



# 5G CHAMPION

**5G** Communication with a **H**eterogeneous,  
**A**gile **M**obile network in the **P**yeongchang  
**w**inter **O**lympic competition

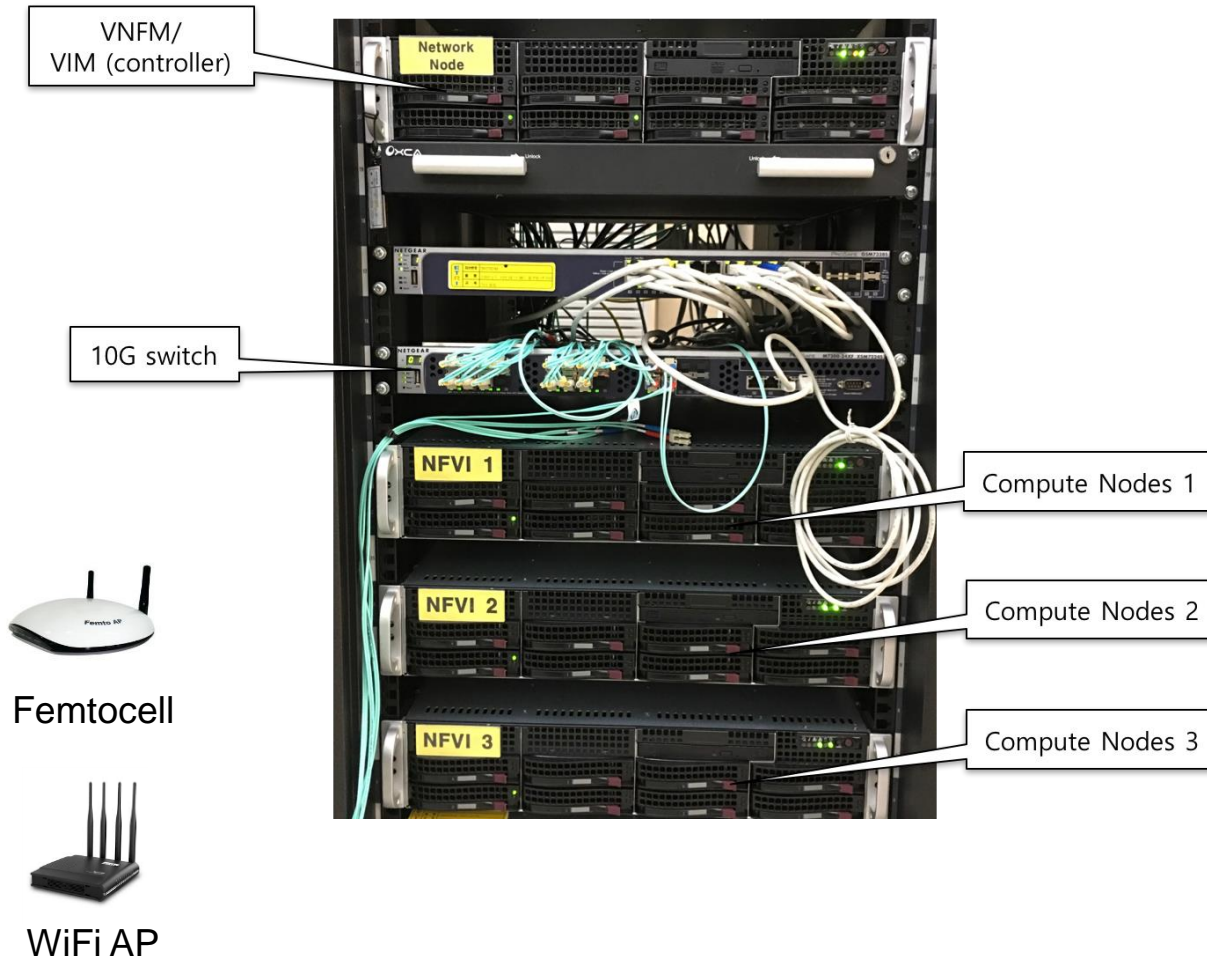
*5G ready: Orchestration of 5G services – Core network aspects*

EU-KR Symposium on 5G, Feb. 23, 2018  
Taesang Choi



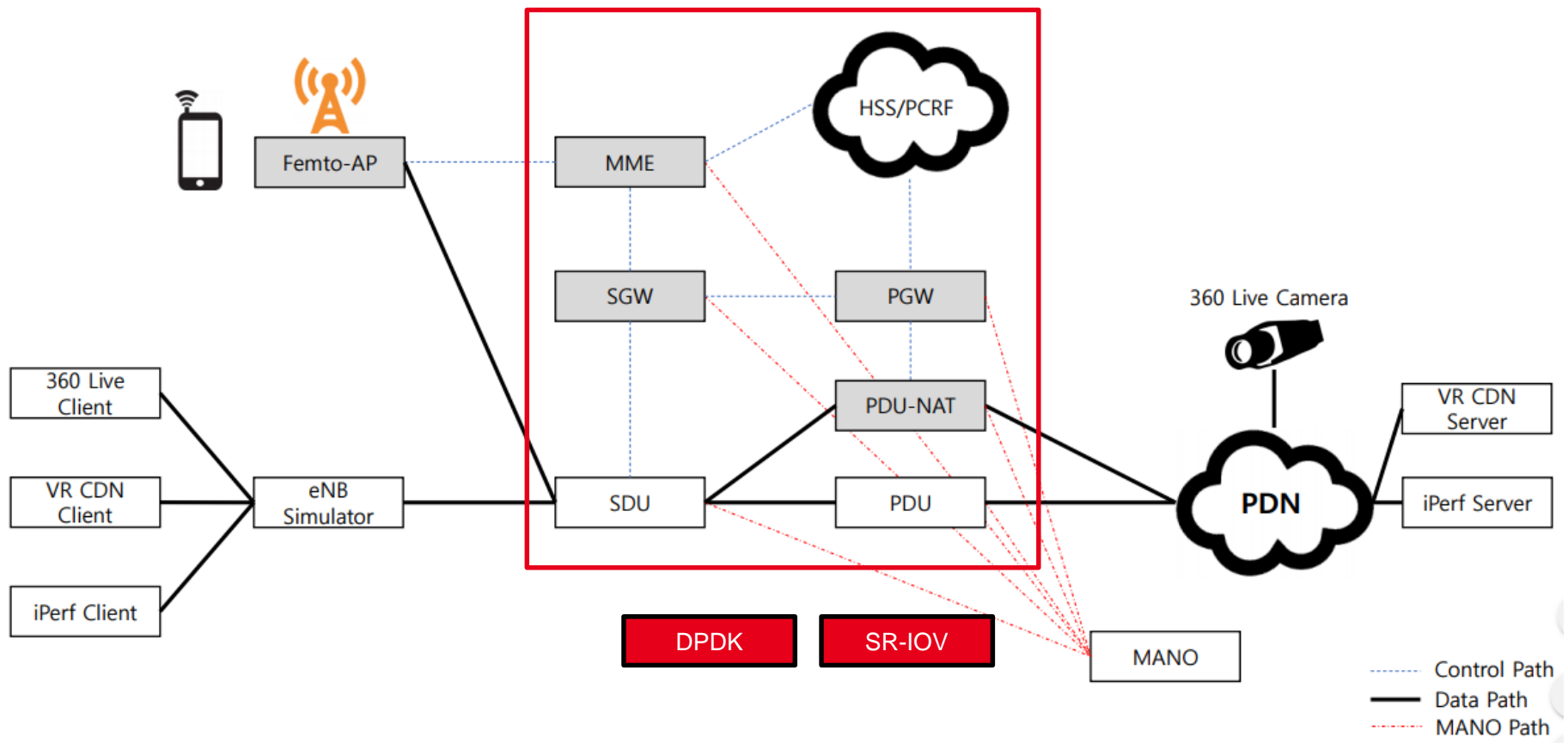


# Mobile Core H/W Infrastructure for GangNeung K-ICT





# Mobile Core Virtual Network Functions (VNFs)

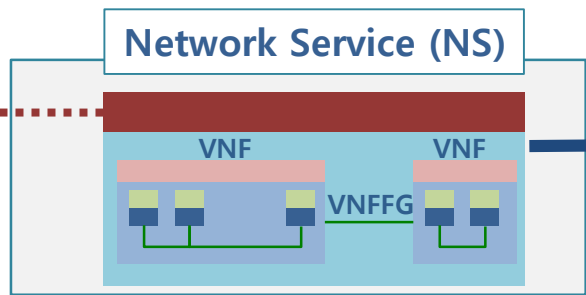


- **Manages the lifecycle of NS**
- Exposes NS lifecycle management interfaces to the OSS/BSS
- Sends NS lifecycle management notifications to the OSS/BSS
- Exposes virtualized resource management interfaces to the VNFM
- Sends virtualized resource management notifications to the VNFM
- Manages the VNF lifecycle via the interfaces exposed by the VNFM
- Manages virtualized resources via the interfaces exposed by the VIM.



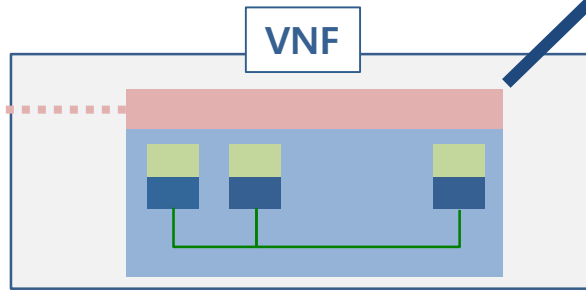
## NFV Management & Orchestration

Functional Management of Network Service



**Network Service (NS) Management**

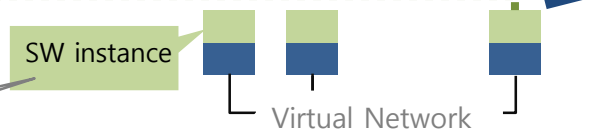
Functional Management of (V)NFs



**VNF Management (VNFM)**

- **Manages the lifecycle of VNFs**
- **Manages virtualized resources associated to the VNF** it manages via the interfaces exposed by the VIM or NFVO
- Exposes VNF lifecycle management interfaces/APIs to the VNF, EM and NFVO.
- Sends VNF lifecycle management notifications to the VNF, EM and NFVO
- **Manages VNF initial configuration** via the interfaces exposed by the VNF.

Functional Management of SW Instances



**Virtualized Resources Management (VIM)**

- **Manages the NFV infrastructure resources** (compute, network and storage) in one or more NFVI-PoPs.
- Exposes virtualized resource management interfaces/APIs to the VNFM and NFVO
- Sends virtualized resource management notifications to the VNFM and the NFVO

Virtual C/N/S + virtual Storage (VM, container, ...)





# MANO Implementation: VNF Instances Management



### NFV Orchestrator

for Cloud NFV System

**INSOFT** **ETRI**

- Network Service
- Network Function
- Template Define
- Management Admin
- Virtual Infra Management
- LifeCycle Management
  - NSR
  - VNFC
  - Alarm

NPVO VNFM
Log out 한국어

## Network Service Record Operator

Network Service Record / Network Service Record Operator

Scale Out  
Scale In  
Reboot  
Binary Update  
Config Update  
Custom  
Embedded VNF

vEPC Demo

- vEPC Demo
  - vPGW-CU
    - vPGW-CU-VDU1
  - vSGW-CU
    - vSGW-CU-VDU1
      - Configuration
  - MME
    - MME-VDU1
  - vPGW-DU
    - vPGW-DU-VDU1
      - pdu config
  - vSGW-DU
    - vSGW-DU-VDU1
      - sdu config

Instance
Network Topology
Forwarding Path

<input type="checkbox"/>	VNFR	INSTANCE	Internal IP	Floating IP	Image	VIM	Compute	VM	Agent	Monitoring	Console
<input type="checkbox"/>	MME	MME_VNFC	192.168.230.3	129.254.191.137	vpec-agent	VIM-1(Gangneung ETRI)	nfv-t02-compute01.domain.tld	ACTIVE	🔄	Information	📄
<input type="checkbox"/>	vPGW-CU	vPGW-CU_VNFC	192.168.231.4	129.254.191.146	vpec-agent	VIM-1(Gangneung ETRI)	nfv-t02-compute01.domain.tld	ACTIVE	🔄	Information	📄
<input type="checkbox"/>	vPGW-DU	vPGW-DU_VNFC	192.168.231.6	129.254.191.216	vpec-agent	VIM-1(Gangneung ETRI)	nfv-t02-compute02.domain.tld	ACTIVE	Running	Information	📄
<input type="checkbox"/>	vSGW-CU	vSGW-CU_VNFC	192.168.231.3	129.254.191.217	vpec-agent	VIM-1(Gangneung ETRI)	nfv-t02-compute01.domain.tld	ACTIVE	🔄	Information	📄
<input type="checkbox"/>	vSGW-DU	vSGW-DU_VNFC	192.168.231.5	129.254.191.142	vpec-agent	VIM-1(Gangneung ETRI)	nfv-t02-compute03.domain.tld	ACTIVE	Running	Information	📄

**Monitoring Information** 🔍 Off

- CPU and Memory Trace
- NFILE USAGE
- QUEUE COUNT
- KBPS Traffic
- Disk Usage
- Inode Usage Rate

NSR | vEPC Demo
🕒 07개월 0일 00:45:26



# MANO Implementation: VNF Topology Management



## NFV Orchestrator

for Cloud NFV System

INSOFT ETRI

- Network Service
- Network Function
- Template Define
- Management Admin
- Virtual Infra Management
- LifeCycle Management
  - NSR
  - VNFC
  - Alarm

Network Service Record Operator

Network Service Record / Network Service Record Operator

Scale Out, Scale In, Reboot, Binary Update, Config Update, Custom, Embedded VNF

vEPC Demo

- vEPC Demo
  - vPGW-CU
    - vPGW-CU-VDU1
  - vSGW-CU
    - vSGW-CU-VDU1
    - Configuration
  - MME
    - MME-VDU1
  - vPGW-DU
    - vPGW-DU-VDU1
    - pdu config
  - vSGW-DU
    - vSGW-DU-VDU1
    - sdu config

NSR | vEPC Demo 0개월 0일 00:43:23

Instance Network Topology Forwarding Path

VIM-1(Gangneung ETRI)

The diagram illustrates a network topology for VIM-1(Gangneung ETRI). It features four vertical network segments: 1. 'admin floating net' (blue) with IP 129.254.191.0/24, connected to two server icons. 2. 'vEPC-MME-Network' (green) with IP 192.168.230.0/24, connected to the admin floating net. 3. 'vEPC-MGT-Network' (orange) with IP 192.168.231.0/24, connected to the vEPC-MME-Network and containing three server icons. 4. 'vEPC-SRIOV-Network' (red) with IP 192.168.235.0/24, connected to the vEPC-MGT-Network and containing three server icons.



# MANO Implementation: VNF Resource Management



**NFV Orchestrator**  
for Cloud NFV System

INSOFT ETRI

Network Service Record Operator

Network Service Record / Network Service Record Operator

Scale Out, Scale In, Reboot, Binary Update, Config Update, Custom, Embedded VNF

VEPC Demo

- VEPC Demo
  - vPGW-CU
    - vPGW-CU-VDU1
  - vSGW-CU
    - vSGW-CU-VDU1
      - Configuration
  - MME
    - MME-VDU1
  - vPGW-DU
    - vPGW-DU-VDU1
      - pdu config
  - vSGW-DU
    - vSGW-DU-VDU1
      - sdu config

NSR | vEPC Demo 08:15 08:28:55

Instance Network Topology Forwarding Path

Instance	Network Topology	Forwarding Path
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

VNFR	INSTANCE	Internal IP	Floating IP	Image	VIM	Compute	VM	Agent	Monitoring	Console
<input type="checkbox"/>	MME	192.168.230.3	129.254.191.137	vpec-agent	VIM-1(Gangneung ETRI)	nfv-t02-compute01.domain.tld	ACTIVE	Running	Information	⌵
<input type="checkbox"/>	vPGW-CU	192.168.231.4	129.254.191.146	vpec-agent	VIM-1(Gangneung ETRI)	nfv-t02-compute01.domain.tld	ACTIVE	Running	Information	⌵
<input checked="" type="checkbox"/>	vPGW-DU	192.168.231.6	129.254.191.216	vpec-agent	VIM-1(Gangneung ETRI)	nfv-t02-compute02.domain.tld	ACTIVE	Running	Information	⌵
<input type="checkbox"/>	vSGW-CU	192.168.231.3	129.254.191.217	vpec-agent	VIM-1(Gangneung ETRI)	nfv-t02-compute01.domain.tld	ACTIVE	Running	Information	⌵
<input type="checkbox"/>	vSGW-DU	192.168.231.5	129.254.191.142	vpec-agent	VIM-1(Gangneung ETRI)	nfv-t02-compute03.domain.tld	ACTIVE	Running	Information	⌵

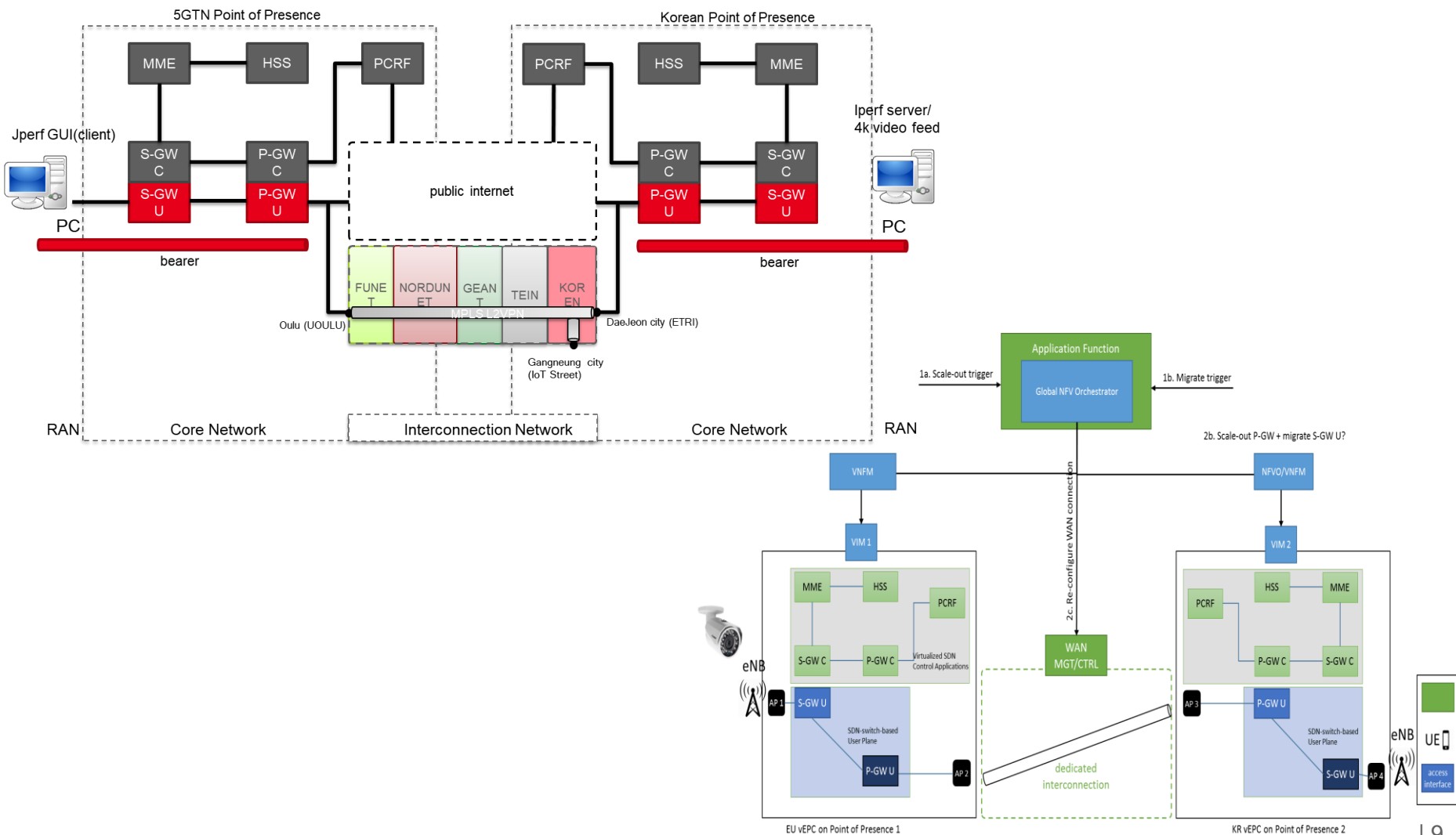
vPGW-DU\_VNFC

- CPU and Memory Trace
- NFILE USAGE
- QUEUE COUNT
- KBPS Traffic
- Disk Usage
- Inode Usage Rate

The monitoring dashboard for vPGW-DU\_VNFC displays several key performance indicators over time (08:15 to 08:20):

- CPU and Memory Trace:** Shows CPU Usage(%) fluctuating around 50%, Memory Usage(%) around 25%, and Swap Usage(%) near 0%. A 'Wait' state is also indicated.
- NFILE USAGE:** Shows Nfile Usage(%) around 50% and Nfile Count increasing from approximately 1,000 to 1,500.
- QUEUE COUNT:** Shows Queue Count dropping from 1.0 to 0.0 at 08:15 and returning to 1.0 at 08:20.
- KBPS Traffic:** Shows a significant spike in traffic (KBPS) at 08:15, reaching approximately 1.5 KBPS, with a corresponding spike in Error Packets.
- Disk Usage:** Shows Usage(%) for /dev/jdev/shm/rubyfs/cgroup and /boot, with /boot showing a notable increase.
- Inode Usage Rate:** Shows Usage(%) for /dev/jdev/shm/rubyfs/cgroup and /boot, with /boot showing a notable increase.

- Intercontinental Interconnection Demo (1st Review Meeting, Finland, June 2017) : System Interoperability
- MANO Interoperability Testing between EU-KR 5G Core (PyeongChang Olympic, Feb. 2018)





# Intercontinental Interconnection Demo: TCP Throughput Testing



JPerf 2.0.2 - Network performance measurement graphical tool

JPerf

Iperf command: `iperf -s -u -P 0 -i 1 -p 5002 -f k`

Choose iPerf Mode:  Client  Server

Server address: \_\_\_\_\_ Port:

Parallel Streams:

Listen Port:   Client Limit

Num Connections:

---

**Application layer options**

Enable Compatibility Mode

Transmit:   Bytes  Seconds

Output Format:

Report Interval:  seconds

Testing Mode:  Dual  Trade

test port:

Representative File:

Print MSS

---

**Transport layer options**

Choose the protocol to use

TCP

Buffer Length:  MBytes

TCP Window Size:  KBytes

Max Segment Size:  KBytes

TCP No Delay

UDP

---

**Bandwidth & Jitter** Tue, 14 Nov 2017 15:13:50

#3: 58.80KBits/s #6: 751426.00KBits/s  
Jitter: 1338.48ms

**Output**

```
[ 6] 26.0-27.0 sec 91785 KBytes 751899 Kbits/sec
[ 6] 27.0-28.0 sec 91463 KBytes 749265 Kbits/sec
[ 6] 28.0-29.0 sec 90160 KBytes 738587 Kbits/sec
[ 6] 29.0-30.0 sec 91604 KBytes 750416 Kbits/sec
[ 6] 0.0-30.0 sec 2751806 KBytes 751426 Kbits/sec
[ 6] Sent 1919494 datagrams
```

Clear Output on each Iperf Run



# Intercontinental Interconnection Demo: UDP Throughput Testing



JPerf 2.0.2 - Network performance measurement graphical tool

JPerf

Iperf command: `iperf -c 192.168.231.10 -u -P 1 -i 1 -p 5002 -f k -b 1000.0M -t 10 -T 1`

Choose IPerf Mode:  Client  Server

Server address: 192.168.231.10 Port: 5,002

Parallel Streams: 1

Listen Port: 5,001  Client Limit

Num Connections: 0

Output Format: KBits

Report Interval: 1 seconds

Testing Mode:  Dual  Trade

test port: 5,001

Representative File: ...

Print MSS

**Transport layer options**

Choose the protocol to use

TCP

Buffer Length: 2 MBytes

TCP Window Size: 56 KBytes

Max Segment Size: 1 KBytes

TCP No Delay

UDP

UDP Bandwidth: 1,000 MBytes/sec

UDP Buffer Size: 41 KBytes

UDP Packet Size: 1,500 Bytes

**IP layer options**

TTL: 1

Type of Service: None

Bind to Host: ...

IPv6

Run Iperf!

Stop IPerf!

Tue, 14 Nov 2017 15:07:14

### Bandwidth

Time (sec)	Bandwidth (Kbits/sec)
1.0	630000
2.0	620000
3.0	600000
4.0	630000
5.0	640000
6.0	580000
7.0	620000
8.0	600000
9.0	630000
10.0	620000

#4: [619647.00KBits/s]

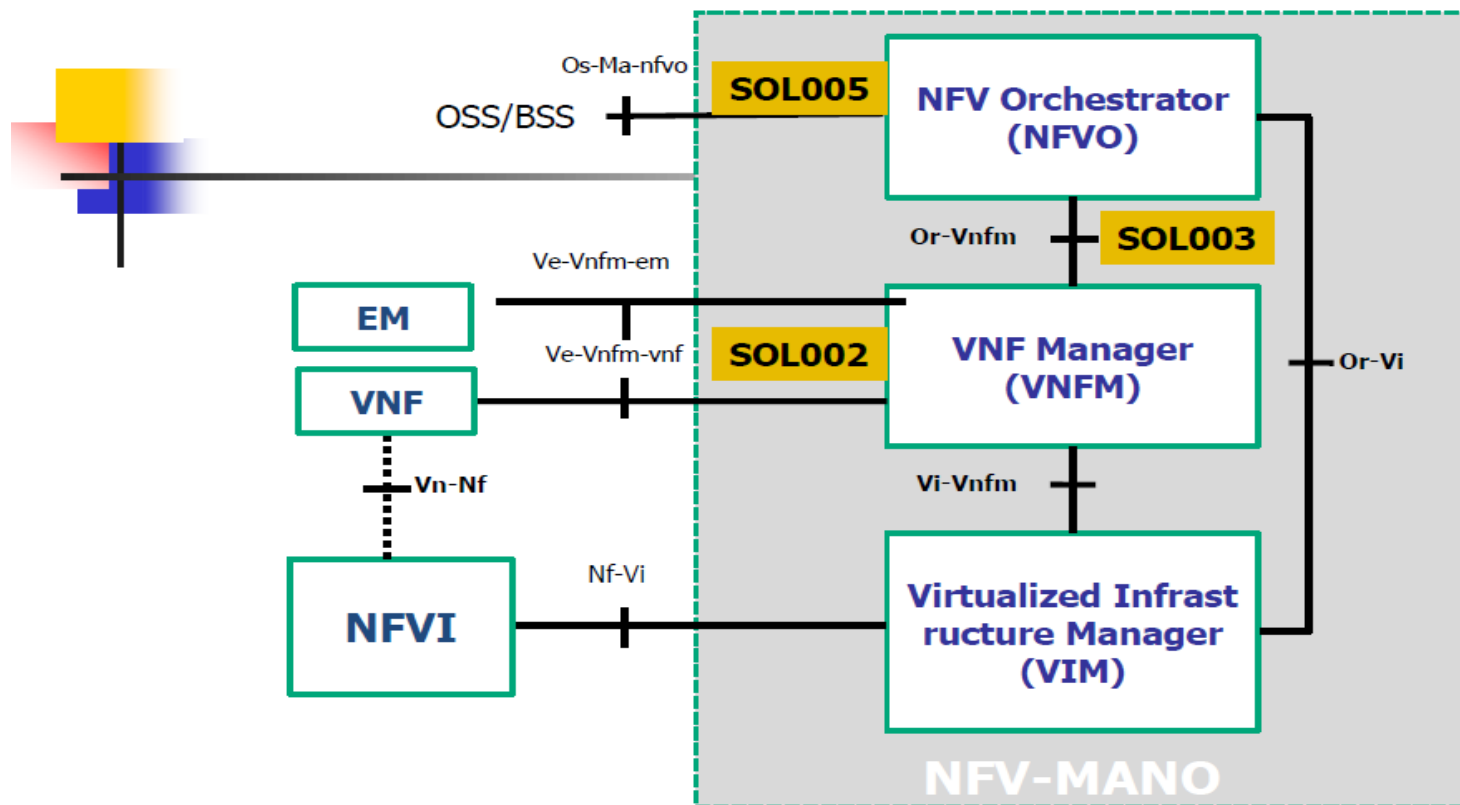
Output

```
[ 4] 0.0-10.0 sec 756469 Kbytes 619647 Kbits/sec
[ 4] Sent 633788 datagrams
[ 4] Server Report:
[ 4] 0.0-10.2 sec 287419 KBytes 229779 Kbits/sec 8.160 ms 433571/633787 (68%)
[ 4] 0.0-10.2 sec 30 datagrams received out-of-order
Done.
```

Save Clear now  Clear Output on each Iperf Run

- ETSI NFV SOL WG (Solution WG)

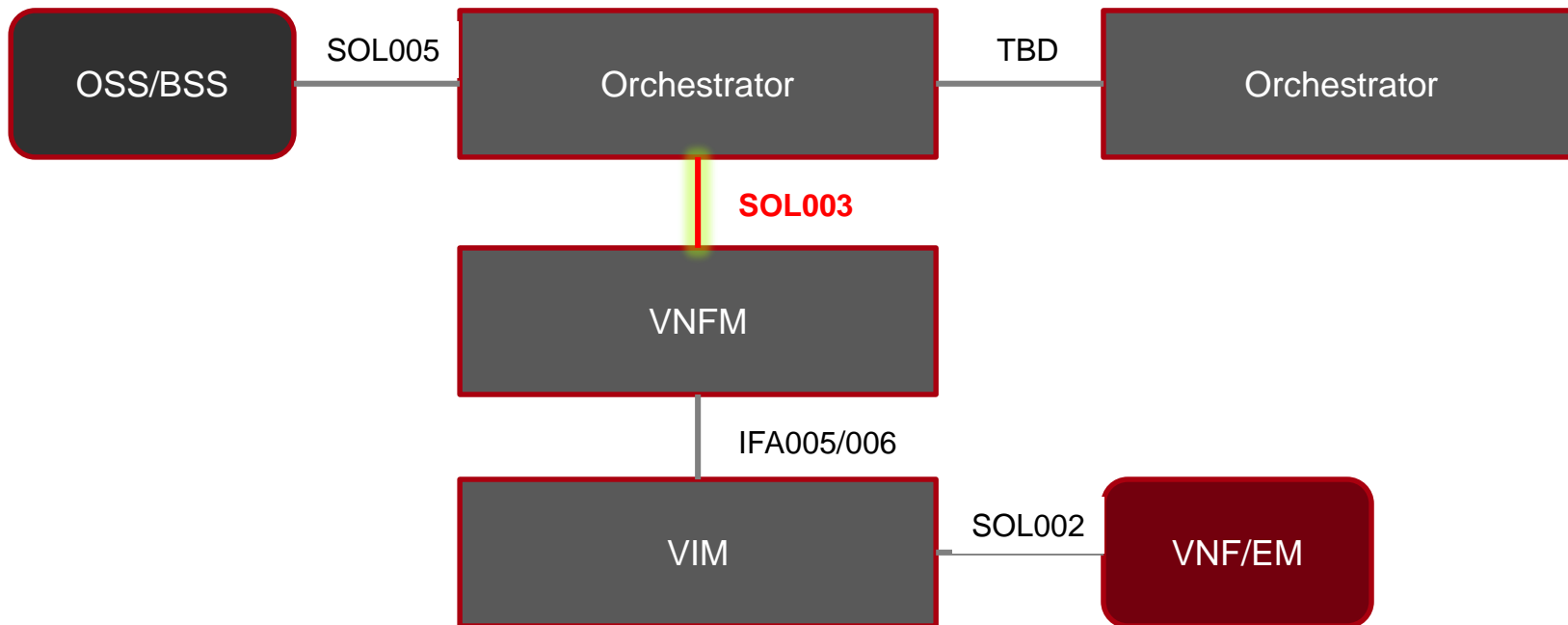
- Started from May 2016
- Stage-3 protocol based on ETSI NFV GS IFA specification



**SOL001** TOSCA based NFV descriptors specification

**SOL004** VNF Package specification

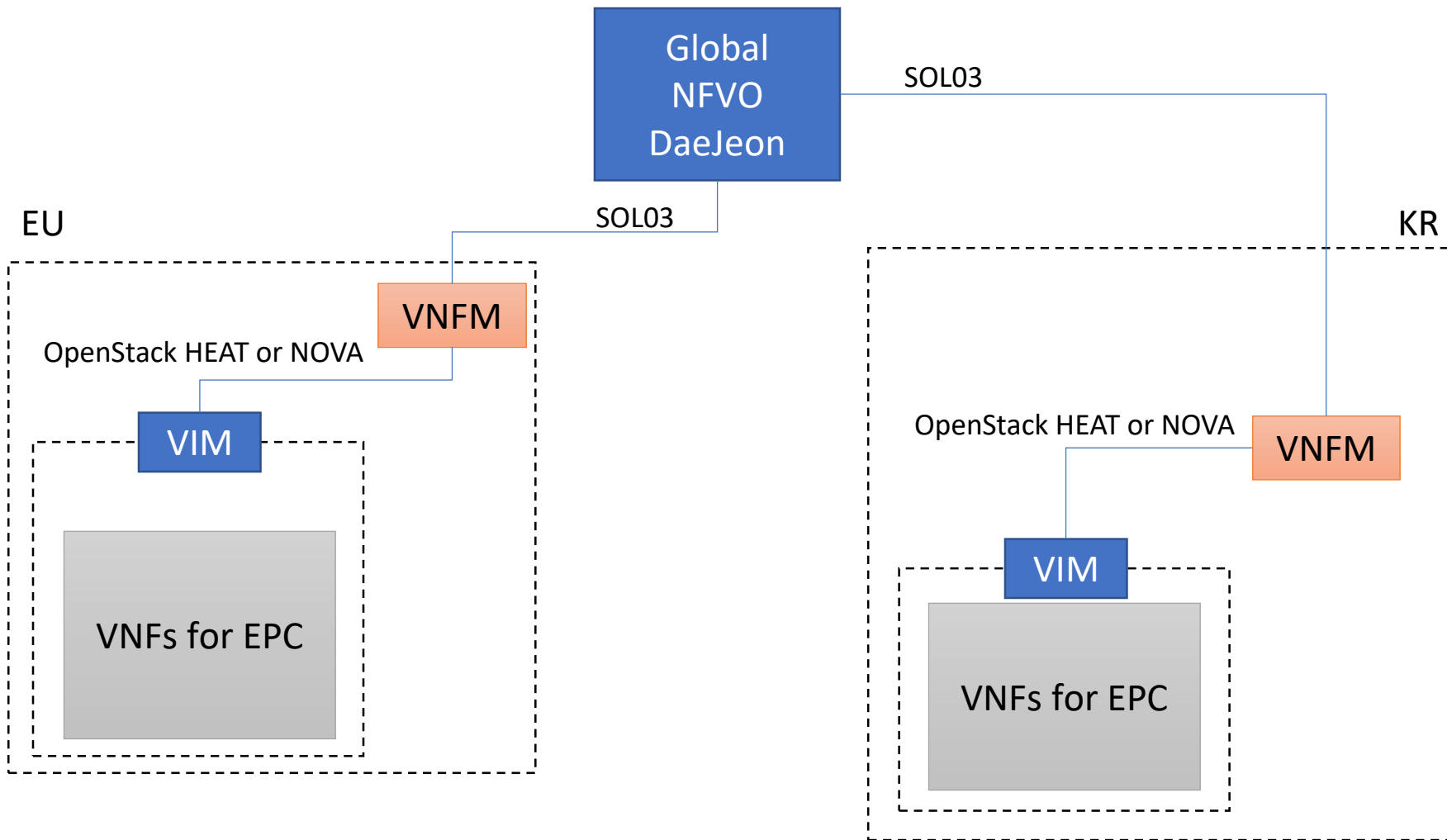
- MANO IOP(Interoperability) for 5GCHAMPION



M-IOP-5GCHAMPION Candidate Interfaces



# MANO Interoperability Testing





- **APIs for IOP**
  - **VNF LCM interface**
    - Create VNF instance resource (VNF Identifier)
    - Delete VNF instance resource (VNF Identifier)
    - Instantiate VNF
    - Terminate VNF
    - Operate VNF
    - Scale VNF
    - Create Subscription
    - Create Notification
  - **VNF Indicator interface**
    - Get Indicator Value
  - **VNF LC operation Granting interface**
    - Grant Lifecycle Operation
  - **VNF Package Management interface**
    - Read VNFD

- **VNF packages for IOP**

- **Contains**

- the VNF descriptor (VNFD) that defines metadata for package onboarding and VNF management
- the software images needed to run the VNF
- optional additional files to manage the VNF (e.g. scripts, vendor-specific files etc.)

- **Is digitally signed and delivered by the VNF provider as a whole**

- **Is immutable (protected from modification)**

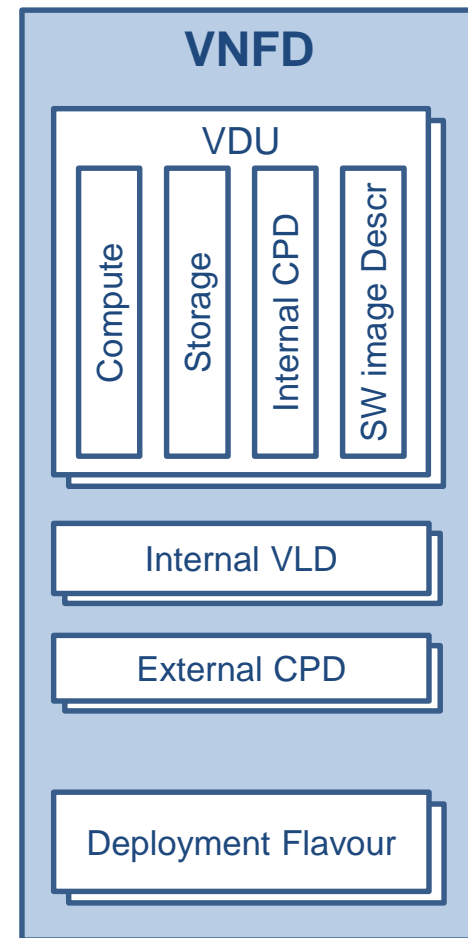
- **Is stored in a repository by the NFVO**

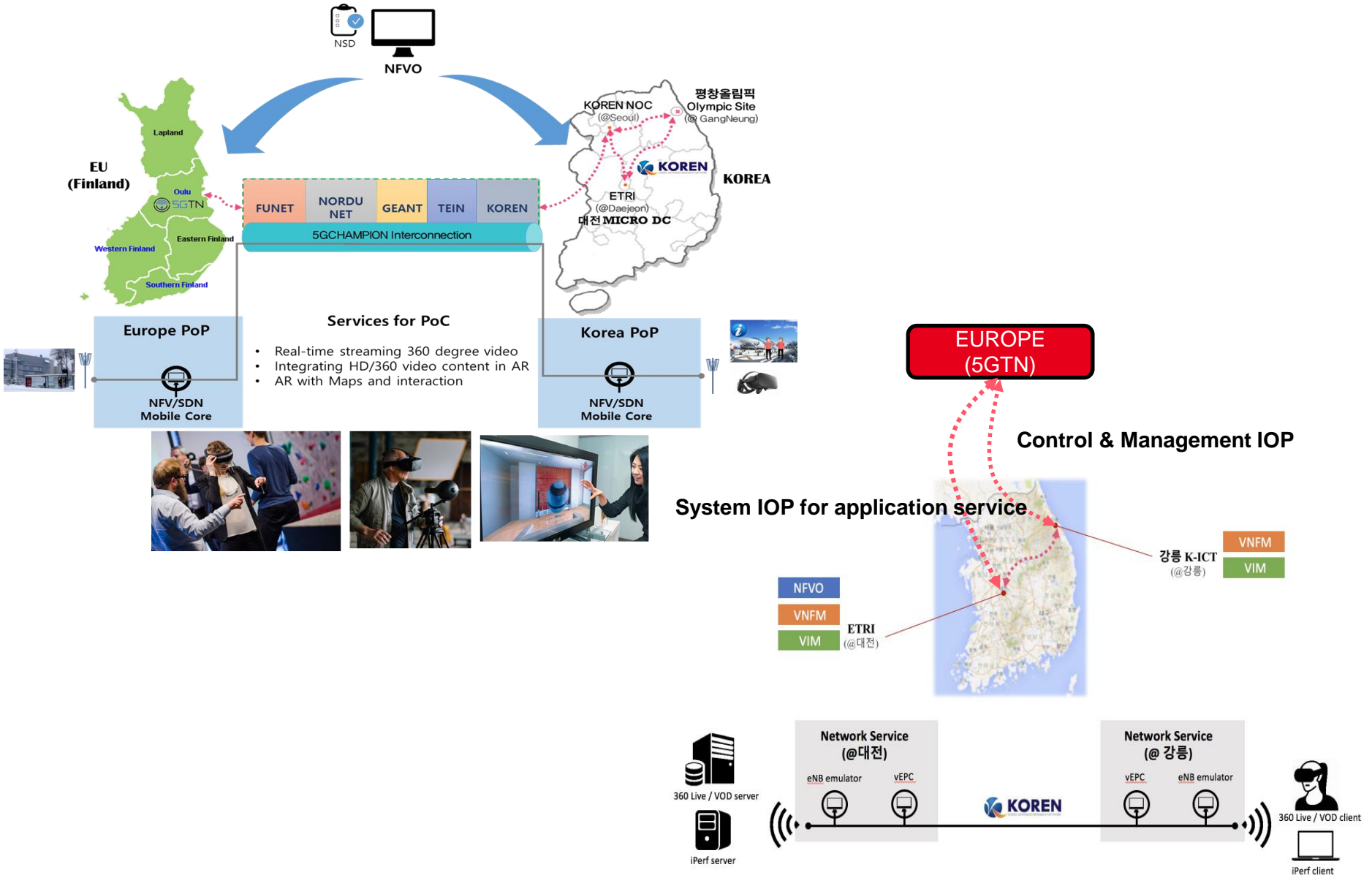
- **Can be accessed by VNFM**



## • VNFD for IOP

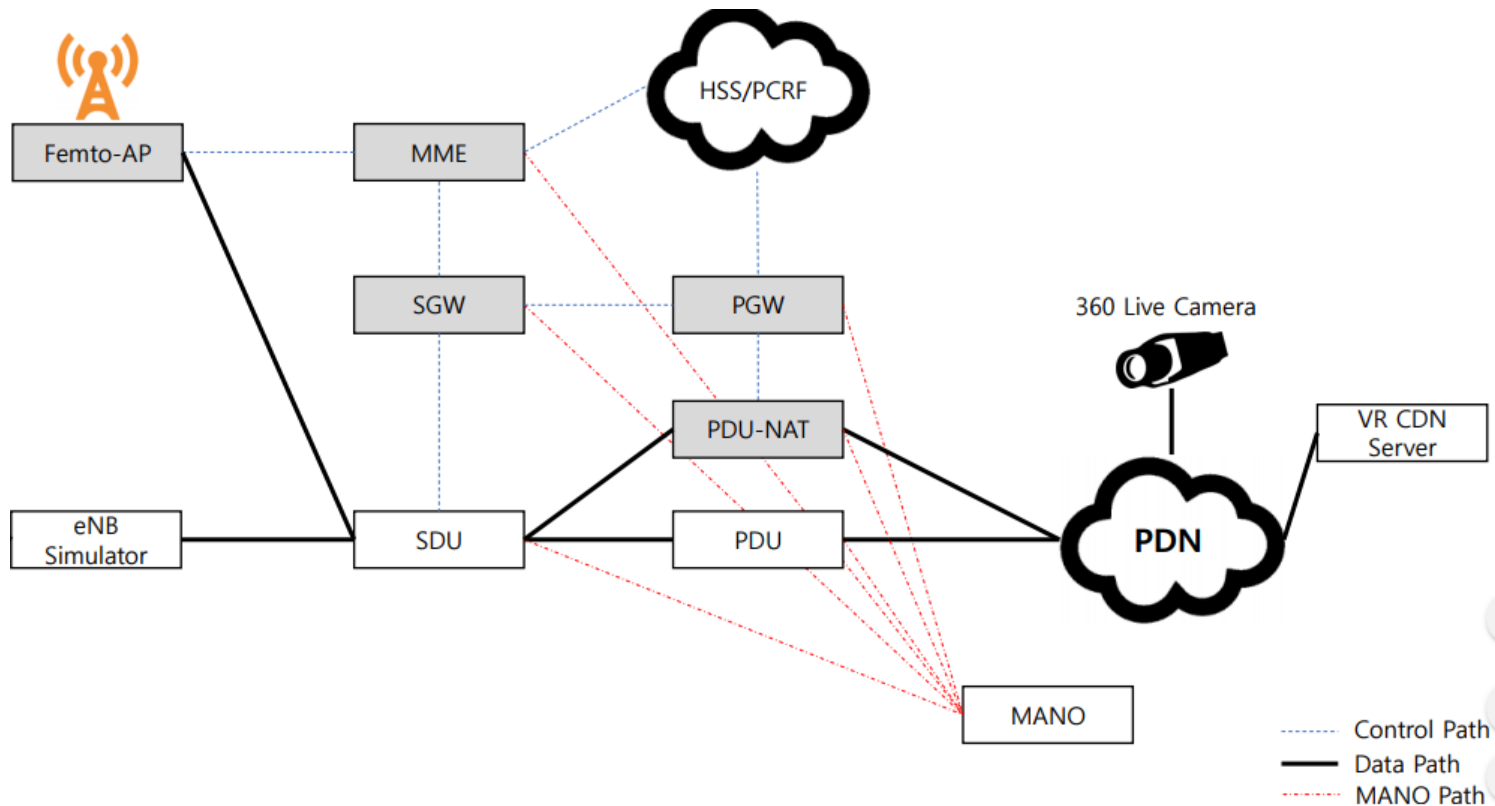
- Defines VNF properties, such as
  - Resources needed (amount and type of Virtual Compute, Storage, Networking)
  - Software metadata
  - Connectivity
    - External Connection Points (described via CP Descriptors, CPD).
    - Internal Virtual Links (described via VL Descriptors, VLD)
    - Internal Connection Points (described via CP Descriptors, CPD)
  - Lifecycle management behavior (e.g. scaling, instantiation)
  - Supported lifecycle management operations, and their configuration
  - Supported VNF specific parameters
  - Affinity / anti-affinity rules
- Defines deployment flavours (size-bounded deployment configurations, e.g. related to capacity)



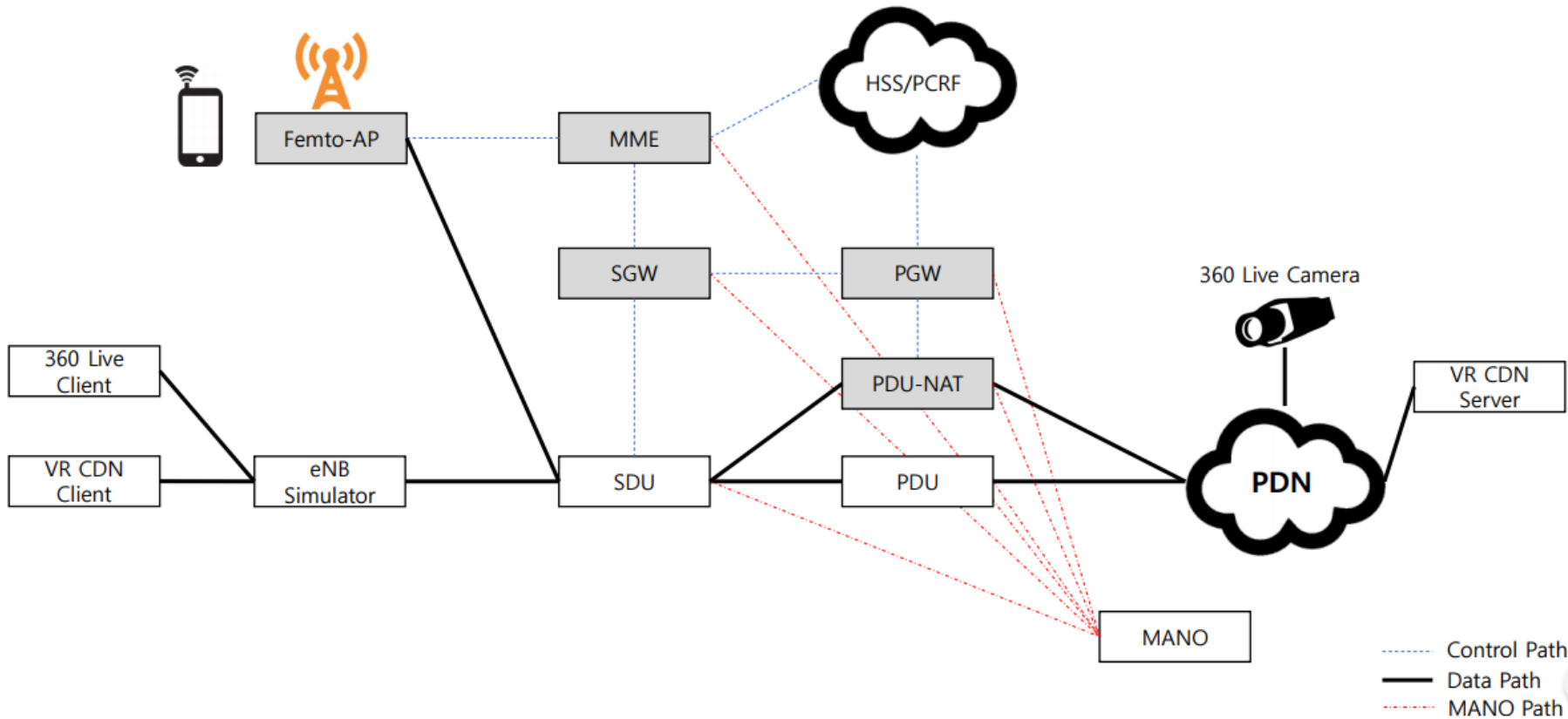


## 1. vEPC instantiated

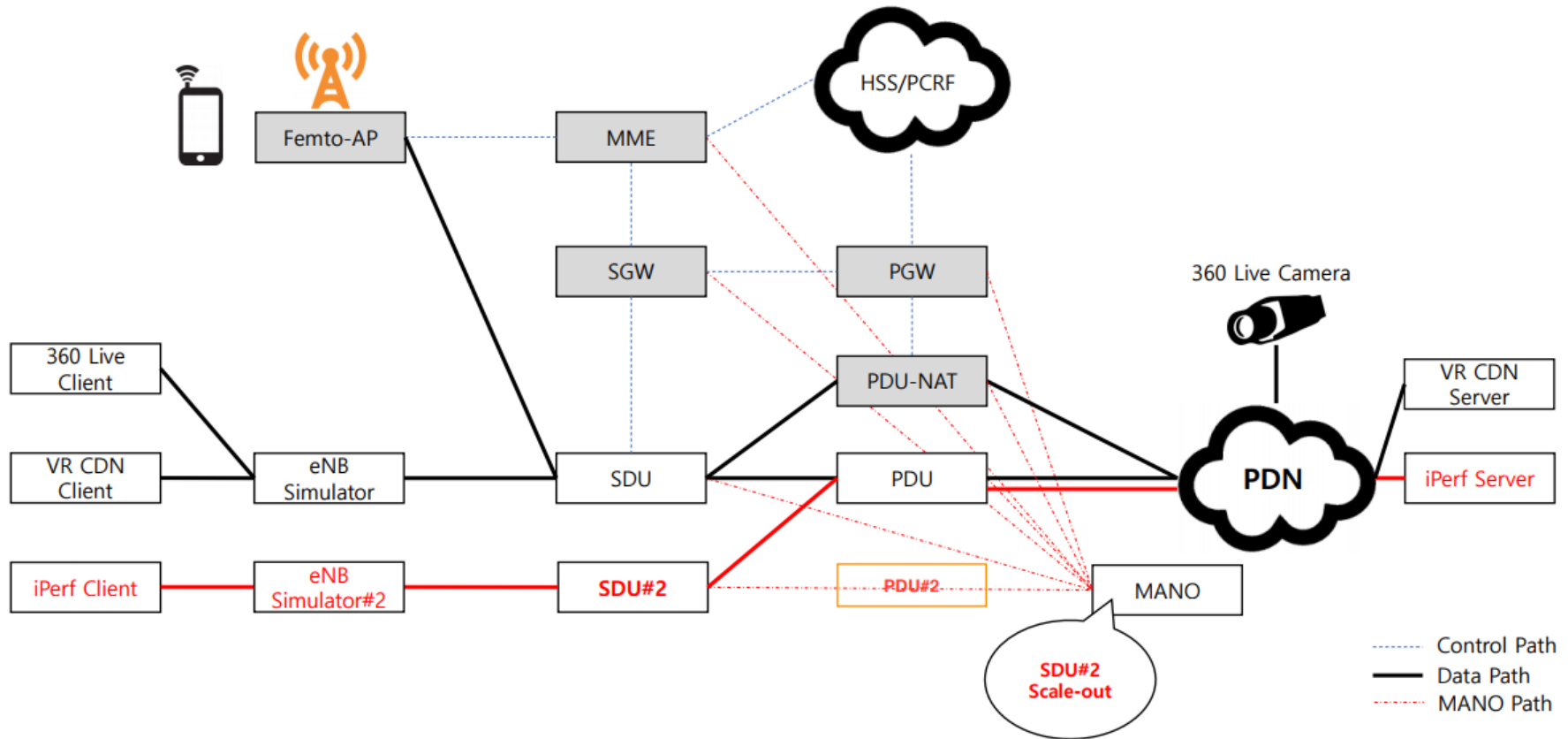
- Femto AP for LTE service
- eNB Simulator for mobile core performance test



## 2. Events from 360 Live & CDN service



## 3. Auto-scaling operation



# Thank You

choits@etri.re.kr

