A 5G service chain on Kubernetes

How to leverage Kubernetes to create a high-performance packet processing pipeline to deliver a 5G service-oriented network

tieto

Introduction

So who here knows about 5G and or Cloud Native?

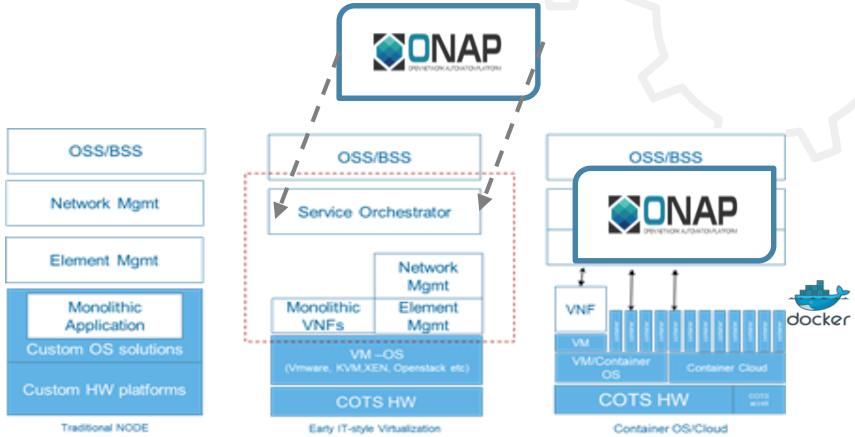
tieto

5G Services - Evolution

5G Evolution in Cloud

Microservices & Containers→ New VNFs

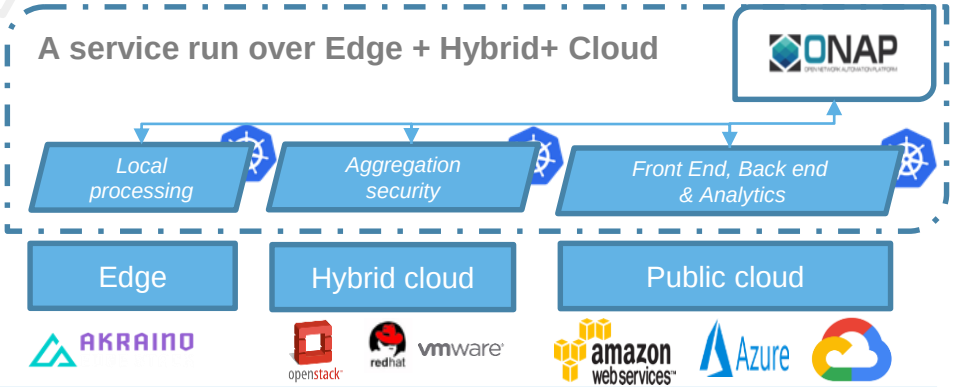
Generic VNFM & Direct mapping from ONAP



Multicloud Evolution

Containerization, new APIs and legacy reuse pushes Service Orchestration towards using **multicloud service delivery**

- Services will be delivered by intelligently using all available cloud resources
- Specialized resources (such as Edge computing or AIML)
- **Multicloud Delivery →**



tieto

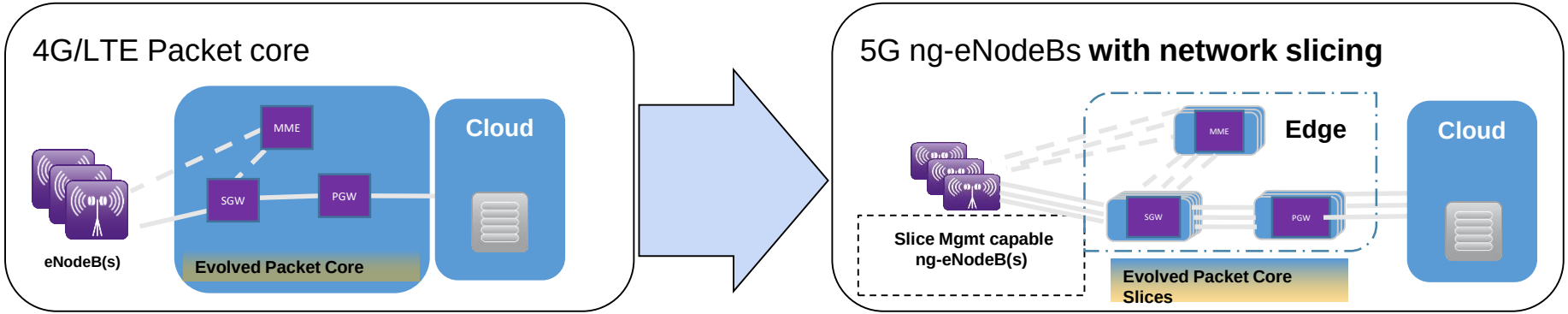
5G Edge Evolution

Current EPC Limitations
Single control logic, with limited multicloud flexibility

- Fixed maximum number of users per instance
- Scaling within POD/Instance and eNodeB(s)
- Dependency on NMS/EM capabilities
- QoS and HA working on large clusters of users

EPC+ with Network slicing ng-eNodeB
Monoliths are decoupled to smaller EPC services that can be placed on RAN Edge (or where relevant)

- Granular scaling of users, Microslices with
- High efficiency, variability of deployment
- QoS on slice or even user level is possible
- Service is decoupled from NMS/EM capabilities



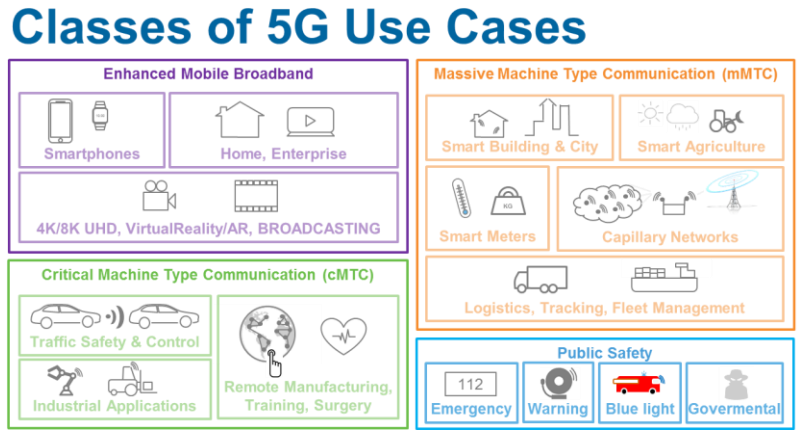
tieto

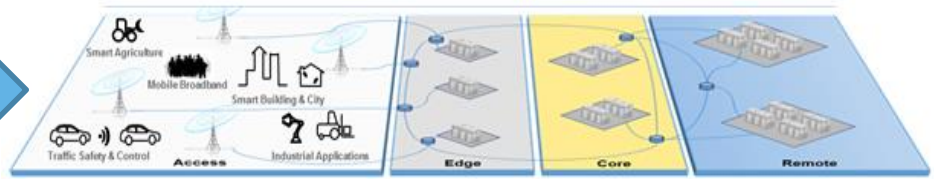
Addressing challenges to support 5G use cases and improve characteristics in cloud

End-to-end QoS for various 5G use case categories
- eMBB,
- mMTC,
- URLLC/cMTC

Fulfilling QoS while addressing demand in
- time,
- location and
- tenancy

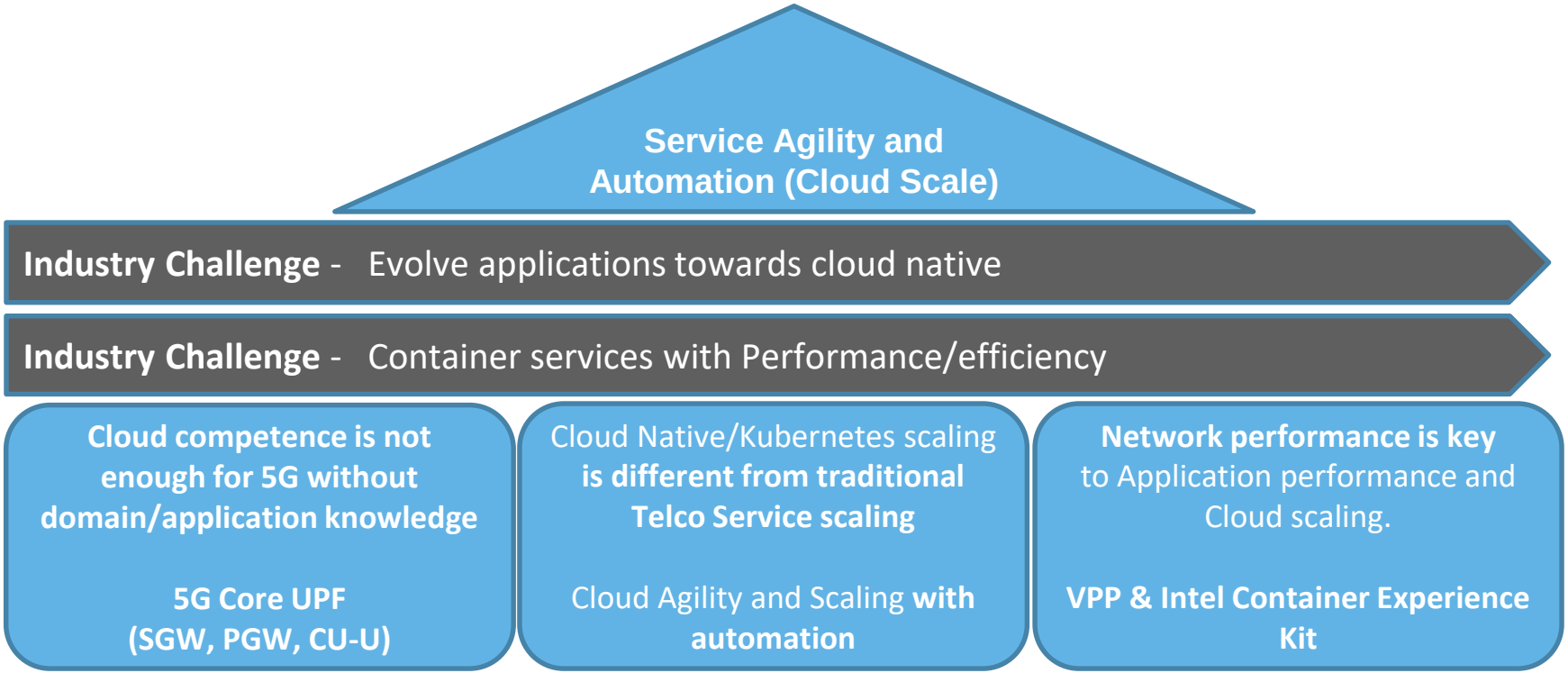
Increased complexity with
- increased network dynamics (on-demand),
- flexible split
- virtualization





tieto

Service Oriented 5G UPF with Kubernetes



Enabling all three requires close access to HW, and SW platforms + Network, execution and opensource competence

Service Oriented 5G UPF with Kubernetes

In this MWC demo we transitioned our VM-based EPC services onto a container-based infrastructure using Intel's Container Experience Kit to evaluate and explore the new stack

- 5G dataplane components running on Kubernetes
  --> **Cloud Native Microservices**
- E2E service stability
- --> **Responsive and Robust Telco Application**
- E2E traffic variability (mimicking real users)
  --> **Ability to test and simulate E2E use-cases**
- Generic microservice and cloud native performance
  --> **Scalable, Highspeed Dataplane and Networking**

Demo Description: 5G UPF with Kubernetes

Traffic for a communication service is generated from large-scale simulation of users in an urban region.
The communication service is realized with chains of services in a service-oriented approach with Kubernetes for scaling and orchestration. A set of technologies is applied to enable this for telecom applications, such as 5G User-Plane Function (UPF).

Technologies Inside

Kubernetes Orchestration

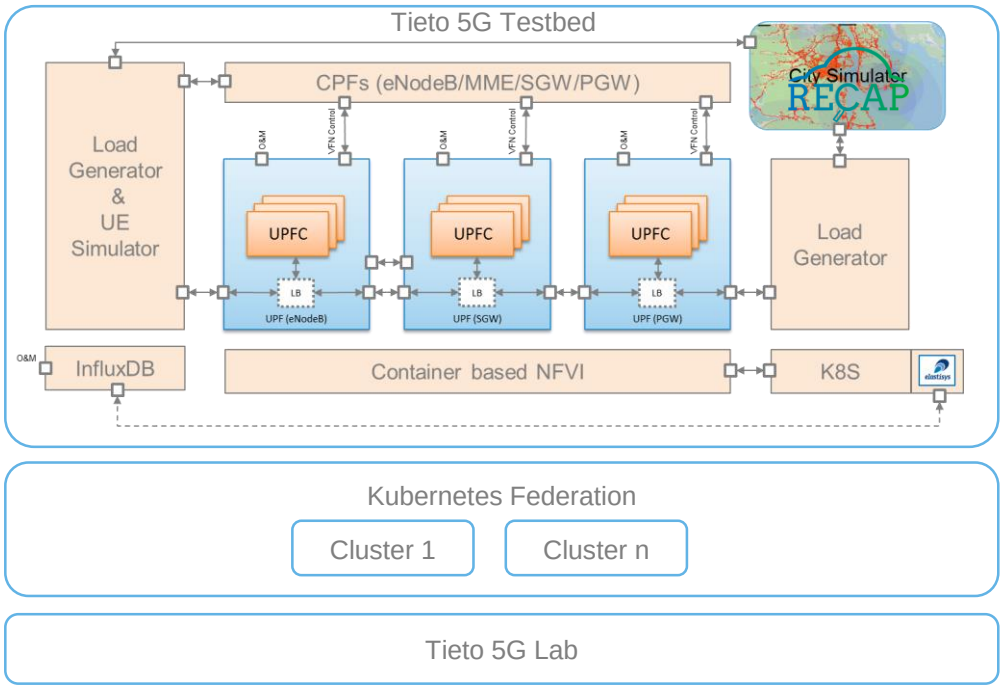
- ❑ Kubernetes scaling
- ❑ Extensions such as IP address assignments

High Performance Packet Processing in Kubernetes

- ❑ Data plane acceleration with DPDK and SR-IOV
- ❑ Vector Packet Processing with FD.io's VPP applied to 5GC and 5G RAN User-Plane Function (UPF)

Intel® Container Experience Kit

- ❑ Kubernetes and bare metal containers for NFV
- ❑ Multiple network interfaces using MULTUS CNI plugin
- ❑ CPU pinning and isolation using Enhanced Platform Awareness (EPA)



MWC19 Barcelona 25-28 February 2019
RECAP
.io The Universal Dataplane
kubernetes
tieto

Why is the needed or useful?

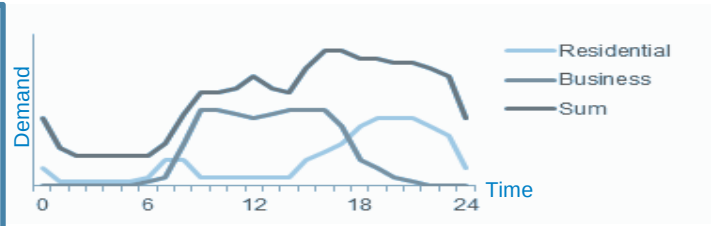
.. le show us some tech!

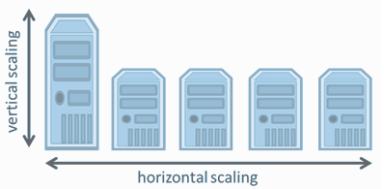
City Simulator : Capable of emulating network load including people

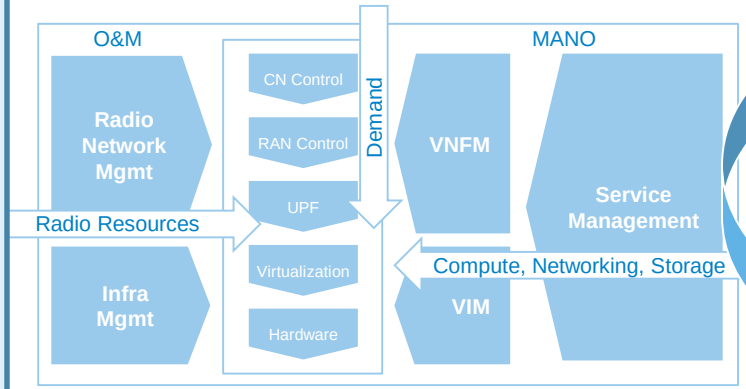
With RECAP: City Simulator we can visualize adaptions to **network dynamics** and set KPIs for various customer use cases & traffic scenarios.

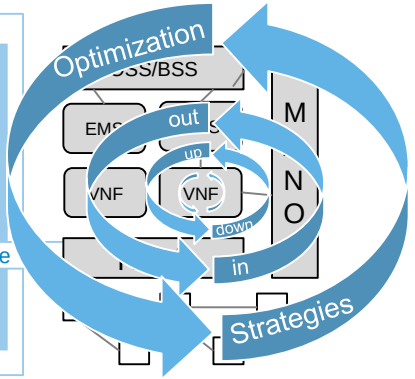
We can from there work on characteristics and enable E2E usecases and achieve high responsiveness, robustness, energy efficiency and high performance

Automated control of application and processing resources leveraging on mix of algorithms for vertical and horizontal scaling at varying workload (Ongoing evaluation)



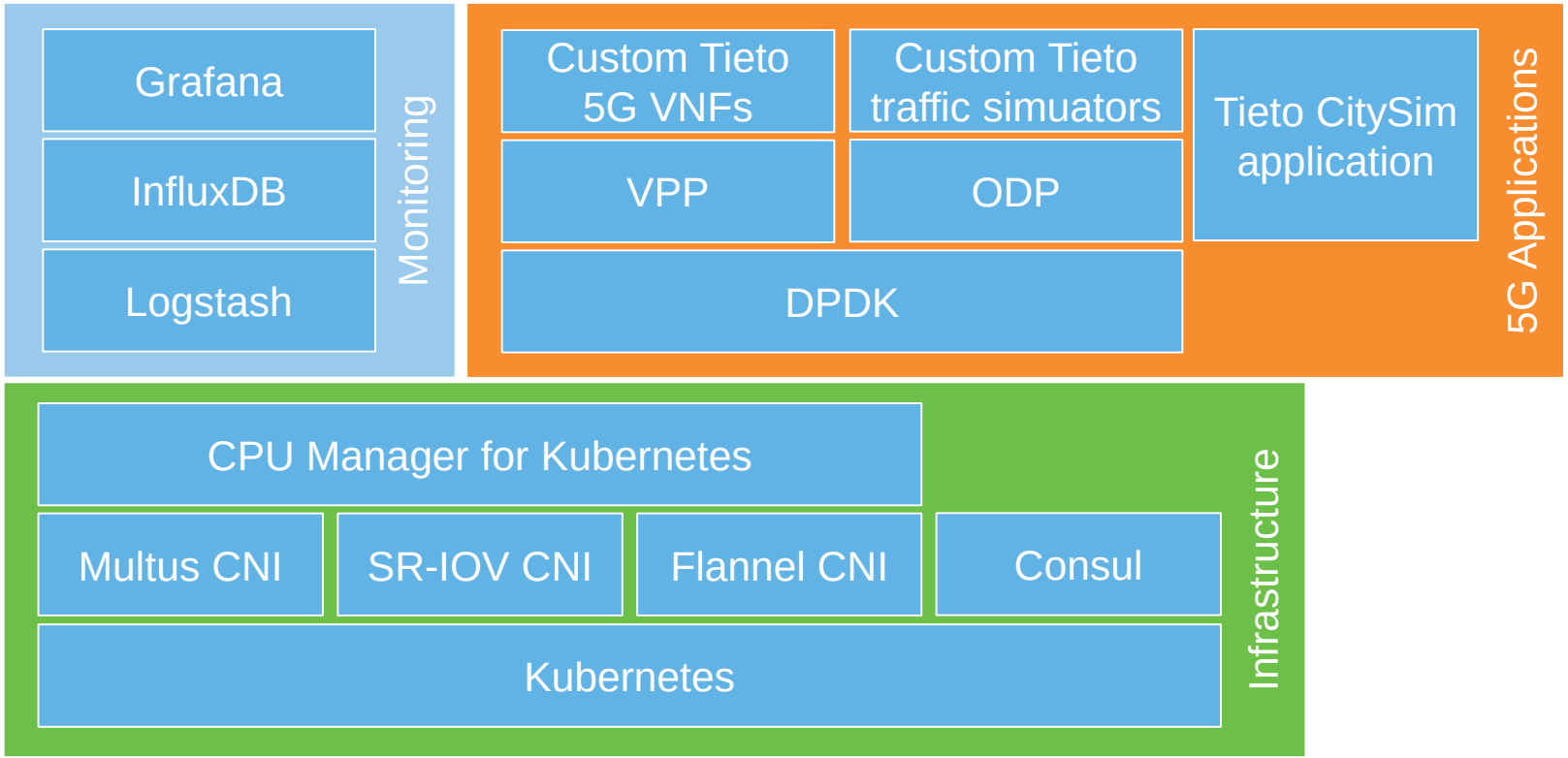




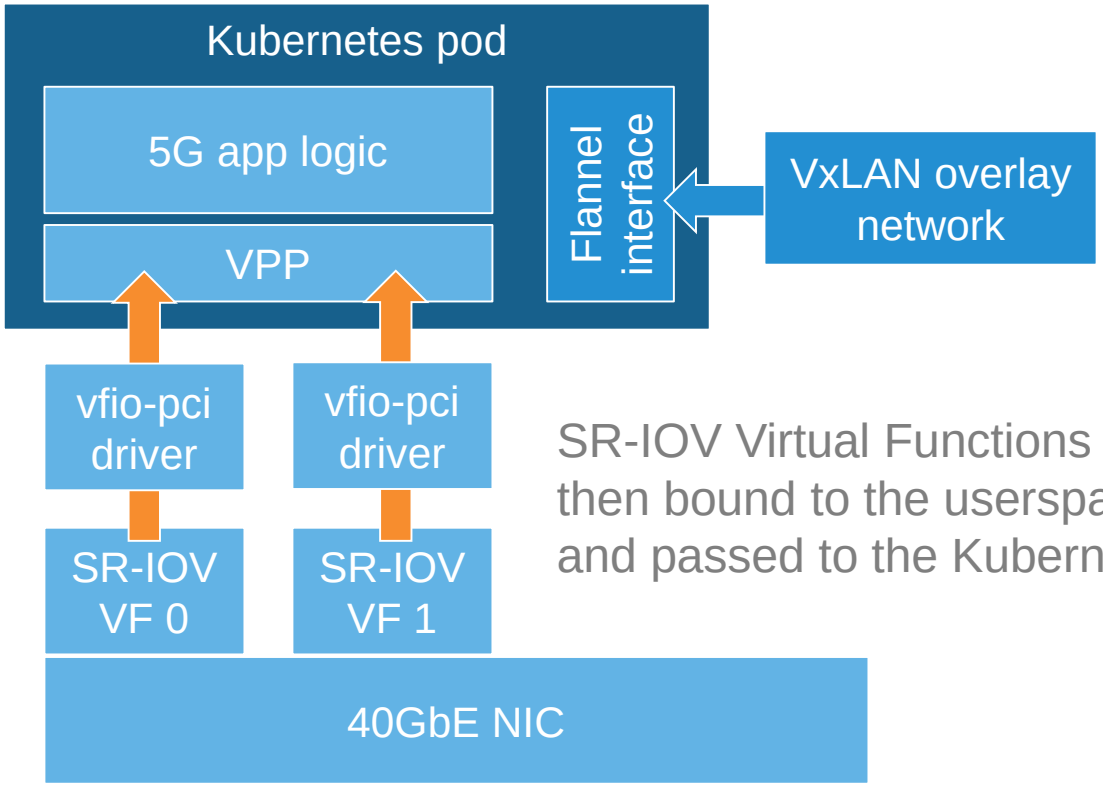


Improve characteristics in cloud and support 5G use cases	Hierarchical resource mgmt in large pools of distributed & clustered VNF instances	Scaling, migration, HA and load balancing

5G UFP demo - Software stack



Network interfaces plumbing

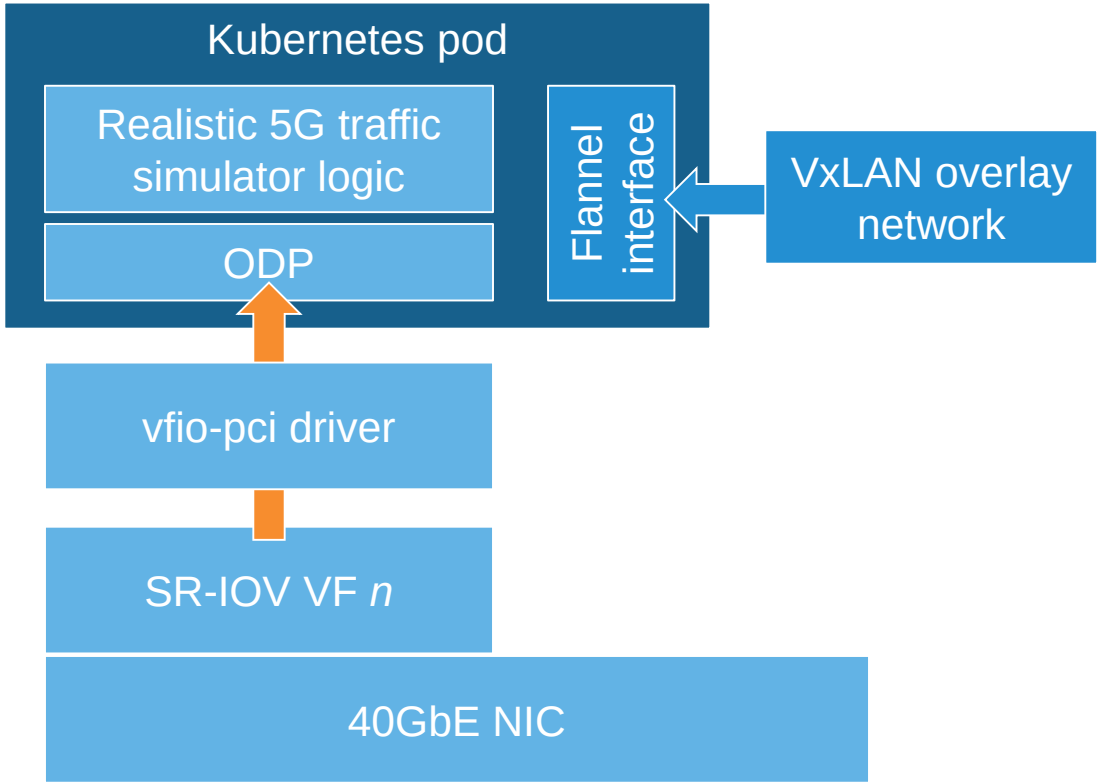


Flannel CNI is used for control-plane communication between apps

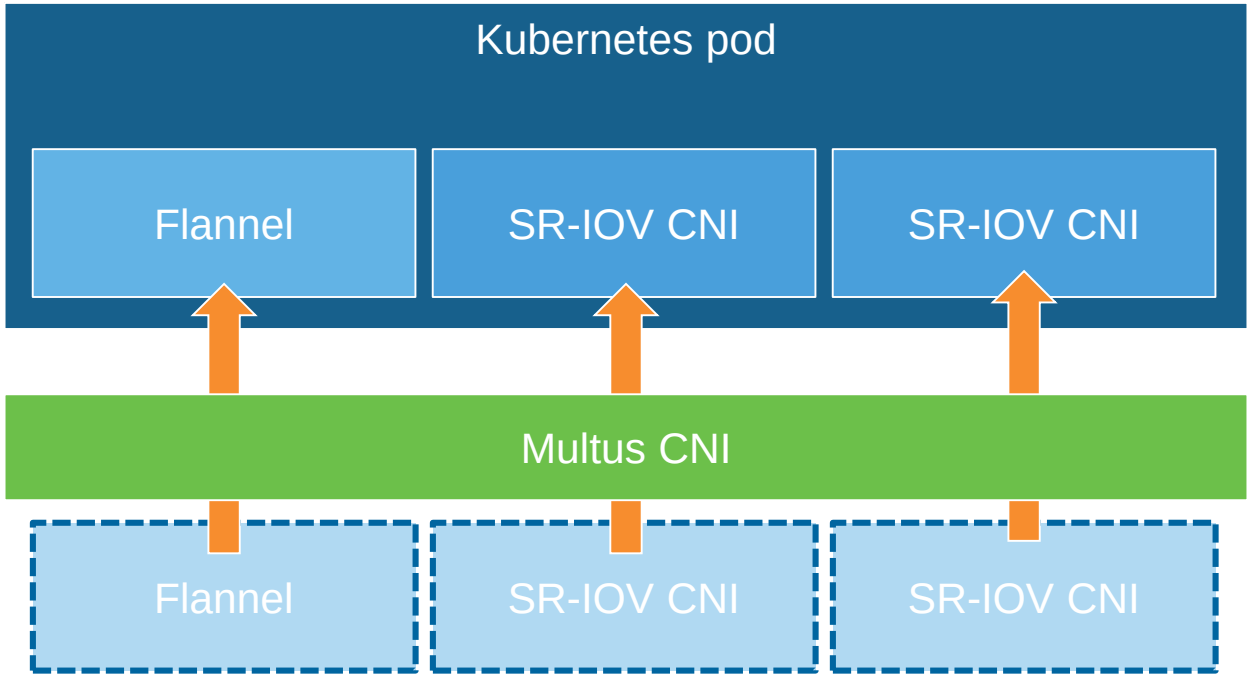
SR-IOV Virtual Functions are allocated dynamically, then bound to the userspace compatible driver and passed to the Kubernetes pod by SR-IOV CNI plugin.

tieto

Network interfaces plumbing – cont.



Network interfaces plumbing – cont.



Multus acts as a middle-man between Kubernetes and CNI plugins to enable suport for using multiple plugins at the same time.

Example Kubernetes app deployment spec*


```
kind: Deployment
metadata:
  name: cnsim
spec: […]
    metadata:
      annotations:
        k8s.v1.cni.cncf.io/networks: cnsim-uldl
    spec:
      containers:
      - command: [ "/opt/bin/cmk isolate", "pool=exclusive", "--no-affinity", "run_cnsim"]
        image: 10.102.58.235:5000/tieto/5g-testbed
        name: cnsim
        resources:
          requests:
            cmk.intel.com/exclusive-cores: "4"
            hugepages-2Mi: 4Gi
```

Dataplane SR-IOV network to be used by this pod

App will run in CMK context, using isolated CPU cores

Request to 4 CPU cores allocated by CMK and 4GB of 2MB hugepages for DPDK app

*a lot of boilerplate and less important settings removed

Example Kubernetes network spec*


```
apiVersion: k8s.cni.cncf.io/v1
kind: NetworkAttachmentDefinition
metadata:
  name: cnsim-uldl
spec:
  config: |
    {
      "name": "cnsim-uldl",
      "type": "sriov",
      "if0name": "cnsim-uldl",
      "dpdk": {
        "kernel_driver": "i40evf",
        "dpdk_driver": "vfio-pci",
        "dpdk_tool": "/usr/src/dpdk/usertools/dpdk-devbind.py"
      },
      "ipam": {
        "subnet": "172.24.2.0/24",
        …
        "gateway": "172.24.2.1"
      }

    }
```

Network attachment definition consumable by Multus CNI plugin

SR-IOV CNI config with both DPDK and IPAM modes enabled

*a lot of boilerplate and less important settings removed

tieto

Other challenges and solutions

CPU Manager for Kubernetes dynamically pins userspace data plane applications to CPU cores.

We had to fork SRIOV CNI plugin to enable IP address management features in DPDK mode.

Service discovery not working (yet) for secondary network interfaces (SR-IOV DPDK in this case) – Consul used as a workaround.

Performance numbers

- Running at ~20 Gbps upstream/downstream with zero packet loss.
- Simulating realistic 5G traffic from ~7000 UE devices.
- 20 Gbps – limited due to HW availability – adding one extra Kubernetes node with dual port NIC should bring further improvements – work in progress.

Not optimized as a product

5G UPF ”Demo” – Tieto Values

Tieto has long LTE and 5G application experience with architecture knowledge to realize and improve complex cloud Native and 5G functions

We can enhance container networking peformance using our semiconductor and dataplane expertise to bring cloud infra together with applications
•DKDP, VPP, OvS, CNI
•SmartNiC, routers and CPU

We can help you with these areas and many more.....