DPDK
DATA PLANE DEVELOPMENT KIT

DPDK Summit

ODL SFC with OVS-DPDK, HW accelerated dataplane and VPP

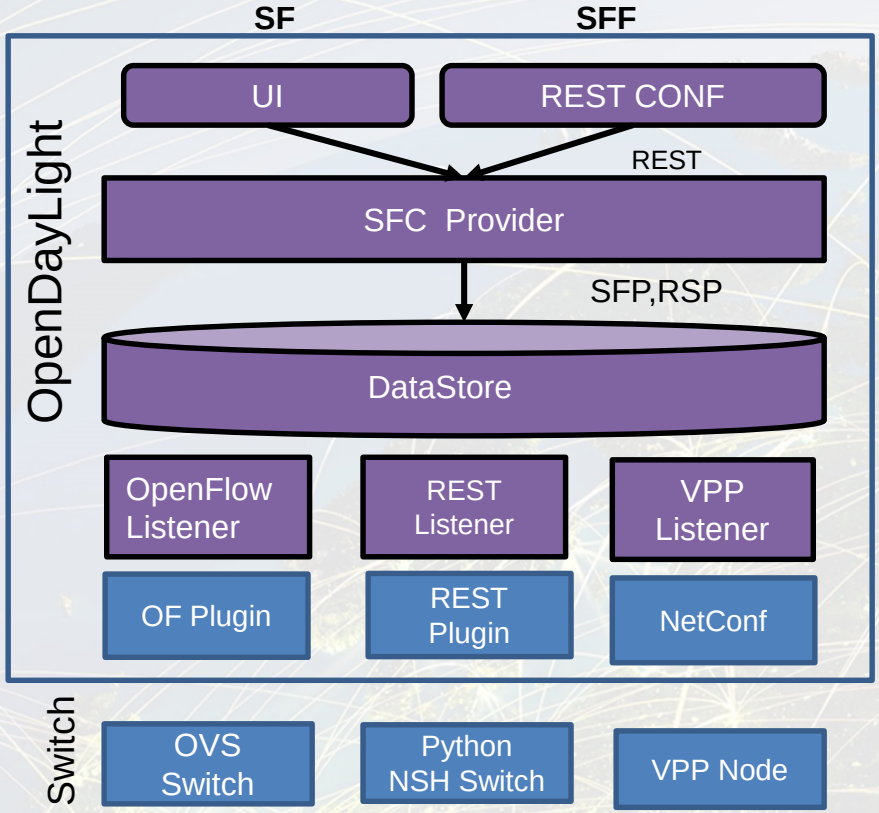
Prasad Gorja, Senior Principal Engineer, NXP
Harish Kumar Ambati, Lead Engineer, NXP
Srikanth Lingala, Lead Engineer, NXP

NXP

Agenda

- SFC Introduction
- ODL SFC with OVS-DPDK
- ODL SFC with HW accelerated dataplane and VPP as vNF

OpenDayLight SFC



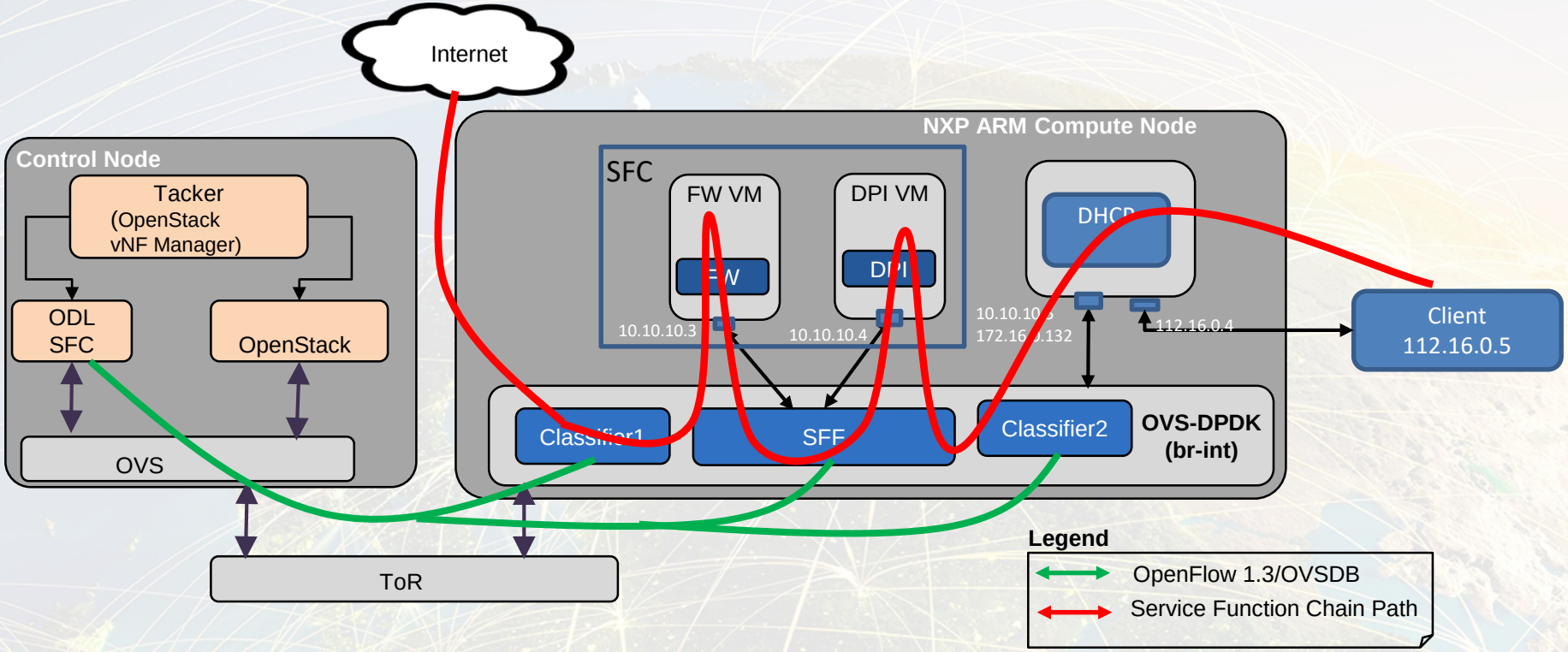
SFC

SF - Service Function
SFF- Service Function Forwarder
SFP- Service Function Path
RSP – Rendered Service Path

NSH

NSH – Network Service Header
NSP- Service Function Forwarder
NSI- Network Service Index

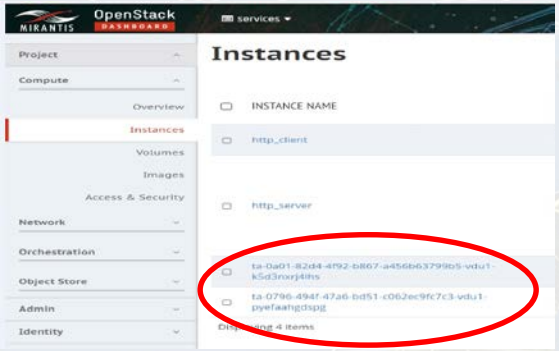
ODL SFC with OVS-DPDK



Tacker and NSH Flows

Vnf-create

1

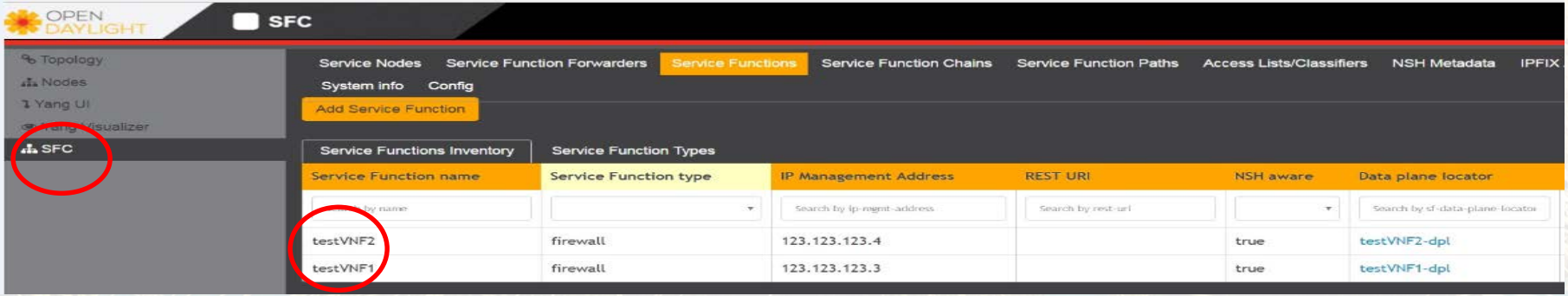


2 SFC and SFC-Classifier

- cookie=0x1110010000440255, duration=632.734s, **table=11,** n_packets=7, n_bytes=518, **tcp** actions=move:NXM_NX_TUN_ID[0..31]->NXM_NX_NSH_C2[],push_nsh,load:0x1->NXM_NX_NSH_MDTYPE[],load:0x3->NXM_NX_NSH_NP[],load:0xc0a80018->NXM_NX_NSH_C1[],load:0x2c->NXM_NX_NSP[0..23],load:0xff->NXM_NX_NSI[],load:0x7b7b7b03->NXM_NX_TUN_IPV4_DST[],load:0x2c->NXM_NX_TUN_ID[0..31],resubmit(,0)

- cookie=0x1110010000440255, duration=632.734s, table=11, n_packets=7, n_bytes=518, tcp,reg0=0x1,tp_src=2000,tp_dst=80 actions=move:NXM_NX_TUN_ID[0..31]->NXM_NX_NSH_C2[],push_nsh,load:0x1->NXM_NX_NSH_MDTYPE[],load:0x3->NXM_NX_NSH_NP[],load:0xc0a80018->NXM_NX_NSH_C1[],load:0x2c->NXM_NX_NSP[0..23],load:0xff->NXM_NX_NSI[],load:0x7b7b7b03->NXM_NX_TUN_IPV4_DST[],load:0x2c->NXM_NX_TUN_ID[0..31],resubmit(,0)

3



Need for Programmable HW Acceleration

General processor are sub-optimal for packet processing	HW Acceleration (Advanced IO Processor)
Low packet processing locality underutilizes cache and pipeline and suffers from **high DDR latency**	Specialized memory hierarchy and explicit DMA operations instead of cache allow **deterministic performance.**
Increasing single thread performance causes super-linear power increase	**Parallelism** with more “small” cores. **Higher Performance/Watt.**
Performance and complexity of software hiding latency of asynchronous access to accelerators	**Hardware scheduler** based multitasking environment hides access latency.
More core cycles are consumed for standard packet processing operations	**Hardware accelerators** for common tasks: Lookups, parse, frame operations, timers, statistics, frag and reassembly
Performance and complexity of software ordering and synchronization	Functions are provided by hardware scheduler
More cores are engaged when OVS runs in GPP	AIOP Offload offers **conservation of cores**

HW Accelerators

Classifier & Table Lookup Unit (CTLU)

- Parser & Keygen
- Parse packet based on parser profile
- Read parser results during processing
- The TLU performs table lookup off load acceleration in the AIOP
- Supports -Exact match (EM),Longest Prefix Match (LPM) and Algorithmic ACL (ACL) table types.
- Rule timestamping for activity.

Timer Manager (TMAN) engine

- Offloads the management of millions of timers.
- Manages the timers, and automatically initiates AIOP tasks when timers expire.
- Micro/Milli/Seconds and Periodic or One shot timers
- Timer priority and Timer containers

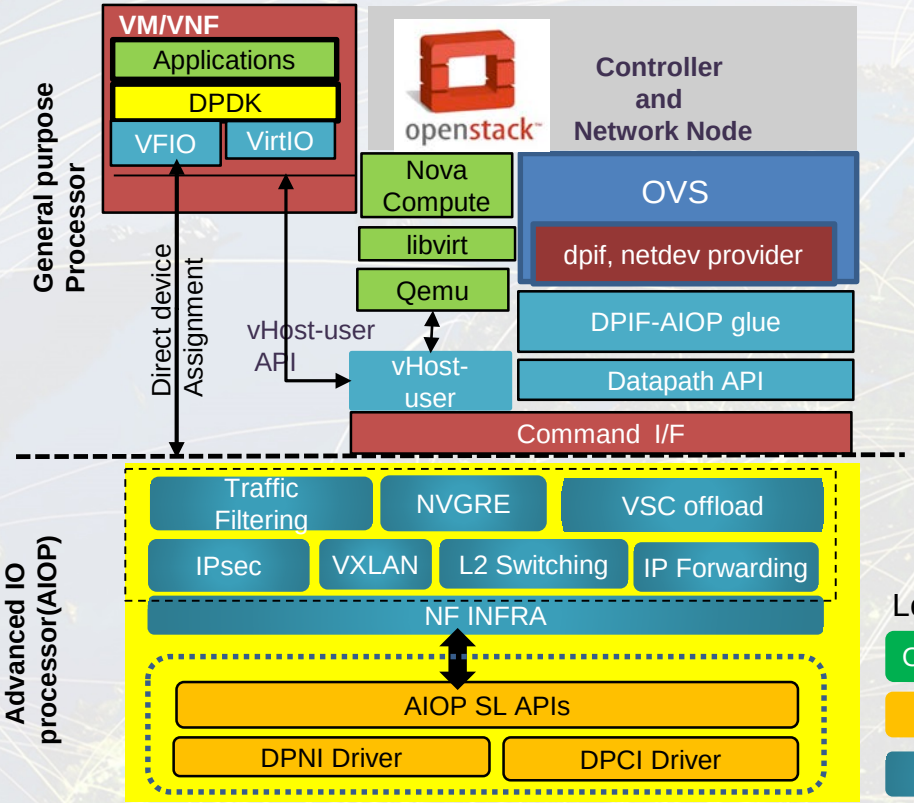
Frame DMA (FDMA)

- Autonomous initial frame presentation & subsequent presentation to AIOP WS.
- Frame modification, Frame replication, Frame concatenation, split acceleration, Frame store, discard acceleration. Enque frames to the Queue Manager (FQ/QD)
- Direct access to system memory (DMA data).

Context DMA (CDMA)

- Read context buffer contents from external memory (DDR/PEB) and present in workspace memory.
- Write modified contents of context buffer (in WS-MEM) to external memory (PEB/DDR)
- Synchronization mutex locks and RCU
- Context buffer reference counting

Hardware OVS Acceleration

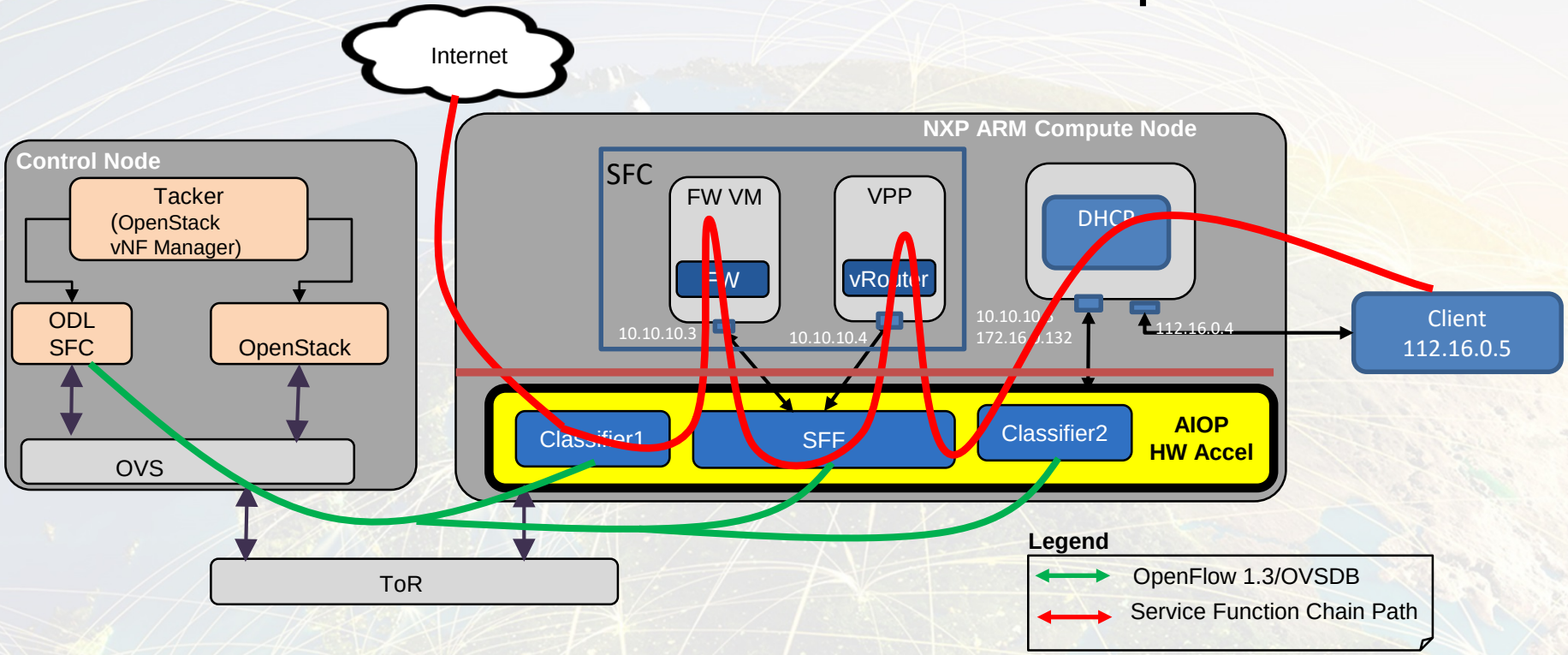


Legend

- OpenSource
- NXP SDK
- NXP ADK

- Conserve cores and increased VM density
- Integrated at well defined ovs-dpif and netdev provider
- Deterministic Performance

ODL SFC with HW accelerated dataplane



NOVA VIF driver

- Create Data Path and connect to SDN Controller in OpenStack Compute Nodes
- Flows pushed from OPENDAYLIGHT SDN Controller .
- While Nova Compute is started all available ports are published to nova in JSON Format.
- Dynamic assignment of ports from JSON format to VMs.
- Create DPRC Containers for VM and DPNI in each in each VM DPRC.
- Connect DPNI in VM DPRC to AIOP DPRC.
- Bind VM DPRC to VFIO and Building libvirt XML with specific DPNI and DPRC IDs .

THANK YOU

prasad.gorja@nxp.com
harish.ambati@nxp.com
srikanth.lingala@nxp.com