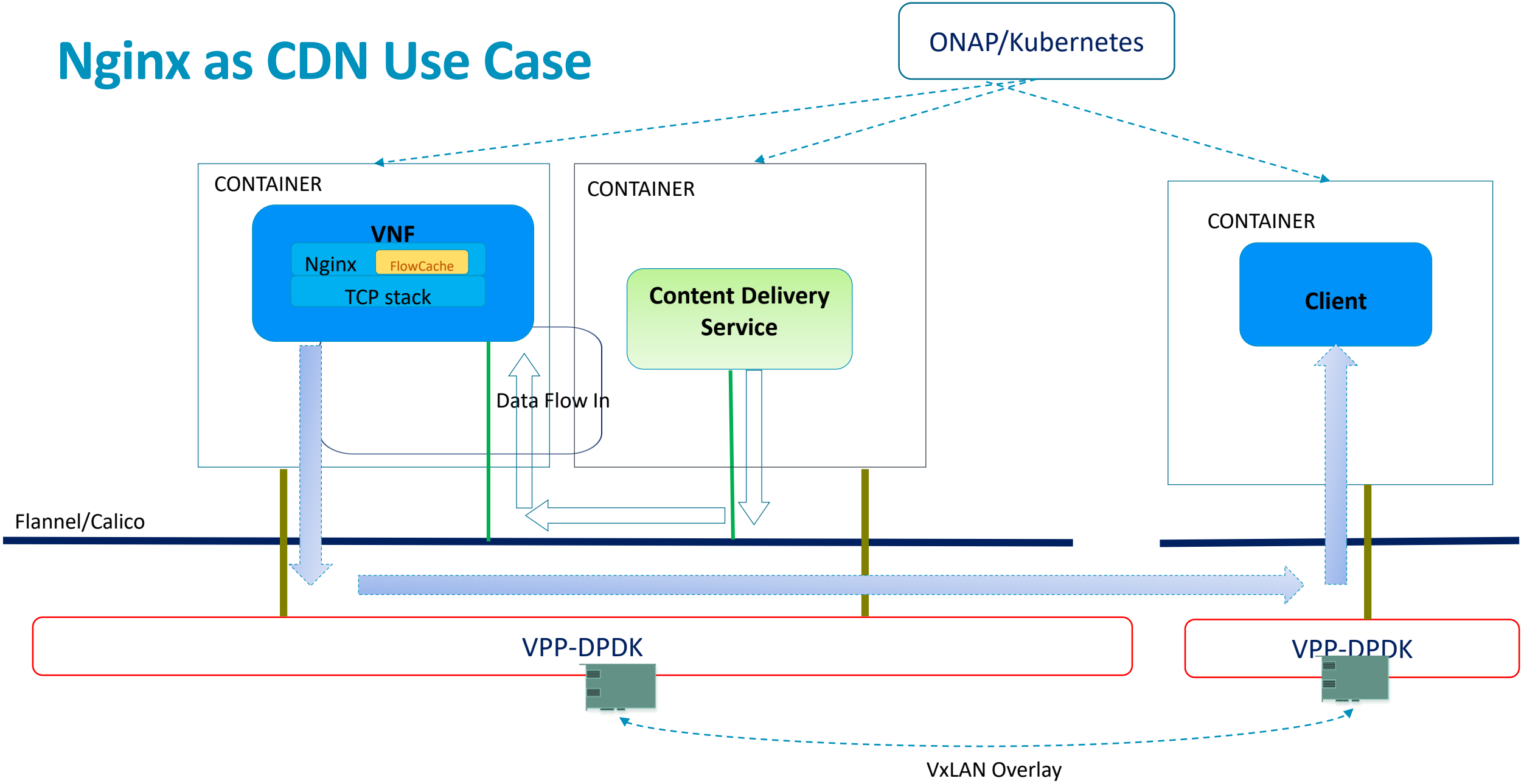
What We Have Done

- Enabled VPP release on Arm64 servers
  - VPP 17.10 running on Arm servers
  - VPP 18.04 released with AArch64 packaging for Ubuntu
- Integrated VPP with Kubernetes for inter-container communication with virtio/vhost-user interfaces on Arm servers
- Enhanced vhost-user CNI for Kubernetes with VPP
- Enabling project Ligato and Contiv/VPP on Arm platforms
- Enabling VPP-based use cases for OPNFV Container4NFV project

Use Cases and Future Plan

arm

Nginx as CDN Use Case



arm

Next Steps

- Continue performance tuning on Arm servers
- Performance benchmarking with NFVbench/VSperf on Arm servers
- VPP integration (CI/CD enablement) in OPNFV Gambia release (Nov 2018)
- Enable and integrate other VPP based CNI solutions (memif, …)
- Enable more VPP-based use cases (microservices and SFC) for NFVi
- Integrate VPP-based NFV solutions with orchestration software (ONAP)

Thank You
Danke
Merci
谢谢
ありがとう
Gracias
Kiitos
감사합니다
धन्यवाद
תודה

arm