BUILDING AN NVFI PLATFORM FOR VEPC WITH CONTRAIL, DPDK, SMARTNICS AND OPENSTACK

ATHENS, JUNE 2019

LIFE IS FOR SHARING.

CONTENT

Chapter 1	The Company	[03 slides]
Chapter 2	The Cloud	[12 slides]
Chapter 3	The vEPC PoC	[06 slides]

THE COMPANY

PAN-NET: A DEUTSCHE TELEKOM COMPANY IN EUROPE

6 locations

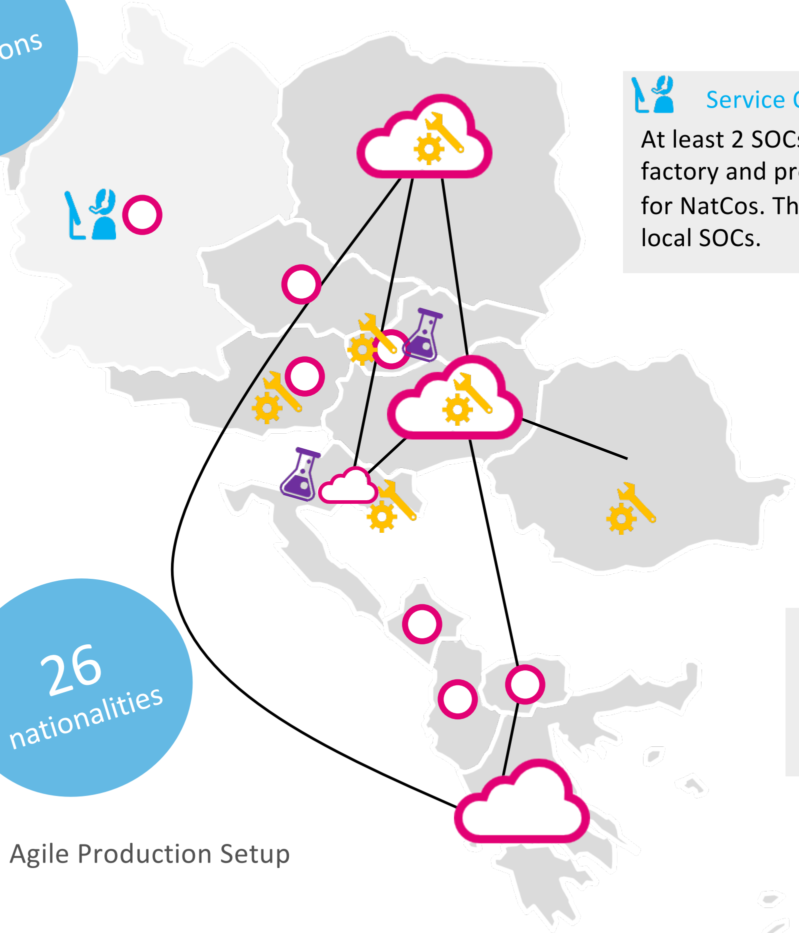
300+ employees

26 nationalities

Back-end Data Center
Three geo-redundant BEDCs provide the core of the infrastructure cloud.

Front-end Data Center
At least two redundant FEDCs in each NatCo provide the basis to connect and serve the NatCo.

International operations support system
A common operating system takes care of all central management functions and provides a common IT integration point for the NatCos.



Service Operation Center
At least 2 SOCs monitor the production factory and provide first level support for NatCos. They are connected to all local SOCs.

Backbone Network
A multi-national network connects all Pan-Net locations.

Test Lab
Testing and development environment for new components, functions and their integration.

Agile Production Setup

PAN-NET: A DEUTSCHE TELEKOM COMPANY IN EUROPE

Back-end Data Center

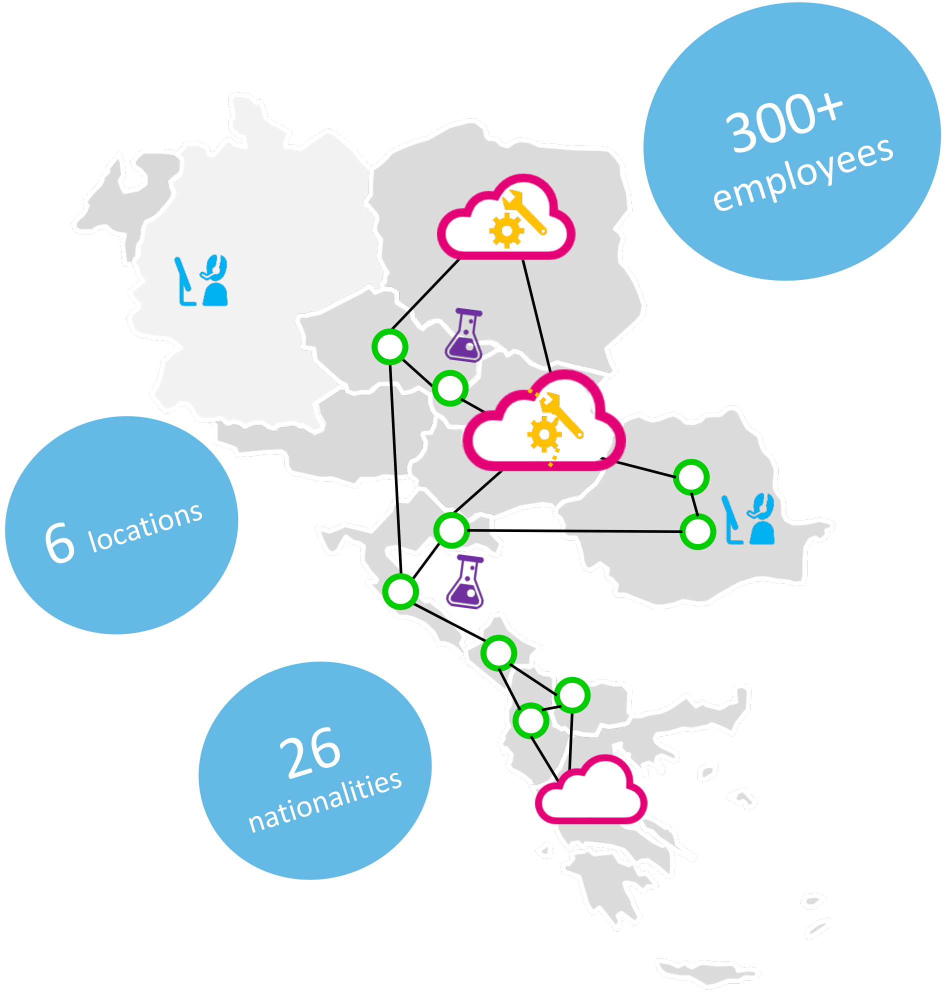
Three geo-redundant BEDCs provide the core of the infrastructure cloud.

○ Front-end Data Center

At least two redundant FEDCs in each NatCo provide the basis to connect and serve the NatCo.

International operations support system

A common IOSS takes care of all central management functions and provides a common IT integration point for the NatCos.



300+ employees

6 locations

26 nationalities

Service Operation Center

At least 2 SOCs monitor the production factory and provide first level support for NatCos. They are connected to all local SOCs.

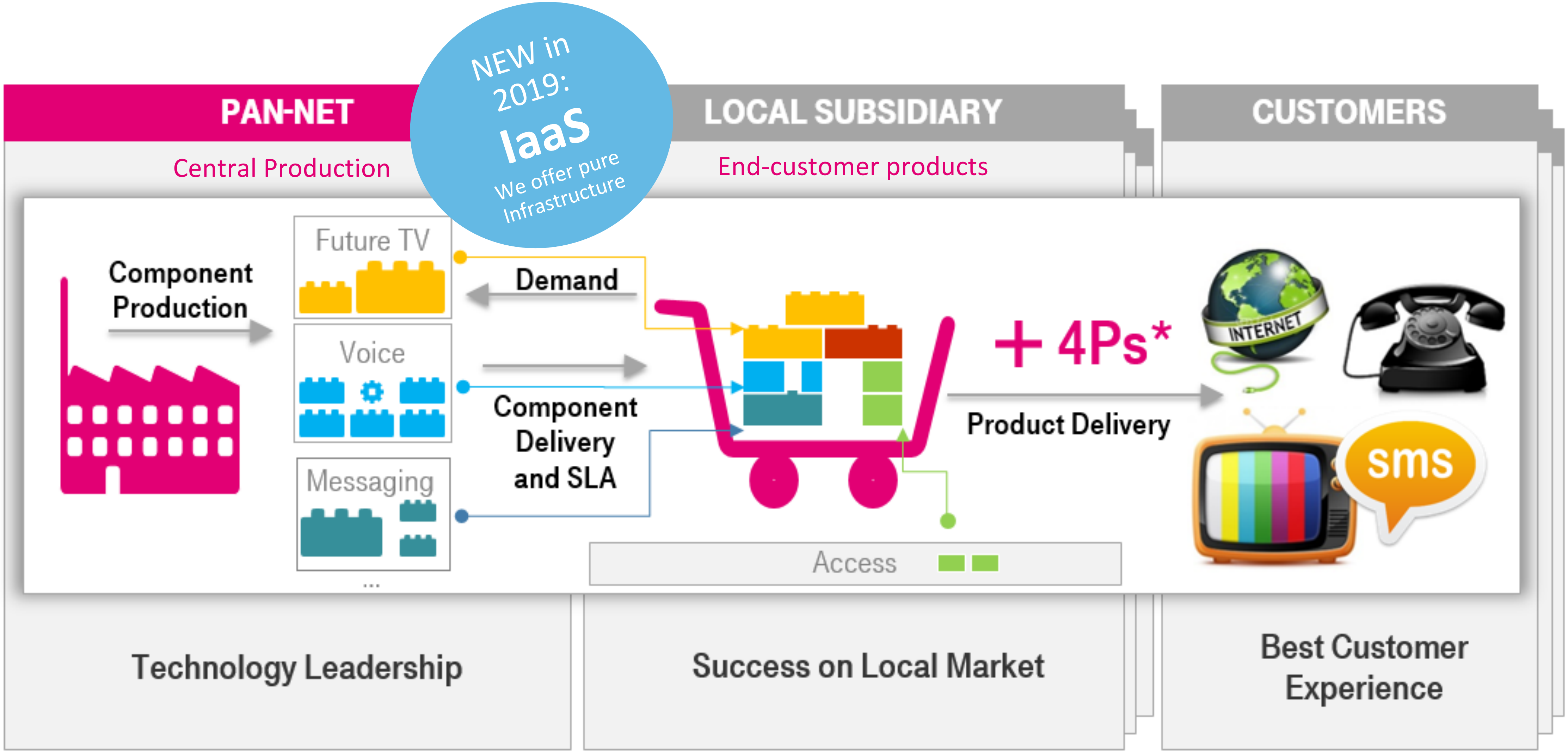
◁ Backbone Network

A multi-national network connects all Pan-Net locations.

Test Lab

Testing and development environment for new components, functions and their integration.

Pan-Net produces services, Natcos sell customer products



NEW in 2019:

IaaS

We offer pure Infrastructure

PAN-NET

Central Production

LOCAL SUBSIDIARY

End-customer products

CUSTOMERS

Component Production

Future TV

Voice

Messaging

...

Demand

Component Delivery and SLA

Access

+ 4Ps*

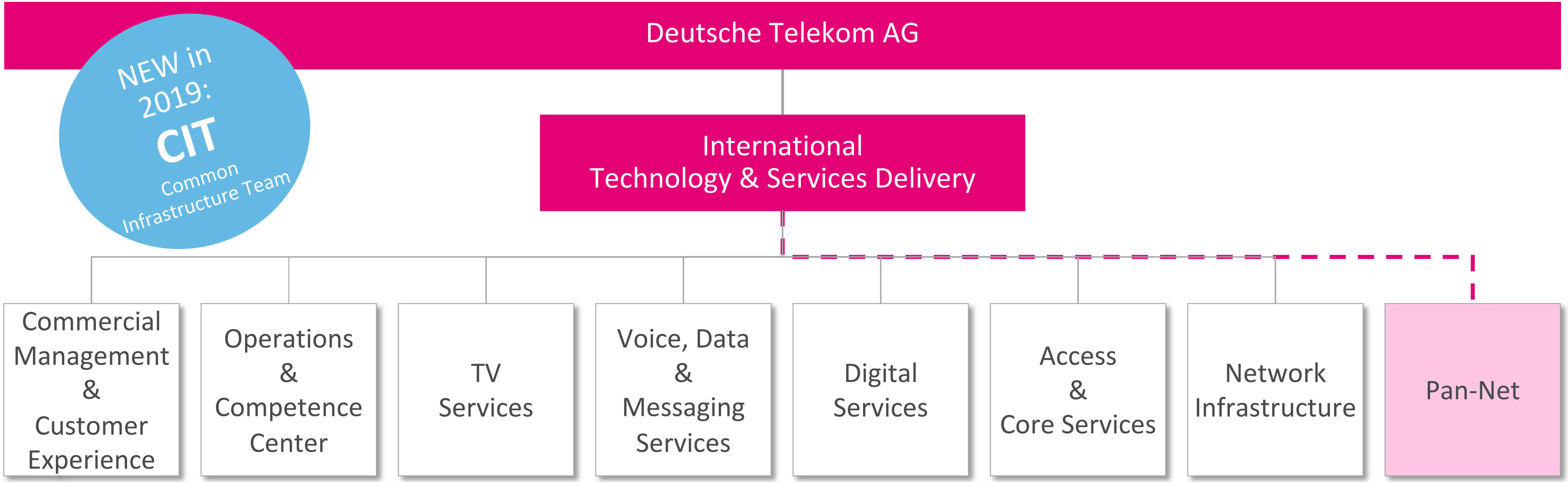
Product Delivery

Technology Leadership

Success on Local Market

Best Customer Experience

Today Pan-net is Part of ITS STRUCTURE which enables synergies and fosters collaboration in e2e delivery



Deutsche Telekom AG

NEW in 2019:
CIT
Common Infrastructure Team

International Technology & Services Delivery

- Commercial Management & Customer Experience
- Operations & Competence Center
- TV Services
- Voice, Data & Messaging Services
- Digital Services
- Access & Core Services
- Network Infrastructure
- Pan-Net

THE CLOUD

IC DEFINITION

IC = [I]nfrastructure [C]loud

an NFVI implementation

To be more profane: it is “just” an OpenStack

…plus PaaS

…plus SOC/Monitoring

…plus Security

…plus Contrail, Smartnic

…plus Ceph, Datera, SWIFT

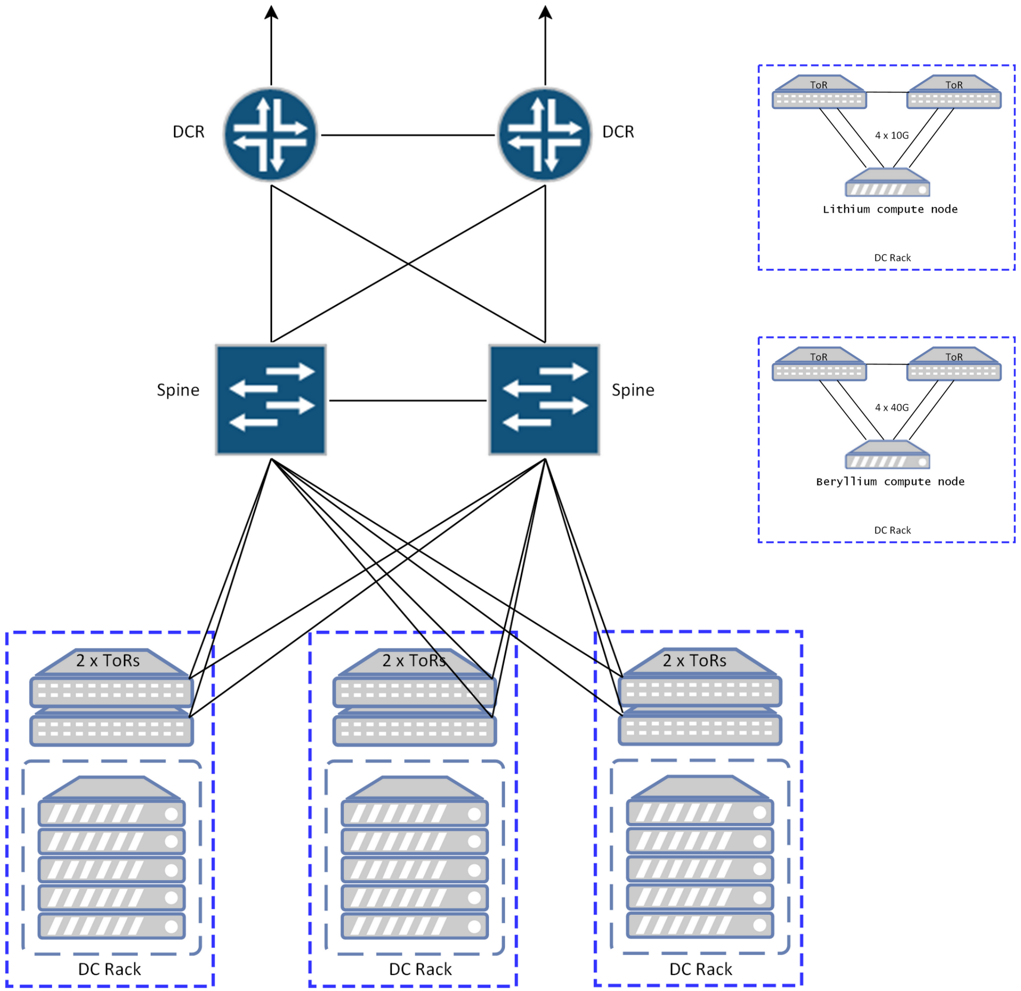
…plus Onboarding

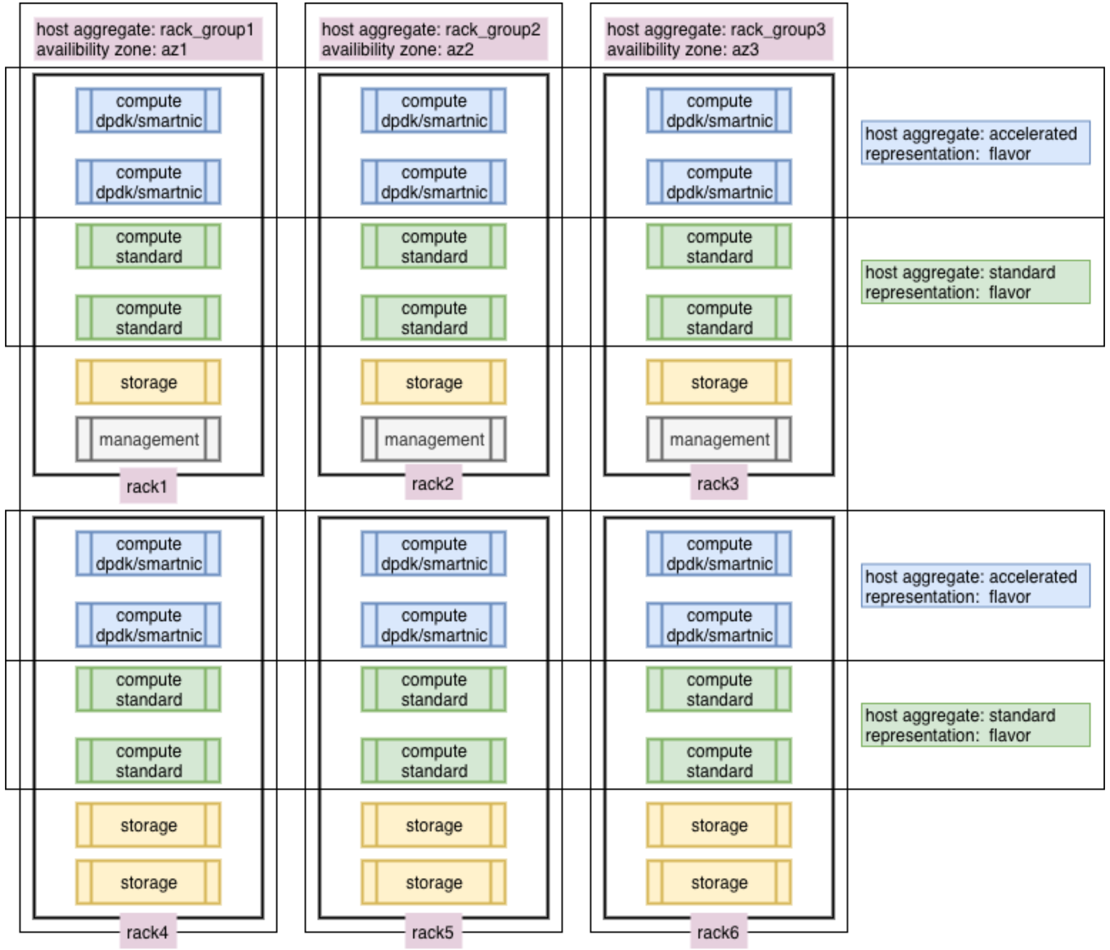
…plus Application Orchestration

…plus TaaS/LaaS

…plus Backbone

UNDERLAY DESIGN: RACKS, SERVERS, SWITCHES




host aggregate: rack_group1
availibility zone: az1

host aggregate: rack_group2
availibility zone: az2

host aggregate: rack_group3
availibility zone: az3

host aggregate: accelerated
representation: flavor

host aggregate: standard
representation: flavor

host aggregate: accelerated
representation: flavor

host aggregate: standard
representation: flavor

TOOLSET: MAAS, JUJU, ANSIBLE

MaaS

- Metal as a Service
- Baremetal/KVM managed as cloud resources
- Integrated with Juju

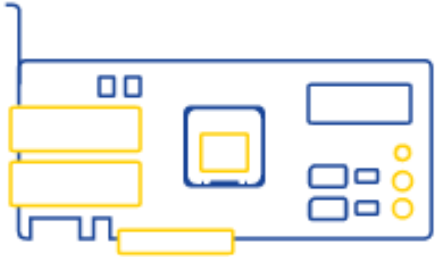
JUJU

- Modeling and configuration tool
- Bundle: charms, configuration, relations
- Integration tool with 3rd party components

Ansible

- Before Juju
- After Juju
- Even *with* Juju

COMPONENTS: CEPH, CONTRAIL, SMARTNICS, OPENSTACK



cloud

- NFVI platform
- CPU pinning
- Hugepages
- Network acceleration

storage

- Nova boot
- Cinder volume
- Glance
- Object storage: s3/swift

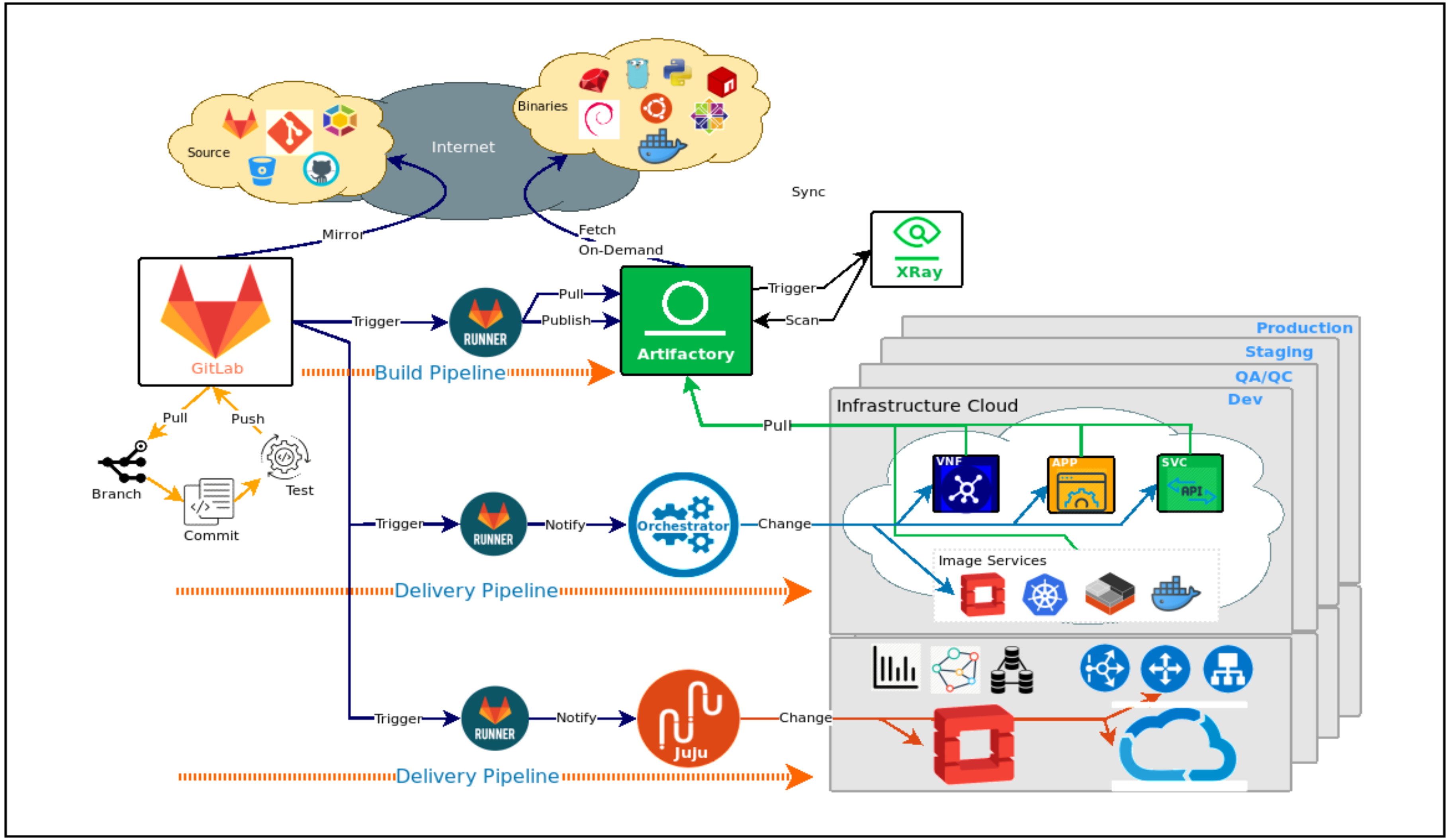
network

- SDN
- vRouter
- SmartNIC integrated

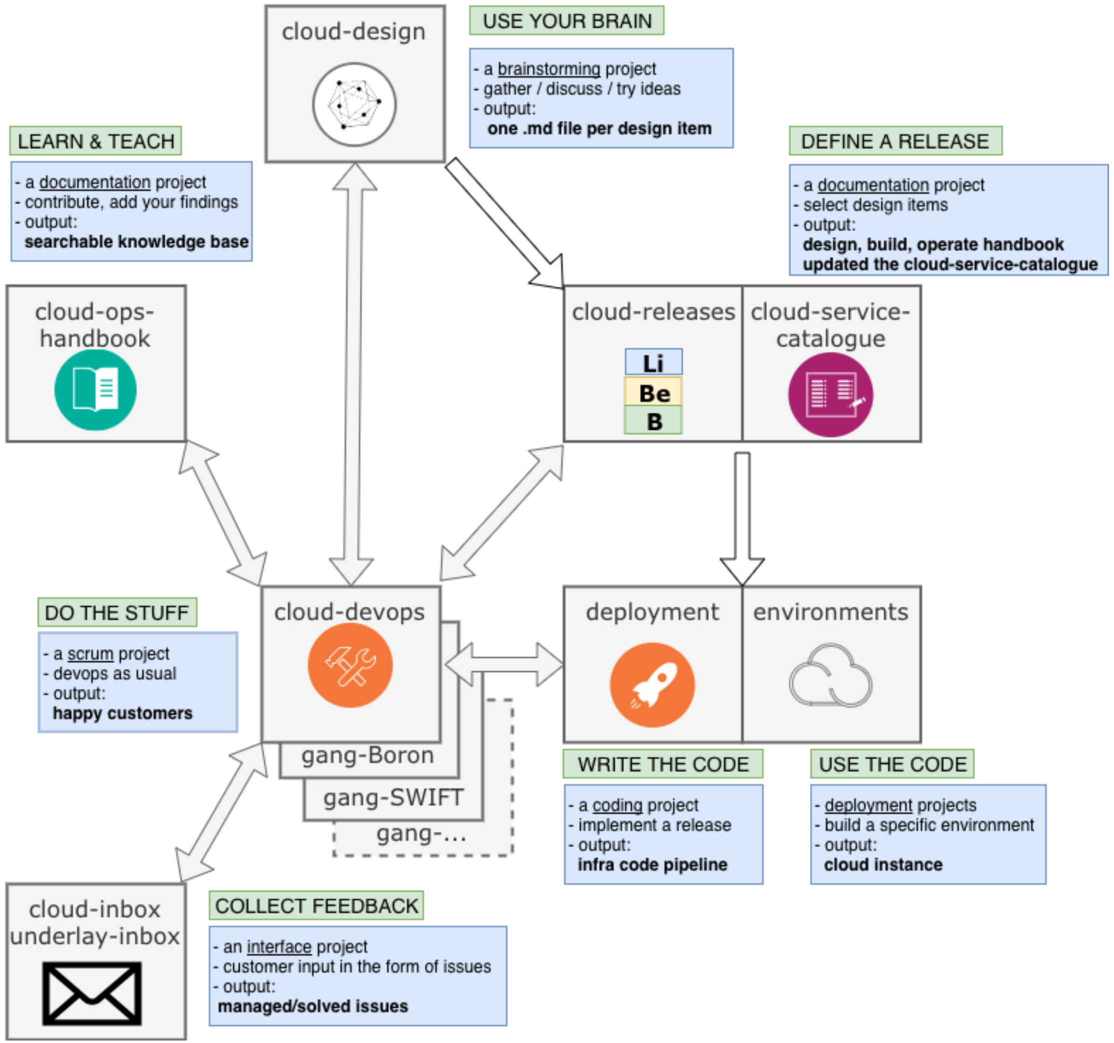
acceleration

- SRIOV
- Virtio-forwarder
- Contrail integrated

INFRASTRUCTURE AS CODE: PIPELINE DRIVEN DEPLOYMENT



CLOUD-FRAMEWORK: MIND MAP



cloud-design

USE YOUR BRAIN

- a brainstorming project
- gather / discuss / try ideas
- output:
  one .md file per design item

LEARN & TEACH

- a documentation project
- contribute, add your findings
- output:
  searchable knowledge base

cloud-ops-handbook

DEFINE A RELEASE

- a documentation project
- select design items
- output:
  design, build, operate handbook
  updated the cloud-service-catalogue

cloud-releases

Li

Be

B

cloud-service-catalogue

DO THE STUFF

- a scrum project
- devops as usual
- output:
  happy customers

cloud-devops

gang-Boron

gang-SWIFT

gang-…

deployment

environments

WRITE THE CODE

- a coding project
- implement a release
- output:
  infra code pipeline

USE THE CODE

- deployment projects
- build a specific environment
- output:
  cloud instance

cloud-inbox
underlay-inbox

COLLECT FEEDBACK

- an interface project
- customer input in the form of issues
- output:
  managed/solved issues

USE CASE: VNIC-TYPE NORMAL

```
- create a port first:
$ openstack port create --network net0 --vnic-type normal port0
- refer to the port directly:
$ openstack server create ... --nic port-id=2a4e8b1e-2904-428d-a743-77202aa82524 vm0
```

```
- ssh to your VM and run tcpdump on the hypervisor - you have a standard tap device:
# tcpdump -plenni tap2a4e8b1e-29
10:56:17.483946 02:78:cf:3e:d3:5a > 02:2a:4e:8b:1e:29, ethertype IPv4 (0x0800), length 74: 192.168.0.10.50548 > 192.168.0.11.22: Flags [S], seq
355862070, win 29200, options [mss 1460,sackOK,TS val 212334934 ecr 0,nop,wscale 7], length 0
10:56:17.484072 02:2a:4e:8b:1e:29 > 02:78:cf:3e:d3:5a, ethertype IPv4 (0x0800), length 74: 192.168.0.11.22 > 192.168.0.10.50548: Flags [S.], seq
113244970, ack 355862071, win 28960, options [mss 1460,sackOK,TS val 1744322024 ecr 212334934,nop,wscale 7], length 0
10:56:17.484375 02:78:cf:3e:d3:5a > 02:2a:4e:8b:1e:29, ethertype IPv4 (0x0800), length 66: 192.168.0.10.50548 > 192.168.0.11.22: Flags [.], ack 1, win
229, options [nop,nop,TS val 212334934 ecr 1744322024], length 0
10:56:17.484742 02:78:cf:3e:d3:5a > 02:2a:4e:8b:1e:29, ethertype IPv4 (0x0800), length 107: 192.168.0.10.50548 > 192.168.0.11.22: Flags [P.], seq 1:42,
ack 1, win 229, options [nop,nop,TS val 212334934 ecr 1744322024], length 41
```

You will see every packet – no surprise here.

USE CASE: VNIC-TYPE DIRECT

```
- create a port first:
$ openstack port create --network net0 --vnic-type direct port1
- refer to the port directly – the same way you did with the normal type:
$ openstack server create ... --nic port-id=7b35351f-fb36-4bf7-a2c2-a44f42768bc3 vm1
```

```
- find the port on the hypervisor - you have a VF assigned to the port:

# virsh dumpxml instance-00001189
...
<interface type='hostdev' managed='yes’>
<mac address='02:7b:35:35:1f:fb’/>

# ip a | grep -B1 02:7b:35:35:1f:fb
135: nfp_v0.57: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast portid 
20000039 state UP group default qlen 13000
    link/ether 02:7b:35:35:1f:fb brd ff:ff:ff:ff:ff:ff
```

USE CASE: VNIC-TYPE DIRECT (CONT'D)

```
- ssh to your VM and run tcpdump on the hypervisor:
# tcpdump -plenni nfp_v0.57
11:30:13.686039 02:78:cf:3e:d3:5a > 02:7b:35:35:1f:fb, ethertype IPv4 (0x0800), length 74: 192.168.0.10.46432 > 192.168.0.16.22: Flags [S], seq
900085604, win 29200, options [mss 1460,sackOK,TS val 212843976 ecr 0,nop,wscale 7], length 0
11:30:13.686265 02:7b:35:35:1f:fb > 02:78:cf:3e:d3:5a, ethertype IPv4 (0x0800), length 74: 192.168.0.16.22 > 192.168.0.10.46432: Flags [S.], seq
3866142375, ack 900085605, win 28960, options [mss 1460,sackOK,TS val 1625897928 ecr 212843976,nop,wscale 7], length 0
11:30:16.468811 02:78:cf:3e:d3:5a > 02:7b:35:35:1f:fb, ethertype IPv4 (0x0800), length 66: 192.168.0.10.46432 > 192.168.0.16.22: Flags [F.], seq 2882,
ack 5530, win 448, options [nop,nop,TS val 212844672 ecr 1625900710], length 0
11:30:16.477233 02:7b:35:35:1f:fb > 02:78:cf:3e:d3:5a, ethertype IPv4 (0x0800), length 66: 192.168.0.16.22 > 192.168.0.10.46432: Flags [F.], seq 5530,
ack 2883, win 312, options [nop,nop,TS val 1625900719 ecr 212844672], length 0
11:30:16.477529 02:78:cf:3e:d3:5a > 02:7b:35:35:1f:fb, ethertype IPv4 (0x0800), length 66: 192.168.0.10.46432 > 192.168.0.16.22: Flags [.], ack 5531, win
448, options [nop,nop,TS val 212844674 ecr 1625900719], length 0
```

This is the full ssh session!

You will see only the first and the last packets – all the rest is managed w/o the kernel, directly between the VM and the SmartNIC.

USE CASE: VNIC-TYPE VIRTIO-FORWARDER

```
- create a port first:
$ openstack port create --network net0 --vnic-type virtio-forwarder port2
- refer to the port directly – the same way you did with the normal and direct type:
$ openstack server create ... --nic port-id=c4ae4e95-5b90-4ec1-ab17-d27ab9f42294 vm2
```

```
- find the port on the hypervisor - you have a VF assigned to the port:

# virsh dumpxml instance-00001183
 <interface type='vhostuser'>
      <mac address='02:c4:ae:4e:95:5b’/>

# vif --list | grep -B5 02:c4:ae:4e:95:5b
vif0/4      OS: nfp_v0.58
            Type:Virtual HWaddr:00:00:5e:00:01:00 IPaddr:192.168.0.14
            ...
            ISID: 0 Bmac: 02:c4:ae:4e:95:5b
```

USE CASE: VNIC-TYPE VIRTIO-FORWARDER (CONT'D)

- *kernel boot messages from a VM with vnic-type=direct:*

```
- ubuntu@langyal-nic-direct-01:~$ dmesg|egrep -i "nic|nfp"
- [    0.759082] pcie_mp2_amd: AMD(R) PCI-E MP2 Communication Driver Version: 1.0
- [    0.967781] nfp: NFP PCIe Driver, Copyright (C) 2014-2017 Netronome Systems
- [    1.012483] nfp_netvf 0000:00:04.0 eth0: Netronome NFP-6xxx VF Netdev: TxQs=1/1 RxQs=1/1
- [    1.014342] nfp_netvf 0000:00:04.0 eth0: VER: 0.0.3.0, Maximum supported MTU: 9216
- [    1.016028] nfp_netvf 0000:00:04.0 eth0: CAP: 0x14063f PROMISC L2BCFILT L2MCFILT RXCSUM TXCSUM GATHER TSO1 AUTOMASK IRQMOD
- [    1.037428] nfp_netvf 0000:00:04.0 ens4: renamed from eth0
- [    5.559701] nfp_netvf 0000:00:04.0 ens4: RV00: irq=028/002
- [    5.572136] nfp_netvf 0000:00:04.0 ens4: NIC Link is Up
- [   21.888259] nfp_netvf 0000:00:04.0 ens4: ens4 down
- [   22.170628] nfp_netvf 0000:00:04.0 ens4: RV00: irq=028/002
- [   22.180139] nfp_netvf 0000:00:04.0 ens4: NIC Link is Up
```

- *kernel boot messages from a VM with vnic-type=virtio-forwarder:*

```
- ubuntu@langyal-nic-virtio-forwarder-01:~$ dmesg|egrep -i "nic|nfp"
- [    0.768081] pcie_mp2_amd: AMD(R) PCI-E MP2 Communication Driver Version: 1.0
```

The *vnic-type=virtio-forwarder* provides close to sriov speed plus easy integration with openstack kvm instances.

SUMMARY: SERVICES

NEW in 2019: Managed K8S is coming!

COMPUTE

- vCPU assignment
- No memory overcommit
- Smartnic acceleration

NETWORK

- Neutron
- Contrail
- Backbone

STORAGE

- Block: Ceph
- Block: Datera
- Object: rados-gw, SWIFT

PAAS

- DNSaaS
- NTPaaS
- LBaaS (L4, contrail ECMP)

MULTISITE

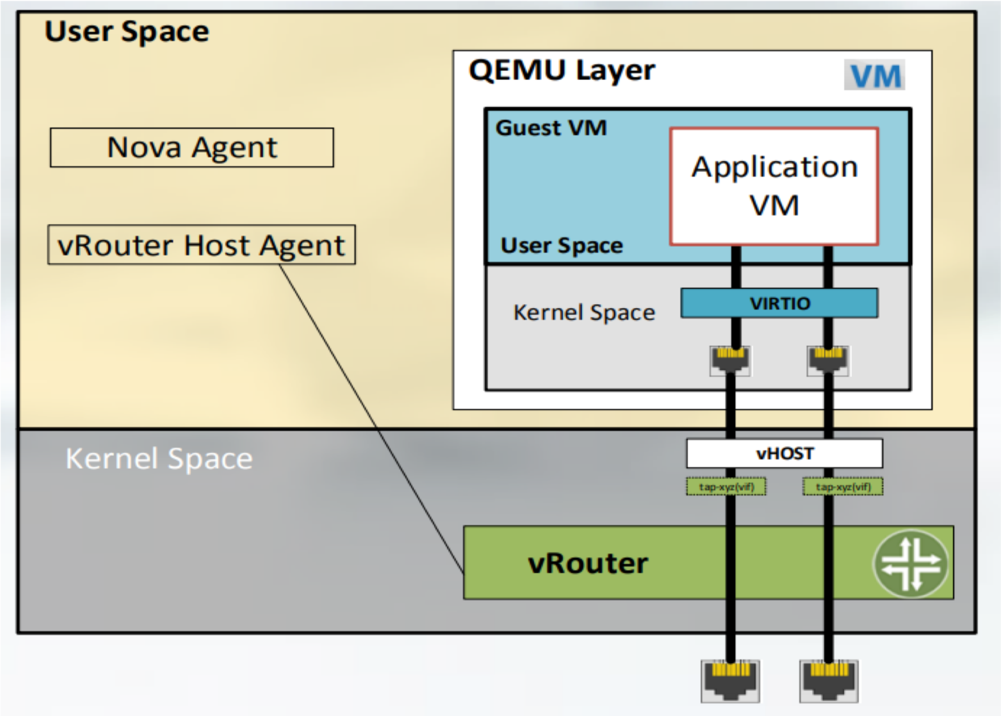
- IP anycast
- Global health check (consul)
- Object storage replication

SOC

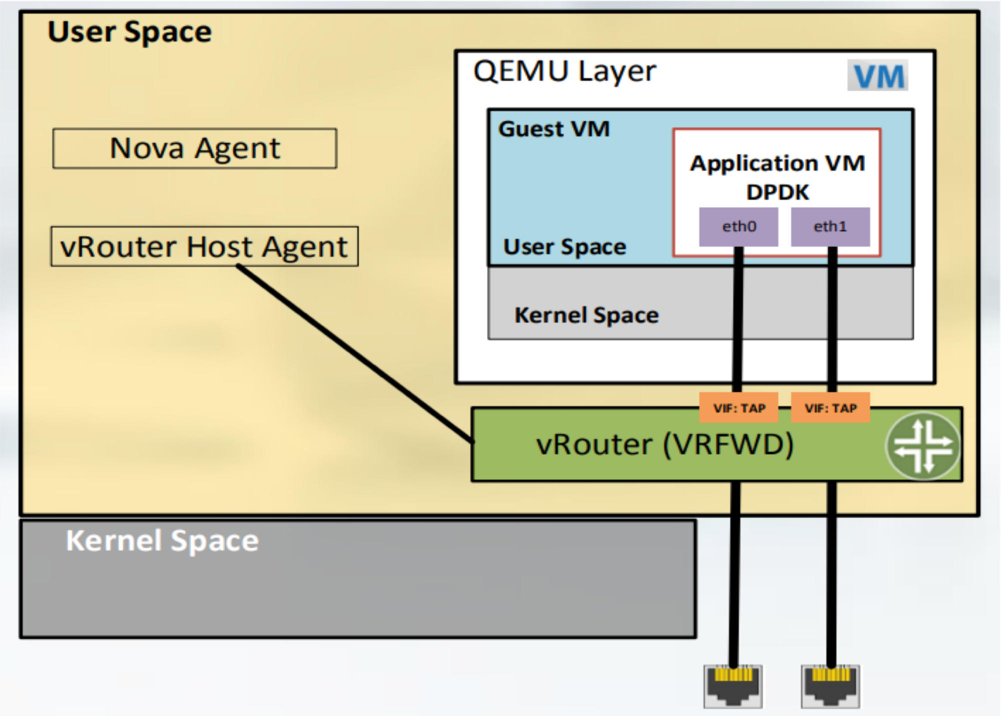
- Continuous monitoring
- Change management
- 7x24 operation

THE VEPC POC

THE BASICS: FORWARDING ARCHITECTURE W/O SMARTNIC

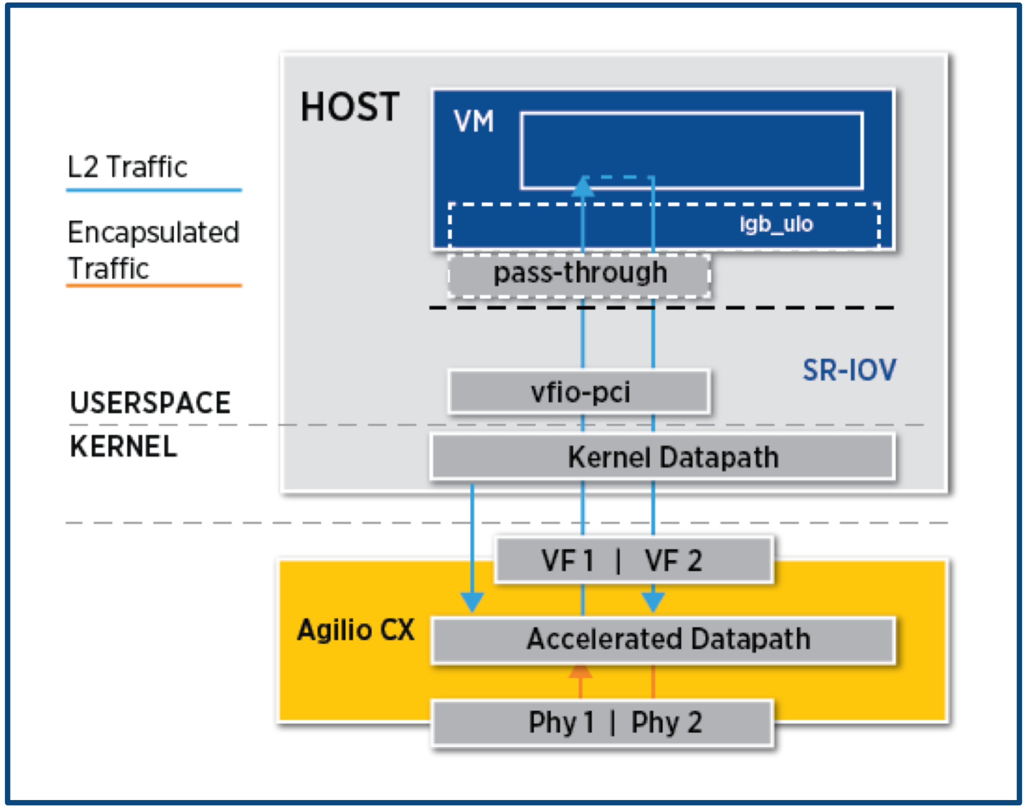


Kernel Space mode / No DPDK

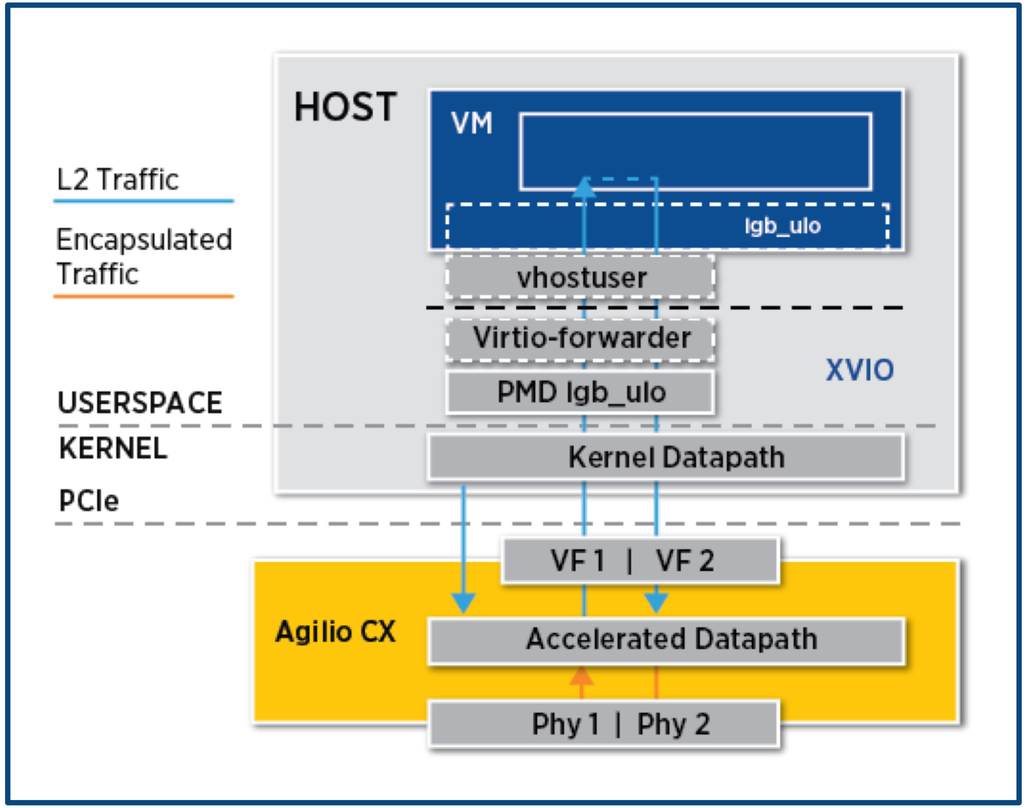


User Space mode / DPDK

THE BASICS: FORWARDING ARCHITECTURE WITH SMARTNIC

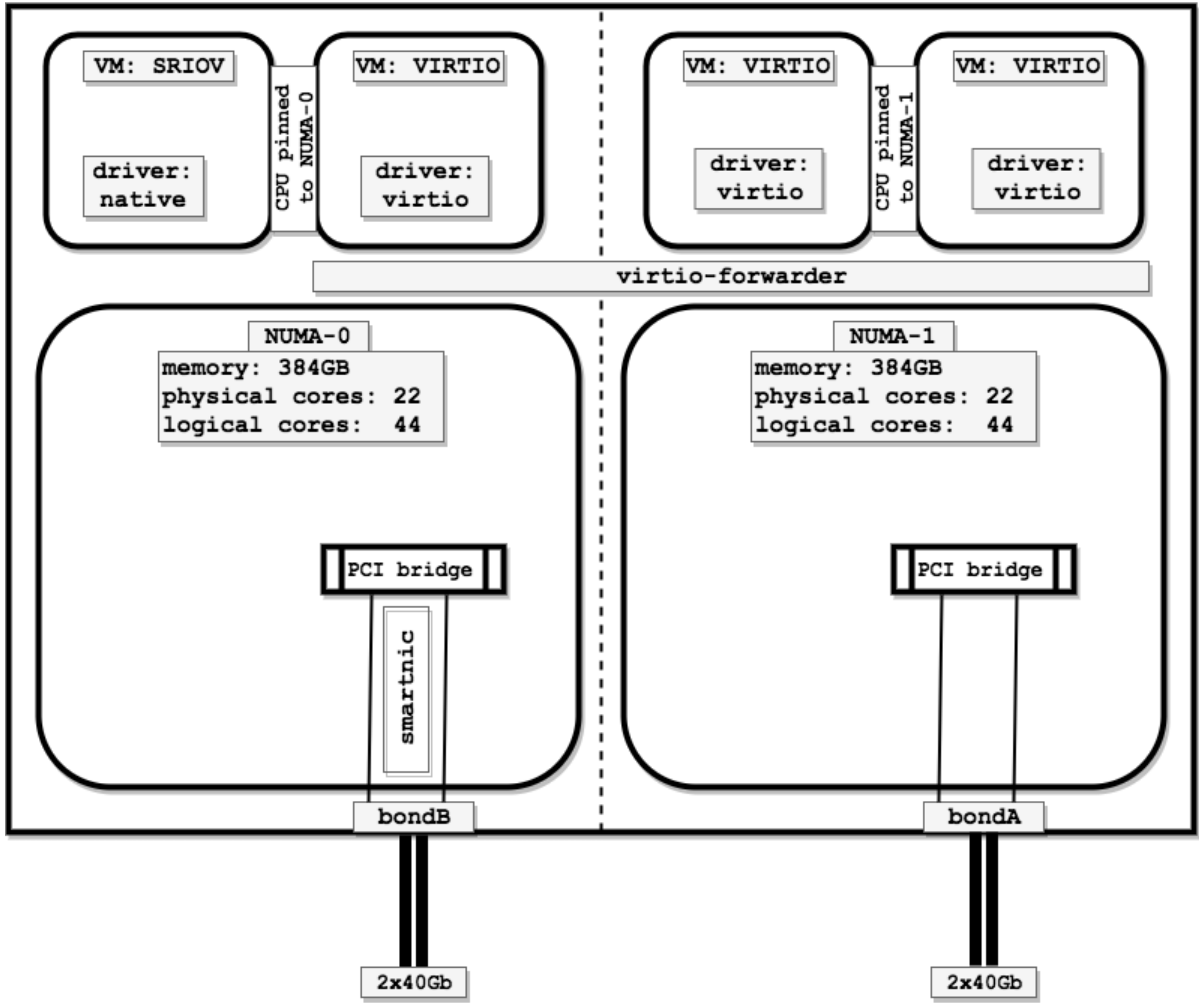


SR-IOV mode

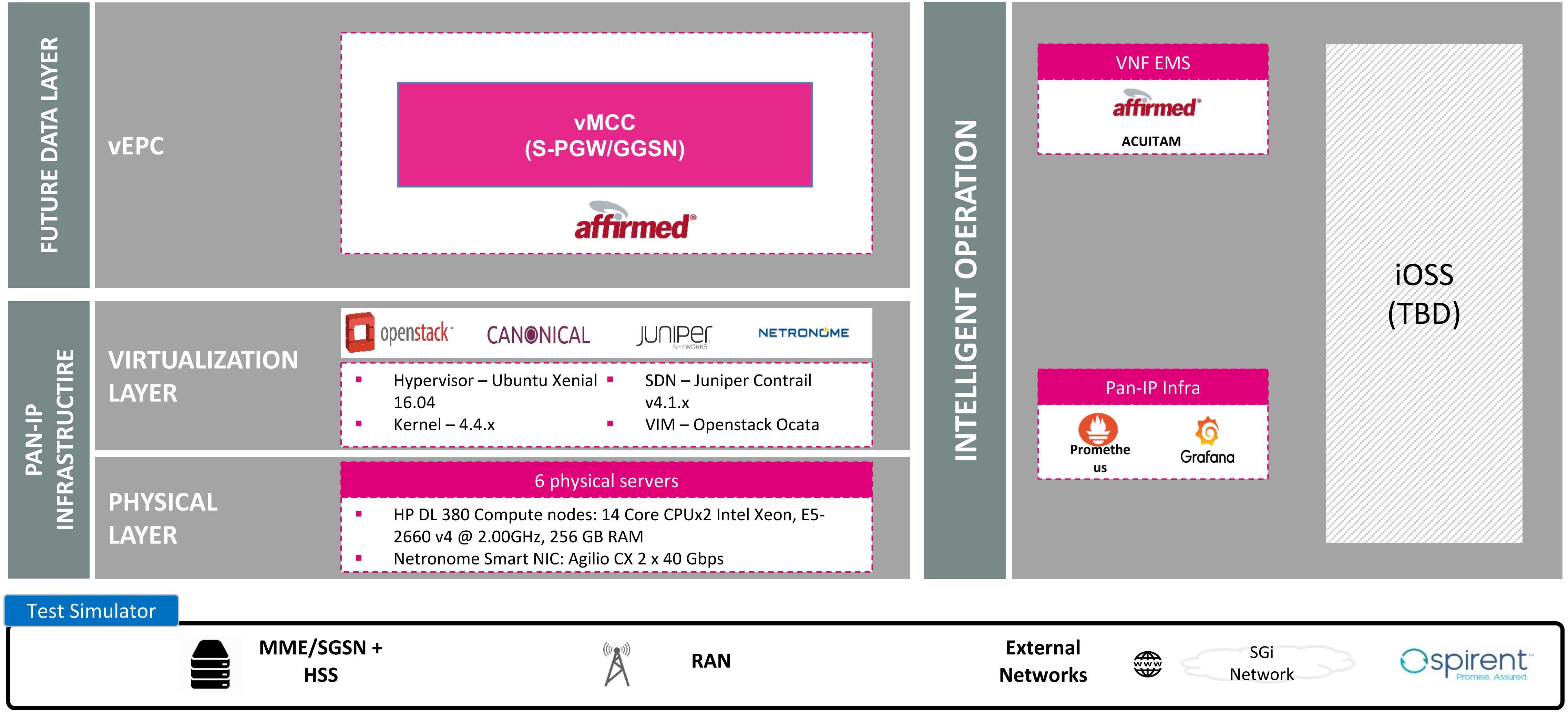


XVIO mode (virtio-forwarder)

COMPUTE NODE DESIGN



VEPC POC – DESIGN



FUTURE DATA LAYER

vEPC

vMCC
(S-PGW/GGSN)

affirmed

PAN-IP INFRASTRUCTIRE

VIRTUALIZATION LAYER

openstack · CANONICAL · JUNIPER NETWORKS · NETRONOME

- Hypervisor – Ubuntu Xenial 16.04
- Kernel – 4.4.x
- SDN – Juniper Contrail v4.1.x
- VIM – Openstack Ocata

PHYSICAL LAYER

6 physical servers

- HP DL 380 Compute nodes: 14 Core CPUx2 Intel Xeon, E5-2660 v4 @ 2.00GHz, 256 GB RAM
- Netronome Smart NIC: Agilio CX 2 x 40 Gbps

INTELLIGENT OPERATION

VNF EMS

affirmed

ACUITAM

Pan-IP Infra

Promethe
us

Grafana

iOSS
(TBD)

Test Simulator

MME/SGSN +
HSS

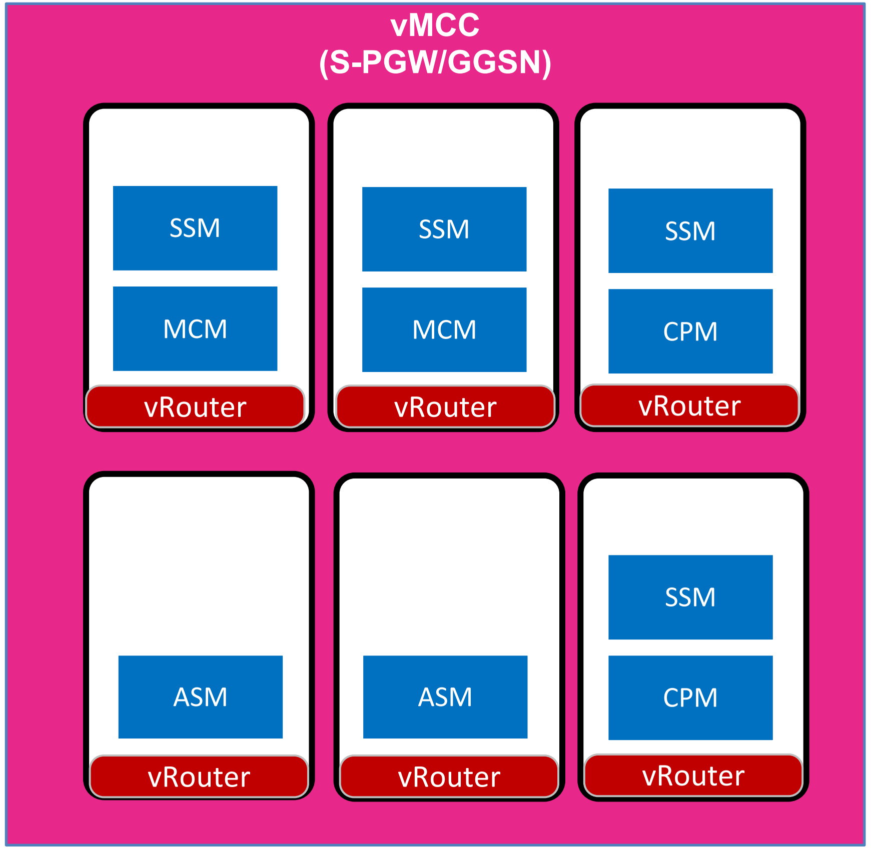
RAN

External
Networks

SGi
Network

spirent
Promise. Assured.

VEPC POC – RESOURCES



Resource allocation

VM type	Instances	vCPU	vRAM (GB)	Storage (GB)
MCM	2	**8**	**32**	544
CPM	2	**16**	**64**	114
SSM	4	**22**	**96**	114
ASM	2	**22**	**64**	114

Legend

MCM – Management Configuration Module
ASM – Advanced Services Module
SSM – Subscriber Services Module
CPM – Control Plane Module

VEPC POC - SUMMARY

High Level Test Objectives - using Pan-Net Infrastructure Cloud

Performance	High Availability	Scalability study	Operationalization	Cloud Readiness
Performance test (with and without Traffic Anchoring enabled): • 4G throughput verification with multiple call profiles • 4G signaling verification	**Failover of:** - Physical, Virtual and logical components	**Performance 4G scenarios with:** • Increasing packet handling VM's on different compute nodes (1 SSM, 2 SSM etc.)	**- Perf/Fault Mgmt on EMS** - Performance / Fault on EMS - Capacity management - Tracing and troubleshooting **- Perf/Fault Mgmt on iOSS (TBD)**	**Scaling / Healing functions** **In-Service Upgrade**
N° Test cases: 25	N° Test cases: 11	N° Test cases: 4	N° Test cases: 10	N° Test cases: 7
• **Validate 4G Performance and KPIs (delay, packet loss)** achieved on top of Pan-Net Infra when employing **SMART NICs** • **Verify the optimization** that TAM mode provides in terms of East-West traffic	**Validate achievable efficiency in BGP network convergence when using** • Beryllium • BFD	**Perform scalability study and possible improvements** when employing **SMART NICs**	**Validate Operational readiness of the end to end solution** (Pan-Net Infrastructure & vEPC): - Events/Counters/KPIs validation - Troubleshooting and tracing capabilities	**Validate cloud functions of the solution** (Pan-Net Infrastructure & vEPC)

NFVI = acceleration

THANK YOU

László Angyal

laszlo.angyal@pan-net.eu

https://langyal.gitlab.io/blog/

LIFE IS FOR SHARING.